JAERI-M
89-126

制御システム設計・解析プログラム群
(DPACS／J)

1 9 8 9年9月

栗原　研一・本多　光輝*・中村　幸治・木村　豊秋

日本原子力研究所
Japan Atomic Energy Research Institute

JAERI-M レポートは，日本原子力研究所が不定期に公刊している研究報告書です。
入手の問合わせは，日本原子力研究所技術情報部情報資料課（〒319－11茨城県那珂郡東海村）あて，お申しこしください。なお，このほかに財団法人原子力弘済会資料センター（〒319-11茨城県那珂郡東海村日本原子力研究所内）で複写による実費頒布をおこなっております。

JAERI-M reports are issued irregularly.
Inquiries about availability of the reports should be addressed to Information Division, Department of Technical Information, Japan Atomic Energy Research Institute, Tokai-mura, Naka-gun, Ibaraki-ken 319-11, Japan.

編集兼発行　日本原子力研究所
印　　刷　日立高速印刷株式会社

制御システム設計・解析プログラム群
（ＤＰＡＣＳ／Ｊ）

日本原子力研究所那珂研究所ＪＴ－60試験部
栗原　研一・本多　光輝*・中村　幸治+・木村　豊秋

（1989年8月22日受理）

制御系の設計や解析に於いては，制御対象のモデル化及び制御システムの最適化のために数多くの試行錯誤を強いられるのが普通である。ＤＰＡＣＳ／Ｊは，この複雑な制御系設計・解析を計算機支援の下で効率良く実施するためのプログラム群であり，トカマク・プラズマの放電制御のために導入・開発された。

本報告は，制御系設計・解析プログラム群ＤＰＡＣＳ／Ｊについて，理論概要，処理機能，使用方法を述べたものであり，一般の利用者は，本報告を利用手引として用いることができる。

那珂研究所：〒311-01 茨城県那珂郡那珂町大字向山801-1
＋　核融合研究部
＊　業務協力員，㈱アイ・ビー・エス・データセンター

Program Package for Control System Design and Analysis (DPACS/J)

Kenichi KURIHARA, Mitsuteru HONDA*, Yukiharu NAKAMURA+ and Toyoaki KIMURA

Department of JT-60 Facility
Naka Fusion Research Establishment
Japan Atomic Energy Research Institute
Naka-machi, Naka-gun, Ibaraki-ken

(Received August 22, 1989)

In order to build an optimal control system, various methods should be applied to control system design and analysis, and many "cut and try" procedures should be taken for persuading optimality.

The useful program package (DPACS/J), which will make it easy to design, and analyze control systems, has been installed in the JAERI computer. This is the manual of DPACS/J.

Keywords : Control Systems, Tokamak Plasma, Optimal Control, DPACS/J

+ Department of Thermonuclear Fusion Research

* On leave from I.B.S. Data Center, Ltd.

目　　次

1. はじめに …… 1
  1.1 動機と位置付け …… 2
  1.2 開発経緯 …… 2
2. プログラム群の構成と機能概要 …… 5
3. 制御系設計・解析パッケージ（ＤＰＡＣＳ／Ｊ）の理論概要・処理機能・利用編 …… 8
  3.1 ＤＰＡＣＳ／Ｊの主な特徴と記号の説明 …… 8
  3.2 ＤＰＡＣＳ／Ｊの構成 …… 10
  3.3 制御系の表現形式 …… 15
  3.4 ＤＰＡＣＳ／Ｊに於けるデータのアクセスと管理体系 …… 18
  3.5 ＤＰＡＣＳ／Ｊに共通的な事柄と一般的な利用上の注意事項 …… 26
  3.6 ＤＰＡＣＳサブシステムの理論概要・処理機能・利用例 …… 35
  3.7 ＤＰＥＸサブシステムの理論概要・処理機能・利用例 …… 220
  3.8 ＳＹＳサブシステムの理論概要・処理機能・利用例 …… 241
  3.9 ＴＳサブシステムの理論概要・処理機能・利用例 …… 260
  3.10 ＭＡＴサブシステムの理論概要・処理機能・利用例 …… 316
4. ＤＰＡＣＳ／Ｊバッチ処理版（ＤＰＡＣＳＢ）の機能・利用編 …… 352
  4.1 機能 …… 352
  4.2 利用例 …… 353
  4.3 利用上の注意事項 …… 356
5. データ編集・登録プログラム（ＥＤＩＴＪ）の機能・利用編 …… 357
  5.1 機能 …… 357
  5.2 利用例 …… 359
6. 導入編 …… 366
  6.1 必要となるファイル …… 366
  6.2 各パッケージの動作環境 …… 376
7. 結言 …… 377
謝　辞 …… 377
参考文献 …… 378
付録　コマンド入力形式一覧表，コマンド索引表 …… 380

Contents

1. Introduction 1
1.1 Motivation 2
1.2 History of the Development 2
2. Configuration of the Programs in the Package 5
3. Descriptions on Theoretical and Operational Aspects of DPACS/J 8
3.1 Features and Notations 8
3.2 Configuration of the Programs 10
3.3 Expressions of Control Systems 15
3.4 Data Management 18
3.5 Common Procedures and Notations in Operations 26
3.6 Theories and Operations in the DPACS Subsystem 35
3.7 Theories and Operations in the DPEX Subsystem 220
3.8 Theories and Operations in the SYS Subsystem 241
3.9 Theories and Operations in the TS Subsystem 260
3.10 Theories and Operations in the MAT Subsystem 316
4. DPACS/J Batch Job Version (DPACSB) 352
4.1 Explanations of the Program 352
4.2 Operations 353
4.3 Notations 356
5. JT-60 Data Edition Program (EDITJ) 357
5.1 Explanations of the Program 357
5.2 Operations 359
6. Installation 366
6.1 Indispensable Files 366
6.2 Environment for the Execution 376
7. Concluding Remarks 377
Acknowledgments 377
References 378
Appendix A Table of the Command Input Formulae and a Command Index 380

1. はじめに

究極のエネルギー源と称されて久しい核融合発電は，その核融合反応そのものを地球上で実現するために数種のアプローチが試みられている。その中で原研では，トカマクと呼ばれるドーナッツ型の容器の中にプラズマを高温，高密度で閉じ込めることで核融合反応を起こす装置を用いたアプローチを推進している。

このトカマク・プラズマの動特性を解明し，制御するという観点で理論的・実験的にこれまでも様々なアプローチで研究が進められている。トカマク・プラズマは，表現できるとしてもおそらく，多数のパラメータ（確率分布関数も含めた）に関する3次元空間（または4次元時空間）中の連立偏微分方程式を，境界面形状及び境界条件が時間変化するという条件の元で解く形式であると思われる。

今，「このトカマク・プラズマを制御する」という観点で検討する。トカマク・プラズマに対して作用を与えるアクチュエータの数は，元々工学的に可能な範囲であり，高々10個程度と予想される。従って，これらのアクチュエータを使い可制御な物理パラメータを操ることで，トカマク・プラズマを実用上問題無く制御できるかどうか，というのが重要なポイントとなる。しかしこの問題は，無数の数学的，物理的，工学的な問題にブレークダウンできる程の内容を持っていると考えられる。これらを一般論として進めることは我々の能力を越えるので，トカマク・プラズマの対象として臨界プラズマ試験装置（ＪＴ－60）の場合を想定して，我々が考えている問題解決の手順を以下に述べる。まず，ＪＴ－60のアクチュエータとプラズマとの関係を整理する。この整理には，最終的にはプラズマ物理学上の未解決な問題（例えば，ディスラプション制御やＨ－モード現象の制御）にまで発展するとしても，まずは何らかの仮定のもとでの単純化を試みることが必要であると考えている。整理の仕方は，理論や数値計算等の理論解析アプローチを基本にして，実験結果の裏付けを図ることが必要になると思われる。理論的に難しい部分は，必要に応じて実験を行い実験式を得ることが想定される。

次に，プラズマの各種状態量のうち観測できるものを明確にする。この“観測できる”という定義は，制御という観点から信頼性を充分持っているもの，と解釈される。また，観測される生信号を観測量に変換する計算も，明確に決定されることが必要である。

最後は，観測データを取り込みアクチュエータに指令を出力する部分，及び指令をアクチュエータがプラズマにどう出力するかという部分，を明確にする。前者と後者は分離できるものとできないものとがあるのであえて分けて書いた。後者は，装置のハードウェアが決まれば明確になるものである。前者は，前述した2つの手順の結果を検討し，最適な制御システムを構築することが必要となる。

これらの作業を通じて，“実用上問題の無い”レベルにまでトカマク・プラズマを生成し，制御できれば良く，特にＪＴ－60にだけしか適用できないような装置固有の部分がもしあればその方法論に立ち入ってまでも明確にすることが重要である。

以上がトカマク・プラズマを制御するための想定している手順である。勿論，この極めて大

雑把なものだけで充分とは考えておらず，実際，これを細かくブレークダウンした実作業の手順に焼き直して開発作業を行っていく必要があるが，現在全体を通しての構築作業の途上にある。

1.1 動機と位置付け

再度，問題解決のための手順を箇条書きすると以下のようになる。

Ⅰ．アクチュエータとプラズマとの相互関係の明確化

Ⅱ．プラズマの観測量の明確化

Ⅲ．アクチュエータの動特性の明確化

Ⅳ．制御アルゴリズムの決定

ＪＴ－60の制御系設計の初期において，Ⅰを単純化して取扱い，Ⅱ，Ⅲを無視し，Ⅳを古典制御理論を用いて考察した報告[1]を始め，各種の検討が行われた。その中で，小方ら[2]が提案した，多変数現代制御理論を用いたトカマク・プラズマの制御は，方法論的に多変数の取扱いを可能にする点，及び合理的な最適制御則の決定を可能にするという点，可制御性・可観測性という構造に関する考察が容易であるという点等注目すべき点が多く存在した。しかも，多変数状態方程式をベースとする各種アルゴリズムは，計算機の進歩も手伝って，R.E.Kalmanら[3]が既にコード化を行っている事実を知った。これらの事実は，多変数の扱いを不可避とされるトカマク・プラズマの制御手法として応用可能と考えられた。一方，理論の前提が，線型システムであるが，線型化により如何なるシステムも本理論体系に組込めるとの認識もあった。このように方法論として，多変数制御理論を採用することは極めて有効であると判断された。

この多変数制御理論がトカマク・プラズマの制御という作業に占める役割は，前述のⅠ～ⅣのうちのⅣの部分のみである。Ⅳの部分に合理的（定量的という意味も含めて）な方法論という意味で完備していると判断された。システムが非線型である上に，線型化がうまく行かない様な場合を除けば，見通しとしては相当に良い方法の一つであると思われる。勿論，制御系設計の作業の全てが，この理論の導入で置き換えられる訳ではないから，Ⅰ～Ⅲの作業は相変わらず推進すべき作業として位置付けられるべきものと考えている。

我々は，このⅣの部分の方法として，多変数制御理論を取り入れることを前提に，昭和54年（1979年）からコードの作成に着手した。その開発経緯を次節で述べることにする。

1.2 開発経緯

今までに行ってきた開発，特にコード開発について以下に述べて行くことにする。

昭和54年（1979年）小方らにより最適制御理論に基づくプラズマ制御についての提案が行われた。これを受けて，最適レギュレータのゲインの決め方についての検討が進められた。

昭和57年（1982年）プラズマ制御コードとしての体系立った開発・整備についての検討を行い，合わせて理論についても詳細検討が開始された。この結果が昭和58年（1983年）3月に「プラズマ制御コード開発に関する概念設計書」[4]にまとめられた。

昭和58年（1983年）4月に上記概念設計に基づく制御コードの開発を開始しようとした時，

殆ど同様のコードが東京工業大学制御工学科古田勝久教授の研究室において開発されている[5]ことを知るに至る。R.E.Kalmanの作成したコード[3]のマニュアルも既にその当時入手していたが，そのコードの内容を完全に包含するものであることが分かり，検討の結果そのコードを導入することに決定した。古田教授にこちらの事情を話し協力を依頼したところ即座に快諾され，プログラムソース及びマニュアルを提供された。これにより当初計画していたコード開発は，提供されたコードを原研大型計算機の環境下で動作可能にするという作業に置き換わった。そのコードはＤＰＡＣＳ／Ｆ（Design Package for Control System／Furuta Laboratory）と呼ばれているコードである。

ＤＰＡＣＳ／Ｆは，もともとＮＯＶＡというＤＥＣ系ミニコンピュータのＦＯＲＴＲＡＮ言語により記述された，約31,000有効ステップのコードであった。このプログラムそのままでは，大型計算機上で動作不可能であったため，昭和58年（1983年）6月からプログラム変換作業を開始した。まず，大型計算機（当時：ＦＡＣＯＭ　Ｍ－380）上のＦＯＲＴＲＡＮ77言語とＮＯＶＡ　ＦＯＲＴＲＡＮとの違いと変換の具体的な作業の網羅から始まった。この作業は，文法上の違いを明確にした後，次のような21項目の変換方法に分類した。①注釈の書き方　②定数　③文字の取扱い　④ＯＶＥＲＬＡＹ文　⑤ＣＯＭＰＩＬＥＲ　ＮＯＳＴＡＣＫ　文　⑥ＣＯＭＰＩＬＥＲ　ＤＯＵＢＬＥ　ＰＲＥＣＩＳＩＯＮ　文　⑦型宣言文　⑧ PARAMETER文　⑨無条件ＧＯ　ＴＯ文　⑩非正規リターン　⑪ＰＡＵＳＥ文，ＳＴＯＰ文　⑫算術式の評価　⑬代入文　⑭入出力並び　⑮書式無し入出力文　⑯バイナリデータの入出力　⑰欄記述子　⑱ビット処理　⑲組込み関数　⑳システムルーチン　㉑会話型入出力文。また，図形出力関係のサブルーチン，ＯＳに依存するサブルーチンなどは一つ一つ全面的に見直しを行った。そして，昭和58年（1983年）10月に作業報告書をまとめた。これを受けて，同年10月から昭和59年（1984年）3月まで変換の実作業を行った。これにより一応の変換は終了した。しかし，かなりのコード変換時のプログラム・バグは，取り除かれていない状態であるため実運用に供するには未だ充分整備されているとは言い難かった。また，取り扱える次元数も，必ずしも充分とは言えず，ベクトルの要素数も20が限度であった。そこで，バグの修正も兼ねて次元拡張作業を昭和59年（1984年）5月から8月にかけて行った。この結果，システム同定に関する部分を除いて，50要素ベクトル取扱い可能なまでの次元拡張が完了した。

さらに，昭和60年（1985年）1月から3月にかけて，システム同定に関する部分の次元拡張を行った。この段階で，総有効ステップ約35,000のコードになり，端末の使用領域も2.5 MByteを必要とするまでになった。

一方，ＪＴ－60は，昭和60年（1985年）4月にファーストプラズマの着火に成功するに至り，いよいよ実験データの解析が開始される段階に進んだ。ところがデータ解析をする大型計算機に実験データを持ち込む方法が充分に整備されないまま実験に入ったため，データ解析上の支障が出た。そこでデータの解析を円滑にするための汎用のＪＴ－60実験データベース・システムの整備作業に取り掛かった[6]。また，これと並行してＤＰＡＣＳで実験データが利用できるようにするため，昭和60年（1985年）10月に，ＪＴ－60実験データ編集コードの作成に入り，翌昭和61年（1986年）2月に完成した。また，ＪＴ－60実験データベース・システムも昭和61年（1986年）4月から運用を開始した。

実際の利用を開始したのは，昭和61年（1986年）5月からであった。使用していくにつれ，機能上やマンマシン・インターフェースの面で改良すべき点があることが分かってきた。第1番目は，グラフィック・ディスプレイの整備によるカラーグラフ表示であり，第2番目は，大型計算機の管理下で端末オペレーションを行うと当然CPU時間の制約が生じて来るが，特にシステム同定プログラムの場合，次元がそれ程大きくなくても計算終了前に打ち切られてしまうという問題であった。第3番目は，計算機の応答時間が遅いという点である。第1番目は，出力装置へのインターフェース・ソフトウェアの問題であり，その改良により対応した。第2番目は，バッチ処理が可能なように，データの入力装置を端末入力からカード入力形式にすることで対応した。この2つの作業を昭和61年（1986年）11月から翌昭和62年（1987年）2月にかけて行った。

さらに，第3番目の応答時間の短縮については，中間的に発生するファイルの入出力を減らし，不要な印字を削除する等のマンマシン部分の全面的な見直しを実施した。この作業は，昭和62年（1987年）5月から12月にかけて行われた。この間，演算内容の追加や，全体のバグ取りを目的としたテストランも合わせて実施した。この間も，DPACS／Jを利用してデータ解析を行ってきており，その際便利と思われる機能追加などを行った。それらにより，プログラム群としての初期導入作業が一段落したと判断されることから，DPACS／J（JはJAERIの頭文字）と名付け，本格的にマニュアルの形でまとめることに至ったものである。

2. プログラム群の構成と機能概要

プログラム群は，大きく次のようなサブシステムに分かれている。

① ＤＰＡＣＳ／Ｊの中心的なツールで各種制御演算を実行するＤＰＡＣＳ

② ＤＰＡＣＳの拡張機能を有するＤＰＥＸ（DPACS Extension）

③ システム・データ特性を解析するＳＹＳ

④ 時系列演算を行うＴＳ

⑤ 行列演算を行うＭＡＴ

⑥ ＪＴ－60実験データベースのデータまたはニコレ社製のオシロスコープから出力されるフォーマットのデータ（勿論，同一フォーマットのものは何でも）をＤＰＡＣＳ／Ｊに取り込むＥＤＩＴＪ

以上の6つのサブシステムが，共通にデータ管理ファイル，データファイルを利用する構成となる。この全体の構成を第2.1図に示す。

また，それぞれのサブシステムに包含された機能の概要は以下の通りである。

(1) ＤＰＡＣＳ(*1) サブシステム

① システム及びシステムデータのハンドリング

② システムデータの相互変換

③ モデルの解析

④ コントローラの構成

⑤ システムの構成

⑥ 周波数領域における手法

⑦ システム同定（ＰＩＰＡＣＫと同じ）

(2) ＤＰＥＸ(*2) サブシステム

① システムデータのハンドリング

② システムの解析

(3) ＳＹＳ(*3)

① システムの解析

(4) ＴＳ(*4)

① 時系列データのハンドリング

② 時系列データの演算

③ 時系列データの生成

④ 時系列データの解析

<脚注> *1 Design Package for Control System

*2 DPACS Extension

*3 System Analysis

*4 Time Series Data Analysis

(5) ＭＡＴ(*5) サブシステム

① 行列データのハンドリング

② 行列データの演算

③ 行列データの解析

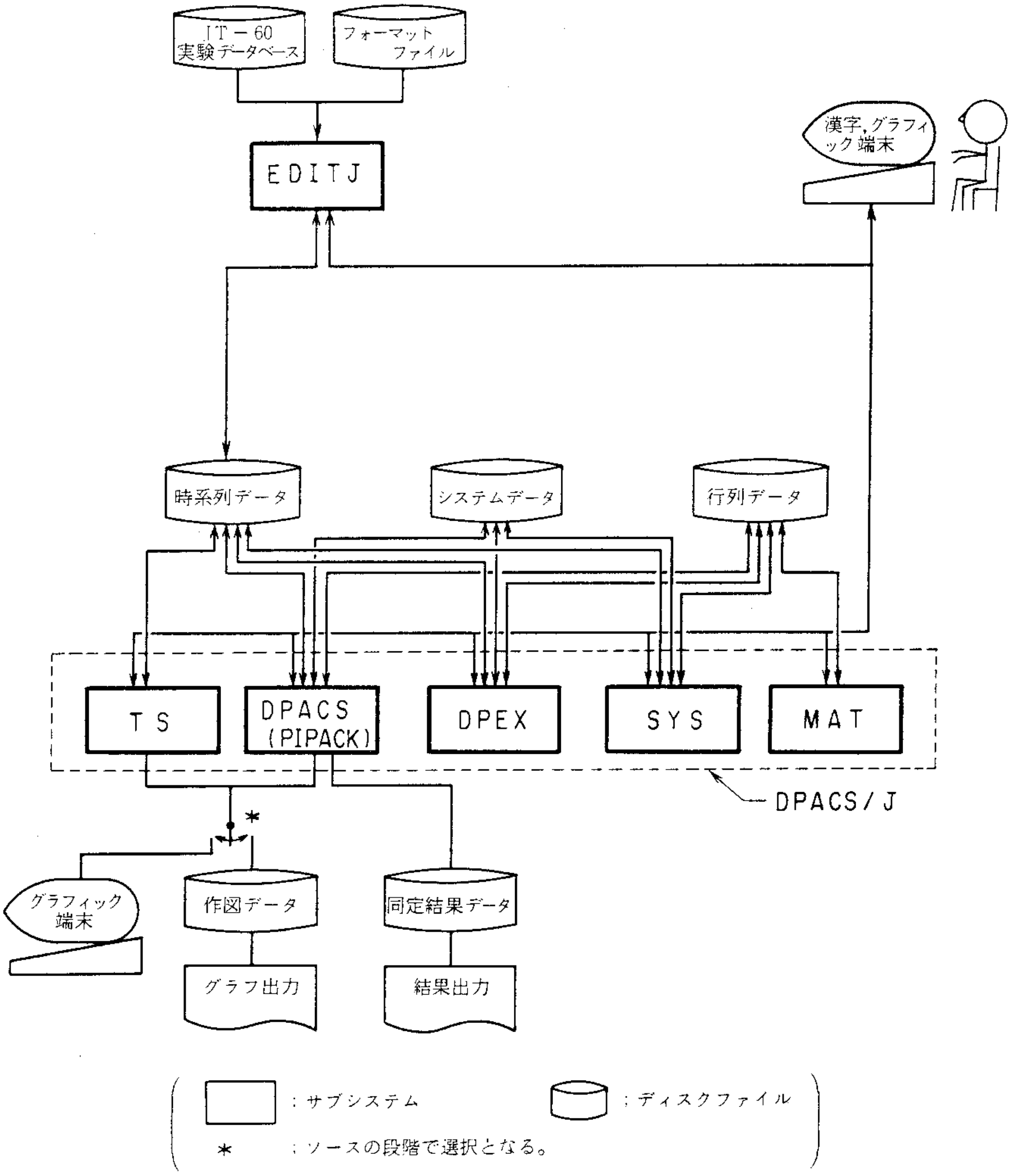
(6) ＥＤＩＴＪ(*6) サブシステム

① ＪＴ－60実験データベースの時系列データをＴＳサブシステムへ登録

② 所定のフォーマットで書かれた時系列データをＴＳサブシステムへ登録

＜脚注＞ ＊5 Matrix Analysis

＊6 Edition for ＪＴ－60 Data



第 2.1 図　プログラム群の構成

3. 制御系設計・解析パッケージ（ＤＰＡＣＳ／Ｊ）の理論概要・処理機能・利用編

3.1 ＤＰＡＣＳ／Ｊの主な特徴と記号の説明

ＤＰＡＣＳ／Ｊ（Design Package for Control System／JAERI）は，多変数制御系の計算機による設計支援（ＣＡＤ：Computer Aided Design）を目的として設計・開発されたプログラム・パッケージである。本パッケージは，線形時不変システムを記述する多様なデータの能率的な管理機能と制御系設計やデータ解析のための豊富な機能とから成り立っており，日本原子力研究所那珂研究所の大型汎用計算機（富士通（株）製Ｍ－780）上において動作可能な様に設計されている。ユーザは，端末（一般のキャラクタ端末ならびに図形処理端末）上にて会話形式でＤＰＡＣＳ／Ｊを使用することにより，各種処理を実行できる。

ＤＰＡＣＳ／Ｊの主な特徴を列挙すると，以下の様に要約することができる。

① 端末上にて会話形式で操作できる。

② コマンド形式にて入力することができる。

③ トライ＆エラーの設計，解析，シミュレーションが可能である。

④ データは管理ファイルによって一元管理されたファイルに格納されており，ファイルを意識することなしに，ユーザの付けたデータ名称を用いて随時アクセスすることができる。

⑤ 現代制御理論，古典制御理論の両方に基づく設計，解析が可能である。

⑥ プログラム上の内部表現は，倍精度実数型である。従って，数値計算上は有効桁で10進16桁あり，この桁まで保証される。数値表現の範囲は，10^{-78}～10^{+75}であるので，範囲を越えるとオーバーフローとなる。尚，アルゴリズムによっては，保証される有効桁数が変化するので注意が必要である。

<記号の説明>

本冊子では，多変数制御系理論を基礎にしているため，用いる諸量はベクトルや行列であることが殆どである。また，ギリシャ文字等も登場する。そこで以下に慣用的な規約を掲げる。

・ベクトル　；英小文字またはギリシャ小文字のボールド・イタリック・タイプ
（例：$\boldsymbol{x}$，$\boldsymbol{y}$，$\boldsymbol{\xi}$，$\boldsymbol{\nu}$）

・行列　；英大文字またはギリシャ大文字のボールド・イタリック・タイプ
（例：$\boldsymbol{X}$，$\boldsymbol{Y}$，$\boldsymbol{\Phi}$，$\boldsymbol{\Psi}$）

・転置行列　；行列の左肩に英小文字ｔ（transpose）を付加。（例：${}^t\boldsymbol{A}$，${}^t\boldsymbol{T}$，${}^t\boldsymbol{U}$）

・変換　；ベクトル量で座標変換後のベクトルを表す時に上棒（￣），山線（＾），波線（～）アスタリスク（＊）をかぶせる。（例：$\bar{\boldsymbol{x}}$，$\hat{\boldsymbol{u}}$，$\tilde{\boldsymbol{y}}$，$\boldsymbol{z}^*$）

- ベキ乗　；通常の記法に基づき，右肩に指数を付加する。
  （例：$\boldsymbol{A}^{k}, \boldsymbol{T}^{-1}$　ただし，指数関数は次のように書く。exp（$\boldsymbol{A}$））
- 単位行列　；n次元単位行列を$\boldsymbol{I}_{n}$と表す。次元に注意する必要がない場合は単に$\boldsymbol{I}$と表す。
- 零行列，零ベクトル；$\boldsymbol{0}$
- 擬似逆行列；右肩に＋を付加する。
- 離散時間と連続時間；原則として離散時間はk，連続時間はtで表すが，両方を共通的に表現する場合はtを使用する場合もある。
- システム　；$\Sigma_{@}$，和を表すΣとは使い方が全く異なるので混乱しない。変換後のシステムは，上棒（$^{-}$）やアスタリスク（*）を右肩につけて表す。
- 対角行列　；$\boldsymbol{A}=\mathrm{diag}(a_{11}, a_{22}, \cdots, a_{nn})$
- スカラー量；通常の英文字，ギリシャ小文字
- 共役複素数；右肩に＊を付けて表す。（例：$\boldsymbol{v}^{*}$）ただし，常に＊が共役複素数を表すとは限らない。混乱の無い範囲で変換の意味でも用いる。
- 時間微分　；$\dfrac{d}{dt}$ または $\dot{}$（ドット）で表す。
- 行列のトレース；tr($\boldsymbol{A}$)と表す。
- 全要素1のベクトル；$\boldsymbol{1}$と表す。

3.2　ＤＰＡＣＳ／Ｊの構成

ＤＰＡＣＳ／Ｊは，第3.1表に示す様に6つのサブシステムから成り立っている。サブシステム間の移行は，移行したいサブシステム名称を入力することにより簡単に実行できる。（3.5.3参照）

第3.1表　ＤＰＡＣＳ／Ｊの構成

No.	サブシステム名	機　能
1	ＤＰＡＣＳ	ＤＰＡＣＳ本体（＊1）
2	ＰＩＰＡＣＫ	モデル同定サブシステム（＊2）
3	ＤＰＥＸ	ＤＰＡＣＳ本体の拡張機能
4	ＳＹＳ	基本システム解析サブシステム
5	ＴＳ	時系列データ処理サブシステム（＊1）
6	ＭＡＴ	行列演算サブシステム

（＊1）ＤＰＡＣＳ／Ｊは，一般のキャラクタ端末ならびにグラフィック端末（Ｆ9434Ａ）で動作させることを想定しており，特にグラフィック端末用としてのサブシステムも用意している。

このサブシステムの特徴は，図をＮＬＰではなく，画面上にカラーで出力する点にある。しかも，この図の出力というただ一点を除けば，それぞれ一般のキャラクタ端末用サブシステム"ＤＰＡＣＳ"ならびに"ＴＳ"と全く同一の取り扱いが可能である。従って，以降違いがある場合を除き説明は省略する。

（＊2）"ＰＩＰＡＣＫ"は，"ＤＰＡＣＳ"のシステム同定機能として取り込まれているので，以降"ＰＩＰＡＣＫ"独自の説明は省略する。

3.2.1　ＤＰＡＣＳサブシステム

ＤＰＡＣＳ／Ｊの中核をなすサブシステムであり，以下の機能を持つ。

Ⅰ．システムおよびシステム・データのハンドリング

Ⅱ．システム・データの相互変換

Ⅲ．モデルの解析

Ⅳ．コントローラの構成

Ⅴ．システム構成

Ⅵ．周波数領域における各種の手法

Ⅶ．システム同定（ＰＩＰＡＣＫと同一機能である。）

本サブシステムに属するコマンドは第3.2表の通り。

第3.2表　DPACSコマンド一覧

機能分類	コマンド名	機能概要
DPACS Ⅰ システムおよびシステムデータのハンドリング	LIST	システムおよびシステムデータの一覧出力
	DELETE	〃　の削除
	RENAME	システムの名称変更
	PROTECT	システムの保護
	SYSIN	システムデータの入力
	TYPE	〃　の出力
	REVISE	〃　の修正
	COPY	〃　の複写
	PICK	空間の座標変換および入出力変数の選出
	DUAL	双対システムの作成
DPACS Ⅱ システムデータの相互変換	COOD	状態空間表現形式システムの正則変換
	CMR	外部記述の正準最小実現の取得
	TF	伝達関数行列表現形式システムの取得
	MRKV	マルコフパラメータ表現形式システムの取得
	DIF	微分（差分）方程式表現形式システムの取得
	IO	入出力データ表現形式システムの取得
	DIGIT	連続系状態空間表現形式システムの離散系化
	CONT	離散系状態空間表現形式システムの連続系化
DPACS Ⅲ モデルの解析	POLE	システムの極による安定判別
	ZERO	システムの零点の算出
	CCCO	システムの可制御性・可観測性判別
	RDCT	システムの低次元化
	EMR	状態空間表現形式システムの ε −最小実現取得
DPACS Ⅳ コントローラの構成	PLOC	極配置問題の求解
	OPT	最適制御問題の求解
	DCPL	非干渉化問題の求解
	OBS	オブザーバの構成
	MFS	モデル追従型サーボ問題の求解
	OSV	最適サーボ問題の求解
DPACS Ⅴ システム構成	LINK	複合線型（非線型）の構成
	CLPS	閉ループ系の構成
	SMLT	シミュレーションの実行
DPACS Ⅵ 周波数領域における手法	NYQST	（逆）ナイキスト線図の作図
	BODE	ボード線図の作図
	LOCI	根軌跡の作図
	BAND	（逆）ゲシュゴリン帯の作図
DPACS Ⅶ システム同定（PIPACK）	GLS	一般化最小2乗推定
	CML	条件付き最尤推定
	SML	対称型条件付き最尤推定
	LI	制限情報下最尤推定

3.2.2　DPEXサブシステム

DPACS拡張ライブラリ・サブシステムであり，以下の機能を持つ。

Ⅰ. システムデータのハンドリング

Ⅱ. システムの解析

本サブシステムに属するコマンドは，第3.3表の通り。

第3.3表　DPEXコマンド一覧

機能分類	コマンド名	機能概要
DPEX Ⅰ システムデータのハンドリング	LIST	システムおよびシステムデータの一覧出力
	DELETE	〃　の削除
	RENAME	システム名称変更
	RROTECT	システムの保護
DPEX Ⅱ システムの解析	INIT	入出力時系列データから内部状態変数の初期値を計算
	TRACK	トラッキング問題（任意目標入力）の求解

3.2.3　SYSサブシステム

基本システム解析サブシステムであり，以下の機能を持つ。

Ⅰ. システム解析

本サブシステムに属するコマンドは，第3.4表の通り。

第3.4表　SYSコマンド一覧

機能分類	コマンド名	機能概要
SYS Ⅰ システム解析	V	状態空間表現の可制御性・可観測性行列の取得
	LCF	Vの可制御性構造の特徴抽出
	SMAT	状態空間表現形式システム行列の取得
	VV	可制御性・可観測性グラム行列の取得
	RCT	定常リカッチ行列方程式の求解
	FILT	入力時系列データに対する応答

3.2.4　TSサブシステム

時系列データ処理用サブシステムであり，以下の機能を持つ。

Ⅰ. 時系列データのハンドリング

Ⅱ. 時系列データの演算

Ⅲ. 時系列データの生成

Ⅳ. 時系列データの解析

本サブシステムに属するコマンドは，第 3.5 表の通り。

第 3.5 表　TSコマンド一覧

機能分類	コマンド名	機能概要
TS Ⅰ 時系列データのハンドリング	LIST	時系列データの一覧出力
	TS	〃 入出力・複写
	REVISE	〃 修正
	PLOT	〃 グラフ出力
	PICK	〃 一部抽出
	DCMP	〃 分解
	APPEND	〃 結合
	PRMT	〃 並べ替え
	SKIP	〃 スキップ
TS Ⅱ 時系列データの演算	ADD	時系列データ同士の加算
	SADD	〃 への列ベクトルの加算
	SUB	〃 同士の減算
	SSUB	〃 からの列ベクトルの減算
	MUL	〃 同士の内積および行列との乗算
	TSMUL	〃 同士の乗算
	SMUL	〃 と列ベクトルとの乗算
	TSDIV	〃 同士の除算
	SDIV	〃 の列ベクトルによる除算
	DIF	〃 の差分値算出
	INT	〃 の積分値算出
	LOG	〃 の自然対数の算出
TS Ⅲ 時系列データの生成	GAUSS	正規雑音時系列データの生成
	MSEQ	最大周期時系列データ（M－系列）の生成
	SIN	正弦波時系列データの生成
	SQU	方形波時系列データの生成
	SAW	鋸波時系列データの生成
	TRI	三角波時系列データの生成
TS Ⅳ 時系列データの解析	STAT	時系列データの各種統計量の計算
	CORF	時系列データの相関関数の算出
	FFT	時系列データの高速フーリエ変換の実行

3.2.6　MAT

行列データ処理用サブシステムであり，以下の機能を持つ。

Ⅰ．行列データのハンドリング

Ⅱ．行列データの演算

Ⅲ．行列データの解析

本サブシステムに属するコマンドは第3.6表の通り。

第3.6表　MATコマンド一覧

機能分類	コマンド名	機能概要
MAT Ⅰ 行列データの ハンドリング	LIST	行列データの一覧出力
	MAT	〃　の入出力・修正
	DCMP	〃　の分解
	APPEND	〃　の結合
	PRMT	〃　の並べ替え
MAT Ⅱ 行列データの 演算	MTCAL	行列データの各種演算を連続に行う ①加算　②減算　③乗算（右） ④逆行列　⑤逆行列との積　⑥スカラー倍 ⑦乗算（左）⑧転置行列　⑨中間結果
	ADD	行列データ同士の加算
	SUB	〃　同士の減算
	MUL	〃　同士の乗算
	SMUL	〃　のスカラー倍
	INV	〃　の（擬似）逆行列の算出
	TRNS	〃　の転置行列の算出
	FUNC	多項式関数値の算出
	EXP	指数関数値の算出
	IEXP	指数関数の積分値の算出
	RSLV	リゾルベント行列の算出
	LYP	リヤプノフ方程式の求解
MAT Ⅲ 行列データの 解析	EIG	固有値問題の求解
	GEIG	一般化固有値問題の求解
	SVD	特異値分解の実行
	MAP	像空間，零化空間の基底の算出
	LSQ	最小2乗解の算出
	COOD	座標変換の実行
	VEC	ベクトル列の一次独立性の検証
	NORM	ノルム，行列式，トレースの算出

3.3 制御系の表現形式

ＤＰＡＣＳ／Ｊで取り扱う制御系の表現形式としては，ｍ入力ｐ出力の線型時不変システムを表現する数式モデルであり，第3.7表に示す5種類である。

第3.7表　制御系の表現形式と略号

制御系の表現形式	英語表現	表現形式の略号
状態空間表現形式	State Space	S
伝達関数行列表現形式	Transfer Function	T
マルコフ・パラメータ表現形式	Markov Parameter	M
微分（差分）方程式表現形式	Differential Equation	D
入出力データ表現形式	Input-Output Data	I

これらの略号を用いて，以降システムそのものを表す場合に $\Sigma_@$（@＝S, T, M, D, I のどれか）という形式を使う。また，同じ表現形式でも，連続系か離散系かによって違いがあるので注意を要する。

(1) 状態空間表現形式

$$\Sigma_S : \begin{cases} \sigma \boldsymbol{x}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) \end{cases}$$

ただし，$\boldsymbol{x}(t)$：状態変数ベクトル（$\boldsymbol{x}(t) \in \mathbf{R}^n$）

$\boldsymbol{u}(t)$：入力ベクトル（$\boldsymbol{u}(t) \in \mathbf{R}^m$）

$\boldsymbol{y}(t)$：出力ベクトル（$\boldsymbol{y}(t) \in \mathbf{R}^p$）

$\boldsymbol{A}$　：$\boldsymbol{A} \in \mathbf{R}^{n\times n}$

$\boldsymbol{B}$　：$\boldsymbol{B} \in \mathbf{R}^{n\times m}$

$\boldsymbol{C}$　：$\boldsymbol{C} \in \mathbf{R}^{n\times p}$

$\boldsymbol{D}$　：$\boldsymbol{D} \in \mathbf{R}^{p\times m}$

σ　：連続系では，微分演算子（d／dt）

離散系では，時間シフトオペレータ（z）

$$z\boldsymbol{x}(t) = \boldsymbol{x}(t+1)$$

状態空間表現システムデータ≡{m, p, n, $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{D}$}

(2) 伝達関数行列表現形式

$$\Sigma_T : \begin{cases} \boldsymbol{y}(\sigma) = \boldsymbol{G}(\sigma) \cdot \boldsymbol{u}(\sigma) \\ \boldsymbol{G}(\sigma) = \boldsymbol{P}(\sigma) / q(\sigma) + \boldsymbol{D} \end{cases}$$

ただし，$\boldsymbol{P}(\sigma)$, $q(\sigma)$：σ のｎ次多項式

$\boldsymbol{P}(\sigma) = \boldsymbol{P}_0\sigma^n + \boldsymbol{P}_1\sigma^{n-1} + \cdot\cdot\cdot + \boldsymbol{P}_n$

$\boldsymbol{P}_i \in \mathbf{R}^{p\times m}$ (i＝0, 1, 2, 3, ・・・, n)

$q(\sigma) = q_0\sigma^n + q_1\sigma^{n-1} + \cdot\cdot\cdot + q_n$

$q_i \in \mathbf{R}$ (i＝0, 1, 2, 3, ・・・, n ; $q_0 \neq 0$)

$\boldsymbol{u}(\sigma)$: 入力ベクトル（$\in \mathbf{R}^m$）

$\boldsymbol{y}(\sigma)$: 出力ベクトル（$\in \mathbf{R}^p$）

$\boldsymbol{D}$: $\boldsymbol{D} \in \mathbf{R}^{p\times m}$

σ : 連続系では，ラプラス変換後のパラメータ（s）

離散系では，時間シフトオペレータ（z）

$z\boldsymbol{u}(t) = \boldsymbol{u}(t+1)$

伝達関数行列表現システムデータ≡{m, p, n, q_0, …, q_n, $\boldsymbol{P}_0$, …, $\boldsymbol{P}_n$, $\boldsymbol{D}$}

＜正規形＞ $\Sigma_{T'}$: $\begin{cases} \boldsymbol{y}(\sigma) = \boldsymbol{G}(\sigma)\boldsymbol{u}(\sigma) \\ \boldsymbol{G}(\sigma) = \bar{\boldsymbol{P}}(\sigma) / \bar{q}(\sigma) + \bar{\boldsymbol{D}} \qquad (\bar{\boldsymbol{D}} \in \mathbf{R}^{p\times m}) \end{cases}$

ただし，$\bar{\boldsymbol{P}}(\sigma)$, $\bar{q}(\sigma)$: σのn次多項式

$\bar{\boldsymbol{P}}(\sigma) = \bar{\boldsymbol{P}}_1\sigma^{n-1} + \cdot\cdot\cdot + \bar{\boldsymbol{P}}_n$

$\bar{\boldsymbol{P}}_i = q_0^{-1}(\boldsymbol{P}_i - q_i q_0^{-1}\boldsymbol{P}_0)$

$\bar{\boldsymbol{P}}_i \in \mathbf{R}^{p\times m}$ (i＝1, 2, 3, ・・・, n)

$\bar{q}(\sigma) = \sigma^n + \bar{q}_1\sigma^{n-1} + \cdot\cdot\cdot + \bar{q}_n$

$\bar{q}_i \in \mathbf{R}$ (i＝1, 2, 3, ・・・, n)

$\bar{q}_i = q_0^{-1} q_i$

$\boldsymbol{D}$: $\boldsymbol{D} \in \mathbf{R}^{p\times m}$

$\boldsymbol{D} = \boldsymbol{D} + q_0^{-1}\boldsymbol{P}_0$

$\boldsymbol{u}(\sigma)$: 入力ベクトル（$\in \mathbf{R}^m$）

$\boldsymbol{y}(\sigma)$: 出力ベクトル（$\in \mathbf{R}^p$）

σ : 連続系では，ラプラス変換オペレータ（s）

離散系では，時間シフトオペレータ（z）

$z\boldsymbol{u}(t) = \boldsymbol{u}(t+1)$

正規形伝達関数行列表現システムデータ≡{m, p, n, 1, $\bar{q}_1$…, $\bar{q}_n$, $\boldsymbol{0}$, $\bar{\boldsymbol{P}}_1$, …, $\bar{\boldsymbol{P}}_n$, $\bar{\boldsymbol{D}}$}

上記に於て，$\Sigma_T = \Sigma_{T'}$である。

(3) マルコフパラメータ表現形式

$$\Sigma_M : \begin{cases} \boldsymbol{y}(\sigma) = \boldsymbol{G}(\sigma)\boldsymbol{u}(\sigma) \\ \boldsymbol{G}(\sigma) = \boldsymbol{D} + \boldsymbol{M}_1\sigma^{-1} + \boldsymbol{M}_2\sigma^{-2} + \cdot\cdot\cdot + \boldsymbol{M}_n\sigma^{-n} + \cdot\cdot\cdot \end{cases}$$

ただし，$\boldsymbol{u}(\sigma)$: 入力ベクトル（$\in \mathbf{R}^m$）

$\boldsymbol{y}(\sigma)$: 出力ベクトル（$\in \mathbf{R}^p$）

$\boldsymbol{M}_i$: $\boldsymbol{M}_i \in \mathbf{R}^{p\times m}$ (i＝1, 2, 3, ・・・, n, ・・・)

$\boldsymbol{D}$: $\boldsymbol{D} \in \mathbf{R}^{p\times m}$

σ　：連続系では，ラプラス変換オペレータ（s）

離散系では，時間シフトオペレータ（z）

$z\boldsymbol{u}(t)=\boldsymbol{u}(t+1)$

マルコフパラメータ表現システムデータ≡$\{m,\ p,\ n,\ \boldsymbol{D},\ \boldsymbol{M}_1,\ \boldsymbol{M}_2,\ \cdots,\ \boldsymbol{M}_n\}$

(4) 微分（差分）方程式表現形式

$\Sigma_D:\ \boldsymbol{T}(\sigma)\ \boldsymbol{y}(t)=\boldsymbol{U}(\sigma)\ \boldsymbol{u}(t)+\boldsymbol{T}(\sigma)\ \boldsymbol{D}\boldsymbol{u}(t)$

ただし，$\boldsymbol{T}(\sigma)=\boldsymbol{T}_0\sigma^n+\boldsymbol{T}_1\sigma^{n-1}+\cdots+\boldsymbol{T}_n$

$(\boldsymbol{T}_i\in\mathbf{R}^{p\times p};\ i=0,\ 1,\ 2,\ 3,\ \cdots,\ n)$

$\boldsymbol{U}(\sigma)=\boldsymbol{U}_0\sigma^n+\boldsymbol{U}_1\sigma^{n-1}+\cdots+\boldsymbol{U}_n$

$(\boldsymbol{U}_i\in\mathbf{R}^{p\times m};\ i=0,\ 1,\ 2,\ 3,\ \cdots,\ n)$

$\boldsymbol{D}$　：$\boldsymbol{D}\in\mathbf{R}^{p\times m}$

$\boldsymbol{u}(\sigma)$：入力ベクトル（$\in\mathbf{R}^m$）

$\boldsymbol{y}(\sigma)$：出力ベクトル（$\in\mathbf{R}^p$）

σ　：連続系では微分演算子（d／dt）

離散系では時間シフトオペレータ（z）

$z\boldsymbol{u}(t)=\boldsymbol{u}(t+1)$

微分（差分）方程式表現システムデータ≡$\{m,\ p,\ n,\ \boldsymbol{T}_0,\ \cdots,\ \boldsymbol{T}_n,\ \boldsymbol{U}_0,\ \cdots,\ \boldsymbol{U}_n,\ \boldsymbol{D}\}$

<正規形>　$\Sigma_{D'}:\ \bar{\boldsymbol{T}}(\sigma)\ \boldsymbol{y}(t)=\bar{\boldsymbol{U}}(\sigma)\ \boldsymbol{u}(t)+\bar{\boldsymbol{T}}(\sigma)\ \bar{\boldsymbol{D}}\boldsymbol{u}(t)$

ただし，$\bar{\boldsymbol{T}}(\sigma)=\boldsymbol{I}_p\sigma^n+\bar{\boldsymbol{T}}_1\sigma^{n-1}+\cdots+\bar{\boldsymbol{T}}_n$

$(\bar{\boldsymbol{T}}_i\in\mathbf{R}^{p\times p};\ i=1,\ 2,\ 3,\ \cdots,\ n)$

$\bar{\boldsymbol{T}}_i=\boldsymbol{T}_0^{-1}\boldsymbol{T}_i$

$\bar{\boldsymbol{U}}(\sigma)=\boldsymbol{U}_1\sigma^{n-1}+\cdots+\bar{\boldsymbol{U}}_n$

$(\bar{\boldsymbol{U}}_i\in\mathbf{R}^{p\times m};\ i=1,\ 2,\ 3,\ \cdots,\ n)$

$\bar{\boldsymbol{U}}_i=\boldsymbol{T}_0^{-1}(\boldsymbol{U}_i-\boldsymbol{T}_i\boldsymbol{T}_0^{-1}\boldsymbol{U}_0)$

$\bar{\boldsymbol{D}}$　：$\boldsymbol{D}+\boldsymbol{T}_0^{-1}\boldsymbol{U}_0\ (\bar{\boldsymbol{D}}\in\mathbf{R}^{p\times m})$

σ　：連続系では微分演算子（d／dt）

離散系では時間シフトオペレータ（z）

$z\boldsymbol{u}(t)=\boldsymbol{u}(t+1)$

正規形微分（差分）方程式表現システムデータ≡$\{m,\ p,\ n,\ \boldsymbol{I}_p,\ \bar{\boldsymbol{T}}_1,\ \cdots,\ \bar{\boldsymbol{T}}_n,\ \boldsymbol{0},\ \bar{\boldsymbol{U}}_1,\ \cdots,$
$\bar{\boldsymbol{U}}_n,\ \bar{\boldsymbol{D}}\}$

上記に於て$\Sigma_D=\Sigma_{D'}$である。

(5) 入出力データ表現形式

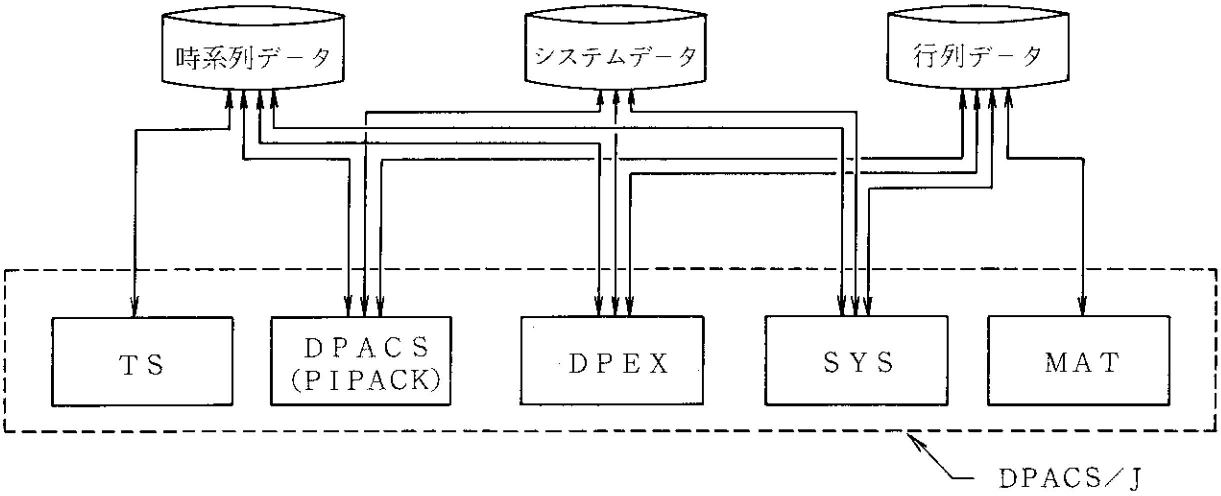
$\Sigma_I:\ \{(\boldsymbol{u}(k),\ \boldsymbol{y}(k)):\ k=0,\ 1,\ 2,\ 3,\ \cdots,\ n\}$

入出力データ表現システムデータ≡$\{m,\ p,\ n,\ \boldsymbol{u}_0,\ \boldsymbol{u}_1,\ \cdots,\ \boldsymbol{u}_n,\ \boldsymbol{y}_0,\ \boldsymbol{y}_1,\ \cdots,\ \boldsymbol{y}_n\}$

nの最大値＝2100

3.4 DPACS／Jに於けるデータのアクセスと管理体系

DPACS／Jの各サブシステムの関連ならびに各サブシステムがアクセスするデータの種類を第3.1図に示す。システムデータを取り扱うサブシステムは，DPACS（PIPACK），DPEX，SYSの3サブシステムであり，これらはまた，時系列データ並びに行列データも利用できる。TS，MATはそれぞれ，時系列データ，行列データのみ扱うことができる。



第3.1図　DPACS／Jの各サブシステムと使用データとの関係

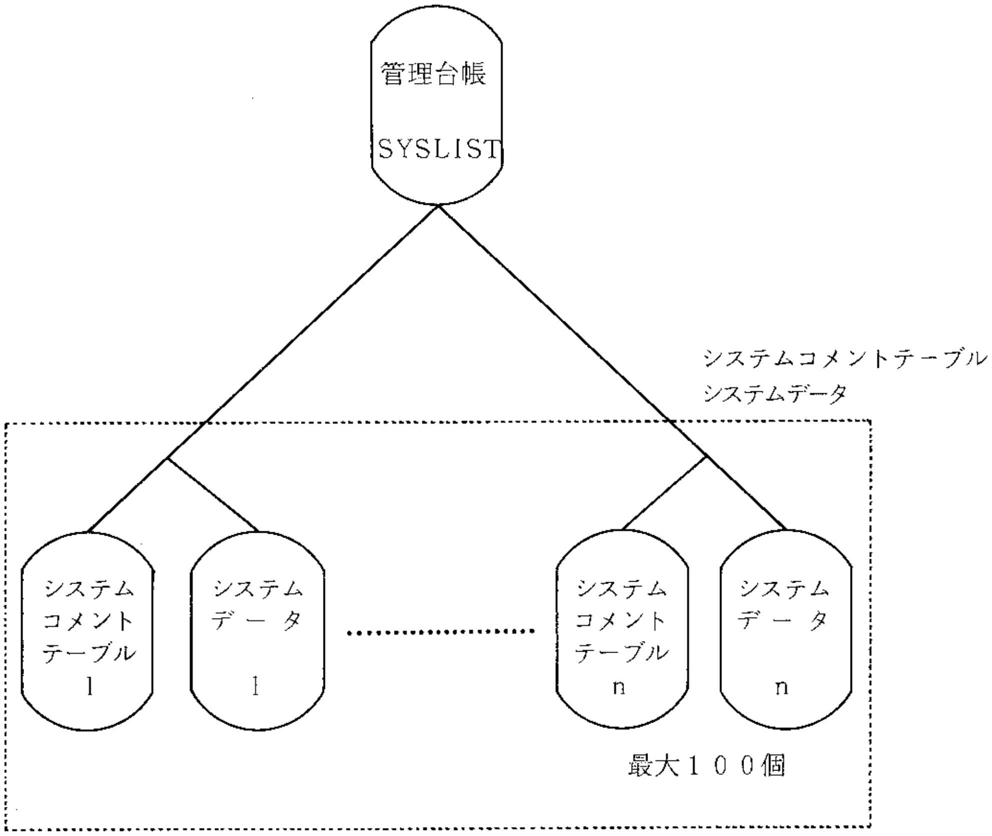
これら，3つのデータの管理体系について以降に記述するが，利用者にとって重要な部分だけは下線を引いておくので，そこだけを注意するだけで用は足りると思われる。

3.4.1　システムデータの管理体系

DPACS／Jにおけるシステムデータの管理をファイルの関連という観点から捉えると，第3.2図に示す様に，管理台帳であるSYSLISTファイルの配下に，システム・コメント・テーブルおよびシステムデータ・ファイルが存在する形式となっている。

DPACS／Jは，システムを管理台帳にて管理している。各々のシステムは，1個のシステム・コメント・テーブルと1個または複数個のシステムデータ・ファイルから構成されている。管理台帳には，登録されているシステムの各称と入出力変数ならびに状態変数の次元とが格納されてる。（第3.3図参照）システム・コメント・テーブルならびにシステムデータ・ファイルの名称の付け方は第3.3図に示した通りになっているが，ユーザはファイルの名称を意識する必要はなく，全てシステム名のみでシステムデータをハンドリングできる。

尚，システムの入力変数・出力変数及び状態変数の各次数の最大値は，それぞれ50, 50, 50である。また，登録できるシステム数は最大100である。



第3.2図　システムデータのファイルの関連

システム管理台帳（ファイル名称：SYSLISTF. DATA）

No.	システム名	入力次数	出力次数	状態次数	システムデータの数	プロテクトの有無	サンプリングインターバル
1	SN1	3	1	5	1	P	0.002
2	SN2	5	2	10	n		
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
100							

システム・コメント・テーブル（ファイル名称：SN2. C0. DATA）

；SN2のシステムデータ群（@0～@n）の情報

システムデータ・ファイル（ファイル名称：SN2. @0. DATA）

；システムの入力変数，出力変数，状態変数ベクトルの次数ならびにシステムの係数行列そのものが，システムデータの表現形式に合わせて格納されている。

⋮

システムデータ・ファイル（ファイル名称：SN2. @n. DATA）（注1）

（注1）　システムデータは，同一システム名の元に表現形式それぞれ最大10個まで管理できる。

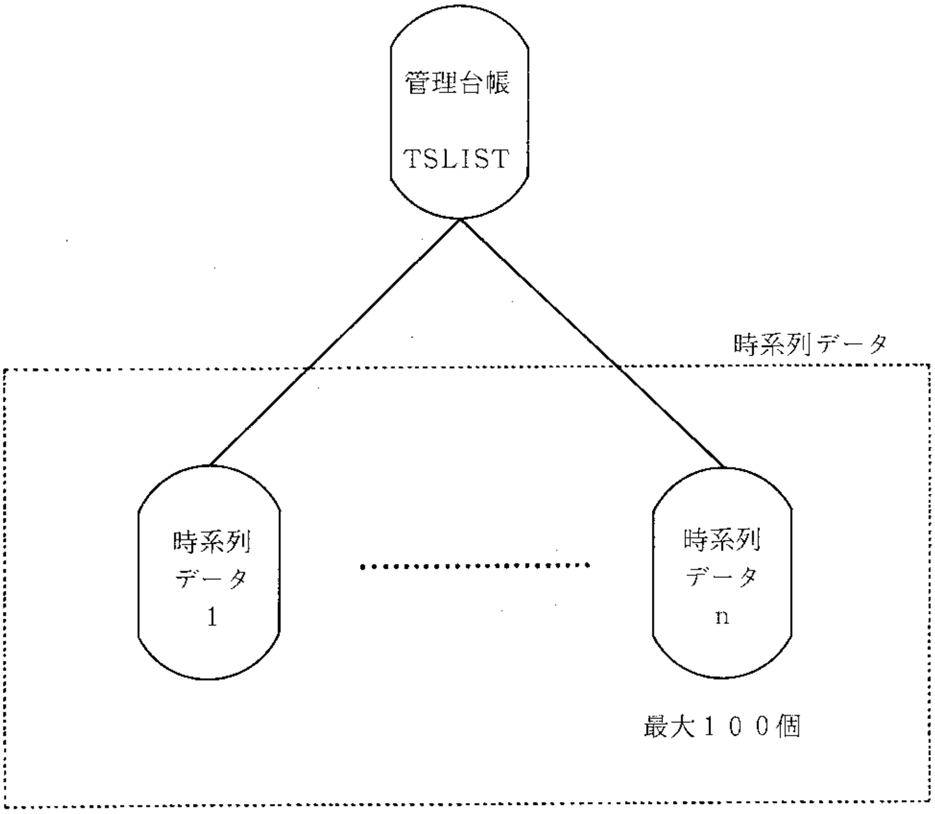
（注2）　図中の@は，@＝S，T，D，M，I（3.3参照）を意味する。

第3.3図　　システムデータの管理体系イメージ

3.4.2　時系列データの管理体系

ＤＰＡＣＳ／Ｊにおける時系列データの管理体系をファイルの関連という観点から見ると，第3.4図に示す様になっている。時系列データは，管理台帳によってその名称を管理される。（第3.4図参照）時系列データ・ファイルの名称の付け方は，第3.5図に示した様になっている。ただし，ユーザはファイル名を意識する必要はなく，全て時系列データ名のみでハンドリングできる。

尚，時系列データの1時間断面のベクトル要素数及び時間点数の最大値は，それぞれ50,2100)である。また，登録できる時系列データの数は，最大100である。



第3.4図　時系列データのファイルの関連

時系列データ管理台帳（ファイル名称：TSLISTF. DATA）

No.	時系列データ名	ベクトルの次元	時間点数
1	TSD1	3	5
2	TSD2	n	m
⋮	⋮	⋮	⋮
100			

時系列データ・ファイル（ファイル名称：TSD2. B0. DATA）

n	m	A_{10}	A_{20}	…	A_{n0}	………	$A_{n-1\ m}$	A_{nm}

この並びは，次の行列を意味する。

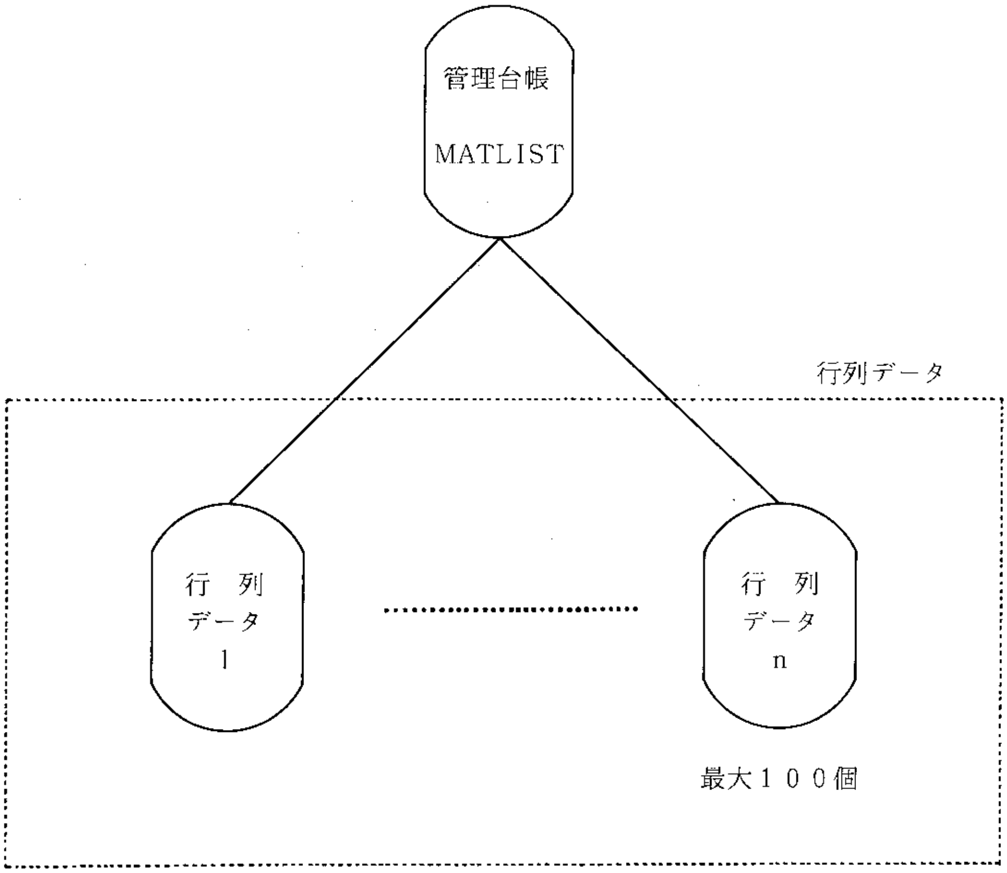
$$\begin{bmatrix} A_{10} & A_{11} & \cdots\cdots & A_{1m} \\ A_{11} & A_{21} & \cdots\cdots & A_{2m} \\ \vdots & & & \vdots \\ A_{n0} & A_{n1} & \cdots\cdots & A_{nm} \end{bmatrix}$$

第 3.5 図　時系列データの管理体系イメージ

3.4.3 行列データの管理体系

ＤＰＡＣＳ／Ｊにおける行列データの管理体系をファイルの関連という観点から見ると，第3.6図に示す様になっている。行列データは，管理台帳によってその名称を管理される。（第3.6図参照）行列データ・ファイルの名称の付け方は，第3.7図に示した様になっている。ただし，ユーザは，ファイル名を意識する必要はなく，全て行列データ名のみでハンドリングできる。

尚，行列データの行ならびに列の最大値は，それぞれ50,50である。また，登録できる行列データの数は，最大100である。



第3.6図　行列データのファイルの関連

行列データ管理台帳（ファイル名称：MATLISTF. DATA）

No.	行列データ名	行	列
1	MATD1	3	5
2	MATD2	n	m
⋮	⋮	⋮	⋮
100			

行列データ・ファイル（ファイル名称：MATD2. A0. DATA）

n	m	A_{11}	A_{21}	A_{31}	…	A_{n1}	A_{12}	……	$A_{n-1\,m}$	A_{nm}

この並びは，次の行列を意味する。

$$\begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1m} \\ A_{21} & A_{22} & \cdots & A_{2m} \\ \vdots & & & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nm} \end{bmatrix}$$

第 3.7 図　行列データの管理体系イメージ

3.4.4 複合線型・非線型システムデータ

ＤＰＡＣＳサブシステムのＬＩＮＫコマンドにより，複合線型または非線型のブロック線図を構成することができるが，その結果として，複合システムを表現するデータファイルが発生する。複合システムの発生と同時に，状態方程式表現形式の離散系及び連続系の２つの修飾子をつけた２つのファイルは無条件に発生する。即ち，名前ＡＢＣの複合システムをＬＩＮＫコマンドで構成すると，

ＡＢＣ．Ｓ０．ＤＡＴＡ，　ＡＢＣ．Ｃ０．ＤＡＴＡ
ＡＢＣ．Ｌ０．ＤＡＴＡ，　ＡＢＣ．Ｎ０．ＤＡＴＡ

の４つのファイルが発生する。このＡＢＣ．Ｌ０．ＤＡＴＡとＡＢＣ．Ｎ０．ＤＡＴＡは，それぞれ線型（Linear）部分と非線型（Nonlinear）部分のデータである。ＳＭＬＴコマンドによるシミュレーション実行時には，これら２つのデータを用いることになる。これらはデータ管理上表には現れてこないが，ＤＥＬＥＴＥコマンドでＡＢＣ．Ｓ０．ＤＡＴＡ及びＡＢＣ．Ｃ０．ＤＡＴＡを削除すると同時にＡＢＣ．Ｌ０．ＤＡＴＡ，ＡＢＣ．Ｎ０．ＤＡＴＡも削除される。登録できる複合システムは，名前毎に１組の非線型データである。最大登録数は，複合システムも含めたシステムデータ全体で，最大100である。

3.5 DPACS／Jに共通的な事柄と一般的な利用上の注意事項

3.5.1 DPACS／Jの起動方法

DPACS／Jは既に述べた様に，DPACS，DPEX，SYS，TS，MATの5個のサブシステムから構成されている。各サブシステムの起動方法は，TSSのREADYモード時にサブシステム名を入力することにより実行される。

実行例

```
READY
DPACS    ←─①
/DPACS/ TIME-19:31:08 CPU-00:00:04
   └─②           └─③
```

＜実行例の説明＞

① 稼動させるサブシステム名を入力する。

- DPACS
- DPEX
- SYS
- TS
- MAT

尚，サブシステム名のみを入力した場合には，F 9434 A 図形処理端末上で稼動する様に設定されている。漢字端末もしくはその他のキャラクタ端末上にて稼動させる場合には，各サブシステム名の後に空白を1個以上空けて，KN Jを入力する。その場合，DPACS並びにTSサブシステムのグリーティングメッセージのみが／DPACSK／…，／TSK／…の様になる。

入力例

```
READY
DPACS KNJ
 /DPACSK/ TIME-09:15:10 CPU-00:00:02
```

② サブシステム名称を示し，このグリーティング・メッセージが出力された状態で各サブシステムのコマンドの入力が可能となる。

③ 現在時刻並びに，ここまでに使用されたCPU時間が時：分：秒単位で表示されている。端末上にて稼動させる場合，CPU時間の最大値は2分である。次に行う処理内容によっては，一旦サブシステムを終了し，TSS端末を立ち上げ直すことが必要となるが，その際の参考にする。尚，実行例ではCPU時間を4秒使用したことを示す。

3.5.2 ＤＰＡＣＳ／Ｊの終了

ＤＰＡＣＳ／Ｊは，各サブシステムのコマンド入力待ち状態において，ＢＹＥを入力すると以下に示すメッセージを出力し，処理を終了する。

実行例

```
/DPACS/TIME-19:31:08 CPU-00:00:14
BYE ← ①
***OUTPUT PLOT TO NLP?===>Y ← ②
***OUTPUT PIPK TO NLP?===>Y ← ③
READY
```

＜実行例の説明＞

① ＤＰＡＣＳ／Ｊを終了するためのコマンドで各サブシステム共通である。

② 漢字端末用のシステムを起動した場合でかつ，作図関連のコマンドを実行した場合，Ｙを入力すると，作成された図がＮＬＰ（デフォルト値＝ＪＴ－60制御棟計算機室ＮＬＰ；Ｒクラス）に出力される。Ｎまたは⏎のみを入力した場合には図は出力されないが，作図ファイル（ＧＲＤＡＴＡ．ＤＡＴＡ）は保存されているので，6．導入編で述べるＪＣＬをＳＵＢＭＩＴすることにより，後で出力することもできる。

尚，図形処理端末用システムを実行した場合には，このメッセージは出力されない。

③ ＤＰＡＣＳサブシステムのシステム同定機能（ＰＩＰＡＣＫサブシステム）を実行した場合，Ｙを入力すると，計算結果がＮＬＰに出力される。Ｎまたは⏎のみを入力した場合は出力されないが，計算結果ファイル（ＰＩＰＡＣＫＬＰ．ＤＡＴＡ）は保存されているので，作図ファイルと同様に，ＪＣＬをＳＵＢＭＩＴすることにより，後で出力することもできる。

3.5.3 他サブシステムへの移行方法

既にＤＰＡＣＳ／Ｊのあるサブシステムが立ち上がっている場合，その他のサブシステムへの移行は，コマンド入力待ちの状態にて，起動しようとするサブシステムの名称を入力することによって行われる。

実行例

```
/DPACS/ TIME-19:31:08 CPU-00:00:14
MAT ← ①
 /MAT/ TIME-19:31:08 CPU-00:00:16 ← ②
```

＜実行例の説明＞

① 移行するサブシステムの名称を入力

② サブシステムに起動が掛かりコマンド入力待ちの状態になったことを示すグリーティング・メッセージ

3.5.4 コマンドの一般的形式

ＤＰＡＣＳ／Ｊのコマンドは，次の２つのタイプに大別できる。

① ＣＭＤ／ＳＷ１／ＳＷ２…／ＳＷｎ␣ＳＮ␣ＯＳＮ

② ＣＭＤ

①のコマンドの形式は，ＤＰＡＣＳ，ＤＰＥＸ，ＳＹＳの各サブシステムに多く現れる形式のコマンドであり，ＳＷｎは，処理の種類を示し，ＳＮは対称となるシステム名称を，ＯＳＮは新しく作成されＤＰＡＣＳ／Ｊの管理下におかれるシステム名称である。これらのパラメータはコマンドにより省略可能であったり，そうでなかったりする。

②のコマンド形式は，ＭＡＴやＴＳサブシステムに多く現れる形式のコマンドであり，パラメータは存在しない。このコマンドを受け付けた後で，処理の対象となる行列データの名称や時系列データの名称の入力要求があるので，それらに従ってパラメータ入力をすることになる。

コマンドは，各サブシステムのコマンド入力待ち状態（／サブシステム名／ＴＩＭＥ－ＨＨ：ＭＭ：ＳＳ　ＣＰＵ－ＨＨ：ＭＭ：ＳＳの後）でのみ入力することができる。

3.5.5 ＤＰＡＣＳ／Ｊに共通な操作

(1) 時系列データの入力操作方法

次のメッセージは，Ｎ×Ｍの時系列データ***A***の入力待ちであることを示す。

INPUT-MODE OF（N，M）TIME SERIES A ：

このメッセージに対し，次の４通りの入力指示が可能である。

第3.8表　時系列データの入力方法

入力指示	意味
↵	50列のブロック毎に分割してマニュアルにて入力する。（注１）
Ｚ ↵	全ての要素を０にする。
Ｃ ↵	全ての要素を同一の値にする。（値は，１個のみ入力）
Ｆ ↵	ファイルから時系列データを読み込む。（注２）

（注１） 最初に初期値（Ａ(Ｎ，０)）を入力し，それ以降は，50ブロックずつの行列(Ａ(50，50))の形式で入力していく。行列の入力形式に付いては，後述する。

（注２） ＦＩＬＥＮＡＭＥ：というメッセージが引き続き表示されるので，この後に続けて読み込む時系列データの名称（８文字以内）を入力する。

(2) 時系列データの出力操作方法

次のメッセージは，N×Mの時系列データ***A***の出力待ちであることを示す。

OUTPUT-MODE OF (N, M) TIME SERIES A :

このメッセージに対し，次の通りの出力指示が可能である。

第 3.9 表　時系列データの出力方法

入力指示	意味
⏎ C ⏎	50列のブロック毎に分割して画面に出力する。
F ⏎	ファイルに時系列データを書き出す。

① ⏎のみまたはC⏎を表示した場合は，

1) 最初に以下のメッセージが出力される。

ALL DATA ?

- 時系列データの行・列全てを表示する場合はYまたは⏎のみを入力する。
- 時系列データの行・列の一部しか表示しない場合にはNを入力する。この場合，以下のメッセージが表示される。

STARTING POINT ? (0-M)

ここで，画面に表示する時系列データの開始列を入力する。

HOW MANY POINTS ? (MIN-MAX)

ここで，画面に出力する時系列データの列数を入力する。

2) 1) でYを入力またはSTARTING POINTを0とした場合のみ，初期値を表示後，以下のメッセージが表示される。

FILENAME :

初期値を行列データとして出力したい場合には，8文字以内の英数字(先頭は英字)で行列データ名を入力する。

出力の必要が無い場合には，そのまま⏎のみを入力する。

3) 指定の時系列データが50列分表示される毎に，次のメッセージが表示される。

FILENAME :

行列データとして出力したい場合には，8文字以内の英数字(先頭は英字)で行列データ名を入力する。

4) 以下のメッセージが表示される。

OUTPUT-FILENAME :

ここで，時系列データ（全ての行列）を出力する場合には，８文字以内の英数字（先頭は英字）で時系列データ名を入力する。

出力の必要がない場合には，そのまま⏎のみを入力する。

② F⏎を入力した場合は

1）以下のメッセージが表示される。

OUTPUT-FILENAME：

ここで，時系列データ（全ての行列）を出力する場合には，８文字以内の英数字（先頭は英字）で時系列データ名を入力する。

出力の必要が無い場合には，そのまま⏎のみを入力する。

(3) 行列データの入力操作方法

次のメッセージは，N×Mの行列データ**A**の入力待ちであることを示す。

INPUT-MODE OF（N，M）MATRIX A：

このメッセージに対し，次の11通りの入力指示が可能である。

第3.10表　行列データの入力方法

入力指示	意味
⏎	N×Mの行列を１行毎に入力する。
I ⏎	（i，i）要素（i≦min（N，M））を１にし，その他の要素は全て０にする。
－I ⏎	（i，i）要素（i≦min（N，M））を－１にし，その他の要素は全て０にする。
D ⏎	（i，i）要素（i≦min（N，M））だけを入力し，その他の要素は全て０にする。
S ⏎	左上対称min（N，M）次小行列の左下三角部だけを入力し，右上三角部は自動的に対称にし，その他の要素は全て０にする。
Z ⏎	全ての要素を０にする。
T ⏎	既に値が入っている場合画面に出力する。
R ⏎	既に値が入っている場合指定した行・列の要素の値を修正する。
F ⏎	ファイルから行列データを読み込む。
－F ⏎	ファイルから行列データを符号を反転して読み込む。
F' ⏎	ファイルから行列データを転置して読み込む。

① F，－F，F′を指定した場合には，以下のメッセージが表示されるので，この後に続けて8文字以内の英数字（先頭は英字）で読み込む行列データの名称を入力する。

FILENAME ：

値の読み込みが終ると，以下のメッセージが表示される。

TYPE OR REVISE ？

これに対して，以下の4通りの指定が可能である。

第3.11表　行列データの画面表示及び修正指示

入力指示	意味
↵ N ↵	行列データの修正は行わず入力を終了する。
T ↵	現在入力している行列データの要素の値を画面に表示する。（＊1）
R ↵	現在入力している行列データの任意の要素を修正する。（＊2）

（＊1）画面に表示後，再び"TYPE OR REVISE？"のメッセージが出力される。

（＊2）" SPECIFY(I，J)：" のメッセージの後に続けて，修正すべき行と列を入力すると，" A (I，J)＝X. XXXXXXXXD＋YY REVISE TO" のメッセージが出力されるので修正データを入力する。

(4) 行列データの出力操作方法

次のメッセージは，N×Mの行列データ$\boldsymbol{A}$の出力待ちであることを示す。

OUTPUT-MODE OF (N，M) MATRIX A：

このメッセージに対し，次の3通りの出力指示が可能である。

第3.12表　行列データの出力方法

入力指示	意味
↵ C ↵	画面に出力する。
F ↵	ファイルに行列データを書き出す。

① ⏎のみまたはC⏎を出力した場合，画面上に行列データを表示後，以下のメッセージが表示される。

FILENAME ：

ここで，行列データを出力する場合には，8文字以内の英数字（先頭は英字）で行列データ名を入力する。

出力の必要が無い場合には，そのまま⏎のみを入力する。

② F⏎を入力した場合，以下のメッセージが表示される。

FILENAME ：

ここで，行列データを出力する場合には，8文字以内の英数字（先頭は英字）で行列データ名を入力する。

出力の必要が無い場合には，そのまま⏎のみを入力する。

(5) その他の入力待ち状態

その他の入力待ち状態としては，以下の2通りがある。

① メッセージの最後にコロン（：）のあるもの

② メッセージの最後にコロンのないもの（？が付加されている場合もある）

①に対しては，適当な数値を入力する。

②に対しては，次の3通りの入力が可能である。

第3.13表　その他の入力待ち状態における入力指示

入 力 指 示	意　　　味
⏎ Y⏎	" YES "（処理を行う。）
N⏎	" NO "（処理を行わない。）

(6) ε（収束判定値）の入力

DPACS／Jにおいては，収束判定値εの入力を必要とするコマンドが多数あるが，これらはすべて収束判定用として使用しているため，あまり小さな値を設定すると収束せず，その後の処理が意味を持たなくなる恐れがある。従ってDPACS／Jにおいて使用するεの値としては，1.0×10^{-10}（1.0D−10）程度がほぼ適当であろう。

尚，この値より精度を上げて収束させようとした場合には，そのコマンドによって得られた結果が妥当か否か検証する必要がある。

(7) システムが複数個のシステムデータを持つ場合のシステムデータの指定

各種のコマンドにおいて，複数個のシステムデータを持つシステムを指定した場合に，以下の様なメッセージが出力される。このメッセージに従って，使用するシステムデータを選択する。

```
/DPACS/ TIME-10:01:25 CPU-00:00:16
TYPE EXIDN
 SPECIFY TYPE OF SYSTEMDATA :    D
 SYSTEM EXIDN     HAS THE FOLLOWING SYSTEMDATAS
  1) EXIDN.D0     (    2) ;GLS       /  /  /  /    (89/ 7/21) :
  2) EXIDN.D1     (    2) ;GLS       /  /  /  /    (89/ 7/24) :
  3) EXIDN.D2     (    2) ;GLS       /  /  /  /    (89/ 7/25) :
  4) EXIDN.D3     (    2) ;GLS       /  /  /  /    (89/ 7/25) :
  5) EXIDN.D4     (    2) ;GLS       /  /  /  /    (89/ 7/25) :
  6) EXIDN.D5     (    2) ;GLS       /  /  /  /    (89/ 7/25) :
  7) EXIDN.D6     (    2) ;GLS       /  /  /  /    (89/ 7/25) :
 WHICH SYSTEMDATA IS USED ? :
```

尚，システムデータが1個のシステムデータしか持たない場合には，上記のメッセージは出力されない。

3.5.6　マニュアル記述上の注意

3.6～3.10において，各サブシステムのコマンドの機能や理論概要，実行例について記述して行くが，その際の記述上の凡例についてここで述べておく。

各サブシステムのコマンドは，次の例の形式にて記述されている。

機能概要（①）	コマンド番号（②）
コマンド記述形式（③）	

機　　能

コマンドの機能ならびにスイッチの説明等について記述している。

理論概要

コマンドの機能の理論的背景ならびにアルゴリズム等について記述している。

実 行 例

コマンドの実行例の中で特徴的なものについて掲載している。（④，⑤）

<詳細説明>

① コマンドの機能概要の記述

② コマンド番号…（サブシステムコード，機能分類番号，通番）

第 3.14表　　サブシステムコード

サブシステム名	サブシステムコード
DPACS	DP
DPEX	DX
SYS	SY
TS	TS
MAT	MT

機能分類番号については，各サブシステムのコマンド一覧（3.2.1～3.2.5章参照）の機能分類のローマ数字である。

通番は，機能分類ごとに付番している。

③ コマンドの記述形式は，以下の規則に従って記述されている。

(　　)…………省略可能なスイッチまたはオペランドであることを示す。但し，省略した場合にはシステムのデフォルトが指定されていなければ後で訊いてくる。

[　　]…………縦に並んでいるスイッチの中から1つを選択することを示す。

([　])…………[　]内のスイッチから1つを選択できるが，これ全体を省略できることを示す。選択を後で訊いてくる。

アンダーライン……スイッチを省略した場合のシステムのデフォルト値を示す。

(例1)

$\left(\begin{bmatrix} \underline{/N} \\ /P \end{bmatrix}\right)$；スイッチのどれかを指定できるが，省略も可。ただし，省略した場合には，下線の選択と見做される。

(例2) $(\underline{/C})$；省略可能で，省略時には本スイッチが指定されたものと見做される

"␣"………………スペースを意味する。

"#"………………スイッチの入力なしを意味し，スペース"␣"すら必要としない。

(例3)

$\begin{bmatrix} \# \\ /L \end{bmatrix}$；スイッチを指定した場合と指定しなかった場合でそれぞれ意味が異なる。

④ 実行例中の入力部分は，下線で示されている。下線の無いものはプログラムや計算機システムが出力したメッセージである。

⑤ 実行例中の"＊＊＊"は，端末画面上の改ページを意味し，本パッケージとは関係の無いものである。

3.6　ＤＰＡＣＳサブシステムの理論概要・処理機能・利用例

以下に次の順序で説明していく。

Ⅰ．システム及びシステムデータのハンドリング

Ⅱ．システムデータの相互変換

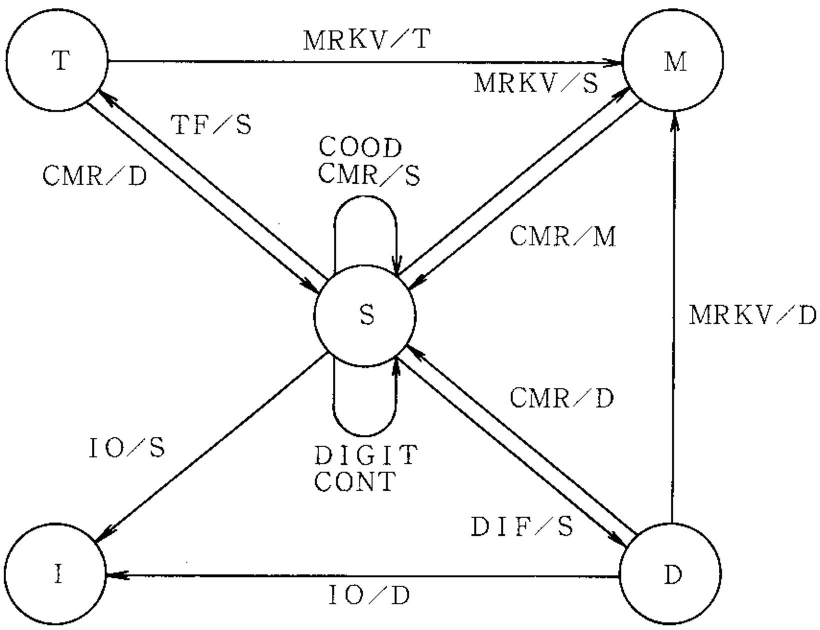
Ⅲ．モデルの解析

Ⅳ．コントローラの構成

Ⅴ．周波数領域における手法

Ⅵ．システム同定

特に，Ⅱ．システムデータの相互変換は，ＤＰＡＣＳ／Ｊで取り扱う５種類のシステム表現形式間の変換を行う。変換とコマンドの対応を第3.8図に示す。



第3.8図　システムデータの相互変換機能

システムおよびシステムデータの一覧出力	ＤＰ．Ⅰ．１
ＬＩＳＴ｛＿ＳＮ１｛＿ＳＮ２｛・・・｝｝｝	

機　　能

現在ＤＰＡＣＳに登録されているシステムの一覧およびその詳細情報であるシステムデータを画面に出力する。

◆システム名を入力しなかった場合は，登録されているシステムの一覧を出力する。

◆システム名を入力した場合は，入力されたシステムの全てのシステムデータを出力する。

実 行 例

〔1〕 システムの一覧出力例

```
LIST
TOTAL NO. OF INSTALLED SYSTEMS :    24
------------------------------------------------------------------------------
        NAME①      ( M, P)②   TYPE③                                NSD⑤   P⑥
--------------------------------------------------------------④---------------
  1     PIP        ( 2, 2)    D     (  1.00000000D-02 )             7
  2     SML        ( 2, 2)    D     (  1.00000000D-02 )             1
  3     LI         ( 2, 2)    D     (  1.00000000D-02 )             1
  4     CMLD       ( 1, 1)    D     (  1.00000000D-02 )             3
  5     SMLD       ( 1, 1)    D     (  1.00000000D-02 )             1
  6     SMLD2      ( 1, 1)    D     (  1.00000000D-03 )             1
  7     SMLD3      ( 1, 1)    D     (  1.00000000D-04 )             3
  8     SMLAB      ( 1, 1)    D     (  1.00000000D-04 )             1
  9     LID        ( 1, 1)    D     (  1.00000000D-02 )             3
 10     PIP2       ( 2, 2)    D     (  1.00000000D-02 )             3
 11     EX4C       ( 4, 2)    C                                     1
 12     EX3        ( 1, 2)    C                                     1
 13     EX10       ( 2, 2)    C                                     1
 14     EX12       ( 1, 1)    C                                     1
 15     EX11       ( 2, 2)    C                                     1
 16     EX13       ( 1, 1)    C                                     1
 17     PIP3       ( 2, 2)    D     (  1.00000000D-02 )             2
 18     PIP4       ( 2, 2)    D     (  1.00000000D-02 )             1
***
 19     PIP5       ( 2, 2)    D     (  1.00000000D-02 )             1
 20     LID2       ( 1, 1)    D     (  1.00000000D-02 )             3
------------------------------------------------------------------------------
------------------------------------------------------------------------------
        NAME       ( M, P)    TYPE                                 NSD    P
------------------------------------------------------------------------------
 21     CMLD2      ( 1, 1)    D     (  1.00000000D-02 )             2
 22     SMLD3A     ( 1, 1)    D     (  1.00000000D-04 )             2
 23     EX9        ( 1, 1)    D     (  1.00000000D-01 )             1
 24     EX1        ( 2, 2)    C                                     1
------------------------------------------------------------------------------
/DPACS/ TIME-22:34:51 CPU-00:00:01
```

【説明】

① システム名

② 入出力変数の次数

③ システムデータ タイプ（C：連続系，D：離散系）

④ サンプル周期

⑤ システムデータ数

⑥ プロテクトの有無（P：有り，␣：無し）

〔2〕 システムデータの一覧出力例

```
/DPACS/ TIME-22:35:03 CPU-00:00:01
LIST PIP2 EX1
--------------------------------------------------------------------------------
 DISCRETE SYSTEM PIP2①  ( 2, 2)②                 (  1.00000000D-02 )③
--------------------------------------------------------------------------------
 1) PIP2.IO ④    (2100)⑤;SYSIN   /I /  /  / ⑥ (87/ 8/25)⑦:FOR GLS ⑧
 2) PIP2.DO      (    1) ;GLS    /  /  /  /   (87/ 8/25) :END
 3) PIP2.D1      (    1) ;GLS    /  /  /  /   (87/ 8/25) :END
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
 CONTINUOUS SYSTEM EX1     ( 2, 2)
--------------------------------------------------------------------------------
 1) EX1.SO       (    3) ;SYSIN   /S /C /  /   (87/11/27) :EX1 TEST DATA
--------------------------------------------------------------------------------
/DPACS/ TIME-22:46:34 CPU-00:00:19
```

【説明】

① システム名
② 入出力変数次数
③ サンプル周期
④ システムデータ名
⑤ 内部状態変数次数
⑥ 作成コマンド及びスイッチ
⑦ 作成日付
⑧ コメント

システムおよびシステムデータの削除	DP. I. 2
DELETE_SN1 {_SN2 {・・・}}	

機　能

指定したシステムのシステムデータを削除する。

◆指定したシステムがシステムデータを1個しか持っていない場合には，システムデータを削除し，そのシステムを管理ファイル上から削除する。

◆指定したシステムが複数個のシステムデータを持っている場合には，削除するシステムデータを選択することができる。全てのシステムデータを削除した場合には，そのシステムを管理ファイル上から削除する。

実行例

〔1〕 システムの削除例

```
 /DPACS/ TIME-22:35:07 CPU-00:00:01
DELETE EX10 EX4C
 /DPACS/ TIME-22:46:41 CPU-00:00:20
```

〔2〕 システムデータの削除例

```
 /DPACS/ TIME-22:35:05 CPU-00:00:01
DELETE SYS1
 DELETE ALL SYSTEMDATAS ? ① N
 SYSTEM SYS1      HAS THE FOLLOWING SYSTEMDATAS
  1) SYS1.S0      (    3) ;SYSIN     /S /C /  /    (88/ 2/26) :
  2) SYS1.S1      (    2) ;CMR       /S /  /  /    (88/ 2/26) :
  3) SYS1.M0      (    7) ;MRKV      /S /  /  /    (88/ 2/26) :
 HOW MANY SYSTEMDATAS ARE DELETED ? : ② 2
 SPECIFY 2-SYSTEMDATAS TO BE DELETED : ③ 2,3
 /DPACS/ TIME-22:46:37 CPU-00:00:19

 /DPACS/ TIME-22:46:38 CPU-00:00:19
LIST SYS1
 ----------------------------------------------------------------
 CONTINUOUS SYSTEM SYS1    ( 1, 1)
 ----------------------------------------------------------------
  1) SYS1.S0      (    3) ;SYSIN     /S /C /  /    (88/ 2/26) :
 ----------------------------------------------------------------
 /DPACS/ TIME-22:46:40 CPU-00:00:20
```

【説明】

① システムがシステムデータを複数個持つ場合に，このメッセージが表示される。これに対しては，3.5.5(5)に示す形式の入力指示が可能である。

② ①でNを入力した場合に，このメッセージが表示される。これに対して，削除するシステムデータの個数（0～（システムデータの総数））を入力する。

③ ②で1～（システムデータの総数－1）を入力した場合に，このメッセージが出力される。これに対して，消去すべきシステムデータの番号を②で指定した個数分，コンマ（，）または空白（␣）で区切って入力する。

システムの名称変更	DP. I. 3
RENAME␣SN␣OSN	

機能

システム名をＳＮからＯＳＮに変更する。

実 行 例

〔1〕 システム名称の変更例

```
 /DPACS/ TIME-22:35:09 CPU-00:00:01
RENAME SYS1 SYS11
 /DPACS/ TIME-22:46:42 CPU-00:00:20
```

システムの保護	DP. I. 4
PROTECT [# / /U] _SN1 {_SN2 {・・・}}	

機 能

システムのプロテクトやプロテクトの解除を行う。

◆第1スイッチ／Uを付加しなかった場合には，当該システムをプロテクトする。

◆第1スイッチ／Uを付加した場合には，当該システムのプロテクトを解除する。

実 行 例

〔1〕 システム保護の設定例，設定解除例

```
 /DPACS/ TIME-17:05:38 CPU-00:00:02
LIST
 TOTAL NO. OF INSTALLED SYSTEMS :    15
 -------------------------------------------------------------
       NAME      ( M, P)   TYPE                     NSD   P
 -------------------------------------------------------------
   1   VINPTSYS( 4, 1)     C                         1    P
   2   EDDYSYS0( 5, 1)     C                         1    P
   3   EDDYSYS ( 5, 1)     C                         1    P
   4   COILONLY( 5, 1)     C                         1    P
   5   WHOLEBV ( 5,40)     C                         1    P
   6   VCOILSYS( 5, 1)     C                         1
   7   EX      ( 1, 1)     C                         2
   8   EXD     ( 1, 2)     C                         1
   9   STEP    ( 2, 2)     C                         1
  10   STEP2   ( 1, 1)     C                         1
  11   TRCKRN  ( 5, 6)     C                         1
  12   TRCKRND ( 5, 6)     C                         1
  13   TRCKRNDD( 5, 6)     C                         1
  14   TEST    ( 1, 1)     C                         1
  15   TEST3   ( 3, 3)     C                         1
 -------------------------------------------------------------
 /DPACS/ TIME-17:07:34 CPU-00:00:03
```

```
 /DPACS/ TIME-17:07:56 CPU-00:00:03
PROTECT EX EXD
 /DPACS/ TIME-17:09:15 CPU-00:00:08
DELETE EX
 *** SYSTEM EX        IS PROTECTED
 /DPACS/ TIME-17:09:49 CPU-00:00:09
LIST
 TOTAL NO. OF INSTALLED SYSTEMS :    15
 ---------------------------------------------------------------------
       NAME     ( M, P)   TYPE                                NSD   P
 ---------------------------------------------------------------------
   1   VINPTSYS( 4, 1)     C                                   1    P
   2   EDDYSYS0( 5, 1)     C                                   1    P
   3   EDDYSYS ( 5, 1)     C                                   1    P
   4   COILONLY( 5, 1)     C                                   1    P
   5   WHOLEBV ( 5,40)     C                                   1    P
   6   VCOILSYS( 5, 1)     C                                   1
   7   EX      ( 1, 1)     C                                   2    P
   8   EXD     ( 1, 2)     C                                   1    P
   9   STEP    ( 2, 2)     C                                   1
  10   STEP2   ( 1, 1)     C                                   1
  11   TRCKRN  ( 5, 6)     C                                   1

  12   TRCKRND ( 5, 6)     C                                   1
  13   TRCKRNDD( 5, 6)     C                                   1
  14   TEST    ( 1, 1)     C                                   1
  15   TEST3   ( 3, 3)     C                                   1
 ---------------------------------------------------------------------
 /DPACS/ TIME-17:13:12 CPU-00:00:10
PROTECT/U EX
 /DPACS/ TIME-17:14:09 CPU-00:00:14
LIST
 TOTAL NO. OF INSTALLED SYSTEMS :    15
 ---------------------------------------------------------------------
       NAME     ( M, P)   TYPE                                NSD   P
 ---------------------------------------------------------------------
   1   VINPTSYS( 4, 1)     C                                   1    P
   2   EDDYSYS0( 5, 1)     C                                   1    P
   3   EDDYSYS ( 5, 1)     C                                   1    P
   4   COILONLY( 5, 1)     C                                   1    P
   5   WHOLEBV ( 5,40)     C                                   1    P
   6   VCOILSYS( 5, 1)     C                                   1
   7   EX      ( 1, 1)     C                                   2
   8   EXD     ( 1, 2)     C                                   1    P
   9   STEP    ( 2, 2)     C                                   1
  10   STEP2   ( 1, 1)     C                                   1

  11   TRCKRN  ( 5, 6)     C                                   1
  12   TRCKRND ( 5, 6)     C                                   1
  13   TRCKRNDD( 5, 6)     C                                   1
  14   TEST    ( 1, 1)     C                                   1
  15   TEST3   ( 3, 3)     C                                   1
 ---------------------------------------------------------------------
 /DPACS/ TIME-17:14:27 CPU-00:00:15
```

システムデータの入力	DP. I. 5
SYSIN ([／@]) $\left[\begin{matrix}\#\\ ／C\end{matrix}\right]$ ␣OSN　　@＝S, T, M, D, I	

機　能

システムデータの入力を付加されたスイッチに従って行う。

◆第１スイッチの／@はシステムデータの形式を示し，@＝S，T，M，D，Iの５種類である。第１スイッチを付加しなかった場合には，以下に示すメッセージが出力される。ここで，システムデータの形式を指定することができる。

SPECIFY TYPE OF SYSTEM DATA :

◆第２スイッチの／Cは連続系のシステムデータの入力であることを示す。第２スイッチを付加しなかった場合には，以下に示すメッセージが出力される。ここで，連続系か離散系かを指定することができる。

CONTINUOUS OR DISCRETE TYPE ? (C OR D) :

（注）第１スイッチが／Iの場合には，第２スイッチの指定は無視される。

◆／S；状態空間表現形式システム入力

$$\begin{cases} \sigma \boldsymbol{x}(t)=\boldsymbol{A}\boldsymbol{x}(t)+\boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{y}(t)=\boldsymbol{C}\boldsymbol{x}(t)+\boldsymbol{D}\boldsymbol{u}(t) \\ \boldsymbol{A}\in\mathbf{R}^{ns\times ns},\ \boldsymbol{B}\in\mathbf{R}^{ns\times m},\ \boldsymbol{C}\in\mathbf{R}^{p\times ns},\ \boldsymbol{D}\in\mathbf{R}^{p\times m} \end{cases}$$

のシステムのm, p, n_s, $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{D}$を入力する。$m, p, n_s \leq 50$。

◆／T；伝達関数表現形式システム入力

$$\begin{cases} \boldsymbol{y}(\sigma)=\boldsymbol{G}(\sigma)\cdot\boldsymbol{u}(\sigma) \\ \boldsymbol{G}(\sigma)=\dfrac{\boldsymbol{P}(\sigma)}{q(\sigma)}+\boldsymbol{D} \quad ,\ \boldsymbol{D}\in\mathbf{R}^{p\times m} \\ q(\sigma)=q_0\sigma^{nT}+q_1\sigma^{nT-1}+\cdots\cdots+q_{nT},\quad q_i\in\mathbf{R} \\ \boldsymbol{P}(\sigma)=\boldsymbol{P}_0\sigma^{nT}+\boldsymbol{P}_1\sigma^{nT-1}+\cdots\cdots+\boldsymbol{P}_{nT},\ \boldsymbol{P}_i\in\mathbf{R}^{p\times m} \end{cases}$$

のシステムを入力する。３種類の方法が用意されている。

① $n_T, q_0 \sim q_{nT}, \boldsymbol{P}_0 \sim \boldsymbol{P}_{nT}, m, p, \boldsymbol{D}$ を入力

② $q(\sigma), \boldsymbol{P}(\sigma)=\{P_{ij}(\sigma)\}, m, p, \boldsymbol{D}$ を入力

③ $\boldsymbol{G}(\sigma)=\{g_{ij}(\sigma)\}, m, p$ を入力

$m, p, n_T \leq 50$。

◆／M；マルコフパラメータ表現形式システム入力

$$\begin{cases} \boldsymbol{y}(\sigma)=\boldsymbol{G}(\sigma)\cdot\boldsymbol{u}(\sigma) \\ \boldsymbol{G}(\sigma)=\boldsymbol{D}+\boldsymbol{M}_1\sigma^{-1}+\boldsymbol{M}_2\sigma^{-2}+\cdots\cdots+\boldsymbol{M}_{n_M}\sigma^{-n_M} \\ \boldsymbol{D}\in\mathbf{R}^{p\times m},\ \boldsymbol{M}_i\in\mathbf{R}^{p\times m} \end{cases}$$

のシステムのm，p，n_M，$\boldsymbol{D}$，$\boldsymbol{M}_i$を入力する。

m，p，$n_M \leq 50$。

◆／D：微分（差分）方程式表現形式システム入力

$$\begin{cases} \boldsymbol{T}(\sigma)\boldsymbol{y}(t)=\boldsymbol{U}(\sigma)\boldsymbol{u}(t)+\boldsymbol{T}(\sigma)\boldsymbol{D}\,\boldsymbol{u}(t) \\ \boldsymbol{T}(\sigma)=\boldsymbol{T}_0\sigma^{n_D}+\boldsymbol{T}_1\sigma^{n_D-1}+\cdots\cdots+\boldsymbol{T}_{n_D}\quad \boldsymbol{T}_i\in\mathbf{R} \\ \boldsymbol{U}(\sigma)=\boldsymbol{U}_0\sigma^{n_D}+\boldsymbol{U}_1\sigma^{n_D-1}+\cdots\cdots+\boldsymbol{U}_{n_D},\ \boldsymbol{U}_i\in\mathbf{R} \\ \boldsymbol{D}\in\mathbf{R}^{p\times m} \end{cases}$$

のシステムのm，p，n_D，$\boldsymbol{D}$，$\boldsymbol{U}_i$，$\boldsymbol{T}_i$ を入力する。

m，p，$n_D \leq 50$

◆／I；入出力表現形式システム

$$\{\boldsymbol{u}(k),\ \boldsymbol{y}(k),\ k=0,1,2,\cdots\cdots\cdots,\ n_I\}$$

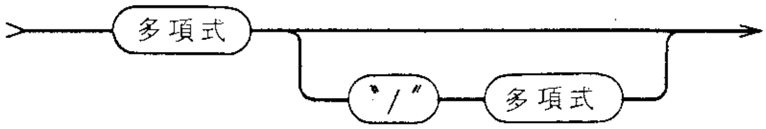
のシステムのm，p，n_I，$\boldsymbol{u}$(k)，$\boldsymbol{y}$(k)を入力する。

m，p ≤ 50，$n_I \leq 2100$.

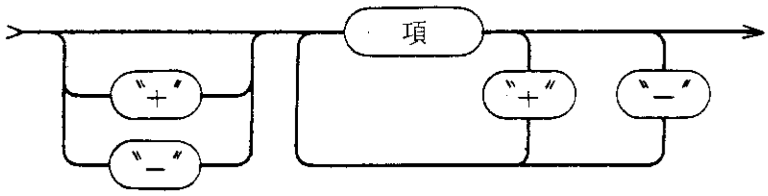
＜多項式の入力形式＞

多項式コンパイラが認識できる入力形式を以下に示す。

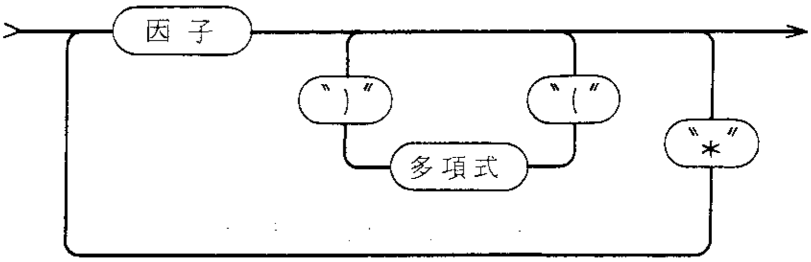
有理多項式：



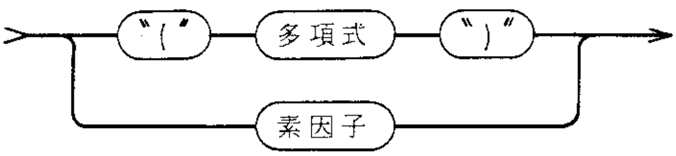
多項式：



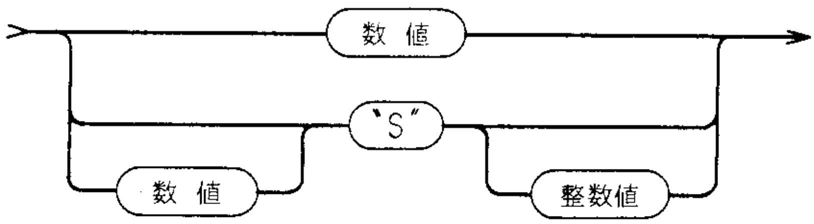
項：



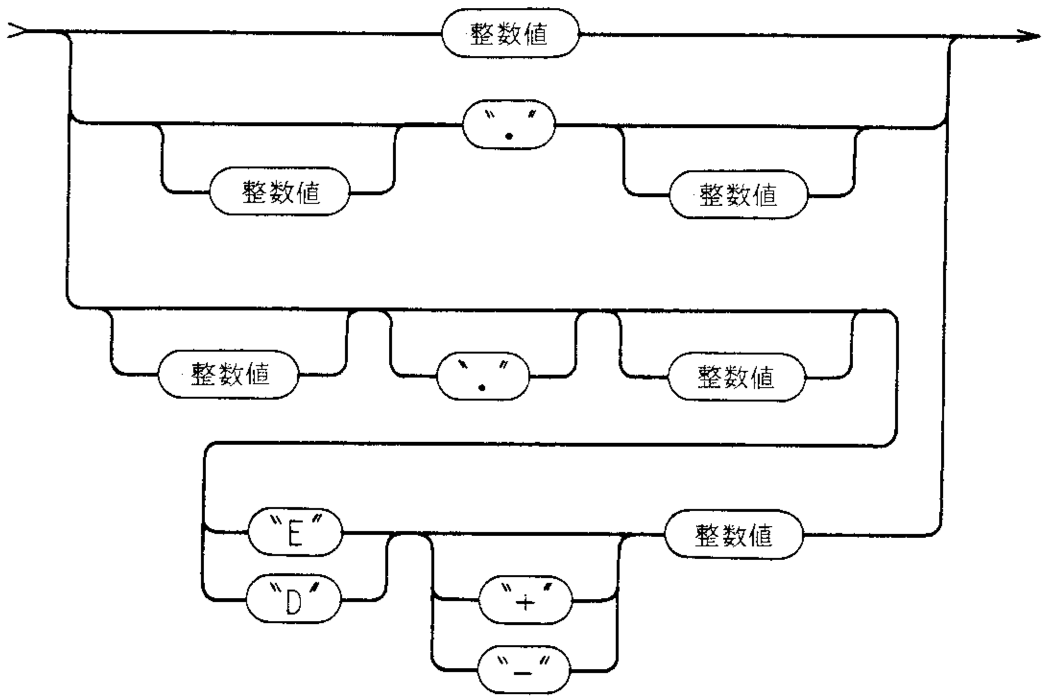
因子：



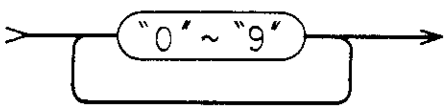
素因子：



数値：



整数値：



（注）

① 有理多項式で"／"以降を省略した場合は分母多項式は"1"と解釈される。

② 数値において，"E"あるいは"D"の直前の仮数部を省略した場合は，仮数部は"1.0"と解釈される。

③ 素因子の"S"と整数値の間および，数値内に空白を入れてはならない。（その他の場所には挿入可）

④ 入力データは80文字で3行，合計240文字まで入力することができる。なお，次の行に続く場合には，"！"をその行の終わりに付加する。

⑤ 零判定のためのεより小さな絶体値を持つ値は0.0と見做される。

⑥ S^2 はS2と表わす。

⑦ Sの多項式の形式にて入力したシステムデータも，ＤＰＡＣＳ／Ｊ内部にて変換され，係数行列の形式にて保持されている。

実 行 例

〔1〕 連続系状態空間表現システムデータの入力例

```
 /DPACS/ TIME-22:35:10 CPU-00:00:01
SYSIN/S/C EX1
 TO INPUT SYSTEMDATA
 SPECIFY NO. OF INPUTS & OUTPUTS :   2,2
 GIVE THE ORDER OF STATE SPACE :   3
 INPUT -MODE OF ( 3, 3) MATRIX   A       : _
 ( 1) ROW
0,1,0
 ( 2) ROW
1,-1,2
 ( 3) ROW
3,0,1
 TYPE OR REVISE ?   N
 INPUT -MODE OF ( 3, 2) MATRIX   B       : _
 ( 1) ROW
0,0
 ( 2) ROW
1,-2
 ( 3) ROW
0,1
 TYPE OR REVISE ?
 INPUT -MODE OF ( 2, 3) MATRIX   C       : _
 ( 1) ROW
1,0,0
 ( 2) ROW
0,1,1
 TYPE OR REVISE ?   N
 INPUT -MODE OF ( 2, 2) MATRIX   D       : Z
 TYPE OR REVISE ?   _
 REVISE PREVIOUS MATRIX ?   N
 GIVE COMMENT FOR SYSTEMDATA : _
EX1 TEST DATA
 /DPACS/ TIME-22:46:43 CPU-00:00:20
```

〔2〕 連続系状態空間表現形式システムデータの入力例（入力データの修正例）

```
 /DPACS/ TIME-22:35:11 CPU-00:00:01
SYSIN/S EXSYS2
 TO INPUT SYSTEMDATA
 CONTINUOUS OR DISCRETE TYPE ? (C OR D) :   C
 SPECIFY NO. OF INPUTS & OUTPUTS :   1,1
 GIVE THE ORDER OF STATE SPACE :   3
 INPUT -MODE OF ( 3, 3) MATRIX   A       : _
 ( 1) ROW
3,2,1
 ( 2) ROW
0,2,1
 ( 3) ROW
0,0,1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3, 1) MATRIX   B       : _
 ( 1) ROW
2
 ( 2) ROW
3
 ( 3) ROW
 ***
1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 1, 3) MATRIX   C       : _
 ( 1) ROW
1,1,1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 1, 1) MATRIX   D       : Z
 TYPE OR REVISE ?
 REVISE PREVIOUS MATRIX ? ①_
 SPECIFY THE MATRIX TO BE REVISED (A,B,C OR D) :   D
 INPUT -MODE OF ( 1, 1) MATRIX   D       : _
 ( 1) ROW
2
 TYPE OR REVISE ? _
 REVISE PREVIOUS MATRIX ?  N
 GIVE COMMENT FOR SYSTEMDATA : _
 /DPACS/ TIME-22:46:44 CPU-00:00:20
```

【説明】

① このメッセージに対して“Y”入力後又はそのままリターンキーを押下すると，システムデータの修正が行える。“N”を入力すると，次のステップに移る。

〔3〕 連続系伝達関数行列表現形式システムデータ入力例（分母子の係数行列入力例）

```
 /DPACS/ TIME-22:35:12 CPU-00:00:01
SYSIN/T/C EX10
 TO INPUT SYSTEMDATA
 SPECIFY NO. OF INPUTS & OUTPUTS :   2,2
 (1) DENOMINATOR & NUMERATOR COEFFICIENT MATRICIES
 (2) DENOMINATOR POLYNOMIAL & NUMERATOR POLYNOMIAL MATRIX
 (3) RATIONAL POLYNOMIAL MATRIX
 SPECIFY INPUT-MODE :   1
 GIVE THE ORDER OF TRANSFER FUNCTION :   2  ①
 1) DENOMINATER PART :
 Q(S)=Q0*S**N+Q1*S**(N-1)+...+Q(N-1)*S+Q(N)
 INPUT -MODE OF ( 1, 1) MATRIX  Q0        : _
 ( 1) ROW
1
 TYPE OR REVISE ?   N
 INPUT -MODE OF ( 1, 1) MATRIX  Q1        : _
 ( 1) ROW
6
 TYPE OR REVISE ?  _
 INPUT -MODE OF ( 1, 1) MATRIX  Q2        : _
 ( 1) ROW
5
 TYPE OR REVISE ?   N
 2) NUMERATER PART :
 P(S)=P0*S**N+P1*S**(N-1)+...+P(N-1)*S+P(N)
 INPUT -MODE OF ( 2, 2) MATRIX  P0        : _
 ( 1) ROW
0,0
 ( 2) ROW
0,0
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX  P1        : _
 ( 1) ROW
1,1
 ( 2) ROW
1,2
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX  P2        : _
 ( 1) ROW
4,1
 ( 2) ROW
3,2
 TYPE OR REVISE ? _
 3) DIRECT PART
 INPUT -MODE OF ( 2, 2) MATRIX  D         : Z
 TYPE OR REVISE ? _
 REVISE PREVIOUS MATRIX ?   N
 GIVE COMMENT FOR SYSTEMDATA : _
EX10 TEST DATA2
 /DPACS/ TIME-22:46:46 CPU-00:00:20
```

【説明】

① q(s), **P**(s)の式中の次数Nの入力を行う。

〔4〕 連続系伝達関数行列表現形式システムデータの入力例
（分母多項式および分子多項式行列の入力例）

```
/DPACS/ TIME-22:35:13 CPU-00:00:01
SYSIN/T/C SYS3T
TO INPUT SYSTEMDATA
SPECIFY NO. OF INPUTS & OUTPUTS :   2,2
(1) DENOMINATOR & NUMERATOR COEFFICIENT MATRICES
(2) DENOMINATOR POLYNOMIAL & NUMERATOR POLYNOMIAL MATRIX
(3) RATIONAL POLYNOMIAL MATRIX
SPECIFY INPUT-MODE :   2
1) DENOMINATOR POLYNOMIAL
Q =
(S-1)*(S-2)*(S-3)-4S-5
2) NUMERATOR POLYNOMIAL MATRIX
P(1,1) =
2S-3
P(1,2) =
3S-5
P(2,1) =
(2S-1)*(S-3)
P(2,2) =
S-1
TYPE THE POLYNOMIALS ? _

SYSINDATA FOR SYSTEM SYS3T.TO
1) DENOMINATOR POLYNOMIAL
Q =  (S-1)*(S-2)*(S-3)-4S-5
2) NUMERATOR POLYNOMIAL MATRIX
P(1,1) =  2S-3
P(1,2) =  3S-5
P(2,1) =  (2S-1)*(S-3)
P(2,2) =  S-1
REVISE THE POLYNOMIALS ?   N
EPSILON FOR COMPILATION :   1D-15
GIVE COMMENT FOR SYSTEMDATA :

/DPACS/ TIME-22:46:47 CPU-00:00:20
```

```
/DPACS/ TIME-22:35:15 CPU-00:00:01
TYPE/T SYS3T
 TO TYPE OUT SYSTEMDATA
 NORMALIZED FORM ?  N
 SYSTEMDATA SYS3T.TO      ( 2, 2, 3)

 1) DENOMINATER PART :
 Q(S)=Q0*S**N+Q1*S**(N-1)+...+Q(N-1)*S+Q(N)
   Q( 0) =    1.00000000D+00
   Q( 1) =   -6.00000000D+00
   Q( 2) =    7.00000000D+00
   Q( 3) =   -1.10000000D+01
 2) NUMERATER PART :
 P(S)=P0*S**N+P1*S**(N-1)+...+P(N-1)*S+P(N)
 --- ( 2, 2) MATRIX  P0         ---
 ( 1)  0.0               0.0
 ( 2)  0.0               0.0
 FILENAME : __
 --- ( 2, 2) MATRIX  P1         ---
 ( 1)  0.0               0.0
 ***
 ( 2)  2.00000000D+00  0.0
 FILENAME : __
 --- ( 2, 2) MATRIX  P2         ---
 ( 1)  2.00000000D+00  3.00000000D+00
 ( 2) -7.00000000D+00  1.00000000D+00
 FILENAME : __
 --- ( 2, 2) MATRIX  P3         ---
 ( 1) -3.00000000D+00 -5.00000000D+00
 ( 2)  3.00000000D+00 -1.00000000D+00
 FILENAME : __
 3) DIRECT PART
 --- ( 2, 2) MATRIX  D          ---
 ( 1)  0.0               0.0
 ( 2)  0.0               0.0
 FILENAME : __
 /DPACS/ TIME-22:46:48 CPU-00:00:20
```

〔5〕 離散系伝達関数行列表現形式システムデータの入力例
（有理多項式行列入力の例）

```
 /DPACS/ TIME-22:35:16 CPU-00:00:01
SYSIN/T SYS23D
 TO INPUT SYSTEMDATA
 CONTINUOUS OR DISCRETE TYPE ? (C OR D) :   D
 GIVE SAMPLING INTERVAL :   0.002
 SPECIFY NO. OF INPUTS & OUTPUTS :   2,2
 (1) DENOMINATOR & NUMERATOR COEFFICIENT MATRICES
 (2) DENOMINATOR POLYNOMIAL & NUMERATOR POLYNOMIAL MATRIX
 (3) RATIONAL POLYNOMIAL MATRIX
 SPECIFY INPUT-MODE :   3
 G(1,1) =
S/S+1  ①
 G(1,2) =
-1/2S-2 ②
 G(2,1) =
S-1/S(S+1) ③
 G(2,2) =
S+2/S2-1  ④
 TYPE THE RATIONAL POLYNOMIAL MATRIX ?   N
 REVISE THE RATIONAL POLYNOMIALS ?   N
 EPSILON FOR COMPILATION :   1D-10
 GIVE COMMENT FOR SYSTEMDATA :

 /DPACS/ TIME-22:46:49 CPU-00:00:20
```

【説明】

①，②，③，④はそれぞれ次を意味する。

① S／（S＋1）

② －1／（2S－2）

③ （S－1）／S（S＋1）

④ （S＋2）／（S^2 －1）

```
 /DPACS/ TIME-22:35:18 CPU-00:00:01
TYPE/T/O SYS23D
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA SYS23D.TO    ( 2, 2, 3)

 1) DENOMINATER PART :
 Q(S)=Q0*S**N+Q1*S**(N-1)+...+Q(N-1)*S+Q(N)
   Q( 0) =    2.000000000D+00
   Q( 1) =    0.0
   Q( 2) =   -2.000000000D+00
   Q( 3) =    0.0
 2) NUMERATER PART :
 P(S)=P0*S**N+P1*S**(N-1)+...+P(N-1)*S+P(N)
 --- ( 2, 2) MATRIX  P0          ---
 ( 1)  2.000000000D+00  0.0
 ( 2)  0.0              0.0
 FILENAME : __
 --- ( 2, 2) MATRIX  P1          ---
 ( 1) -2.000000000D+00 -1.000000000D+00
 ( 2)  2.000000000D+00  2.000000000D+00
 ***
 FILENAME : __
 --- ( 2, 2) MATRIX  P2          ---
 ( 1)  0.0             -1.000000000D+00
 ( 2) -4.000000000D+00  4.000000000D+00
 FILENAME : __
 --- ( 2, 2) MATRIX  P3          ---
 ( 1)  0.0              0.0
 ( 2)  2.000000000D+00  0.0
 FILENAME : __
 3) DIRECT PART
 --- ( 2, 2) MATRIX  D           ---
 ( 1)  0.0              0.0
 ( 2)  0.0              0.0
 FILENAME : __
 /DPACS/ TIME-22:46:59 CPU-00:00:20
```

〔6〕 伝達関数行列表現形式システムデータの入力例
（多項式が複数行にまたがる場合の例）

```
/DPACS/ TIME-22:35:26 CPU-00:00:01
SYSIN/T/C EX31
 TO INPUT SYSTEMDATA
 SPECIFY NO. OF INPUTS & OUTPUTS :   2,2
 (1) DENOMINATOR & NUMERATOR COEFFICIENT MATRICES
 (2) DENOMINATOR POLYNOMIAL & NUMERATOR POLYNOMIAL MATRIX
 (3) RATIONAL POLYNOMIAL MATRIX
 SPECIFY INPUT-MODE :   2
 1) DENOMINATOR POLYNOMIAL
 Q =
(S-1)  (S-2)  (S-3)  (S2-4S-5) !    ①
(S-4)  (S-5)
 2) NUMERATOR POLYNOMIAL MATRIX
 P(1,1) =
S-2
 P(1,2) =
S2-3S-2
 P(2,1) =
S3-2S2-3
 P(2,2) =
 ***
S-2
 TYPE THE POLYNOMIALS ?   Y
 SYSINDATA FOR SYSTEM EX31.TO
 1) DENOMINATOR POLYNOMIAL
 Q =  (S-1)  (S-2)  (S-3)  (S2-4S-5) (S-4)  (S-5)   ②
 2) NUMERATOR POLYNOMIAL MATRIX
 P(1,1) =  S-2
 P(1,2) =  S2-3S-2③
 P(2,1) =  S3-2S2-3④
 P(2,2) =  S-2
 REVISE THE POLYNOMIALS ?   N
 EPSILON FOR COMPILATION :   1D-10
 GIVE COMMENT FOR SYSTEMDATA :

/DPACS/ TIME-22:47:00 CPU-00:00:20
```

【説明】

① 複数行にまたがる場合の入力方法

② 表示が複数行にまたがる場合は，“¬”が行の最後に表示される。

③ S^2-3S-2を意味する。

④ S^3-2S^2-3を意味する。

〔7〕 連続系マルコフパラメータ表現形式システムデータの入力例

```
/DPACS/ TIME-22:35:28 CPU-00:00:01
SYSIN/M/C SYS3
 TO INPUT SYSTEMDATA
 SPECIFY NO. OF INPUTS & OUTPUTS :   2,2
 GIVE THE NUMBER OF MARKOV PARAMETERS :   3
 INPUT -MODE OF ( 2, 2) MATRIX  D        : _
 ( 1) ROW
1,1
 ( 2) ROW
0,-1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX  M1       : _
 ( 1) ROW
0,1
 ( 2) ROW
1,-2
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX  M2       : _
 ( 1) ROW
0,-1
 ( 2) ROW
2,1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX  M3       : _
 ( 1) ROW
1,2
 ( 2) ROW
0,1
 TYPE OR REVISE ? _
 REVISE PREVIOUS MATRIX ?   N
 GIVE COMMENT FOR SYSTEMDATA : _
 /DPACS/ TIME-22:47:02 CPU-00:00:20
```

〔8〕 連続系微分（差分）方程式表現形式システムデータの入力例

```
 /DPACS/ TIME-22:35:29 CPU-00:00:01
SYSIN/D/C EX8
 TO INPUT SYSTEMDATA
 SPECIFY NO. OF INPUTS & OUTPUTS :   2,2
 GIVE THE OBSERVABILITY INDEX :   3
 1) DENOMINATER PART :
 T(S)=T0*S**N+T1*S**(N-1)+...+T(N-1)*S+T(N)
 INPUT -MODE OF ( 2, 2) MATRIX  T0       : _
 ( 1) ROW
1,0
 ( 2) ROW
0,1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX  T1       : Z
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX  T2       : _
 ( 1) ROW
-2,0
 ( 2) ROW
0,-2
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX  T3       : _
 ( 1) ROW
-5,0
 ( 2) ROW
0,-5
 TYPE OR REVISE ? _
 2) NUMERATER PART :
 U(S)=U0*S**N+U1*S**(N-1)+...+U(N-1)*S+U(N)
 INPUT -MODE OF ( 2, 2) MATRIX  U0       : Z
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX  U1       : _
 ( 1) ROW
0,0
 ( 2) ROW
1,-1
 TYPE OR REVISE ?  N
 INPUT -MODE OF ( 2, 2) MATRIX  U2       : _
 ( 1) ROW
1,-2
 ( 2) ROW
-1,5
 TYPE OR REVISE ?  N
 INPUT -MODE OF ( 2, 2) MATRIX  U3       : _
 ( 1) ROW
-1,4
 ( 2) ROW
3,-7
 TYPE OR REVISE ?  _
 3) DIRECT PART
 INPUT -MODE OF ( 2, 2) MATRIX  D        : Z
 TYPE OR REVISE ?  _
 REVISE PREVIOUS MATRIX ?  N
 GIVE COMMENT FOR SYSTEMDATA : _
EX8 TEST DATA3
 /DPACS/ TIME-22:47:03 CPU-00:00:20
```

〔9〕 入出力データ表現形式システムデータの入力例

```
 /DPACS/ TIME-22:35:31 CPU-00:00:01
SYSIN/I SYS1
 TO INPUT SYSTEMDATA
 GIVE SAMPLING INTERVAL :  0.001
 SPECIFY NO. OF INPUTS & OUTPUTS :  1,1
 GIVE THE NUMBER OF INTERVALS :  2
 INPUT -MODE OF ( 1,   2) TIME SERIES INPUT     : _
 === INITIAL DATA ===
 INPUT -MODE OF ( 1, 1) MATRIX X(0)       : Z
 TYPE OR REVISE ? _
 === INTERVAL (   1-   2) ===
 INPUT -MODE OF ( 2, 1) MATRIX DATA       : _
 ( 1) ROW
1
 ( 2) ROW
1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 1,   2) TIME SERIES OUTPUT    : _
 === INITIAL DATA ===
 INPUT -MODE OF ( 1, 1) MATRIX X(0)       : _
 ( 1) ROW
0
 TYPE OR REVISE ? _
 === INTERVAL (   1-   2) ===
 INPUT -MODE OF ( 2, 1) MATRIX DATA       : _
 ( 1) ROW
2
 ( 2) ROW
2
 TYPE OR REVISE ? _
 REVISE PREVIOUS TIME SERIES ?  N
 GIVE COMMENT FOR SYSTEMDATA :
TYPE(I)
 /DPACS/ TIME-22:47:04 CPU-00:00:20
```

システムデータの出力	DP. I. 6
TYPE ([／@]) ([／N ／O]) ␣SN　　@＝S, T, M, D, I	

機　能

システムデータを画面に出力する。

◆第1スイッチの／@は，システムデータの形式を示し，@＝S，T，M，D，Iの5種類である。第1スイッチを指定しなかった場合には，以下に示すメッセージが出力される。ここで，システムデータの形式を指定することができる。

SPECIFY TYPE OF SYSTEM DATA :

◆第2スイッチは，システムデータを正規化して出力する（／N）か正規化せずに出力する（／O）かの指定を行うものである。第2スイッチは，第1スイッチが／Dまたは／Tの場合にのみ有効であり，その他の場合は無視される。第2スイッチを省略した場合には，以下のメッセージが出力される。ここで，システムデータを正規化する（Y）か否か（Nまたは⏎のみを押下）を指定することができる。

NORMALIZED FORM ?

実行例

〔1〕 状態空間表現形式システムデータ出力例

```
 /DPACS/ TIME-22:35:32 CPU-00:00:01
TYPE/S EX1
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1.SO        ( 2, 2, 3)

 --- ( 3, 3) MATRIX    A        ---
 ( 1)  0.0                  1.00000000D+00  0.0
 ( 2)  1.00000000D+00 -1.00000000D+00  2.00000000D+00
 ( 3)  3.00000000D+00  0.0             1.00000000D+00
 FILENAME : _
 --- ( 3, 2) MATRIX    B        ---
 ( 1)  0.0             0.0
 ( 2)  1.00000000D+00 -2.00000000D+00
 ( 3)  0.0             1.00000000D+00
 FILENAME : _
 --- ( 2, 3) MATRIX    C        ---
 ( 1)  1.00000000D+00  0.0             0.0
 ( 2)  0.0             1.00000000D+00  1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX    D        ---
 ( 1)  0.0             0.0
 ( 2)  0.0             0.0
 FILENAME : _
 /DPACS/ TIME-22:47:05 CPU-00:00:20
```

〔2〕 伝達関数行列表現形式システムデータ出力例

```
/DPACS/ TIME-22:35:33 CPU-00:00:01
TYPE/T/O SYS2
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA SYS2.TO      ( 1, 1, 3)

 1) DENOMINATER PART :
 Q(S)=Q0*S**N+Q1*S**(N-1)+...+Q(N-1)*S+Q(N)
   Q( 0) =    3.00000000D+00
   Q( 1) =    5.00000000D+00
   Q( 2) =    1.00000000D+00
   Q( 3) =    4.00000000D+00
 2) NUMERATER PART :
 P(S)=P0*S**N+P1*S**(N-1)+...+P(N-1)*S+P(N)
 --- ( 1, 1) MATRIX  P0       ---
 ( 1)  3.00000000D+00
 FILENAME :_
 --- ( 1, 1) MATRIX  P1       ---
 ( 1)  5.00000000D+00
 FILENAME :_
 --- ( 1, 1) MATRIX  P2       ---
 ***
 ( 1)  5.00000000D+00
 FILENAME :_
 --- ( 1, 1) MATRIX  P3       ---
 ( 1) -3.00000000D+00
 FILENAME :_
 3) DIRECT PART
 --- ( 1, 1) MATRIX  D        ---
 ( 1) -2.00000000D+00
 FILENAME :_
 /DPACS/ TIME-22:47:06 CPU-00:00:20
```

〔3〕 正規形伝達関数行列表現形式システムデータ出力例

```
 /DPACS/ TIME-22:35:34 CPU-00:00:01
TYPE/T/N SYS2
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA SYS2.TO      ( 1, 1, 3)

 1) DENOMINATER PART :
 Q(S)=Q0*S**N+Q1*S**(N-1)+...+Q(N-1)*S+Q(N)
   Q( 0) =    1.00000000D+00
   Q( 1) =    1.66666667D+00
   Q( 2) =    3.33333333D-01
   Q( 3) =    1.33333333D+00
 2) NUMERATER PART :
 P(S)=P0*S**N+P1*S**(N-1)+...+P(N-1)*S+P(N)
 --- ( 1, 1) MATRIX  P0        ---
 ( 1)  0.0
 FILENAME : _
 --- ( 1, 1) MATRIX  P1        ---
 ( 1)  1.48029737D-16
 FILENAME : _
 --- ( 1, 1) MATRIX  P2        ---
 ***
 ( 1)  1.33333333D+00
 FILENAME :_
 --- ( 1, 1) MATRIX  P3        ---
 ( 1) -2.33333333D+00
 FILENAME :_
 3) DIRECT PART
 --- ( 1, 1) MATRIX  D         ---
 ( 1) -1.00000000D+00
 FILENAME :_
 /DPACS/ TIME-22:47:08 CPU-00:00:20
```

〔4〕 マルコフパラメータ表現形式システムデータ出力例

```
 /DPACS/ TIME-22:35:35 CPU-00:00:01
TYPE/M SYS3
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA SYS3.MO      ( 2, 2, 3)

 --- ( 2, 2) MATRIX  D         ---
 ( 1)  1.00000000D+00  1.00000000D+00
 ( 2)  0.0            -1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  M1        ---
 ( 1)  0.0             1.00000000D+00
 ( 2)  1.00000000D+00 -2.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  M2        ---
 ( 1)  0.0            -1.00000000D+00
 ( 2)  2.00000000D+00  1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  M3        ---
 ( 1)  1.00000000D+00  2.00000000D+00
 ( 2)  0.0             1.00000000D+00
 FILENAME : _
 /DPACS/ TIME-22:47:09 CPU-00:00:20
```

〔5〕 微分（差分）方程式表現形式システムデータ出力例

```
/DPACS/ TIME-22:35:37 CPU-00:00:01
TYPE/D/O SYS3
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA SYS3.DO      ( 2, 2, 2)

 1) DENOMINATER PART :
 T(S)=T0*S**N+T1*S**(N-1)+...+T(N-1)*S+T(N)
 --- ( 2, 2) MATRIX  T0        ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  1.00000000D+00  1.00000000D+00
 FILENAME :_
 --- ( 2, 2) MATRIX  T1        ---
 ( 1)  1.00000000D+00  0.0
 ( 2) -2.00000000D+00  1.00000000D+00
 FILENAME :_
 --- ( 2, 2) MATRIX  T2        ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  0.0             1.00000000D+00
 FILENAME :_
 2) NUMERATER PART :
 ***
 U(S)=U0*S**N+U1*S**(N-1)+...+U(N-1)*S+U(N)
 --- ( 2, 2) MATRIX  U0        ---
 ( 1)  2.00000000D+00  2.00000000D+00
 ( 2)  1.00000000D+00  3.00000000D+00
 FILENAME :_
 --- ( 2, 2) MATRIX  U1        ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  0.0             1.00000000D+00
 FILENAME :_
 --- ( 2, 2) MATRIX  U2        ---
 ( 1)  3.00000000D+00  0.0
 ( 2)  0.0             5.00000000D+00
 FILENAME :_
 3) DIRECT PART
 --- ( 2, 2) MATRIX  D         ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  0.0             1.00000000D+00
 FILENAME :_
 /DPACS/ TIME-22:47:10 CPU-00:00:20
```

〔6〕 正規形微分（差分）方程式表現形式システムデータ出力例

```
 /DPACS/ TIME-22:35:39 CPU-00:00:01
TYPE/D/N SYS3
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA SYS3.DO      ( 2, 2, 2)

 1) DENOMINATER PART :
 T(S)=T0*S**N+T1*S**(N-1)+...+T(N-1)*S+T(N)
 --- ( 2, 2) MATRIX  T0        ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  0.0             1.00000000D+00
 FILENAME :_
 --- ( 2, 2) MATRIX  T1        ---
 ( 1)  1.00000000D+00  0.0
 ( 2) -3.00000000D+00  1.00000000D+00
 FILENAME :_
 --- ( 2, 2) MATRIX  T2        ---
 ( 1)  1.00000000D+00  0.0
 ( 2) -1.00000000D+00  1.00000000D+00
 FILENAME :_
 2) NUMERATER PART :
 ***
 U(S)=U0*S**N+U1*S**(N-1)+...+U(N-1)*S+U(N)
 --- ( 2, 2) MATRIX  U0        ---
 ( 1)  0.0             0.0
 ( 2)  0.0             0.0
 FILENAME :_
 --- ( 2, 2) MATRIX  U1        ---
 ( 1) -1.00000000D+00 -2.00000000D+00
 ( 2)  6.00000000D+00  6.00000000D+00
 FILENAME :_
 --- ( 2, 2) MATRIX  U2        ---
 ( 1)  1.00000000D+00 -2.00000000D+00
 ( 2)  0.0             6.00000000D+00
 FILENAME :_
 3) DIRECT PART
 --- ( 2, 2) MATRIX  D         ---
 ( 1)  3.00000000D+00  2.00000000D+00
 ( 2) -1.00000000D+00  2.00000000D+00
 FILENAME :_
 /DPACS/ TIME-22:47:11 CPU-00:00:20
```

〔7〕 入出力データ表現形式システムデータ出力例

```
 /DPACS/ TIME-17:13:26 CPU-00:00:02
TYPE/I SYS1
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA SYS1.IO       ( 1, 1,   2)

 *** ( 1,    2) TIME SERIES INPUT     ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 1, 1) MATRIX X(0)       ---
 ( 1)  0.0
 FILENAME : _
 === INTERVAL (    1-    2) ===
 --- ( 1, 2) MATRIX DATA       ---
 ( 1)  1.00000000D+00  1.00000000D+00
 FILENAME : _
 OUTPUT-FILENAME : _
 *** ( 1,    2) TIME SERIES OUTPUT    ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 1, 1) MATRIX X(0)       ---
 ( 1)  0.0
 FILENAME : _
 === INTERVAL (    1-    2) ===
 --- ( 1, 2) MATRIX DATA       ---
 ( 1)  2.00000000D+00  2.00000000D+00
 FILENAME : _
 OUTPUT-FILENAME : _
 /DPACS/ TIME-17:18:40 CPU-00:00:13
```

システムデータの修正	DP. I. 7
REVISE ([／@]) ␣SN	@＝S, T, M, D, I

機　能

指定したシステムの修正を行う。このコマンドにて修正できる項目は，

① システムデータ

② サンプリング・ピッチ

③ コメント

である。尚，サンプリング・ピッチについての修正は，既にシステムが保有しているサンプリング・ピッチのみを書き換えるものであり，システムデータに対して補間処理等の修正は加えないので注意を要する。

◆第1スイッチの／@は，システムデータの形式を示し，@＝S，T，M，D，Iの5種類である。第1スイッチを付加しなかった場合には以下のメッセージが出力される。ここで，システムデータの形式を指定することができる。

SPECIFY TYPE OF SYSTEM DATA :

実行例

〔1〕 状態空間表現形式システムデータのA行列修正例

```
/DPACS/ TIME-17:13:53 CPU-00:00:02
REVISE/S SYS1
 TO REVISE SYSTEMDATA
 WHAT'S REVISED, (1) SYSTEMDATA, (2) DELT, (3) COMMENT ? :    1
 SPECIFY THE MATRIX TO BE REVISED (A,B,C OR D) :   A
 INPUT -MODE OF ( 3, 3) MATRIX   A      : R
 SPECIFY (I,J) :  1,2
   A      ( 1, 2)=  0.0            REVISE TO   -1
 END ? _
 TYPE OR REVISE ? _
 WHAT'S REVISED, (1) SYSTEMDATA, (2) DELT, (3) COMMENT ? :
 /DPACS/ TIME-17:18:42 CPU-00:00:13
```

〔2〕 伝達関数行列表現形式システムデータのD行列修正例

```
/DPACS/ TIME-17:13:55 CPU-00:00:02
REVISE/T SYS2
 TO REVISE SYSTEMDATA
 WHAT'S REVISED, (1) SYSTEMDATA, (2) DELT, (3) COMMENT ? :    1
 SPECIFY THE MATRIX TO BE REVISED
 (Q0-Q3 , P0-P3  OR D) :   D
 INPUT -MODE OF ( 1, 1) MATRIX  D        : Z
 TYPE OR REVISE ? _
 WHAT'S REVISED, (1) SYSTEMDATA, (2) DELT, (3) COMMENT ? :    2
 GIVE NEW SAMPLING INTERVAL :   0.01
 WHAT'S REVISED, (1) SYSTEMDATA, (2) DELT, (3) COMMENT ? :    _
 /DPACS/ TIME-17:18:45 CPU-00:00:13
```

〔3〕 マルコフパラメータ表現形式システムデータのマルコフパラメータ修正例

```
/DPACS/ TIME-17:13:57 CPU-00:00:02
REVISE/M SYS3
 TO REVISE SYSTEMDATA
 WHAT'S REVISED, (1) SYSTEMDATA, (2) DELT, (3) COMMENT ? :    1
 SPECIFY THE MATRIX TO BE REVISED (D OR M1-M3 ) :  M1
 INPUT -MODE OF ( 2, 2) MATRIX  M1       : T
 --- ( 2, 2) MATRIX  M1       ---
 ( 1)  0.0             1.000000000D+00
 ( 2)  1.000000000D+00 -2.000000000D+00
 TYPE OR REVISE ?   R
 SPECIFY (I,J) :  2,2
  M1      ( 2, 2)= -2.000000000D+00  REVISE TO   -1
 END ?
 TYPE OR REVISE ?   T
 --- ( 2, 2) MATRIX  M1       ---
 ( 1)  0.0             1.000000000D+00
 ( 2)  1.000000000D+00 -1.000000000D+00
 TYPE OR REVISE ? _
 WHAT'S REVISED, (1) SYSTEMDATA, (2) DELT, (3) COMMENT ? :    3
 GIVE NEW COMMENT :
REVISE
 WHAT'S REVISED, (1) SYSTEMDATA, (2) DELT, (3) COMMENT ? :    _
 /DPACS/ TIME-17:18:47 CPU-00:00:13
```

〔4〕 微分（差分）方程式表現形式システムデータの修正例

```
 /DPACS/ TIME-17:13:59 CPU-00:00:02
REVISE/D SYS3
 TO REVISE SYSTEMDATA
 WHAT'S REVISED, (1) SYSTEMDATA, (2) DELT, (3) COMMENT ? :   1
 SPECIFY THE MATRIX TO BE REVISED
 (TO-T2 , UO-U2  OR D) :   D
 INPUT -MODE OF ( 2, 2) MATRIX  D        : _
 ( 1) ROW
4,5
 ( 2) ROW
2,0
 TYPE OR REVISE ? _
 WHAT'S REVISED, (1) SYSTEMDATA, (2) DELT, (3) COMMENT ? : _
 KEQ56650I TIME-14:09:17 CPU-00:00:33 SERVICE-305395 SESSION-00:06:36
6,1988
 /DPACS/ TIME-17:18:48 CPU-00:00:13

 /DPACS/ TIME-17:14:00 CPU-00:00:02
TYPE/D/O SYS3
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA SYS3.DO      ( 2, 2, 2)

 1) DENOMINATER PART :
 T(S)=TO*S**N+T1*S**(N-1)+...+T(N-1)*S+T(N)
 --- ( 2, 2) MATRIX  TO        ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  1.00000000D+00  1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  T1        ---
 ( 1)  1.00000000D+00  0.0
 ( 2) -2.00000000D+00  1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  T2        ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  0.0             1.00000000D+00
 FILENAME : _
 2) NUMERATER PART :
 ***
 U(S)=UO*S**N+U1*S**(N-1)+...+U(N-1)*S+U(N)
 --- ( 2, 2) MATRIX  UO        ---
 ( 1)  2.00000000D+00  2.00000000D+00
 ( 2)  1.00000000D+00  3.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  U1        ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  0.0             1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  U2        ---
 ( 1)  3.00000000D+00  0.0
 ( 2)  0.0             5.00000000D+00
 FILENAME : _
 3) DIRECT PART
 --- ( 2, 2) MATRIX  D         ---
 ( 1)  4.00000000D+00  5.00000000D+00
 ( 2)  2.00000000D+00  0.0
 FILENAME : _
 /DPACS/ TIME-17:18:50 CPU-00:00:13
```

〔5〕 入出力データ表現形式システムデータ修正例

```
/DPACS/ TIME-17:14:01 CPU-00:00:02
TYPE/I EX11DG
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX11DG.IO    ( 2, 2, 200)

 *** ( 2, 200) TIME SERIES INPUT     ***
 ALL DATA ?   N
 STARTING POINT ? (0- 200) :   0
 HOW MANY POINTS ? (1- 201) :   50
 === INITIAL DATA ===
 --- ( 2, 1) MATRIX X(0)       ---
 ( 1)  3.00000000D+00
 ( 2)  3.00000000D+00
 FILENAME : _
 === INTERVAL (    1-   49) ===
 --- ( 2,49) MATRIX DATA       ---
 ( 1) -3.00000000D+00  3.00000000D+00 -3.00000000D+00 -3.00000000D+00
 ( 2)  3.00000000D+00  3.00000000D+00  3.00000000D+00 -3.00000000D+00
 ( 1)  3.00000000D+00  3.00000000D+00 -3.00000000D+00 -3.00000000D+00
 ( 2) -3.00000000D+00  3.00000000D+00 -3.00000000D+00  3.00000000D+00
 ( 1)  3.00000000D+00  3.00000000D+00  3.00000000D+00 -3.00000000D+00
 ***
 ( 2) -3.00000000D+00  3.00000000D+00  3.00000000D+00  3.00000000D+00
 ( 1)  3.00000000D+00  3.00000000D+00  3.00000000D+00 -3.00000000D+00
 ( 2) -3.00000000D+00 -3.00000000D+00  3.00000000D+00  3.00000000D+00
 ( 1)  3.00000000D+00 -3.00000000D+00 -3.00000000D+00  3.00000000D+00
 ( 2) -3.00000000D+00  3.00000000D+00 -3.00000000D+00 -3.00000000D+00
 ( 1) -3.00000000D+00  3.00000000D+00  3.00000000D+00 -3.00000000D+00
 ( 2) -3.00000000D+00  3.00000000D+00 -3.00000000D+00 -3.00000000D+00
 ( 1) -3.00000000D+00 -3.00000000D+00  3.00000000D+00  3.00000000D+00
 ( 2)  3.00000000D+00  3.00000000D+00  3.00000000D+00  3.00000000D+00
 ( 1) -3.00000000D+00  3.00000000D+00  3.00000000D+00  3.00000000D+00
 ( 2) -3.00000000D+00 -3.00000000D+00 -3.00000000D+00  3.00000000D+00
 ( 1)  3.00000000D+00 -3.00000000D+00  3.00000000D+00  3.00000000D+00
 ( 2) -3.00000000D+00  3.00000000D+00 -3.00000000D+00 -3.00000000D+00
 ( 1) -3.00000000D+00  3.00000000D+00 -3.00000000D+00  3.00000000D+00
 ( 2) -3.00000000D+00 -3.00000000D+00  3.00000000D+00  3.00000000D+00
 ( 1)  3.00000000D+00 -3.00000000D+00  3.00000000D+00  3.00000000D+00
 ( 2) -3.00000000D+00 -3.00000000D+00 -3.00000000D+00 -3.00000000D+00
 ( 1) -3.00000000D+00 -3.00000000D+00  3.00000000D+00 -3.00000000D+00
 ( 2) -3.00000000D+00  3.00000000D+00 -3.00000000D+00 -3.00000000D+00
 ( 1) -3.00000000D+00
 ( 2) -3.00000000D+00
 FILENAME : _
 OUTPUT-FILENAME : _
```

```
*** ( 2, 200) TIME SERIES OUTPUT   ***
ALL DATA ?   N
STARTING POINT ? (0- 200) :   0
HOW MANY POINTS ? (1- 201) :   50
=== INITIAL DATA ===
--- ( 2, 1) MATRIX X(0)      ---
( 1) -6.00000000D+00
( 2)  1.00000000D+01
FILENAME : __
=== INTERVAL (    1-   49) ===
--- ( 2,49) MATRIX DATA      ---
( 1) -5.96108836D+00 -5.92236782D+00 -5.88383740D+00 -5.84549620D+00
( 2)  9.95910532D+00  9.91242958D+00  9.87197165D+00  9.82573048D+00
( 1) -5.80737018D+00 -5.76945523D+00 -5.73172337D+00 -5.69418867D+00
( 2)  9.77372593D+00  9.72795981D+00  9.68840998D+00  9.63708437D+00
( 1) -5.65686506D+00 -5.61973652D+00 -5.58278710D+00 -5.54600397D+00
( 2)  9.59199081D+00  9.54711919D+00  9.50845941D+00  9.46999845D+00
( 1) -5.50940128D+00 -5.47299008D+00 -5.43676642D+00 -5.40071738D+00
( 2)  9.42574429D+00  9.38170789D+00  9.33789113D+00  9.30028094D+00
( 1) -5.36484511D+00 -5.32916069D+00 -5.29366317D+00 -5.25836658D+00
( 2)  9.25687333D+00  9.21367923D+00  9.17069759D+00  9.12193634D+00
( 1) -5.22326686D+00 -5.18836306D+00 -5.15364216D+00 -5.11908833D+00
( 2)  9.07939737D+00  9.03107959D+00  8.99496990D+00  8.95305831D+00
***
( 1) -5.08472766D+00 -5.05056213D+00 -5.01657880D+00 -4.98276176D+00
( 2)  8.90536479D+00  8.86388525D+00  8.82260666D+00  8.78751903D+00
( 1) -4.94909527D+00 -4.91560552D+00 -4.88230356D+00 -4.84917348D+00
( 2)  8.75261246D+00  8.70590699D+00  8.66541357D+00  8.62512217D+00
( 1) -4.81620245D+00 -4.78338969D+00 -4.75074938D+00 -4.71828065D+00
( 2)  8.59101982D+00  8.55110557D+00  8.51138747D+00  8.47186453D+00
( 1) -4.68597963D+00 -4.65386045D+00 -4.62192216D+00 -4.59015189D+00
( 2)  8.43253878D+00  8.38741822D+00  8.34850181D+00  8.30977659D+00
( 1) -4.55853681D+00 -4.52707319D+00 -4.49578719D+00 -4.46467789D+00
( 2)  8.27722962D+00  8.23886302D+00  8.19469684D+00  8.15673007D+00
( 1) -4.43372943D+00 -4.40295599D+00 -4.37235962D+00 -4.34192449D+00
( 2)  8.11895274D+00  8.07537291D+00  8.03798658D+00  8.00078383D+00
( 1) -4.31166176D+00
( 2)  7.95777573D+00
FILENAME : __
OUTPUT-FILENAME : __
/DPACS/ TIME-17:18:52 CPU-00:00:13
```

```
 /DPACS/ TIME-17:14:03 CPU-00:00:02
REVISE/I EX11DG
 TO REVISE SYSTEMDATA
 WHAT'S REVISED, (1) SYSTEMDATA, (2) DELT, (3) COMMENT ? :   1
 SPECIFY THE TIME SERIES TO BE REVISED (I OR O) :  I
 STARTING POINT ? (0- 200) :   20
 HOW MANY POINTS ? (1-50) :   10 ①
 === INTERVAL (  20-  29) ===
 INPUT -MODE OF (10, 2) MATRIX DATA      : _
 ( 1) ROW           ②
3,4
 ( 2) ROW
2,3
 ( 3) ROW
4,-4
 ( 4) ROW
2,2
 ( 5) ROW
5,5
 ( 6) ROW
 ***
3,4
 ( 7) ROW
4,6
 ( 8) ROW
6,1
 ( 9) ROW
1,0
 (10) ROW
3,2
 TYPE OR REVISE ? __
 WHAT'S REVISED, (1) SYSTEMDATA, (2) DELT, (3) COMMENT ? : _
 /DPACS/ TIME-17:18:54 CPU-00:00:13
```

【説明】

① 一度に修正できる入出力表現形式システムデータの時間点数は，最大50である。従って，50以上に亘って修正を加え様とする場合には，再度，REVISEコマンドの入力から行う必要がある。

② 修正時系列データの入力形式にて入力することになるが，その場合，行列データは転置して入力する様になる。

```
/DPACS/ TIME-17:14:04 CPU-00:00:02
TYPE/I EX11DG
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX11DG.IO    ( 2, 2, 200)

 *** ( 2, 200) TIME SERIES INPUT     ***
 ALL DATA ?  N
 STARTING POINT ? (0- 200) :   0
 HOW MANY POINTS ? (1- 201) :   50
 === INITIAL DATA ===
 --- ( 2, 1) MATRIX X(0)       ---
 ( 1)  3.00000000D+00
 ( 2)  3.00000000D+00
 FILENAME : __
 === INTERVAL (    1-   49) ===
 --- ( 2,49) MATRIX DATA       ---
 ( 1) -3.00000000D+00  3.00000000D+00 -3.00000000D+00 -3.00000000D+00
 ( 2)  3.00000000D+00  3.00000000D+00  3.00000000D+00 -3.00000000D+00
 ( 1)  3.00000000D+00  3.00000000D+00 -3.00000000D+00 -3.00000000D+00
 ( 2) -3.00000000D+00  3.00000000D+00 -3.00000000D+00  3.00000000D+00
 ( 1)  3.00000000D+00  3.00000000D+00  3.00000000D+00 -3.00000000D+00
 ***
 ( 2) -3.00000000D+00  3.00000000D+00  3.00000000D+00  3.00000000D+00
 ( 1)  3.00000000D+00  3.00000000D+00  3.00000000D+00 -3.00000000D+00
 ( 2) -3.00000000D+00 -3.00000000D+00  3.00000000D+00  3.00000000D+00
 ( 1)  3.00000000D+00 -3.00000000D+00 -3.00000000D+00  3.00000000D+00
 ( 2) -3.00000000D+00  3.00000000D+00 -3.00000000D+00  4.00000000D+00
 ( 1)  2.00000000D+00  4.00000000D+00  2.00000000D+00  5.00000000D+00
 ( 2)  3.00000000D+00 -4.00000000D+00  2.00000000D+00  5.00000000D+00
 ( 1)  3.00000000D+00  4.00000000D+00  6.00000000D+00  1.00000000D+00
 ( 2)  4.00000000D+00  6.00000000D+00  1.00000000D+00  0.0
 ( 1)  3.00000000D+00  3.00000000D+00  3.00000000D+00  3.00000000D+00
 ( 2)  2.00000000D+00 -3.00000000D+00 -3.00000000D+00  3.00000000D+00
 ( 1)  3.00000000D+00 -3.00000000D+00  3.00000000D+00  3.00000000D+00
 ( 2) -3.00000000D+00  3.00000000D+00 -3.00000000D+00 -3.00000000D+00
 ( 1) -3.00000000D+00  3.00000000D+00 -3.00000000D+00  3.00000000D+00
 ( 2) -3.00000000D+00 -3.00000000D+00  3.00000000D+00  3.00000000D+00
 ( 1)  3.00000000D+00 -3.00000000D+00  3.00000000D+00  3.00000000D+00
 ( 2) -3.00000000D+00 -3.00000000D+00 -3.00000000D+00 -3.00000000D+00
 ( 1) -3.00000000D+00 -3.00000000D+00  3.00000000D+00 -3.00000000D+00
 ( 2) -3.00000000D+00  3.00000000D+00 -3.00000000D+00 -3.00000000D+00
 ( 1) -3.00000000D+00
 ( 2) -3.00000000D+00
 FILENAME : __
 OUTPUT-FILENAME : __
```

```
*** ( 2, 200) TIME SERIES OUTPUT    ***
ALL DATA ?   N
STARTING POINT ? (0- 200) :   0
HOW MANY POINTS ? (1- 201) :   50
=== INITIAL DATA ===
--- ( 2, 1) MATRIX X(0)      ---
( 1) -6.00000000D+00
( 2)  1.00000000D+01
FILENAME : _
=== INTERVAL (    1-   49) ===
--- ( 2,49) MATRIX DATA      ---
( 1) -5.96108836D+00 -5.92236782D+00 -5.88383740D+00 -5.84549620D+00
( 2)  9.95910532D+00  9.91242958D+00  9.87197165D+00  9.82573048D+00
( 1) -5.80737018D+00 -5.76945523D+00 -5.73172337D+00 -5.69418867D+00
( 2)  9.77372593D+00  9.72795981D+00  9.68840998D+00  9.63708437D+00
( 1) -5.65686506D+00 -5.61973652D+00 -5.58278710D+00 -5.54600397D+00
( 2)  9.59199081D+00  9.54711919D+00  9.50845941D+00  9.46999845D+00
( 1) -5.50940128D+00 -5.47299008D+00 -5.43676642D+00 -5.40071738D+00
( 2)  9.42574429D+00  9.38170789D+00  9.33789113D+00  9.30028094D+00
( 1) -5.36484511D+00 -5.32916069D+00 -5.29366317D+00 -5.25836658D+00
( 2)  9.25687333D+00  9.21367923D+00  9.17069759D+00  9.12193634D+00
( 1) -5.22326686D+00 -5.18836306D+00 -5.15364216D+00 -5.11908833D+00
( 2)  9.07939737D+00  9.03107959D+00  8.99496990D+00  8.95305831D+00
***
( 1) -5.08472766D+00 -5.05056213D+00 -5.01657880D+00 -4.98276176D+00
( 2)  8.90536479D+00  8.86388525D+00  8.82260666D+00  8.78751903D+00
( 1) -4.94909527D+00 -4.91560552D+00 -4.88230356D+00 -4.84917348D+00
( 2)  8.75261246D+00  8.70590699D+00  8.66541357D+00  8.62512217D+00
( 1) -4.81620245D+00 -4.78338969D+00 -4.75074938D+00 -4.71828065D+00
( 2)  8.59101982D+00  8.55110557D+00  8.51138747D+00  8.47186453D+00
( 1) -4.68597963D+00 -4.65386045D+00 -4.62192216D+00 -4.59015189D+00
( 2)  8.43253878D+00  8.38741822D+00  8.34850181D+00  8.30977659D+00
( 1) -4.55853681D+00 -4.52707319D+00 -4.49578719D+00 -4.46467789D+00
( 2)  8.27722962D+00  8.23886302D+00  8.19469684D+00  8.15673007D+00
( 1) -4.43372943D+00 -4.40295599D+00 -4.37235962D+00 -4.34192449D+00
( 2)  8.11895274D+00  8.07537291D+00  8.03798658D+00  8.00078383D+00
( 1) -4.31166176D+00
( 2)  7.95777573D+00
FILENAME : _
OUTPUT-FILENAME : _
/DPACS/ TIME-17:18:56 CPU-00:00:13
```

システムデータの複写	DP. I. 8
COPY ([／@]) ␣SN␣OSN　　@＝S, T, M, D, I	

機　能

指定したシステムを新システムに複写する。

◆第1スイッチの／@は，システムデータの形式を示し，@＝S，T，M，D，Iの5種類である。第1スイッチを付加しなかった場合には以下のメッセージが出力される。ここで，システムデータの形式を指定することができる。

SPECIFY TYPE OF SYSTEM DATA :

実 行 例

〔1〕 状態空間表現形式システムデータ複写例

```
 /DPACS/ TIME-17:14:06 CPU-00:00:02
COPY/S EX1 EX1CP
 TO COPY SYSTEMDATA
 GIVE COMMENT FOR SYSTEMDATA :
COPY EX1
 /DPACS/ 19:35:57
TYPE/S EX1CP
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1CP.SO     ( 2, 2, 3)

 --- ( 3, 3) MATRIX   A        ---
 ( 1)  0.0               1.00000000D+00  0.0
 ( 2)  1.00000000D+00 -1.00000000D+00  2.00000000D+00
 ( 3)  3.00000000D+00  0.0             1.00000000D+00
 FILENAME : _
 --- ( 3, 2) MATRIX   B        ---
 ( 1)  0.0             0.0
 ( 2)  1.00000000D+00 -2.00000000D+00
 ( 3)  0.0             1.00000000D+00
 FILENAME : _
 --- ( 2, 3) MATRIX   C        ---
 ( 1)  1.00000000D+00  0.0             0.0
 ( 2)  0.0             1.00000000D+00  1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX   D        ---
 ( 1)  0.0             0.0
 ( 2)  0.0             0.0
 FILENAME : _
 /DPACS/ TIME-17:19:03 CPU-00:00:13
```

〔2〕 伝達関数行列表現形式システムデータ複写例

```
/DPACS/ TIME-17:14:07 CPU-00:00:02
COPY/T EX10 EX10CP
 TO COPY SYSTEMDATA
 GIVE COMMENT FOR SYSTEMDATA :
COPY EX10
 /DPACS/ 19:40:57
TYPE/T EX10CP
 TO TYPE OUT SYSTEMDATA
 NORMALIZED FORM ?  N
 SYSTEMDATA EX10CP.TO    ( 2, 2, 2)

 1) DENOMINATER PART :
 Q(S)=Q0*S**N+Q1*S**(N-1)+...+Q(N-1)*S+Q(N)
   Q( 0) =   1.00000000D+00
   Q( 1) =   6.00000000D+00
   Q( 2) =   5.00000000D+00
 2) NUMERATER PART :
 P(S)=P0*S**N+P1*S**(N-1)+...+P(N-1)*S+P(N)
 --- ( 2, 2) MATRIX  P0         ---
 ( 1)  0.0              0.0
 ( 2)  0.0              0.0
 FILENAME : _
 --- ( 2, 2) MATRIX  P1         ---
 ( 1)  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  2.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  P2         ---
 ( 1)  4.00000000D+00  1.00000000D+00
 ( 2)  3.00000000D+00  2.00000000D+00
 FILENAME : _
 3) DIRECT PART
 --- ( 2, 2) MATRIX  D          ---
 ( 1)  0.0              0.0
 ( 2)  0.0              0.0
 FILENAME : _
 /DPACS/ TIME-17:19:05 CPU-00:00:13
```

〔3〕 マルコフパラメータ表現形式システムデータ複写例

```
/DPACS/ TIME-17:14:09 CPU-00:00:02
COPY/M SYS3 SYS3CP
 TO COPY SYSTEMDATA
 GIVE COMMENT FOR SYSTEMDATA :
COPY SY3
 /DPACS/ 19:44:59
TYPE/M SYS3CP
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA SYS3CP.MO    ( 2, 2, 3)

 --- ( 2, 2) MATRIX  D          ---
 ( 1)  1.00000000D+00  1.00000000D+00
 ( 2)  0.0            -1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  M1         ---
 ( 1)  0.0             1.00000000D+00
 ( 2)  1.00000000D+00 -2.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  M2         ---
 ( 1)  0.0            -1.00000000D+00
 ( 2)  2.00000000D+00  1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  M3         ---
 ( 1)  1.00000000D+00  2.00000000D+00
 ( 2)  0.0             1.00000000D+00
 FILENAME : _
 /DPACS/ TIME-17:19:09 CPU-00:00:13
```

〔4〕 微分（差分）方程式表現形式システムデータ複写例

```
/DPACS/ TIME-17:14:38 CPU-00:00:02
COPY/D EX8 EX8CP
 TO COPY SYSTEMDATA
 GIVE COMMENT FOR SYSTEMDATA :
COPY EX8
 /DPACS/ 19:48:37
TYPE/D EX8CP
 TO TYPE OUT SYSTEMDATA
 NORMALIZED FORM ?   N
 SYSTEMDATA EX8CP.D0     ( 2, 2, 3)

 1) DENOMINATER PART :
 T(S)=T0*S**N+T1*S**(N-1)+...+T(N-1)*S+T(N)
 --- ( 2, 2) MATRIX  T0         ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  0.0             1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  T1         ---
 ( 1)  0.0             0.0
 ( 2)  0.0             0.0
 FILENAME : _
 --- ( 2, 2) MATRIX  T2         ---
 ( 1) -2.00000000D+00  0.0
 ( 2)  0.0            -2.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  T3         ---
 ( 1) -5.00000000D+00  0.0
 ( 2)  0.0            -5.00000000D+00
 FILENAME : _
 2) NUMERATER PART :
 U(S)=U0*S**N+U1*S**(N-1)+...+U(N-1)*S+U(N)
 --- ( 2, 2) MATRIX  U0         ---
 ( 1)  0.0             0.0
 ( 2)  0.0             0.0
 FILENAME : _
 --- ( 2, 2) MATRIX  U1         ---
 ( 1)  0.0             0.0
 ( 2)  1.00000000D+00 -1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  U2         ---
 ( 1)  1.00000000D+00 -2.00000000D+00
 ( 2) -1.00000000D+00  5.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  U3         ---
 ( 1) -1.00000000D+00  4.00000000D+00
 ( 2)  3.00000000D+00 -7.00000000D+00
 FILENAME : _
 3) DIRECT PART
 --- ( 2, 2) MATRIX  D          ---
 ( 1)  0.0             0.0
 ( 2)  0.0             0.0
 FILENAME : _
 /DPACS/ TIME-17:19:12 CPU-00:00:13
```

〔5〕 入出力データ表現形式システムデータ複写例

```
/DPACS/ TIME-17:14:40 CPU-00:00:02
COPY/I SYS11 SYS11CP
 TO COPY SYSTEMDATA
 GIVE COMMENT FOR SYSTEMDATA :
COPY SYS11
 /DPACS/ 19:52:57
TYPE/I SYS11CP
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA SYS11CP.IO  ( 1, 1,   2)

 *** ( 1,   2) TIME SERIES INPUT    ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 1, 1) MATRIX X(0)      ---
 ( 1)  0.0
 FILENAME : _
 === INTERVAL (    1-   2) ===
 --- ( 1, 2) MATRIX DATA      ---
 ( 1)  1.000000000D+00  1.000000000D+00
 FILENAME : _
 OUTPUT-FILENAME : _
 *** ( 1,   2) TIME SERIES OUTPUT   ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 1, 1) MATRIX X(0)      ---
 ( 1)  0.0
 FILENAME : _
 === INTERVAL (    1-   2) ===
 --- ( 1, 2) MATRIX DATA      ---
 ( 1)  2.000000000D+00  2.000000000D+00
 FILENAME : _
 OUTPUT-FILENAME : _
 /DPACS/ TIME-17:19:14 CPU-00:00:13
```

空間の座標変換および入出力変数の選出	DP. I. 9
PICK([／@])[#／C]␣SN␣OSN　　@＝S, T, M	

機　能

入出力変数から必要なものだけを取り出す変換を行う。

◆第1スイッチの／@は，システムデータの形式を示し，@＝S，T，Mの3種類である。第1スイッチを付加しなかった場合には以下のメッセージが出力される。ここで，システムデータの形式を指定することができる。

SPECIFY TYPE OF SYSTEM DATA :

◆第2スイッチ／Cを付加した場合は，座標変換行列を入力し，入出力の空間座標変換を行う。

理論概要

◆第1スイッチが／Sの場合

$$\begin{cases} \sigma \boldsymbol{x} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} \\ \boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}\boldsymbol{u} \end{cases}$$ に対し

$\boldsymbol{u} = \boldsymbol{G}\overline{\boldsymbol{u}}$, $\overline{\boldsymbol{y}} = \boldsymbol{H}\boldsymbol{y}$, $\boldsymbol{G} \in \mathbf{R}^{m\times\overline{m}}$, $\boldsymbol{H} \in \mathbf{R}^{\overline{p}\times p}$ なる変換

$$\begin{cases} \sigma \boldsymbol{x} = \overline{\boldsymbol{A}}\boldsymbol{x} + \overline{\boldsymbol{B}}\overline{\boldsymbol{u}} \\ \overline{\boldsymbol{y}} = \overline{\boldsymbol{C}}\boldsymbol{x} + \overline{\boldsymbol{D}}\overline{\boldsymbol{u}} \end{cases}$$

$\overline{\boldsymbol{A}} = \boldsymbol{A}$, $\overline{\boldsymbol{B}} = \boldsymbol{B}\cdot\boldsymbol{G}$, $\overline{\boldsymbol{C}} = \boldsymbol{H}\cdot\boldsymbol{C}$, $\overline{\boldsymbol{D}} = \boldsymbol{H}\cdot\boldsymbol{D}\cdot\boldsymbol{G}$

◆第1スイッチが／Tの場合

$$\begin{cases} \boldsymbol{y} = \boldsymbol{G}(\sigma)\mathbf{u} \\ \boldsymbol{G}(\sigma) = \dfrac{\boldsymbol{P}_0\sigma^{n} + \boldsymbol{P}_1\sigma^{n-1} + \cdots\cdots + \boldsymbol{P}_n}{q_0\sigma^{n} + q_1\sigma^{n-1} + \cdots\cdots + q_n} + \boldsymbol{D} \end{cases}$$

$\boldsymbol{u} = \boldsymbol{G}\overline{\boldsymbol{u}}$, $\overline{\boldsymbol{y}} = \boldsymbol{H}\boldsymbol{y}$, $\boldsymbol{G} \in \mathbf{R}^{m\times\overline{m}}$, $\boldsymbol{H} \in \mathbf{R}^{\overline{p}\times p}$ なる変換に対し

$$\begin{cases} \overline{\boldsymbol{y}} = \overline{\boldsymbol{G}}(\sigma)\overline{\boldsymbol{u}} \\ \overline{\boldsymbol{G}}(\sigma) = \dfrac{\overline{\boldsymbol{P}}_0\sigma^{n} + \overline{\boldsymbol{P}}_1\sigma^{n-1} + \cdots\cdots + \overline{\boldsymbol{P}}_n}{\overline{q}_0\sigma^{n} + \overline{q}_1\sigma^{n-i} + \cdots\cdots + \overline{q}_n} + \overline{\boldsymbol{D}} \end{cases}$$

$\overline{q}_i = q_i$, $\overline{\boldsymbol{P}}_i = \boldsymbol{H}\cdot\boldsymbol{P}_i\cdot\boldsymbol{G}$, $\overline{\boldsymbol{D}} = \boldsymbol{H}\cdot\boldsymbol{D}\cdot\boldsymbol{G}$

◆第1スイッチが／Mの場合

$$\begin{cases} \boldsymbol{y} = \boldsymbol{G}(\sigma)\boldsymbol{u} \\ \boldsymbol{G}(\sigma) = \boldsymbol{D} + \boldsymbol{M}_1\sigma^{-1} + \boldsymbol{M}_2\sigma^{-2} + \cdots\cdots\cdots + \boldsymbol{M}_n\sigma^{-n} \end{cases}$$

$\boldsymbol{u} = \boldsymbol{G}\overline{\boldsymbol{u}}$, $\overline{\boldsymbol{y}} = \boldsymbol{H}\boldsymbol{y}$, $\boldsymbol{G} \in \mathbf{R}^{m\times\overline{m}}$, $\boldsymbol{H} \in \mathbf{R}^{\overline{p}\times p}$ なる変換に対し

$$\begin{cases} \overline{\boldsymbol{y}} = \overline{\boldsymbol{G}}(\sigma)\cdot\overline{\boldsymbol{u}} \\ \overline{\boldsymbol{G}}(\sigma) = \overline{\boldsymbol{D}} + \overline{\boldsymbol{M}}_1\sigma^{-1} + \cdots\cdots + \overline{\boldsymbol{M}}_n\sigma^{-n} \end{cases}$$

$\overline{\boldsymbol{D}} = \boldsymbol{H}\cdot\boldsymbol{D}\cdot\boldsymbol{G}$, $\overline{\boldsymbol{M}}_i = \boldsymbol{H}\cdot\boldsymbol{M}_i\cdot\boldsymbol{G}$

実 行 例

〔1〕 状態空間表現形式システムの入出力変数から必要なものだけを取り出す例

```
 /DPACS/ TIME-17:14:42 CPU-00:00:02
PICK/S EX1 EX1PK
 TO TRANSFORM THE COORDINATE OF INPUT AND OUTPUT SPACE
 NO. OF NEW INPUTS AND OUTPUTS ? (1- 2),(1- 2) :  1,1
 SPECIFY  1-INPUTS TO BE PICKED UP :   1
 SPECIFY  1-OUTPUTS TO BE PICKED UP :   2
 GIVE COMMENT FOR SYSTEMDATA :
EX1 PICK/S
 /DPACS/ 19:59:58
TYPE/S EX1PK
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1PK.SO     ( 1, 1, 3)

 --- ( 3, 3) MATRIX   A        ---
 ( 1)  0.0              1.00000000D+00  0.0
 ( 2)  1.00000000D+00 -1.00000000D+00  2.00000000D+00
 ( 3)  3.00000000D+00  0.0             1.00000000D+00
 FILENAME : _
 --- ( 3, 1) MATRIX   B        ---
 ( 1)  0.0
 ( 2)  1.00000000D+00
 ( 3)  0.0
 FILENAME : _
 --- ( 1, 3) MATRIX   C        ---
 ( 1)  0.0              1.00000000D+00  1.00000000D+00
 FILENAME : _
 --- ( 1, 1) MATRIX   D        ---
 ( 1)  0.0
 FILENAME : _
 /DPACS/ TIME-17:19:17 CPU-00:00:13
```

〔2〕 状態空間表現形式システムの入出力空間の座標変換の例

```
/DPACS/ TIME-17:14:43 CPU-00:00:02
PICK/S/C EX1 EX1PKC
 TO TRANSFORM THE COORDINATE OF INPUT AND OUTPUT SPACE
 SPECIFY NO. OF NEW INPUTS AND OUTPUTS :   1,1
 INPUT MATRIX G AND H, WHERE U=G*U¬ AND Y¬=H*Y
 INPUT -MODE OF ( 2, 1) MATRIX   G       : _
 ( 1) ROW
5
 ( 2) ROW
3
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 1, 2) MATRIX   H       : _
 ( 1) ROW
5,3
 TYPE OR REVISE ? _
 GIVE COMMENT FOR SYSTEMDATA :
PICK/S/C EX1
 /DPACS/ 20:12: 4
 /DPACS/ 20:12: 9
TYPE/S EX1PKC
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1PKC.SO    ( 1, 1, 3)

 --- ( 3, 3) MATRIX   A        ---
 ( 1)  0.0               1.00000000D+00  0.0
 ( 2)  1.00000000D+00 -1.00000000D+00  2.00000000D+00
 ( 3)  3.00000000D+00  0.0             1.00000000D+00
 FILENAME : _
 --- ( 3, 1) MATRIX   B        ---
 ( 1)  0.0
 ( 2) -1.00000000D+00
 ( 3)  3.00000000D+00
 FILENAME : _
 --- ( 1, 3) MATRIX   C        ---
 ( 1)  5.00000000D+00  3.00000000D+00  3.00000000D+00
 FILENAME : _
 --- ( 1, 1) MATRIX   D        ---
 ( 1)  0.0
 FILENAME : _
 /DPACS/ TIME-17:19:18 CPU-00:00:13
```

〔3〕 伝達関数行列表現形式システムの入出力変数から必要なものだけを取り出す例

```
 /DPACS/ TIME-17:14:44 CPU-00:00:02
PICK/T EX10 EX10PKT
 TO TRANSFORM THE COORDINATE OF INPUT AND OUTPUT SPACE
 NO. OF NEW INPUTS AND OUTPUTS ? (1- 2),(1- 2) :  1,1
 SPECIFY  1-INPUTS TO BE PICKED UP :   2
 SPECIFY  1-OUTPUTS TO BE PICKED UP :   1
 GIVE COMMENT FOR SYSTEMDATA :
PICK/T EX10
 /DPACS/ 20:21: 3
TYPE/T EX10PKT
 TO TYPE OUT SYSTEMDATA
 NORMALIZED FORM ? _
 SYSTEMDATA EX10PKT.TO  ( 1, 1, 2)

 1) DENOMINATER PART :
 Q(S)=Q0*S**N+Q1*S**(N-1)+...+Q(N-1)*S+Q(N)
   Q( 0) =    1.00000000D+00
   Q( 1) =    6.00000000D+00
   Q( 2) =    5.00000000D+00
 2) NUMERATER PART :
 P(S)=P0*S**N+P1*S**(N-1)+...+P(N-1)*S+P(N)
 --- ( 1, 1) MATRIX  P0        ---
 ( 1)  0.0
 FILENAME : _
 --- ( 1, 1) MATRIX  P1        ---
 ( 1)  1.00000000D+00
 FILENAME : _
 --- ( 1, 1) MATRIX  P2        ---
 ( 1)  1.00000000D+00
 FILENAME : _
 3) DIRECT PART
 --- ( 1, 1) MATRIX  D         ---
 ( 1)  0.0
 FILENAME : _
 /DPACS/ TIME-17:19:21 CPU-00:00:13
```

〔4〕 伝達関数行列表現形式システムの入出力空間の座標変換の例

```
 /DPACS/ TIME-17:14:45 CPU-00:00:02
PICK/T/C EX10 EX10PKC
 TO TRANSFORM THE COORDINATE OF INPUT AND OUTPUT SPACE
 SPECIFY NO. OF NEW INPUTS AND OUTPUTS :   1,1
 INPUT MATRIX G AND H, WHERE U=G*U¬ AND Y¬=H*Y
 INPUT -MODE OF ( 2, 1) MATRIX   G       : _
 ( 1) ROW
3
 ( 2) ROW
3
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 1, 2) MATRIX   H       : _
 ( 1) ROW
5,5
 TYPE OR REVISE ? _
 GIVE COMMENT FOR SYSTEMDATA :
PICK/T/C EX10
 /DPACS/ 20:32:33
TYPE/T EX10PKC
 TO TYPE OUT SYSTEMDATA
 NORMALIZED FORM ?  N
 SYSTEMDATA EX10PKC.TO  ( 1, 1, 2)

 1) DENOMINATER PART :
 Q(S)=Q0*S**N+Q1*S**(N-1)+...+Q(N-1)*S+Q(N)
   Q( 0) =    1.00000000D+00
   Q( 1) =    6.00000000D+00
   Q( 2) =    5.00000000D+00
 2) NUMERATER PART :
 P(S)=P0*S**N+P1*S**(N-1)+...+P(N-1)*S+P(N)
 --- ( 1, 1) MATRIX  P0       ---
 ( 1)  0.0
 FILENAME : _
 --- ( 1, 1) MATRIX  P1       ---
 ( 1)  7.50000000D+01
 FILENAME : _
 --- ( 1, 1) MATRIX  P2       ---
 ( 1)  1.50000000D+02
 FILENAME : _
 3) DIRECT PART
 --- ( 1, 1) MATRIX  D        ---
 ( 1)  0.0
 FILENAME : _
 /DPACS/ TIME-17:19:23 CPU-00:00:13
```

〔5〕 マルコフパラメータ表現形式システムの入出力変数から必要なものだけを取り出す例

```
/DPACS/ TIME-17:14:47 CPU-00:00:02
PICK/M SYS3 SYS3PK
 TO TRANSFORM THE COORDINATE OF INPUT AND OUTPUT SPACE
 NO. OF NEW INPUTS AND OUTPUTS ? (1- 2),(1- 2) :  1,1
 SPECIFY  1-INPUTS TO BE PICKED UP :   2
 SPECIFY  1-OUTPUTS TO BE PICKED UP :   1
 GIVE COMMENT FOR SYSTEMDATA :
PICK/K SYS3
 /DPACS/ 20:38:26
TYPE/M SYS3PK
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA SYS3PK.MO    ( 1, 1, 3)

 --- ( 1, 1) MATRIX  D        ---
 ( 1)  1.00000000D+00
 FILENAME : _
 --- ( 1, 1) MATRIX  M1       ---
 ( 1)  1.00000000D+00
 FILENAME : _
 --- ( 1, 1) MATRIX  M2       ---
 ( 1) -1.00000000D+00
 FILENAME : _
 --- ( 1, 1) MATRIX  M3       ---
 ( 1)  2.00000000D+00
 FILENAME : _
 /DPACS/ TIME-17:19:24 CPU-00:00:13
```

〔6〕 マルコフパラメータ表現形式システムの入出力空間の座標変換の例

```
 /DPACS/ TIME-17:15:15 CPU-00:00:02
PICK/M/C SYS3 SYS3PKC
 TO TRANSFORM THE COORDINATE OF INPUT AND OUTPUT SPACE
 SPECIFY NO. OF NEW INPUTS AND OUTPUTS :   1,1
 INPUT MATRIX G AND H, WHERE U=G*U¬ AND Y¬=H*Y
 INPUT -MODE OF ( 2, 1) MATRIX   G       : _
 ( 1) ROW
3
 ( 2) ROW
3
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 1, 2) MATRIX   H       : _
 ( 1) ROW
5,5
 TYPE OR REVISE ?_
 GIVE COMMENT FOR SYSTEMDATA :
PICK/M/C SYS3
 /DPACS/ 20:43:53
TYPE/M SYS3PKC
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA SYS3PKC.MO  ( 1, 1, 3)

 --- ( 1, 1) MATRIX  D       ---
 ( 1)  1.500000000D+01
 FILENAME : _
 --- ( 1, 1) MATRIX  M1      ---
 ( 1)  0.0
 FILENAME : _
 --- ( 1, 1) MATRIX  M2      ---
 ( 1)  3.000000000D+01
 FILENAME : _
 --- ( 1, 1) MATRIX  M3      ---
 ( 1)  6.000000000D+01
 FILENAME : _
 /DPACS/ TIME-17:19:26 CPU-00:00:13
```

双対システムの作成	DP. I. 10
DUAL ([／@]) ␣SN␣OSN	@＝S, T, M

機　能

指定したシステムの双対システムを作成する。

◆第1スイッチ／@は，システムデータの形式を示し，S，T，Mの3種類である。第1スイッチを付加しなかった場合には，以下のメッセージが出力される。ここで，システムデータの形式を指定することができる。

SPECIFY TYPE OF SYSTEM DATA :

理論概要

◆第1スイッチが／Sの場合

$$\begin{cases} \sigma\, \boldsymbol{x} = \boldsymbol{A}\,\boldsymbol{x} + \boldsymbol{B}\,\boldsymbol{u} \\ \boldsymbol{y} = \boldsymbol{C}\,\boldsymbol{x} + \boldsymbol{D}\,\boldsymbol{u} \end{cases}$$

↓ 変換

$$\begin{cases} \sigma\, \overline{\boldsymbol{x}} = {}^{t}\boldsymbol{A}\,\overline{\boldsymbol{x}} + {}^{t}\boldsymbol{C}\,\overline{\boldsymbol{u}} \\ \overline{\boldsymbol{y}} = {}^{t}\boldsymbol{B}\,\overline{\boldsymbol{x}} + {}^{t}\boldsymbol{D}\,\overline{\boldsymbol{u}} \end{cases}$$

（注）本来，制御理論における双対システム[7]は，次式。

$$\begin{cases} \sigma\, {}^{t}\boldsymbol{x} = -{}^{t}\boldsymbol{A}\,{}^{t}\boldsymbol{x} - {}^{t}\boldsymbol{C}\,{}^{t}\boldsymbol{u} \\ {}^{t}\boldsymbol{y} = {}^{t}\boldsymbol{B}\,{}^{t}\boldsymbol{x} + {}^{t}\boldsymbol{D}\,{}^{t}\boldsymbol{u} \end{cases}$$

◆第1スイッチが／Tの場合

$$\begin{cases} \boldsymbol{y} = \boldsymbol{G}(\sigma)\boldsymbol{u} \\ \boldsymbol{G}(\sigma) = \dfrac{\boldsymbol{P}_0\sigma^{n} + \boldsymbol{P}_1\sigma^{n-1} + \cdots\cdots + \boldsymbol{P}_n}{q_0\sigma^{n} + q_1\sigma^{n-1} + \cdots\cdots + q_n} + \boldsymbol{D} \end{cases}$$

↓ 変換

$$\begin{cases} \overline{\boldsymbol{y}} = \overline{\boldsymbol{G}}(\sigma)\overline{\boldsymbol{u}} \\ \overline{\boldsymbol{G}}(\sigma) = \dfrac{{}^{t}\boldsymbol{P}_0\sigma^{n} + {}^{t}\boldsymbol{P}_1\sigma^{n-1} + \cdots\cdots + {}^{t}\boldsymbol{P}_n}{q_0\sigma^{n} + q_1\sigma^{n-1} + \cdots\cdots + q_n} + {}^{t}\boldsymbol{D} \end{cases}$$

（注）本来，制御理論における双対システムは上式で

$${}^{t}\boldsymbol{P}_i \rightarrow (-1)^{n-i}\,{}^{t}\boldsymbol{P}_i\ ,\quad q_i \rightarrow (-1)^{n-i}\, q_i$$

としたもの。

◆第1スイッチが／Mの場合

$$\begin{cases} \boldsymbol{y} = \boldsymbol{G}(\sigma)\boldsymbol{u} \\ \boldsymbol{G}(\sigma) = \boldsymbol{D} + \boldsymbol{M}_1\sigma^{-1} + \cdots\cdots + \boldsymbol{M}_n\sigma^{-n} \end{cases}$$

↓ 変換

$$\begin{cases} \overline{\boldsymbol{y}} = \overline{\boldsymbol{G}}(\sigma)\overline{\boldsymbol{u}} \\ \overline{\boldsymbol{G}}(\sigma) = {}^{t}\boldsymbol{D} + {}^{t}\boldsymbol{M}_1\sigma^{-1} + \cdots\cdots + {}^{t}\boldsymbol{M}_n\sigma^{-n} \end{cases}$$

（注）本来，制御理論における双対システムは，上式で

$${}^{t}\boldsymbol{M}_i \rightarrow (-1)^{n-i}\,{}^{t}\boldsymbol{M}_i$$

としたもの。

実行例

〔1〕 状態空間表現形式システムの双対システムの作成例

```
/DPACS/ TIME-17:14:48 CPU-00:00:02
DUAL/S EX1 EX1DUAL
 TO OBTAIN DUAL SYSTEM
 GIVE COMMENT FOR SYSTEMDATA :
DUAL/S EX1
 /DPACS/ 21: 3: 3
TYPE/S EX1DUAL
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1DUAL.SO  ( 2, 2, 3)

 --- ( 3, 3) MATRIX   A       ---
 ( 1)  0.0              1.00000000D+00  3.00000000D+00
 ( 2)  1.00000000D+00 -1.00000000D+00  0.0
 ( 3)  0.0              2.00000000D+00  1.00000000D+00
 FILENAME : __
 --- ( 3, 2) MATRIX   B       ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  0.0              1.00000000D+00
 ( 3)  0.0              1.00000000D+00
 FILENAME : __
 --- ( 2, 3) MATRIX   C       ---
 ( 1)  0.0              1.00000000D+00  0.0
 ( 2)  0.0             -2.00000000D+00  1.00000000D+00
 FILENAME : __
 --- ( 2, 2) MATRIX   D       ---
 ( 1)  0.0              0.0
 ( 2)  0.0              0.0
 FILENAME : __
 /DPACS/ TIME-17:19:28 CPU-00:00:13
```

〔2〕 伝達関数行列表現形式システムの双対システムの作成例

```
/DPACS/ TIME-17:14:49 CPU-00:00:02
DUAL/T EX10 EX10DUAL
 TO OBTAIN DUAL SYSTEM
 GIVE COMMENT FOR SYSTEMDATA :
EX10 DUAL
 /DPACS/ 21: 6:36
TYPE/T EX10DUAL
 TO TYPE OUT SYSTEMDATA
 NORMALIZED FORM ?   N
 SYSTEMDATA EX10DUAL.TO ( 2, 2, 2)

 1) DENOMINATER PART :
 Q(S)=Q0*S**N+Q1*S**(N-1)+...+Q(N-1)*S+Q(N)
   Q( 0) =    1.00000000D+00
   Q( 1) =    6.00000000D+00
   Q( 2) =    5.00000000D+00
 2) NUMERATER PART :
 P(S)=P0*S**N+P1*S**(N-1)+...+P(N-1)*S+P(N)
 --- ( 2, 2) MATRIX  P0          ---
 ( 1)  0.0               0.0
 ( 2)  0.0               0.0
 FILENAME : _
 --- ( 2, 2) MATRIX  P1          ---
 ( 1)  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  2.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  P2          ---
 ( 1)  4.00000000D+00  3.00000000D+00
 ( 2)  1.00000000D+00  2.00000000D+00
 FILENAME : _
 3) DIRECT PART
 --- ( 2, 2) MATRIX  D           ---
 ( 1)  0.0               0.0
 ( 2)  0.0               0.0
 FILENAME :
 /DPACS/ TIME-17:19:30 CPU-00:00:13
```

〔3〕 マルコフパラメータ表現形式システムの双対システム作成例

```
/DPACS/ TIME-17:14:50 CPU-00:00:02
DUAL/M SYS3 SYS3DUAL
 TO OBTAIN DUAL SYSTEM
 GIVE COMMENT FOR SYSTEMDATA :
DUAL SYS3
 /DPACS/ 21:10:32
TYPE/M SYS3DUAL
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA SYS3DUAL.MO ( 2, 2, 3)

 --- ( 2, 2) MATRIX  D        ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  1.00000000D+00 -1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  M1       ---
 ( 1)  0.0             1.00000000D+00
 ( 2)  1.00000000D+00 -2.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  M2       ---
 ( 1)  0.0             2.00000000D+00
 ( 2) -1.00000000D+00  1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  M3       ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  2.00000000D+00  1.00000000D+00
 FILENAME : _
 /DPACS/ TIME-17:19:32 CPU-00:00:13
```

状態空間表現形式システムの正則変換	DP. II. 1
COOD ($\begin{bmatrix} /1 \\ /2 \end{bmatrix}$) _SN	

機　　能

状態空間表現形式システム Σ_s に対して正則変換を施し，変換後の状態空間表現形式システム $\overline{\Sigma}_s$ を得る。正則行列 $\boldsymbol{T}$ を本コマンド発行後に入力する。

◆第1スイッチが／1の場合は，$\boldsymbol{x}=\boldsymbol{T}\overline{\boldsymbol{x}}$ なる変換を行い，第1スイッチが／2の場合は，$\boldsymbol{x}=\boldsymbol{T}^{-1}\overline{\boldsymbol{x}}$ なる変換を行う。

◆第1スイッチを省略した場合には，以下のメッセージが出力される。

WHICH TRANSFORMATION ? (1) X＝T＊X￢ (2) X￢＝T＊X

ここで，変換の形式を選択できる。

理論概要

$$\Sigma_s \left\{ \begin{array}{l} \sigma \boldsymbol{x}(t)=\boldsymbol{A}\boldsymbol{x}(t)+\boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{y}(t)=\boldsymbol{C}\boldsymbol{x}(t)+\boldsymbol{D}\boldsymbol{u}(t) \end{array} \right.$$

$$\downarrow$$

$$\overline{\Sigma}_s \left\{ \begin{array}{l} \sigma \overline{\boldsymbol{x}}(t)=\overline{\boldsymbol{A}}\overline{\boldsymbol{x}}(t)+\overline{\boldsymbol{B}}\boldsymbol{u}(t) \\ \boldsymbol{y}(t)=\overline{\boldsymbol{C}}\overline{\boldsymbol{x}}(t)+\boldsymbol{D}\boldsymbol{u}(t) \end{array} \right.$$

の変換を行うとすると

$$(\overline{\boldsymbol{A}},\ \overline{\boldsymbol{B}},\ \overline{\boldsymbol{C}}) = \left\{ \begin{array}{l} \text{第1スイッチが／1の時}\ (\boldsymbol{T}^{-1}\boldsymbol{A}\boldsymbol{T},\ \boldsymbol{T}^{-1}\boldsymbol{B},\ \boldsymbol{C}\boldsymbol{T}) \\ \text{第1スイッチが／2の時}\ (\boldsymbol{T}\boldsymbol{A}\boldsymbol{T}^{-1},\ \boldsymbol{T}\boldsymbol{B},\ \boldsymbol{C}\boldsymbol{T}^{-1}) \end{array} \right.$$

実 行 例

〔1〕 状態空間表現形式システムの正則変換（$\boldsymbol{x} = \boldsymbol{T}\bar{\boldsymbol{x}}$）の例

```
 /DPACS/ TIME-17:14:51 CPU-00:00:02
COOD EX1
 TO TRANSFORM COORDINATE OF STATE SPACE
 WHITCH TRANSFORMATION ?  (1) X=T*X¬  (2) X¬=T*X :   1
 INPUT -MODE OF ( 3, 3) MATRIX   T        : _
 ( 1) ROW
1,-2,0
 ( 2) ROW
-1,0,3
 ( 3) ROW
0,1,1
 TYPE OR REVISE ?   T
 --- ( 3, 3) MATRIX   T       ---
 ( 1)  1.00000000D+00 -2.00000000D+00  0.0
 ( 2) -1.00000000D+00  0.0             3.00000000D+00
 ( 3)  0.0             1.00000000D+00  1.00000000D+00
 TYPE OR REVISE ? _
 GIVE COMMENT FOR SYSTEMDATA : _
COOD EX1
 /DPACS/ 21:15:17
TYPE/S EX1
 SYSTEM EX1       HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0       (    3) ;SYSIN   /S /C /  /    (87/11/27) :EX1 TEST DATA
  2) EX1.S1       (    3) ;COOD    /1 /  /  /    (87/11/30) :COOD EX1
 WHICH SYSTEMDATA IS USED ? :   1
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1.S0       ( 2, 2, 3)

 --- ( 3, 3) MATRIX   A       ---
 ( 1)  0.0             1.00000000D+00  0.0
 ( 2)  1.00000000D+00 -1.00000000D+00  2.00000000D+00
 ( 3)  3.00000000D+00  0.0             1.00000000D+00
 FILENAME : _
 --- ( 3, 2) MATRIX   B       ---
 ( 1)  0.0             0.0
 ( 2)  1.00000000D+00 -2.00000000D+00
 ( 3)  0.0             1.00000000D+00
 FILENAME : _
 --- ( 2, 3) MATRIX   C       ---
 ( 1)  1.00000000D+00  0.0             0.0
 ( 2)  0.0             1.00000000D+00  1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX   D       ---
 ( 1)  0.0             0.0
 ( 2)  0.0             0.0
 FILENAME : _
 /DPACS/ TIME-17:19:39 CPU-00:00:13
```

〔2〕 状態空間表現形式システムの正則変換（$\boldsymbol{x} = \boldsymbol{T}\bar{\boldsymbol{x}}$）の例

```
 /DPACS/ TIME-17:14:53 CPU-00:00:02
COOD/1 EX1
 SYSTEM EX1         HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0         (    3) ;SYSIN    /S /C /  /    (87/11/27) :EX1 TEST DATA
  2) EX1.S1         (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD EX1
 WHICH SYSTEMDATA IS USED ? :   1
 TO TRANSFORM COORDINATE OF STATE SPACE
 INPUT -MODE OF ( 3, 3) MATRIX   T       : _
 ( 1) ROW
1,-2,0
 ( 2) ROW
-1,0,3
 ( 3) ROW
0,1,1
 TYPE OR REVISE ? _
 GIVE COMMENT FOR SYSTEMDATA :
COOD/1 EX1
 /DPACS/ 21:19:47
TYPE/S EX1
 SYSTEM EX1         HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0         (    3) ;SYSIN    /S /C /  /    (87/11/27) :EX1 TEST DATA
  2) EX1.S1         (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD EX1
  3) EX1.S2         (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD/1 EX1
 WHICH SYSTEMDATA IS USED ? :   3
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1.S2        ( 2, 2, 3)

 --- ( 3, 3) MATRIX   A      ---
 ( 1)  2.20000000D+00 -6.00000000D+00  3.40000000D+00
 ( 2)  1.60000000D+00 -3.00000000D+00  2.00000000D-01
 ( 3)  1.40000000D+00 -2.00000000D+00  8.00000000D-01
 FILENAME : _
 --- ( 3, 2) MATRIX   B      ---
 ( 1) -4.00000000D-01  2.00000000D+00
 ( 2) -2.00000000D-01  1.00000000D+00
 ( 3)  2.00000000D-01  5.55111512D-17
 FILENAME : _
 --- ( 2, 3) MATRIX   C      ---
 ( 1)  1.00000000D+00 -2.00000000D+00  0.0
 ( 2) -1.00000000D+00  1.00000000D+00  4.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX   D      ---
 ( 1)  0.0             0.0
 ( 2)  0.0             0.0
 FILENAME : _
 /DPACS/ TIME-17:19:41 CPU-00:00:13
```

〔3〕 状態空間表現形式システムの正則変換（$\boldsymbol{x} = \boldsymbol{T}^{-1}\overline{\boldsymbol{x}}$）の例

```
/DPACS/ TIME-17:15:09 CPU-00:00:02
COOD/2 EX1
 SYSTEM EX1        HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0        (    3) ;SYSIN    /S /C /  /     (87/11/27) :EX1 TEST DATA
  2) EX1.S1        (    3) ;COOD     /1 /  /  /     (87/11/30) :COOD EX1
  3) EX1.S2        (    3) ;COOD     /1 /  /  /     (87/11/30) :COOD/1 EX1
 WHICH SYSTEMDATA IS USED ? :  1
 TO TRANSFORM COORDINATE OF STATE SPACE
 INPUT -MODE OF ( 3, 3) MATRIX   T       : _
 ( 1) ROW
1,-2,0
 ( 2) ROW
-1,0,3
 ( 3) ROW
0,1,1
 TYPE OR REVISE ?  N
 GIVE COMMENT FOR SYSTEMDATA :
COOD/2 EX1
 /DPACS/ 21:24:11
TYPE/S EX1
 SYSTEM EX1        HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0        (    3) ;SYSIN    /S /C /  /     (87/11/27) :EX1 TEST DATA
  2) EX1.S1        (    3) ;COOD     /1 /  /  /     (87/11/30) :COOD EX1
  3) EX1.S2        (    3) ;COOD     /1 /  /  /     (87/11/30) :COOD/1 EX1
  4) EX1.S3        (    3) ;COOD     /2 /  /  /     (87/11/30) :COOD/2 EX1
 WHICH SYSTEMDATA IS USED ? :  4
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1.S3        ( 2, 2, 3)

 --- ( 3, 3) MATRIX   A        ---
 ( 1) -2.60000000D+00 -6.00000000D-01 -2.20000000D+00
 ( 2)  6.20000000D+00 -2.80000000D+00  1.14000000D+01
 ( 3)  3.20000000D+00 -8.00000000D-01  5.40000000D+00
 FILENAME : _
 --- ( 3, 2) MATRIX   B        ---
 ( 1) -2.00000000D+00  4.00000000D+00
 ( 2)  0.0             3.00000000D+00
 ( 3)  1.00000000D+00 -1.00000000D+00
 FILENAME : _
 --- ( 2, 3) MATRIX   C        ---
 ( 1)  6.00000000D-01 -4.00000000D-01  1.20000000D+00
 ( 2) -2.77555756D-17 -2.77555756D-17  1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX   D        ---
 ( 1)  0.0             0.0
 ( 2)  0.0             0.0
 FILENAME : _
 /DPACS/ TIME-17:19:42 CPU-00:00:13
```

外部記述の正準最小実現の取得	DP. II. 2
CMR $\left(\begin{bmatrix} /@\left(\begin{bmatrix}/C\\/O\end{bmatrix}\right)\left(\begin{bmatrix}/1\\/2\end{bmatrix}\right) \\ /D\left(\underline{/C}\right)\left(\begin{bmatrix}/1\\/2\end{bmatrix}\right)\end{bmatrix} \right)$ ␣SN　　@ = S, T, M	

機　能

(1) 第1スイッチ／@が／S，／T，／Mのとき，それぞれ状態空間表現形式システム(S)，伝達関数行列表現形式システム(T)，マルコフパラメータ表現形式システム(M)から，対応する最小実現（最小次数を持つ状態空間表現モデル）を，その可制御対（／C）または可観測対（／O）が Luenberger の第1正準形（／1）または第2正準形（／2）となるように得る。

(2) 第1スイッチが／Dのとき，微分（差分）方程式表現形式システム（D）から，対応する最小実現を，その可制御対が Luenberger の第1正準形（／1）または第2正準形（／2）となるように得る。尚，微分（差分）方程式表現形式システムの場合には，可観測対を正準形にするような最小実現を直接得ることができないので，第2スイッチの／O指定は無視される。

◆第1スイッチを付加しなかった場合には以下のメッセージが出力される。ここで，システムデータの形式を指定することができる。

SPECIFY TYPE OF SYSTEM DATA :

◆第1スイッチが／Sの場合で，第2および第3スイッチを省略した場合は，以下のメッセージが出力される。

WHICH SUBSPACE IS REMOVED,
(1) UNCONTROLLABLE ONE OR (2) UNOBSERVABLE ONE ? :
SPECIFY TYPE OF CANONICAL FORM (1 OR 2) :

◆第1スイッチが／Tまたは／Mの場合で，第2および第3スイッチを省略した場合には，以下のメッセージが出力される。

WHICH PAIR IN MINIMAL REALIZATION IS TRANSFORMED
INTO CANONICAL FORM, (1) (A, B) OR (2) (A, C) ?
SPECIFY TYPE OF CANONICAL FORM (1 OR 2) :

ここで，可制御対または可観測対の Luenberger の第1正準形または第2正準形の種別を指定できる。

◆第1スイッチが／Dの場合で，第3スイッチを省略した場合には，以下のメッセージが出力される。

SPECIFY TYPE OF CANONICAL FORM (1 OR 2) :

ここで，Luenberger 第1正準形または第2正準形の種別を指定できる。

理論概要 [8)]

$\Sigma_@$から最小実現 Σ_s^* を求めるアルブリズムは，以下のとおり，　　@＝S，T，M，D

Step　1：第2スイッチが／Oのとき，$\Sigma_@$の双対モデルを$\Sigma_@$とする。

Step　2：ＳＹＳのＶにより行列Ｖ@を得る。

（Ｖ；可制御性行列を求める。）

Step　3：ＳＹＳのＬＣＦ（ＳＹ．Ｉ．２参照）によりＶ@から第３スイッチに応じて，パラメータ$\{n, \boldsymbol{A}_i, \boldsymbol{B}_i, \boldsymbol{T}_i\}$を求める。

$$\Sigma_s^* \left\{ \begin{array}{l} \sigma \boldsymbol{x} = \boldsymbol{A}_i \boldsymbol{x} + \boldsymbol{B}_i \boldsymbol{u} \\ \boldsymbol{y} = \boldsymbol{C}_i \boldsymbol{x} + \boldsymbol{D}_i \boldsymbol{u} \end{array} \right. \qquad i = 1 \text{ or } 2$$

ただし

$$\boldsymbol{C}_i = \begin{cases} \boldsymbol{C}\boldsymbol{T}_i & @=S \\ [\boldsymbol{0} \quad \boldsymbol{I}_p]\boldsymbol{T}_i & @=T \\ [\boldsymbol{I}_p \quad \boldsymbol{0}]\boldsymbol{T}_i & @=M \\ [\boldsymbol{0} \quad \boldsymbol{I}_p]\boldsymbol{T}_i & @=D \end{cases}$$

Step　4：第2スイッチが／Oのとき，Σ_s^* の双対モデルをΣ_s^*とする。

実　行　例

〔1〕　状態空間表現形式システムの最小実現を可制御対が Luenberger の第２正準形となるように取得する例

```
/DPACS/ TIME-17:15:10 CPU-00:00:02
CMR/S EX6
 SYSTEM EX6        HAS THE FOLLOWING SYSTEMDATAS
  1) EX6.SO        (    3) ;SYSIN    /S /C /  /    (88/ 3/ 4) :
  2) EX6.S1        (    1) ;CMR      /S /  /  /    (88/ 3/ 4) :
 WHICH SYSTEMDATA IS USED ? :   1
 TO OBTAIN CANONICAL MINIMAL REALIZATION
 WHICH SUBSPACE IS REMOVED ,
 (1) UNCONTROLLABLE ONE OR (2) UNOBSERVABLE ONE ? :   1
 SPECIFY TYPE OF CANONICAL FORM (1 OR 2) :   2
 GIVE EPSILON, TO DETERMINE RANK OF MATRIX V@ EPS= 1.0D-10
 COLUMN INDEPENDENCY OF MATRIX V@ IS INVESTIGATED
  ( 1)
   1
  ( 2)
   1
  ( 3)
   0
 CONDITION NUMBERS ARE AS FOLLOWS
 --- ( 3, 1) MATRIX CNUM      ---
 ***
 ( 1)  1.000000000D+00
 ( 2)  6.959814860-01
 ( 3)  1.54294184D-16
 CHANGE THE EPSILON ?   N
 END ?   N
 WHICH SUBSPACE IS REMOVED ,
```

```
(1) UNCONTROLLABLE ONE OR (2) UNOBSERVABLE ONE ? :   2
SPECIFY TYPE OF CANONICAL FORM (1 OR 2) :   1
GIVE EPSILON, TO DETERMINE RANK OF MATRIX V@ EPS= 1.0D-10
COLUMN INDEPENDENCY OF MATRIX V@ IS INVESTIGATED
 ( 1)
  1
 ( 2)
  0
CONDITION NUMBERS ARE AS FOLLOWS
--- ( 2, 1) MATRIX CNUM      ---
( 1)  1.00000000D+00
( 2)  7.94410929D-16
CHANGE THE EPSILON ?   N
END ? __
GIVE COMMENT FOR SYSTEMDATA :
CMR/S EX6
/DPACS/ TIME-17:19:44 CPU-00:00:13

/DPACS/ TIME-17:15:11 CPU-00:00:02
TYPE/S EX6
 SYSTEM EX6      HAS THE FOLLOWING SYSTEMDATAS
  1) EX6.S0      (    3) ;SYSIN   /S /C /  /    (88/ 3/ 4) :
  2) EX6.S1      (    1) ;CMR     /S /  /  /    (88/ 3/ 4) :
  3) EX6.S2      (    1) ;CMR     /S /  /  /    (88/ 3/ 4) :CMR/S EX6
WHICH SYSTEMDATA IS USED ? :   3
TO TYPE OUT SYSTEMDATA
SYSTEMDATA EX6.S2       ( 1, 1, 1)

--- ( 1, 1) MATRIX   A      ---
( 1) -2.00000000D+00
FILENAME : __
--- ( 1, 1) MATRIX   B      ---
( 1)  1.00000000D+00
FILENAME : __
--- ( 1, 1) MATRIX   C      ---
( 1)  1.00000000D+00
FILENAME : __
--- ( 1, 1) MATRIX   D      ---
( 1)  0.0
FILENAME : __
/DPACS/ TIME-17:19:45 CPU-00:00:13
```

〔2〕 伝達関数行列表現形式システムの最小実現を可制御対が Luenberger の第２正準形となるように取得する例

```
 /DPACS/ TIME-17:15:13 CPU-00:00:02
TYPE/T EX11
 TO TYPE OUT SYSTEMDATA
 NORMALIZED FORM ?   N
 SYSTEMDATA EX11.TO       ( 2, 2, 2)
 1) DENOMINATER PART :
 Q(S)=Q0*S**N+Q1*S**(N-1)+...+Q(N-1)*S+Q(N)
   Q( 0) =    1.00000000D+00
   Q( 1) =    6.00000000D+00
   Q( 2) =    5.00000000D+00
 2) NUMERATER PART :
 P(S)=P0*S**N+P1*S**(N-1)+...+P(N-1)*S+P(N)
 --- ( 2, 2) MATRIX  P0        ---
 ( 1)  0.0              0.0
 ( 2)  0.0              0.0
 FILENAME : 
 --- ( 2, 2) MATRIX  P1        ---
 ( 1)  0.0              0.0
 ( 2)  1.00000000D+00   1.00000000D+00
 FILENAME : 
 --- ( 2, 2) MATRIX  P2        ---
 ( 1)  5.00000000D+00   4.00000000D+00
 ( 2)  3.00000000D+00   2.00000000D+00
 FILENAME : 
 3) DIRECT PART
 --- ( 2, 2) MATRIX  D         ---
 ( 1)  0.0              0.0
 ( 2)  0.0              0.0
 FILENAME : 
 /DPACS/ TIME-17:19:47 CPU-00:00:13

 /DPACS/ TIME-17:15:14 CPU-00:00:02
CMR/T EX11
 TO OBTAIN CANONICAL MINIMAL REALIZATION
 WHICH PAIR IN MINIMAL REALIZATION IS TRANSFORMED
 INTO CANONICAL FORM, (1) (A,B) OR (2) (A,C) ? :    1
 SPECIFY TYPE OF CANONICAL FORM (1 OR 2) :   2
 GIVE EPSILON, TO DETERMINE RANK OF MATRIX V@ EPS= 1.0D-10
 COLUMN INDEPENDENCY OF MATRIX V@ IS INVESTIGATED
  ( 1)
   1   1
  ( 2)
   1   1
  ( 3)
   0   0
 CONDITION NUMBERS ARE AS FOLLOWS
 --- ( 3, 2) MATRIX CNUM      ---
 ( 1)  1.00000000D+00  3.27641020D-01
 ( 2)  3.10776640D-01  2.90567607D-01
 ( 3)  2.02412872D-24  0.0
 CHANGE THE EPSILON ?  N
 GIVE COMMENT FOR SYSTEMDATA :
CMR/T EX11
 /DPACS/ TIME-17:19:48 CPU-00:00:13
```

```
/DPACS/ TIME-17:15:16 CPU-00:00:02
TYPE EX11
 SPECIFY TYPE OF SYSTEMDATA :   S
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX11.SO      ( 2, 2, 4)

 --- ( 4, 4) MATRIX   A        ---
 ( 1)  0.0              1.00000000D+00  0.0              0.0
 ( 2) -5.00000000D+00 -6.00000000D+00 -1.11993748D-14 -4.21884749D-15
 ( 3)  0.0              0.0              0.0              1.00000000D+00
 ( 4)  1.07830411D-14  4.88498131D-15 -5.00000000D+00 -6.00000000D+00
 FILENAME : _
 --- ( 4, 2) MATRIX   B        ---
 ( 1)  0.0              0.0
 ( 2)  1.00000000D+00  0.0
 ( 3)  0.0              0.0
 ( 4)  0.0              1.00000000D+00
 FILENAME : _
 --- ( 2, 4) MATRIX   C        ---
 ( 1)  5.00000000D+00  0.0              4.00000000D+00  0.0
 ***
 ( 2)  3.00000000D+00  1.00000000D+00  2.00000000D+00  1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX   D        ---
 ( 1)  0.0              0.0
 ( 2)  0.0              0.0
 FILENAME : _
 /DPACS/ TIME-17:19:50 CPU-00:00:13
```

〔3〕 マルコフパラメータ表現形式システムの最小実現を可制御対がLuenberger の第2正準形となるように取得する例

```
 /DPACS/ TIME-17:15:18 CPU-00:00:02
CMR/M EX1
 TO OBTAIN CANONICAL MINIMAL REALIZATION
 WHICH PAIR IN MINIMAL REALIZATION IS TRANSFORMED
 INMO CANONICAL FORM, (1) (A,B) OR (2) (A,C) ? :   1
 SPECIFY TYPE OF CANONICAL FORM (1 OR 2) :   2
 SPECIFY NO. OF LOW & COLUMN BLOCKS OF VM (1-50) , (1-50) :   4,4
 GIVE EPSILON, TO DETERMINE RANK OF MATRIX V@ EPS= 1.0D-10
 COLUMN INDEPENDENCY OF MATRIX V@ IS INVESTIGATED
  ( 1)
   1   1
  ( 2)
   1   0
  ( 3)
   0   0
 CONDITION NUMBERS ARE AS FOLLOWS
 --- ( 3, 2) MATRIX CNUM     ---
 ( 1)  1.00000000D+00  4.95680402D+00
 ( 2)  2.88204368D+00  4.36630013D-16
 ( 3)  8.37040929D-15  1.21908267D-15
 ***
 CHANGE THE EPSILON ?   N
 GIVE COMMENT FOR SYSTEMDATA :
EX1 CMR
/DPACS/ TIME-17:19:53 CPU-00:00:13
```

```
/DPACS/ TIME-17:15:20 CPU-00:00:02
TYPE/S EX1
 SYSTEM EX1         HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0         (    3) ;SYSIN    /S /C /  /    (88/ 2/24) :
  2) EX1.S1         (    3) ;CMR      /M /C /2 /    (88/ 3/ 4) :EX1 CMR
 WHICH SYSTEMDATA IS USED ? :   2
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1.S1        ( 2, 2, 3)

 --- ( 3, 3) MATRIX    A          ---
 ( 1)  0.0                1.000000000D+00  0.0
 ( 2)  1.000000000D+00 -1.000000000D+00  2.000000000D+00
 ( 3)  3.000000000D+00  0.0               1.000000000D+00
 FILENAME :
 --- ( 3, 2) MATRIX    B          ---
 ( 1)  0.0                0.0
 ( 2)  1.000000000D+00 -2.000000000D+00
 ( 3)  0.0                1.000000000D+00
 FILENAME :
 --- ( 2, 3) MATRIX    C          ---
 ( 1)  1.000000000D+00  0.0               0.0
 ( 2) -4.440892100D-16  1.000000000D+00  1.000000000D+00
 FILENAME :
 --- ( 2, 2) MATRIX    D          ---
 ( 1)  0.0                0.0
 ( 2)  0.0                0.0
 FILENAME :
/DPACS/ TIME-17:19:54 CPU-00:00:14
```

〔4〕 微分（差分）方程式表現形式システムの最小実現を可制御対が Luenberger の第2正準形となるように取得する例

```
/DPACS/ TIME-17:15:21 CPU-00:00:02
CMR/D EX8
 TO OBTAIN CANONICAL MINIMAL REALIZATION
 SPECIFY TYPE OF CANONICAL FORM (1 OR 2) :    2
 GIVE EPSILON, TO DETERMINE RANK OF MATRIX V@ EPS= 1.0D-10
 COLUMN INDEPENDENCY OF MATRIX V@ IS INVESTIGATED
  ( 1)
   1    1
  ( 2)
   1    0
  ( 3)
   0    0
 CONDITION NUMBERS ARE AS FOLLOWS
 --- ( 3, 2) MATRIX CNUM      ---
 ( 1)  1.000000000D+00  1.17013270D+00
 ( 2)  9.15931411D-01  5.92175494D-16
 ( 3)  2.36802512D-16  1.72910068D-16
 CHANGE THE EPSILON ?   N
 GIVE COMMENT FOR SYSTEMDATA :
CMR/D EX8
/DPACS/ TIME-17:20:07 CPU-00:00:14
```

```
 /DPACS/ TIME-17:15:26 CPU-00:00:02
TYPE/S EX8
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX8.SO       ( 2, 2, 3)

 --- ( 3, 3) MATRIX   A        ---
 ( 1)  0.0               1.00000000D+00  0.0
 ( 2)  1.00000000D+00 -1.00000000D+00  2.00000000D+00
 ( 3)  3.00000000D+00  0.0             1.00000000D+00
 FILENAME : _
 --- ( 3, 2) MATRIX   B        ---
 ( 1)  0.0               0.0
 ( 2)  1.00000000D+00 -2.00000000D+00
 ( 3)  0.0               1.00000000D+00
 FILENAME : _
 --- ( 2, 3) MATRIX   C        ---
 ( 1)  1.00000000D+00  0.0             0.0
 ( 2) -6.93889390D-17  1.00000000D+00  1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX   D        ---
 ( 1)  0.0               0.0
 ( 2)  0.0               0.0
 FILENAME : _
 /DPACS/ TIME-17:20:09 CPU-00:00:14
```

伝達関数行列表現形式システムの取得	D P. Ⅱ. 3
T F (／S) ␣ S N	

機　　能

状態空間表現形式システムΣ_s(SN)から，対応する伝達関数行列表現Σ_Tを求める。

◆第１スイッチは，システムデータの形式を示し，／Sのみである。第１スイッチを付加しなかった場合には，以下のメッセージが出力される。ここで，システムデータの形式を指定することができる。

SPECIFY TYPE OF SYSTEM DATA :

理論概要[9]

Σ_Tは次式を用いて計算できる。

$$\boldsymbol{G}(s)=\boldsymbol{C}\ (s\boldsymbol{I}-\boldsymbol{A})^{-1}\boldsymbol{B}+\boldsymbol{D}$$

<アルゴリズム>(Souriau－Frame－Faddeev の方法)

Step 1 ： $\boldsymbol{P}_0=\boldsymbol{0}$, $q_0=1$, $n_T=n_s$

Step 2 ： $\boldsymbol{X}_1=\boldsymbol{I}_n$ とおく

Step 3 ： $k=1, \sim n_T$まで繰り返す

$$\begin{cases} \boldsymbol{P}_k=\boldsymbol{C}\cdot\boldsymbol{X}_k\cdot\boldsymbol{B} \\ q_k=-\mathrm{tr}\,(\boldsymbol{A}\boldsymbol{X}_k)\,/\,k \\ \boldsymbol{X}_{k+1}=\boldsymbol{A}\boldsymbol{X}_k+q_k\boldsymbol{I}_n \end{cases}$$

ここで，$\boldsymbol{P}_i$, q_iは伝達関数行列表現形式システム内の係数。

実 行 例

〔1〕 状態空間表現形式システムから伝達関数行列表現形式システムへの変換例

```
/DPACS/ TIME-17:15:28 CPU-00:00:02
TF/S EX1
 SYSTEM EX1       HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0       (    3) ;SYSIN   /S /C /  /    (87/11/27) :EX1 TEST DATA
  2) EX1.S1       (    3) ;COOD    /1 /  /  /    (87/11/30) :COOD EX1
  3) EX1.S2       (    3) ;COOD    /1 /  /  /    (87/11/30) :COOD/1 EX1
  4) EX1.S3       (    3) ;COOD    /2 /  /  /    (87/11/30) :COOD/2 EX1
  5) EX1.S4       (    3) ;CMR     /M /C /2 /    (87/11/30) :EX1 CMR
 WHICH SYSTEMDATA IS USED ? :   1
 TO OBTAIN TRANSFER FUNCTION
 GIVE COMMENT FOR SYSTEMDATA :
TF/S EX1
 /DPACS/ 22: 7: 1
TYPE/T EX1
 TO TYPE OUT SYSTEMDATA
 NORMALIZED FORM ?   N
 SYSTEMDATA EX1.T0        ( 2, 2, 3)

 1) DENOMINATER PART :
 Q(S)=Q0*S**N+Q1*S**(N-1)+...+Q(N-1)*S+Q(N)
   Q( 0) =    1.00000000D+00
   Q( 1) =    0.0
   Q( 2) =   -2.00000000D+00
   Q( 3) =   -5.00000000D+00
 2) NUMERATER PART :
 P(S)=P0*S**N+P1*S**(N-1)+...+P(N-1)*S+P(N)
 --- ( 2, 2) MATRIX  P0       ---
 ( 1)  0.0              0.0
 ( 2)  0.0              0.0
 FILENAME : _
 --- ( 2, 2) MATRIX  P1       ---
 ( 1)  0.0              0.0
 ( 2)  1.00000000D+00 -1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  P2       ---
 ( 1)  1.00000000D+00 -2.00000000D+00
 ( 2) -1.00000000D+00  5.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  P3       ---
 ( 1) -1.00000000D+00  4.00000000D+00
 ( 2)  3.00000000D+00 -7.00000000D+00
 FILENAME : _
 3) DIRECT PART
 --- ( 2, 2) MATRIX  D        ---
 ( 1)  0.0              0.0
 ( 2)  0.0              0.0
 FILENAME : _
/DPACS/ TIME-17:20:12 CPU-00:00:14
```

マルコフパラメータ表現形式システムの取得	DP. II. 4
MRKV（[／@]）_SN　　@＝S, T, D	

機　能

状態空間表現形式システムΣ_S(SN)，伝達関数行列表現形式システムΣ_T(SN)，微分（差分）方程式表現形式システムΣ_D(SN)から，対応するマルコフパラメータ表現形式システムΣ_Mを求める。

◆第1スイッチ／@は，システムデータの形式を示し，@＝S，T，Dの3種類である。第1スイッチを付加しなかった場合には，以下のメッセージが出力される。ここで，システムデータの形式を指定することができる。

SPECIFY TYPE OF SYSTEM DATA :

理論概要

◆<$\Sigma_S \rightarrow \Sigma_M$のアルゴリズム>

Step 1 : $n_M = 2n_S + 1$

Step 2 : $k = 1, \sim, n_M$に対し，　$\boldsymbol{M}_k = \boldsymbol{C} \cdot \boldsymbol{A}^{k-1} \cdot \boldsymbol{B}$

◆<$\Sigma_T \rightarrow \Sigma_M$のアルゴリズム>

Step 1 : $n_M = 2pn_T + 1$

Step 2 : $\boldsymbol{W}_{i1} = \boldsymbol{P}_{n_T-i+1}\ (i = 1, \sim, n_T)$　$\boldsymbol{M}_1 = \boldsymbol{W}_{n_T 1}$

Step 3 : $j = 2, \sim, n_M$に対し

$$\begin{cases} \boldsymbol{W}_{1j} = -q_{n_T}\boldsymbol{W}_{n_T\,j-1}, \quad \boldsymbol{W}_{ij} = \boldsymbol{W}_{i-1\,j-1} - q_{n_T-j+1} \cdot \boldsymbol{W}_{n_T\,j-1} \\ \qquad (i = 2, \sim, n_T) \\ \boldsymbol{M}_j = \boldsymbol{W}_{n_T j} \end{cases}$$

◆<$\Sigma_D \rightarrow \Sigma_M$のアルゴリズム>

Step 1 : $n_M = 2pn_D + 1$

Step 2 : $\boldsymbol{W}_{i1} = \boldsymbol{U}_{n_D-i+1}\ (i = 1, \sim, n_D)$，$\boldsymbol{M}_1 = \boldsymbol{W}_{n_D 1}$

Step 3 : $j = 2, \sim, n_M$に対し

$$\begin{cases} \boldsymbol{W}_{1j} = -\boldsymbol{T}_{n_D} \cdot \boldsymbol{W}_{n_D\,j-1}, \quad \boldsymbol{W}_{ij} = \boldsymbol{W}_{i-1\,j-1} - \boldsymbol{T}_{n_D-i+1} \cdot \boldsymbol{W}_{n_D\,j-1} \\ \qquad (i = 2, \sim, n_D) \\ \boldsymbol{M}_j = \boldsymbol{W}_{n_D j} \end{cases}$$

実 行 例

〔1〕 状態空間表現形式システムからマルコフパラメータ表現形式システムを取得する例

```
 /DPACS/ TIME-17:15:30 CPU-00:00:02
MRKV/S EX1
 SYSTEM EX1        HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0        (    3) ;SYSIN    /S /C /  /    (87/11/27) :EX1 TEST DATA
  2) EX1.S1        (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD EX1
  3) EX1.S2        (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD/1 EX1
  4) EX1.S3        (    3) ;COOD     /2 /  /  /    (87/11/30) :COOD/2 EX1
 WHICH SYSTEMDATA IS USED ? :    1
 TO OBTAIN MARKOV PARAMETERS
 GIVE COMMENT FOR SYSTEMDATA :
MRKV
 /DPACS/ 21:54:37
TYPE/M EX1
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1.MO       ( 2, 2, 7)

 --- ( 2, 2) MATRIX  D        ---
 ( 1)  0.0              0.0
 ( 2)  0.0              0.0
 FILENAME :
 --- ( 2, 2) MATRIX  M1       ---
 ( 1)  0.0              0.0
 ( 2)  1.00000000D+00 -1.00000000D+00
 FILENAME :
 --- ( 2, 2) MATRIX  M2       ---
 ( 1)  1.00000000D+00 -2.00000000D+00
 ( 2) -1.00000000D+00  5.00000000D+00
 FILENAME :
 --- ( 2, 2) MATRIX  M3       ---
 ( 1) -1.00000000D+00  4.00000000D+00
 ( 2)  5.00000000D+00 -9.00000000D+00
 FILENAME :
 --- ( 2, 2) MATRIX  M4       ---
 ( 1)  2.00000000D+00 -4.00000000D+00
 ( 2)  3.00000000D+00  5.00000000D+00
 FILENAME :
 --- ( 2, 2) MATRIX  M5       ---
 ( 1)  3.00000000D+00 -2.00000000D+00
 ( 2)  5.00000000D+00  7.00000000D+00
 FILENAME :
 --- ( 2, 2) MATRIX  M6       ---
 ( 1) -1.00000000D+00  1.20000000D+01
 ( 2)  3.10000000D+01 -3.50000000D+01
 FILENAME :
 --- ( 2, 2) MATRIX  M7       ---
 ( 1)  1.60000000D+01 -2.40000000D+01
 ( 2)  2.50000000D+01  3.90000000D+01
 FILENAME :
 /DPACS/ TIME-17:20:13 CPU-00:00:14
```

〔2〕 伝達関数行列表現形式システムからマルコフパラメータ表現形式システムを取得する例

```
 /DPACS/ TIME-17:15:32 CPU-00:00:02
MRKV/T EX1
 TO OBTAIN MARKOV PARAMETERS
 GIVE COMMENT FOR SYSTEMDATA :
MRKV/T
 /DPACS/ 22:11: 3
TYPE/M EX1
 SYSTEM EX1        HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.M0        (    7) ;MRKV     /S /  /  /    (87/11/30) :MRKV
  2) EX1.M1        (   13) ;MRKV     /T /  /  /    (87/11/30) :MRKV/T
 WHICH SYSTEMDATA IS USED ? :   2
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1.M1        ( 2, 2,13)

 --- ( 2, 2) MATRIX  D         ---
 ( 1)  0.0              0.0
 ( 2)  0.0              0.0
 FILENAME : _
 --- ( 2, 2) MATRIX  M1        ---
 ( 1)  0.0              0.0
 ( 2)  1.00000000D+00 -1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  M2        ---
 ( 1)  1.00000000D+00 -2.00000000D+00
 ( 2) -1.00000000D+00  5.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  M3        ---
 ( 1) -1.00000000D+00  4.00000000D+00
 ( 2)  5.00000000D+00 -9.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  M4        ---
 ( 1)  2.00000000D+00 -4.00000000D+00
 ( 2)  3.00000000D+00  5.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  M5        ---
 ( 1)  3.00000000D+00 -2.00000000D+00
 ( 2)  5.00000000D+00  7.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  M6        ---
 ( 1) -1.00000000D+00  1.20000000D+01
 ( 2)  3.10000000D+01 -3.50000000D+01
 FILENAME :
 --- ( 2, 2) MATRIX  M7        ---
 ( 1)  1.60000000D+01 -2.40000000D+01
 ( 2)  2.50000000D+01  3.90000000D+01
 FILENAME : _
 --- ( 2, 2) MATRIX  M8        ---
 ( 1)  1.30000000D+01  1.40000000D+01
 ( 2)  8.70000000D+01 -3.50000000D+01
 FILENAME : _
 --- ( 2, 2) MATRIX  M9        ---
 ( 1)  2.70000000D+01  1.20000000D+01
 ( 2)  2.05000000D+02 -9.70000000D+01
 FILENAME : _
```

```
--- ( 2, 2) MATRIX  M10        ---
( 1)  1.06000000D+02 -9.20000000D+01
( 2)  2.99000000D+02  1.25000000D+02
FILENAME : __
--- ( 2, 2) MATRIX  M11        ---
( 1)  1.19000000D+02  9.40000000D+01
( 2)  8.45000000D+02 -3.69000000D+02
FILENAME : __
--- ( 2, 2) MATRIX  M12        ---
( 1)  3.47000000D+02 -1.24000000D+02
( 2)  1.62300000D+03 -2.35000000D+02
FILENAME : __
--- ( 2, 2) MATRIX  M13        ---
( 1)  7.68000000D+02 -2.72000000D+02
( 2)  3.18500000D+03 -1.13000000D+02
FILENAME : __
/DPACS/ TIME-17:20:15 CPU-00:00:14
```

〔3〕 微分（差分）方程式表現形式システムからマルコフパラメータ表現形式システムを取得する例

```
 /DPACS/ TIME-17:15:31 CPU-00:00:02
MRKV/D EX8
 TO OBTAIN MARKOV PARAMETERS
 GIVE COMMENT FOR SYSTEMDATA :
MRKV/D
 /DPACS/ 22:14:59
TYPE/M EX8
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX8.MO        ( 2, 2,13)

 --- ( 2, 2) MATRIX  D         ---
 ( 1)  0.0             0.0
 ( 2)  0.0             0.0
 FILENAME :__
 --- ( 2, 2) MATRIX  M1        ---
 ( 1)  0.0             0.0
 ( 2)  1.00000000D+00 -1.00000000D+00
 FILENAME :__
 --- ( 2, 2) MATRIX  M2        ---
 ( 1)  1.00000000D+00 -2.00000000D+00
 ( 2) -1.00000000D+00  5.00000000D+00
 FILENAME :__
 --- ( 2, 2) MATRIX  M3        ---
 ( 1) -1.00000000D+00  4.00000000D+00
 ( 2)  5.00000000D+00 -9.00000000D+00
 FILENAME :__
 --- ( 2, 2) MATRIX  M4        ---
 ( 1)  2.00000000D+00 -4.00000000D+00
 ( 2)  3.00000000D+00  5.00000000D+00
 FILENAME :
```

```
--- ( 2, 2) MATRIX  M5          ---
( 1)   3.00000000D+00 -2.00000000D+00
( 2)   5.00000000D+00  7.00000000D+00
FILENAME :
--- ( 2, 2) MATRIX  M6          ---
( 1)  -1.00000000D+00  1.20000000D+01
( 2)   3.10000000D+01 -3.50000000D+01
FILENAME :
--- ( 2, 2) MATRIX  M7          ---
( 1)   1.60000000D+01 -2.40000000D+01
( 2)   2.50000000D+01  3.90000000D+01
FILENAME :
--- ( 2, 2) MATRIX  M8          ---
( 1)   1.30000000D+01  1.40000000D+01
( 2)   8.70000000D+01 -3.50000000D+01
FILENAME :
--- ( 2, 2) MATRIX  M9          ---
( 1)   2.70000000D+01  1.20000000D+01
( 2)   2.05000000D+02 -9.70000000D+01
FILENAME :
--- ( 2, 2) MATRIX  M10         ---
( 1)   1.06000000D+02 -9.20000000D+01
( 2)   2.99000000D+02  1.25000000D+02
FILENAME :
--- ( 2, 2) MATRIX  M11         ---
( 1)   1.19000000D+02  9.40000000D+01
( 2)   8.45000000D+02 -3.69000000D+02
FILENAME :
--- ( 2, 2) MATRIX  M12         ---
( 1)   3.47000000D+02 -1.24000000D+02
( 2)   1.62300000D+03 -2.35000000D+02
FILENAME :
--- ( 2, 2) MATRIX  M13         ---
( 1)   7.68000000D+02 -2.72000000D+02
( 2)   3.18500000D+03 -1.13000000D+02
FILENAME :
/DPACS/ TIME-17:20:16 CPU-00:00:14
```

微分（差分）方程式表現形式システムの取得	DP. II. 5
DIF（／S）␣SN	

機　能

状態空間表現形式システムΣ_S（SN）から，対応する微分（差分）方程式表現形式システムΣ_Dを求める。

◆第1スイッチは，システムデータの形式を示し，／Sのみである。第1スイッチを付加しなかった場合には以下のメッセージが出力される。ここで，システムデータの形式を指定することができる。

SPECIFY TYPE OF SYSTEM DATA ：

理論概要

$\Sigma_S \to \Sigma_D$は一意的では無いので，$\Sigma_S \to \Sigma_T$としΣ_TをΣ_Dとして認識させる方法をとる。

<アルゴリズム>

Step 1 ： $\boldsymbol{U}_0=\boldsymbol{0}$，$\boldsymbol{T}_0=\boldsymbol{I}_P$，$n_D=n_S$

Step 2 ： $\boldsymbol{X}_1=\boldsymbol{I}_n$

Step 3 ： k＝1，～，n_Dに対し下記を繰り返す。

$$\begin{cases} \boldsymbol{U}_k=\boldsymbol{C}\boldsymbol{X}_k\boldsymbol{B} \\ q_k=\mathrm{tr}(\boldsymbol{A}\boldsymbol{X}_k)/k \\ \boldsymbol{T}_k=q_k\boldsymbol{I}_P \\ \boldsymbol{X}_{k+1}=\boldsymbol{A}\boldsymbol{X}_k+q_k\boldsymbol{I}_n \end{cases}$$

実 行 例

〔1〕 状態空間表現形式システムから微分（差分）方程式表現形式を取得する例

```
/DPACS/ TIME-17:15:34 CPU-00:00:02
DIF/S EX1
 SYSTEM EX1        HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0        (    3) ;SYSIN    /S /C /  /    (87/11/27) :EX1 TEST DATA
  2) EX1.S1        (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD EX1
  3) EX1.S2        (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD/1 EX1
  4) EX1.S3        (    3) ;COOD     /2 /  /  /    (87/11/30) :COOD/2 EX1
  5) EX1.S4        (    3) ;CMR      /M /C /2 /    (87/11/30) :EX1 CMR
 WHICH SYSTEMDATA IS USED ? :   1
 TO OBTAIN DIFFERENTIAL (DIFFERENCE) EQUATION
 GIVE COMMENT FOR SYSTEMDATA :
DIF/S
 /DPACS/ 22:20:32
TYPE/D EX1
 TO TYPE OUT SYSTEMDATA
 NORMALIZED FORM ?
 SYSTEMDATA EX1.D0        ( 2, 2, 3)

 1) DENOMINATER PART :
 T(S)=T0*S**N+T1*S**(N-1)+...+T(N-1)*S+T(N)
 --- ( 2, 2) MATRIX  T0        ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  0.0             1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  T1        ---
 ( 1)  0.0             0.0
 ( 2)  0.0             0.0
 FILENAME : _
 --- ( 2, 2) MATRIX  T2        ---
 ( 1) -2.00000000D+00  0.0
 ( 2)  0.0            -2.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  T3        ---
 ( 1) -5.00000000D+00  0.0
 ( 2)  0.0            -5.00000000D+00
 FILENAME : _
 2) NUMERATER PART :
 U(S)=U0*S**N+U1*S**(N-1)+...+U(N-1)*S+U(N)
 --- ( 2, 2) MATRIX  U0        ---
 ( 1)  0.0             0.0
 ( 2)  0.0             0.0
 FILENAME : _
 --- ( 2, 2) MATRIX  U1        ---
 ( 1)  0.0             0.0
 ( 2)  1.00000000D+00 -1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  U2        ---
 ( 1)  1.00000000D+00 -2.00000000D+00
 ( 2) -1.00000000D+00  5.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX  U3        ---
 ( 1) -1.00000000D+00  4.00000000D+00
 ( 2)  3.00000000D+00 -7.00000000D+00
 FILENAME : _
 3) DIRECT PART
 --- ( 2, 2) MATRIX  D         ---
 ( 1)  0.0             0.0
 ( 2)  0.0             0.0
 FILENAME : _
 /DPACS/ TIME-17:20:18 CPU-00:00:14
```

入出力データ表現形式システムの取得	DP. Ⅱ. 6
IO([/@])_SN　　　@＝S, D	

機　能

状態空間表現形式システムΣ_Sまたは微分（差分）方程式表現システムΣ_Dから，最大周期時系列データ（M－系列）(MSEQ（TS. Ⅲ. 2）参照）を入力とする入出力データ表現形式システムΣ_Iを作成する。

◆第1スイッチ／@は，システムデータの形式を示し，／Sまたは／Dの2種類である。第1スイッチを付加しなかった場合には，以下のメッセージが出力される。ここで，システムデータの形式を指定することができる。

SPECIFY TYPE OF SYSTEM DATA :

（注）取り扱うシステムΣ_Sは，離散系でなければならない。

理論概要

◆第1スイッチが／Sの場合（状態空間表現形式システムΣ_S）

$$\begin{cases} \boldsymbol{x}(k+1) = \boldsymbol{A}\,\boldsymbol{x}(k)+\boldsymbol{B}\,\boldsymbol{u}(k) \\ \boldsymbol{y}(k) = \boldsymbol{C}\,\boldsymbol{x}(k)+\boldsymbol{D}\,\boldsymbol{u}(k) \end{cases}$$

$\boldsymbol{u}(k)$; $k=0, \sim, N$,　$\boldsymbol{x}(0)$ を与えて順次 $\boldsymbol{y}(k)$ を求める。

◆第1スイッチが／Dの場合（微分（差分）方程式表現形式システムΣ_D）

$$\sum_{i=0}^{n_T} \boldsymbol{T}_i \boldsymbol{Y}(k-i) = -\sum_{i=1}^{n_T} \boldsymbol{U}_i \boldsymbol{u}(k-i) + \sum_{i=0}^{n_T} \boldsymbol{T}_i \boldsymbol{D}\boldsymbol{u}(k-i)$$

$$\Longrightarrow \boldsymbol{y}(k) = -\sum_{i=1}^{n_T} \boldsymbol{T}_i \boldsymbol{Y}(k-i) + \sum_{i=1}^{n_T} \boldsymbol{U}_i \boldsymbol{u}(k-i) + \sum_{i=0}^{n_T} \boldsymbol{T}_i \boldsymbol{D}\boldsymbol{u}(k-i)$$

$\boldsymbol{u}(k)$: $k=0, \sim, N$,　$\boldsymbol{u}(k)$, $\boldsymbol{y}(k)$: $k=-n_T, \sim, -1$ のとき
順次 $\boldsymbol{y}(k)$ を求める。

実 行 例

〔1〕 状態空間表現形式システムから入出力データ表現形式システムを取得する例

```
 /DPACS/ TIME-17:15:35 CPU-00:00:02
IO/S EX9SD
 TO MAKE I/O DATA
 SPECIFY LENGTH OF I/O DATA :   100                ①
 NO. OF REGISTERS FOR M-SEQUENCE GENERATER ? :      7
 GIVE LOW & HIGH VALUE OF M-SEQUENCE :     -1,1
 GIVE INITIAL STATE X(0)
 INPUT -MODE OF ( 1, 1) MATRIX X(0)       : Z
 TYPE OR REVISE ? _
 GIVE COMMENT FOR SYSTEMDATA :
IO/S
 /DPACS/ TIME-17:20:19 CPU-00:00:14
```

【説明】

① M系列信号の最大周期（$2^{i}-1$）を規定するビット量iを指定する。

〔2〕 微分（差分）方程式表現形式システムから入出力データ表現形式システムを取得する例

```
 /DPACS/ TIME-17:15:36 CPU-00:00:02
IO/D EX9D
 TO MAKE I/O DATA
 SPECIFY LENGTH OF I/O DATA :   100
 NO. OF REGISTERS FOR M-SEQUENCE GENERATER ? :      7
 GIVE LOW & HIGH VALUE OF M-SEQUENCE :     -1,1
 GIVE INITIAL INPUT U(0) AND OUTPUT Y(0)
 INPUT -MODE OF ( 1, 1) MATRIX U(0)       : Z ①
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 1, 1) MATRIX Y(0)       : Z ①
 TYPE OR REVISE ? _
 GIVE COMMENT FOR SYSTEMDATA :
IO/D EX9D
 /DPACS/ TIME-17:20:21 CPU-00:00:14
```

【説明】

① 時刻 $k=-1, \cdots, -n_T$ での入力 $\boldsymbol{u}(k)$ 及び出力 $\boldsymbol{y}(k)$ を与える。行列 $\boldsymbol{u}(0)$ 及び $\boldsymbol{y}(0)$ には，第 j 行に時刻 $k=-j$ の入力値を入れる。

連続系状態空間表現形式システムの離散系化	DP. Ⅱ. 7
DIGIT (／D) [# ／L] ␣SN␣OSN	

機　能

連続系状態空間表現形式システムΣ_S(SN) に対し，入力したサンプル周期により離散系状態空間表現形式システム$\overline{\Sigma}_S$(OSN) した形式システムを得る。尚，入力側に△tなる遅れが存在する場合，状態変数ベクトルの次数が（元の次数）＋（入力ベクトルの次数）となる様な拡大系を得る。

◆第1スイッチは，計算機が計算して求めたサンプル周期を表示するか否かを示すものであり，／Dを付加した場合には計算機が求めた標準的なサンプル周期を表示しないことを示し，ユーザが必ずサンプル周期Tを入力することを示す。

◆第2スイッチは，入力側に遅れがある場合に，拡大系を求めるためのものである。／Lを付加した場合には，拡大系が求められる。

理論概要

a）遅れが無ければ

$$\overline{\Sigma}_s \begin{cases} \boldsymbol{x}(k+1) = \exp(\boldsymbol{A}T)\cdot\boldsymbol{x}(k)+\left(\int_0^T \exp(\boldsymbol{A}\tau)\,\boldsymbol{B}\,d\tau\right)\cdot\boldsymbol{u}(k) \\ \boldsymbol{y}(k) = \boldsymbol{C}\,\boldsymbol{x}(k)+\boldsymbol{D}\,\boldsymbol{u}(k) \end{cases}$$

を計算。ここでTはサンプル周期を表わす。

b）遅れが有れば，△tを遅れ時間とすると，

$$\overline{\Sigma}_s \begin{cases} \boldsymbol{x}(k+1) = \overline{\boldsymbol{A}}\,\boldsymbol{x}(k)+\overline{\boldsymbol{B}}_1\boldsymbol{u}(k)+\overline{\boldsymbol{B}}_2\boldsymbol{u}(k-1) \\ \boldsymbol{y}(k) = \boldsymbol{C}\,\boldsymbol{x}(k)+\boldsymbol{D}\,\boldsymbol{u}(k) \end{cases}$$

$$\text{又は} \begin{cases} \overline{\boldsymbol{x}}(k+1) = \overline{\overline{\boldsymbol{A}}}\,\overline{\boldsymbol{x}}(k)+\overline{\overline{\boldsymbol{B}}}\,\boldsymbol{u}(k) \\ \boldsymbol{y}(k) = \overline{\overline{\boldsymbol{C}}}\,\overline{\boldsymbol{x}}(k)+\boldsymbol{D}\,\boldsymbol{u}(k) \end{cases}$$

$$\overline{\boldsymbol{x}}(k) = \begin{bmatrix} \boldsymbol{x}(k) \\ \boldsymbol{u}(k-1) \end{bmatrix}$$

$$\overline{\overline{\boldsymbol{A}}} = \begin{bmatrix} \overline{\boldsymbol{A}} & \overline{\boldsymbol{B}}_2 \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix},\quad \overline{\overline{\boldsymbol{B}}} = \begin{bmatrix} \overline{\boldsymbol{B}}_1 \\ \boldsymbol{I}_m \end{bmatrix},\quad \overline{\overline{\boldsymbol{C}}} = [\,\boldsymbol{C},\ \boldsymbol{0}\,]$$

$$\overline{\boldsymbol{A}} = \exp(\boldsymbol{A}T),\quad \overline{\boldsymbol{B}}_1 = \int_0^{T-\triangle t} \exp(\boldsymbol{A}\tau)\cdot\boldsymbol{B}\,d\tau,$$

$$\overline{\boldsymbol{B}}_2 = \exp(\boldsymbol{A}(T-\triangle t))\int_0^{\triangle t} \exp(\boldsymbol{A}\tau)\cdot\boldsymbol{B}\,d\tau$$

尚，ＥＸＰの計算は，級数展開の100次までに収束することを前提としている。
収束判定値は10^{-10} 固定である。

実　行　例

〔1〕　連続系状態空間表現形式システムの離散系化（遅れなし）の例(1)

```
 /DPACS/ TIME-17:15:37 CPU-00:00:02
DIGIT EX1 EX1DI
 TO DIGITIZE CONTINOUS SYSTEM
 INPUT SAMPLING INTERVAL DIRECTLY ?_
 GIVE SAMPLING INTERVAL : 0.01
 GIVE COMMENT FOR SYSTEMDATA :
_____
 KEQ56650I TIME-13:35:24 CPU-00:00:21 SERVICE-195680 SESSION-00:04:45
988
 /DPACS/ 13:35:25
TYPE/S EX1DI
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1DI.SO     ( 2, 2, 3)

 --- ( 3, 3) MATRIX   A        ---
 ( 1)  1.00005083D+00  9.95033458D-03  1.00001667D-04
 ( 2)  1.02503396D-02  9.90100500D-01  2.00006708D-02
 ( 3)  3.01510087D-02  1.50002500D-04  1.01005117D+00
 FILENAME :__
 --- ( 3, 2) MATRIX   B        ---
 ***
 ( 1)  4.97516750D-05 -9.90033389D-05
 ( 2)  9.95050250D-03 -1.98010016D-02
 ( 3)  7.50016667D-07  1.00487558D-02
 FILENAME :__
 --- ( 2, 3) MATRIX   C        ---
 ( 1)  1.00000000D+00  0.0             0.0
 ( 2)  0.0             1.00000000D+00  1.00000000D+00
 FILENAME :__
 --- ( 2, 2) MATRIX   D        ---
 ( 1)  0.0             0.0
 ( 2)  0.0             0.0
 FILENAME :__
 /DPACS/ TIME-17:20:24 CPU-00:00:14
```

〔2〕 連続系状態空間表現形式システムの離散系化（遅れなし）の例(2)

```
 /DPACS/ TIME-17:15:38 CPU-00:00:02
DIGIT EX1 EX1DDD
 TO DIGITIZE CONTINOUS SYSTEM              ①
 INPUT SAMPLING INTERVAL DIRECTLY ?   N
 ADJUST NOMINATED SAMPLING INTERVAL (     0.95485836D-01   )  ? _  ②
 GIVE SAMPLING INTERVAL :  0.01
 GIVE COMMENT FOR SYSTEMDATA :
 ___
 KEQ56650I TIME-11:24:10 CPU-00:00:17 SERVICE-147598 SESSION-00:08:08
988
 /DPACS/ 11:24:10
TYPE/S EX1DDD
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1DDD.SO    ( 2, 2, 3)

 --- ( 3, 3) MATRIX    A        ---
 ( 1)   1.00005083D+00  9.95033458D-03  1.00001667D-04
 ( 2)   1.02503396D-02  9.90100500D-01  2.00006708D-02
 ( 3)   3.01510087D-02  1.50002500D-04  1.01005117D+00
 FILENAME : __
 --- ( 3, 2) MATRIX    B        ---
 ***
 ( 1)   4.97516750D-05 -9.90033389D-05
 ( 2)   9.95050250D-03 -1.98010016D-02
 ( 3)   7.50016667D-07  1.00487558D-02
 FILENAME : __
 --- ( 2, 3) MATRIX    C        ---
 ( 1)   1.00000000D+00  0.0             0.0
 ( 2)   0.0             1.00000000D+00  1.00000000D+00
 FILENAME : __
 --- ( 2, 2) MATRIX    D        ---
 ( 1)   0.0             0.0
 ( 2)   0.0             0.0
 FILENAME : __
 /DPACS/ TIME-17:20:23 CPU-00:00:14
```

【説明】

① ここでNを入力することにより，推奨値を出力する。

② 出力された推奨値であり，ここでNを入力すると，サンプル周期Tの値として推奨値が設定され，Yを入力または，リターンキーのみ押下の場合は，次にTを入力することになる。

〔3〕 連続系状態空間表現形式システムの離散系化（スイッチ指定）の例

```
/DPACS/ TIME-13:49:45 CPU-00:00:21
DIGIT/D EX1 EX1B
 TO DIGITIZE CONTINOUS SYSTEM
 GIVE SAMPLING INTERVAL :  0.01
 GIVE COMMENT FOR SYSTEMDATA :

 /DPACS/ TIME-13:50:57 CPU-00:00:28
TYPE/S EX1;TYPE/S EX1B
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1.SO        ( 2, 2, 3)

 --- ( 3, 3) MATRIX    A        ---
 ( 1)  0.0              1.000000000D+00  0.0
 ( 2)  1.00000000D+00  -1.000000000D+00  2.00000000D+00
 ( 3)  3.00000000D+00   0.0              1.00000000D+00
 FILENAME : _
 --- ( 3, 2) MATRIX    B        ---
 ( 1)  0.0              0.0
 ( 2)  1.00000000D+00  -2.00000000D+00
 ( 3)  0.0              1.000000000D+00
 FILENAME : _
 --- ( 2, 3) MATRIX    C        ---
 ***
 ( 1)  1.00000000D+00   0.0              0.0
 ( 2)  0.0              1.00000000D+00   1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX    D        ---
 ( 1)  0.0              0.0
 ( 2)  0.0              0.0
 FILENAME : _
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1B.SO       ( 2, 2, 3)

 --- ( 3, 3) MATRIX    A        ---
 ( 1)  1.00005083D+00   9.95033458D-03   1.00001667D-04
 ( 2)  1.02503396D-02   9.90100500D-01   2.00006708D-02
 ( 3)  3.01510087D-02   1.50002500D-04   1.01005117D+00
 FILENAME : _
 --- ( 3, 2) MATRIX    B        ---
 ( 1)  4.97516750D-05  -9.90033389D-05
 ( 2)  9.95050250D-03  -1.98010016D-02
 ( 3)  7.50016667D-07   1.00487558D-02
 FILENAME : _
 --- ( 2, 3) MATRIX    C        ---
 ( 1)  1.00000000D+00   0.0              0.0
 ( 2)  0.0              1.00000000D+00   1.00000000D+00
 ***
 FILENAME : _
 --- ( 2, 2) MATRIX    D        ---
 ( 1)  0.0              0.0
 ( 2)  0.0              0.0
 FILENAME : _
 /DPACS/ TIME-13:53:53 CPU-00:00:36
```

〔4〕 連続系状態空間表現形式システムの離散系化（遅れあり）の例

```
 /DPACS/ TIME-17:03:46 CPU-00:00:02
DIGIT/L EX1 EX1DGIT
 TO DIGITIZE CONTINOUS SYSTEM
 INPUT SAMPLING INTERVAL DIRECTLY ?
 GIVE SAMPLING INTERVAL :  0.02
 GIVE CONTROL DELAY  :  0.002
 GIVE COMMENT FOR SYSTEMDATA :

 /DPACS/ TIME-17:46:40 CPU-00:00:20

 /DPACS/ TIME-17:03:43 CPU-00:00:02
TYPE/S EX1DGIT
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1DGIT.SO  ( 2, 2, 5)

 --- ( 5, 5) MATRIX   A      ---
 ( 1)  1.00020668D+00  1.98026866D-02  4.00026933D-04  3.76426437D-05
 ( 2)  2.10027674D-02  9.80403994D-01  4.00054002D-02  1.96273091D-03
 ( 3)  6.06081407D-02  6.00040400D-04  1.02020938D+00  1.08606688D-06
 ( 4)  0.0             0.0             0.0             0.0
 ( 5)  0.0             0.0             0.0             0.0
 ( 1) -7.45612429D-05
 ( 2) -3.84945248D-03
 ( 3)  2.03620608D-03
 ( 4)  0.0
 ( 5)  0.0
 FILENAME :
 --- ( 5, 2) MATRIX   B      ---
 ( 1)  1.60559653D-04 -3.18203094D-04
 ( 2)  1.78409422D-02 -3.53578489D-02
 ***
 ( 3)  4.37431848D-06  1.81547710D-02
 ( 4)  1.00000000D+00  0.0
 ( 5)  0.0             1.00000000D+00
 FILENAME :
 --- ( 2, 5) MATRIX   C      ---
 ( 1)  1.00000000D+00  0.0             0.0             0.0
 ( 2)  0.0             1.00000000D+00  1.00000000D+00  0.0
 ( 1)  0.0
 ( 2)  0.0
 FILENAME :
 --- ( 2, 2) MATRIX   D      ---
 ( 1)  0.0             0.0
 ( 2)  0.0             0.0
 FILENAME :
 /DPACS/ TIME-17:46:38 CPU-00:00:20
```

離散系状態空間表現形式システムの連続系化	DP. II. 8
CONT_SN_OSN	

機　　能

離散系状態空間表現形式システムΣ_s(SN)から，連続系状態空間表現形式システム$\bar{\Sigma}_s$(OSN)を得る。

理論概要

離散型状態空間表現システムΣ_s (注)

$$\begin{cases} \boldsymbol{x}(k+1) = \boldsymbol{A}\,\boldsymbol{x}(k) + \boldsymbol{B}\,\boldsymbol{u}(k) \\ \boldsymbol{y}(k) = \boldsymbol{C}\,\boldsymbol{x}(k) + \boldsymbol{D}\,\boldsymbol{u}(k) \end{cases}$$

から

連続型状態空間表現システム$\bar{\Sigma}_s$

$$\begin{cases} \dfrac{d}{dt}\boldsymbol{x}(t) = \bar{\boldsymbol{A}}\,\boldsymbol{x}(t) + \bar{\boldsymbol{B}}\,\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}\,\boldsymbol{x}(t) + \boldsymbol{D}\,\boldsymbol{u}(t) \end{cases}$$

$$\bar{\boldsymbol{A}} = \frac{1}{T}\ln \boldsymbol{A}\,,\quad \bar{\boldsymbol{B}} = \left[\int_0^T \exp(\bar{\boldsymbol{A}}\,\tau)\,d\tau\right]^{-1}\boldsymbol{B}$$

（注）Σ_s がシステムに遅れを持つ系の拡大系の場合は$\boldsymbol{A}$が正則でないので，$\bar{\Sigma}_s$ を求めることはできない。

実 行 例

〔1〕 離散系状態空間表現形式システムの連続系化の例

```
 /DPACS/ TIME-17:03:48 CPU-00:00:02
CONT EX1DIGIT EX1CONT
 TO MAKE CONTINUOUS SYSTEM
 GIVE COMMENT FOR SYSTEMDATA :
CONT
 KEQ56650I TIME-20:29:05 CPU-00:00:30 SERVICE-481499 SESSION-00:05:54
7,1987
 /DPACS/ 20:29: 5
TYPE/S EX1CONT
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1CONT.SO  ( 2, 2, 3)

 --- ( 3, 3) MATRIX   A        ---
 ( 1) -1.99533445D-12  1.00000000D+00  1.28933114D-11
 ( 2)  1.00000000D+00 -1.00000000D+00  2.00000000D+00
 ( 3)  3.00000000D+00 -9.46293044D-12  1.00000000D+00
 FILENAME : _
 --- ( 3, 2) MATRIX   B        ---
 ( 1) -5.46308094D-13  3.79216244D-13
 ( 2)  1.00000000D+00 -2.00000000D+00
 ( 3)  5.05817610D-13  1.00000000D+00
 FILENAME : _
 --- ( 2, 3) MATRIX   C        ---
 ( 1)  1.00000000D+00  0.0             0.0
 ( 2)  0.0             1.00000000D+00  1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX   D        ---
 ( 1)  0.0             0.0
 ( 2)  0.0             0.0
 FILENAME : _
 /DPACS/ TIME-17:46:41 CPU-00:00:20
```

システムの極による安定判別	DP. Ⅲ. 1
POLE [# / /V] _SN	

機　　能

状態空間表現形式システムΣ_sの極を求め，その漸近安定性を判別する。
◆第1スイッチは，固有ベクトルの画面への出力を行う場合に付加する。

理論概要

連続系と離散系とで扱いが異なる。

連続系　$\frac{d}{dt}\boldsymbol{x} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u}$　の場合

$\boldsymbol{A}$ の固有値（極）の実部が負→漸近安定

離散系　$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k)$　の場合

$\boldsymbol{A}$ の固有値（極）の絶対値が1より小→漸近安定

（注）任意の初期状態について，入力＝0に対する系の応答が$t\to\infty$又は$k\to\infty$で0に収束することを漸近安定と言う。

実 行 例

〔1〕　状態空間表現形式システムの漸近安定性を判別する例

```
 /DPACS/ TIME-17:03:49 CPU-00:00:02
TYPE/S EX1
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1.SO       ( 2, 2, 3)

 --- ( 3, 3) MATRIX   A       ---
 ( 1)  0.0               1.000000000D+00  0.0
 ( 2)  1.000000000D+00  -1.000000000D+00  2.000000000D+00
 ( 3)  3.000000000D+00   0.0              1.000000000D+00
 FILENAME :
 --- ( 3, 2) MATRIX   B       ---
 ( 1)  0.0               0.0
 ( 2)  1.000000000D+00  -2.000000000D+00
 ( 3)  0.0               1.000000000D+00
 FILENAME :
 --- ( 2, 3) MATRIX   C       ---
 ( 1)  1.000000000D+00   0.0              0.0
 ( 2)  0.0               1.000000000D+00  1.000000000D+00
```

```
FILENAME : __
--- ( 2, 2) MATRIX    D        ---
( 1)  0.0              0.0
***
( 2)  0.0              0.0
FILENAME : __
/DPACS/ TIME-17:46:42 CPU-00:00:20

/DPACS/ TIME-17:03:50 CPU-00:00:02
POLE EX1
 TO CALCULATE SYSTEM'S POLES
 POLES OF CONTINUOUS SYSTEM EX1.SO

 ---------------------------------------------------------------
  NO.      REAL                 IMAGINARY              ABSOLUTE
 ---------------------------------------------------------------
 ( 1)    2.09455148D+00     -4.44089210D-16      2.09455148D+00
 ( 2)   -1.04727574D+00      1.13593989D+00      1.54503913D+00
 ( 3)   -1.04727574D+00     -1.13593989D+00      1.54503913D+00
 ---------------------------------------------------------------
 FILENAME : __
 CONTINUOUS SYSTEM EX1.SO        IS UNSTABLE
 /DPACS/ TIME-17:46:44 CPU-00:00:20
```

〔2〕 状態空間表現形式システムの漸近安定性を判別する例（固有ベクトルの画面表示を伴う例）

```
/DPACS/ TIME-17:03:51 CPU-00:00:02
POLE/V EX1
 TO CALCULATE SYSTEM'S POLES
 POLES OF CONTINUOUS SYSTEM EX1.SO

 ---------------------------------------------------------------
  NO.      REAL                 IMAGINARY              ABSOLUTE
 ---------------------------------------------------------------
 ( 1)    2.09455148D+00     -4.44089210D-16      2.09455148D+00
 ( 2)   -1.04727574D+00      1.13593989D+00      1.54503913D+00
 ( 3)   -1.04727574D+00     -1.13593989D+00      1.54503913D+00
 ---------------------------------------------------------------
 FILENAME : __
 CONTINUOUS SYSTEM EX1.SO        IS UNSTABLE
 CORRESPONDING EIGENVECTORS

 --- ( 3, 3) MATRIX   VR        ---
 ( 1)  3.64850494D-01  6.05236038D-01 -1.36099153D-01
 ( 2)  7.64198142D-01 -3.15825031D-01  4.76752253D-01
 ( 3)  1.00000000D+00 -8.52168007D-01 -3.04210983D-02
 ***
 FILENAME : __
 --- ( 3, 3) MATRIX   VI        ---
 ( 1)  1.04083409D-17 -2.79965509D-01  2.94222358D-01
 ( 2)  2.25514052D-17  9.80712843D-01 -1.53531481D-01
 ( 3)  0.0            -6.25783342D-02 -4.14262972D-01
 FILENAME : __
 /DPACS/ TIME-17:46:45 CPU-00:00:20
```

システムの零点の算出	DP. Ⅲ. 2
ZERO_SN	

機　能

等しい次数の入出力ベクトルを持つ状態空間表現形式システム Σ_s の零点を求める。

理論概要

Σ_s の伝達関数行列 $\boldsymbol{G}(s)$ は

$$\boldsymbol{G}(s)=\boldsymbol{C}(s\boldsymbol{I}-\boldsymbol{A})^{-1}\boldsymbol{B}$$

$$\det \boldsymbol{G}(s)=\frac{p(s)}{q(s)},\quad q(s)=\det(s\boldsymbol{I}-\boldsymbol{A})$$

p(s)の零点を Σ_s の零点という

$$p(s)=q(s)\det \boldsymbol{G}(s)=\det\begin{bmatrix}\boldsymbol{A}-s\boldsymbol{I}_n & \boldsymbol{B}\\ \boldsymbol{C} & \boldsymbol{D}\end{bmatrix}\times(-1)^n$$

従って

$$\begin{bmatrix}\boldsymbol{A} & \boldsymbol{B}\\ \boldsymbol{C} & \boldsymbol{D}\end{bmatrix}\boldsymbol{v}=\lambda\begin{bmatrix}\boldsymbol{I} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{0}\end{bmatrix}\boldsymbol{v}$$

なる一般化固有値問題を解いて，零点を計算する。

実 行 例

〔1〕　状態空間表現形式システムの零点の算出例

```
/DPACS/ TIME-17:03:53 CPU-00:00:02
ZERO EX1
 SYSTEM EX1        HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0        (    3) ;SYSIN    /S /C /  /    (87/11/27) :EX1 TEST DATA
  2) EX1.S1        (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD EX1
  3) EX1.S2        (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD/1 EX1
  4) EX1.S3        (    3) ;COOD     /2 /  /  /    (87/11/30) :COOD/2 EX1
  5) EX1.S4        (    3) ;CMR      /M /C /2 /    (87/11/30) :EX1 CMR
 WHICH SYSTEMDATA IS USED ? :   1
 TO CALCULATE SYSTEM'S ZEROS
 ZEROS OF CONTINUOUS SYSTEM EX1.S0
 ------------------------------------------------------------------------
  NO.      REAL                     IMAGINARY              ABSOLUTE
 ------------------------------------------------------------------------
 ( 1)    1.08884966D+08          0.0                    1.08884966D+08
 ( 2)   -1.08884959D+08          0.0                    1.08884959D+08
 ------------------------------------------------------------------------
 FILENAME : __
 NO. OF  INFINITE ZEROS  :   3 ①
 NO. OF DEGENERATED CASE :   0 ②
/DPACS/ TIME-17:46:47 CPU-00:00:20
```

【説明】

① 零点の数

② 退化した零点の数（数値的に分母子が約されてしまう場合）

システムの可制御性・可観測性判別	D P. Ⅲ. 3
C C C O ($\begin{bmatrix} /C \\ /O \end{bmatrix}$) _S N	

機　　能

状態空間表現形式システム Σ_s の可制御性または可観測性を調べる。

◆第1スイッチが／Cの場合，可制御性を調べる。

◆第1スイッチが／Oの場合，可観測性を調べる。

理論概要

$\Sigma_s(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C})$とすると，

$$\boldsymbol{V} = [\ \boldsymbol{B},\ \boldsymbol{A}\boldsymbol{B}, \cdots\cdots, \boldsymbol{A}^{n-1}\boldsymbol{B}\ ]$$

rank $\boldsymbol{V} = n$ なら可制御

$${}^t\boldsymbol{N} = [\ {}^t\boldsymbol{C}, {}^t\boldsymbol{A}\,{}^t\boldsymbol{C}, \cdots\cdots, {}^t\boldsymbol{A}^{n-1}\,{}^t\boldsymbol{C}\ ]$$

rank ${}^t\boldsymbol{N} = n$ なら可観測

判定は$\boldsymbol{V}\,{}^t\boldsymbol{V}$,${}^t\boldsymbol{N}\boldsymbol{N}$を求めて，その固有値の絶対値が$10^{-10}$以下のときrank 落ちであると判定する。

実 行 例

〔1〕　状態空間表現形式システムの可観測性・可制御性の判別例

```
/DPACS/ TIME-17:03:55 CPU-00:00:02
CCCO EX1
 SYSTEM EX1       HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0       (    3) ;SYSIN   /S /C /  /    (87/11/27) :EX1 TEST DATA
  2) EX1.S1       (    3) ;COOD    /1 /  /  /    (87/11/30) :COOD EX1
  3) EX1.S2       (    3) ;COOD    /1 /  /  /    (87/11/30) :COOD/1 EX1
  4) EX1.S3       (    3) ;COOD    /2 /  /  /    (87/11/30) :COOD/2 EX1
  5) EX1.S4       (    3) ;CMR     /M /C /2 /    (87/11/30) :EX1 CMR
 WHICH SYSTEMDATA IS USED ? :   1
 TO INVESTIGATE (1) CONTROLLABILITY OR (2) OBSERVABILI TY
 WHICH PROPERTY ? :  2
 EIGENVALUES OF MATRIX N'N ON SYSTEM EX1.S0

 ---------------------------------------------------------------
  NO.      REAL                 IMAGINARY             ABSOLUTE
 ---------------------------------------------------------------
 ( 1)   1.04004315D+02       0.0                 1.04004315D+02
 ( 2)   1.96925905D+01       0.0                 1.96925905D+01
 ( 3)   2.30309401D+00       0.0                 2.30309401D+00
 ---------------------------------------------------------------
 FILENAME :_
 SYSTEM EX1.S0        IS OBSERVABLE ①
/DPACS/ TIME-17:46:52 CPU-00:00:20
```

```
/DPACS/ TIME-17:03:54 CPU-00:00:02
CCCO EX110G
 TO INVESTIGATE (1) CONTROLLABILITY OR (2) OBSERVABILITY
 WHICH PROPERTY ? :  1
 EIGENVALUES; OF MATRIX VV' ON SYSTEM EX110G.SO

 ------------------------------------------------------------
  NO.     REAL                  IMAGINARY             ABSOLUTE
 ------------------------------------------------------------
 ( 1)   3.99990972D-06       0.0                 3.99990972D-06
 ( 2)   3.99990972D-06       0.0                 3.99990972D-06
 ( 3)   4.88165741D-12       0.0                 4.88165741D-12
 ( 4)   4.88165741D-12       0.0                 4.88165741D-12
 ------------------------------------------------------------
 FILENAME : __
 SYSTEM EX110G.SO     IS UNCONTROLLABLE ②
 /DPACS/ TIME-17:46:48 CPU-00:00:20
```

【説明】

① 可観測性の判別

② 可制御性の判別

〔2〕 状態空間表現形式システム・モデルの可制御性の判別（スイッチ指定）例

```
/DPACS/ TIME-17:03:57 CPU-00:00:02
CCCO/C EX1
 SYSTEM EX1        HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.SO       (    3) ;SYSIN    /S /C /  /    (87/11/27) :EX1 TEST DATA
  2) EX1.S1       (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD EX1
  3) EX1.S2       (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD/1 EX1
  4) EX1.S3       (    3) ;COOD     /2 /  /  /    (87/11/30) :COOD/2 EX1
  5) EX1.S4       (    3) ;CMR      /M /C /2 /    (87/11/30) :EX1 CMR
 WHICH SYSTEMDATA IS USED ? :   1
 TO INVESTIGATE CONTROLLABILITY
 EIGENVALUES; OF MATRIX VV' ON SYSTEM EX1.SO

 ------------------------------------------------------------
  NO.     REAL                  IMAGINARY             ABSOLUTE
 ------------------------------------------------------------
 ( 1)   8.77500400D+01       0.0                 8.77500400D+01
 ( 2)   1.08631083D+01       0.0                 1.08631083D+01
 ( 3)   1.38685170D+00       0.0                 1.38685170D+00
 ------------------------------------------------------------
 FILENAME : _
 SYSTEM EX1.SO         IS CONTROLLABLE
 /DPACS/ TIME-17:46:51 CPU-00:00:20
```

〔3〕 状態空間表現形式システム・モデルの可観測性の判別（スイッチ指定）例

```
/DPACS/ TIME-17:03:58 CPU-00:00:02
CCCO/O EX1
 SYSTEM EX1       HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.SO       (    3) ;SYSIN   /S /C /  /   (87/11/27) :EX1 TEST DATA
  2) EX1.S1       (    3) ;COOD    /1 /  /  /   (87/11/30) :COOD EX1
  3) EX1.S2       (    3) ;COOD    /1 /  /  /   (87/11/30) :COOD/1 EX1
  4) EX1.S3       (    3) ;COOD    /2 /  /  /   (87/11/30) :COOD/2 EX1
  5) EX1.S4       (    3) ;CMR     /M /C /2 /   (87/11/30) :EX1 CMR
 WHICH SYSTEMDATA IS USED ? :   1
 TO INVESTIGATE OBSERVABILITY
 EIGENVALUES OF MATRIX N'N ON SYSTEM EX1.SO

 ---------------------------------------------------------------
  NO.     REAL                  IMAGINARY          ABSOLUTE
 ---------------------------------------------------------------
 ( 1)    1.04004315D+02        0.0               1.04004315D+02
 ( 2)    1.96925905D+01        0.0               1.96925905D+01
 ( 3)    2.30309401D+00        0.0               2.30309401D+00
 ---------------------------------------------------------------
 FILENAME :_
 SYSTEM EX1.SO          IS OBSERVABLE
 /DPACS/ TIME-17:46:49 CPU-00:00:20
```

システムの低次元化	D P. Ⅲ. 4

$$\mathrm{RDCT}\left[\begin{array}{l} /\mathrm{E}\left(\left[\begin{array}{l}/\mathrm{N}\\ /\mathrm{P}\end{array}\right]\right) \\ /\mathrm{D}\end{array}\right]\_\mathrm{SN}\_\mathrm{OSN}$$

機　能

ε－最小実現の手法またはE. J. Davison の方法に基づいて，状態空間表現形式システムΣ_sの低次元化モデルを得る。

◆第1スイッチは手法の選択を示し，ε－最小実現の手法を選択する場合には／E，E. J. Davisonの方法により次元縮小を行う場合には／Dとする。

◆第2スイッチは，ε－最小実現を指定した場合にのみ有効で，可観測性グラム行列並びに可制御性グラム行列を入力する場合には／Pを付加し，システムが作成した可観測性・可制御性グラム行列を使用する場合には，第2スイッチを省略するか／Nを付加する。

理論概要

◆＜ε－最小実現の方法＞

Step 1 ： $\boldsymbol{V}\,{}^t\boldsymbol{V}$ と ${}^t\boldsymbol{N}\boldsymbol{N}$ を計算

Step 2 ： $\boldsymbol{V}\,{}^t\boldsymbol{V}\,{}^t\boldsymbol{N}\boldsymbol{N}$ の固有値 $\{\sigma_i\}\ i=1,\sim,n\ (\sigma_i \geq \sigma_{i+1})$ と対応する固有ベクトル $\{\boldsymbol{t}_i\}\ i=1,\ \sim,\ n$ を求める。

Step 3 ： 適当に与えられた $\varepsilon>0$ に対し次式により $\overline{n}$ を求める

$$\sigma_1 \geq \cdots\cdots \geq \sigma_{\overline{n}} > \varepsilon > \sigma_{\overline{n}+1} \geq \cdots\cdots \geq \sigma_n$$

Step 4 ： $\overline{\Sigma}_s$ を次式により求める。

$$(\overline{\boldsymbol{A}},\ \overline{\boldsymbol{B}},\ \overline{\boldsymbol{C}}) = (\boldsymbol{UAT},\ \boldsymbol{UB},\ \boldsymbol{CT})$$

ここで， $\boldsymbol{T} = [\ \alpha_1 \boldsymbol{t}_1 \sim \alpha_{\overline{n}} \boldsymbol{t}_{\overline{n}}\ ] \in \mathbf{R}^{n\times\overline{n}}$

$\boldsymbol{U} = {}^t\boldsymbol{T}\,{}^t\boldsymbol{N}\boldsymbol{N} \in \mathbf{R}^{\overline{n}\times n}$

ただし， $\alpha_i \in \mathbf{R}$ は次式が成立するように決定する

$$\boldsymbol{UT} = \boldsymbol{I}_{\overline{n}}$$

◆＜Davisonの方法＞ [10]

Step 1 ： $\boldsymbol{A}$ の固有値，固有ベクトルを求め，システム Σ_s を対角化し Σ_s^* を得る。

$$\Sigma_s^* \begin{cases} \dfrac{d}{dt}\boldsymbol{x}^*(t) = \boldsymbol{A}^*\boldsymbol{x}^*(t) + \boldsymbol{B}^*\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}^*\boldsymbol{x}^*(t) \end{cases}$$

ただし，$\boldsymbol{A}^*$は固有値の real part の大きいものから順に並ぶようにする。

Step 2 ： $\boldsymbol{A}$ の固有値を見て低次元化モデル $\overline{\Sigma}_s$ の次数 $\overline{n}$ を決定する。

Step 3 以降は＜ε－最小実現の方法＞と同様。なお Step 4 では $\boldsymbol{U}=\boldsymbol{T}^{-1}$，

$\boldsymbol{T} = \{\ \boldsymbol{t}_i\ \}_{i=1,\sim,n}$ で置き換える。

実 行 例

〔1〕 ε－最小実現の手法による状態空間表現形式システムの低次元化モデルの取得例

```
/DPACS/ TIME-17:04:00 CPU-00:00:02
RDCT/E EX1 EX1RDCTE
 SYSTEM EX1        HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0        (    3) ;SYSIN    /S /C /  /    (87/11/27) :EX1 TEST DATA
  2) EX1.S1        (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD EX1
  3) EX1.S2        (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD/1 EX1
  4) EX1.S3        (    3) ;COOD     /2 /  /  /    (87/11/30) :COOD/2 EX1
  5) EX1.S4        (    3) ;CMR      /M /C /2 /    (87/11/30) :EX1 CMR
 WHICH SYSTEMDATA IS USED ? :   1
 TO EXECUTE MODEL REDUCTION
 EIGENVALUES OF MATRIX VV'N'N ON SYSTEM EX1.S0
```

```
------------------------------------------------------------------------------
 NO.      REAL                   IMAGINARY              ABSOLUTE
------------------------------------------------------------------------------
( 1)    3.15527445D+02       0.0                    3.15527445D+02
( 2)    2.24401087D+02       0.0                    2.24401087D+02
( 3)    8.80714677D+01       0.0                    8.80714677D+01
------------------------------------------------------------------------------
FILENAME : _
SPECIFY REDUCED DIMENSION :   2 ①
GIVE COMMENT FOR SYSTEMDATA :
RDCT/E EX1
/DPACS/ TIME-17:46:57 CPU-00:00:20
```

【説明】

① 次元縮小した結果として何次元にするかを入力。

＜低次元化した結果のシステム＞

```
/DPACS/ TIME-17:04:23 CPU-00:00:02
TYPE/S EX1RDCTE
TO TYPE OUT SYSTEMDATA
SYSTEMDATA EX1RDCTE.SO ( 2, 2, 2)

--- ( 2, 2) MATRIX    A       ---
( 1) -2.42075388D-01 -1.31483134D+00
( 2)  1.39911901D+00 -1.48125189D+00
FILENAME : _
--- ( 2, 2) MATRIX    B       ---
( 1) -3.50469983D+00  8.61845888D+00
( 2) -3.85319745D+00  7.22169718D+00
FILENAME : _
--- ( 2, 2) MATRIX    C       ---
( 1)  1.86116927D-01 -2.10098702D-01
( 2) -4.83542536D-01  3.79845308D-01
FILENAME : _
--- ( 2, 2) MATRIX    D       ---
( 1)  0.0             0.0
( 2)  0.0             0.0
FILENAME : _
/DPACS/ TIME-17:47:00 CPU-00:00:20
```

〔2〕 ε－最小実現の手法による状態空間表現形式システムの低次元化モデルの取得例
（可観測性・可制御性グラム行列入力）

```
 /DPACS/ TIME-17:04:25 CPU-00:00:02
RDCT/E/P EX1 EX1EP
 TO EXECUTE MODEL REDUCTION
 INPUT -MODE OF ( 3, 3) MATRIX  VV'       : _
 ( 1) ROW
9,8,5
 ( 2) ROW
3,7,5
 ( 3) ROW
6,8,10
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3, 3) MATRIX  N'N       : _
 ( 1) ROW
7,3,8
 ( 2) ROW
6,4,8
 ( 3) ROW
9,7,2
 TYPE OR REVISE ? _
 EIGENVALUES OF MATRIX VV'N'N ON SYSTEM EX1.SO
```

NO.	REAL	IMAGINARY	ABSOLUTE
(1)	3.75581800D+02	0.0	3.75581800D+02
(2)	2.81371318D+00	0.0	2.81371318D+00
(3)	-1.83955127D+01	0.0	1.83955127D+01

```
 FILENAME : _
 SPECIFY REDUCED DIMENSION :  2
 GIVE COMMENT FOR SYSTEMDATA : _
RDCT/E/P EX1EP
 /DPACS/ TIME-17:47:03 CPU-00:00:20
```

＜低次元化した結果のシステム＞

```
 /DPACS/ TIME-17:04:27 CPU-00:00:02
TYPE/S EX1EP
  TO TYPE OUT SYSTEMDATA
  SYSTEMDATA EX1EP.SO     ( 2, 2, 2)

  --- ( 2, 2) MATRIX    A        ---
  ( 1)   2.37505032D+00  3.72550834D+00
  ( 2)  -1.89306197D+00 -3.71419689D+00
  FILENAME : _
  --- ( 2, 2) MATRIX    B        ---
  ( 1)  -1.94062782D+00  1.59631967D+00
  ( 2)   2.40254416D+00 -3.38810238D+00
  FILENAME : _
  --- ( 2, 2) MATRIX    C        ---
  ( 1)  -1.47165389D-01 -6.43650280D-01
  ( 2)  -2.56617259D-01  9.52073637D-01
  FILENAME : _
  --- ( 2, 2) MATRIX    D        ---
  ( 1)   0.0             0.0
  ( 2)   0.0             0.0
  FILENAME : _
 /DPACS/ TIME-17:47:04 CPU-00:00:20
```

〔3〕 Davisonの手法による状態空間表現形式システムの低次元化モデルの取得例

```
 /DPACS/ TIME-17:04:28 CPU-00:00:02
RDCT/D EX1 EX1RDCTD
 SYSTEM EX1        HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.SO        (    3) ;SYSIN    /S /C /  /   (87/11/27) :EX1 TEST DATA
  2) EX1.S1        (    3) ;COOD     /1 /  /  /   (87/11/30) :COOD EX1
  3) EX1.S2        (    3) ;COOD     /1 /  /  /   (87/11/30) :COOD/1 EX1
  4) EX1.S3        (    3) ;COOD     /2 /  /  /   (87/11/30) :COOD/2 EX1
  5) EX1.S4        (    3) ;CMR      /M /C /2 /   (87/11/30) :EX1 CMR
 WHICH SYSTEMDATA IS USED ? :    1
 TO EXECUTE MODEL REDUCTION
 EIGENVALUES OF MATRIX A ON SYSTEM EX1.SO

 ---------------------------------------------------------------------
  NO.       REAL                  IMAGINARY               ABSOLUTE
 ---------------------------------------------------------------------
 ( 1)    2.09455148D+00        0.0                     2.09455148D+00
 ( 2)   -1.04727574D+00        1.13593989D+00          1.54503913D+00
 ( 3)   -1.04727574D+00       -1.13593989D+00          1.54503913D+00
 ---------------------------------------------------------------------
 FILENAME : _
 SPECIFY REDUCED DIMENSION :   2
 GIVE COMMENT FOR SYSTEMDATA : _
RDCT/D EX1
 /DPACS/ TIME-17:47:06 CPU-00:00:20
```

<低次元化した結果のシステム>

```
 /DPACS/ TIME-17:04:30 CPU-00:00:02
TYPE/S EX1RDCTD
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1RDCTD.SO ( 2, 2, 2)

 --- ( 2, 2) MATRIX    A        ---
 ( 1)  2.09455148D+00  0.0
 ( 2)  0.0            -1.13969312D+00
 FILENAME : _
 --- ( 2, 2) MATRIX    B        ---
 ( 1) -3.52209256D-01  6.08503255D-02
 ( 2)  1.56734971D-01 -2.04310700D-01
 FILENAME : _
 --- ( 2, 2) MATRIX    C        ---
 ( 1) -2.78429526D-01 -4.15861695D-01
 ( 2) -1.34631818D+00  1.25530597D+00
 FILENAME : _
 --- ( 2, 2) MATRIX    D        ---
 ( 1)  3.04010207D-01 -8.82639622D-01
 ( 2) -5.46244393D-01  1.58592354D+00
 FILENAME : _
 /DPACS/ TIME-17:47:07 CPU-00:00:20
```

状態空間表現形式システムの ε 一最小実現取得	D P. Ⅲ. 5
E M R_S N_O S N	

機　　能

状態空間表現形式システムΣ_s(SN) から，ε 一最小実現したモデル（OSN）を，Luenberger の第1正準形または，第2正準形となるように得る。

理論概要

◆ ε 一最小実現のアルゴリズム

Step 1 ： 正規化モデル$\bar{\Sigma}_s$ を得る。

$$\bar{A} = T A T^{-1},\quad \bar{B} = T B,\quad \bar{C} = C T^{-1},\quad \bar{D} = D$$

ただし $^tN = [\ ^tC,\ ^tA\ ^tC,\ \cdots\cdots,\ ^tA^{(n-1)}\ ^tC\ ]$

$N = U S\ ^tW$， S (full matrix)は Nの特異値分解

$T = S\ ^tW$

Step 2 ： $\bar{\Sigma}_s$ より ε 一最小実現モデルΣ_s^* を得る

$$A^* = {}^t\bar{T}\bar{A}\bar{T},\quad B^* = {}^t\bar{T}\bar{B},\quad C^* = \bar{C}\bar{T},\quad D^* = \bar{D}$$

ただし $\bar{V} = [\ \bar{B},\ \bar{A}\bar{B},\ \cdots\cdots,\ \bar{A}^{(n-1)}\bar{B}\ ]$

$\bar{V} = \bar{U}\bar{S}\ ^t\bar{W}$， S (full matrix)は Vの特異値分解

$\bar{U} = \{\ \bar{u}_1,\ \bar{u}_2,\ \cdots\cdots,\ \bar{u}_{nc},\ \cdots\cdots\ \}$ ， $\bar{T} = \{\ \bar{u}_1,\ \bar{u}_2, \cdots\cdots, \bar{u}_{nc}\}$

$\bar{S} = \mathrm{diag}\{\ \bar{\sigma}_1\ ,\ \bar{\sigma}_2\ , \cdots\cdots,\ \bar{\sigma}_{nc}\ ,\ \cdots\cdots\ \}$

$\bar{\sigma}_1 \geq \bar{\sigma}_2 \geq \cdots\cdots \geq \bar{\sigma}_{nc} > \varepsilon > \bar{\sigma}_{nc+1}$

Step 3 ：Σ_s^* に対して正準最小実現問題を解く。

→ Luenberger の第１正準形又は第２正準形を得る。

（ＣＭＲコマンド（ＤＰ．Ⅱ．２）参照）

実 行 例

〔1〕 状態空間表現形式システムからのε－最小実現モデルの取得例

```
/DPACS/ TIME-17:04:31 CPU-00:00:02
EMR EX1 EX1A
 SYSTEM EX1         HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0         (    3) ;SYSIN    /S /C /  /     (87/11/27) :EX1 TEST DATA
  2) EX1.S1         (    3) ;COOD     /1 /  /  /     (87/11/30) :COOD EX1
  3) EX1.S2         (    3) ;COOD     /1 /  /  /     (87/11/30) :COOD/1 EX1
  4) EX1.S3         (    3) ;COOD     /2 /  /  /     (87/11/30) :COOD/2 EX1
  5) EX1.S4         (    3) ;CMR      /M /C /2 /     (87/11/30) :EX1 CMR
 WHICH SYSTEMDATA IS USED ? :   1
 TO OBTAIN E-MINIMAL REALIZATION
 VALUES OF EPS
  1)    6.50968816D+01
  2)    1.51790723D+01
  3)    9.62446126D-01
 GIVE EPS : EPS= 1.0
 OBTAIN CANONICAL FORM ?  :    N
 GIVE COMMENT FOR SYSTEMDATA :
EMR EX1 EX1A
 /DPACS/ 19:40:19
 /DPACS/ 19:40:24
TYPE/S EX1A
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1A.S0       ( 2, 2, 2)

 --- ( 2, 2) MATRIX    A        ---
 ( 1)  7.87305661D-01 -3.52923611D+00
 ( 2) -8.17982548D-01 -7.21097142D+00
 FILENAME : _
 --- ( 2, 2) MATRIX    B        ---
 ( 1) -4.89806508D-01 -1.40360814D+00
 ( 2) -3.14995164D+00  1.34777670D+01
 FILENAME : _
 --- ( 2, 2) MATRIX    C        ---
 ( 1) -2.98509728D-01 -7.28713554D-01
 ( 2)  1.38777878D-17  1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX    D        ---
 ( 1)  0.0             0.0
 ( 2)  0.0             0.0
 FILENAME : _
 /DPACS/ TIME-17:47:10 CPU-00:00:20
```

〔2〕 状態空間表現形式システムの ε −最小実現モデルを Luenberger の第２正準形となる様に取得する例

```
/DPACS/ TIME-17:04:32 CPU-00:00:02
 EMR EX1 EX2
  TO OBTAIN E-MINIMAL REALIZATION
  VALUES OF EPS
   1)    6.50968816D+01
   2)    1.51790723D+01
   3)    9.62446126D-01
  GIVE EPS : EPS= 1D-5
  OBTAIN CANONICAL FORM ?  :   Y
  WHICH SUBSPACE IS REMOVED
  (1) UNCONTROLLABLE ONE OR (2) UNOBSERVABLE ONE ? :  2
  SPECIFY TYPE OF CANONICAL FORM (1 OR 2) :   2
  GIVE EPSILON, TO DETERMINE RANK OF MATRIX V@ EPS= 1D-10
  COLUMN INDEPENDENCY OF MATRIX V@ IS INVESTIGATED
   ( 1)
    1   1
   ( 2)
    1   0
   ( 3)
    0   0
  ***
  CONDITION NUMBERS ARE AS FOLLOWS
  --- ( 3, 2) MATRIX CNUM      ---
  ( 1)  1.00000000D+00  8.73330531D-01
  ( 2)  1.91442870D-01  0.0
  ( 3)  0.0             0.0
  CHANGE THE EPSILON ? _
  GIVE EPSILON, TO DETERMINE RANK OF MATRIX V@ EPS= 1D-12
  COLUMN INDEPENDENCY OF MATRIX V@ IS INVESTIGATED
   ( 1)
    1   1
   ( 2)
    1   0
   ( 3)
    0   0
  CONDITION NUMBERS ARE AS FOLLOWS
  --- ( 3, 2) MATRIX CNUM      ---
  ( 1)  1.00000000D+00  8.73330531D-01
  ( 2)  1.91442870D-01  0.0
  ( 3)  0.0             0.0
  CHANGE THE EPSILON ?   N
  END ? _
  GIVE COMMENT FOR SYSTEMDATA :
  ______
  /DPACS/ TIME-17:47:09 CPU-00:00:20
```

```
/DPACS/ TIME-17:04:33 CPU-00:00:02
TYPE/S EX2
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX2.SO        ( 2, 2, 3)

 --- ( 3, 3) MATRIX    A          ---
 ( 1)  0.0                3.68459400D+00  2.33874363D+01
 ( 2)  1.00000000D+00 -1.08669169D+01  0.0
 ( 3)  0.0                5.86747704D-01  2.54606018D+00
 FILENAME : _
 --- ( 3, 2) MATRIX    B          ---
 ( 1) -1.48102982D+00  2.67373255D+01
 ( 2)  4.13148048D+00 -1.44349410D+01
 ( 3) -1.97468060D-01  3.16211229D+00
 FILENAME : _
 --- ( 2, 3) MATRIX    C          ---
 ( 1)  0.0                1.00000000D+00  0.0
 ( 2)  0.0               -7.14630892D-01  1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX    D          ---
 ( 1)  0.0                0.0
 ***
 ( 2)  0.0                0.0
 FILENAME : _
/DPACS/ TIME-17:47:12 CPU-00:00:20
```

極配置問題の求解	D P. Ⅳ. 1
P L O C ([／X ／Y]) _S N	

機　能

状態空間表現形式システムΣ_Sに対して，状態フィードバックまたは出力フィードバックによる極配置問題を解く。

◆第1スイッチが／Xの場合は，状態フィードバックを行う。

◆第1スイッチが／Yの場合は，出力フィードバックを行う。

理論概要

$$\Sigma_s:\quad \frac{d}{dt}\boldsymbol{x}(t)=\boldsymbol{A}\boldsymbol{x}(t)+\boldsymbol{B}\boldsymbol{u}(t),\ \boldsymbol{y}(t)=\boldsymbol{C}\boldsymbol{x}(t)$$

$$\boldsymbol{x}(t)\in\mathbf{R}^n,\ \boldsymbol{u}(t)\in\mathbf{R}^m,\ \boldsymbol{y}(t)\in\mathbf{R}^p$$

に対し，$\boldsymbol{u}(t)=-\boldsymbol{F}\boldsymbol{x}(t)$ 又は，$\boldsymbol{u}(t)=-\boldsymbol{F}\boldsymbol{y}(t)=-\boldsymbol{F}\boldsymbol{C}\boldsymbol{x}(t)$ のフィードバックを施す時，閉ループ系の方程式として，

$$\begin{cases}\dfrac{d\boldsymbol{x}}{dt}=(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{F})\boldsymbol{x} & ：状態フィードバック\\[2mm] \dfrac{d\boldsymbol{x}}{dt}=(\boldsymbol{A}-\boldsymbol{B}\boldsymbol{F}\boldsymbol{C})\boldsymbol{x} & ；出力フィードバック\end{cases}$$

が得られ，$\boldsymbol{A}-\boldsymbol{B}\boldsymbol{F}$ 又は $\boldsymbol{A}-\boldsymbol{B}\boldsymbol{F}\boldsymbol{C}$ の固有値が指定の複素固有値 Λ に一致するように $\boldsymbol{F}$を定める問題である。

◆疋田のアルゴリズム（状態フィードバック）[11]

Step 1：以下に定義するベクトル列$\{\boldsymbol{v}_i\}_{i=1,\sim,n}$，$\boldsymbol{v}_i\in\mathbf{R}^n$ が1次独立になるようにベクトル列$\{\xi_i\}_{i=1,\sim,n}$，$\xi_i\in\mathbf{R}^m$ を選ぶ。

$$\begin{cases}\boldsymbol{v}_i\equiv\boldsymbol{V}_i\xi_i,\quad \boldsymbol{V}_i=(\boldsymbol{A}-\lambda_i\boldsymbol{I})^{-1}\cdot\boldsymbol{B}\\ \text{特に}\quad \lambda_i=\lambda_{iR}+j\lambda_{iI},\ \lambda_{i+1}=\lambda_{iR}-j\lambda_{iI}\end{cases}$$

従って
$$\begin{bmatrix}\boldsymbol{v}_i\\ \boldsymbol{v}_{i+1}\end{bmatrix}=\begin{bmatrix}\boldsymbol{V}_i & -\boldsymbol{V}_{i+1}\\ \boldsymbol{V}_{i+1} & \boldsymbol{V}_i\end{bmatrix}\begin{bmatrix}\xi_i\\ \xi_{i+1}\end{bmatrix}$$

$$\begin{cases}\boldsymbol{V}_i=\{(\boldsymbol{A}-\lambda_{iR}\boldsymbol{I})^2+\lambda_{iI}^2\boldsymbol{I})\}^{-1}(\boldsymbol{A}-\lambda_{iR}\boldsymbol{I})\boldsymbol{B}\\ \boldsymbol{V}_{i+1}=\{(\boldsymbol{A}-\lambda_{iR}\boldsymbol{I})^2+\lambda_{iI}^2\boldsymbol{I}\}^{-1}\lambda_{iI}\cdot\boldsymbol{B}\end{cases}$$

Step 2：次式により状態フィードバックゲイン$\boldsymbol{F}$を求める。

$$\boldsymbol{F}=[\xi_1,\cdots\cdots,\xi_n][\boldsymbol{v}_1,\cdots\cdots,\boldsymbol{v}_n]^{-1}$$

＜前提＞ 1° λ_i は $\boldsymbol{A}$ の固有と等しくない。

2° $\lambda_i\neq\lambda_j\quad(i\neq j)$

◆ Ackermann の公式（1入力系状態フィードバック）

$$\boldsymbol{F}=[\boldsymbol{0},\cdots\cdots,\boldsymbol{0},\boldsymbol{1}]\boldsymbol{V}^{-1}p(\lambda)$$

$$\boldsymbol{V}\equiv[\boldsymbol{B},\ \boldsymbol{A}\boldsymbol{B},\cdots\cdots,\boldsymbol{A}^{n-1}]\in\mathbf{R}^{n\times n}$$

$$p(\lambda)\equiv\prod_{i=1}^{n}(\lambda-\lambda_i)$$

(1) 1入力系の場合は，与えられたΛに対して $\boldsymbol{F}$ は一意に定まるが，多入力系の場合は与えられた固有値に対応する固有固有ベクトルの方向を定めるパラメータ$\{\xi_i\}$を指定しなければ解は一意的に求まらない。この自由度は応答波形の調整に利用できる。

(2) この問題に対しては，モデル観測方程式は無関係である。ただし状態ベクトルの値は，直接には利用できないので，オブザーバを用いる必要がある。

◆出力フィードバック則による極配置問題（木村のアルゴリズム）[12]

出力フィードバックは状態フィードバックの限られたクラスであるので任意の Λ に対してこの問題の解が存在するのは特殊な場合に限定されることが予想される。実際，$n<m+p$ が成立するとき，ほとんど任意の Λ に対して，この問題の解が存在することが知られている。以下においては，次の2ケースを考える。

Case 1：$n<p+m$ かつ $n\geq p$，Case 2：$n\geq p+m$ かつ $p\geq m$

このとき実軸に対称な複素数の集合

$$\Lambda_1=\{\lambda_{11}\cdots\cdots\lambda_{1r}\},\ \Lambda_2=\{\lambda_{21}\cdots\cdots\lambda_{2p}\},\quad r\equiv\begin{cases}n-p\ (\text{Case }1)\\ m-1\ (\text{Case }2)\end{cases}$$

Λ_1,Λ_2 に疋田のアルゴリズムの前提を採用する。このとき $\Lambda=\Lambda_1\cup\Lambda_2$ の極配置に対する出力フィードバック則による極配置問題の解法アルゴリズムを以下に示す。

(i) $r\neq 0$ の場合

Step 1： 以下に定義するベクトル列 $\{\boldsymbol{v}_{1i}\}\ i=1,\cdots\cdots,r$ $(\boldsymbol{v}_{1i}\in\mathbf{R}^n)$ が1次独立となるようにベクトル列 $\{\boldsymbol{\xi}_{1i}\}\ i=1,\cdots\cdots,r$ $(\boldsymbol{\xi}_{1i}\in\mathbf{R}^p)$ を選ぶ。

$$\begin{cases}\lambda_{1i}\text{ が実数のとき}\quad:\boldsymbol{v}_{1i}=\boldsymbol{V}_{1i}\boldsymbol{\xi}_{1i},\quad \boldsymbol{V}_{1i}=({}^t\boldsymbol{A}-\lambda_{1i}\boldsymbol{I})^{-1}\cdot{}^t\boldsymbol{C}\\ \lambda_{1i}\text{ が実数でないとき}:\lambda_{1i}=\lambda_{1iR}+j\lambda_{1iI},\quad \lambda_{1i+1}=\lambda_{1iR}-j\lambda_{1iI}\end{cases}$$

$$\begin{bmatrix}\boldsymbol{v}_{1i}\\ \boldsymbol{v}_{1i+1}\end{bmatrix}=\begin{bmatrix}\boldsymbol{V}_{1i} & -\boldsymbol{V}_{1i+1}\\ \boldsymbol{V}_{1i+1} & \boldsymbol{V}_{1i}\end{bmatrix}\begin{bmatrix}\boldsymbol{\xi}_{1i}\\ \boldsymbol{\xi}_{1i+1}\end{bmatrix}$$

$$\begin{cases}\boldsymbol{V}_{1i}=\{({}^t\boldsymbol{A}-\lambda_{1iR}\boldsymbol{I})^2+\lambda_{1iI}^2\boldsymbol{I}\}^{-1}(\boldsymbol{A}-\lambda_{1iR}\boldsymbol{I})^t\boldsymbol{C}\\ \boldsymbol{V}_{1i+1}=\{({}^t\boldsymbol{A}-\lambda_{1iR}\boldsymbol{I})^2+\lambda_{1iI}^2\boldsymbol{I}\}^{-1}\lambda_{1iI}{}^t\boldsymbol{C}\end{cases}$$

Step 2： 次式からベクトル列 $\{\boldsymbol{\xi}_{2i}\}_{i=1,\cdots,p}$ $(\boldsymbol{\xi}_{2i}\in\mathbf{R}^m)$ を求める。

$$\begin{cases}\lambda_{2i}\text{ が実数のとき}\quad:\boldsymbol{W}\boldsymbol{V}_{2i}\boldsymbol{\xi}_{2i}=\boldsymbol{0},\ \boldsymbol{V}_{2i}=(\boldsymbol{A}-\lambda_{2i}\boldsymbol{I})^{-1}\cdot\boldsymbol{B}\\ \lambda_{2i}\text{ が実数でないとき}:\lambda_{2i}=\lambda_{2iR}+j\lambda_{2iI},\quad \lambda_{2i+1}=\lambda_{2iR}-j\lambda_{2iI}\end{cases}$$

$$\begin{bmatrix}\boldsymbol{W}\boldsymbol{V}_{2i} & -\boldsymbol{W}\boldsymbol{V}_{2i+1}\\ \boldsymbol{W}\boldsymbol{V}_{2i+1} & \boldsymbol{W}\boldsymbol{V}_{2i}\end{bmatrix}\begin{bmatrix}\boldsymbol{\xi}_{2i}\\ \boldsymbol{\xi}_{2i+1}\end{bmatrix}=\boldsymbol{0}$$

$$\begin{cases}\boldsymbol{V}_{2i}=\{(\boldsymbol{A}-\lambda_{2iR}\boldsymbol{I})^2+\lambda_{2iI}\boldsymbol{I}\}^{-1}(\boldsymbol{A}-\lambda_{2iR}\boldsymbol{I})\boldsymbol{B}\\ \boldsymbol{V}_{2i+1}=\{(\boldsymbol{A}-\lambda_{2iR}\boldsymbol{I})^2+\lambda_{2iI}\boldsymbol{I}\}^{-1}\lambda_{2iI}\boldsymbol{B}\end{cases}$$

ただし　$\boldsymbol{W}={}^t({}^t\boldsymbol{v}_{11},\cdots\cdots,{}^t\boldsymbol{v}_{1r})\in\mathbf{R}^{r\times n}$

Step 3： 次式からベクトル列 $\{\boldsymbol{v}_{2i}\}_{i=1,\cdots,p}$ を求める。

$$\begin{cases}\lambda_{2i}\text{ が実数のとき}\quad:\ \boldsymbol{v}_{2i}=\boldsymbol{V}_{2i}\boldsymbol{\xi}_{2i}\\ \lambda_{2i}\text{ が実数でないとき}:\begin{bmatrix}\boldsymbol{v}_{2i}\\ \boldsymbol{v}_{2i+1}\end{bmatrix}=\begin{bmatrix}\boldsymbol{V}_{2i} & -\boldsymbol{V}_{2i}\\ \boldsymbol{V}_{2i+1} & \boldsymbol{V}_{2i+1}\end{bmatrix}\begin{bmatrix}\boldsymbol{\xi}_{2i}\\ \boldsymbol{\xi}_{2i+1}\end{bmatrix}\end{cases}$$

Step 4： 次式から出力フィードバックゲイン $\boldsymbol{F}$ を求める。

$$\boldsymbol{F}=[\boldsymbol{\xi}_{21},\cdots\cdots,\boldsymbol{\xi}_{2p}](\boldsymbol{C}\cdot[\boldsymbol{v}_{21},\cdots\cdots,\boldsymbol{v}_{2p}])^{-1}$$

(ii) $r = 0$のとき

Step 1：上記 Step 3の$\{\boldsymbol{v}_{2i}\}_{i=1,\cdots,p}$，($\boldsymbol{v}_{2i} \in \mathbf{R}^n$) が1次独立になるようにベクトル列$\{\boldsymbol{\xi}_{2i}\}_{i=1,\cdots,r}$ ($\boldsymbol{\xi}_{2i} \in \mathbf{R}^p$) を選ぶ。

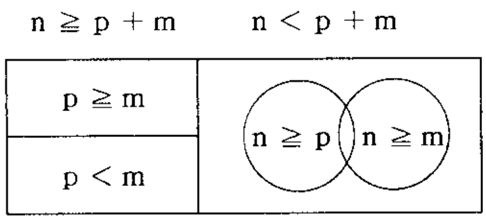
Step 2：上記 Step 4と同じ

(注) Step 4の逆行列が求まらない場合は，Step 1 または2へ戻るか、指定した極をわずかだけずらしてみる。

＜注意1＞ Case 1 とCase 2 以外の可能性は，

Case 1′：$n < p + m$ かつ$n < p$ かつ $n \geq m$，

Case 2′：$n \geq p + m$ かつ$p < m$ でありモデルの双対をとりそれぞれ Case 1と2に帰着させる。



＜注意2＞ 数式モデルΣ_sに対しては，その観測方程式を

$\bar{\boldsymbol{y}}(t) = \boldsymbol{y}(t) - \boldsymbol{D}\,\boldsymbol{u}(t) = \boldsymbol{C}\,\boldsymbol{x}(t)$ として出力フィードバック則

$\boldsymbol{u}(t) = -\bar{\boldsymbol{F}}\,\bar{\boldsymbol{y}}(t)$ を求め次式により $\boldsymbol{F}$ を求める。

$\boldsymbol{F} = (\boldsymbol{I} - \bar{\boldsymbol{F}}\,\boldsymbol{D})^{-1}\,\bar{\boldsymbol{F}}$

実 行 例

〔1〕 状態空間表現形式システムの極配置問題を状態フィードバック則により解く例

```
/DPACS/ TIME-17:04:34 CPU-00:00:02
PLOC/X EX2
 TO SOLVE POLE ASSIGNMENT PROBLEM
 GIVE POLES TO BE ASSIGNED
 INPUT -MODE OF ( 6, 2) MATRIX ROOT      : _
 ( 1) ROW
-1,0           ①
 ( 2) ROW
-2,3
 ( 3) ROW
-2,-3
 ( 4) ROW
-5,0
 ( 5) ROW
-4,10
 ( 6) ROW
-4,-10
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3, 6) MATRIX  F0       : _
 ( 1) ROW
1,1,2,1,3,2
 ( 2) ROW
-2,3,1,1,-1,1
 ( 3) ROW
2,1,3,2,-1,1
 TYPE OR REVISE ? _
 THE SOLUTION OF POLE ASSIGNMENT PROBLEM FOR SYSTEM EX2.SO
 1) SPECIFIED POLES
 ---------------------------------------------------------------
  NO.      REAL                IMAGINARY            ABSOLUTE
 ---------------------------------------------------------------
 ( 1)  -1.00000000D+00      0.0                  1.00000000D+00
 ( 2)  -2.00000000D+00      3.00000000D+00       3.60555128D+00
 ( 3)  -2.00000000D+00     -3.00000000D+00       3.60555128D+00
 ( 4)  -5.00000000D+00      0.0                  5.00000000D+00
 ( 5)  -4.00000000D+00      1.00000000D+01       1.07703296D+01
 ( 6)  -4.00000000D+00     -1.00000000D+01       1.07703296D+01
 ---------------------------------------------------------------
 2) STATE FEEDBACK GAIN
 --- ( 3, 6) MATRIX   F      ---
 ( 1) -2.98318737D+02 -6.77554831D+01  3.23362972D+02 -1.93759689D+01
 ( 2)  4.74384410D+01  1.97472430D+01 -4.20614493D+01 -1.92116650D+00
 ( 3)  2.11642559D+01  1.01158081D+01 -1.33174061D+01 -4.16286389D+00
 ( 1)  1.43554123D+02 -3.46224567D+01
 ( 2) -1.26675748D+01  6.94992257D+00
 ( 3)  5.61860919D+00  6.34124984D+00
 FILENAME :   FX
 /DPACS/ TIME-17:47:13 CPU-00:00:20
```

【説明】

① 複素数として入力する。この例では，$-1+0\cdot j\ (j=\sqrt{-1})$

```
/DPACS/ TIME-17:04:36 CPU-00:00:02
CLPS/X EX2 EX2C
 TO OBTAIN CLOSED LOOP SYSTEM
 BY STATE FEEDBACK (U=-FX+GV)
 INPUT -MODE OF ( 3, 6) MATRIX   F      : F
 FILENAME :  FX
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3, 3) MATRIX   G      : I
 TYPE OR REVISE ? _
 GIVE COMMENT FOR SYSTEMDATA :
CLPS/X
 /DPACS/ 20: 3: 5

POLE EX2C
 TO CALCULATE SYSTEM'S POLES
 POLES OF CONTINUOUS SYSTEM EX2C.SO

 ------------------------------------------------------------
  NO.      REAL                 IMAGINARY             ABSOLUTE
 ------------------------------------------------------------
 ( 1)  -1.00000000D+00       0.0                  1.00000000D+00
 ( 2)  -2.00000000D+00       3.00000000D+00       3.60555128D+00
 ( 3)  -2.00000000D+00      -3.00000000D+00       3.60555128D+00
 ( 4)  -4.00000000D+00       1.00000000D+01       1.07703296D+01
 ( 5)  -4.00000000D+00      -1.00000000D+01       1.07703296D+01
 ( 6)  -5.00000000D+00       0.0                  5.00000000D+00
 ------------------------------------------------------------
 FILENAME : _
 CONTINUOUS SYSTEM EX2C.SO      IS STABLE
 /DPACS/ TIME-17:47:15 CPU-00:00:20
```

〔2〕 状態空間表現形式システム・モデルの極配置問題を出力フィードバック則により解く例

```
 /DPACS/ TIME-17:04:37 CPU-00:00:02
PLOC/Y EX4
 TO SOLVE POLE ASSIGNMENT PROBLEM
 GIVE  2 &  3-POLES TO BE ASSIGND
 INPUT -MODE OF ( 2, 2) MATRIX ROOT      : _
 ( 1) ROW
-1,2
 ( 2) ROW
-1,-2
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3, 2) MATRIX ROOT      : _
 ( 1) ROW
-2,0
 ( 2) ROW
-3,5
 ( 3) ROW
-3,-5
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 3) MATRIX  FO       : _
 ( 1) ROW
1,2,-1
 ( 2) ROW
3,1,2
 TYPE OR REVISE ? _
 --- ( 3, 2) MATRIX  W'V      ---
 ( 1) -7.40726924D-16  7.71951947D-17
 ( 2)  1.41976274D-16  1.75098651D-17
 ( 3) -8.32667268D-17  2.42861287D-17
 THE SOLUTION OF POLE ASSIGNMENT PROBLEM FOR SYSTEM EX4.SO
 1) SPECIFIED POLES
 -------------------------------------------------------------
  NO.      REAL                 IMAGINARY          ABSOLUTE
 -------------------------------------------------------------
 ( 1)  -1.00000000D+00     2.00000000D+00     2.23606798D+00
 ( 2)  -1.00000000D+00    -2.00000000D+00     2.23606798D+00
 ( 3)  -2.00000000D+00     0.0                2.00000000D+00
 -------------------------------------------------------------
 2) OUTPUT FEEDBACK GAIN
 --- ( 4, 2) MATRIX   F     ---
 ( 1)  2.38820282D+02  2.01933053D+02
 ( 2)  1.07823306D+01  1.74532163D+01
 ( 3) -9.74758622D+01 -8.12718514D+01
 ( 4)  1.93610026D+01  1.45890752D+01
 FILENAME :   FY
 /DPACS/ TIME-17:47:37 CPU-00:00:20
```

```
/DPACS/ TIME-17:04:38 CPU-00:00:02
CLPS/Y EX4 EX4C
 TO OBTAIN CLOSED LOOP SYSTEM
 BY OUTPUT FEEDBACK (U=-FY+GV)
 INPUT -MODE OF ( 4, 2) MATRIX   F      : F
 FILENAME :  FY
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 4, 4) MATRIX   G      : I
 TYPE OR REVISE ? _
 GIVE COMMENT FOR SYSTEMDATA :
CLPS/Y
 /DPACS/ 20:35: 1
POLE EX4C
 TO CALCULATE SYSTEM'S POLES
 POLES OF CONTINUOUS SYSTEM EX4C.SO
```

NO.	REAL	IMAGINARY	ABSOLUTE
(1)	-1.000000000D+00	2.000000000D+00	2.23606798D+00
(2)	-1.000000000D+00	-2.000000000D+00	2.23606798D+00
(3)	-2.000000000D+00	0.0	2.00000000D+00
(4)	-3.000000000D+00	5.000000000D+00	5.83095189D+00
(5)	-3.000000000D+00	-5.000000000D+00	5.83095189D+00

```
 FILENAME : _
 CONTINUOUS SYSTEM EX4C.SO      IS STABLE
 /DPACS/ TIME-17:47:38 CPU-00:00:21
```

最適制御問題の求解	DP. Ⅳ. 2
OPT [# / /S [# / /I]] ␣SN	

機　能

状態空間表現形式システムΣ_sに対し，2次形式評価関数を用いて，最適ゲインを求める。

◆第1スイッチを／Sとしたときは，連続系モデル（SN）をサンプル周期T，遅れ時間△tで制御する場合について計算する。

◆第2スイッチを／Iとしたときは，評価関数が積分形式で示されたものを使用することを表す。

理論概要

2次形式評価関数

$$J = \begin{cases} \displaystyle\int_0^{\infty} \{ {}^t\boldsymbol{z}(t)\boldsymbol{Q}\,\boldsymbol{z}(t) + 2\,{}^t\boldsymbol{z}(t)\boldsymbol{S}\,\boldsymbol{u}(t) + {}^t\boldsymbol{u}(t)\boldsymbol{R}\,\boldsymbol{u}(t) \}\,dt \\ \qquad\qquad\qquad\qquad\qquad\qquad \cdots\cdots\cdots\cdots\text{積分型評価関数} \\ \displaystyle\sum_{k=0}^{\infty} \{ {}^t\boldsymbol{z}(k)\boldsymbol{Q}\,\boldsymbol{z}(k) + 2\,{}^t\boldsymbol{z}(k)\boldsymbol{S}\,\boldsymbol{u}(k) + {}^t\boldsymbol{u}(k)\boldsymbol{R}\,\boldsymbol{u}(k) \}\,dt \\ \qquad\qquad\qquad\qquad\qquad\qquad \cdots\cdots\cdots\cdots\text{離散型評価関数} \end{cases}$$

（ただし $\boldsymbol{R} > 0$，$\boldsymbol{Q} - \boldsymbol{S}\boldsymbol{R}^{-1}\,{}^t\boldsymbol{S} > 0$）

を最小にするように，システム

$$\begin{cases} \sigma\,\boldsymbol{x}(t) = \boldsymbol{A}\,\boldsymbol{x}(t) + \boldsymbol{B}\,\boldsymbol{u}(t) \\ \boldsymbol{z}(t) = \boldsymbol{H}\,\boldsymbol{x}(t) \end{cases}$$

の制御則を求めるのが最適制御問題である。制御則は，システムが可制御かつ可観測のとき次式で求められる。

$$\boldsymbol{u}^{\#}(t) = -\boldsymbol{F}^{\#}\boldsymbol{x}(t)$$

$$\boldsymbol{F}^{\#} = \begin{cases} \boldsymbol{R}^{-1}\,{}^t\boldsymbol{B}\boldsymbol{P} + \boldsymbol{R}^{-1}\,{}^t\boldsymbol{S}\boldsymbol{H} & \text{（積分型評価関数連続時間システムの場合）} \\ (\boldsymbol{R} + {}^t\boldsymbol{B}\boldsymbol{P}\boldsymbol{B})^{-1}\,{}^t\boldsymbol{B}\boldsymbol{P}\boldsymbol{A} + \boldsymbol{R}^{-1}\,{}^t\boldsymbol{S}\boldsymbol{H} & \text{（離散型評価関数離散時間システムの場合）} \end{cases}$$

ここで $\boldsymbol{P} \in \mathbf{R}^{n\times n}$ は次の定常 Riccati 方程式の解である。なお，$\boldsymbol{P}$ は一意な正定（非負定）対称解である。

$$\begin{cases} \boldsymbol{P}\tilde{\boldsymbol{A}} + {}^t\tilde{\boldsymbol{A}}\boldsymbol{P} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\,{}^t\boldsymbol{B}\boldsymbol{P} + {}^t\boldsymbol{H}\tilde{\boldsymbol{Q}}\boldsymbol{H} = \boldsymbol{0} & \text{（積分型評価関数連続時間システム）} \\ \boldsymbol{P} = {}^t\tilde{\boldsymbol{A}}\boldsymbol{P}\tilde{\boldsymbol{A}} - {}^t\tilde{\boldsymbol{A}}\boldsymbol{P}\boldsymbol{B}\,(\boldsymbol{R} + {}^t\boldsymbol{B}\boldsymbol{P}\boldsymbol{B})^{-1}\,{}^t\boldsymbol{B}\boldsymbol{P}\tilde{\boldsymbol{A}} + {}^t\boldsymbol{H}\tilde{\boldsymbol{Q}}\boldsymbol{H} & \\ & \text{（離散型評価関数離散時間システム）} \\ \tilde{\boldsymbol{A}} = \boldsymbol{A} - \boldsymbol{B}\boldsymbol{R}^{-1}\,{}^t\boldsymbol{S}\boldsymbol{H},\ \tilde{\boldsymbol{Q}} = \boldsymbol{Q} - \boldsymbol{S}\boldsymbol{R}^{-1}\,{}^t\boldsymbol{S} & \end{cases}$$

J の最小値は，${}^t\boldsymbol{x}(0)\boldsymbol{P}\,\boldsymbol{x}(0)$ である。

◆第1スイッチを指定しない場合（この時，第2スイッチも無効となる）には，最も基本的な形となり，指定したシステムの表現形式に従った評価関数を用いて最適制御則を計算する。

◆第1スイッチが／Sの場合には，指定した制御対象システムは連続時間システムでなければならない。また入力に遅れがない場合とある場合の両方のアルゴリズムがあるので以下にそれぞれ述べる。

(a) 遅れのない場合

連続時間表現形式システムを

$$\begin{cases} \dfrac{d}{dt}\ \boldsymbol{x}(t) = \boldsymbol{A}_C \boldsymbol{x}(t) + \boldsymbol{B}_C \boldsymbol{u}(t) \\ \qquad \boldsymbol{z}(t) = \boldsymbol{H}\boldsymbol{x}(t) \end{cases}$$

をサンプル周期；T (sec)，対象の入力に0次ホールドを前置して離散化する。

すると離散時間システム

$$\begin{cases} \boldsymbol{x}(k+s) = \boldsymbol{A}\ \boldsymbol{x}(k) + \boldsymbol{B}\ \boldsymbol{u}(k) \\ \boldsymbol{z}(k) \qquad\quad = \boldsymbol{H}\ \boldsymbol{x}(k) \end{cases}$$

ただし，$\boldsymbol{A} = \exp(\boldsymbol{A}_C T)$, $\boldsymbol{B} = \int_0^T \exp(\boldsymbol{A}_C \tau)\boldsymbol{B}_C d\tau$，また，$(\boldsymbol{A}_C, \boldsymbol{B}_C)$が可制御であっても$(\boldsymbol{A}, \boldsymbol{B})$が可制御でなくなることがあり得るが，可制御を仮定する。

評価関数が離散型の場合には，本理論概要の冒頭に書いた離散型評価関数離散時間システムの場合に対応している。（ただし，評価関数中の $\boldsymbol{S} = \boldsymbol{0}$ のアルゴリズムのみ有り）

評価関数が積分型の場合には，離散型に変換する。（ただし，評価関数中の $\boldsymbol{S} = \boldsymbol{0}$ の場合のアルゴリズムのみ有り）即ち

$$J = \int_0^\infty \{{}^t\boldsymbol{z}(t)\boldsymbol{Q}\boldsymbol{z}(t) + {}^t\boldsymbol{u}(t)\boldsymbol{R}\boldsymbol{u}(t)\}\,dt$$

$$\Longrightarrow J = \sum_{i=0}^{\infty} [{}^t\boldsymbol{x}(i)\widehat{\boldsymbol{Q}}\boldsymbol{x}(i) + 2\,{}^t\boldsymbol{x}(i)\widehat{\boldsymbol{S}}\boldsymbol{u}(i) + {}^t\boldsymbol{u}(i)\widehat{\boldsymbol{R}}\boldsymbol{u}(i)]$$

ただし

$$\begin{bmatrix} \widehat{\boldsymbol{Q}} & \widehat{\boldsymbol{S}} \\ {}^t\widehat{\boldsymbol{S}} & \widehat{\boldsymbol{R}} \end{bmatrix} = \int_0^T \begin{bmatrix} {}^t\boldsymbol{\Phi}(s) & \boldsymbol{0} \\ {}^t\boldsymbol{\Psi}(s) & \boldsymbol{I}_m \end{bmatrix} \begin{bmatrix} {}^t\boldsymbol{H}\boldsymbol{Q}\boldsymbol{H} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{R} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Phi}(s) & \boldsymbol{\Psi}(s) \\ \boldsymbol{0} & \boldsymbol{I}_m \end{bmatrix} ds$$

$$\boldsymbol{\Phi}(s) = \exp(\boldsymbol{A}_C s)\ ,\quad \boldsymbol{\Psi}(s) = \int_0^s \exp(\boldsymbol{A}_C \tau)\cdot \mathbf{B}_C\, d\tau$$

Taylor 展開すると，

$$\begin{bmatrix} \widehat{\boldsymbol{Q}} & \widehat{\boldsymbol{S}} \\ {}^t\widehat{\boldsymbol{S}} & \widehat{\boldsymbol{R}} \end{bmatrix} = \boldsymbol{W}_1 T + \boldsymbol{W}_2 T^2 + \boldsymbol{W}_3 T^3 + \cdots\cdots$$

$$\boldsymbol{W}_1 = \begin{bmatrix} {}^t\boldsymbol{H}\boldsymbol{Q}\boldsymbol{H} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{R} \end{bmatrix}$$

$$\boldsymbol{W}_{k+1} = \frac{1}{k+1}\left(\begin{bmatrix} {}^t\boldsymbol{A} & \boldsymbol{0} \\ {}^t\boldsymbol{B} & \boldsymbol{0} \end{bmatrix} \boldsymbol{W}_k + \boldsymbol{W}_k \begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \right),\ k = 1, 2, \cdots\cdots$$

となり，この式で計算して $\widehat{\boldsymbol{Q}}$, $\widehat{\boldsymbol{S}}$, $\widehat{\boldsymbol{R}}$ を求める。これ以降は，離散型評価関数離散時間システムの場合と同様である。

(b) 遅れのある場合

前項と同じ連続時間システムに対し，サンプルした時点から△t（$0 \leq \triangle t < T$）だけ時間遅れがあって制御出力が出るとすると，離散化した状態方程式は，

$$\begin{cases} \boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}_1\boldsymbol{u}(k) + \boldsymbol{B}_2\boldsymbol{u}(k-1) \\ \boldsymbol{z}(k) \qquad\quad = \boldsymbol{H}\boldsymbol{x}(k) \end{cases}$$

これを書き換えれば，

$$\begin{cases} \begin{bmatrix} \boldsymbol{x}(k+1) \\ \boldsymbol{u}(k) \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(k) \\ \boldsymbol{u}(k-1) \end{bmatrix} + \begin{bmatrix} \boldsymbol{B} \\ \boldsymbol{I} \end{bmatrix} \boldsymbol{u}(k) \\ \qquad \boldsymbol{z}(k) \quad = [\,\boldsymbol{H} \;\; \boldsymbol{0}\,] \begin{bmatrix} \boldsymbol{x}(k) \\ \boldsymbol{u}(k-1) \end{bmatrix} \end{cases}$$

ただし，$\boldsymbol{A} = \exp(\boldsymbol{A}_C T)$，$\boldsymbol{B}_1 = \int_0^{T-\triangle t} \exp(\boldsymbol{A}_C \tau)\boldsymbol{B}_C d\tau$，

$$\boldsymbol{B}_2 = \int_{T-\triangle t}^{T} \exp(\boldsymbol{A}_C \tau)\boldsymbol{B}_C d\tau, \quad \boldsymbol{B} = {}^t[\,\boldsymbol{B}_1, \boldsymbol{B}_2\,]$$

評価関数が離散型の場合には，離散型評価関数離散時間システムの扱いとなる。（ただし $\boldsymbol{S} = \boldsymbol{0}$ のアルゴリズムのみ有り）

評価関数が積分型の場合には，離散型に変換する。（ただし $\boldsymbol{S} = \boldsymbol{0}$ のアルゴリズムのみ有り）即ち，

$$J = \int_0^\infty \{{}^t\boldsymbol{z}(t)\boldsymbol{Q}\boldsymbol{z}(t) + {}^t\boldsymbol{u}(t)\boldsymbol{R}\boldsymbol{u}(t)\}\,dt$$

$$\begin{aligned} => J &= \sum_{i=0}^{\infty} \{{}^t\boldsymbol{x}(i)\boldsymbol{Q}_{11}\boldsymbol{x}(i) + 2\,{}^t\boldsymbol{x}(i)\boldsymbol{Q}_{12}\boldsymbol{u}(i-1) + {}^t\boldsymbol{u}(i-1)\boldsymbol{Q}_{22}\boldsymbol{u}(i-1) \\ &\quad + 2\,{}^t\boldsymbol{x}(i)\boldsymbol{S}_1\boldsymbol{u}(i) + 2\,{}^t\boldsymbol{u}(i-1)\boldsymbol{S}_2\boldsymbol{u}(i) + {}^t\boldsymbol{u}(i)\widehat{\boldsymbol{R}}\boldsymbol{u}(i)\} \\ &= \sum_{i=0}^{\infty} \Bigg\{ [\,{}^t\boldsymbol{x}(i)\,{}^t\boldsymbol{u}(i-1)\,] \begin{bmatrix} \boldsymbol{Q}_{11} & \boldsymbol{Q}_{12} \\ {}^t\boldsymbol{Q}_{12} & \boldsymbol{Q}_{22} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(i) \\ \boldsymbol{u}(i-1) \end{bmatrix} \\ &\quad + 2\,[\,{}^t\boldsymbol{x}(i)\boldsymbol{u}(i-1)\,] \begin{bmatrix} \boldsymbol{S}_1 \\ \boldsymbol{S}_2 \end{bmatrix} \boldsymbol{u}(i) + {}^t\boldsymbol{u}(i)\widehat{\boldsymbol{R}}\boldsymbol{u}(i) \Bigg\} \end{aligned}$$

ただし　$\boldsymbol{Q}_{11} = {}^t\boldsymbol{\Phi}(\triangle t)\,\widehat{\boldsymbol{Q}}\,\boldsymbol{\Phi}(\triangle t)$，$\boldsymbol{Q}_{12} = {}^t\boldsymbol{\Phi}(\triangle t)\,\widehat{\boldsymbol{Q}}\,\boldsymbol{\Psi}(\triangle t)$

$\boldsymbol{Q}_{22} = {}^t\boldsymbol{\Psi}(\triangle t)\,\widehat{\boldsymbol{Q}}\,\boldsymbol{\Psi}(\triangle t)$，$\boldsymbol{S}_1 = {}^t\boldsymbol{\Phi}(\triangle t)\,\widehat{\boldsymbol{S}}$，$\boldsymbol{S}_2 = {}^t\boldsymbol{\Psi}(\triangle t)\,\widehat{\boldsymbol{S}}$

$\widehat{\boldsymbol{Q}}$，$\widehat{\boldsymbol{S}}$，$\widehat{\boldsymbol{R}}$，$\boldsymbol{\Phi}(\triangle t)$，$\boldsymbol{\Psi}(\triangle t)$は(a)と同様である。このような拡大系で離散型評価関数離散時間システムの問題を解いて，制御則　$\boldsymbol{u}^{\#}(k) = -\boldsymbol{F}^{\#}\begin{bmatrix} \boldsymbol{x}(k) \\ \boldsymbol{u}(k-1) \end{bmatrix}$ を得る。

実 行 例

〔1〕 連続系状態空間表現形式システムに対し2次形式評価関数を用いて最適ゲインを求める例

```
/DPACS/ TIME-17:04:52 CPU-00:00:02
OPT EX1
 SYSTEM EX1       HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0       (    3) ;SYSIN    /S /C /  /    (87/11/27) :EX1 TEST DATA
  2) EX1.S1       (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD EX1
  3) EX1.S2       (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD/1 EX1
  4) EX1.S3       (    3) ;COOD     /2 /  /  /    (87/11/30) :COOD/2 EX1
  5) EX1.S4       (    3) ;CMR      /M /C /2 /    (87/11/30) :EX1 CMR
 WHICH SYSTEMDATA IS USED ? :   1
 TO SOLVE OPTMAL REGULATOR PROBLEM
 SPECIFY CONTROL VARIABLES ?   N  ①
 SYSTEM'S OUTPUTS ARE REGARDED AS CONTROL VARIABLES
 GIVE QUADRATIC CRITERION FUNCTION
 INPUT -MODE OF ( 2, 2) MATRIX   Q       : I
 TYPE OR REVISE ? __
 INPUT -MODE OF ( 2, 2) MATRIX   R       : I
 TYPE OR REVISE ?
 CROSS TERM IN CRITERION FUNCTION ? __
 THE SOLUTION OF OPTIMAL REGULATOR PROBLEM FOR SYSTEM EXSO  EXCO
 1) COEFFICIENT MATRIX OF CONTROL VARIABLES
 --- ( 2, 3) MATRIX   H       ---
 ( 1)  1.00000000D+00  0.0             0.0
 ( 2)  0.0             1.00000000D+00  1.00000000D+00
 FILENAME : __
 2) SPECIFIED CRITERION FUNCTION
 --- ( 2, 2) MATRIX   Q       ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  0.0             1.00000000D+00
 FILENAME : __
 --- ( 2, 2) MATRIX   R       ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  0.0             1.00000000D+00
 FILENAME : __
 OUTPUT THE SOLUTION OF RICCATI EQUATION ? __
 3) SOLUTION OF RICCATI EQUATION
 --- ( 3, 3) MATRIX   P       ---
 ( 1)  0.0             0.0             0.0
 ( 2)  0.0             0.0             0.0
 ( 3)  0.0             0.0             0.0
 FILENAME : __
 4) OPTIMAL CONTROL LAW
 --- ( 2, 3) MATRIX   F       ---
 ( 1)  1.33502145D+01  4.32588509D+00  8.07265264D+00
 ( 2) -1.32143208D+00 -5.79117535D-01 -1.88907257D-01
 FILENAME : __
 5) POLES OF CLOSED LOOP
 --------------------------------------------------------------
  NO.     REAL                 IMAGINARY             ABSOLUTE
 --------------------------------------------------------------
 ( 1)  -1.40205713D+00      1.43824560D+00       2.00856033D+00
 ( 2)  -1.40205713D+00     -1.43824560D+00       2.00856033D+00
 ( 3)  -2.49109863D+00      0.0                  2.49109863D+00
 --------------------------------------------------------------
 FILENAME : __
/DPACS/ TIME-17:47:02 CPU-00:00:20
```

【説明】

① 評価すべき状態変数を指定する場合にはNを入力して $\boldsymbol{H}$ 行列を指定する。そうでない場合には，Yを入力するかリターン・キーのみを押下すると $\boldsymbol{H} = \boldsymbol{C}$ として先に進む。

〔2〕 離散系状態空間表現形式システムに対し2次形式評価関数を用いて最適ゲインを求める例(2)

```
/OPACS/ TIME-17:04:53 CPU-00:00:02
OPT EX1D
 TO SOLVE OPTMAL REGULATOR PROBLEM
 SPECIFY CONTROL VARIABLES ?   N
 SYSTEM'S OUTPUTS ARE REGARDED AS CONTROL VARIABLES
 GIVE QUADRATIC CRITERION FUNCTION
 INPUT -MODE OF ( 2, 2) MATRIX   Q        : I
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX   R        : I
 TYPE OR REVISE ? _
 CROSS TERM IN CRITERION FUNCTION ?   Y
 INPUT -MODE OF ( 2, 2) MATRIX   S        : _
 ( 1) ROW
0.5,0
 ( 2) ROW
0,0.5
 TYPE OR REVISE ? _
 THE SOLUTION OF OPTIMAL REGULATOR PROBLEM FOR SYSTEM EX.S  EX.C
 1) COEFFICIENT MATRIX OF CONTROL VARIABLES
 --- ( 2, 3) MATRIX   H      ---
 ( 1)  1.00000000D+00  0.0             0.0
 ***
 ( 2)  0.0             1.00000000D+00  1.00000000D+00
 FILENAME : _
 2) SPECIFIED CRITERION FUNCTION
 --- ( 2, 2) MATRIX   Q      ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  0.0             1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX   R      ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  0.0             1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX   S      ---
 ( 1)  5.00000000D-01  0.0
 ( 2)  0.0             5.00000000D-01
 FILENAME : _
 OUTPUT THE SOLUTION OF RICCATI EQUATION ? _
 3) SOLUTION OF RICCATI EQUATION
 --- ( 3, 3) MATRIX   P      ---
 ( 1)  0.0             0.0             0.0
 ( 2)  0.0             0.0             0.0
 ( 3)  0.0             0.0             0.0
 FILENAME : _
 4) OPTIMAL CONTROL LAW
 ***
```

```
--- ( 2, 3) MATRIX   F      ---
( 1)  1.35667289D+02  2.15638349D+01  4.46540906D+01
( 2) -1.38304870D+02  1.02733008D+00  2.26028944D+01
FILENAME : _
5) POLES OF CLOSED LOOP
---------------------------------------------------------------------
 NO.      REAL                  IMAGINARY               ABSOLUTE
---------------------------------------------------------------------
( 1)   -9.13142744D-01        1.78263557D-01         9.30380334D-01
( 2)   -9.13142744D-01       -1.78263557D-01         9.30380334D-01
( 3)   -1.22859077D+00        0.0                    1.22859077D+00
---------------------------------------------------------------------
FILENAME : _
/DPACS/ TIME-17:47:40 CPU-00:00:21
```

〔3〕 連続系状態空間表現形式システムを遅れなしで離散系化し，離散型2次形式評価関数を用いて最適ゲインを求める例

```
/DPACS/ TIME-17:04:55 CPU-00:00:02
OPT/S EX1
 TO SOLVE OPTMAL REGULATOR PROBLEM
 GIVE SAMPLING INTERVAL
 DELT =  0.01
 CONTROL DELAY ?   N
 SPECIFY CONTROL VARIABLES ?   N
 SYSTEM'S OUTPUTS ARE REGARDED AS CONTROL VARIABLES
 GIVE QUADRATIC CRITERION FUNCTION
 INPUT -MODE OF ( 2, 2) MATRIX   Q       : I
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX   R       : I
 TYPE OR REVISE ? _
 CROSS TERM IN CRITERION FUNCTION ?   N
 SOLUTION OF OPTIMAL SAMPLED DATA REGULATOR PROBLEM FOR SYSTEM  EX1.SO
 SAMPLING INTERVAL =     0.010000
 1) COEFFICIENT MATRIX OF CONTROL MATRIX
 --- ( 2, 3) MATRIX   H      ---
 ( 1)  1.00000000D+00  0.0             0.0
 ( 2)  0.0             1.00000000D+00  1.00000000D+00
 ***
 FILENAME : _
 2) SPECIFIED CRITERION FUNCTION
 --- ( 2, 2) MATRIX   Q      ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  0.0             1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX   R      ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  0.0             1.00000000D+00
 FILENAME : _
 OUTPUT DISCRETIZED SYSTEM ? _
 3) DISCRETIZED SYSTEM
 --- ( 3, 3) MATRIX   A      ---
 ( 1)  1.00005083D+00  9.95033458D-03  1.00001667D-04
 ( 2)  1.02503396D-02  9.90100500D-01  2.00006708D-02
 ( 3)  3.01510087D-02  1.50002500D-04  1.01005117D+00
 FILENAME : _
 --- ( 3, 2) MATRIX   B      ---
 ( 1)  4.97516750D-05 -9.90033389D-05
 ( 2)  9.95050250D-03 -1.98010016D-02
 ( 3)  7.50016667D-07  1.00487558D-02
 FILENAME : _
 OUTPUT THE SOLUTION OF RICCATI EQUATION ? _
```

```
5) THE SOLUTION OF RICCATI EQUATION
--- ( 3, 3) MATRIX   P       ---
( 1)  0.0              0.0              0.0
( 2)  0.0              0.0              0.0
( 3)  0.0              0.0              0.0
FILENAME : _
6) OPTIMAL CONTROL LAW
--- ( 2, 3) MATRIX   F       ---
( 1)  1.32014933D+01  4.27616549D+00  7.98182692D+00
( 2) -1.28242551D+00 -5.67591647D-01 -1.71563714D-01
FILENAME : _
7) POLES OF CLOSED LOOP
-------------------------------------------------------------
 NO.      REAL                IMAGINARY             ABSOLUTE
-------------------------------------------------------------
( 1)    9.859756820-01      1.41816223D-02       9.86077666D-01
( 2)    9.859756820-01     -1.41816223D-02       9.86077666D-01
( 3)    9.75396515D-01      0.0                  9.75396515D-01
-------------------------------------------------------------
FILENAME : _
/DPACS/ TIME-17:47:42 CPU-00:00:21
```

〔4〕 連続系状態空間表現形式システムを遅れなしで離散系化し，積分型2次式評価関数を用いて最適ゲインを求める例

```
/DPACS/ TIME-17:04:56 CPU-00:00:02
OPT/S/I EX1
TO SOLVE OPTMAL REGULATOR PROBLEM
GIVE SAMPLING INTERVAL
DELT =  0.1
CONTROL DELAY ?   N ①
SPECIFY CONTROL VARIABLES ?   N
SYSTEM'S OUTPUTS ARE REGARDED AS CONTROL VARIABLES
GIVE QUADRATIC CRITERION FUNCTION
INPUT -MODE OF ( 2, 2) MATRIX   Q       : I
TYPE OR REVISE ? _
INPUT -MODE OF ( 2, 2) MATRIX   R       : I
TYPE OR REVISE ? _
SOLUTION OF OPTIMAL SAMPLED DATA REGULATOR PROBLEM FOR SYSTEM  EX1.SO
SAMPLING INTERVAL =    0.100000
1) COEFFICIENT MATRIX OF CONTROL MATRIX
--- ( 2, 3) MATRIX   H       ---
( 1)  1.00000000D+00  0.0             0.0
( 2)  0.0             1.00000000D+00  1.00000000D+00
FILENAME : _
2) SPECIFIED CRITERION FUNCTION
--- ( 2, 2) MATRIX   Q       ---
( 1)  1.00000000D+00  0.0
( 2)  0.0             1.00000000D+00
FILENAME : _
--- ( 2, 2) MATRIX   R       ---
( 1)  1.00000000D+00  0.0
( 2)  0.0             1.00000000D+00          ②
FILENAME : _
OUTPUT DISCRETIZED SYSTEM ?  N
OUTPUT DISCRETIZED CRITERION FUNCTION ?  N
OUTPUT THE SOLUTION OF RICCATI EQUATION ? N
```

```
6) OPTIMAL CONTROL LAW
--- ( 2, 3) MATRIX   F        ---
( 1)  1.19674942D+01  3.86484378D+00  7.22781710D+00
( 2) -9.65643692D-01 -4.74671618D-01 -3.61442386D-02
FILENAME : __
7) POLES OF CLOSED LOOP
------------------------------------------------------------------------------
 NO.      REAL                     IMAGINARY                 ABSOLUTE
------------------------------------------------------------------------------
( 1)    8.60412253D-01           1.24464106D-01            8.69367908D-01
***
( 2)    8.60412253D-01          -1.24464106D-01            8.69367908D-01
( 3)    7.79142508D-01           0.0                       7.79142508D-01
------------------------------------------------------------------------------
FILENAME : __
/DPACS/ TIME-17:47:41 CPU-00:00:21
```

【説明】

① 遅れを指定する場合にはＹを入力し，そうでない場合にはＮを入力するかまたはリターン・キーのみを押下する。指定しなかった場合には，遅れは０と見做される。

② 出力する場合にはＹを入力するかまたはリターン・キーのみを押下する。そうでない場合にはＮを入力する。

〔５〕 連続系状態空間表現形式システムを遅れ時間△ｔ，サンプル周期Ｔで離散系化し，積分型の２次形式評価関数を用いて最適ゲインを求める例

```
/DPACS/ TIME-17:04:58 CPU-00:00:02
OPT/S/I EX1
TO SOLVE OPTMAL REGULATOR PROBLEM
GIVE SAMPLING INTERVAL
DELT =  0.1
CONTROL DELAY ?   Y
CDELT =   0.03
SPECIFY CONTROL VARIABLES ?   N
SYSTEM'S OUTPUTS ARE REGARDED AS CONTROL VARIABLES
GIVE QUADRATIC CRITERION FUNCTION
INPUT -MODE OF ( 2, 2) MATRIX   Q       : I
TYPE OR REVISE ?__
INPUT -MODE OF ( 2, 2) MATRIX   R       : I
TYPE OR REVISE ?__
SOLUTION OF OPTIMAL SAMPLED DATA REGULATOR PROBLEM FOR SYSTEM  EX1.SO
SAMPLING INTERVAL =     0.100000
CONTROL DELAY     =     0.030000
1) COEFFICIENT MATRIX OF CONTROL MATRIX
--- ( 2, 3) MATRIX   H        ---
( 1)  1.00000000D+00  0.0             0.0
( 2)  0.0             1.00000000D+00  1.00000000D+00
***
FILENAME : __
2) SPECIFIED CRITERION FUNCTION
--- ( 2, 2) MATRIX   Q        ---
( 1)  1.00000000D+00  0.0
( 2)  0.0             1.00000000D+00
```

```
FILENAME : __
--- ( 2, 2) MATRIX   R       ---
( 1)  1.00000000D+00  0.0
( 2)  0.0             1.00000000D+00
FILENAME : __
OUTPUT DISCRETIZED SYSTEM ?  N
OUTPUT DISCRETIZED CRITERION FUNCTION ?  Y
4) DISCRETIZED CRITERION FUCTION
--- ( 3, 3) MATRIX   Q11     ---
( 1)  1.14915570D-01  4.03199452D-02  4.55177147D-02
( 2)  4.03199452D-02  8.87583024D-02  1.16750581D-01
( 3)  4.55177147D-02  1.16750581D-01  1.55815254D-01
FILENAME : __
--- ( 3, 2) MATRIX   Q12     ---
( 1)  1.17669740D-03 -1.04280729D-03
( 2)  2.68781367D-03 -2.00427564D-03
( 3)  3.53896720D-03 -2.57710645D-03
FILENAME : __

--- ( 2, 2) MATRIX   Q22     ---
( 1)  8.14137304D-05 -6.06325994D-05
( 2) -6.06325994D-05  4.68858087D-05
FILENAME : __
--- ( 3, 2) MATRIX   S1      ---
( 1)  2.23550768D-03 -2.08704545D-03
( 2)  4.52481489D-03 -3.93877191D-03
( 3)  6.29444080D-03 -5.43549604D-03
FILENAME : __
--- ( 2, 2) MATRIX   S2      ---
( 1)  1.36946901D-04 -1.19144532D-04
( 2) -9.18968727D-05  8.11537611D-05
FILENAME : __
--- ( 2, 2) MATRIX   R       ---
( 1)  1.00312553D-01 -2.65949125D-04
( 2) -2.65949125D-04  1.00227348D-01
FILENAME : __
OUTPUT THE SOLUTION OF RICCATI EQUATION ? N
6) OPTIMAL CONTROL LAW
--- ( 2, 3) MATRIX   F1      ---
( 1)  1.27883717D+01  4.12550424D+00  7.70861028D+00
( 2) -9.85316113D-01 -4.88828452D-01 -6.54497462D-02
FILENAME : __

--- ( 2, 2) MATRIX   F2      ---
( 1)  1.19989800D-01 -1.56831038D-02
( 2) -1.44424776D-02  2.73772741D-02
FILENAME : __
7) POLES OF CLOSED LOOP
------------------------------------------------------------------------
 NO.       REAL                   IMAGINARY              ABSOLUTE
------------------------------------------------------------------------
( 1)    8.60301902D-01         1.24582195D-01        8.69275610D-01
( 2)    8.60301902D-01        -1.24582195D-01        8.69275610D-01
( 3)    7.79285424D-01         0.0                   7.79285424D-01
( 4)   -6.24470861D-15         0.0                   6.24470861D-15
( 5)    6.80558779D-18         0.0                   6.80558779D-18
------------------------------------------------------------------------
FILENAME : __
/DPACS/ TIME-17:47:44 CPU-00:00:21
```

＜ＯＰＴで使用した連続系状態空間表現形式システム＞

```
 /DPACS/ TIME-17:04:59 CPU-00:00:02
TYPE/S EX1
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1.SO        ( 2, 2, 3)

 --- ( 3, 3) MATRIX    A          ---
 ( 1)  0.0                1.00000000D+00  0.0
 ( 2)  1.00000000D+00 -1.00000000D+00  2.00000000D+00
 ( 3)  3.00000000D+00  0.0             1.00000000D+00
 FILENAME :
 --- ( 3, 2) MATRIX    B          ---
 ( 1)  0.0                0.0
 ( 2)  1.00000000D+00 -2.00000000D+00
 ( 3)  0.0                1.00000000D+00
 FILENAME :
 --- ( 2, 3) MATRIX    C          ---
 ( 1)  1.00000000D+00  0.0             0.0
 ( 2)  0.0                1.00000000D+00  1.00000000D+00
 FILENAME :
 --- ( 2, 2) MATRIX    D          ---
 ***
 ( 1)  0.0                0.0
 ( 2)  0.0                0.0
 FILENAME :
 /DPACS/ TIME-17:47:45 CPU-00:00:21

 /DPACS/ TIME-17:05:00 CPU-00:00:02
TYPE/S EX1D
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1D.SO       ( 2, 2, 3)

 --- ( 3, 3) MATRIX    A          ---
 ( 1)  1.00000000D+00  9.53000000D-02  1.00000000D-02
 ( 2)  1.25000000D-01  9.10000000D-01  2.00000000D-01
 ( 3)  3.16000000D-01  1.50000000D-02  1.10000000D+00
 FILENAME :
 --- ( 3, 2) MATRIX    B          ---
 ( 1)  5.00000000D-03 -9.00000000D-03
 ( 2)  9.50000000D-02 -1.80000000D-01
 ( 3)  4.00000000D-04  1.00000000D-01
 FILENAME :
 --- ( 2, 3) MATRIX    C          ---
 ( 1)  1.00000000D+00  0.0             0.0
 ( 2)  0.0                1.00000000D+00  1.00000000D+00
 FILENAME :
 --- ( 2, 2) MATRIX    D          ---
 ***
 ( 1)  0.0                0.0
 ( 2)  0.0                0.0
 FILENAME :
 /DPACS/ TIME-17:47:46 CPU-00:00:21
```

非干渉化問題の求解	DP. Ⅳ. 3
DCPL_SN	

機　能

入力ベクトルと出力ベクトルの次数が等しい状態空間表現形式システムΣ_sに対して，状態フィードバック則による非干渉化問題を解く。

理論概要[14)]

状態方程式を $\begin{cases} \sigma \boldsymbol{x}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t), \ \boldsymbol{x} \in \mathbf{R}^n, \ \boldsymbol{u} \in \mathbf{R}^m, \ \boldsymbol{y} \in \mathbf{R}^m \end{cases}$

これに対し，次の状態フィードバック則を適用する。

$$\boldsymbol{u}(t) = -\boldsymbol{F}\boldsymbol{x}(t) + \boldsymbol{G}\boldsymbol{v}(t), \quad \boldsymbol{v} \in \mathbf{R}^m$$

$\boldsymbol{v}$ は新しい入力ベクトルである。閉ループ系は，

$$\begin{cases} \sigma \boldsymbol{x}(t) = (\boldsymbol{A} - \boldsymbol{B}\boldsymbol{F})\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{G}\boldsymbol{v}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) \end{cases}$$

この式で，伝達関数行列が対角となるようにフィードバック則を求めることが可能かどうか検討し，もし可能ならば望ましい対角伝達関数となるようにフィードバック則を求める。

非干渉化については，次の行列の正則性を調べればよい。

$$\boldsymbol{B}^* \equiv {}^t[{}^t\boldsymbol{C}_1\boldsymbol{A}^{\sigma_1-1}\boldsymbol{B}, \cdots\cdots, {}^t\boldsymbol{C}_m\boldsymbol{A}^{\sigma_m-1}\boldsymbol{B}] \in \mathbf{R}^{m\times m}$$

ただし，$\{\sigma_i\}_{i=1,\cdots,m}$ は，$\boldsymbol{C}_i$ を $\boldsymbol{C}$ の第 i 番目の行ベクトルとすると

$$\sigma_i = \begin{cases} \min \ \{ j \mid {}^t\boldsymbol{C}_i \cdot \boldsymbol{A}^{j-1} \cdot \boldsymbol{B} \neq 0 \} \\ n+1 \ (\forall j \ : {}^t\boldsymbol{C}_i\boldsymbol{A}^j\boldsymbol{B} = 0 \text{ のとき}) \end{cases}$$

もし $\boldsymbol{B}^*$ が正則ならば

$$\boldsymbol{A}^* \equiv {}^t[{}^t\boldsymbol{C}_1\boldsymbol{A}^{\sigma_1}, \cdots\cdots, {}^t\boldsymbol{C}_m\boldsymbol{A}^{\sigma_m}] \in \mathbf{R}^{m\times n}$$

とすると

$$\begin{cases} \boldsymbol{F} = \boldsymbol{B}^{*-1}\boldsymbol{A}^* \\ \boldsymbol{G} = \boldsymbol{B}^{*-1} \end{cases}$$

とおいた閉ループ系の伝達関数は

$$\mathrm{diag}\ \{1/s^{\sigma_i}\}_{i=1,\cdots,m}$$

この関係を利用して

$$A^{*}=\begin{bmatrix} {}^{t}\boldsymbol{C}_{1}\boldsymbol{A}^{\sigma_1}+a_{11}\,{}^{t}\boldsymbol{C}_{1}\boldsymbol{A}^{\sigma_1-1}+\cdots\cdots+a_{1\sigma_1}\,{}^{t}\boldsymbol{C}_{1} \\ \vdots \\ {}^{t}\boldsymbol{C}_{m}\boldsymbol{A}^{\sigma_m}+a_{m1}\,{}^{t}\boldsymbol{C}_{m}\boldsymbol{A}^{\sigma_m-1}+\cdots\cdots+a_{m\sigma_m}\,{}^{t}\boldsymbol{C}_{m} \end{bmatrix}$$

とおけば，閉ループ系の伝達関数は，

$$\mathrm{diag}\ \{1/(s^{\sigma_i}+a_{i1}s^{\sigma_i-1}+\cdots\cdots+a_{i\sigma_i})\}_{i=1,\cdots,m}$$

となる。

従って，まず，閉ループ系の伝達関数の極を指定すれば，それにより伝達関数が固定され，これから，$\boldsymbol{A}^{*}$が定まり，一方$\boldsymbol{B}^{*}$が求まるとすれば，

$$\boldsymbol{F}=(\boldsymbol{B}^{*})^{-1}\boldsymbol{A}^{*}\quad,\ \boldsymbol{G}=(\boldsymbol{B}^{*})^{-1}$$

により制御則が得られる。

実 行 例

〔1〕 状態空間表現形式システムの非干渉化問題を解く例

```
 /DPACS/ TIME-17:05:01 CPU-00:00:02
DCPL EX1
  SYSTEM EX1       HAS THE FOLLOWING SYSTEMDATAS
   1) EX1.S0        (   3) ;SYSIN    /S /C /  /   (87/11/27) :EX1 TEST DATA
   2) EX1.S1        (   3) ;COOD     /1 /  /  /   (87/11/30) :COOD EX1
   3) EX1.S2        (   3) ;COOD     /1 /  /  /   (87/11/30) :COOD/1 EX1
   4) EX1.S3        (   3) ;COOD     /2 /  /  /   (87/11/30) :COOD/2 EX1
   5) EX1.S4        (   3) ;CMR      /M /C /2 /   (87/11/30) :EX1 CMR
  WHICH SYSTEMDATA IS USED ? :   1
  TO SOLVE DECOUPLING PROBLEM
  VALUES OF DECOUPLING-INDICES ①
       2   1
  GIVE  2-DESIRED POLES
  FOR  1-TH TRANSFER FUNCTION DIAGONAL ELEMENT
  INPUT -MODE OF ( 2, 2) MATRIX ROOT       : _
  ( 1) ROW ②
-1,2
  ( 2) ROW
-1,-2
  TYPE OR REVISE ? _
  GIVE  1-DESIRED POLES
  FOR  2-TH TRANSFER FUNCTION DIAGONAL ELEMENT
  INPUT -MODE OF ( 1, 2) MATRIX ROOT       : _
  ( 1) ROW
-1,0
  TYPE OR REVISE ? _
  THE SOLUTION OF DECOUPLING PROBLEM FOR SYSTEM EX1.S0
  1) DECOUPLING-INDICES
       2   1
  2) SPECIFIED POLES
  ------------------------------------------------------------------
   NO.      REAL                 IMAGINARY            ABSOLUTE
  ------------------------------------------------------------------
  ( 1)   -1.00000000D+00      2.00000000D+00       2.23606798D+00
  ( 2)   -1.00000000D+00     -2.00000000D+00       2.23606798D+00
  ( 3)   -1.00000000D+00      0.0                  1.00000000D+00
  ------------------------------------------------------------------
```

```
3) STATE FEEDBACK GAIN
--- ( 2, 3) MATRIX   F      ---
( 1)  2.00000000D+00 -1.00000000D+00  6.00000000D+00
( 2) -2.00000000D+00 -1.00000000D+00  2.00000000D+00
FILENAME : _
4) INPUT TRANSFORMATION MATRIX
--- ( 2, 2) MATRIX   G      ---
( 1) -1.00000000D+00  2.00000000D+00
( 2) -1.00000000D+00  1.00000000D+00
FILENAME : _
/DPACS/ TIME-17:47:48 CPU-00:00:21
```

【説明】

① フィードバックした結果の（$\boldsymbol{A}-\boldsymbol{B}\boldsymbol{F}$）が，実対角行列になれば，Decoupling Indicesは，1がn個並ぶが，もし，実対角にならなければ，極が複素共役になる数分だけ2が並ぶことになる。従って，この2個の極を指定する場合には，複素共役になるように指定しなければならない。

② 複素数の入力である。（この例は，－1＋2j）

オブザーバの構成	DP. Ⅳ. 4
OBS ([/G, /M, /F, /K]) ␣SN␣OSN	

機　能

状態空間表現形式システムΣ_sに対して，オブザーバを構成する。

◆Gopinath のn－p次元状態オブザーバ（／G）

◆Millerのn－p次元最適状態オブザーバ（／M）

◆（可観測性指数－1）次元汎関数オブザーバ（／F）

◆n次元定常カルマンフィルタ（／K）

理論概要

$$\Sigma_s \begin{cases} \sigma \boldsymbol{x}(t)=\boldsymbol{A}\boldsymbol{x}(t)+\boldsymbol{B}\boldsymbol{u}(t) \ , \ \boldsymbol{x}(0)=\boldsymbol{x}_0 \\ \boldsymbol{y}(t)=\boldsymbol{C}\boldsymbol{x}(t)+\boldsymbol{D}\boldsymbol{u}(t), \ \boldsymbol{x}\in\mathbf{R}^n \ , \ \boldsymbol{u}\in\mathbf{R}^m \ , \ \boldsymbol{y}\in\mathbf{R}^p \end{cases}$$

に対して，次式で表わされるもう1つの数式モデルを考える。

$$\Sigma_o \begin{cases} \sigma \boldsymbol{z}(t)=\hat{\boldsymbol{A}}_o\boldsymbol{z}(t)+\hat{\boldsymbol{B}}_o(\boldsymbol{y}(t)-\boldsymbol{D}\boldsymbol{u}(t))+\hat{\boldsymbol{J}}_o\boldsymbol{u}(t) \ , \ \boldsymbol{z}(0)=\boldsymbol{z}_0 \\ \boldsymbol{w}(t)=\hat{\boldsymbol{C}}_o\boldsymbol{z}(t)+\hat{\boldsymbol{D}}_o(\boldsymbol{y}(t)-\boldsymbol{D}\boldsymbol{u}(t)) \ , \ \boldsymbol{z}\in\mathbf{R}^q \ , \ \boldsymbol{w}\in\mathbf{R}^r \end{cases}$$

また状態ベクトルの任意の線型関数を考える。

$$\boldsymbol{x}(t) \ \mapsto \ \boldsymbol{K}\boldsymbol{x}(t) \ , \ \boldsymbol{K}\in\mathbf{R}^{r\times n}$$

もし，$\lim_{t\to\infty}(\boldsymbol{w}(t)-\boldsymbol{K}\boldsymbol{x}(t))=\boldsymbol{0}$のとき，$\Sigma_O$は$\Sigma_S$の$\boldsymbol{K}\boldsymbol{x}$－オブザーバであるという。$\boldsymbol{K}=\boldsymbol{I}_n$ならば状態オブザーバという。$\Sigma_O$が$\boldsymbol{K}\boldsymbol{x}$－オブザーバであるための十分条件は，$\Sigma_O$が漸近安定である行列$\boldsymbol{U}\in\mathbf{R}^{q\times n}$が存在して次の条件が成立することである。

$$\begin{cases} \boldsymbol{U}\boldsymbol{A}=\widehat{\boldsymbol{A}}_O\boldsymbol{U}+\widehat{\boldsymbol{B}}_O\boldsymbol{C} \\ \boldsymbol{U}\boldsymbol{B}=\widehat{\boldsymbol{J}}_O \\ \boldsymbol{K}=\widehat{\boldsymbol{C}}_O\boldsymbol{U}+\widehat{\boldsymbol{D}}_O\boldsymbol{C} \end{cases}$$

以下においてΣ_Sの可観測性と rank $\boldsymbol{C}=p$を仮定。

$\boldsymbol{g}(t)\equiv{}^t[\,{}^t\boldsymbol{y}(t),{}^t\boldsymbol{u}(t)]$とすると$\Sigma_O$は，

$$\Sigma_O\begin{cases} \sigma\boldsymbol{z}(t)=\widehat{\boldsymbol{A}}\boldsymbol{z}(t)+\widehat{\boldsymbol{B}}\boldsymbol{g}(t) \\ \boldsymbol{w}(t)=\widehat{\boldsymbol{C}}\boldsymbol{z}(t)+\widehat{\boldsymbol{D}}\boldsymbol{g}(t) \end{cases}$$

と書け，この$\widehat{\boldsymbol{A}}$，$\widehat{\boldsymbol{B}}$，$\widehat{\boldsymbol{C}}$，$\widehat{\boldsymbol{D}}$を計算する。

◆／G：Gopinath のn－p次元状態オブザーバ[15]

<アルゴリズム>

Step 1： Σ_Sにおける$\boldsymbol{C}$の行ベクトルに独立な行ベクトルからなる$\boldsymbol{C}^*\in\mathbf{R}^{(n-p)\times n}$を選び，次を定義 $\boldsymbol{T}=\begin{bmatrix}\boldsymbol{C}^*\\ \boldsymbol{C}\end{bmatrix}\in\mathbf{R}^{n\times n}$。$\Sigma_S$をこの$\boldsymbol{T}$を用いて，座標変換COOD／2コマンドにより座標変換して$\overline{\Sigma}_S$を得る。(DP.Ⅱ.1参照)

$$\overline{\Sigma}_S\begin{cases} \sigma\overline{\boldsymbol{x}}(t)=\begin{array}{c} n-p\,\{ \\ p\,\{ \end{array}\underbrace{\begin{bmatrix}\overline{\boldsymbol{A}}_{11} & \overline{\boldsymbol{A}}_{12}\\ \overline{\boldsymbol{A}}_{22} & \overline{\boldsymbol{A}}_{22}\end{bmatrix}}_{n-p\quad p}\overline{\boldsymbol{x}}(t)+\begin{array}{c} n-p\,\{ \\ p\,\{ \end{array}\underbrace{\begin{bmatrix}\overline{\boldsymbol{B}}_1\\ \overline{\boldsymbol{B}}_2\end{bmatrix}}_{p}\boldsymbol{u}(t) \\ \boldsymbol{y}(t)=[\,\boldsymbol{0}\quad\boldsymbol{I}_p\,]\,\overline{\boldsymbol{x}}(t)+\boldsymbol{D}\boldsymbol{u}(t) \end{cases}$$

Step 2： $\overline{\Sigma}_S$から次の数式モデルを考える。

$$\begin{cases} \sigma\boldsymbol{x}^{\#}(t)={}^t\overline{\boldsymbol{A}}_{11}\boldsymbol{x}^{\#}(t)+{}^t\overline{\boldsymbol{A}}_{21}\boldsymbol{u}^{\#}(t) \\ \boldsymbol{y}^{\#}(t)=\boldsymbol{x}^{\#}(t) \end{cases}$$

Σ_Sが可観測であれば，$\Sigma^{\#}$は可制御であるので，次の形の状態フィードバックによる安定化則を得ることができる。この${}^t\boldsymbol{L}$は極配置問題や最適制御問題を解くことで得られる。

$$\boldsymbol{u}^{\#}(t)=-{}^t\boldsymbol{L}\,\boldsymbol{x}^{\#}(t)$$

Step 3： 次の行列を計算する。

$$\begin{cases} \hat{A}_0 = \bar{A}_{11} - L\bar{A}_{21} \\ \hat{B}_0 = \bar{A}_{12} - L\bar{A}_{22} + \hat{A}_0 L \\ \hat{C}_0 = T^{-1}\begin{bmatrix} I_p \\ 0 \end{bmatrix} \\ \hat{D}_0 = T^{-1}\begin{bmatrix} L \\ I_p \end{bmatrix} \\ \hat{J}_0 = \bar{B}_1 - L\bar{B}_2 \end{cases}$$

Step 4： Σ_0 のパラメータを次式のように構成する。

$$\hat{A} = \hat{A}_0$$
$$\hat{B} = [\ \hat{B}_0 \ \vdots\ \hat{J}_0 - \hat{B}_0 \cdot D\ ]$$
$$\hat{C} = \hat{C}_0$$
$$\hat{D} = [\ \hat{D}_0 \ \vdots\ -\hat{D}_0 \cdot D\ ]$$

◆／M：Millerのn－p次元最適状態オブザーバ[16]

<アルゴリズム>

数式モデルΣ_Sの初期状態$\boldsymbol{x}_0$の平均値$\boldsymbol{x}_m$と共分散行列$\boldsymbol{S}$が与えられるとき，次のアルゴリズムによりオブザーバが得られる。 ($\boldsymbol{S} \in \boldsymbol{R}^{n\times m}$)

Step 1： Σ_S における C のベクトルに独立な行ベクトルからなる $C^* \in \mathbf{R}^{(n-p)\times n}$ を選び，次式から $V^* \in \mathbf{R}^{n\times(n-p)}$ を求める。

$$[\ V \quad V^*\ ] = \begin{bmatrix} C \\ C^* \end{bmatrix}^{-1}$$

Step 2： 次の定常リカッチ方程式の解 $\bar{P} \in \mathbf{R}^{(n-p)\times(n-p)}$ を求める。

$$\bar{P}\,{}^t\bar{A} + \bar{A}\bar{P} - \bar{P}\,{}^t\bar{C}\ \bar{R}^{-1}\bar{C}\bar{P} + \bar{Q} = 0$$

ここで，$\bar{A} \in \mathbf{R}^{(n-p)\times(n-p)}$，$\bar{C} \in \mathbf{R}^{p\times(n-p)}$，$\bar{Q} \in \mathbf{R}^{(n-p)\times(n-p)}$，$\bar{R} \in \mathbf{R}^{p\times p}$

$$\bar{A} = C^*(I_{n-p} - S\,{}^tC(CS\,{}^tC)^{-1}C)AV^*$$
$$\bar{C} = CAV\ ,\ \bar{Q} = C^*(I_{n-p} - S\,{}^tC(CS\,{}^tC)^{-1}C)SV^*$$
$$\bar{R} = CS\,{}^tC$$

Step 3： 次の行列を計算する。

$$\hat{A}_0 = UAV^*\ ,\ \hat{B}_0 = UA\hat{D}\ ,\ \hat{C}_0 = V^*$$
$$\hat{D}_0 = (V^*\bar{P}\,{}^tV^*\,{}^tA + S)\,{}^tC(CS\,{}^tC)^{-1}\ ,\ \hat{J}_0 = UB\ ,\ U = C^*(I_n - \hat{D}_0 C)$$

Step 4： Σ_oのパラメータを次式により求める。

$$\hat{A}=\hat{A}_o$$
$$\hat{B}=[\ \hat{B}_o \vdots \hat{J}_o-\hat{B}_o D\ ]$$
$$\hat{C}=\hat{C}_o$$
$$\hat{D}=[\ \hat{D}_o \vdots -\hat{D}_o\ D\ ]$$

◆／F：（可観測性指数－１）次元汎関数オブザーバ[17]

<梶原，古田のアルゴリズム>

数式モデルΣ_s及び状態ベクトルの線型汎関数：$w(t)=-{}^t\boldsymbol{f}\cdot\boldsymbol{x}(t)$，$w(t)\in\mathbf{R}$を考え，wを推定する汎関数オブザーバは次のアルゴリズムにより得られる。その次数は，可観測性指数νとするとき，$\nu-1$である。

Step 1：（${}^t\boldsymbol{A}, {}^t\boldsymbol{C}$）のクロネッカ不変量$\{n_i\}_{i=1,\sim,p}$とレギュラベクトルからなる行列$\boldsymbol{T}\in\mathbf{R}^{n\times n}$をSYSサブシステムLCF（SY．I．2参照）により求める。 $\nu=\max\{\nu_i\}$とおく

Step 2：実軸に対称で安定な（$\nu-1$）個の極$\{\lambda_i\}_{i=1,\sim,\nu-1}$を与え
$\prod_{i=1}^{\nu-1}(\lambda-\lambda_i)=\lambda^{\nu-1}+\alpha_1\lambda^{\nu-2}+\cdots\cdots+\alpha_{\nu-1}$ から $\{\alpha_i\}_{i=1,\sim,\nu-1}$を求める。

Step 3：次式から$\boldsymbol{x}$を計算する。

$$\begin{aligned}\boldsymbol{x}&={}^t[\,x_{11},\sim,x_{1n1},\sim,\ x_{p1},\sim x_{pnp}\,]\\&=\boldsymbol{T}^{-1}[\ \boldsymbol{f},{}^t\boldsymbol{A}\boldsymbol{f},\cdots\cdots,{}^t\boldsymbol{A}^{\nu-1}\boldsymbol{f}\ ]^t[\ \alpha_{\nu-1},\cdots\cdots,\alpha_1,1\,]\end{aligned}$$

Step 4：次式により$\hat{B}_o\in\mathbf{R}^{(\nu-1)}$と${}^t\hat{d}_o\in\mathbf{R}^{1\times p}$を求める。

$$\begin{bmatrix}{}^t\hat{d}_o\\ \hat{B}_o\end{bmatrix}=\begin{bmatrix}\alpha_{\nu-1}&\cdots&\alpha_1&1\\ \vdots&\iddots&\iddots&\\ \alpha_1&\iddots&&\boldsymbol{0}\\ 1&&&\end{bmatrix}\begin{bmatrix}x_{11}&\cdots&x_{i1}&\cdots&x_{p1}\\ \vdots&&\vdots&&\vdots\\ x_{1n1}&&\vdots&&x_{pnp}\\ \boldsymbol{0}&&x_{i\nu}&&\boldsymbol{0}\end{bmatrix}$$

Step 5：次式により，$\boldsymbol{U}\in\mathbf{R}^{(\nu-1)\times p}$を求める。

$${}^t\boldsymbol{u}_1={}^t\boldsymbol{f}-{}^t\hat{\boldsymbol{d}}_o\cdot\boldsymbol{C}\ ,\ {}^t\boldsymbol{u}_i={}^t\boldsymbol{u}_{i-1}\boldsymbol{A}-{}^t\hat{\boldsymbol{b}}_{i-1}\cdot\boldsymbol{C}\qquad(i=2,\sim,\ \nu-1)$$

ここで，${}^t\boldsymbol{u}_i$と${}^t\boldsymbol{b}_i$はそれぞれ$\boldsymbol{U}$と$\hat{B}_o$のi番目の行ベクトルである。

Step 6：次式により$\hat{A}_o$，${}^t\hat{C}_o$，$\hat{J}_o$を求める。

$$\hat{A}_o=\begin{bmatrix}\boldsymbol{0}&1&&\boldsymbol{0}\\ &&\ddots&\\ &\boldsymbol{0}&&1\\ -\alpha_{\nu-1}&\cdots&\cdots&-\alpha_1\end{bmatrix}\in\mathbf{R}^{(\nu-1)\times(\nu-1)}\ ,$$

$${}^t\hat{C}_o=[1,0\cdots\cdots,0]\in\mathbf{R}^{1\times(\nu-1)}\ ,\ \hat{J}_o=\boldsymbol{U}\boldsymbol{B}$$

Step 7：Σ_oのパラメータを次式により求める。

$$\hat{A}=\hat{A}_o\ ,\ \hat{B}=[\ \hat{B}_o \vdots \hat{J}_o-\hat{B}_o\boldsymbol{D}\ ]\ ,\ \hat{C}={}^t\hat{C}_o\ ,\ \hat{D}=[\ \boldsymbol{d}_o \vdots -\hat{\boldsymbol{d}}_o\boldsymbol{D}\ ]$$

◆／K：n次元定常カルマンフィルタ[18)]

＜アルゴリズム＞

$$\begin{cases} \sigma \boldsymbol{x}(t)=\boldsymbol{A}\boldsymbol{x}(t)+\boldsymbol{B}\boldsymbol{u}(t)+\boldsymbol{B}^{*}\boldsymbol{v}(t), & \boldsymbol{x}(0)=\boldsymbol{x}_0 \\ \boldsymbol{y}(t)=\boldsymbol{C}\boldsymbol{x}(t)+\boldsymbol{D}\boldsymbol{u}(t)+\boldsymbol{w}(t) \end{cases}$$

$$\boldsymbol{x}(t)\in\mathbf{R}^{n},\ \boldsymbol{u}(t)\in\mathbf{R}^{m},\ \boldsymbol{y}(t)\in\mathbf{R}^{p},\ \boldsymbol{v}(t)\in\mathbf{R}^{r},\ \boldsymbol{w}(t)\in\mathbf{R}^{p}$$

システム雑音 $\boldsymbol{v}(t)$ と観測雑音 $\boldsymbol{w}(t)$ は平均＝0の正規分布に従う互いに独立な白色雑音であり，その共分散行列をそれぞれ，$\boldsymbol{Q}>0$，$\boldsymbol{R}>0$ とする。また，$(\boldsymbol{A},\boldsymbol{B}^{*})$ は可制御（安定化可能）対，$(\boldsymbol{A},\boldsymbol{C})$ は可観測（検出可能）対であるとする。このとき定常カルマンフィルタΣ_0は，次式で与えられる。

$$\hat{\boldsymbol{A}}=\boldsymbol{A}-\boldsymbol{K}\boldsymbol{C},\ \hat{\boldsymbol{B}}=[\ \boldsymbol{K}\ \vdots\ \boldsymbol{B}-\boldsymbol{K}\boldsymbol{D}\ ],\ \hat{\boldsymbol{C}}=\boldsymbol{I}_n,\ \hat{\boldsymbol{D}}=\boldsymbol{0}$$

ここで，${}^{t}\boldsymbol{K}=\boldsymbol{R}^{-1}\boldsymbol{C}\boldsymbol{P}$（連続系）$=(\boldsymbol{C}\boldsymbol{P}{}^{t}\boldsymbol{C}+\boldsymbol{R})^{-1}\cdot{}^{t}\boldsymbol{C}\boldsymbol{P}{}^{t}\boldsymbol{A}$（離散系）

ただし　$\boldsymbol{P}\in\mathbf{R}^{n\times n}$ は次の定常リカッチ方程式の正定（非負定）対称解である。

$$\begin{cases} \boldsymbol{P}{}^{t}\boldsymbol{A}+\boldsymbol{A}\boldsymbol{P}-\boldsymbol{P}{}^{t}\boldsymbol{C}\boldsymbol{R}^{-1}\boldsymbol{C}\boldsymbol{P}+\boldsymbol{B}^{*}\boldsymbol{Q}{}^{t}\boldsymbol{B}^{*}=\boldsymbol{0} & \text{（連続系）} \\ \boldsymbol{P}=\boldsymbol{A}\boldsymbol{P}{}^{t}\boldsymbol{A}-{}^{t}\boldsymbol{A}\boldsymbol{P}{}^{t}\boldsymbol{C}(\boldsymbol{C}\boldsymbol{P}{}^{t}\boldsymbol{C}+\boldsymbol{R})^{-1}\boldsymbol{C}\boldsymbol{P}{}^{t}\boldsymbol{A}+\boldsymbol{B}^{*}\boldsymbol{Q}{}^{t}\boldsymbol{B}^{*} & \text{（離散系）} \end{cases}$$

実行例

〔1〕　Gopinath のn－p次元状態オブザーバの構成例

```
/DPACS/ TIME-17:05:04 CPU-00:00:02
OBS/G EX1 EX1G
 SYSTEM EX1       HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0       (    3) ;SYSIN    /S /C /  /    (87/11/27) :EX1 TEST DATA
  2) EX1.S1       (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD EX1
  3) EX1.S2       (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD/1 EX1
  4) EX1.S3       (    3) ;COOD     /2 /  /  /    (87/11/30) :COOD/2 EX1
  5) EX1.S4       (    3) ;CMR      /M /C /2 /    (87/11/30) :EX1 CMR
 WHICH SYSTEMDATA IS USED ? :    1
 BY GOPINATH'S METHOD
 WHICH STABILIZING METHOD IS USED ?
 (1) POLE ASSIGNMENT OR (2) OPTIMAL CONTROL :   1 ①
 GIVE POLES TO BE ASSIGNED
 INPUT -MODE OF ( 1, 2) MATRIX ROOT      : _
 ( 1) ROW
-5,0
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 1) MATRIX  F0       : _
 ( 1) ROW
1
 ( 2) ROW
-1
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 1, 3) MATRIX   U       : _
 --- ( 1, 3) MATRIX   U       ---
 ( 1) -1.60000000D+00  1.00000000D+00  0.0
 FILENAME : _
 GIVE COMMENT FOR SYSTEMDATA :
OBS/G
 /DPACS/ TIME-17:47:49 CPU-00:00:21
```

【説明】

① 安定化手法として極配置問題を解く場合には1を，最適制御問題を解く場合には2を指定する。（理論概要参照）以降の操作はＰＬＯＣコマンド（ＤＰ．Ⅳ．１），ＯＰＴコマンド（ＤＰ．Ⅳ．２）と同じ。

〔２〕 Millerのｎ－ｐ次元最適状態オブザーバの構成例

```
/DPACS/ TIME-17:05:11 CPU-00:00:02
OBS/M EX1 EX1M
 SYSTEM EX1       HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0       (    3) ;SYSIN    /S /C /  /    (87/11/27) :EX1 TEST DATA
  2) EX1.S1       (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD EX1
  3) EX1.S2       (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD/1 EX1
  4) EX1.S3       (    3) ;COOD     /2 /  /  /    (87/11/30) :COOD/2 EX1
  5) EX1.S4       (    3) ;CMR      /M /C /2 /    (87/11/30) :EX1 CMR
 WHICH SYSTEMDATA IS USED ? :   1
 TO OBTAIN MILLER'S OBSERVER
 GIVE MEAN AND COVARIANCE OF THE INITIAL STATE
 INPUT -MODE OF ( 1, 3) MATRIX MEAN      : Z
 TYPE OR REVISE ? _                            ①
 INPUT -MODE OF ( 3, 3) MATRIX VAL.      : I
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 1, 3) MATRIX  U        : _
 --- ( 1, 3) MATRIX  U        ---
 ( 1)   1.49467542D-01 -7.98935084D-01  2.01064916D-01
 FILENAME :_
 OUTPUT-MODE OF ( 1, 1) MATRIX Z(0)      : _
 --- ( 1, 1) MATRIX Z(0)      ---
 ( 1)   0.0
 FILENAME :_
 GIVE COMMENT FOR SYSTEMDATA :
OBS/M
 /DPACS/ TIME-17:48:25 CPU-00:00:21
```

【説明】

① 初期状態変数に関する平均値と共分散行列の入力

```
/DPACS/ TIME-17:05:13 CPU-00:00:02
TYPE/S EX1M
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1M.SO      ( 4, 3, 1)

 --- ( 1, 1) MATRIX   A       ---
 ( 1) -2.34520788D+00
 FILENAME : _
 --- ( 1, 4) MATRIX   B       ---
 ( 1)  1.54792120D-01 -9.25266229D-01 -7.98935084D-01  1.79893508D+00
 FILENAME : _
 --- ( 3, 1) MATRIX   C       ---
 ( 1)  0.0
 ( 2) -1.00000000D+00
 ( 3)  1.00000000D+00
 FILENAME : _
 --- ( 3, 4) MATRIX   D       ---
 ( 1)  1.00000000D+00  0.0             0.0             0.0
 ( 2)  1.49467542D-01  2.01064916D-01  0.0             0.0
 ( 3) -1.49467542D-01  7.98935084D-01  0.0             0.0
 FILENAME : _
 /DPACS/ TIME-17:48:27 CPU-00:00:21
```

〔3〕 (可観測性指数－1) 次元汎関数オブザーバの構成例

```
 /DPACS/ TIME-17:05:16 CPU-00:00:02
OBS/F EX3 EX3F
 TO OBTAIN FUNCTIONAL OBSERVER
 GIVE EPSILON, TO DETERMINE RANK OF MATRIX V@ EPS= 1.0D-10
 COLUMN INDEPENDENCY OF MATRIX V@ IS INVESTIGATED
  ( 1)
   1    1
  ( 2)
   1    0
  ( 3)
   0    0
 CONDITION NUMBERS ARE AS FOLLOWS
 --- ( 3, 2) MATRIX CNUM        ---
 ( 1)  1.00000000D+00  1.00000000D+00
 ( 2)  9.79016940D-01  2.04975209D-25
 ( 3)  1.51812054D-18  2.61120934D-22
 CHANGE THE EPSILON ?  N
 GIVE LINEAR FUNCTIONAL
 INPUT -MODE OF ( 1, 3) MATRIX   F        : _  ①

 ( 1) ROW
1,2,1
 TYPE OR REVISE ?_
 OBSERVABILITY INDEX IS  2
  1 -DIMENSIONAL FUNCTIONAL OBSERVER IS CONSTRUCTED
 GIVE  1 -STABLE POLES OF FUNCTIONAL OBSERVER
 INPUT -MODE OF ( 1, 2) MATRIX ROOT       : _  ②
```

```
  ( 1) ROW
-5,0
  TYPE OR REVISE ? _
  OUTPUT-MODE OF ( 1, 3) MATRIX   U      : _
  --- ( 1, 3) MATRIX   U      ---
  ( 1)  1.00000000D+00  2.85714286D-01  2.71428571D+00
  FILENAME : ____
  GIVE COMMENT FOR SYSTEMDATA :
OBS/F
 /DPACS/ TIME-17:48:37 CPU-00:00:21
```

【説明】

① 状態変数の線型結合係数を指定する。

② オブザーバの極を指定する。（複素数）

```
 /DPACS/ TIME-17:05:18 CPU-00:00:02
LINK EX3 EX3F EX3FT
 TO LINK SYSTEMS
 SYSTEM EX3.SO       REFERRED BY SYSTEM NO.  1 HAS BEEN  LOADED
 SYSTEM EX3F.SO      REFERRED BY SYSTEM NO.  2 HAS BEEN  LOADED
 DEFINE ADDER ?  N
 DEFINE INTEGRATER ?  N
 SPECIFY CONNECTION OF SYSTEMS
 FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :   1,2
 INPUT -MODE OF ( 3, 2) MATRIX GAIN      : I
 TYPE OR REVISE ? _
 REPEAT CONNECTING ? _
 FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :   2,2
 INPUT -MODE OF ( 3, 1) MATRIX GAIN      : _
 ( 1) ROW
0
 ( 2) ROW
0
 ( 3) ROW
-1
 TYPE OR REVISE ? _
 REPEAT CONNECTING ?  N
 TYPE OUT THE LINKAGE ?  N
 REVISE THE CONNECTION ?  N
 SET NEW CONNECTION ?  N
 GIVE COMMENT FOR SYSTEMDATA :
EX3 LINK
 /DPACS/ TIME-17:48:40 CPU-00:00:21
```

〔4〕　n次元定常カルマンフィルタの構成例

```
 /DPACS/ TIME-17:05:20 CPU-00:00:03
OBS/K EX1 EX1K
 SYSTEM EX1         HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0         (    3) ;SYSIN    /S /C /  /     (87/11/27) :EX1 TEST DATA
  2) EX1.S1         (    3) ;COOD     /1 /  /  /     (87/11/30) :COOD EX1
  3) EX1.S2         (    3) ;COOD     /1 /  /  /     (87/11/30) :COOD/1 EX1
  4) EX1.S3         (    3) ;COOD     /2 /  /  /     (87/11/30) :COOD/2 EX1
  5) EX1.S4         (    3) ;CMR      /M /C /2 /     (87/11/30) :EX1 CMR
 WHICH SYSTEMDATA IS USED ? :    1
 TO OBTAIN KALMAN FILTER
 SPECIFY DIMENSION OF SYSTEM NOISE VECTOR (N= 3,M= 2) : 3
 INPUT -MODE OF ( 3, 3) MATRIX  B*        : I
 TYPE OR REVISE ? _
 GIVE COVARIANCE MATRICES OF SYSTEM AND OBSERVATION NO ISE
 INPUT -MODE OF ( 3, 3) MATRIX  Q         : I
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX  R         : I
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 3, 2) MATRIX  K         : _
 --- ( 3, 2) MATRIX  K       ---
 ( 1)  7.59863969D-01  1.17142543D+00
 ( 2)  4.74815399D-01  1.93855659D+00
 ( 3)  6.96610035D-01  2.92978844D+00
 FILENAME : _
 GIVE COMMENT FOR SYSTEMDATA :
OBS/K
 /DPACS/ TIME-17:48:42 CPU-00:00:21


 /DPACS/ TIME-17:05:21 CPU-00:00:03
TYPE/S EX1K
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1K.S0      ( 4, 3, 3)

 --- ( 3, 3) MATRIX   A      ---
 ( 1) -7.59863969D-01 -1.71425434D-01 -1.17142543D+00
 ( 2)  5.25184601D-01 -2.93855659D+00  6.14434059D-02
 ( 3)  2.30338997D+00 -2.92978844D+00 -1.92978844D+00
 FILENAME : _
 --- ( 3, 4) MATRIX   B      ---
 ( 1)  7.59863969D-01  1.17142543D+00  0.0             0.0
 ( 2)  4.74815399D-01  1.93855659D+00  1.00000000D+00 -2.00000000D+00
 ( 3)  6.96610035D-01  2.92978844D+00  0.0             1.00000000D+00
 FILENAME : _
 --- ( 3, 3) MATRIX   C      ---
 ( 1)  1.00000000D+00  0.0             0.0
 ( 2)  0.0             1.00000000D+00  0.0
 ( 3)  0.0             0.0             1.00000000D+00
 FILENAME : _
 --- ( 3, 4) MATRIX   D      ---
 ( 1)  0.0             0.0             0.0             0.0
 ( 2)  0.0             0.0             0.0             0.0
 ( 3)  0.0             0.0             0.0             0.0
 FILENAME : _
 /DPACS/ TIME-17:48:43 CPU-00:00:21
```

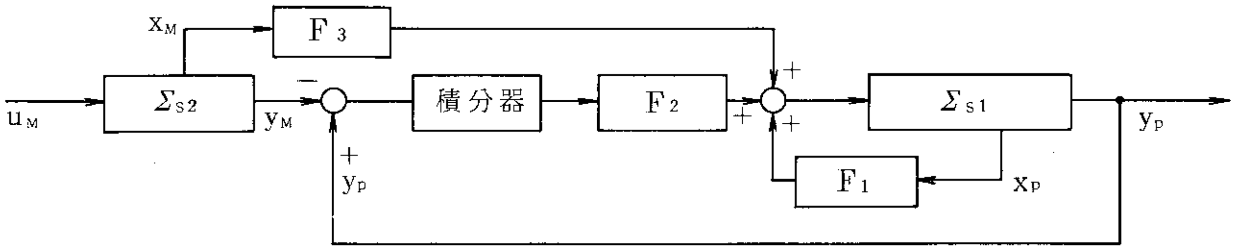
モデル追従型サーボ問題の求解	DP. Ⅳ. 5
MFS_SN1_SN2	

機　能

状態空間表現形式システムΣ_{s1}（SN1）の出力を，ステップ状外乱を抑制しながら状態空間表現形式システムΣ_{s2}（SN2）のステップ入力に対する出力に追従させる制御則を得る。

理論概要

次の線図に対して制御則を得る。



制御則は，
$$\begin{cases} \boldsymbol{u}_p(t) = \boldsymbol{F}_1 \boldsymbol{x}_p(t) + \boldsymbol{\eta}(t) + \boldsymbol{F}_3 \boldsymbol{x}_M(t) \\ \boldsymbol{\eta}(t) = \boldsymbol{F}_2 \int_0^t (\boldsymbol{y}_p(\tau) - \boldsymbol{y}_M(\tau))\, d\tau \end{cases}$$

今数式モデル

$$\Sigma_{s1} \begin{cases} \sigma \boldsymbol{x}_1(t) = \boldsymbol{A}_1 \boldsymbol{x}_1(t) + \boldsymbol{B}_1 \boldsymbol{u}_1(t) \\ \boldsymbol{y}_1(t) = \boldsymbol{C}_1 \boldsymbol{x}_1(t) \end{cases}$$

$$\Sigma_{s2} \begin{cases} \sigma \boldsymbol{x}_2(t) = \boldsymbol{A}_2 \boldsymbol{x}_2(t) + \boldsymbol{B}_2 \boldsymbol{u}_2(t) \\ \boldsymbol{y}_2(t) = \boldsymbol{C}_2 \boldsymbol{x}_2(t) \end{cases}$$

について次の仮定をおく

(1) $m_1 \geq p_1$　(2) $\mathrm{rank} \begin{bmatrix} \boldsymbol{A}_1 & \boldsymbol{B}_1 \\ \boldsymbol{C}_1 & \boldsymbol{0} \end{bmatrix} = n_1 + p_1$

(3) $p_1 = p_2$　(4) Σ_{s2} は可制御・可観測

なお，$\boldsymbol{x}_i \in \mathbf{R}^{ni}$，$\boldsymbol{u}_i \in \mathbf{R}^{mi}$，$\boldsymbol{y}_i \in \mathbf{R}^{pi}$，$i = 1, 2$

このとき次の拡大系を得る。

$$\Sigma_E \begin{cases} \sigma \overline{\boldsymbol{x}}(t)=\begin{bmatrix} \boldsymbol{A}_1 & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{C}_1 & * & -\boldsymbol{C}_2 \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{A}_2 \end{bmatrix} \overline{\boldsymbol{x}}(t)+\begin{bmatrix} \boldsymbol{B}_1 \\ \boldsymbol{0} \\ \boldsymbol{0} \end{bmatrix} \overline{\boldsymbol{u}}(t) \\ \overline{\boldsymbol{y}}(t)=[\boldsymbol{0},\ \boldsymbol{I},\ \boldsymbol{0}]\ \overline{\boldsymbol{x}}(t) \end{cases}$$

$$\overline{\boldsymbol{x}}(t)\in \mathbf{R}^{(n1+p1+n2)},\ \overline{\boldsymbol{u}}(t)\in \mathbf{R}^{m1},\ \overline{\boldsymbol{y}}(t)\in \mathbf{R}^{p1}$$

ここで $*$ は Σ_{S1} が連続型の時，$p_1\times p_1$ の零行列，離散型の時 $\boldsymbol{I}_{p1}$ を表わす。

拡大型 Σ_E に対する出力の最適レギュレータ問題を解くことにより，次の形の状態フィードバック則を得る。

$$\overline{\boldsymbol{u}}(t)=\overline{\boldsymbol{F}}\,\overline{\boldsymbol{x}}(t)$$

$\overline{\boldsymbol{x}} \equiv {}^t({}^t\boldsymbol{x}_1\ {}^t\eta\ {}^t\boldsymbol{x}_2)$ より， $\boldsymbol{F}_1$ ， $\boldsymbol{F}_2$ ， $\boldsymbol{F}_3$ を求める。

実 行 例

〔1〕 モデル追従型サーボ問題を解く例

```
/DPACS/ TIME-17:26:05 CPU-00:00:02
TYPE/S EX1
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1.SO        ( 2, 2, 3)

 --- ( 3, 3) MATRIX    A        ---
 ( 1)  0.0               1.00000000D+00  0.0
 ( 2)  1.00000000D+00 -1.00000000D+00  2.00000000D+00
 ( 3)  3.00000000D+00  0.0             1.00000000D+00
 FILENAME : _
 --- ( 3, 2) MATRIX    B        ---
 ( 1)  0.0               0.0
 ( 2)  1.00000000D+00 -2.00000000D+00
 ( 3)  0.0               1.00000000D+00
 FILENAME : _
 --- ( 2, 3) MATRIX    C        ---
 ( 1)  1.00000000D+00  0.0             0.0
 ( 2)  0.0             1.00000000D+00  1.00000000D+00
 FILENAME :   EX1C
 --- ( 2, 2) MATRIX    D        ---
 ( 1)  0.0               0.0
 ( 2)  0.0               0.0
 FILENAME : _

/DPACS/ TIME-17:28:55 CPU-00:00:08
TYPE/S EX1A
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1A.SO       ( 2, 3, 3)
```

```
--- ( 3, 3) MATRIX   A          ---
( 1)  0.0              1.00000000D+00  0.0
( 2)  1.00000000D+00 -1.00000000D+00  2.00000000D+00
( 3)  3.00000000D+00  0.0              1.00000000D+00
FILENAME : _
--- ( 3, 2) MATRIX   B          ---
( 1)  0.0              0.0
( 2)  1.00000000D+00 -2.00000000D+00
( 3)  0.0              1.00000000D+00
FILENAME : _
--- ( 3, 3) MATRIX   C          ---
( 1)  1.00000000D+00  0.0              0.0
( 2)  0.0              1.00000000D+00  0.0
( 3)  0.0              0.0              1.00000000D+00
FILENAME : _
--- ( 3, 2) MATRIX   D          ---
( 1)  0.0              0.0
( 2)  0.0              0.0
***
( 3)  0.0              0.0
FILENAME : _
/DPACS/ TIME-17:30:04 CPU-00:00:13

/DPACS/ TIME-17:31:34 CPU-00:00:17
TYPE/S MODEL
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA MODEL.SO     ( 2, 2, 4)

--- ( 4, 4) MATRIX   A          ---
( 1)  0.0              1.00000000D+00  0.0              0.0
( 2) -2.00000000D+00 -2.00000000D+00  0.0              0.0
( 3)  0.0              0.0              0.0              1.00000000D+00
( 4)  0.0              0.0             -2.00000000D+00 -2.00000000D+00
FILENAME : _
--- ( 4, 2) MATRIX   B          ---
( 1)  0.0              0.0
( 2)  2.00000000D+00  0.0
( 3)  0.0              0.0
( 4)  0.0              2.00000000D+00
FILENAME : _
--- ( 2, 4) MATRIX   C          ---
( 1)  1.00000000D+00  0.0              0.0              0.0
( 2)  0.0              0.0              1.00000000D+00  0.0
FILENAME :   MDLC
--- ( 2, 2) MATRIX   D          ---
***
( 1)  0.0              0.0
( 2)  0.0              0.0
FILENAME : _
/DPACS/ TIME-17:35:10 CPU-00:00:24
TYPE/S MODELA
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA MODELA.SO   ( 2, 4, 4)
```

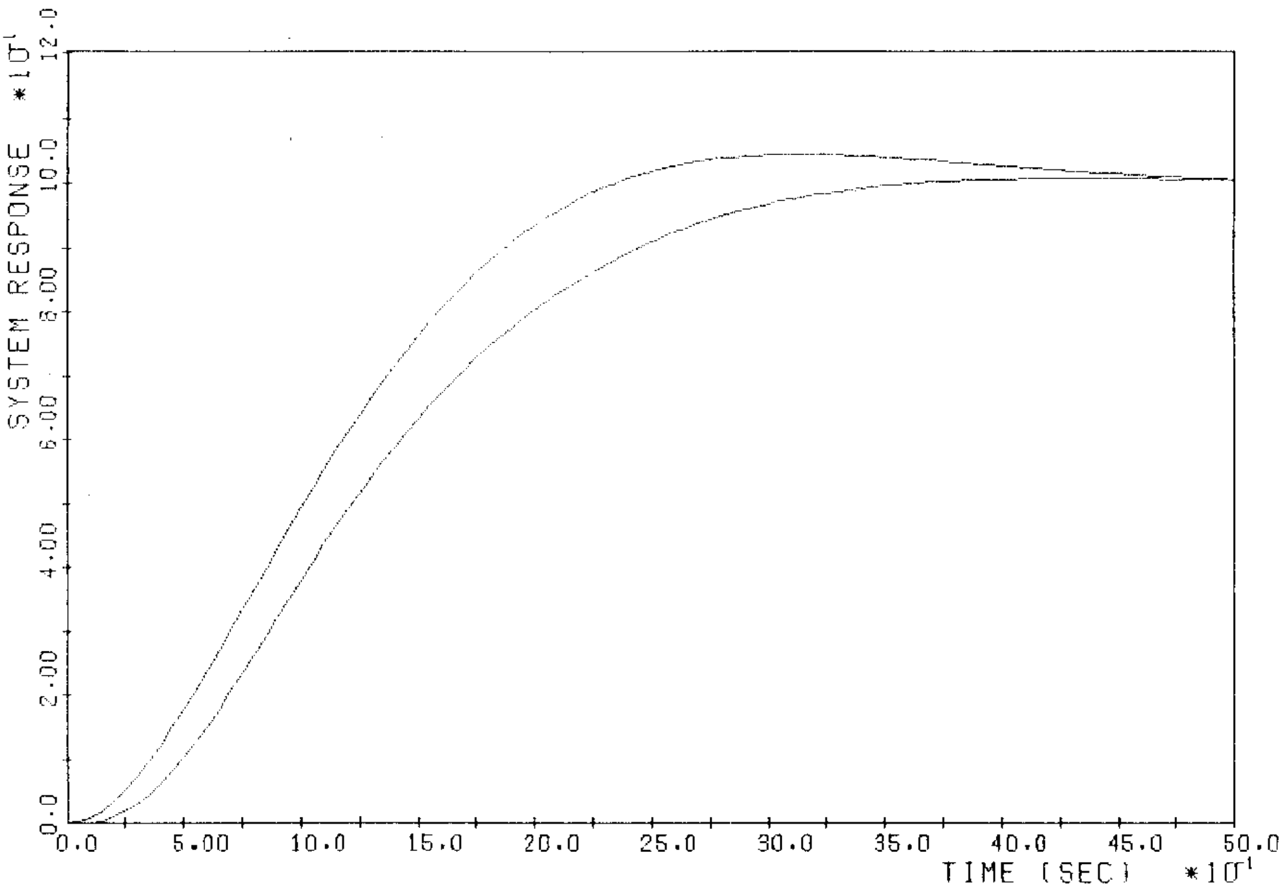
```
--- ( 4, 4) MATRIX    A        ---
( 1)  0.0               1.00000000D+00  0.0               0.0
( 2) -2.00000000D+00 -2.00000000D+00  0.0               0.0
( 3)  0.0               0.0              0.0               1.00000000D+00
( 4)  0.0               0.0             -2.00000000D+00 -2.00000000D+00
FILENAME : _
--- ( 4, 2) MATRIX    B        ---
( 1)  0.0               0.0
( 2)  2.00000000D+00  0.0
( 3)  0.0               0.0
( 4)  0.0               2.00000000D+00
FILENAME : _
--- ( 4, 4) MATRIX    C        ---
( 1)  1.00000000D+00  0.0              0.0               0.0
( 2)  0.0               1.00000000D+00  0.0               0.0
***
( 3)  0.0               0.0              1.00000000D+00  0.0
( 4)  0.0               0.0              0.0               1.00000000D+00
FILENAME : _
--- ( 4, 2) MATRIX    D        ---
( 1)  0.0               0.0
( 2)  0.0               0.0
( 3)  0.0               0.0
( 4)  0.0               0.0
FILENAME : _
/DPACS/ TIME-17:36:23 CPU-00:00:28

/DPACS/ TIME-17:36:51 CPU-00:00:28
MFS EX1 MODEL
 TO SOLVE MODEL FOLLOWING SERVO PROBLEM
 SYSTEMS ARE IN CONTINUOUS TYPE
 GIVE QUADRATIC CRITERION FUNCTION
 INPUT -MODE OF ( 2, 2) MATRIX    Q        : D
 (ONLY DIAGONAL PART)
 ( 1) ROW
100
 ( 2) ROW
100
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX    R        : I
 TYPE OR REVISE ? _
 THE SOLUTION OF MODEL FOLLOWING SERVO PROBLEM FOR SYSTEM EX1.SO
 AND MODEL MODEL.SO
 1)SPECIFIED CRITERION FUNCTION
 --- ( 2, 2) MATRIX    Q        ---
 ( 1)  1.00000000D+02  0.0
 ( 2)  0.0               1.00000000D+02
 FILENAME : _
 --- ( 2, 2) MATRIX    R        ---
 ***
 ( 1)  1.00000000D+00  0.0
 ( 2)  0.0               1.00000000D+00
 FILENAME : _
 2)OPTIMAL CONTROL LAW
 --- ( 2, 3) MATRIX    F1       ---
 ( 1) -1.27622603D+01 -5.62635857D+00 -1.11021269D+01
 ( 2)  6.00246031D+00  1.50590249D-01 -4.34998749D+00
```

```
FILENAME :   F1
--- ( 2, 2) MATRIX    F2    ---
( 1) -4.92452657D+00 -8.70339233D+00
( 2)  8.70339233D+00 -4.92452657D+00
FILENAME :   F2
--- ( 2, 4) MATRIX    F3    ---
( 1)  3.65399418D+00  1.23457030D+00  4.94972703D+00  1.51909614D+00
( 2) -6.12552844D+00 -1.95012162D+00  3.89849093D+00  9.25007239D-01
FILENAME :   F3
3)POLES OF CLOSED LOOP
-------------------------------------------------------------------
 NO.      REAL                  IMAGINARY              ABSOLUTE
-------------------------------------------------------------------
( 1)   -1.26928011D+00       0.0                    1.26928011D+00
( 2)   -2.04945561D+00       1.30267546D+00         2.42842164D+00
( 3)   -2.04945561D+00      -1.30267546D+00         2.42842164D+00
***
( 4)   -2.45466761D+00       2.70817910D+00         3.65508236D+00
( 5)   -2.45466761D+00      -2.70817910D+00         3.65508236D+00
-------------------------------------------------------------------
FILENAME :  _
/DPACS/ TIME-17:44:26 CPU-00:00:53

/DPACS/ TIME-20:29:46 CPU-00:00:02
LINK EX1A MODELA EXMFS1
TO LINK SYSTEMS
SYSTEM EX1A.SO      REFERRED BY SYSTEM NO.  1 HAS BEEN LOADED
SYSTEM MODELA.SO    REFERRED BY SYSTEM NO.  2 HAS BEEN LOADED
DEFINE ADDER ?   Y
SPECIFY DIMENSION OF ADDER :   2
THE ADDER REFERRED BY SYSTEM NO.  3 HAS BEEN LOADED
DEFINE ADDER ?   Y
SPECIFY DIMENSION OF ADDER :   2
THE ADDER REFERRED BY SYSTEM NO.  4 HAS BEEN LOADED
DEFINE ADDER ?   N
DEFINE INTEGRATER ?   Y
SPECIFY DIMENSION OF INTEGRATER :   2
THE INTEGRATER REFERRED BY SYSTEM NO.  5 HAS BEEN LOADED
DEFINE INTEGRATER ?   Y
SPECIFY DIMENSION OF INTEGRATER :   2
THE INTEGRATER REFERRED BY SYSTEM NO.  6 HAS BEEN LOADED
DEFINE INTEGRATER ?   N
SPECIFY CONNECTION OF SYSTEMS
FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :   1,3
INPUT -MODE OF ( 2, 3) MATRIX GAIN      : F
FILENAME :  F1
TYPE OR REVISE ? _
REPEAT CONNECTING ?   Y
FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :   5,3
INPUT -MODE OF ( 2, 2) MATRIX GAIN      : F
FILENAME :  F2
TYPE OR REVISE ? _
REPEAT CONNECTING ? _
FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :   2,3
INPUT -MODE OF ( 2, 4) MATRIX GAIN      : F
FILENAME :  F3
TYPE OR REVISE ? _
REPEAT CONNECTING ? _
FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :   1,4
INPUT -MODE OF ( 2, 3) MATRIX GAIN      : F
```

```
FILENAME :  EX1C
TYPE OR REVISE ? _
REPEAT CONNECTING ? _
FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :    2,4
INPUT -MODE OF ( 2, 4) MATRIX GAIN      : -F
FILENAME :  MDLC
TYPE OR REVISE ? _
REPEAT CONNECTING ? _

FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :    3,1
INPUT -MODE OF ( 2, 2) MATRIX GAIN      : I
TYPE OR REVISE ? _
REPEAT CONNECTING ? _
FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :    4,5
INPUT -MODE OF ( 2, 2) MATRIX GAIN      : I
TYPE OR REVISE ? _
REPEAT CONNECTING ? _
FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :    6,2
INPUT -MODE OF ( 2, 2) MATRIX GAIN      : I
TYPE OR REVISE ? _
REPEAT CONNECTING ?   N
TYPE OUT THE LINKAGE ?    Y
LINKAGE DATA FOR SYSTEM EXMFS1.SO

------------------------------------
 NO.     SYSTEM            ( M, L, N)
------------------------------------
  1      EX1A.SO           ( 2, 3, 3)
  2      MODELA.SO         ( 2, 4, 4)
  3      ADDER             ( 2, 2, 0)
  4      ADDER             ( 2, 2, 0)
  5      INTEGRATER        ( 2, 2, 2)
***
  6      INTEGRATER        ( 2, 2, 2)
------------------------------------
( 1)  FROM  1-TH OUTPUT TO  3-TH INPUT
--- ( 2, 3) MATRIX GAIN      ---
( 1) -1.27622603D+01 -5.62635857D+00 -1.11021269D+01
( 2)  6.00246031D+00  1.50590249D-01 -4.34998749D+00
FILENAME :  _
( 2)  FROM  5-TH OUTPUT TO  3-TH INPUT
--- ( 2, 2) MATRIX GAIN      ---
( 1) -4.92452657D+00 -8.70339233D+00
( 2)  8.70339233D+00 -4.92452657D+00
FILENAME :  _
( 3)  FROM  2-TH OUTPUT TO  3-TH INPUT
--- ( 2, 4) MATRIX GAIN      ---
( 1)  3.65399418D+00  1.23457030D+00  4.94972703D+00  1.51909614D+00
( 2) -6.12552844D+00 -1.95012162D+00  3.89849093D+00  9.25007239D-01
FILENAME :  _
( 4)  FROM  1-TH OUTPUT TO  4-TH INPUT
--- ( 2, 3) MATRIX GAIN      ---
( 1)  1.00000000D+00  0.0             0.0
( 2)  0.0             1.00000000D+00  1.00000000D+00
FILENAME :  _
( 5)  FROM  2-TH OUTPUT TO  4-TH INPUT
***
```

```
--- ( 2, 4) MATRIX GAIN       ---
( 1) -1.00000000D+00  0.0                  0.0               0.0
( 2)  0.0             0.0             -1.00000000D+00       0.0
FILENAME :
( 6)  FROM  3-TH OUTPUT TO  1-TH INPUT
--- ( 2, 2) MATRIX GAIN       ---
( 1)  1.00000000D+00  0.0
( 2)  0.0             1.00000000D+00
FILENAME :
( 7)  FROM  4-TH OUTPUT TO  5-TH INPUT
--- ( 2, 2) MATRIX GAIN       ---
( 1)  1.00000000D+00  0.0
( 2)  0.0             1.00000000D+00
FILENAME :
( 8)  FROM  6-TH OUTPUT TO  2-TH INPUT
--- ( 2, 2) MATRIX GAIN       ---
( 1)  1.00000000D+00  0.0
( 2)  0.0             1.00000000D+00
FILENAME :
REVISE THE CONNECTION ?  N
SET NEW CONNECTION ?  N
GIVE COMMENT FOR SYSTEMDATA :

/DPACS/ TIME-20:49:09 CPU-00:00:30

/DPACS/ TIME-20:55:50 CPU-00:00:44
SMLT EXMFS1
TO SIMULATE SYSTEM'S RESPONSE
--- FREE RESPONSE ---
ADJUST NOMINATED SAMPLING INTERVAL (  4.07387133D-02) ?
DELT =   0.05
HOW MANY POINTS ? :   100
GIVE INITIAL STATE X(0)
INPUT -MODE OF (11, 1) MATRIX X(0)       : Z
TYPE OR REVISE ?   R
SPECIFY (I,J) :  10,1
X(0)     (10, 1)=  0.0                 REVISE TO   1
END ?
TYPE OR REVISE ?   T
--- (11, 1) MATRIX X(0)       ---
( 1)  0.0
( 2)  0.0
( 3)  0.0
( 4)  0.0
( 5)  0.0
( 6)  0.0
( 7)  0.0
***
( 8)  0.0
( 9)  0.0
(10)  1.00000000D+00
(11)  0.0
```

```
TYPE OR REVISE ? _
STARTING POINT ? (0-  99) :   0
HOW MANY POINTS ? (1- 100) :   100
MINIMUM AND MAXIMUM OF EACH VARIABLE
 Y( 1) :   0.0                1.00528455D+00
 Y( 2) :  -4.36515496D-03    5.87906903D-01
 Y( 3) :  -5.22964433D-01    4.81743072D-03
 Y( 4) :   0.0                1.04321088D+00
 Y( 5) :  -2.78495433D-02    6.44657738D-01
 Y( 6) :   0.0                0.0
 Y( 7) :   0.0                0.0
 Y( 8) :  -7.03587046D+00    3.13951993D-01
 Y( 9) :  -3.00920971D+00    0.0
 Y(10) :  -1.34236200D-01    0.0
 Y(11) :  -9.94571983D-02    1.83146782D-01
 Y(12) :  -3.56300613D-01    0.0
 Y(13) :  -4.45850703D-02    1.25990164D-01
 Y(14) :   1.00000000D+00    1.00000000D+00
 Y(15) :   0.0                0.0
***
HOW MANY VARIABLES ARE PLOTTED ?   0
OUTPUT-MODE OF (15, 100) TIME SERIES OUTPUT   :  F
OUTPUT-FILENAME :    MFSPLOT
/DPACS/ TIME-20:58:00 CPU-00:00:53
```



EXMFS1 SYSTEM RESPONSE (MFS)

最適サーボ問題の求解	DP. Ⅳ. 6
OSV_SN	

機　能

状態空間表現形式システム Σ_s の出力を，下式で示されたモデルの出力に追従させる制御則を得る。

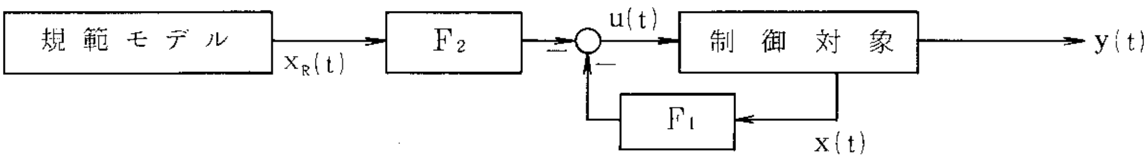
$$\begin{cases} \dfrac{d}{dt}\boldsymbol{x}_R(t) = \boldsymbol{A}_R\boldsymbol{x}_R(t) \\ \boldsymbol{y}_R(t) = \boldsymbol{C}_R\boldsymbol{x}_R(t) \end{cases}$$

制御則は，下式である。

$$\boldsymbol{u}(t) = -\boldsymbol{F}_1\boldsymbol{x}(t) - \boldsymbol{F}_2\boldsymbol{x}_R(t)$$

理論概要

次の線図に対して制御則を得る。



数式モデル Σ_s $\begin{cases} \sigma\boldsymbol{x}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) \end{cases}$

は可制御・可観測である。

規範モデル Σ_R $\begin{cases} \sigma\boldsymbol{x}_R(t) = \boldsymbol{A}_R\boldsymbol{x}_R(t) \\ \boldsymbol{y}_R(t) = \boldsymbol{C}_R\boldsymbol{x}_R(t) \end{cases}$ に対し

拡大系　Σ_E $\begin{cases} \sigma\overline{\boldsymbol{x}}(t) = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A}_R \end{bmatrix}\overline{\boldsymbol{x}}(t) + \begin{bmatrix} \boldsymbol{B} \\ \boldsymbol{0} \end{bmatrix}\boldsymbol{u}(t) \\ \overline{\boldsymbol{y}}(t) = [\,\boldsymbol{C} - \boldsymbol{C}_R\,]\,\overline{\boldsymbol{x}}(t) \end{cases}$

に対して最適レギュレータ問題を解き

$$\overline{\boldsymbol{u}}(t) = \overline{\boldsymbol{F}}\,\overline{\boldsymbol{x}}(t)$$

これを $\overline{\boldsymbol{x}}(t) = {}^t[{}^t\boldsymbol{x}(t),\ {}^t\boldsymbol{x}_R(t)]$ により分解すれば，対応する制御則

$$\boldsymbol{u}(t) = -\boldsymbol{F}_1\boldsymbol{x}(t) - \boldsymbol{F}_2\boldsymbol{x}_R(t)$$

を得る。

実行例

〔1〕 最適サーボ問題の求解の例

```
/DPACS/ TIME-16:06:25 CPU-00:00:01
OSV EX1A2
 TO SOLVE OPTIMAL SERVO PROBLEM
 SPECIFY THE ORDER OF COMMAND GENERATOR (MAX33 ) :  5
 INPUT -MODE OF ( 5, 5) MATRIX  AR         : I
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 5) MATRIX  CR         : I
 TYPE OR REVISE ? _
 GIVE QUADRATIC CRITERION FUNCTION
 INPUT -MODE OF ( 2, 2) MATRIX   Q         : I
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX   R         : I
 TYPE OR REVISE ? _
 THE SOLUTION OF OPTIMAL SERVO PROBLEM FOR SYSTEM EX1A2.SO
 1) COMMAND GENERATOR
 --- ( 5, 5) MATRIX  AR        ---
 ( 1)  1.00000000D+00  0.0             0.0             0.0
 ( 2)  0.0             1.00000000D+00  0.0             0.0
 ( 3)  0.0             0.0             1.00000000D+00  0.0
 ( 4)  0.0             0.0             0.0             1.00000000D+00
 ( 5)  0.0             0.0             0.0             0.0
 ( 1)  0.0
 ( 2)  0.0
 ( 3)  0.0
 ( 4)  0.0
 ( 5)  1.00000000D+00
 FILENAME : _
 --- ( 2, 5) MATRIX  CR        ---
 ( 1)  1.00000000D+00  0.0             0.0             0.0
 ( 2)  0.0             1.00000000D+00  0.0             0.0
 ( 1)  0.0
 ( 2)  0.0
 FILENAME : _
 2) SPECIFIED CRITERION FUNCTION
 --- ( 2, 2) MATRIX   Q        ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  0.0             1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX   R        ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  0.0             1.00000000D+00
 FILENAME : _
 3) OPTIMAL CONTROL LAW
 --- ( 2, 3) MATRIX  F1        ---
 ( 1)  1.33502145D+01  4.32588509D+00  8.07265264D+00
 ( 2) -1.32143208D+00 -5.79117535D-01 -1.88907257D-01
 FILENAME : _
 --- ( 2, 5) MATRIX  F2        ---
 ( 1)  6.58229148D-01 -1.78358542D+00  0.0             0.0
 ( 2)  6.23270071D-01 -1.23200882D+00  0.0             0.0
 ( 1)  0.0
 ( 2)  0.0
 FILENAME : _
 4) POLES OF CLOSED LOOP
```

NO.	REAL	IMAGINARY	ABSOLUTE
(1)	-1.40205713D+00	1.43824560D+00	2.00856033D+00
(2)	-1.40205713D+00	-1.43824560D+00	2.00856033D+00
(3)	-2.49109863D+00	0.0	2.49109863D+00

```
FILENAME : _
/DPACS/ TIME-16:17:53 CPU-00:00:27
```

複合線型（非線型）系の構成	DP. V. 1
$\mathrm{LINK}\begin{bmatrix}\#\\ /\mathrm{N}\end{bmatrix}\begin{bmatrix}\#\\ /\mathrm{R}\end{bmatrix}$ ␣SN1 {␣SN2 {・・・}} ␣OSN	

機　能

複合線型システム，または複合非線型システムの構成を行う。

◆第１スイッチは，複合線型／非線型の種別を示し，／Ｎを付加した場合には，複合非線型システムの構成を示す。

◆第２スイッチは，既に設定している系の結合の修正（／Ｒ）か新規作成かの違いを示す。

理論概要[20)]

◆複合線型システムの構成

次式で表わされるＮ個の線型時不変システムを考える。

$$\begin{cases}\sigma \boldsymbol{x}_i = \boldsymbol{A}_i \boldsymbol{x}_i + \boldsymbol{B}_i \boldsymbol{u}_i \\ \boldsymbol{y}_i = \boldsymbol{C}_i \boldsymbol{x}_i + \boldsymbol{D}_i \boldsymbol{u}_i\end{cases}$$

$$\boldsymbol{x}_i \in \mathbf{R}^{n_i},\ \boldsymbol{u}_i \in \mathbf{R}^{m_i},\ \boldsymbol{y}_i \in \mathbf{R}^{p_i}\ (i=1,\ 2,\ \cdots,\ N)$$

これより，次のシステムを構成する。

$$\begin{cases}\sigma \boldsymbol{x} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} \\ \boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}\boldsymbol{u}\end{cases}$$

ここで $\boldsymbol{x} = {}^t[{}^t\boldsymbol{x}_1,\ {}^t\boldsymbol{x}_2\ \cdots,\ {}^t\boldsymbol{x}_N] \in \mathbf{R}^n,\quad n=\Sigma n_i$

$\boldsymbol{u} = {}^t[{}^t\boldsymbol{u}_1,\ {}^t\boldsymbol{u}_2\ \cdots,\ {}^t\boldsymbol{u}_N] \in \mathbf{R}^m,\quad m=\Sigma m_i$

$\boldsymbol{y} = {}^t[{}^t\boldsymbol{y}_1,\ {}^t\boldsymbol{y}_2\ \cdots,\ {}^t\boldsymbol{y}_N] \in \mathbf{R}^p,\quad p=\Sigma p_i$

$\boldsymbol{A} = \mathrm{diag}\{A_i\},\ i=1,\ \cdots,\ N$

$\boldsymbol{B} = \mathrm{diag}\{B_i\},\ i=1,\ \cdots,\ N$

$\boldsymbol{C} = \mathrm{diag}\{C_i\},\ i=1,\ \cdots,\ N$

$\boldsymbol{D} = \mathrm{diag}\{D_i\},\ i=1,\ \cdots,\ N$

i 番目のシステムの出力を j 番目のシステムの入力にゲイン $\boldsymbol{F}_{ji} \in \mathbf{R}^{m_j \times p_i}$ を介して結合することは $\boldsymbol{u} = \boldsymbol{F}\boldsymbol{y} + \boldsymbol{v}$ の feedbackすることと等しい。

$$\boldsymbol{F} = \begin{array}{c} \overbrace{\qquad}^{\sum_{k=1}^{i-1} p_k} \qquad\qquad \\ \left[\begin{array}{c:c:c} 0 & 0 & 0 \\ \hdashline 0 & F_{ji} & 0 \\ \hdashline 0 & 0 & 0 \end{array}\right] \end{array} \begin{array}{l} \}\sum_{k=1}^{j-1} m_k \\ \\ \\ \end{array} \in \mathbf{R}^{m \times p}\text{；接続関係}$$

このとき，閉ループ系は次のように表わされる。

$$\begin{cases} \sigma \boldsymbol{x} = \bar{\boldsymbol{A}} \boldsymbol{x} + \bar{\boldsymbol{B}} \boldsymbol{v} \\ \boldsymbol{y} = \bar{\boldsymbol{C}} \boldsymbol{x} + \bar{\boldsymbol{D}} \boldsymbol{v} \end{cases}$$

$$\bar{\boldsymbol{A}} = \boldsymbol{A} + \boldsymbol{B}\boldsymbol{F}(\boldsymbol{I} - \boldsymbol{D}\boldsymbol{F})^{-1}\boldsymbol{C}, \quad \bar{\boldsymbol{B}} = \boldsymbol{B} + \boldsymbol{B}\boldsymbol{F}(\boldsymbol{I} - \boldsymbol{D}\boldsymbol{F})^{-1}\boldsymbol{D}$$

$$\bar{\boldsymbol{C}} = (\boldsymbol{I} - \boldsymbol{D}\boldsymbol{F})^{-1}\boldsymbol{C} \quad , \quad \bar{\boldsymbol{D}} = (\boldsymbol{I} - \boldsymbol{D}\boldsymbol{F})^{-1}\boldsymbol{D}$$

$\boldsymbol{v} \to \boldsymbol{u}$ にすることで複合線型系が構成できる。

◆複合非線型システムの構成

非線型系は一般に N_1 個の線型系

$$\begin{cases} \sigma \boldsymbol{x}_i(t) = \boldsymbol{A}_i \boldsymbol{x}_i(t) + \boldsymbol{B}_i \boldsymbol{u}_{1i}(t) \\ \boldsymbol{y}_{1i}(t) = \boldsymbol{C}_i \boldsymbol{x}_i(t) + \boldsymbol{D}_i \boldsymbol{u}_{1i}(t) \end{cases}$$

$$\boldsymbol{x}_i(t) \in \mathbf{R}^{n_i}, \ \boldsymbol{u}_{1i} \in \mathbf{R}^{m_{1i}}, \ \boldsymbol{y}_{1i} \in \mathbf{R}^{p_{1i}} \quad (i = 1, \sim, N_1)$$

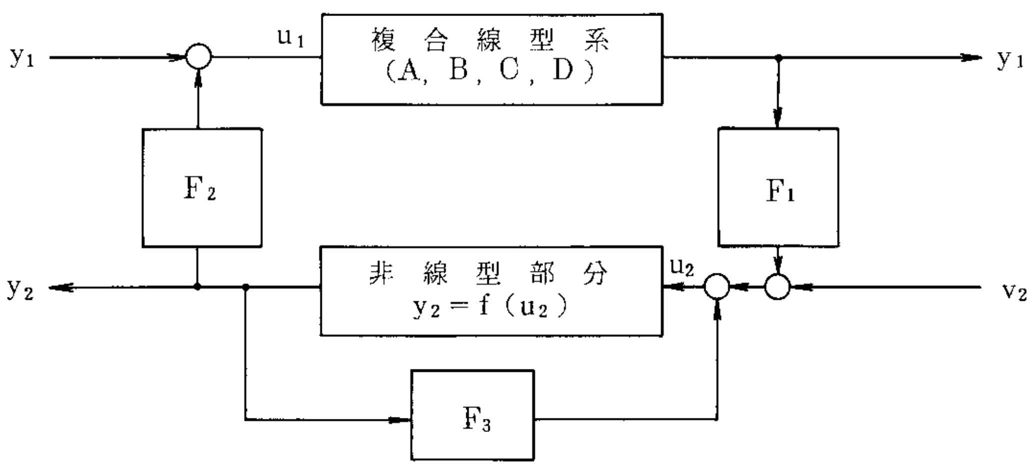
と N_2 個の非線型要素

$$\boldsymbol{y}_{2i}(t) = \boldsymbol{f}_i(\boldsymbol{u}_{2i}(t))$$

$$\boldsymbol{u}_{2i} \in \mathbf{R}^{m_{2i}}, \ \boldsymbol{y}_{2i} \in \mathbf{R}^{p_{2i}} \ (i = 1, \sim, N_2)$$

これらの入出力を結ぶ結合ゲインとに分解できる。

これらから，m_1 入力，p_1 出力，n 次元の複合線型系と，m_2 入力 p_2 出力の非線型部分，および結合ゲインとに構成しなおす。



$$\begin{cases} \sigma \boldsymbol{x}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}_1(t) & \boldsymbol{u}_1 \in \mathbf{R}^{m_1}, \quad \boldsymbol{x} \in \mathbf{R}^n \\ \boldsymbol{y}_1(t) = \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}_1(t) & \boldsymbol{y}_1 \in \mathbf{R}^{p_1} \end{cases}$$

$$\{ \quad \boldsymbol{y}_2(t) = \boldsymbol{f}(\boldsymbol{u}_2(t)) \qquad \boldsymbol{u}_2 \in \mathbf{R}^{m_2}, \quad \boldsymbol{y}_2 \in \mathbf{R}^{p_2}$$

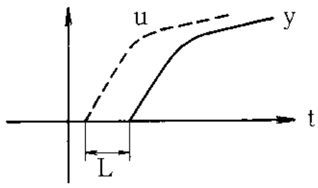
結合ゲイン $\boldsymbol{F}_1 \in \mathbf{R}^{m_2 \times p_1}$, $\boldsymbol{F}_2 \in \mathbf{R}^{m_1 \times p_2}$, $\boldsymbol{F}_3 \in \mathbf{R}^{m_2 \times p_2}$

$$m_1 = \sum_{i=1}^{N_1} m_{1i}, \quad p_1 = \sum_{i=1}^{N_1} p_{1i}, \quad n = \sum_{i=1}^{N_1} n_i, \quad m_2 = \sum_{i=1}^{N_2} m_{2i}, \quad p_2 = \sum_{i=1}^{N_2} p_{2i}$$

ここで $\boldsymbol{F}_1 = \{\boldsymbol{F}_{1ji}\}$, $\boldsymbol{F}_{1ji} \in \mathbf{R}^{m_{2j} \times p_{1i}}$
$\boldsymbol{F}_2 = \{\boldsymbol{F}_{2ji}\}$, $\boldsymbol{F}_{2ji} \in \mathbf{R}^{p_{2j} \times m_{1i}}$
$\boldsymbol{F}_3 = \{\boldsymbol{F}_{3ji}\}$, $\boldsymbol{F}_{3ji} \in \mathbf{R}^{p_{2j} \times m_{2i}}$

ただし，非線型部分内でループを構成している場合は，ループ内の非線型要素の出力を求めるのに非線型代数方程式を解かねばならず，通常不可能である。この場合，判定は，アソシエーションmatrix $= \boldsymbol{0}$か否かで行う。もしループを形成するような非線型系を構成する場合そのループを切るような位置に$\boldsymbol{y}(t) = \boldsymbol{u}(t)$の線型システムを作って対処する。以下に，使用できる非線型要素の種類を掲げる。

(1) Delay ; 遅延要素 $y(t) = u(t - L)$ パラメータ：L

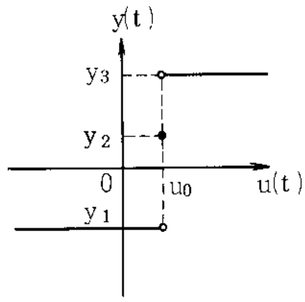


(2) Polynomial Function ; 多項式関数 $y(t) = f(u(t))$
$f(x) = p_0 x^n + p_1 x^{n-1} + \cdots + p_n$
パラメータ：n, p_0, p_1, $\cdots$, p_n

(3) Function Switch ; 関数スイッチ

$$y(t) = \begin{cases} y_1 & (u(t) < u_0) \\ y_2 & (u(t) = u_0) \\ y_3 & (u(t) > u_0) \end{cases}$$

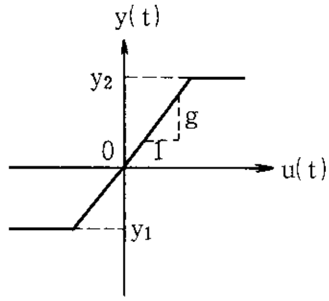
パラメータ：u_0, y_1, y_2, y_3



(4) Limiter;
飽和関数

$$y(t)=\begin{cases} y_1 & (gu(t)<y_1) \\ gu(t) & (y_1 \leqq gu(t) \leqq y_2) \\ y_2 & (gu(t)>y_2) \end{cases}$$

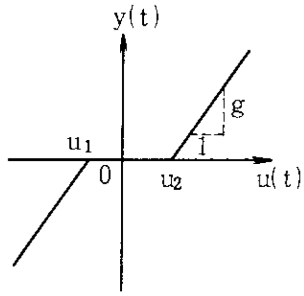
パラメータ：y_1, y_2, g



(5) Dead Band;
不感帯要素

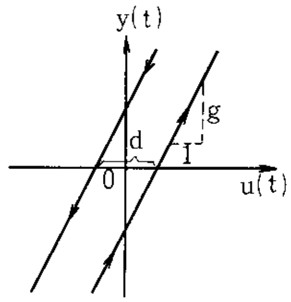
$$y(t)=\begin{cases} g(u(t)-u_1) & (u(t)<u_1) \\ 0 & (u_1 \leqq u(t) \leqq u_2) \\ g(u(t)-u_2) & (u(t)>u_2) \end{cases}$$

パラメータ：u_1, u_2, g



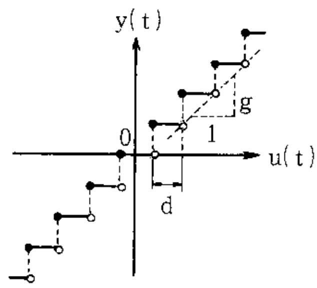
(6) Backlash;
バックラッシュ

パラメータ：d, g



(7) Quantifier;
量子化関数

パラメータ：d, g



(8) Sin; Sine関数

$y(t)=g\cdot \sin(u(t)+p)$

パラメータ：p, g

(9) Multiplier; 乗算器　$y(t)=u_1(t)\cdot u_2(t)$

(10) Divider; 除算器　$y(t)=u_1(t)/u_2(t)$

(11) NOT; 否定素子

$$y(t)=\begin{cases} 0 & (u(t)>0) \\ 1 & (u(t)\leqq 0) \end{cases}$$

(12) AND; AND素子
$$y(t)=\begin{cases}0 & (u_1(t)\leqq 0 \text{ または } u_2(t)\leqq 0)\\ 1 & (u_1(t)>0,\ u_2(t)>0)\end{cases}$$

(13) NAND; NAND素子
$$y(t)=\begin{cases}0 & (u_1(t)>0,\ u_2(t)>0)\\ 1 & (u_1(t)\leqq 0 \text{ または } u_2(t)\leqq 0)\end{cases}$$

(14) OR; OR素子
$$y(t)=\begin{cases}0 & (u_1(t)\leqq 0,\ u_2(t)\leqq 0)\\ 1 & (u_1(t)>0 \text{ または } u_2(t)>0)\end{cases}$$

(15) NOR; NOR素子
$$y(t)=\begin{cases}0 & (u_1(t)>0 \text{ または } u_2(t)>0)\\ 1 & (u_1(t)\leqq 0,\ u_2(t)\leqq 0)\end{cases}$$

(16) EX−OR;
Exclusive−OR素子
$$y(t)=\begin{cases}0 & (u_1(t)\leqq 0,\ u_2(t)\leqq 0 \text{ または } u_1(t)>0,\ u_2(t)>0)\\ 1 & (u_1(t)\leqq 0,\ u_2(t)>0 \text{ または } u_1(t)>0,\ u_2(t)\leqq 0)\end{cases}$$

(17) IN−OR;
Inclusive−OR素子
$$y(t)=\begin{cases}0 & (u_1(t)\leqq 0,\ u_2(t)>0 \text{ または } u_1(t)>0,\ u_2(t)\leqq 0)\\ 1 & (u_1(t)\leqq 0,\ u_2(t)\leqq 0 \text{ または } u_1(t)>0\ \ u_2(t)>0)\end{cases}$$

(18) FLIP FLOP
R−S型フリップフロップ

$u_1(t)>0,\ u_2(t)\leq 0$ → $y_1(t)=0,\ y_2(t)=1$

$u_1(t)\leq 0,\ u_2(t)>0$ → $y_1(t)=1,\ y_2(t)=0$

連動

実 行 例

〔1〕 複合線型システムの構成例

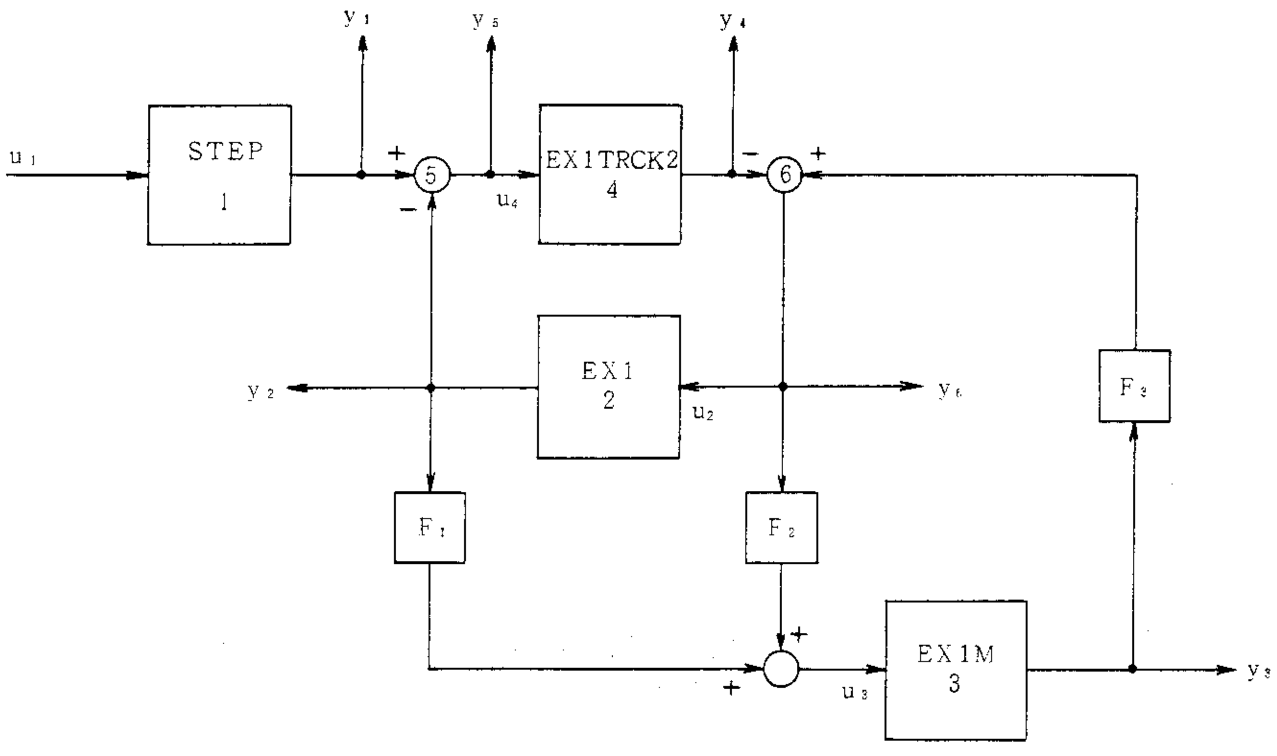
```
 /DPACS/ TIME-16:06:26 CPU-00:00:01
LINK STEP EX1 EX1M EX1TRCK2 EX1T
 TO LINK SYSTEMS
 SYSTEM STEP.SO     REFERRED BY SYSTEM NO.  1 HAS BEEN  LOADED
 SYSTEM EX1.SO      REFERRED BY SYSTEM NO.  2 HAS BEEN  LOADED
 SYSTEM EX1M.SO     REFERRED BY SYSTEM NO.  3 HAS BEEN  LOADED
 SYSTEM EX1TRCK2.SO REFERRED BY SYSTEM NO.  4 HAS BEEN  LOADED
 DEFINE ADDER ? _
 SPECIFY DIMENSION OF ADDER :  2
 THE ADDER REFERRED BY SYSTEM NO.  5 HAS BEEN LOADED
 DEFINE ADDER ? _
 SPECIFY DIMENSION OF ADDER :  2
 THE ADDER REFERRED BY SYSTEM NO.  6 HAS BEEN LOADED
 DEFINE ADDER ?  N
 DEFINE INTEGRATER ?  N
 SPECIFY CONNECTION OF SYSTEMS
 FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :   1,5
 INPUT -MODE OF ( 2, 2) MATRIX GAIN     : I
 TYPE OR REVISE ? _
 REPEAT CONNECTING ? _
 FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :   5,4
 INPUT -MODE OF ( 2, 2) MATRIX GAIN     : I
 TYPE OR REVISE ? _
 REPEAT CONNECTING ? _
 FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :   4,6
 INPUT -MODE OF ( 2, 2) MATRIX GAIN     : -I
 TYPE OR REVISE ? _
 REPEAT CONNECTING ? _
 FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :   6,2
 INPUT -MODE OF ( 2, 2) MATRIX GAIN     : I
 TYPE OR REVISE ? _
 REPEAT CONNECTING ? _
 FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :   2,3
 INPUT -MODE OF ( 4, 2) MATRIX GAIN     : I
 TYPE OR REVISE ? _
 REPEAT CONNECTING ? _
 FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :   6,3
 INPUT -MODE OF ( 4, 2) MATRIX GAIN     : _
 ( 1) ROW
0,0
 ( 2) ROW
0,0
 ( 3) ROW
 ***
1,0
 ( 4) ROW
0,1
 TYPE OR REVISE ? _
 REPEAT CONNECTING ? _
 FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :   3,6
 INPUT -MODE OF ( 2, 3) MATRIX GAIN     : -F
 FILENAME :  K1
 TYPE OR REVISE ? _
 REPEAT CONNECTING ? _
 FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :   2,5
 INPUT -MODE OF ( 2, 2) MATRIX GAIN     : -I
 TYPE OR REVISE ? _
 REPEAT CONNECTING ?  N
```

```
TYPE OUT THE LINKAGE ? _
LINKAGE DATA FOR SYSTEM EX1T.SO

-------------------------------------
 NO.     SYSTEM              ( M, L, N)
-------------------------------------
   1     STEP.SO             ( 2, 2, 2)
   2     EX1.SO              ( 2, 2, 3)
   3     EX1M.SO             ( 4, 3, 1)
***
   4     EX1TRCK2.SO         ( 2, 2, 2)
   5     ADDER               ( 2, 2, 0)
   6     ADDER               ( 2, 2, 0)
-------------------------------------
( 1)  FROM  1-TH OUTPUT TO  5-TH INPUT
--- ( 2, 2) MATRIX GAIN      ---
( 1)  1.00000000D+00  0.0
( 2)  0.0             1.00000000D+00
FILENAME : _
( 2)  FROM  5-TH OUTPUT TO  4-TH INPUT
--- ( 2, 2) MATRIX GAIN      ---
( 1)  1.00000000D+00  0.0
( 2)  0.0             1.00000000D+00
FILENAME : _
( 3)  FROM  4-TH OUTPUT TO  6-TH INPUT
--- ( 2, 2) MATRIX GAIN      ---
( 1) -1.00000000D+00  0.0
( 2)  0.0            -1.00000000D+00
FILENAME : _
( 4)  FROM  6-TH OUTPUT TO  2-TH INPUT
--- ( 2, 2) MATRIX GAIN      ---
( 1)  1.00000000D+00  0.0
( 2)  0.0             1.00000000D+00
***
FILENAME : _
( 5)  FROM  2-TH OUTPUT TO  3-TH INPUT
--- ( 4, 2) MATRIX GAIN      ---
( 1)  1.00000000D+00  0.0
( 2)  0.0             1.00000000D+00
( 3)  0.0             0.0
( 4)  0.0             0.0
FILENAME : _
( 6)  FROM  6-TH OUTPUT TO  3-TH INPUT
--- ( 4, 2) MATRIX GAIN      ---
( 1)  0.0             0.0
( 2)  0.0             0.0
( 3)  1.00000000D+00  0.0
( 4)  0.0             1.00000000D+00
FILENAME : _
( 7)  FROM  3-TH OUTPUT TO  6-TH INPUT
--- ( 2, 3) MATRIX GAIN      ---
( 1)  3.74727472D+01 -2.10517739D+01 -3.54640516D+01
( 2)  8.68953283D+01  6.63949631D+00 -8.00400694D+00
FILENAME : _
( 8)  FROM  2-TH OUTPUT TO  5-TH INPUT
--- ( 2, 2) MATRIX GAIN      ---
( 1) -1.00000000D+00  0.0
***
( 2)  0.0            -1.00000000D+00
FILENAME : _
REVISE THE CONNECTION ?  N
SET NEW CONNECTION ?  N
GIVE COMMENT FOR SYSTEMDATA :

/DPACS/ TIME-16:17:57 CPU-00:00:27
```

```
/DPACS/ TIME-16:06:28 CPU-00:00:01
TYPE/S EX1T
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1T.SO      (14,13, 8)

 --- ( 8, 8) MATRIX   A       ---
 ( 1)  0.0                 0.0                 0.0                 0.0
 ( 2)  0.0                 0.0                 0.0                 0.0
 ( 3)  0.0                 0.0                 0.0                 1.00000000D+00
 ( 4)  0.0                 0.0                -1.37541199D+02 -2.34468238D+01
 ( 5)  0.0                 0.0                 9.20840567D+01 -5.05971219D+00
 ( 6)  0.0                 0.0                 1.28751895D+02  1.59909582D+01
 ( 7)  1.00000000D+00      0.0                -1.00000000D+00  0.0
 ( 8)  0.0                 1.00000000D+00      0.0            -1.00000000D+00
 ( 1)  0.0                 0.0                 0.0                 0.0
 ( 2)  0.0                 0.0                 0.0                 0.0
 ( 3)  0.0                 0.0                 0.0                 0.0
 ( 4) -2.04468238D+01  1.48747289D+01 -5.11021327D+02 -4.88730195D+02
 ( 5) -4.05971219D+00 -1.46435033D+01  3.02154570D+02  9.32878127D+01
 ( 6)  1.59909582D+01 -1.71734454D+01  4.69025550D+02  4.09220606D+02
***
 ( 7)  0.0                 0.0                 0.0                 0.0
 ( 8) -1.00000000D+00  0.0                 0.0                 0.0
FILENAME : _
--- ( 8,14) MATRIX   B       ---
( 1)  0.0                 0.0                 0.0                 0.0
( 2)  0.0                 0.0                 0.0                 0.0
( 3)  0.0                 0.0                 0.0                 0.0
( 4)  0.0                 0.0                 1.00000000D+00 -2.00000000D+00
( 5)  0.0                 0.0                 0.0                 1.00000000D+00
( 6)  0.0                 0.0                 0.0                 0.0
( 7)  0.0                 0.0                 0.0                 0.0
( 8)  0.0                 0.0                 0.0                 0.0
( 1)  0.0                 0.0                 0.0                 0.0
( 2)  0.0                 0.0                 0.0                 0.0
( 3)  0.0                 0.0                 0.0                 0.0
( 4) -1.38541199D+02 -2.24468238D+01  0.0                 0.0
( 5)  8.90840567D+01 -5.05971219D+00  0.0                 0.0
( 6)  1.28751895D+02  1.59909582D+01 -7.98935084D-01  1.79893508D+00
( 7)  0.0                 0.0                 0.0                 0.0
( 8)  0.0                 0.0                 0.0                 0.0
( 1)  0.0                 0.0                 0.0                 0.0
( 2)  0.0                 0.0                 0.0                 0.0
( 3)  0.0                 0.0                 0.0                 0.0
***
( 4)  0.0                 0.0                 0.0                 0.0
( 5)  0.0                 0.0                 0.0                 0.0
( 6)  0.0                 0.0                 0.0                 0.0
( 7)  1.00000000D+00      0.0                 1.00000000D+00      0.0
( 8)  0.0                 1.00000000D+00      0.0                 1.00000000D+00
( 1)  0.0                 0.0
( 2)  0.0                 0.0
( 3)  0.0                 0.0
( 4)  1.00000000D+00 -2.00000000D+00
( 5)  0.0                 1.00000000D+00
( 6) -7.98935084D-01  1.79893508D+00
( 7)  0.0                 0.0
( 8)  0.0                 0.0
FILENAME :_
```

```
--- (13, 8) MATRIX   C        ---
( 1)  1.00000000D+00  0.0             0.0             0.0
( 2)  0.0             1.00000000D+00  0.0             0.0
( 3)  0.0             0.0             1.00000000D+00  0.0
( 4)  0.0             0.0             0.0             1.00000000D+00
( 5)  0.0             0.0             1.00000000D+00  0.0
( 6)  0.0             0.0             1.49467542D-01  2.01064916D-01
( 7)  0.0             0.0            -1.49467542D-01  7.98935084D-01
( 8)  0.0             0.0             0.0             0.0
***
( 9)  0.0             0.0             0.0             0.0
(10)  1.00000000D+00  0.0            -1.00000000D+00  0.0
(11)  0.0             1.00000000D+00  0.0            -1.00000000D+00
(12)  0.0             0.0             3.96269149D+01 -3.25662482D+01
(13)  0.0             0.0             8.90840567D+01 -5.05971219D+00
( 1)  0.0             0.0             0.0             0.0
( 2)  0.0             0.0             0.0             0.0
( 3)  0.0             0.0             0.0             0.0
( 4)  1.00000000D+00  0.0             0.0             0.0
( 5)  0.0             0.0             0.0             0.0
( 6)  2.01064916D-01 -1.00000000D+00  0.0             0.0
( 7)  7.98935084D-01  1.00000000D+00  0.0             0.0
( 8)  0.0             0.0            -9.32878127D+01  3.02154570D+02
( 9)  0.0             0.0            -3.02154570D+02 -9.32878127D+01
(10)  0.0             0.0             0.0             0.0
(11) -1.00000000D+00  0.0             0.0             0.0
(12) -3.25662482D+01 -1.44122776D+01  9.32878127D+01 -3.02154570D+02
(13) -5.05971219D+00 -1.46435033D+01  3.02154570D+02  9.32878127D+01
FILENAME :_
--- (13,14) MATRIX   D        ---
( 1)  0.0             0.0             0.0             0.0
( 2)  0.0             0.0             0.0             0.0
( 3)  0.0             0.0             0.0             0.0
***
( 4)  0.0             0.0             0.0             0.0
( 5)  0.0             0.0             0.0             0.0
( 6)  0.0             0.0             0.0             0.0
( 7)  0.0             0.0             0.0             0.0
( 8)  0.0             0.0             0.0             0.0
( 9)  0.0             0.0             0.0             0.0
(10)  0.0             0.0             0.0             0.0
(11)  0.0             0.0             0.0             0.0
(12)  0.0             0.0             0.0             0.0
(13)  0.0             0.0             0.0             0.0
( 1)  0.0             0.0             0.0             0.0
( 2)  0.0             0.0             0.0             0.0
( 3)  0.0             0.0             0.0             0.0
( 4)  0.0             0.0             0.0             0.0
( 5)  1.00000000D+00  0.0             0.0             0.0
( 6)  1.49467542D-01  2.01064916D-01  0.0             0.0
( 7) -1.49467542D-01  7.98935084D-01  0.0             0.0
( 8)  0.0             0.0             0.0             0.0
( 9)  0.0             0.0             0.0             0.0
(10)  0.0             0.0             0.0             0.0
(11)  0.0             0.0             0.0             0.0
(12)  3.96269149D+01 -3.25662482D+01  0.0             0.0
(13)  8.90840567D+01 -5.05971219D+00  0.0             0.0
```

```
( 1)  0.0             0.0             0.0               0.0
( 2)  0.0             0.0             0.0               0.0
( 3)  0.0             0.0             0.0               0.0
( 4)  0.0             0.0             0.0               0.0
( 5)  0.0             0.0             0.0               0.0
( 6)  0.0             0.0             0.0               0.0
( 7)  0.0             0.0             0.0               0.0
( 8)  0.0             0.0             0.0               0.0
( 9)  0.0             0.0             0.0               0.0
(10)  0.0             0.0             1.000000000D+00   0.0
(11)  0.0             0.0             0.0               1.000000000D+00
(12)  0.0             0.0             0.0               0.0
(13)  0.0             0.0             0.0               0.0
( 1)  0.0             0.0
( 2)  0.0             0.0
( 3)  0.0             0.0
( 4)  0.0             0.0
( 5)  0.0             0.0
( 6)  0.0             0.0
( 7)  0.0             0.0
( 8)  0.0             0.0
( 9)  0.0             0.0
(10)  0.0             0.0
***
(11)  0.0             0.0
(12)  1.000000000D+00 0.0
(13)  0.0             1.000000000D+00
FILENAME : _
/DPACS/ TIME-16:17:59 CPU-00:00:27
```



複合線型システムＥＸ１Ｔの構成図

〔2〕 複合非線型システムの構成例

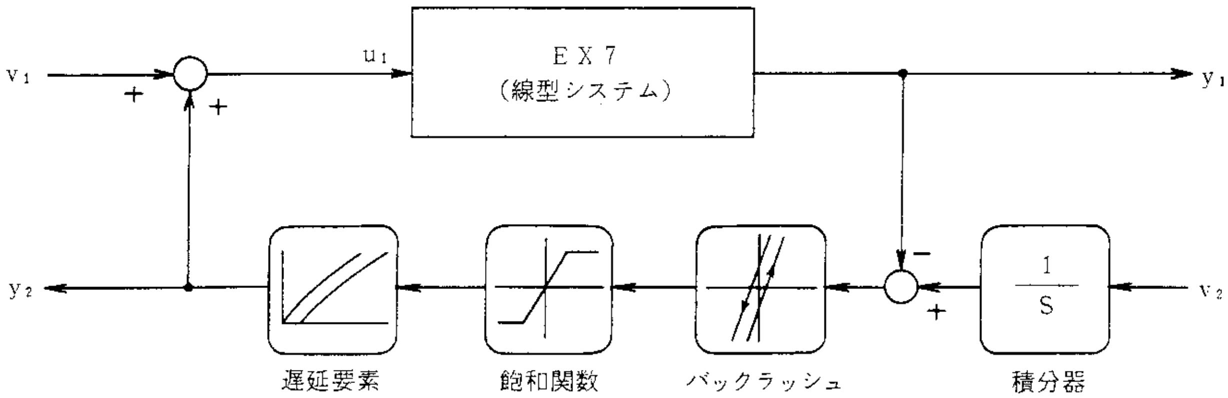
```
/DPACS/ TIME-16:06:29 CPU-00:00:01
LINK/N EX7 EX7N
 TO LINK SYSTEMS
 --- NONLINEAR LINKAGE ---
 SYSTEM EX7.SO       REFERRED BY SYSTEM NO.  1 HAS BEEN  LOADED
 DEFINE ADDER ?  N
 DEFINE INTEGRATER ? _
 SPECIFY DIMENSION OF INTEGRATER :   1
 THE INTEGRATER REFERRED BY SYSTEM NO.  2 HAS BEEN LOADED
 DEFINE INTEGRATER ?  N
  (1)DELAY         (2)POL. FUNC.  (3)FUNC. SW.   (4)LIMITER     (5)DEAD BAND
  (6)BACKLASH      (7)QUANTIFIER  (8)SIN         (9)MULTIPLIER (10)DIVIDER
 (11)NOT          (12)AND        (13)NAND       (14)OR         (15)NOR
 (16)EX-OR        (17)IN-OR      (18)FLIP FLOP
 WHICH ELEMENT ? :  1
 SPECIFY DELAY TIME :   0.01
 THE DELAY       ELEMENT REFERRED BY SYSTEM NO.  3 HAS BEEN LOADED
 DEFINE OTHER ELEMENT ? _
  (1)DELAY         (2)POL. FUNC.  (3)FUNC. SW.   (4)LIMITER     (5)DEAD BAND
  (6)BACKLASH      (7)QUANTIFIER  (8)SIN         (9)MULTIPLIER (10)DIVIDER
 ***
 (11)NOT          (12)AND        (13)NAND       (14)OR         (15)NOR
 (16)EX-OR        (17)IN-OR      (18)FLIP FLOP
 WHICH ELEMENT ? :  6
 SPECIFY (D,G) :   2,1.0D10
 THE BACKLASH    ELEMENT REFERRED BY SYSTEM NO.  4 HAS BEEN LOADED
 DEFINE OTHER ELEMENT ? _
  (1)DELAY         (2)POL. FUNC.  (3)FUNC. SW.   (4)LIMITER     (5)DEAD BAND
  (6)BACKLASH      (7)QUANTIFIER  (8)SIN         (9)MULTIPLIER (10)DIVIDER
 (11)NOT          (12)AND        (13)NAND       (14)OR         (15)NOR
 (16)EX-OR        (17)IN-OR      (18)FLIP FLOP
 WHICH ELEMENT ? :  4
 SPECIFY (Y1,Y2,G) :   -1,1,1
 THE LIMITTER    ELEMENT REFERRED BY SYSTEM NO.  5 HAS BEEN LOADED
 DEFINE OTHER ELEMENT ?  N
 SPECIFY CONNECTION OF SYSTEMS
 FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :   2,4
 INPUT -MODE OF ( 1, 1) MATRIX GAIN      : I
 TYPE OR REVISE ? _
 REPEAT CONNECTING ? _
 FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :   4,5
 INPUT -MODE OF ( 1, 1) MATRIX GAIN      : I
 TYPE OR REVISE ? _
 REPEAT CONNECTING ? _

 FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :   5,3
 INPUT -MODE OF ( 1, 1) MATRIX GAIN      : I
 TYPE OR REVISE ? _
 REPEAT CONNECTING ? _
 FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :   3,1
 INPUT -MODE OF ( 1, 1) MATRIX GAIN      : I
 TYPE OR REVISE ? _
 REPEAT CONNECTING ? _
 FROM WHICH OUTPUT(I) TO WHICH INPUT(J) ? (I,J) :   1,4
 INPUT -MODE OF ( 1, 1) MATRIX GAIN      : I
 TYPE OR REVISE ? _
 REPEAT CONNECTING ?  N
 TYPE OUT THE LINKAGE ? _
 LINKAGE DATA FOR SYSTEM EX7N.SO
```

```
------------------------------------
 NO.     SYSTEM             ( M, L, N)
------------------------------------
  1      EX7.SO             ( 1, 1, 1)
  2      INTEGRATER         ( 1, 1, 1)
  3      DELAY              ( 1, 1, 0)
            L =  1.00000000D-02
  4      BACKLASH           ( 1, 1, 0)
***
            D =  2.00000000D+00
            G =  1.00000000D+10
  5      LIMITTER           ( 1, 1, 0)
            Y1= -1.00000000D+00
            Y2=  1.00000000D+00
            G =  1.00000000D+00
------------------------------------
( 1)   FROM  2-TH OUTPUT TO  4-TH INPUT
--- ( 1, 1) MATRIX GAIN      ---
( 1)  1.00000000D+00
FILENAME : _
( 2)   FROM  4-TH OUTPUT TO  5-TH INPUT
--- ( 1, 1) MATRIX GAIN      ---
( 1)  1.00000000D+00
FILENAME : _
( 3)   FROM  5-TH OUTPUT TO  3-TH INPUT
--- ( 1, 1) MATRIX GAIN      ---
( 1)  1.00000000D+00
FILENAME : _
( 4)   FROM  3-TH OUTPUT TO  1-TH INPUT
--- ( 1, 1) MATRIX GAIN      ---
( 1)  1.00000000D+00
FILENAME : _

( 5)   FROM  1-TH OUTPUT TO  4-TH INPUT
--- ( 1, 1) MATRIX GAIN      ---
( 1)  1.00000000D+00
FILENAME : _
REVISE THE CONNECTION ? _
WHICH CONNECTION ? : 5
GIVE NEW CONNECTION GAIN
INPUT -MODE OF ( 1, 1) MATRIX GAIN      : -I
TYPE OR REVISE ? _
REVISE THE CONNECTION ?   N
SET NEW CONNECTION ?   N
TYPE OUT THE LINKAGE ?   Y
```

```
LINKAGE DATA FOR SYSTEM EX7N.SO

------------------------------------
 NO.     SYSTEM              ( M, L, N)
------------------------------------
  1      EX7.SO              ( 1, 1, 1)
  2      INTEGRATER          ( 1, 1, 1)
  3      DELAY               ( 1, 1, 0)
            L =  1.00000000D-02
***
  4      BACKLASH            ( 1, 1, 0)
            D =  2.00000000D+00
            G =  1.00000000D+10
  5      LIMITTER            ( 1, 1, 0)
            Y1= -1.00000000D+00
            Y2=  1.00000000D+00
            G =  1.00000000D+00
------------------------------------
( 1)  FROM  2-TH OUTPUT TO  4-TH INPUT
--- ( 1, 1) MATRIX GAIN      ---
( 1)  1.00000000D+00
FILENAME : _
( 2)  FROM  4-TH OUTPUT TO  5-TH INPUT
--- ( 1, 1) MATRIX GAIN      ---
( 1)  1.00000000D+00
FILENAME : _
( 3)  FROM  5-TH OUTPUT TO  3-TH INPUT
--- ( 1, 1) MATRIX GAIN      ---
( 1)  1.00000000D+00
FILENAME : _
( 4)  FROM  3-TH OUTPUT TO  1-TH INPUT
--- ( 1, 1) MATRIX GAIN      ---
( 1)  1.00000000D+00
***
FILENAME : _
( 5)  FROM  1-TH OUTPUT TO  4-TH INPUT
--- ( 1, 1) MATRIX GAIN      ---
( 1) -1.00000000D+00
FILENAME : _
REVISE THE CONNECTION ?  N
SET NEW CONNECTION ?  N
GIVE COMMENT FOR SYSTEMDATA :

/DPACS/ TIME-16:18:00 CPU-00:00:27
```

〔3〕 複合非線型システムの結合修正例

```
/DPACS/ TIME-16:06:31 CPU-00:00:01
LINK/N/R EX7N EX7NN
 TO LINK SYSTEMS
 --- NONLINEAR LINKAGE ---
 TYPE OUT THE LINKAGE ?   N
 REVISE THE CONNECTION ?
 WHICH CONNECTION ? :   5
 GIVE NEW CONNECTION GAIN
 INPUT -MODE OF ( 1, 1) MATRIX GAIN      : -I
 TYPE OR REVISE ? _
 REVISE THE CONNECTION ?   N
 SET NEW CONNECTION ?   N
 TYPE OUT THE LINKAGE ?   N
 REVISE THE CONNECTION ?   N
 SET NEW CONNECTION ?   N
 GIVE COMMENT FOR SYSTEMDATA :
LINK/N/R EX7
 /DPACS/ TIME-16:18:02 CPU-00:00:27
```



複合非線型システムEX7NNの構成図

閉ループ系の構成	DP. V. 2
CLPS ([／X ／Y]) ␣SN␣OSN	

機　能

状態空間表現形式システム Σ_s（SN）に対して，状態フィードバック則または出力フィードバック則を施した場合の閉ループ系のシステム（OSN）を得る。

◆第1スイッチが／Xの場合は，状態フィードバックを行う。

◆第1スイッチが／Yの場合は，出力フィードバックを行う。

理論概要

システム Σ_s $\begin{cases} \sigma \boldsymbol{x} = \boldsymbol{Ax} + \boldsymbol{Bu} \\ \boldsymbol{y} = \boldsymbol{Cx} + \boldsymbol{Du} \end{cases}$

／X；状態フィードバック則により，次式を施す。

$$\boldsymbol{u}(t) = -\boldsymbol{Fx}(t) + \boldsymbol{Gv}(t)$$

$$\begin{cases} \sigma \boldsymbol{x}(t) = (\boldsymbol{A} - \boldsymbol{BF})\,\boldsymbol{x}(t) + \boldsymbol{BGv}(t) \\ \boldsymbol{y}(t) = (\boldsymbol{C} - \boldsymbol{DF})\,\boldsymbol{x}(t) + \boldsymbol{DGv}(t) \end{cases}$$

なる新しい系を得る。

／Y；出力フィードバック則により，次式を施す。

$$\boldsymbol{u}(t) = -\boldsymbol{Fy}(t) + \boldsymbol{Gv}(t)$$

$$\begin{cases} \sigma \boldsymbol{x}(t) = (\boldsymbol{A} - \boldsymbol{BFX}^{-1}\boldsymbol{C})\,\boldsymbol{x}(t) + \boldsymbol{B}(\boldsymbol{I} - \boldsymbol{FX}^{-1}\boldsymbol{D})\,\boldsymbol{Gv}(t) \\ \boldsymbol{y}(t) = \boldsymbol{X}^{-1}\boldsymbol{C}\;\boldsymbol{x}(t) + \boldsymbol{X}^{-1}\boldsymbol{DGv}(t) \end{cases}$$

なる新しい系を得る。

ここで，$\boldsymbol{X} \equiv \boldsymbol{I} + \boldsymbol{DF}$は正則とする。

実 行 例

〔1〕 状態フィードバック則に従った閉ループ系の構成例

```
 /DPACS/ TIME-16:06:32 CPU-00:00:01
CLPS/X EX1 EX1CLPSX
 SYSTEM EX1        HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0       (    3) ;SYSIN    /S /C /  /    (87/11/27) :EX1 TEST DATA
  2) EX1.S1       (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD EX1
  3) EX1.S2       (    3) ;COOD     /1 /  /  /    (87/11/30) :COOD/1 EX1
  4) EX1.S3       (    3) ;COOD     /2 /  /  /    (87/11/30) :COOD/2 EX1
  5) EX1.S4       (    3) ;CMR      /M /C /2 /    (87/11/30) :EX1 CMR
 WHICH SYSTEMDATA IS USED ? :    1
 TO OBTAIN CLOSED LOOP SYSTEM
 BY STATE FEEDBACK (U=-FX+GV)
 INPUT -MODE OF ( 2, 3) MATRIX   F       : _
 ( 1) ROW
1,2,-1
 ( 2) ROW
2,0,1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX   G       : _
 ( 1) ROW
1,1
 ( 2) ROW
0,1
 TYPE OR REVISE ? _
 GIVE COMMENT FOR SYSTEMDATA :
CLPS/X EX1
 /DPACS/ TIME-16:18:03 CPU-00:00:27
```

```
/DPACS/ TIME-16:06:33 CPU-00:00:01
TYPE/S EX1CLPSX
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1CLPSX.S0 ( 2, 2, 3)

 --- ( 3, 3) MATRIX   A         ---
 ( 1)  0.0              1.00000000D+00  0.0
 ( 2)  4.00000000D+00  -3.00000000D+00  5.00000000D+00
 ( 3)  1.00000000D+00   0.0             0.0
 FILENAME : _
 --- ( 3, 2) MATRIX   B         ---
 ( 1)  0.0              0.0
 ( 2)  1.00000000D+00  -1.00000000D+00
 ( 3)  0.0              1.00000000D+00
 FILENAME : _
 --- ( 2, 3) MATRIX   C         ---
 ( 1)  1.00000000D+00   0.0             0.0
 ( 2)  0.0              1.00000000D+00  1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX   D         ---
 ( 1)  0.0              0.0
 ( 2)  0.0              0.0
 FILENAME : _
 /DPACS/ TIME-16:18:04 CPU-00:00:27
```

〔2〕 出力フィードバック則に従った閉ループ系の構成例

```
 /DPACS/ TIME-16:06:34 CPU-00:00:01
CLPS/Y EX1 EX1CLPSY
 SYSTEM EX1       HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0       (    3) ;SYSIN   /S /C /  /   (87/11/27) :EX1 TEST DATA
  2) EX1.S1       (    3) ;COOD    /1 /  /  /   (87/11/30) :COOD EX1
  3) EX1.S2       (    3) ;COOD    /1 /  /  /   (87/11/30) :COOD/1 EX1
  4) EX1.S3       (    3) ;COOD    /2 /  /  /   (87/11/30) :COOD/2 EX1
  5) EX1.S4       (    3) ;CMR     /M /C /2 /   (87/11/30) :EX1 CMR
 WHICH SYSTEMDATA IS USED ? :   1
 TO OBTAIN CLOSED LOOP SYSTEM
 BY OUTPUT FEEDBACK (U=-FY+GV)
 INPUT -MODE OF ( 2, 2) MATRIX   F       : _
 ( 1) ROW
1,1
 ( 2) ROW
-2,-3
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX   G       : I
 TYPE OR REVISE ? _
 GIVE COMMENT FOR SYSTEMDATA :
CLPS/Y EX1
 /DPACS/ TIME-16:18:05 CPU-00:00:27
```

```
/DPACS/ TIME-16:06:36 CPU-00:00:01
TYPE/S EX1CLPSY
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA EX1CLPSY.SO ( 2, 2, 3)

 --- ( 3, 3) MATRIX   A       ---
 ( 1)  0.0               1.00000000D+00  0.0
 ( 2) -4.00000000D+00 -8.00000000D+00 -5.00000000D+00
 ( 3)  5.00000000D+00  3.00000000D+00  4.00000000D+00
 FILENAME : _
 --- ( 3, 2) MATRIX   B       ---
 ( 1)  0.0               0.0
 ( 2)  1.00000000D+00 -2.00000000D+00
 ( 3)  0.0               1.00000000D+00
 FILENAME : _
 --- ( 2, 3) MATRIX   C       ---
 ( 1)  1.00000000D+00  0.0               0.0
 ( 2)  0.0               1.00000000D+00  1.00000000D+00
 FILENAME : _
 --- ( 2, 2) MATRIX   D       ---
 ( 1)  0.0               0.0
 ( 2)  0.0               0.0
 FILENAME : _
 /DPACS/ TIME-16:18:07 CPU-00:00:27
```

シミュレーションの実行	DP. V. 3
SMLT [# / ／N] [# / ／U] ␣SN	

機　能

システムの応答をシミュレートし，NLPまたは画面にグラフにして出力する。(注)

◆第1スイッチは，線形／複合非線形の種別を表し，／Nを付加した場合には，複合非線形システムのシミュレーションをする。

◆第2スイッチは，任意入力を加えた場合のシミュレーションを行うかどうかの選択である。本スイッチを付加しない場合には，自由応答のシミュレーションを行う。

(注) 図形処理端末用システムと一般の端末用システムを起動（3.5.1　DPACS／Jの起動方法参照）した場合で出力先がそれぞれ画面（図形処理端末）とNLP（一般の端末）とに分かれる。NLPへの出力方法については，3.5.2　DPACS／Jの終了方法を参照のこと。

理論概要

◆線型システムのシミュレーション

自由応答：適当な初期状態 $\boldsymbol{x}(0)$からの自由応答を次式に従いシミュレーションする。

$$\boldsymbol{x}(k+1)=\boldsymbol{A}\boldsymbol{x}(k),\quad \boldsymbol{y}(k)=\boldsymbol{C}\boldsymbol{x}(k)$$

連続系の場合は，適当なサンプリングで離散化する。

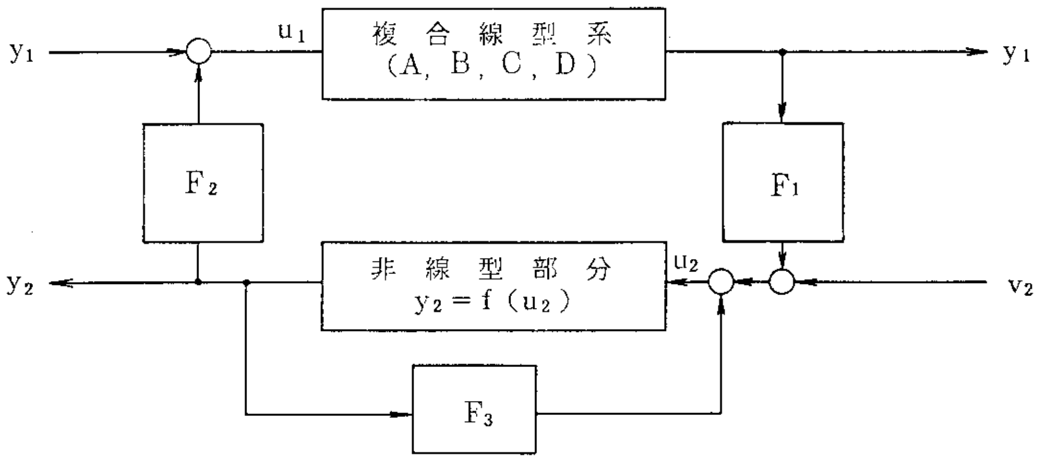
任意入力：適当な初期状態 $\boldsymbol{x}(0)$ からの任意入力による応答を次式に従いシミュレーションする。

$$\begin{aligned}\boldsymbol{x}(k+1)&=\boldsymbol{A}\boldsymbol{x}(k)+\boldsymbol{B}\boldsymbol{u}(k)\\ \boldsymbol{y}(k)&=\boldsymbol{C}\boldsymbol{x}(k)+\boldsymbol{D}\boldsymbol{u}(k)\end{aligned}$$

連続系の場合は，適当なサンプリングで離散化する。

◆非線型システムのシミュレーション

ＬＩＮＫにより結合された複合非線型系について



線型部分の初期状態は任意，非線型要素のうち過去の入力値または出力値を必要とするものは，その初期値を０とする。

＜非線型要素の計算順序＞

個々の非線型要素の出力を決定するには，それぞれの入力がすでに求まっている必要がある。その計算順序は，アソシエーションマトリクス$\boldsymbol{H}$から次のように求まる。まず$\boldsymbol{H}$の中から要素がすべて０である行を探す。その行番号に相当する非線型要素の入力には，他の非線型要素の出力が加わっていないので，その出力は計算可能である。対応する行と列を$\boldsymbol{H}$から取り除く。その際，残った行と列の番号はもとのままにしておく。同じ操作を行と列が残らなくなるまで繰り返す。こうして取り除いていった行の番号の順序が，非線型要素の計算順序となる。

◆シミュレーションの計算順序

○ $\boldsymbol{D}=\boldsymbol{0}$ の場合

k＝0～Nについて以下の手順でシミュレーションを行う。

① $\boldsymbol{y}_1(k)=\boldsymbol{C}\boldsymbol{x}(k)$

② $\boldsymbol{y}_2(k)=\boldsymbol{f}(\boldsymbol{v}_2(k)+\boldsymbol{F}_1\boldsymbol{y}_1(k)+\boldsymbol{F}_3\boldsymbol{y}_2(k))$ を計算。

③ $\boldsymbol{x}(k+1)=\boldsymbol{A}\boldsymbol{x}(k)+\boldsymbol{B}(\boldsymbol{v}_1(k)+\boldsymbol{F}_2\boldsymbol{y}_2(k))$ を計算。

○ $\boldsymbol{D}\neq\boldsymbol{0}$ の場合

同時刻の $\boldsymbol{y}_1$ と $\boldsymbol{y}_2$ を計算することは，非線型代数方程式を解くことになり困難である。従って非線型部分の出力に一時刻分の遅れを持たせて線型部分に入力するという形でシミュレーションを行う

① $\boldsymbol{y}_2^0=\boldsymbol{f}(\boldsymbol{v}_2(0)+\boldsymbol{F}_3\boldsymbol{y}_2^0)$ を計算

② $\boldsymbol{u}_1(0)=\boldsymbol{v}_1(0)+\boldsymbol{F}_2\boldsymbol{y}_2^0$ として $\boldsymbol{y}_1(0)$，$\boldsymbol{x}(1)$ を計算

③ $\boldsymbol{y}_2(0)=\boldsymbol{f}(\boldsymbol{v}_2(0)+\boldsymbol{F}_1\boldsymbol{y}_1(0)+\boldsymbol{F}_3\boldsymbol{y}_2^0)$ を計算

以後k＝1，～，Nとして次を繰り返す。

④ $\boldsymbol{u}_1(k)=\boldsymbol{v}_1(k)+\boldsymbol{F}_2\boldsymbol{y}_2(k-1)$ として $\boldsymbol{y}_1(k)$，$\boldsymbol{x}(k+1)$ を計算する。

⑤ $\boldsymbol{y}_2(k)=\boldsymbol{f}(\boldsymbol{v}_2(k)+\boldsymbol{F}_1\boldsymbol{y}_1(k)+\boldsymbol{F}_3\boldsymbol{y}_2(k-1))$ を計算する。

この場合，$\boldsymbol{y}_2(k)$に遅れを持たせることによりシミュレーションには誤差を生ずる。このため，サンプル周期は，十分小さくする必要がある。

◆アソシェーションマトリクス

N個の節点からなる有向グラフが与えられたとき，N×Nのブールマトリクスを考え，その（i，j）要素をj番目の節点からi番目の節点へ向かう枝があるときには1とし，ないときには0とする。こうして得られるブールマトリクスをアソシエーションマトリクス（節点接続行列）という。アソシエーションマトリクスは与えられた有向グラフのN個の節点の接続関係を表わしている。アソシエーションマトリクスは次の性質を持つ。

① アソシエーションマトリクスにおいて，ある行，たとえば第i行の要素がすべて0ならば，i番目の節点へ向かう枝は存在しない。

② ある列，たとえば第j列の要素がすべて0ならば，j番目の節点から出発する枝は存在しない。

③ アソシエーションマトリクスを$\boldsymbol{A}$とすると $\boldsymbol{A}^N=0$ ならばその有向グラフにはループは存在せず，$\boldsymbol{A}^N\neq0$ ならばループが存在する。

実行例

〔1〕 連続系線型状態空間表現形式システムの自由応答の例

```
 /DPACS/ TIME-16:06:37 CPU-00:00:01
SMLT SYSM2CL
 TO SIMULATE SYSTEM'S RESPONSE
 --- FREE RESPONSE ---
 ADJUST NOMINATED SAMPLING INTERVAL (  9.93148038D-04) ? _
 DELT =   1D-3
 HOW MANY POINTS ? :    500
 GIVE INITIAL STATE X(0)
 INPUT -MODE OF ( 6, 1) MATRIX X(0)      : _
 ( 1) ROW
0.1
 ( 2) ROW
0.1
 ( 3) ROW
0.1
 ( 4) ROW
0.1
 ( 5) ROW
0.1
 ( 6) ROW
0.1
 TYPE OR REVISE ? _
 STARTING POINT ? (0- 499) :    0
 HOW MANY POINTS ? (1- 500) :    500
 MINIMUM AND MAXIMUM OF EACH VARIABLE
  Y( 1) :    3.92490903D-02  1.00747066D-01
  Y( 2) :   -1.25213347D-03  1.00000000D-01
  Y( 3) :   -4.13660467D-02  1.00000000D-01
  Y( 4) :    8.59205992D-02  1.00000000D-01
  Y( 5) :    5.28511490D-02  1.00572279D-01
  Y( 6) :    3.25161715D-03  1.00000000D-01
 HOW MANY VARIABLES ARE PLOTTED ?   6 ①
 SPECIFY MINIMUM AND MAXIMUM OF Y-AXIS

  YMIN =< -4.13660467D-02
  YMAX => 1.00747066D-01
 GIVE RANGE OF Y-AXIS : -4.5D-2,1.5D-1
 SPECIFY TYPE OF LINE ?   N ②
 DRAW AXIS ? _
 NORMAL TYPE ? _
 NAME OF X-AXIS :
TIME (SEC)
 NAME OF Y-AXIS :
OUTPUT
 TITLE :
SMLT OUTPUT
 OUTPUT-MODE OF ( 6, 500) TIME SERIES OUTPUT    :  F
 OUTPUT-FILENAME : _
 /DPACS/ TIME-16:18:08 CPU-00:00:27
```

【説明】

①0を入力すると，ファイル出力までの間の処理を省略する。

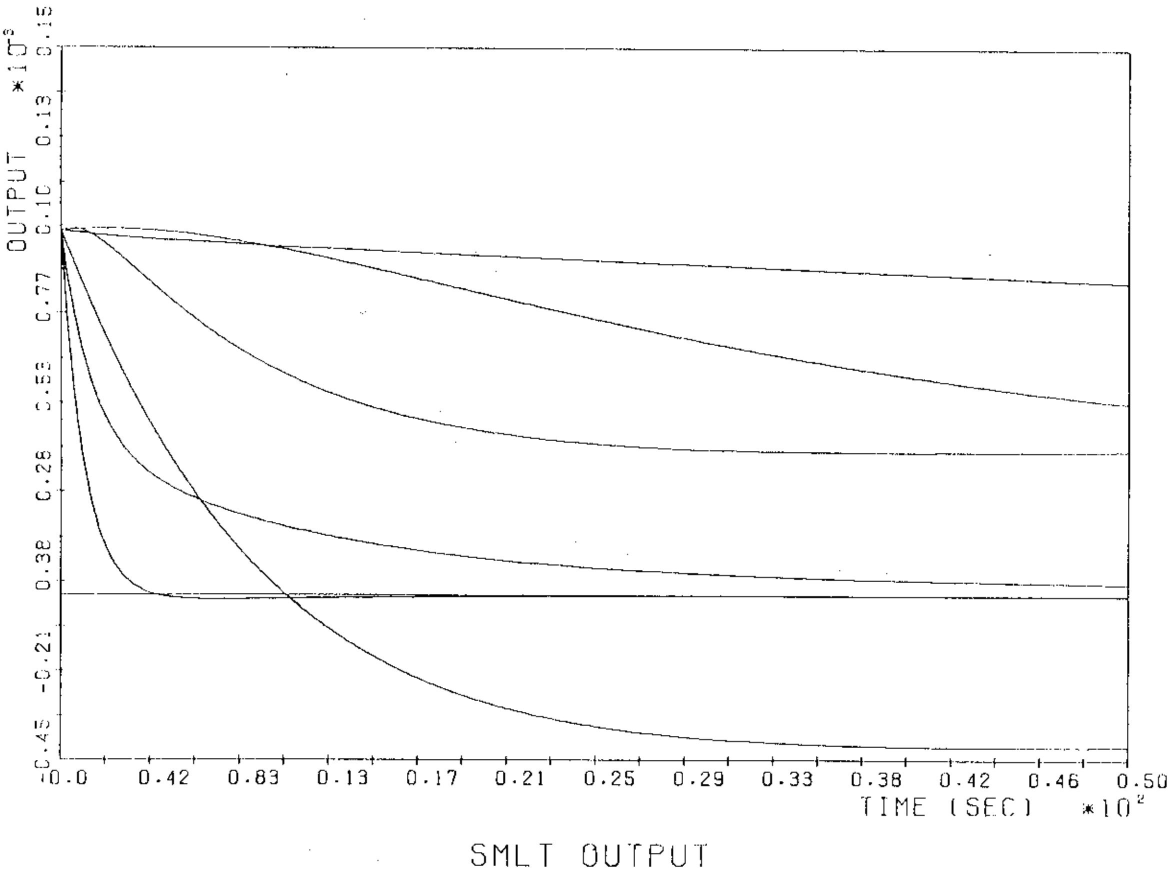
②Yを入力すると，線種／色を選択する旨のメッセージが表示される。

＜図形処理端末用＞

1. RED 2. GREEN 3. PURPLE 4. BLUE 5. PINK 6. WHITE

＜一般の端末（NLP出力）用＞

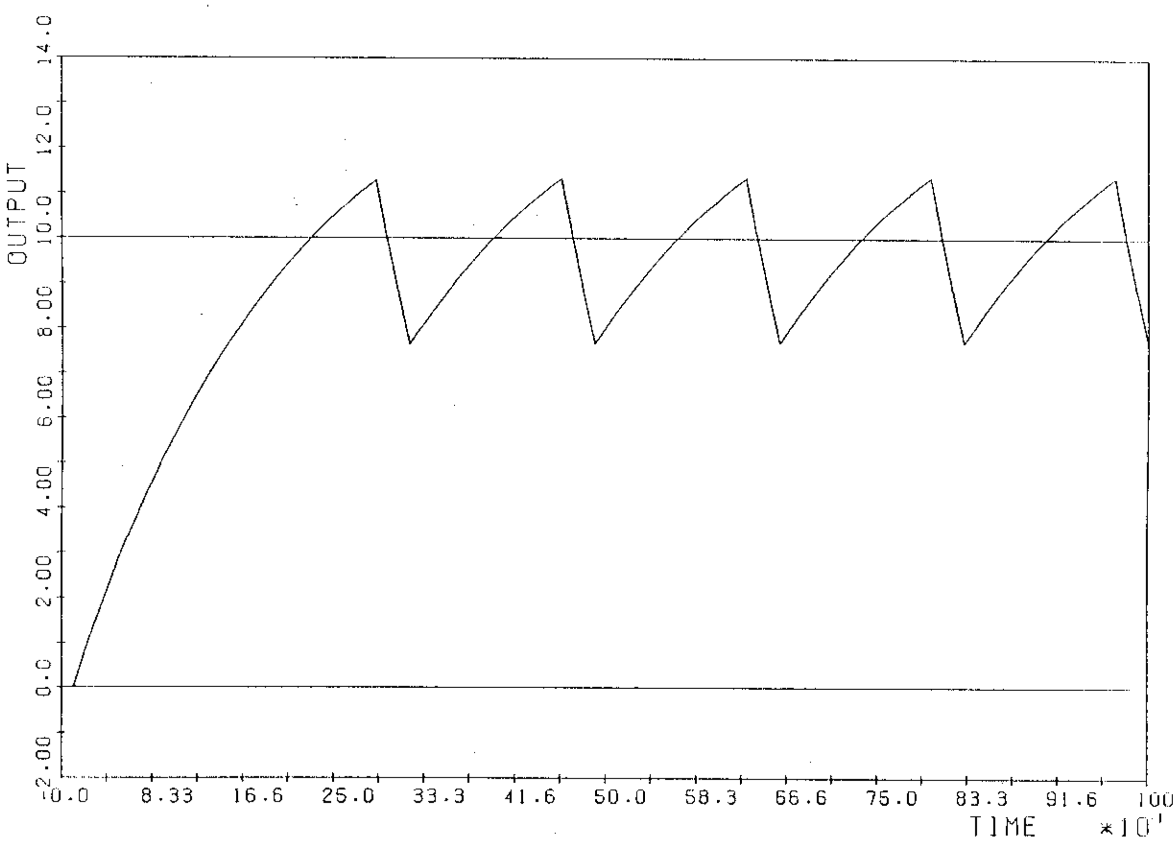
1. SOLID 2. DOTTED-1 3. DOTTED-2 4. CHAIN



自由応答の出力例

〔2〕 複合非線型システムの自由応答例

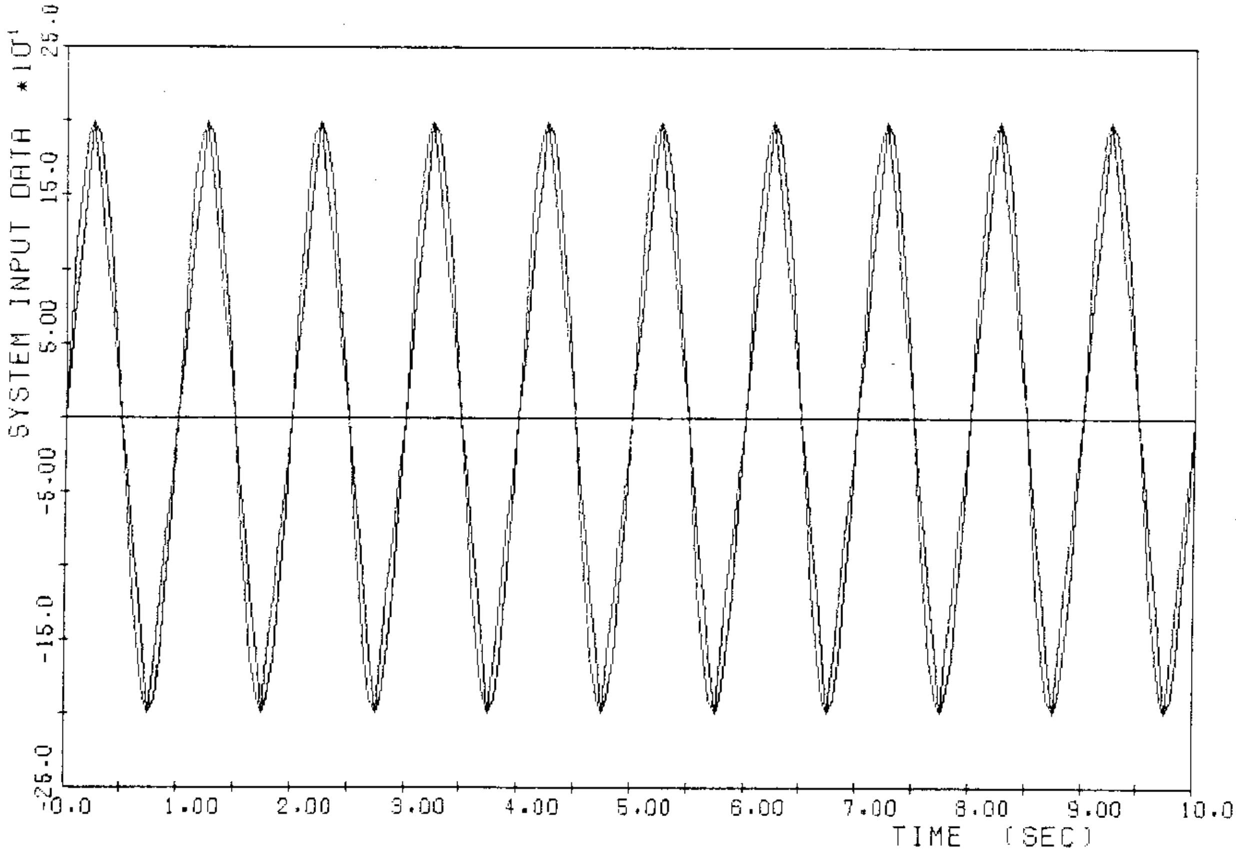
```
 /DPACS/ TIME-16:06:41 CPU-00:00:02
SMLT/N EX7NN2
 TO SIMULATE SYSTEM'S RESPONSE
 --- FREE RESPONSE ---
 ADJUST NOMINATED SAMPLING INTERVAL (  2.00000000D-01) ? _
 DELT =   0.1
 HOW MANY POINTS ? :   100
 GIVE INITIAL STATE X(0)
 INPUT -MODE OF ( 2, 1) MATRIX X(0)      : _
 ( 1) ROW
0
 ( 2) ROW
10
 TYPE OR REVISE ? _
 * OUTPUT DATA OF LINEAR PART *
 STARTING POINT ? (0-  99) :   0
 HOW MANY POINTS ? (1- 100) :   100
 MINIMUM AND MAXIMUM OF EACH VARIABLE
  Y( 1) :   0.0             1.13733924D+01
  Y( 2) :   1.00000000D+01  1.00000000D+01
 ***
 HOW MANY VARIABLES ARE PLOTTED ?  2
 SPECIFY MINIMUM AND MAXIMUM OF Y-AXIS

  YMIN =<  0.0
  YMAX =>  1.13733924D+01
 GIVE RANGE OF Y-AXIS : -2,14
 SPECIFY TYPE OF LINE ?  N
 DRAW AXIS ? _
 NORMAL TYPE ? _
 NAME OF X-AXIS :
TIME
 NAME OF Y-AXIS :
OUTPUT
 TITLE :
SMLT/N EX7NN2  (PLOT)
 OUTPUT-MODE OF ( 2, 100) TIME SERIES OUTPUT   :  F
 OUTPUT-FILENAME : ___
 * OUTPUT DATA OF NONLINEAR PART *
 PLOT OUTPUT OF NONLINEAR PART ?  N
 OUTPUT-MODE OF ( 3, 100) TIME SERIES OUTPUT   :  F
 OUTPUT-FILENAME : ___
 /DPACS/ TIME-16:18:09 CPU-00:00:27
```



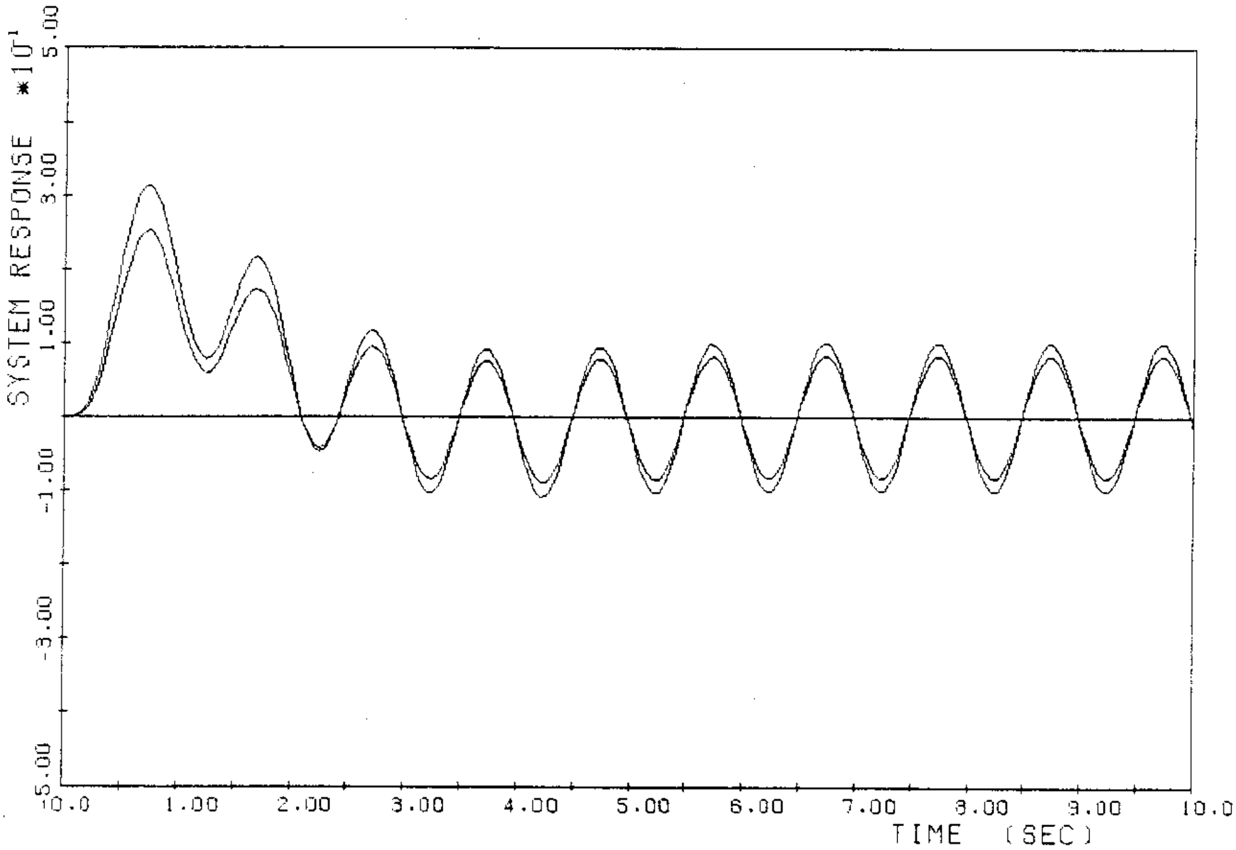
SMLT/N EX7NN2 (PLOT)

〔3〕 連続系線型状態空間表現形式システムの任意入力時系列データに対する応答の例

```
 /DPACS/ TIME-16:06:46 CPU-00:00:02
SMLT/U MODEL
 TO SIMULATE SYSTEM'S RESPONSE
 ADJUST NOMINATED SAMPLING INTERVAL (  1.000000000-01) ?_
 DELT =   0.05
 SPECIFY LENGTH OF INPUT DATA :   200
 INPUT -MODE OF ( 2, 200) TIME SERIES INPUT    :  F
 INPUT -FILENAME :
INPUT
 GIVE INITIAL STATE X(0)
 INPUT -MODE OF ( 4, 1) MATRIX X(0)      :  _
 ( 1) ROW
0
 ( 2) ROW
0
 ( 3) ROW
0
 ( 4) ROW
0
 TYPE OR REVISE ? _

 STARTING POINT ? (0- 199) :   0
 HOW MANY POINTS ? (1- 200) :   200
 MINIMUM AND MAXIMUM OF EACH VARIABLE
  Y( 1) :  -1.08244678D-01  3.13252207D-01
  Y( 2) :  -8.89489615D-02  2.51940819D-01
 HOW MANY VARIABLES ARE PLOTTED ?  2
 SPECIFY MINIMUM AND MAXIMUM OF Y-AXIS

  YMIN =< -1.08244678D-01
  YMAX =>  3.13252207D-01
 GIVE RANGE OF Y-AXIS : -0.5,0.5
 SPECIFY TYPE OF LINE ?  N
 DRAW AXIS ?  Y
 NORMAL TYPE ?  N
 NO. OF UNIT INTERVALS FOR X-AXIS (1-24) :   20
 DRAW SCALE ?  Y
 NAME OF X-AXIS :
TIME  (SEC)
 NO. OF UNIT INTERVALS FOR Y-AXIS (1-16) :   10
 DRAW SCALE ?  Y
 NAME OF Y-AXIS :
SYSTEM RESPONSE
 TITLE :
 ***
LINEAR SYSTEM RESPONSE
 DRAW FRAME ?  Y
 DRAW ZERO LEVEL ?  Y
 OUTPUT-MODE OF ( 2, 200) TIME SERIES OUTPUT    :  F
 OUTPUT-FILENAME :  OUTPUT
 /DPACS/ TIME-16:18:31 CPU-00:00:27
```



LINEAR SYSTEM INPUT DATA

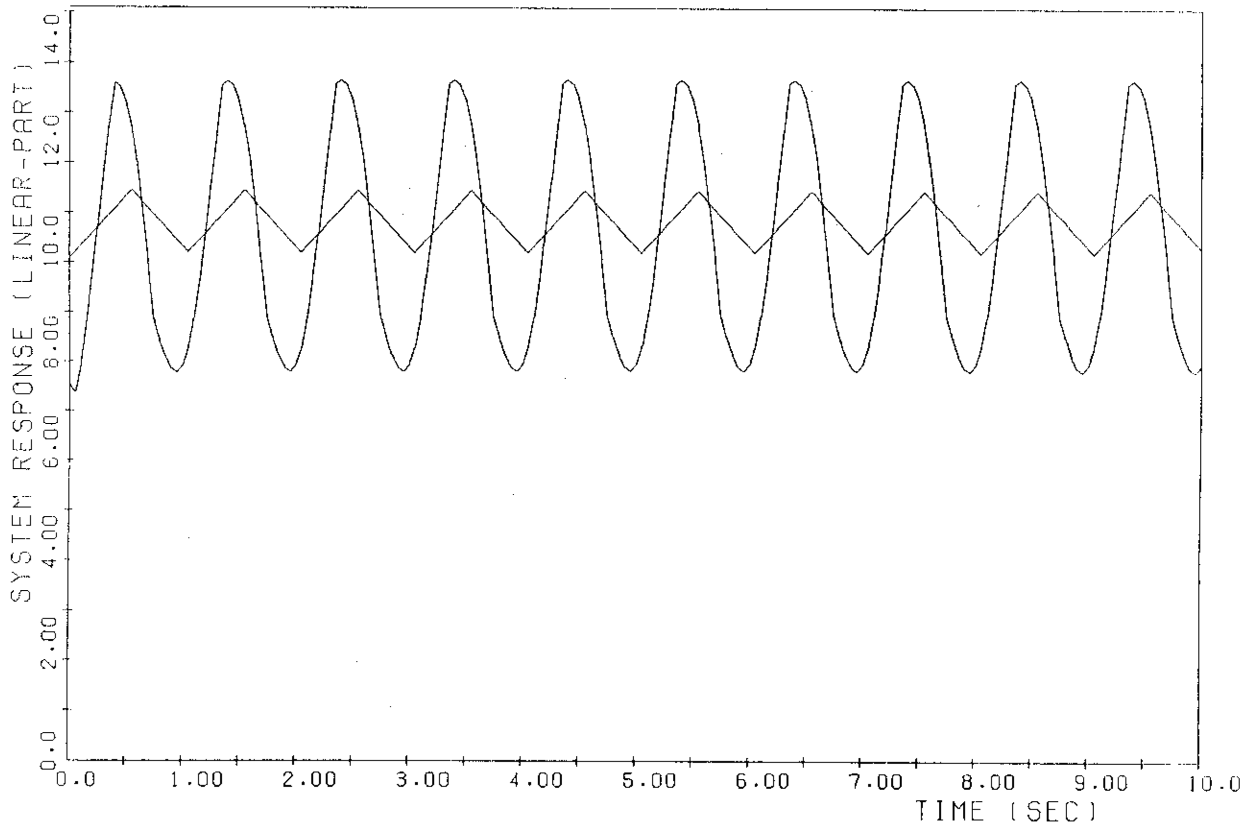


LINEAR SYSTEM RESPONSE

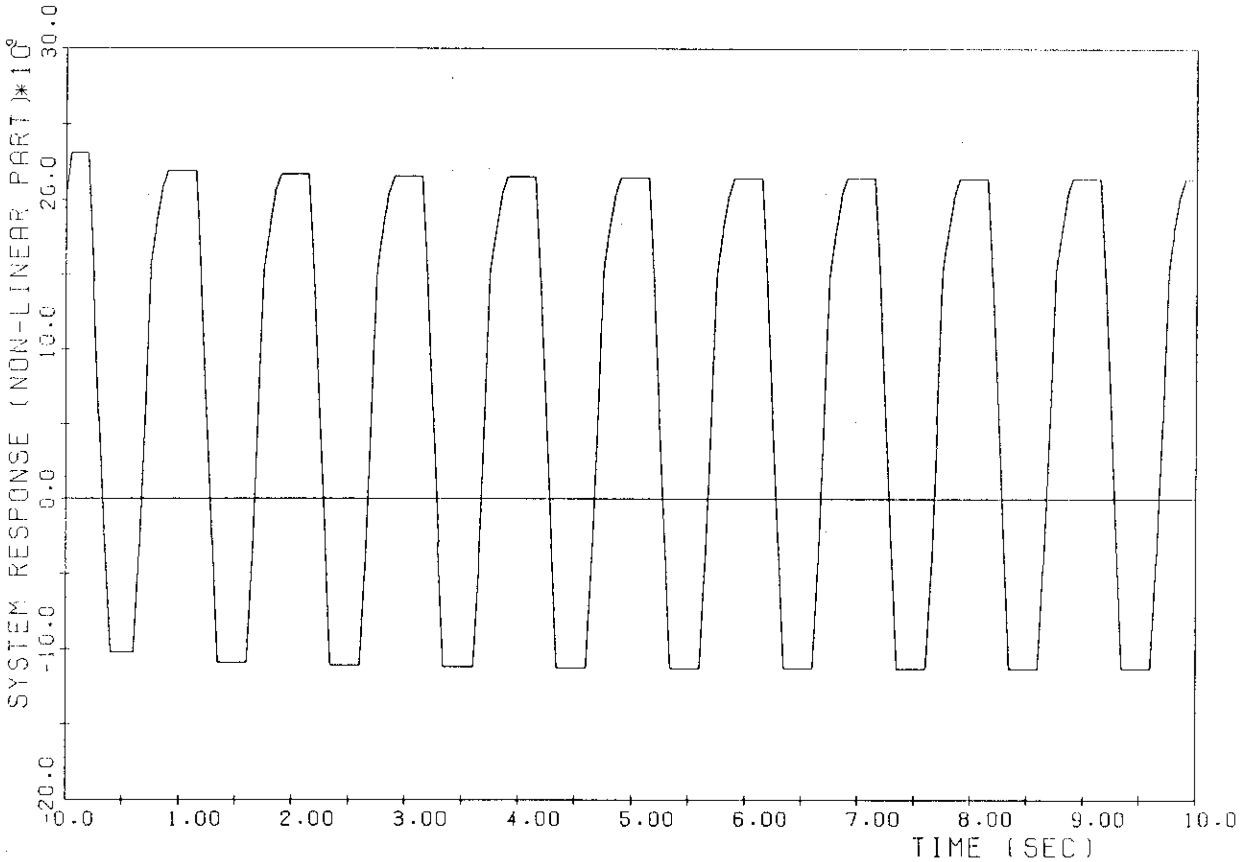
〔4〕 連続系複合非線型システムの任意入力時系列データに対する応答の例

```
 /DPACS/ TIME-16:06:48 CPU-00:00:02
SMLT/N/U EX7N
 TO SIMULATE SYSTEM'S RESPONSE
 ADJUST NOMINATED SAMPLING INTERVAL (  2.00000000D-01) ? _
 DELT =   0.05
 SPECIFY LENGTH OF INPUT DATA :   200
 * INPUT DATA OF LINEAR PART *
 INPUT -MODE OF ( 2, 200) TIME SERIES INPUT     :  F
 INPUT -FILENAME : _
INPUT
 * INPUT DATA OF NONLINEAR PART *
 INPUT -MODE OF ( 3, 200) TIME SERIES INPUT     :  C
 CONSTANT VALUE =   0.5
 GIVE INITIAL STATE X(0)
 INPUT -MODE OF ( 2, 1) MATRIX X(0)       :  _
 ( 1) ROW
1
 ( 2) ROW
10
 TYPE OR REVISE ?_
 * OUTPUT DATA OF LINEAR PART *
 ***
 STARTING POINT ? (0- 199) :   0
 HOW MANY POINTS ? (1- 200) :   200
 MINIMUM AND MAXIMUM OF EACH VARIABLE
  Y( 1) :   7.31482434D+00  1.35899924D+01
  Y( 2) :   1.00000000D+01  1.13750000D+01
 HOW MANY VARIABLES ARE PLOTTED ?  2
 SPECIFY MINIMUM AND MAXIMUM OF Y-AXIS

  YMIN =<  7.31482434D+00
  YMAX =>  1.35899924D+01
 GIVE RANGE OF Y-AXIS : 0,15
 SPECIFY TYPE OF LINE ?   N
 DRAW AXIS ?   Y
 NORMAL TYPE ?   N
 NO. OF UNIT INTERVALS FOR X-AXIS (1-24) :    20
 DRAW SCALE ?   Y
 NAME OF X-AXIS :
TIME (SEC)
 NO. OF UNIT INTERVALS FOR Y-AXIS (1-16) :    15
 DRAW SCALE ?   Y
 NAME OF Y-AXIS :
SYSTEM RESPONSE (LINEAR-PART)
 TITLE :
 ***
SIMULATION OF NON-LINEAR SYSTEM EX7N
 DRAW FRAME ?   Y
 OUTPUT-MODE OF ( 2, 200) TIME SERIES OUTPUT    :  F
 OUTPUT-FILENAME :   LINEAR
```

```
* OUTPUT DATA OF NONLINEAR PART *
PLOT OUTPUT OF NONLINEAR PART ?   Y
STARTING POINT ? (0- 199) :   0
HOW MANY POINTS ? (1- 200) :   200
MINIMUM AND MAXIMUM OF EACH VARIABLE
 Y( 1) :  -5.00000000D-01  1.50000000D+00
 Y( 2) :  -1.13447719D+10  2.31017566D+10
 Y( 3) :  -1.00000000D+00  1.00000000D+00
HOW MANY VARIABLES ARE PLOTTED ?   3
SPECIFY MINIMUM AND MAXIMUM OF Y-AXIS

 YMIN =< -1.13447719D+10
 YMAX =>  2.31017566D+10
GIVE RANGE OF Y-AXIS : -2D10,3D10
SPECIFY TYPE OF LINE ?   N
ERASE TEKTRO CRT ?  ( Y/N )  N
DRAW AXIS ?
NORMAL TYPE ?   N
NO. OF UNIT INTERVALS FOR X-AXIS (1-24) :    20

20
DRAW SCALE ?   Y
NAME OF X-AXIS :
TIME (SEC)
NO. OF UNIT INTERVALS FOR Y-AXIS (1-16) :    10
DRAW SCALE ?   Y
NAME OF Y-AXIS :
SYSTEM RESPONSE (NON-LINEAR PART)
TITLE :
SIMULATION OF NON-LINEAR SYSTEM EX7N
DRAW FRAME ?   Y
DRAW ZERO LEVEL ?    Y
OUTPUT-MODE OF ( 3, 200) TIME SERIES OUTPUT    :  F
OUTPUT-FILENAME :   NLINEAR
/DPACS/ TIME-16:18:33 CPU-00:00:27
```



SIMULATION OF NON-LINEAR SYSTEM EX7N



SIMULATION OF NON-LINEAR SYSTEM EX7N

(逆) ナイキスト線図の作図	DP. Ⅵ. 1
NYQST [# / /I] _SN	

機能

システムのナイキスト線図または逆ナイキスト線図を描く。

◆第1スイッチが省略された場合には，ナイキスト線図を描く。

◆第1スイッチ／Iが付加された場合には，逆ナイキスト線図を描く。

◆計算結果時系列データには，第1行に角周波数 ω，第2行にゲインの実数部，第3行にゲインの虚数部が格納されている。

理論概要

◆ナイキスト線図

1巡伝達関数行列 $\boldsymbol{G}(s)=\{g_{ij}(s)\}$ において $s=j\omega$ とし $\omega=-\infty\sim\infty$ に対し $g_{ij}(j\omega)$ を複素平面に描いたものをナイキスト線図という。

<1入力1出力系の安定判別>

$(-1+j\cdot 0)$ 点からナイキスト線図上の一点にベクトルを引く。$\omega=-\infty\rightarrow\infty$ としたときに，このシステムが安定であるための必要十分条件は，

（このベクトルの反時計回りの回転数）＝（$\boldsymbol{G}(s)$の正実部を持つ極の数）

ただし，系が線型で，$\boldsymbol{G}(s)$が $s\rightarrow 0$ のとき0又は一定値に収束する場合である。

◆逆ナイキスト線図

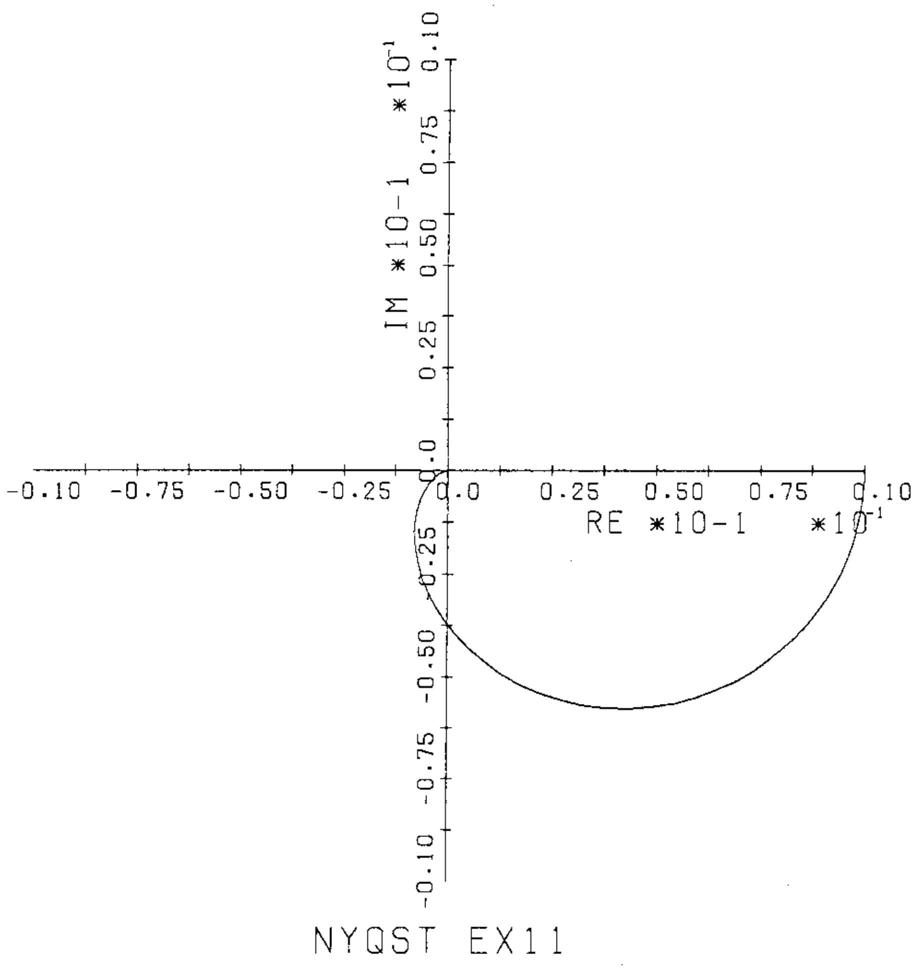
1巡伝達関数行列 $\boldsymbol{G}(s)$の逆行列 $\boldsymbol{G}(s)^{-1}=\{\hat{g}_{ij}(s)\}$ に対し上記と同様の図を描く。1入力1出力系の安定判別は「反時計回り」が「時計回り」に変る以外は同様。

実行例

〔1〕 ナイキスト線図の出力例

```
 /DPACS/ TIME-16:06:49 CPU-00:00:02
NYQST EX11
 TO DRAW NYQUIST DIAGRAM
 SPECIFY PARAMETERS FOR DRAWING NYQUIST DIAGRAM
 1) ON WHICH ELEMENT ? (I,J) :   1,1
 2) MINIMUM VALUE OF OMEGA :    0.01
 3) MAXIMUM VALUE OF OMEGA :    100
 4) NUMBER OF CALCULATIONS :    100
 GIVE FULL SCALE OF AXIS (MAXIMUM VALUE =  9.99876012D-01 :  1.0
 TYPE OF LINE  1. SOLID  2. DOTTED-1  3. DOTTED-2  4. CHAIN
 WHICH TYPE ? :  1
 DRAW AXIS ? _
 NORMAL TYPE ? _
 NAME OF X-AXIS :
```

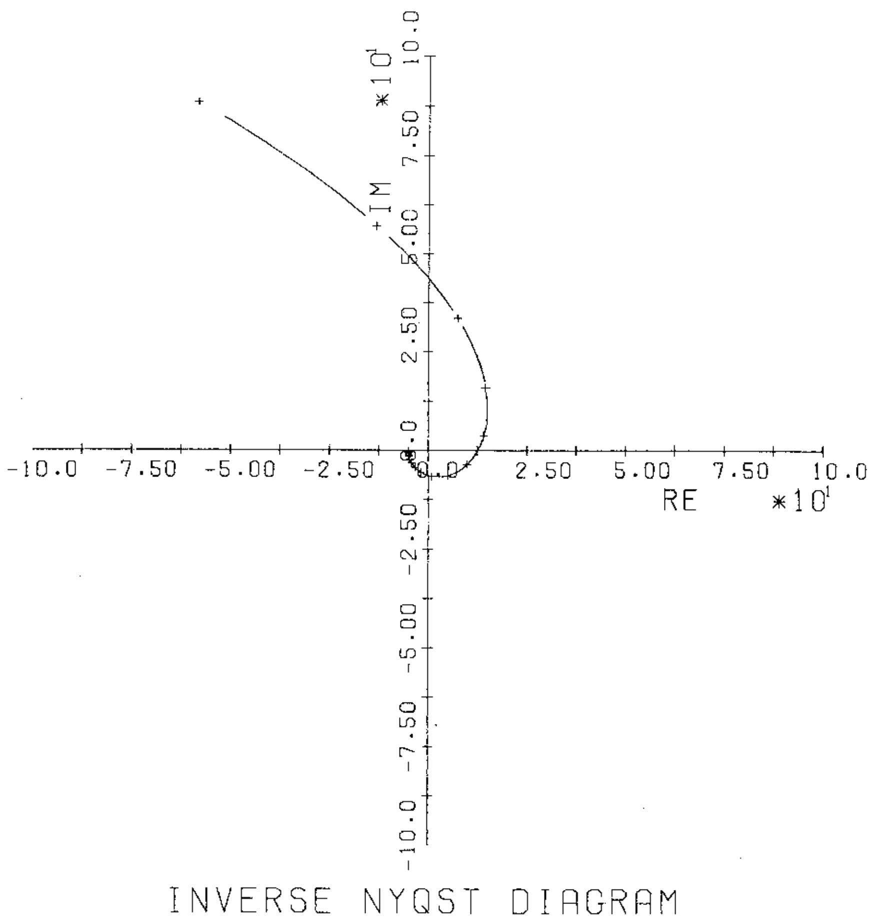
```
RE *10-1
 NAME OF Y-AXIS :
IM *10-1
 TITLE :
NYQST EX11
 DRAW OTHER ELEMENT ?   N
 /DPACS/ TIME-16:18:34 CPU-00:00:27
```



ナイキスト線図例

〔2〕 逆ナイキスト線図の作図例

```
 /DPACS/ TIME-16:06:51 CPU-00:00:02
NYQST/I EX11
 TO DRAW INVERSE NYQUIST DIAGRAM
 SPECIFY PARAMETERS FOR DRAWING INVERSE NYQUIST DIAGRA M
 1) ON WHICH ELEMENT ? (I,J) :   1,1
 2) MINIMUM VALUE OF OMEGA :   0.01
 3) MAXIMUM VALUE OF OMEGA :   10
 4) NUMBER OF CALCULATIONS :   100
 GIVE FULL SCALE OF AXIS (MAXIMUM VALUE =  9.19230769D+01 :  100
 TYPE OF LINE  1. SOLID  2. DOTTED-1  3. DOTTED-2  4. CHAIN
 WHICH TYPE ? :   2
 DRAW AXIS ? _
 NORMAL TYPE ? _
 NAME OF X-AXIS :
RE
 NAME OF Y-AXIS :
IM
 TITLE :
INVERSE NYQST DIAGRAM
 DRAW OTHER ELEMENT ?   N
 /DPACS/ TIME-16:18:36 CPU-00:00:27
```



逆ナイキスト線図出力例

ボード線図の作図	DP. VI. 2
BODE_SN	

機　能

システムのボード線図を描く。

◆計算結果時系列データには，第1行に角周波数ω，第2行にゲイン（dB），第3行に位相（deg）が格納されている。

理論概要

伝達関数行列$G(s)=\{g_{ij}(s)\}$で$s=j\omega$とおき

ゲイン$=20\log_{10}|g_{ij}(j\omega)|$　　(dB)

位　相$=\tan^{-1}\{Im(g_{ij}(j\omega))/Re(g_{ij}(j\omega))\}$　　(deg)

を縦軸に，$\log\omega$を横軸にとった線図で，

[(ゲイン0dBの時の位相+180°]を位相余有

[(位相が−180°の時のゲインの絶対値]をゲイン余有

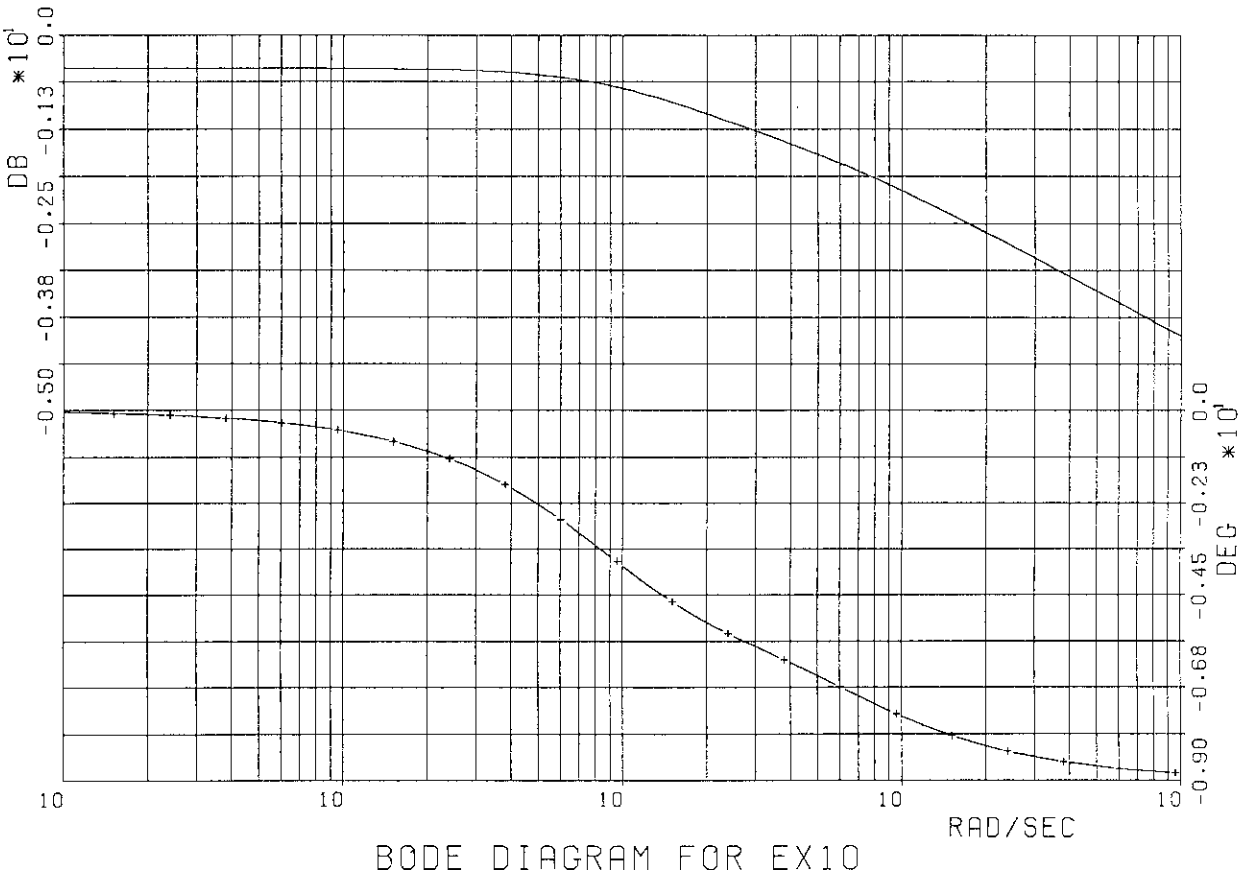
と呼び，システムの安定性が調べられる。

実 行 例

〔1〕　ボード線図の出力例

```
/DPACS/ TIME-16:06:52 CPU-00:00:02
BODE EX10
 TO DRAW BODE DIAGRAM
 SPECIFY PARAMETERS FOR DRAWING BODE DIAGRAM
 1) ON WHICH ELEMENT ? (I,J) :   2,1
 POLES OF THE SYSTEM
 ------------------------------------------------------------------------
  NO.      REAL                   IMAGINARY                 ABSOLUTE
 ------------------------------------------------------------------------
 ( 1)   -1.000000000D+00       0.0                      1.000000000D+00
 ( 2)   -5.000000000D+00       0.0                      5.000000000D+00
 ------------------------------------------------------------------------
 ZEROS OF THE ELEMENT
 ------------------------------------------------------------------------
  NO.      REAL                   IMAGINARY                 ABSOLUTE
 ------------------------------------------------------------------------
 ( 1)   -3.000000000D+00       0.0                      3.000000000D+00
 ------------------------------------------------------------------------
 2) MINIMUM VALUE OF OMEGA :    0.01
 3) MAXIMUM VALUE OF OMEGA :    100
 4) NUMBER OF CALCULATIONS :    100
  GAIN-AXIS RANG (MIN,MAX)=-40.0073711931917266 ,-4.43737838216593317
 ***
  PHASE-AXIS RANG (MIN,MAX)=-88.2830140778602193 ,-0.496544879561584426
 GIVE GAIN-AXIS(MIN,MAX) :  -50,0
 GIVE PHASE-AXIS(MIN,MAX) -90,0
 TYPES OF LINES  1. SOLID  2. DOTTED-1  3. DOTTED-2  4. CHAIN
```

```
 GIVE TYPES OF LINES FOR GAIN AND PHASE :    1,2
2
 DRAW AXIS ? _
 NORMAL TYPE ? _
 NAME OF OMEGA-AXIS :
RAD/SEC
 NAME OF GAIN-AXIS :
DB
 NAME OF PHASE-AXIS :
DEG
 TITLE :
BODE DIAGRAM FOR EX10
 DRAW OTHER ELEMENT ?   N
 /DPACS/ TIME-16:18:37 CPU-00:00:27
```



ボード線図出力例

根軌跡の作図	DP. VI. 3
LOCI_SN	

機　能

システムの根軌跡を描く。

＜注意事項＞

①図上で1mm以上離れた点だけが小さな⊠印でプロットされる。

②システムの極は，大きい⊠印でプロットされる。

③零点は，×印でプロットされる。

理論概要

1巡伝達関数G(s)が

$$G(s)=K\prod_{i=1}^{m}(s-z_i)\Big/\prod_{i=1}^{n}(s-p_i)$$

このとき閉ループ系の特性方程式は

$$1+G(s)=1+K\prod_{i=1}^{m}(s-z_i)\Big/\prod_{i=1}^{n}(s-p_i)=0$$

K＝0→∞とした時の閉ループ系の極がs平面上で動く軌跡を根軌跡という。

根軌跡は次のような性質を持つ

(i)根軌跡は実軸について対称である。

(ii)根軌跡はG(s)の極から出発し零点に至る連続曲線。

(iii)根軌跡の本数は，極の数に等しい。n＞mのとき（n－m）個の根軌跡は無限遠に至り，その際，実軸と $\dfrac{(2k-1)\pi}{n-m}$ （k＝0, 1,…）なる角度で交わる漸近線が存在する。

(iv)漸近線の出発点は $\left(\sum_{i=1}^{n}p_i-\sum_{i=1}^{m}z_i\right)\Big/(n-m)$ で，これを根の重心という。

(v)右側にG(s)の極または零点が奇数個存在する実軸部分は，根軌跡の一部となる。

(vi)根軌跡と実軸との交点は，特性方程式をK＝f(s)と表わした時のdK／ds＝0の根である。

(vii)根軌跡の出発点又は到着点における実軸との角度θは，

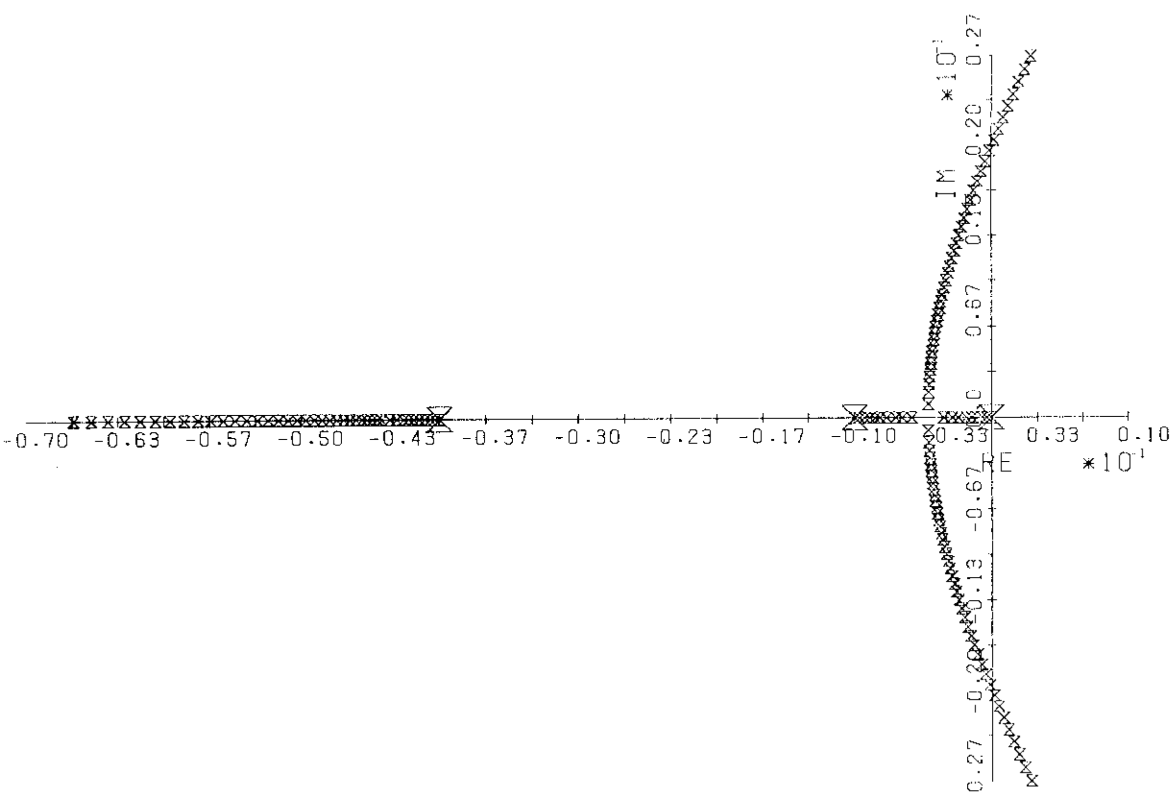
$$\theta=(2k-1)\pi-(\text{他の全ての極からの角度の和})$$
$$+(\text{他の全ての零点からの角度の和})$$

(viii)根軌跡と虚軸との交点は安定限界点であり，その点におけるKおよびsはラウス・フルビッツ，ナイキストなどの安定判別法を用いて求められる。

実 行 例

〔1〕 根軌跡の作図例（1）

```
 /DPACS/ TIME-16:06:53 CPU-00:00:02
LOCI EX12
 TO DRAW ROOT LOCUS
 SPECIFY PARAMETERS FOR DRAWING ROOT LOCUS
 1) ON WHICH ELEMENT ? (I,J) :  1,1
 2) MINIMUM VALUE OF GAIN :   0.01
 3) MAXIMUM VALUE OF GAIN :   100
 4) NO. OF CALCULATIONS :   100
 GIVE MINIMUM AND MAXIMUM OF REAL-AXIS
  MIMIMUM OF REAL = -6.65616108D+00
  MAXIMUM OF REAL =  8.28080538D-01
  MAXIMUM OF IMAGINARY =  3.78654955D+00
 RMIN, RMAX =   -7,1
 MAXIMUM OF IMAGINARY-AXIS =  2.66666667D+00
 DRAW AXIS ? _
 NORMAL TYPE ? _
 NAME OF X-AXIS :
RE
 NAME OF Y-AXIS :
IM
 TITLE :
LOCI EX12
 DRAW OTHER ELEMENT ?  N
 /DPACS/ TIME-16:18:39 CPU-00:00:27
```

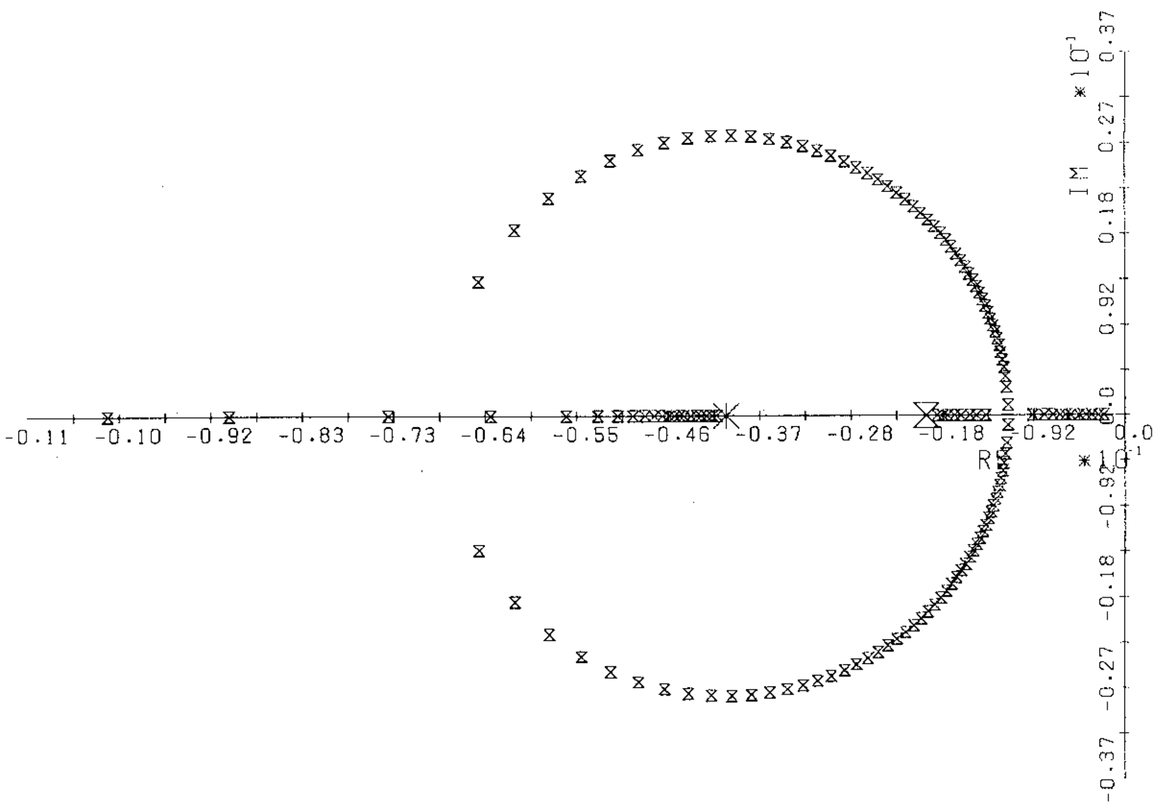


LOCI EX12

根軌跡の作図例（1）

〔2〕 根軌跡の作図例（2）

```
 /DPACS/ TIME-16:06:55 CPU-00:00:02
LOCI EX13
 TO DRAW ROOT LOCUS
 SPECIFY PARAMETERS FOR DRAWING ROOT LOCUS
 1) ON WHICH ELEMENT ? (I,J) :   1,1
 2) MINIMUM VALUE OF GAIN :   0.1
 3) MAXIMUM VALUE OF GAIN :   100
 4) NO. OF CALCULATIONS :   100
 GIVE MINIMUM AND MAXIMUM OF REAL-AXIS
  MIMIMUM OF REAL = -9.79148164D+01
  MAXIMUM OF REAL =  0.0
  MAXIMUM OF IMAGINARY =  2.82787701D+00
 RMIN, RMAX =   -11,0
 MAXIMUM OF IMAGINARY-AXIS =  3.66666667D+00
 DRAW AXIS ? __
 NORMAL TYPE ? __
 NAME OF X-AXIS :
RE
 NAME OF Y-AXIS :
IM
 TITLE :
ROOT LOCUS FOR EX13
 DRAW OTHER ELEMENT ?   N
 /DPACS/ TIME-16:18:41 CPU-00:00:27
```



ROOT LOCUS FOR EX13

根軌跡の作図例（2）

(逆) ゲシュゴリン帯の作図	DP. VI. 4
BAND $\begin{bmatrix} \# \\ /\mathrm{I} \end{bmatrix}$_SN	

機　能

システムのゲシュゴリン帯または逆ゲシュゴリン帯を描く。

◆第1スイッチが付加されなかった場合には，ゲシュゴリン帯を描く。

◆第1スイッチ／Iが付加された場合には，逆ゲシュゴリン帯を描く。

<注意事項>

図上で1mm以上離れた線だけがプロットされる。

理論概要[21)]

◆対角優勢とは

m×m行列で$\boldsymbol{Z}(s)=(Z_{ij}(s))$が与えられた時，複素平面上の領域Dにおいて全てのi＝1，～，mに対し，対角行優勢$\{|Z_{ii}|>\sum_{j=1,j\neq i}^{m}|Z_{ij}|\}$又は対角列優勢$\{|Z_{ii}|>\sum_{j=1,j\neq i}^{m}|Z_{ij}|\}$であるとき対角優勢であると言う。

$\boldsymbol{Z}(s)$が虚軸上で，対角優勢であることは，次の不等式のいずれかが，各ωについて成立していることと等価である。

$$|Z_{ii}(j\omega)|>d_i(j\omega)\equiv\sum_{j\neq i}|Z_{ij}(j\omega)|,\quad i=1,2,\cdots,m$$
$$|Z_{ii}(j\omega)|>d_i'(j\omega)\equiv\sum_{j\neq i}|Z_{ji}(j\omega)|,\quad i=1,2,\cdots,m$$

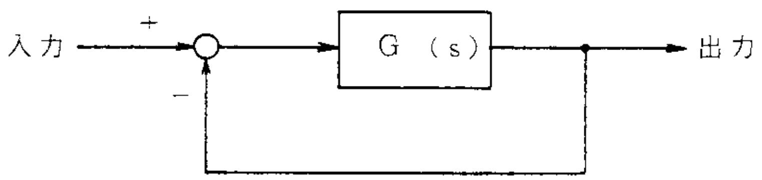
◆ゲシュゴリン帯

上式の$d_i(j\omega)$，$d_i'(j\omega)$を，それぞれ$Z(j\omega)$の行ゲシュゴリン帯，列ゲシュゴリン帯と呼ぶ。図形的な意味としては，$Z_{ii}(j\omega)$を中心として半径$d_i(j\omega)$又は$d_i'(j\omega)$の円が原点を含まないことと等価である。伝達関数行列$\boldsymbol{G}(s)$の逆行列$\hat{\boldsymbol{G}}(s)=\boldsymbol{G}(s)^{-1}$のゲシュゴリン帯は，制御対象の相互干渉の度合を示す量となる。

◆Rosenbrockの安定判別法

一巡伝達関数行列を$\boldsymbol{G}(s)$，閉ループ伝達関数行列を$\boldsymbol{T}(s)$とし，$\hat{\boldsymbol{G}}(s)=\boldsymbol{G}^{-1}(s)=(\hat{g}_{ij}(s))$，$\hat{\boldsymbol{T}}(s)=\boldsymbol{T}^{-1}(s)$とする。今，$\hat{\boldsymbol{G}}(s)$，$\hat{\boldsymbol{T}}(s)$が共に虚軸上で，対角優勢であれば，閉ループ系が安定であるための必要十分条件は$\sum_{i=1}^{m}(\tilde{n}_i-n_i)=P$である。ただし，$\tilde{n}_i$，$n_i$は$\hat{g}_{ii}(\omega)$のナイキスト線図が原点，$(-1+j\cdot0)$を回る回数，Pは，$\boldsymbol{G}(s)$の正実部を持つ極の数である。

下図のようなシステムにおいて，$\hat{\boldsymbol{G}}(s)$，$\hat{\boldsymbol{T}}(s)$が共に虚軸上で対角優勢であるということは



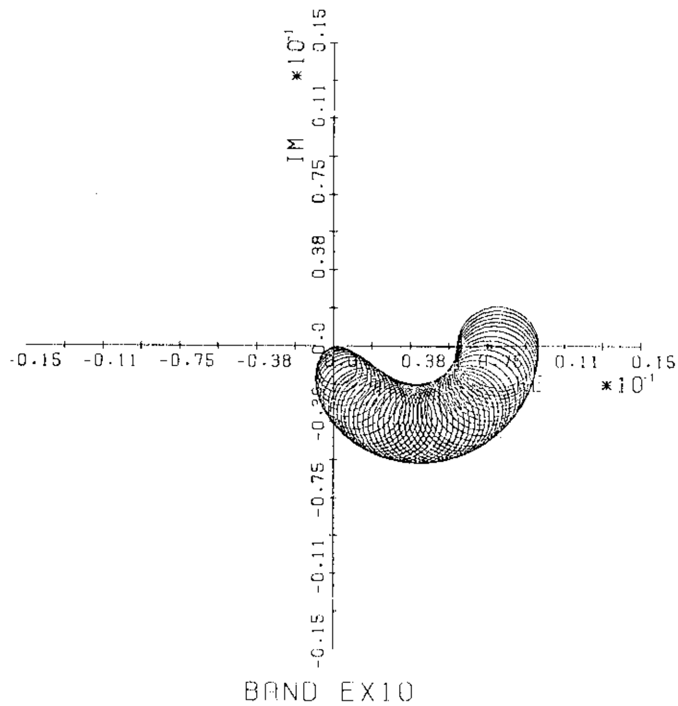
$$\hat{\boldsymbol{T}}(s)=\boldsymbol{I}+\hat{\boldsymbol{G}}(s)$$

という関係から$\hat{\boldsymbol{G}}(s)$のゲシュゴリン帯を描いた時，その円群が原点も（－１＋j・０）点も含まないことである。

実行例

〔1〕　ゲシュゴリン帯の作図例

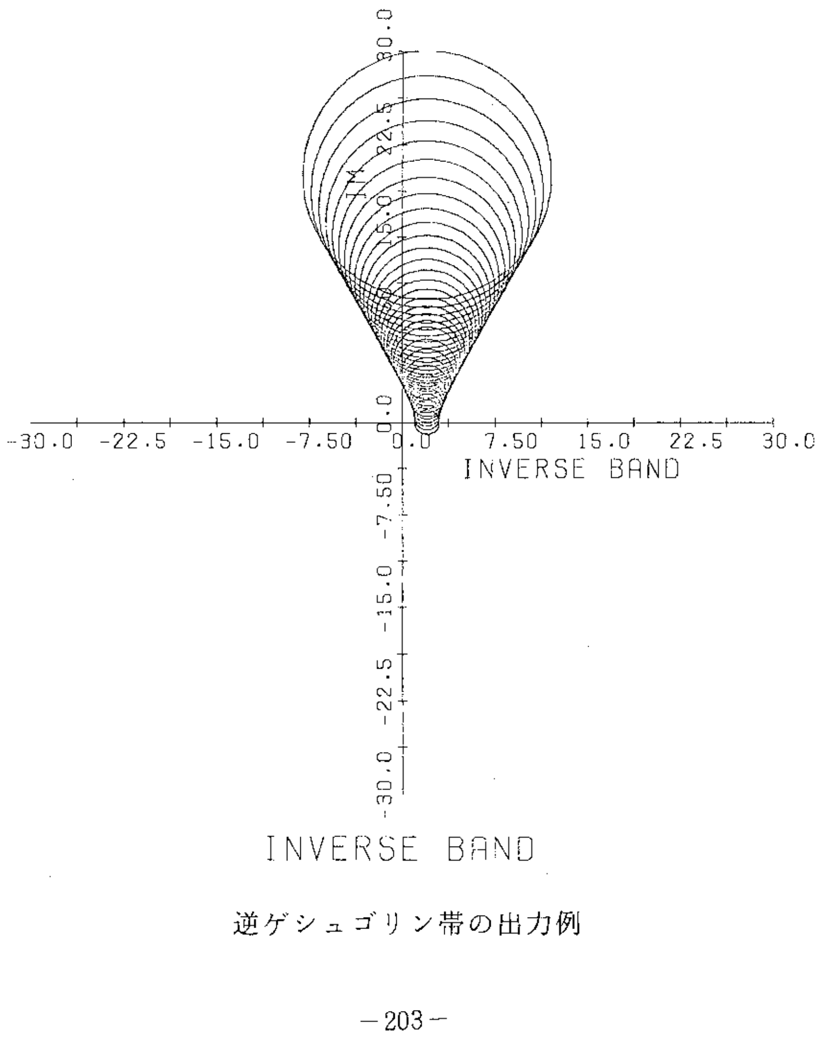
```
 /DPACS/ TIME-16:06:56 CPU-00:00:02
BAND EX10
 TO DRAW GERSHGORIN'S BAND
 WHICH DIAGRAM IS DRAWN,
  (1) ROW OR (2) COLUMN GERSHGORIN'S BAND ? :    1
 SPECIFY PARAMETERS FOR DRAWING DIAGRAM
 1) ON WHICH ELEMENT (I,I) ? :    1
 2) MINIMUM VALUE OF OMEGA :    0.01
 3) MAXIMUM VALUE OF OMEGA :    10
 4) NUMBER OF CALCULATIONS :    100
 GIVE FULL SCALE OF AXIS (MAXIMUM VALUE =  9.99924408D-01 :  1.5
 DRAW AXIS ? 
 NORMAL TYPE ? 
 NAME OF X-AXIS :
RE
 NAME OF Y-AXIS :
IM
 TITLE :
BAND EX10
 DRAW OTHER CASE ?   N
 /DPACS/ TIME-16:18:42 CPU-00:00:27
```



ゲシュゴリン帯の出力例

〔2〕 逆ゲシュゴリン帯の作図例

```
 /DPACS/ TIME-16:06:58 CPU-00:00:02
BAND/I EX10
 TO DRAW INVERSE GERSHGORIN'S BAND
 WHICH DIAGRAM IS DRAWN,
  (1) ROW OR (2) COLUMN INVERSE GERSHGORIN'S BAND ? :   2
 SPECIFY PARAMETERS FOR DRAWING DIAGRAM
 1) ON WHICH ELEMENT (I,I) ? :   1,1
 2) MINIMUM VALUE OF OMEGA :   0.01
 3) MAXIMUM VALUE OF OMEGA :   10
 4) NUMBER OF CALCULATIONS :   100
 GIVE FULL SCALE OF AXIS (MAXIMUM VALUE =  3.00498756D+01 :  30
 DRAW AXIS ? _
 NORMAL TYPE ? __
 NAME OF X-AXIS :
RE
 NAME OF Y-AXIS :
IM
 TITLE :
INVERSE BAND
 DRAW OTHER CASE ?   N
 /DPACS/ TIME-16:18:44 CPU-00:00:27
```



逆ゲシュゴリン帯の出力例

一般化最小２乗推定	ＤＰ．Ⅶ．１
ＧＬＳ＿ＳＮ	

機　　能

入出力データ表現形式のシステムデータ Σ_I（ＳＮ）を用いてＧＬＳ法（Generalized Least Square Method）を用いて，システム同定を行う。

理論概要 [8), 22)]

◆同定モデル

同定モデルは，(1)，(2)式で記述されるｍ入力ｐ出力系とする。

$$\begin{cases} \boldsymbol{T}(\xi)\,\boldsymbol{z}(t) = \boldsymbol{U}(\xi)\boldsymbol{u}(t) + \boldsymbol{\eta}(t) & \cdots\cdots\cdots\cdots (1) \\ \boldsymbol{F}(\xi)\,\boldsymbol{\eta}(t) = \boldsymbol{w}(t) & \cdots\cdots\cdots\cdots (2) \end{cases}$$

あるいは

$$\boldsymbol{T}(\xi)\,\boldsymbol{z}(t) = \boldsymbol{U}(\xi)\boldsymbol{u}(t) + \boldsymbol{E}(\xi)\boldsymbol{w}(t) \qquad \cdots\cdots\cdots\cdots (3)$$

なお $\boldsymbol{E}(\xi) = \boldsymbol{F}^{-1}(\xi)$

ここで

ξ ：delay operator　（$\xi^k \boldsymbol{z}(t) = \boldsymbol{z}(t-k)$），$\boldsymbol{u} \in \mathbf{R}^m$：入力ベクトル，
$\boldsymbol{z} \in \mathbf{R}^p$：出力ベクトル，$\boldsymbol{w} \in \mathbf{R}^p$：ノイズベクトル，$\boldsymbol{T}(\xi)$，$\boldsymbol{U}(\xi)$，$\boldsymbol{E}(\xi)$ はそれぞれ $\mathbf{R}^{p\times p}$，$\mathbf{R}^{p\times m}$，$\mathbf{R}^{p\times p}$ の大きさの ξ の多項式行列

以下に上式に対する細かな記号と，制約条件を列記する。

$$\boldsymbol{T}(\xi) = \boldsymbol{T}_0 + \boldsymbol{T}_1 \xi^1 + \cdots\cdots + \boldsymbol{T}_r \xi^r$$
$$\boldsymbol{U}(\xi) = \boldsymbol{U}_0 + \boldsymbol{U}_1 \xi^1 + \cdots\cdots + \boldsymbol{U}_r \xi^r$$
$$\boldsymbol{E}(\xi) = \boldsymbol{I} \quad + \boldsymbol{E}_1 \xi^1 + \cdots\cdots + \boldsymbol{E}_g \xi^q$$

ここで，$\boldsymbol{T}_i$，$\boldsymbol{U}_i$，$\boldsymbol{E}_i$ は，それぞれ $\mathbf{R}^{p\times p}$，$\mathbf{R}^{p\times m}$，$\mathbf{R}^{p\times p}$ の定数行列

さらに各要素ごとに表現すると

$$\begin{cases} t_{ij}(\xi) = t^0_{ij} + t^1_{ij}\xi^1 + \cdots\cdots + t^r_{ij}\xi^r & (t^0_{ij} = 1) \\ u_{ij}(\xi) = u^0_{ij} + u^1_{ij}\xi^1 + \cdots\cdots + u^r_{ij}\xi^r & \\ e_{ij}(\xi) = \delta_{ij} + e^1_{ij}\xi^1 + \cdots\cdots + e^q_{ij}\xi^q & \left(\delta_{ij} = \begin{cases} 1: & i=j \\ 0: & i \neq j \end{cases}\right) \end{cases}$$

である。ここで，t^k_{ij}．u^k_{ij}．e^k_{ij} はすべて存在するとは限らず０である場合もある。

正しい同定が行なわれるためには，

(i)　時系列 $\{\boldsymbol{w}(t)\}$ は平均値零，正則で対角な共分散行列 $\boldsymbol{\Phi}$ をもつｐ次元独立白色過程である。

(ii) $\boldsymbol{u}(t)$は正定な共分散行列をもつ covariance stationary process である。即ち，すべての j に対して正定な

$$R_j = E[\{\boldsymbol{u}(t)-E(\boldsymbol{u}(t))\}\{\boldsymbol{u}(t-j)-E(\boldsymbol{u}(t-j))\}] \quad (j=0, \pm 1, \cdots)$$

が存在する。ここでEは「平均をとる」という演算子である。

(iii) $\boldsymbol{u}$と$\boldsymbol{w}$は統計的に独立であり，さらに$\boldsymbol{w}(t)$は，$\boldsymbol{y}(t-j)(\forall j \geq 1)$とも独立である。

(iv) $\det \boldsymbol{T}(\xi)$は恒等的に零ではなく，その零点の絶対値は1より大きい。
さらに$\det \boldsymbol{T}_0 = 1$である。

が満たされなければならない。さらに1出力系においては，

(v) $[\boldsymbol{T}(\xi), \boldsymbol{U}(\xi), \boldsymbol{E}(\xi)]$の Smith canonical form[30]が$[\boldsymbol{I}\ \boldsymbol{0}]$

であればパラメータを一意に定めることができるが，多出力系では(v)の制約の下でも，パラメータの一意性が一般には保証されない。この不都合を避けるため，(1)式の$\boldsymbol{T}$，$\boldsymbol{U}$，$\boldsymbol{E}$，$\boldsymbol{\Phi}$にある制約をつけた canonical form が必要となる。これには例えば

A) $\boldsymbol{T}(\xi)$は下三角行列，$\boldsymbol{T}_0 = \boldsymbol{I}$
$\boldsymbol{U}$，$\boldsymbol{E}$，$\boldsymbol{\Phi}$は任意
$[\boldsymbol{T}(\xi), \boldsymbol{U}(\xi), \boldsymbol{E}(\xi)]$の Smith canonical form が$[\boldsymbol{I}\ \boldsymbol{0}]$

B) $\boldsymbol{E}(\xi)$は対角行列　$\boldsymbol{T}_0 = \boldsymbol{I}$
$\boldsymbol{T}_i\ (i \neq 0)$，$\boldsymbol{U}$，$\boldsymbol{\Phi}$は任意
$\{e_{ii}(\xi), t_{i1}(\xi), \cdots, t_{ip}(\xi), u_{i1}(\xi), \cdots, u_{im}(\xi)\}$　$(i=1, \cdots, p)$の最大共通因子は零次である。

C) $\boldsymbol{E}(\xi)$，$\boldsymbol{\Phi}$は対角行列
$\boldsymbol{T}_0$は下三角行列，$\boldsymbol{T}_i\ (i \neq 0)$は任意
$\{e_{ii}(\xi), t_{i1}(\xi), \cdots, t_{ip}(\xi), u_{i1}(\xi), \cdots, u_{im}(\xi)\}$　$(i=1, \cdots, p)$の最大共通因子は零次である。

等がある。従ってこれ以外の$\boldsymbol{T}$，$\boldsymbol{U}$，$\boldsymbol{E}$ の形を指定する場合には困難を有する。そこでＧＬＳの場合には，上のＢ）の形を基本とし，

(vi) $\boldsymbol{F}(\xi)$は対角行列（即ち$\boldsymbol{E}(\xi)$も対角行列）であり，(i, i)要素$f_i(\xi)(i=1, \cdots, p)$は$f_i(\xi) = 1 + \sum_{j=1}^{q_i} f_i{}^k \xi^k$でその零点の絶対値は1より大きい

を満たすものとする。ここで原理的にはq_iが無限大となる場合を含むのがＧＬＳの特徴である。

◆理論詳細

ＧＬＳは各行ごとに計算するので，(1)，(2)式もｉ行目$(i=1, \cdots, p)$について抜き出しておくと（$\boldsymbol{u}_i$は$\boldsymbol{U}$のｉ列目を意味する。）

$$\begin{cases} {}^t\boldsymbol{t}_i(\xi)\boldsymbol{z}(t) = {}^t\boldsymbol{u}_i(\xi)\boldsymbol{u}(t) + \eta_i(t) & \cdots\cdots\cdots (1)' \\ \boldsymbol{t}_i(\xi)\eta(t) = \boldsymbol{w}_i(t) & \cdots\cdots\cdots (2)' \end{cases}$$

(1)′式において未知パラメータ$\boldsymbol{\theta}_i = \{{}^t\boldsymbol{t}_i(\xi), {}^t\boldsymbol{u}_i(\xi)\}$ の代わりに${}^t\hat{\boldsymbol{t}}_i(\xi)$，${}^t\hat{\boldsymbol{u}}_i(\xi)$を用いて$\varepsilon_i(t)$を

$$\varepsilon_i(t) = {}^t\boldsymbol{t}_i(\xi)\boldsymbol{z}(t) - {}^t\boldsymbol{u}_i(\xi)\boldsymbol{u}(t)$$
$$= z_i(t) - \{({}^t\hat{\boldsymbol{t}}_i(\xi) - {}^t\boldsymbol{e}_i)(-\boldsymbol{z}(t)) + {}^t\hat{\boldsymbol{u}}_i(\xi)\boldsymbol{u}(t)\}$$
$$= z_i(t) - {}^t\boldsymbol{s}_i(t)\ \hat{\boldsymbol{\theta}}_i \qquad \cdots\cdots\cdots (4)$$

ここで$\boldsymbol{e}_i$は，i 要素目が1の単位ベクトル。

『${}^t\boldsymbol{s}_i(t) = \chi_i[-\boldsymbol{z}(t),\ \cdots,\ -\boldsymbol{z}(t-\nu_i),\ \boldsymbol{u}(t),\ \cdots,\ \boldsymbol{u}(t-\nu_i)]$』

χ_i；$\boldsymbol{z}(t)$，$\boldsymbol{u}(t)$等から${}^t\hat{\boldsymbol{t}}_i(\xi) - {}^t\boldsymbol{e}_i$，${}^t\hat{\boldsymbol{u}}_i(\xi)$に対応する部分を抜き出すオペレータ

例：$\boldsymbol{t}_i(\xi) = 1 + \boldsymbol{t}^1{}_i\xi + \boldsymbol{t}^3{}_i\xi^3$のとき

$\chi_i[\boldsymbol{z}(t),\ \boldsymbol{z}(t-1),\ \boldsymbol{z}(t-2),\ \boldsymbol{z}(t-3)] = [\boldsymbol{z}(t),\ \boldsymbol{z}(t-1),\ \boldsymbol{z}(t-3)]$

$\hat{\boldsymbol{\theta}}_i$：${}^t\hat{\boldsymbol{t}}_i(\xi)$，${}^t\hat{\boldsymbol{u}}_i(\xi)$に含まれるパラメータを${}^t\boldsymbol{s}_i(t)$に対応する順に並べた列ベクトルと定義する。このとき

$$J_N = \sum_{t=1}^{N} \varepsilon^i{}_2(t) = \sum_{t=1}^{N} \{z_i(t) - {}^t\boldsymbol{s}_i(t)\ \hat{\boldsymbol{\theta}}_i\}^2 \qquad \cdots\cdots\cdots (5)$$
$$= \|\ \overline{\boldsymbol{e}_i}\ \|^2 = \|\ \overline{\boldsymbol{z}}_i - {}^t\boldsymbol{S}_i\ \hat{\boldsymbol{\theta}}_i\ \|^2 \qquad \cdots\cdots\cdots (5)'$$

ただし，

$$\overline{\boldsymbol{e}}_i = {}^t[\varepsilon_i(1),\ \cdots,\ \varepsilon_i(N)]$$
$$\overline{\boldsymbol{z}}_i = {}^t[z_i(1),\ \cdots,\ z_i(N)]$$
$$\boldsymbol{S}_i = [{}^t\boldsymbol{s}_i(1),\ \cdots,\ {}^t\boldsymbol{s}_i(N)]$$

(5)（(5)′）式を最小にする最小2乗解は

$$\hat{\boldsymbol{\theta}}_i = \{\sum_{t=1}^{N} \boldsymbol{s}_i(t)\,{}^t\boldsymbol{s}_i(t)\}^{-1}\{\sum_{t=1}^{N} \boldsymbol{s}_i(t) z_i(t)\} \qquad \cdots\cdots\cdots (6)$$
$$= (\boldsymbol{S}_i\,{}^t\boldsymbol{S}_i)^{-1}(\boldsymbol{S}_i\overline{\boldsymbol{z}}_i) \qquad \cdots\cdots\cdots (6)'$$

で与えられる。(6)′において，測定値に関して平均をとれば

$$E[\hat{\boldsymbol{\theta}}_i] = E[(\boldsymbol{S}_i\,{}^t\boldsymbol{S}_i)^{-1}\cdot\boldsymbol{S}_i\{{}^t\boldsymbol{S}_i\ \boldsymbol{\theta}_i + \overline{\boldsymbol{\eta}}_i\}] \quad \text{((1)′式より)}$$
$$= \boldsymbol{\theta}_i + E[(\boldsymbol{S}_i\,{}^t\boldsymbol{S}_i)^{-1}\cdot\boldsymbol{S}_i\ \overline{\boldsymbol{\eta}}_i] \qquad \cdots\cdots\cdots (7)$$

ただし，

$$\overline{\boldsymbol{\eta}}_i = {}^t[\eta_i(1),\ \cdots,\ \eta_i(N)]$$

となる。(2)′式において，$\boldsymbol{f}_i(\xi)$の次数が1以上ならば$\eta_i(t)$は有色となるから，例えば$\boldsymbol{S}_i$の中の$\boldsymbol{z}_i(t-1)$と$\overline{\boldsymbol{\eta}}_i$の中の$\eta_i(t)$とは相関を持ち，従って，一般に$E(\hat{\boldsymbol{\theta}}_i) = \boldsymbol{\theta}_i$とはならない。

ところが，(1)′式の両辺に$\boldsymbol{f}_i(\xi)$をかけて

$${}^t\boldsymbol{t}_i(\xi)\boldsymbol{z}^*(t) = {}^t\boldsymbol{u}_i(\xi)\boldsymbol{u}^*(t) + \boldsymbol{w}_i(t) \qquad \cdots\cdots\cdots (8)$$

ただし，

$$\boldsymbol{z}^*(t)=\boldsymbol{f}_i(\xi)\,\boldsymbol{z}(t)$$
$$\boldsymbol{u}^*(t)=\boldsymbol{f}_i(\xi)\,\boldsymbol{u}(t)$$

とすれば$\boldsymbol{w}(t)$は白色雑音であり，(8)式の最小２乗解はbiasしない。しかし一般に$\boldsymbol{f}_i(\xi)$も未知であり，これに対しても最小２乗解を用いることとして，(8)式，(2)′式の未知パラメータ${}^t\hat{\boldsymbol{t}}_i(\xi)$，${}^t\hat{\boldsymbol{u}}_i(\xi)$，$\hat{\boldsymbol{f}}_i(\xi)$を順番に求めて順次精度を上げていくのがＧＬＳである。
また，推定パラメータの分散は（Ｖを「分散をとる」ことを意味する演算子とする），

$$\boldsymbol{f}_i(\xi)\,\boldsymbol{S}_i=\boldsymbol{S}_i^*$$

とすれば

$$\begin{aligned}V[\hat{\boldsymbol{\theta}}_i]&=E[\{\hat{\boldsymbol{\theta}}_i-E(\hat{\boldsymbol{\theta}}_i)\}^t\{\hat{\boldsymbol{\theta}}_i-E(\hat{\boldsymbol{\theta}}_i)\}]\\&=E[\{\hat{\boldsymbol{\theta}}_i-\boldsymbol{\theta}_i\}^t\{\hat{\boldsymbol{\theta}}_i-\boldsymbol{\theta}_i\}]\\&=E[(\boldsymbol{S}_i{}^t\boldsymbol{S}_i^*)^{-1}\boldsymbol{S}_i\overline{\boldsymbol{w}}_i{}^t\overline{\boldsymbol{w}}_i{}^t\boldsymbol{S}_i^*(\boldsymbol{S}^*{}_i{}^t\boldsymbol{S}_i^*)^{-1}]\\&=\sigma^2{}_i\,E[(\boldsymbol{S}_i{}^t\boldsymbol{S}^*{}_i)^{-1}]\qquad(V[\overline{\boldsymbol{w}}_i{}^t\overline{\boldsymbol{w}}_i]=\sigma^2{}_i)\end{aligned}\qquad\cdots\cdots\cdots(9)$$

と書ける。プログラムではこれを下式で代用している。

$$V[\hat{\boldsymbol{\theta}}_i]\approx\sigma_i\,(\boldsymbol{S}_i^*{}^t\boldsymbol{S}^*{}_i)^{-1}\qquad\cdots\cdots\cdots(10)$$

◆ Parametrization（固定パラメータの係数の決め方）の説明[31]
同定モデルに対して，あらかじめ，$\boldsymbol{U}_0$は$\boldsymbol{0}$であるとか，ノイズ項$\boldsymbol{E}(\xi)$は対角であるとかの知識がある場合や，同定をしていく過程において$\boldsymbol{0}$と推定されるパラメータがある場合のように，モデルのある部分のパラメータを$\boldsymbol{0}$とおいて同定を行なった方が計算上も，同定方法としても有利であることが少なくない。
このように計算を行なう前に，あらかじめ，$\boldsymbol{0}$と決めて同定しないという情報を与えることをParametrizationと呼ぶことにする。

$$\boldsymbol{T}(\xi)\,\boldsymbol{z}(t)=\boldsymbol{U}(\xi)\,\boldsymbol{u}(t)+\boldsymbol{E}(\xi)\,\boldsymbol{w}(t)$$

なるモデルに対して，$\boldsymbol{T}(\xi)$，$\boldsymbol{U}(\xi)$のParametrizationに関しては，本プログラムでは，ＡＵＴＯ，ＭＡＮＵＡＬと呼ぶ２方式がある。
さらに$\boldsymbol{E}(\xi)$（$=\boldsymbol{F}(\xi)^{-1}$）では，その次数を指定する。
まず$\boldsymbol{E}(\xi)$に関して説明する。（ｉ，ｊ）要素を$e_{ij}(\xi)$とし，$\boldsymbol{E}(\xi)$の多項式要素の最高次をｑとすると，$e_{ij}(\xi)$は，

$$e_{ij}(\xi)=\delta_{ij}+e^1{}_{ij}\xi^1+\cdots+e^q{}_{ij}\xi^q\qquad(\delta_{ij}：クロネッカのデルタ記号)$$

と書ける。ここで$e_{ij}(\xi)$の次数をｒ（$r\leq q$）と指定すれば

$$e_{ij}^{r+1},\ \cdots,\ e_{ij}^{q}$$

を0と見做したことになる。
次に$\boldsymbol{T}(\xi)$, $\boldsymbol{U}(\xi)$（（i，j）要素 $t_{ij}(\xi)$（i＝1，…，p）（j＝1，…，p），$u_{ij}(\xi)$（i＝1，…，p）（j＝1，…，m））に関するParametrization方式，AUTOとMANUALについて述べる。

(1) AUTO

Luenbergerの第1可観測正準形に基づいた入出力関係のParametrizationを行なう。この場合，observability indicesを与え，直接項の有無を指定すれば，Parametrizationは決定する。
即ち，

$$\begin{cases} t_{ij}(\xi) = t_{ij}^{\mu(i,j)}\,\xi^{\mu(i,j)} + \cdots + t_{ij}^{\nu_i}\,\xi^{\nu_i} & (t_{ii}^{0}=1) \\ u_{ij}(\xi) = (u_{ij}^{0}) + u_{ij}^{1}\xi^{1} + \cdots + u_{ij}^{\nu_i}\xi^{\nu_i} & (u_{ij}^{0}\text{は直接項がないとき}0) \end{cases}$$

$$\mu(i,j) = \begin{cases} 0 & : i>j \\ 1 & : i<j \end{cases} \Big\} \ \nu_i < \nu_j \qquad \nu_i - \nu_j + 1 \quad : \nu_i \geq \nu_j$$

である。

(2) MANUAL

$t_{ij}(\xi)$，$u_{ij}(\xi)$の各係数 t^{k}_{ij}，u^{k}_{ij}（$k=0, \cdots, \nu_i$）の有無を1, 0で指定する。ただしt^{0}_{ij}は1, 0の指定に無関係に$t^{0}_{ij}=1$である。

◆アルゴリズム

Step 1 ： $q=0$, $\hat{\boldsymbol{f}}_1^{(0)}=\boldsymbol{1}$, $\boldsymbol{S}_i^{*(0)}=\boldsymbol{S}_i$, $\overline{\boldsymbol{z}}_i^{*(0)}=\overline{\boldsymbol{z}}_i$, $\hat{\boldsymbol{\theta}}_i^{(0)}=\boldsymbol{0}$, $\hat{\boldsymbol{\beta}}_i^{(0)}=\boldsymbol{0}$,

Step 2 ： $q=q+1$

Step 3 ： $\hat{\boldsymbol{\theta}}_i^{(q)}=(\boldsymbol{S}_i^{*(q)}\,{}^{t}\boldsymbol{S}_i^{*(q)})^{+}\,\boldsymbol{S}_i^{*(q)}\overline{\boldsymbol{z}}_i^{*(q)}$ （＋…擬似逆行列）

$\boldsymbol{S}_i^{*(q)}=\hat{\boldsymbol{f}}_i^{(q-1)}$, $\overline{\boldsymbol{z}}_i^{*(q)}=\hat{\boldsymbol{f}}_i^{(q-1)}\overline{\boldsymbol{z}}_i^{*(q-1)}$

Step 4 ： $\overline{\boldsymbol{w}}_i^{(q)}=\overline{\boldsymbol{z}}_i^{*(q)}-\boldsymbol{S}_i^{*(q)}\hat{\boldsymbol{\theta}}_i^{(q)}$, $\boldsymbol{\eta}_i^{(q)}=\overline{\boldsymbol{w}}_i^{(q)}$

$(\hat{\sigma}_i^{(q)})^2=\dfrac{1}{N}\,{}^{t}\overline{\boldsymbol{w}}_i^{(q)}\overline{\boldsymbol{w}}_i^{(q)}$

Step 5 ： $\boldsymbol{\eta}_i^{(q)}(t)=\hat{\boldsymbol{\beta}}_i\boldsymbol{E}_i^{(q)}(t)$

$\boldsymbol{E}_i^{(q)}(t)={}^{t}[-\boldsymbol{\eta}_i^{(q)}(t-1), \cdots -\boldsymbol{\eta}_i^{(q)}(t-r_i)]$

の最小2乗推定値を$\hat{\boldsymbol{\beta}}_i^{(q)}$（$\hat{f}_i^{(q)}(\xi)$の係数）とする。

Step 6 ： $\delta\hat{\theta}_i^{(q)}=\|\hat{\boldsymbol{\theta}}_i^{(q)}-\hat{\boldsymbol{\theta}}_i^{(q-1)}\|$, $\delta\hat{\beta}_i^{(q)}=\|\hat{\boldsymbol{\beta}}_i^{(q)}-\hat{\boldsymbol{\beta}}_i^{(q-1)}\|$とし

（ただし$\|\boldsymbol{x}\|=\sum_{i=1}^{n}|x_i|$）

$\delta\hat{\theta}_i^{(q)}<\varepsilon_1$かつ$\delta\hat{\beta}_i^{(q)}<\varepsilon_2$ならばi行目の推定は終了

Step 7 ： Step 2へ

以上は無限フィルタのアルゴリズムである。

Step 3において，$\boldsymbol{S}_i{}^{*(q)}=\hat{\boldsymbol{f}}_i{}^{(q-1)}\boldsymbol{S}_i$，$\overline{\boldsymbol{z}}_i{}^{*(q)}=\hat{\boldsymbol{f}}_i{}^{(q-1)}\overline{\boldsymbol{z}}_i$，Step 4において，$\overline{\boldsymbol{w}}_i{}^{(q)}=\overline{\boldsymbol{z}}_i-\boldsymbol{S}_i\hat{\boldsymbol{\theta}}_i{}^{(q)}$ とすれば有限フィルタである。

またStep 3において逆行例の代わりに擬似逆行列を用いたのは冗長なParametrizationの場合でも解を持たせるためである。

実行例

〔1〕 一般化最小2乗推定法によるシステムの同定例

```
/DPACS/ TIME-16:07:28 CPU-00:00:02
GLS PIP2
TO IDENTIFY SYSTEM BY G.L.S.
SPECIFY THE FOLLOWING PARAMETERS FOR G.L.S. :
1)  2 OBSERVABILITY INDICES :   1,1 ①
2)  2 DEGREES IN EACH ROW OF F(.) :   1,1 ②
3)  STARTING DATA-POINT ( 3-2100) :   3 ③
4)  DATA-LENGTH USED FOR G.L.S. (1-2098) :   2098 ④
5)  WHICH TYPE OF FILTER IS USED,
     (1) INFINITY FILTER OR (2) FINITE FILTER ? :   1 ⑤
6)  CONVERGENCE CRITERION (EPSA,EPSB) :   0.05,0.05 ⑥
7)  NO. OF ITERATION OF OPTIMIZATION :   10 ⑦
8)  COEFFICIENT PATTERN ⑧
 (1) AUTO , (2) MANUAL :   1
 (1) (U(0).NE.0) , (2) (U(0)=0) :   1
---------------------------------------------------------------------------
   G.L.S. DRIVING PARAMETERS FOR SYSTEMDATA PIP2.IO
---------------------------------------------------------------------------
1)  2 OBSERVABILITY INDICES :     1  1
2)  2 DEGREES IN EACH ROW OF F(.) :    1  1
3)  STARTING DATA-POINT :      3
4)  DATA-LENGTH USED FOR G.L.S. :  2098
5)  TYPE OF FILTER :
***
INFINITY FILTER
6)  CONVERGENCE CRITERION (EPSA,EPSB) :    5.000000000D-02  5.000000000D-02
7)  NO. OF ITERATION OF OPTIMIZATION :    10
8)  COEFFICIENT PATERN
     1-TH ROW OF T :     3  4
     1-TH ROW OF U :     1  2  3  4
     2-TH ROW OF T :     3  4
     2-TH ROW OF U :     1  2  3  4
---------------------------------------------------------------------------
( 1)-TH OUTPUT-SUBSYSTEM'S ESTIMATION
---------------------------------------------------------------------------
---------------------------------------------------------------------------
(  1)-TH ITERATION FOR 1-TH OUTPUT-SUBSYSTEM
---------------------------------------------------------------------------
--- ( 1, 2) MATRIX  T0       ---
( 1)  1.000000000D+00  0.0
--- ( 1, 2) MATRIX  T1       ---
( 1)  1.230959780-14 -3.913536160-15
--- ( 1, 2) MATRIX  U0       ---
( 1)  5.000000000D-01  5.000000000D-01
--- ( 1, 2) MATRIX  U1       ---
( 1)  3.941291740-15  3.941291740-15
--- ( 1, 1) MATRIX FILTER    ---
***
( 1)  0.0
--- ( 2, 1) MATRIX  COR.     ---
```

```
( 1)  3.800471510-32
( 2)  1.000000000D+00
D(ALFA) =  1.000000000D+00      D(BETA) =  0.0

--- ( 1, 6) MATRIX A.VAR.   ---
( 1)  2.722019180-03  8.908445250-18  3.89988591D+00  3.687453610-21
( 1)  4.569351530-38  4.049004260-70
-----------------------------------------------------------------------

( 2)-TH OUTPUT-SUBSYSTEM'S ESTIMATION
-----------------------------------------------------------------------
-----------------------------------------------------------------------
(  1)-TH ITERATION FOR 2-TH OUTPUT-SUBSYSTEM
-----------------------------------------------------------------------
--- ( 1, 2) MATRIX  TO          ---
( 1)  0.0              1.000000000D+00
--- ( 1, 2) MATRIX  T1          ---
( 1)  1.230959780-14 -3.913536160-15
--- ( 1, 2) MATRIX  U0          ---
( 1)  5.000000000-01  5.000000000-01
--- ( 1, 2) MATRIX  U1          ---
( 1)  3.941291740-15  3.941291740-15
--- ( 1, 1) MATRIX FILTER   ---
( 1)  0.0
--- ( 2, 1) MATRIX  COR.        ---
( 1)  3.800471510-32
( 2)  1.000000000D+00
D(ALFA) =  1.000000000D+00     D(BETA) =  0.0

--- ( 1, 6) MATRIX A.VAR.   ---
( 1)  2.722019180-03  8.908445250-18  3.89988591D+00  3.687453610-21
( 1)  4.569351530-38  4.049004260-70
-----------------------------------------------------------------------
   PARAMETERS IDENTIFIED BY G.L.S.
-----------------------------------------------------------------------
--- ( 2, 2) MATRIX  TO          ---
( 1)  1.000000000D+00  0.0
( 2)  0.0              1.000000000D+00
--- ( 2, 2) MATRIX  T1          ---
( 1)  1.230959780-14 -3.913536160-15
( 2)  1.230959780-14 -3.913536160-15
--- ( 2, 2) MATRIX  U0          ---
( 1)  5.000000000-01  5.000000000-01
( 2)  5.000000000-01  5.000000000-01
--- ( 2, 2) MATRIX  U1          ---
( 1)  3.941291740-15  3.941291740-15
( 2)  3.941291740-15  3.941291740-15
OUTPUT-MODE OF ( 2, 1) MATRIX FILTER    : _
--- ( 2, 1) MATRIX FILTER   ---
( 1)  0.0
( 2)  0.0
FILENAME : _
OUTPUT-MODE OF ( 2, 1) MATRIX SIGMA     : _
--- ( 2, 1) MATRIX SIGMA    ---
( 1)  1.000000000D+00
( 2)  3.800471510-32
FILENAME : _
OUTPUT-MODE OF ( 2, 1) MATRIX  DALF     : _
--- ( 2, 1) MATRIX  DALF    ---
( 1)  1.000000000D+00
( 2)  1.000000000D+00
FILENAME : _
OUTPUT-MODE OF ( 2, 1) MATRIX  DBET     : _
--- ( 2, 1) MATRIX  DBET    ---
( 1)  0.0
( 2)  0.0
FILENAME : _
OUTPUT-MODE OF ( 2, 6) MATRIX A.VAL.    : _
```

```
--- ( 2, 6) MATRIX A.VAL.   ---
( 1)  2.72201918D-03  8.90844525D-18  3.89988591D+00  3.68745361D-21
( 2)  2.72201918D-03  8.90844525D-18  3.89988591D+00  3.68745361D-21
( 1)  4.56935153D-38  4.04900426D-70
( 2)  4.56935153D-38  4.04900426D-70
FILENAME : _
GIVE COMMENT FOR SYSTEMDATA :
DPACS GLS
/DPACS/ TIME-16:18:46 CPU-00:00:27
```

【説明】

m入力p出力のシステムの同定を行うとする。（$1 \leq p, m \leq 6$）

① p個の可観測指数 ν_i $(i=1, \cdots, p)$ を入力する。

ただし $\begin{cases} 0 \leq \nu_i \leq 10 & (i=1, \cdots, p) \\ \nu_i \text{の最大値} \nu \text{は} (\nu+1)\cdot(p+m) \leq 50 \end{cases}$

ここで可観測指数とは，何時間ステップ遅れ分の影響が今の時間ステップに現われるかを示す指数である。例えば1次遅れシステムであれば，可観測指数＝1となる。

② $\boldsymbol{F}(\xi)$の各対角要素の時間ステップ遅れ次数 δf を入力する。この次数とは，⑤で有限フィルタを指定した場合には，その次数となる。無限フィルタを指定した場合には，その設定は逐次増加する次数の増加分を意味する。ただし，$1 \leq \delta f \leq 10$，（δf の最大値）$\times p \leq 50$である。

③ 入出力データの使用開始点を指定する。（$(\nu+\delta f+1)$ 以上）

④ GLSの計算に用いるデータ長を指定する。

⑤ ノイズフィルタとして，同定の繰り返しの都度次数を増加させる無限フィルタを使用するか，次数固定のフィルタである有限フィルタを使用するかの選択である。

⑥ 固定すべきパラメータ，及びフィルタの係数の収束判定値（EPSA，EPSB）の設定である。この収束判定に合格するか，⑦で設定する繰り返し最大値に達した場合に計算を終了する。

⑦ 同定の繰り返し計算回数を設定する。

⑧ Parametrization を指定する。（理論概要の項参照）

<注意> 同定モデルの未知パラメータの数の合計は，50を越えてはならない。

条件付き最尤推定	DP. Ⅶ. 2
CML_SN	

機　能

入出力データ表現形式のシステムデータΣ_1（SN）を用いてCML法（Conditional Maximum Likelihood Method）によりシステム同定を行う。

理論概要

文献(8)，(22) を参照のこと。

実行例

〔1〕 条件付き最尤推定法によるシステムの同定例

```
/DPACS/ TIME-16:07:01 CPU-00:00:02
CML CMLD2
 TO IDENTIFY SYSTEM BY C.M.L.
 SPECIFY THE FOLLOWING PARAMETERS FOR C.M.L. :
 1)  1 OBSERVABILITY INDICES :   1
 2)  DEGREES OF E(.)
     1  DEGREES IN 1-TH ROW OF E(.) :   1
 3)  DEGREES OF F(.)
      1 DEGREES IN 1-TH ROW OF F(.) :   1
 4)  USE INPUT-DATA ?  N
 5)  STARTING DATA-POINT (0-2099) :   0
 6)  DATA-LENGTH USED FOR INITIALIZATION : (2-2100) : 100
 7)  DATA-LENGTH USED FOR C.M.L. : (2-2100)   2100
 8)  CONVERGENCE CRITERION (EPS) :   0.005
 9)  NO. OF REPETITION OF NEWTON-RAPSON METHOD :   10
 10) TYPE OUT INFORMATION MATRIX EVERY REPETITION ?  N
 12) COEFFICIENT PATTERN
      (1) AUTO , (2) MANUAL :   1
 ----------------------------------------------------------------
    C.M.L. DRIVING PARAMETERS FOR SYSTEMDATA CMLD2.IO
 ----------------------------------------------------------------
 1)  1 OBSERVABILITY INDICES :   1
 ***
 2)  DEGREES OF E(.)
      1-TH ROW OF E :     1
 3)  DEGREES OF F(.)
      1-TH ROW OF F :     1
 4)  STARTING DATA-POINT :     0
 5)  DATA-LENGTH USED FOR INITIALIZATION :  100
 6)  DATA-LENGTH USED FOR C.M.L. : 2100
 7)  CONVERGENCE CRITERION (EPS) :   5.000000000D-03
 8)  NO. OF REPETITION OF NEWTON-RAPSON METHOD :   10
 9)  COEFFICIENT PATTERN
      1-TH ROW OF T :   2
 10) NO. OF PARAMETERS :     2
 ----------------------------------------------------------------
    INITIAL PARAMETERS FOR C.M.L. (BY L.S.)
 ----------------------------------------------------------------
 --- ( 1, 1) MATRIX  TO      ---
 ( 1)  1.000000000D+00
 --- ( 1, 1) MATRIX  T1      ---
 ( 1) -9.563869900D-01
 --- ( 1, 1) MATRIX  E1      ---
 ( 1)  1.825463940D-01
 --- ( 1, 1) MATRIX  COV.    ---
 ( 1)  5.158785880D-03
```

```
--------------------------------------------------------------------------
( 1)   PARAMETERS IDENTIFIED BY C.M.L.   (EPS =   2.61144139D-01)
--------------------------------------------------------------------------
--- ( 1, 1) MATRIX  T0        ---
( 1)  1.00000000D+00
--- ( 1, 1) MATRIX  T1        ---
( 1) -9.86931094D-01
--- ( 1, 1) MATRIX  E1        ---
( 1)  4.41898122D-01
--- ( 1, 1) MATRIX  COV.      ---
( 1)  3.58541816D-03
--------------------------------------------------------------------------
( 2)   PARAMETERS IDENTIFIED BY C.M.L.   (EPS =   5.46889309D-02)
--------------------------------------------------------------------------
--- ( 1, 1) MATRIX  T0        ---
( 1)  1.00000000D+00
--- ( 1, 1) MATRIX  T1        ---
( 1) -9.97856423D-01
--- ( 1, 1) MATRIX  E1        ---
( 1)  3.88311591D-01
--- ( 1, 1) MATRIX  COV.      ---
( 1)  3.50338392D-03
--------------------------------------------------------------------------
( 3)   PARAMETERS IDENTIFIED BY C.M.L.   (EPS =   1.81425051D-03)
--------------------------------------------------------------------------
--- ( 1, 1) MATRIX  T0        ---
( 1)  1.00000000D+00
--- ( 1, 1) MATRIX  T1        ---
( 1) -9.97478766D-01
--- ( 1, 1) MATRIX  E1        ---
( 1)  3.86537083D-01
--- ( 1, 1) MATRIX  COV.      ---
( 1)  3.50326937D-03
OUTPUT-MODE OF ( 1, 1) MATRIX  E0        : _
--- ( 1, 1) MATRIX  E0        ---
( 1)  1.00000000D+00
FILENAME : _
OUTPUT-MODE OF ( 1, 1) MATRIX  E1        : _
--- ( 1, 1) MATRIX  E1        ---
( 1)  3.86537083D-01
FILENAME : _
OUTPUT-MODE OF ( 1, 1) MATRIX  COV.      : _
--- ( 1, 1) MATRIX  COV.      ---
( 1)  3.50326937D-03
FILENAME : _
OUTPUT-MODE OF ( 2, 2) MATRIX INFORM     : _
--- ( 2, 2) MATRIX INFORM   ---
( 1)  2.40270494D-06  4.72308000D-02
( 2)  4.72308000D-02  4.05979287D-04
FILENAME : _
OUTPUT-MODE OF ( 1,2099) TIME SERIES  REG      : F
OUTPUT-FILENAME : _
GIVE COMMENT FOR SYSTEMDATA :
CML CMLD2
/DPACS/ TIME-16:18:47 CPU-00:00:27
```

対称型条件付き最尤推定	DP. Ⅶ. 3
SML_SN	

機　能

入出力データ表現形式のシステムデータΣ_1（SN）を用いて，入出力関係が対称な伝達関数行列で与えられ，さらにそのマルコフパラメータの総和がすべての要素とも等しい値になるという制約条件の元で，CML法によるシステム同定を行う。

理論概要

文献(8)，(22)参照のこと。

実 行 例

〔1〕　対称型モデルの条件付き最尤推定法によるシステムの同定例

```
/DPACS/ TIME-16:07:02 CPU-00:00:02
SML SMLD3A
 TO IDENTIFY SYSTEM BY S.M.L.
 SPECIFY THE FOLLOWING PARAMETERS FOR S.M.L. :
 1)  DEGREE OF TU :  2
 2)  DEGREES OF E(.)
      1 DEGREES IN 1-TH ROW OF E(.) :   1
 3)  DEGREES OF F(.)
      1 DEGREES IN 1-TH ROW OF F(.) :   1
 4)  STARTING DATA-POINT (0-2098) :   20
 5)  DATA-LENGTH USED FOR INITIALIZATION (2-2079) :  30
 6)  DATA-LGTH USED FOR S.M.L. (2-2079) :   200
 7)  CONVERGENCE CRITERION (EPS) :   0.005
 8)  NO. OF REPETITION OF NEWTON-RAPSON METHOD :   1
 9) TYPE OUT INFORMATION MATRIX EVERY REPETITION ?   N
 10) COEFFICIENTS PATTERN
      (1) AUTO , (2) MANUAL :   1
      (1) (U(0).NE.0) , (2) (U(0)=0) :   1
----------------------------------------------------------------
    S.M.L. DRIVING PARAMETERS FOR SYSTEMDATA SMLD3A.IO
----------------------------------------------------------------
 1)  DEGREE OF TU :   2
 2)  DEGREES OF E(.)
***
      1-TH ROW OF E :     1
 3)  DEGREES OF F(.)
      1-TH ROW OF F :     1
 4)  STARTING DATA-POINT :   20
 5)  DATA-LENGTH USED FOR INITIALIZATION :   30
 6)  DATA-LENGTH USED FOR S.M.L. :  200
 7)  CONVERGENCE CRITERION (EPS) :    5.000000000D-03
 8)  NO. OF REPETITION OF NEWTON-RAPHSON METHOD :     1
 9)  COEFFICIENT PATTERN
          T          :    2  3
      1-TH ROW OF U :    1  2  3
```

```
10) NO. OF PARAMETERS :      6
-----------------------------------------------------------------
   INITIAL PARAMETERS FOR S.M.L. (BY L.S.)
-----------------------------------------------------------------
--- ( 1, 3) MATRIX    T       ---
( 1)  1.000000000D+00 -7.643345640-01 -2.65551273D-01
--- ( 1, 1) MATRIX  U0        ---
( 1) -3.38088374D-01
--- ( 1, 1) MATRIX  U1        ---
( 1)  5.20715389D-01
--- ( 1, 1) MATRIX  U2        ---
( 1) -2.09873090D-01
--- ( 1, 1) MATRIX  KAPPA     ---
( 1)  9.11671813D-01
--- ( 1, 1) MATRIX  E1        ---
( 1) -9.42366480D-02
--- ( 1, 1) MATRIX  COV.      ---
( 1)  1.54493624D-04
-----------------------------------------------------------------
( 1)   PARAMETERS IDENTIFIED BY S.M.L.   (EPS =   3.18076644D+00)
-----------------------------------------------------------------
--- ( 1, 3) MATRIX    T       ---
( 1)  1.000000000D+00 -1.35532097D+00  4.78955548D-01
--- ( 1, 1) MATRIX  U0        ---
( 1) -2.14768714D-01
--- ( 1, 1) MATRIX  U1        ---
( 1)  3.58383995D-01
--- ( 1, 1) MATRIX  U2        ---
( 1)  3.40703749D-01
--- ( 1, 1) MATRIX  KAPPA     ---
( 1)  3.91734299D+00
--- ( 1, 1) MATRIX  E1        ---
( 1) -4.65856537D-01
--- ( 1, 1) MATRIX  COV.      ---
( 1)  4.27971004D-03
OUTPUT-MODE OF ( 1, 1) MATRIX  E0       : _
--- ( 1, 1) MATRIX  E0        ---
( 1)  1.000000000D+00
FILENAME : _
OUTPUT-MODE OF ( 1, 1) MATRIX  E1       : _
--- ( 1, 1) MATRIX  E1        ---
( 1) -4.65856537D-01
FILENAME : _
OUTPUT-MODE OF ( 1, 1) MATRIX  COV.     : _
--- ( 1, 1) MATRIX  COV.      ---
( 1)  4.27971004D-03
FILENAME : _
OUTPUT-MODE OF ( 6, 6) MATRIX INFORM    : _
--- ( 6, 6) MATRIX INFORM   ---
( 1)  3.27734436D-01 -9.68577199D-01 -9.23365174D-02 -2.04572357D-01
( 2) -9.68577199D-01  2.95144152D-01  1.25936562D-01  1.54526650D-01
( 3) -9.23365174D-02  1.25936562D-01  1.54687169D-02 -8.16559365D-01
( 4) -2.04572357D-01  1.54526650D-01 -8.16559365D-01  4.18176946D-02
( 5) -1.07796322D-01 -1.32698894D-01  2.50876084D-03  6.82579298D-02
( 6)  6.34430188D-01 -5.91961045D-01  3.05560718D-01 -4.76341305D-01
( 1) -1.07796322D-01  6.34430188D-01
( 2) -1.32698894D-01 -5.91961045D-01
( 3)  2.50876084D-03  3.05560718D-01
( 4)  6.82579298D-02 -4.76341305D-01
```

```
( 5)  1.867207090+01  5.58473817D-02
( 6)  5.58473817D-02  4.79740213D-02
FILENAME : _
OUTPUT-MODE OF ( 1, 199) TIME SERIES  REG     : F
OUTPUT-FILENAME : _
GIVE COMMENT FOR SYSTEMDATA :
DPACS SML
/DPACS/ TIME-16:18:53 CPU-00:00:27
```

制限情報下最尤推定	DP. Ⅶ. 4
LI_SN	

機　　能

入出力データ表現形式のシステムデータ Σ_1（SN）を用いて，入出力関係のLI法（Limited Information Method）によるシステム同定を行う。

（注）LI推定値は，CML推定値と同様の計算を，各行独立に行うため，推定値としては，CML推定値より良くなることは期待できないが，反面，一度に同定するパラメータ数は，入出力関係のある1つの行に含まれる数しかないため，計算機のメモリ上の制約がある場合でも遥かに多くのパラメータを持つモデルに適用できる。

理論概要

文献(8)，(22) を参照のこと。

実 行 例

〔1〕　制限情報下最尤推定法によるモデルの同定例

```
/DPACS/ TIME-16:07:26 CPU-00:00:02
LI LI2
TO IDENTIFY SYSTEM BY L.I.
SPECIFY THE FOLLOWING PARAMETERS FOR L.I. :
1)  2 OBSERVABILITY INDICES :   1,1
2)  2 DEGREES IN EACH ROW OF E(.) :   1,1
3)    DEGREES IN EACH ROW OF F(.) :   1,1
4)  USE INPUT-DATA ? _
5)  STARTING DATA-POINT (0-2099) :   0
6)  DATA-LENGTH USED FOR INITIALIZATION (2-2100) :  2100
7)  DATA-LENGTH USED FOR L.I. (2-2100) :  2100
8)  CONVERGENCE CRITERION (EPS) :   0.01
9)  NO. OF REPETITION OF NEWTON-RAPSON METHOD :  3
10) TYPE OUT INFORMATION MATRIX EVERY REPETITION ? _
11) COEFFICIENT PATTERN
     (1) AUTO , (2) MANUAL :   1
     (1) (U(0).NE.0) , (2) (U(0)=0) :   2
--------------------------------------------------------------------
    L.I. DRIVING PARAMETERS FOR SYSTEMDATA LI2.IO
--------------------------------------------------------------------
1)  2 OBSERVABILITY INDICES :   1  1
***
2)  2 DEGREES IN EACH ROW OF E(.) :   1  1
```

```
 3)  2 DEGREES IN EACH ROW OF F(.) :   1  1
 4)  STARTING DATA-POINT :    0
 5)  DATA-LENGTH USED FOR INITIALIZATION : 2100
 6)  DATA-LENGTH USED FOR L.I. : 2100
 7)  CONVERGENCE CRITERION (EPS) :   1.00000000D-02
 8)  NO. OF REPETITION OF NEWTON-RAPSON METHOD :    3
 9)  COEFFICIENT PATTERN
      1-TH ROW OF T :   3  4
      1-TH ROW OF U :   3  4
      2-TH ROW OF T :   3  4
      2-TH ROW OF U :   3  4
10) NOS. OF PARAMETERS IN EACH ROW :    5  5
---------------------------------------------------------------------
   INITIAL PARAMETERS (1-TH ROW) FOR L.I.   (BY L.S.)
---------------------------------------------------------------------
--- ( 1, 2) MATRIX  TO      ---
( 1)  1.00000000D+00  0.0
--- ( 1, 2) MATRIX  T1      ---
( 1) -9.99860286D-01  7.27204378D-04
--- ( 1, 2) MATRIX  U0      ---
( 1)  0.0             0.0
--- ( 1, 2) MATRIX  U1      ---
***
( 1)  3.03656523D-03  2.21356596D-03
--- ( 1, 2) MATRIX    E     ---
( 1)  1.00000000D+00 -4.38151264D-01
--- ( 1, 1) MATRIX  COV.    ---
( 1)  7.09235315D-05
--- ( 5, 5) MATRIX INFORM   ---
( 1)  6.13010747D-09  7.26275015D-01  5.33021668D-01 -6.01047403D-01
( 2)  7.26275015D-01  8.56220375D-07  8.98966092D-01 -8.07484030D-01
( 3)  5.33021668D-01  8.98966092D-01  8.75110197D-08 -8.78154121D-01
( 4) -6.01047403D-01 -8.07484030D-01 -8.78154121D-01  3.67301476D-07
( 5)  7.84633486D-02  8.24591436D-02  9.03745269D-02 -1.30221204D-01
( 1)  7.84633486D-02
( 2)  8.24591436D-02
( 3)  9.03745269D-02
( 4) -1.30221204D-01
( 5)  4.67088357D-04

---------------------------------------------------------------------
( 1)   PARAMETERS (1-TH ROW) IDENTIFIE D BY
                           L.I.   (EPS =   1.79537505D-01)
---------------------------------------------------------------------
--- ( 1, 2) MATRIX  TO      ---
( 1)  1.00000000D+00  0.0
--- ( 1, 2) MATRIX  T1      ---
***
( 1) -9.99929446D-01 -2.40409529D-04
--- ( 1, 2) MATRIX  U0      ---
( 1), 0.0             0.0
--- ( 1, 2) MATRIX  U1      ---
( 1)  2.68284858D-03  3.08383348D-03
--- ( 1, 2) MATRIX    E     ---
( 1)  1.00000000D+00 -6.17683691D-01
--- ( 1, 1) MATRIX  COV.    ---
( 1)  6.76157477D-05
--- ( 5, 5) MATRIX INFORM   ---
( 1)  2.96276038D-09  7.52484722D-01  5.81648091D-01 -6.44366782D-01
( 2)  7.52484722D-01  4.30975584D-07  9.11721072D-01 -8.30800637D-01
( 3)  5.81648091D-01  9.11721072D-01  4.50129593D-08 -8.94182971D-01
( 4) -6.44366782D-01 -8.30800637D-01 -8.94182971D-01  2.06126832D-07
( 5)  2.41271185D-01  2.76171481D-01  3.00459185D-01 -3.86903635D-01
( 1)  2.41271185D-01
( 2)  2.76171481D-01
( 3)  3.00459185D-01
( 4) -3.86903635D-01
```

```
( 5)  2.94275672D-04
----------------------------------------------------------------------
( 2)   PARAMETERS (1-TH ROW) IDENTIFIE D BY
                         L.I.   (EPS =    7.64547236D-02)
***
----------------------------------------------------------------------
--- ( 1, 2) MATRIX  T0       ---
( 1)  1.00000000D+00  0.0
--- ( 1, 2) MATRIX  T1       ---
( 1) -9.99954017D-01 -7.42290825D-04
--- ( 1, 2) MATRIX  U0       ---
( 1)  0.0             0.0
--- ( 1, 2) MATRIX  U1       ---
( 1)  2.49827266D-03  3.42098828D-03
--- ( 1, 2) MATRIX    E      ---
( 1)  1.00000000D+00 -5.41231584D-01
--- ( 1, 1) MATRIX  COV.     ---
( 1)  6.67791596D-05
--- ( 5, 5) MATRIX INFORM    ---
( 1)  3.99056704D-09  7.36839944D-01  5.52553275D-01 -6.19074996D-01
( 2)  7.36839944D-01  5.65700331D-07  9.04083308D-01 -8.16324972D-01
( 3)  5.52553275D-01  9.04083308D-01  5.82114320D-08 -8.83774919D-01
( 4) -6.19074996D-01 -8.16324972D-01 -8.83774919D-01  2.54092477D-07
( 5)  1.52677229D-01  1.63253290D-01  1.75687441D-01 -2.52189804D-01
( 1)  1.52677229D-01
( 2)  1.63253290D-01
( 3)  1.75687441D-01
( 4) -2.52189804D-01
***
( 5)  3.61669598D-04
```

⋮

この間同定を各行毎に，繰り返し回数を越えるか，同定精度を達成するまでくり返す。

⋮

```
--------------------------------------------------------------------------
   PARAMETERS IDENTIFIED BY L.I.
--------------------------------------------------------------------------
***
--- ( 2, 2) MATRIX  T0        ---
( 1)  0.0              1.000000000D+00
( 2)  0.0              0.0
--- ( 2, 2) MATRIX  T1        ---
( 1)  4.940376620-04 -9.935651420-01
( 2)  0.0              0.0
--- ( 2, 2) MATRIX  U0        ---
( 1)  0.0              0.0
( 2)  0.0              0.0
--- ( 2, 2) MATRIX  U1        ---
( 1)  1.916449170-03 -4.025885610-03
( 2)  0.0              0.0
--- ( 2, 2) MATRIX  E1        ---
( 1) -8.979506250-01  0.0
( 2)  0.0              0.0
--- ( 2, 2) MATRIX  COV.      ---
( 1)  1.247367620-03  0.0
( 2)  0.0              0.0
OUTPUT-MODE OF ( 2, 2) MATRIX  E0        : _
--- ( 2, 2) MATRIX  E0        ---
( 1)  1.000000000D+00  0.0
( 2)  0.0              1.000000000D+00
FILENAME : _

OUTPUT-MODE OF ( 2, 2) MATRIX  E1        : _
--- ( 2, 2) MATRIX  E1        ---
( 1) -8.979506250-01  0.0
( 2)  0.0              0.0
FILENAME : _
OUTPUT-MODE OF ( 2, 2) MATRIX  COV.      : _
--- ( 2, 2) MATRIX  COV.      ---
( 1)  1.247367620-03  0.0
( 2)  0.0              0.0
FILENAME : _
OUTPUT-MODE OF ( 2,2099) TIME SERIES  REG       : F
OUTPUT-FILENAME :   REG
GIVE COMMENT FOR SYSTEMDATA :

/DPACS/ TIME-16:18:55 CPU-00:00:27
```

3.7　ＤＰＥＸサブシステムの理論概要・処理機能・利用例

Ⅰ．　システムデータのハンドリング

Ⅱ．　システムの解析

システムおよびシステムデータの一覧出力	ＤＸ．Ⅰ．１
ＬＩＳＴ｛␣ＳＮ１｛␣ＳＮ２｛・・・｝｝｝	

機　能

現在ＤＰＡＣＳに登録されているシステムの一覧およびその詳細情報であるシステム・データを画面に出力する。

◆システム名を入力しなかった場合は，登録されているシステムの一覧を出力する。

◆システム名を入力した場合は，入力されたシステムの全てのシステム・データを出力する。

実 行 例

本コマンドは，ＤＰＡＣＳサブシステムのＬＩＳＴコマンド（ＤＰ．Ⅰ．１）と同一であるので，実行例についてはそちらを参照されたい。

システムおよびシステムデータの削除	ＤＸ．Ⅰ．２
ＤＥＬＥＴＥ␣ＳＮ１｛␣ＳＮ２｛・・・｝｝	

機　能

指定したシステムのシステムデータを削除する。

◆指定したシステムがシステムデータを１個しか持っていない場合には，システムデータを削除し，そのシステムを管理ファイル上から削除する。

◆指定したシステムが複数個のシステムデータを持っている場合には，削除するシステムデータを選択することができる。全てのシステムデータを削除した場合には，そのシステムを管理ファイル上から削除する。

実 行 例

本コマンドは，ＤＰＡＣＳサブシステムのＤＥＬＥＴＥコマンド（ＤＰ．Ⅰ．２）と同一であるので，実行例についてはそちらを参照されたい。

システムの名称変更	ＤＸ．Ⅰ．３
ＲＥＮＡＭＥ␣ＳＮ␣ＯＳＮ	

機　　能

システム名をＳＮからＯＳＮに変更する。

実行例

本コマンドは，ＤＰＡＣＳサブシステムのＲＥＮＡＭＥコマンド（ＤＰ．Ⅰ．３）と同一であるので，実行例についてはそちらを参照されたい。

システムの保護	ＤＸ．Ⅰ．４
ＰＲＯＴＥＣＴ [# / ／Ｕ] ␣ＳＮ１｛␣ＳＮ２｛・・・｝｝	

機　　能

システムのプロテクトやプロテクトの解除を行う。

◆第１スイッチ／Ｕを付加しなかった場合には，当該システムをプロテクトする。

◆第１スイッチ／Ｕを付加した場合には，当該システムのプロテクトを解除する。

実行例

本コマンドは，ＤＰＡＣＳサブシステムのＰＲＯＴＥＣＴコマンド（ＤＰ．Ⅰ．４）と同一であるので，実行例についてはそちらを参照されたい。

状態変数ベクトル初期値算出	ＤＸ．Ⅱ．１
ＩＮＩＴ␣ＳＮ１␣ＳＮ２	

機　　能

通常，同定問題を解いた後，ＣＭＲコマンドによりシステム・モデルを求めたとき，そのシステムが妥当なものかどうかを調べる必要がある。元の同定問題を解く際に用いた入力データをシステム・モデルに入力して，その出力と同定問題を解く際に用いた出力データとを比較するのが通常行うチェック方法である。ＤＰＡＣＳサブシステムのＳＭＬＴコマンドを用いて行う際，その中でシステム・モデルの内部状態量の初期値 $\boldsymbol{x}(0)$ を入力しなければな

らない。本コマンドは，その$\boldsymbol{x}(0)$を計算するものである。

◆ＳＮ１は，離散系状態空間表現形式システム（可観測でなければならない）である。

◆ＳＮ２は，入出力データ表現形式システムである。ただし，ＳＮ１はＳＮ２のシステム同定の結果得られたシステムであることを前提としている。

理論概要

離散時間状態空間表現形式システムΣ_S（ＳＮ１）を

$$\Sigma_S \begin{cases} \boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k) \\ \boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k) \end{cases}$$

とし，同定問題を解く際に用いた入出力データ表現形式システムΣ_I（ＳＮ２）は，次とする。

$$\Sigma_I \left\{ \boldsymbol{u}(k),\ \boldsymbol{y}(k),\ k=0,\ 1, \cdots\cdots,\ N \right\}$$

尚　$\boldsymbol{A} \in \mathbf{R}^{n \times n}$, $\boldsymbol{B} \in \mathbf{R}^{n \times m}$, $\boldsymbol{C} \in \mathbf{R}^{r \times n}$である。

今，初期値を求めたい時刻をt_0とする。ここにt_0は０～（$N-\frac{n}{r}$）の整数とする。

下式の連立一次方程式を解くことになる。

$$\begin{array}{l} \boldsymbol{y}(j+t_0-1) - \boldsymbol{C}\boldsymbol{B}\boldsymbol{u}(j+t_0-2) - \boldsymbol{C}\boldsymbol{A}\boldsymbol{B}\boldsymbol{u}(j+t_0-3) \cdots\cdots \boldsymbol{C}\boldsymbol{A}^{j-2}\boldsymbol{B}\boldsymbol{u}(t_0) \\ \boldsymbol{y}(j+t_0-2) - \boldsymbol{C}\boldsymbol{B}\boldsymbol{u}(j+t_0-3) \cdots\cdots \boldsymbol{C}\boldsymbol{A}^{j-3}\boldsymbol{B}\boldsymbol{u}(t_0) \\ \vdots \\ \boldsymbol{y}(t_0) \end{array} \begin{array}{c} = \\ = \\ \vdots \\ = \end{array} \begin{array}{l} \boldsymbol{C}\boldsymbol{A}^{j-1} \\ \boldsymbol{C}\boldsymbol{A}^{j-2} \\ \vdots \\ \boldsymbol{C} \end{array} \begin{array}{l} \boldsymbol{x}(t_0) \\ \boldsymbol{x}(t_0) \\ \vdots \\ \boldsymbol{x}(t_0) \end{array}$$

$$\Downarrow \qquad\qquad \Downarrow$$

$$\boldsymbol{f} = \boldsymbol{G} \cdot \boldsymbol{x}(t_0)$$

ここで，$j = \begin{cases} \frac{n}{r} & \text{（nがrの整数倍の時）} \\ [\frac{n}{r}] + 1 & \text{（nがrの整数倍でない時）} \end{cases}$

$\boldsymbol{f} \in \mathbf{R}^{r \cdot j \times 1}$, $\boldsymbol{G} \in \mathbf{R}^{r \cdot j \times n}$

従って，（ａ）nがrの整数倍なら　　$\boldsymbol{x}(t_0) = \boldsymbol{G}^{-1} \cdot \boldsymbol{f}$

（ｂ）nがrの整数倍でないなら　　$\boldsymbol{x}(t_0) = ({}^t\boldsymbol{G} \cdot \boldsymbol{G})^{-1t}\boldsymbol{G} \cdot \boldsymbol{f}$

実 行 例

〔1〕 入出力時系列データから内部状態変数の初期値を計算する例

```
/DPEX/ TIME-22:39:52 CPU-00:00:08
INIT SIM1302 SIM1302
  INIT START
  1) SIM1302.IO  (1302) ;SYSIN    /I/ / /   (87/ 9/26) :
  2) SIM1302.DO  (   4) ;GLS      / / / /   (87/ 9/26) :
  3) SIM1302.SO  (   4) ;CMR      /D/1/ /   (87/ 9/26) :
  4) SIM1302.D1  (   8) ;GLS      / / / /   (87/ 9/28) :
  5) SIM1302.S1  (   8) ;CMR      /D/1/ /   (87/ 9/29) :
 WHICH SYSTEM DATA USED ? (1- 5) :  3
  1) SIM1302.IO  (1302) ;SYSIN    /I/ / /   (87/ 9/26) :
  2) SIM1302.DO  (   4) ;GLS      / / / /   (87/ 9/26) :
  3) SIM1302.SO  (   4) ;CMR      /D/1/ /   (87/ 9/26) :
  4) SIM1302.D1  (   8) ;GLS      / / / /   (87/ 9/28) :
  5) SIM1302.S1  (   8) ;CMR      /D/1/ /   (87/ 9/29) :
 WHICH SYSTEM DATA USED ? (1- 5) :  1
 STARTING POINT ? (0-1302) :  0
 OUTPUT-MODE OF ( 4, 1) MATRIX  X(TO)    : _
 --- ( 4, 1) MATRIX  X(TO)   ---
 ( 1)   7.93212587D+01
 ( 2)   1.04902267D+02
 ( 3)   2.53799673D+02
 ( 4)  -4.13098226D+02
 FILENAME :   G
/DPEX/ TIME-22:45:37 CPU-00:00:17
```

トラッキング問題（任意目標入力）の求解	DX. II. 2
TRACK ([／S ／T]) [# ／R] ␣SN	

機 能

状態空間表現システムΣ_S（SN）の出力を，目標入力$z(t)$に追従させる最適制御則を求め，ノイズを含めた系に対してシミュレーションする。

第1スイッチは，最適制御則を終端時間T→∞とした定常Riccati方程式を用いて解く（／S）か，終端時間をTとした非定常Riccati方程式を用いて解く（／T）かを指定する。

◆第1スイッチを省略した場合には，定常Riccati方程式（／S)を用いて解く方法を選択したものと見做される。

◆第2スイッチは，リスタート機能を使用するか否かのスイッチで，／Rが指定された場合には，直前に本機能により求めた最適制御則を用い，シミュレーションすることを示す。ただし，その場合ノイズは変更可能である。

理論概要 [13)]

システムを可観測・可制御の連続系の次式で表現できるものとする。

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) \end{cases}$$

今，出力 $\boldsymbol{y}(t)$ が出力目標値 $\boldsymbol{z}(t)$ に追従（トラッキング）する最適制御則を求める。
まず評価関数 $J(\boldsymbol{u})$ を次式とする。

$$J(\boldsymbol{u}) = \frac{1}{2}\ {}^t\boldsymbol{eFe}\ |_{t=T} + \frac{1}{2}\int_t^T ({}^t\boldsymbol{eQe} + {}^t\boldsymbol{uRu})\ dt$$

ここに，$\boldsymbol{e}(t) = \boldsymbol{z}(t) - \boldsymbol{y}(t)$，Tは終端時刻，$\boldsymbol{Q}$ は非負定行列，$\boldsymbol{R}$ は正定行列，$\boldsymbol{F}$ は非負定行列である。これより，Hamiltonian（評価関数の中の被積分関数を $\dot{\boldsymbol{x}}$ に対してLegendre変換する）を作ると，

$$H = \frac{1}{2}\ {}^t\boldsymbol{eQe} + \frac{1}{2}\ {}^t\boldsymbol{uRu} + {}^t\dot{\boldsymbol{x}}(t)\cdot\boldsymbol{p}$$

これらから，最適制御の条件 $\frac{\partial H}{\partial \boldsymbol{u}} = \boldsymbol{0}$，さらにcostateの満たす正準方程式 $\dot{\boldsymbol{p}} = -\frac{\partial H}{\partial \boldsymbol{x}}$，系の方程式 $\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u}$ の3式を使って整理すると

$$\begin{pmatrix} \dot{\boldsymbol{x}} \\ \dot{\boldsymbol{p}} \end{pmatrix} = \left[\begin{array}{c|c} \boldsymbol{A} - \boldsymbol{B}({}^t\boldsymbol{DQD}+\boldsymbol{R})^{-1\,t}\boldsymbol{DQC} & -\boldsymbol{B}({}^t\boldsymbol{DQD}+\boldsymbol{R})^{-1\,t}\boldsymbol{B} \\ \hline -{}^t\boldsymbol{CQC} + {}^t\boldsymbol{CQD}({}^t\boldsymbol{DQD}+\boldsymbol{R})^{-1\,t}\boldsymbol{DQC} & -{}^t\boldsymbol{A} + {}^t\boldsymbol{CQD}({}^t\boldsymbol{DQD}+\boldsymbol{R})^{-1\,t}\boldsymbol{B} \end{array}\right]\begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{p} \end{pmatrix}$$
$$+ \begin{pmatrix} \boldsymbol{B}({}^t\boldsymbol{DQD}+\boldsymbol{R})^{-1\,t}\boldsymbol{DQz} \\ \{{}^t\boldsymbol{CQ} - {}^t\boldsymbol{CQD}({}^t\boldsymbol{DQD}+\boldsymbol{R})^{-1\,t}\boldsymbol{DQ}\}\boldsymbol{z} \end{pmatrix} \qquad (*)$$

ここで，次の重要な仮定を置く。（これは，レギュレータ問題と同様である。レギュレータの場合は $\boldsymbol{p} = \boldsymbol{Kx}$）

$$\boldsymbol{p}(t) = \boldsymbol{K}(t)\cdot\boldsymbol{x}(t) - \boldsymbol{g}(t)$$

一方，$\frac{\partial H}{\partial \boldsymbol{u}} = 0$ より得られる $\boldsymbol{u}$ の式は，次式である。

$$\boldsymbol{u} = ({}^t\boldsymbol{DQD}+\boldsymbol{R})^{-1}(-{}^t\boldsymbol{Bp} + {}^t\boldsymbol{DQz} - {}^t\boldsymbol{DQCx})$$

従って，$\boldsymbol{K}(t)$ と $\boldsymbol{g}(t)$ を求めれば良いことになる。

$\boldsymbol{x}$，$\boldsymbol{p}$，$\dot{\boldsymbol{x}}$，$\dot{\boldsymbol{p}}$ に関する最適制御則からの2式（*式）と，$\boldsymbol{x}$ と $\boldsymbol{p}$ との仮定した関係式 $\boldsymbol{p} = \boldsymbol{Kx} - \boldsymbol{g}$，$\dot{\boldsymbol{p}} = \dot{\boldsymbol{K}}\boldsymbol{x} + \boldsymbol{K}\dot{\boldsymbol{x}} - \dot{\boldsymbol{g}}$ の2式との計4式から $\boldsymbol{x}$（又は $\boldsymbol{p}$ のどちらでも良い）に関する互いに全く独立な1階常微分方程式が2つできる。これらが同一であるためには，それぞれの係数行列が等しいと置いて，$\boldsymbol{K}$ と $\boldsymbol{g}$ に関する次の2微分方程式を得る。

$$\dot{\boldsymbol{K}} = -\boldsymbol{K}\boldsymbol{W}_{11} - \boldsymbol{W}_{22}\boldsymbol{K} + \boldsymbol{K}\boldsymbol{W}_{12}\boldsymbol{K} - \boldsymbol{W}_{21} \quad \text{（非定常Riccati 方程式）}$$
$$\dot{\boldsymbol{g}} = -(\boldsymbol{W}_{22} - \boldsymbol{K}\boldsymbol{W}_{12})\boldsymbol{g} + \boldsymbol{K}\boldsymbol{V}_1 - \boldsymbol{V}_2 \quad \text{（時変行列係数1階常微分方程式）}$$
$$\boldsymbol{W}_{11} = \boldsymbol{A} - \boldsymbol{B}({}^t\boldsymbol{DQD}+\boldsymbol{R})^{-1\,t}\boldsymbol{DQC}$$
$$\boldsymbol{W}_{12} = \boldsymbol{B}({}^t\boldsymbol{DQD}+\boldsymbol{R})^{-1\,t}\boldsymbol{B}$$

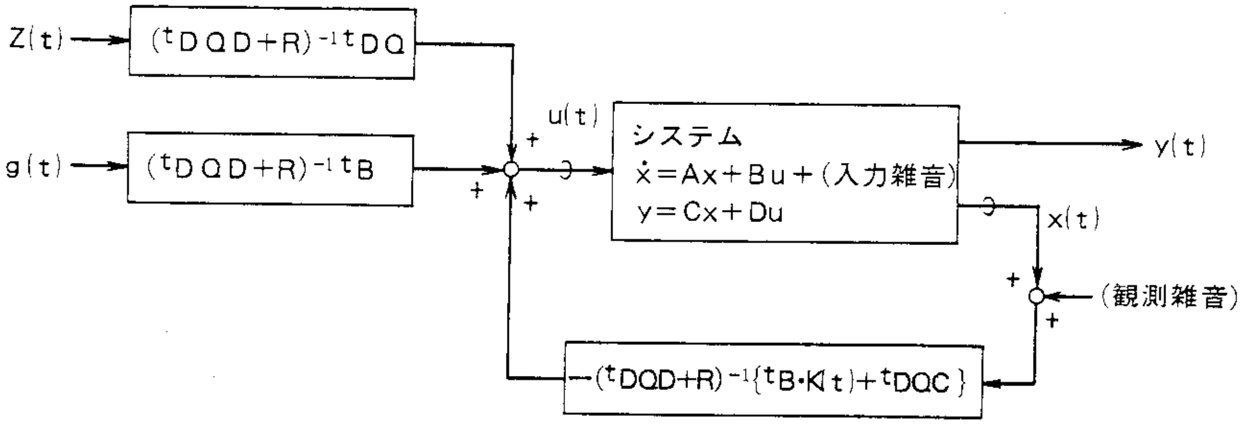
$$
\begin{aligned}
\boldsymbol{W}_{21} &= {}^t\boldsymbol{CQC} - {}^t\boldsymbol{CQD}({}^t\boldsymbol{DQD}+\boldsymbol{R})^{-1}\,{}^t\boldsymbol{DQC} \\
\boldsymbol{W}_{22} &= {}^t\boldsymbol{A} - {}^t\boldsymbol{CQD}({}^t\boldsymbol{DQD}+\boldsymbol{R})^{-1}\,{}^t\boldsymbol{B} \\
\boldsymbol{V}_{1} &= \boldsymbol{B}({}^t\boldsymbol{DQD}+\boldsymbol{R})^{-1}\,{}^t\boldsymbol{DQz} \\
\boldsymbol{V}_{2} &= \{{}^t\boldsymbol{CQ} - {}^t\boldsymbol{CQD}({}^t\boldsymbol{DQD}+\boldsymbol{R})^{-1}\,{}^t\boldsymbol{DQ}\}\boldsymbol{z} \\
\boldsymbol{u} &= ({}^t\boldsymbol{DQD}+\boldsymbol{R})^{-1}(-{}^t\boldsymbol{Bp} + {}^t\boldsymbol{DQz} - {}^t\boldsymbol{DQCx})
\end{aligned}
$$

これらを終端条件$\boldsymbol{K}(\mathrm{T}) = {}^t\boldsymbol{CFC}$，$\boldsymbol{g}(\mathrm{T}) = {}^t\boldsymbol{CFz}(\mathrm{T})$により逆時間で求めていく方法をとる。まず，非定常Riccati方程式の解法は，一般解[33]を固有値問題に帰着させて解く方法を用いた。又，T→∞とした時の定常問題については，Potterの方法[23]によって定常Riccati方程式を解く。

時変行列係数1階常微分方程式は，求めた時系列行列$\boldsymbol{K}(t)$を用いて逆時間で計算する。

さて，以上の手続きにより$\boldsymbol{g}$，$\boldsymbol{K}$が求まり，最適制御則$\boldsymbol{u}$が求まることになる。しかし，ここで用いている制御則は，連続時間系でのそれであり，離散時間系では，制御則（場合によっては評価関数も）が異なってくる筈であるということである。サンプル周期が系の特性時間に比べて短かければ，連続時間系の制御則をそのまま離散時間系に適用しても最適性は近似の範囲で成立するであろうと思われる。（実際に本機能で検証してみると確かに成立することが分る。）

実際の系への適用を考えるとき，先のサンプルによる離散化と共に雑音が系に入ってくることになる。これらの影響，言い換えれば，最適制御則のロバスト性を検査するために，シミュレーションをすることが大事である。終端条件を入れたトラッキング制御則では，$\boldsymbol{K}(t)$という時系列行列が必要となり，この大容量の時系列データを用いたシミュレーションを引き続き行うことになる。このシミュレーションのブロック図を下図に示す。



トラッキング制御シミュレーションのブロック図

このシミュレーションの結果$\boldsymbol{y}(t)$と$\boldsymbol{z}(t)$を比較すれば，直ちにそのロバスト性が分る。ここでのノイズデータは予め時系列データとして登録しておく必要がある。

実 行 例

〔1〕 トラッキング問題の求解（定常 Riccati 方程式）の例

```
/OPEX/ TIME-16:50:19 CPU-00:00:07                                                   ①
TRACK/S TEST3
 SPECIFY DIGITALIZED PITCH & NUMBER OF POINTS. DT,NZ :  0.001,2100
 INPUT -MODE OF ( 3, 3) MATRIX    Q      : I
 TYPE OR REVISE ? _                                   ②
 INPUT -MODE OF ( 3, 3) MATRIX    R      : D
 (ONLY DIAGONAL PART)
 ( 1) ROW
0.0001
 ( 2) ROW
0.0001
 ( 3) ROW
0.0001
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3,2100) TIME SERIES   ZINP   :  F  ③
 INPUT -FILENAME :   TRACKR
 OUTPUT RICCATI EQU. SOLUTION ? _  ④
 OUTPUT-MODE OF ( 3, 3) MATRIX  K(2100) : _
 --- ( 3, 3) MATRIX  K(2100) ---
 ( 1)  1.00999903D-02  1.00959438D-04  5.14685356D-07
 ( 2)  1.00959437D-04  1.03075262D-02 -1.01966335D-04
 ( 3)  5.14685356D-07 -1.01966335D-04  1.02014800D-02
 FILENAME : _
 TIME-16:53:31 CPU-00:00:14 SIMULATION. ? (Y OR N) : _      ⑤
 SPECIFY NOISE-APPLIED PART.(IN/OBS/BOTH/NO) :  BOTH        ⑥
 INPUT -MODE OF ( 3, 3) MATRIX  I-NOISE : D
 (ONLY DIAGONAL PART)
 ( 1) ROW
0.1
 ( 2) ROW                                            ⑦
0.1
 ( 3) ROW
0.1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3, 3) MATRIX  O-NOISE : D
 (ONLY DIAGONAL PART)
 ( 1) ROW
0.1
 ( 2) ROW                                            ⑦
0.1
 ( 3) ROW
0.1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3, 1) MATRIX   X(0)    : Z  ⑧
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3,2100) TIME SERIES  NOISE    :  F      ⑨
 INPUT -FILENAME :   GAUSS
 SPECIFY SAMPL PITCH FOR CONTROL ( INTEGER, MULT BY DT ) :  1   ⑩
 OUTPUT-MODE OF ( 3,2100) TIME SERIES   X(T)    :  F
 OUTPUT-FILENAME :   X
 OUTPUT-MODE OF ( 3,2100) TIME SERIES   U(T)    :  F
 OUTPUT-FILENAME :   U
 OUTPUT-MODE OF ( 3,2100) TIME SERIES   Y(T)    :  F
 OUTPUT-FILENAME :   Y
 OUTPUT-MODE OF ( 3,2100) TIME SERIES   G(T)    :  F
 OUTPUT-FILENAME :   G
```

```
TIME-16:57:31 CPU-00:00:26 SIMULATION. ? (Y OR N) : _        ⑪
SPECIFY NOISE-APPLIED PART.(IN/OBS/BOTH/NO)  :  BOTH
INPUT -MODE OF ( 3, 3) MATRIX  I-NOISE : D
(ONLY DIAGONAL PART)
( 1) ROW
0.1
( 2) ROW
0.1
( 3) ROW
0.1
TYPE OR REVISE ? _
INPUT -MODE OF ( 3, 3) MATRIX  O-NOISE : D

(ONLY DIAGONAL PART)
( 1) ROW
0.1
( 2) ROW
0.1
( 3) ROW
0.1
TYPE OR REVISE ? _
INPUT -MODE OF ( 3, 1) MATRIX   X(O)    : Z
TYPE OR REVISE ? _
INPUT -MODE OF ( 3,2100) TIME SERIES  NOISE    :  F
INPUT -FILENAME :   GAUSS
SPECIFY SAMPL PITCH FOR CONTROL ( INTEGER, MULT BY DT ) :  5
OUTPUT-MODE OF ( 3, 420) TIME SERIES   X(T)    :  F
OUTPUT-FILENAME :   XT
OUTPUT-MODE OF ( 3, 420) TIME SERIES   U(T)    :  F
OUTPUT-FILENAME :   UT
OUTPUT-MODE OF ( 3, 420) TIME SERIES   Y(T)    :  F
OUTPUT-FILENAME :   YT
OUTPUT-MODE OF ( 3, 420) TIME SERIES   G(T)    :  F
OUTPUT-FILENAME :   GT
TIME-17:01:38 CPU-00:00:37 SIMULATION. ? (Y OR N) : _
SPECIFY NOISE-APPLIED PART.(IN/OBS/BOTH/NO)  :  BOTH
INPUT -MODE OF ( 3, 3) MATRIX  I-NOISE : D
(ONLY DIAGONAL PART)
( 1) ROW
0.1
( 2) ROW
0.1
( 3) ROW
0.1
TYPE OR REVISE ? _
INPUT -MODE OF ( 3, 3) MATRIX  O-NOISE : D
(ONLY DIAGONAL PART)
( 1) ROW
0.1
( 2) ROW
0.1
( 3) ROW
0.1
TYPE OR REVISE ? _
INPUT -MODE OF ( 3, 1) MATRIX   X(O)    : Z
TYPE OR REVISE ? _
INPUT -MODE OF ( 3,2100) TIME SERIES  NOISE    :  F
```

```
INPUT -FILENAME :   GAUSS
SPECIFY SAMPL PITCH FOR CONTROL ( INTEGER, MULT BY DT ) :  10
OUTPUT-MODE OF ( 3, 210) TIME SERIES   X(T)    :  F
OUTPUT-FILENAME :   XTD
OUTPUT-MODE OF ( 3, 210) TIME SERIES   U(T)    :  F
OUTPUT-FILENAME :   UTD
OUTPUT-MODE OF ( 3, 210) TIME SERIES   Y(T)    :  F
OUTPUT-FILENAME :   YTD
OUTPUT-MODE OF ( 3, 210) TIME SERIES   G(T)    :  F
OUTPUT-FILENAME :   GTD
TIME-17:05:50 CPU-00:00:49 SIMULATION. ? (Y OR N) :  N
/DPEX/ TIME-17:05:52 CPU-00:00:49
```

【説明】

① トラッキングさせる連続系システムの離散化ピッチタイム及び目標時系列データのデータ点数を入力する。

② 重み係数行列$\boldsymbol{Q}$（準正定マトリックス），$\boldsymbol{R}$（正定マトリックス）の入力

③ 目標時系列データの入力

④ 定常リカッチ方程式の解の出力指示（Ｙまたは☑：出力，Ｎ：非出力）

⑤ シミュレーションの実行指示

⑥ ノイズの入力部分の指示
　（ＩＮ：入力部，ＯＢＳ：観測部，ＢＯＴＨ：両方，ＮＯ：無し）

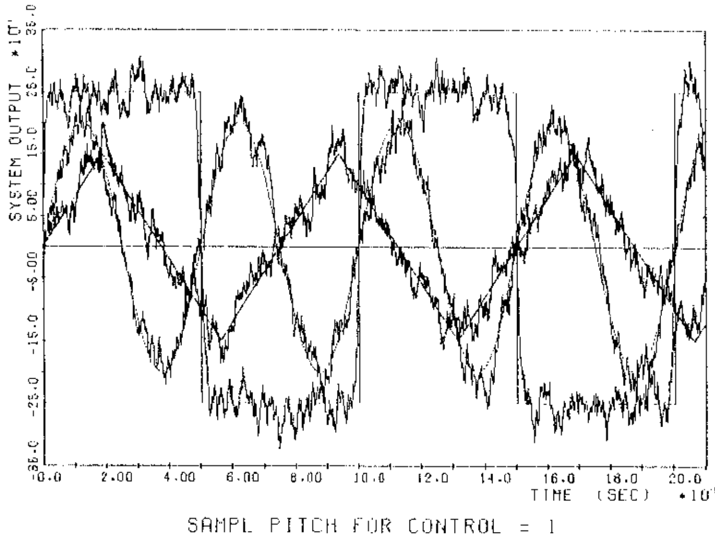
⑦ ノイズの重み係数行列の入力

⑧ シミュレーション時の状態変数ベクトルの初期値入力

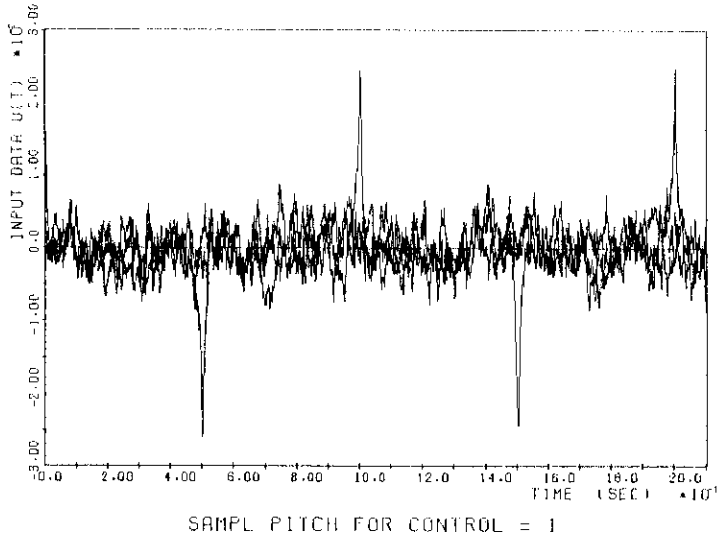
⑨ ノイズの時系列データ入力

⑩ シミュレーション時の離散化サンプルピッチの入力
　離散化ピッチタイムの倍数を入力する。

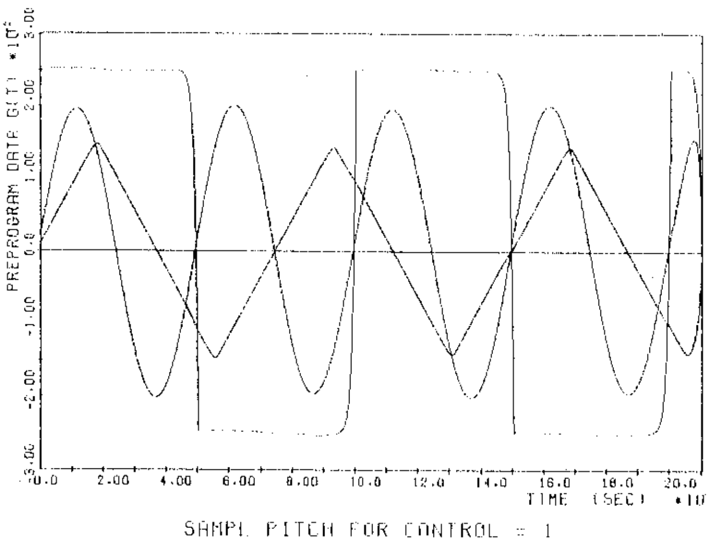
⑪ シミュレーションの実行指示（Ｙまたは☑で⑤以降の処理に続く。）



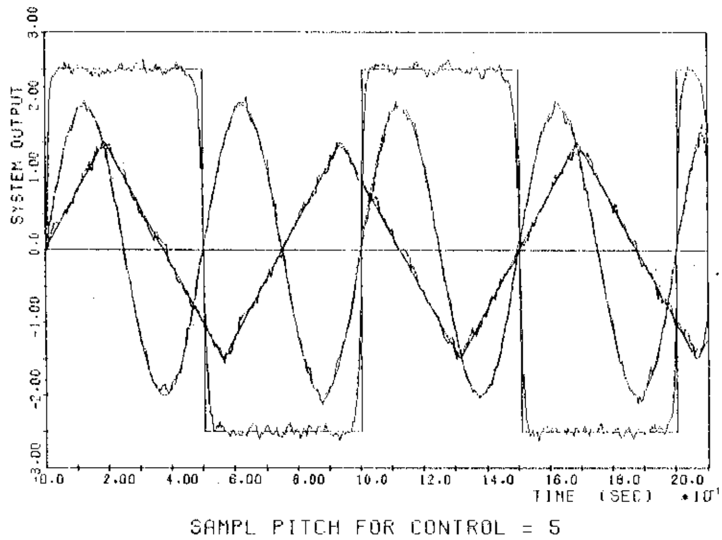
定常 Riccati 方程式の解を用いたトラッキング問題［目標値とシステムの出力：$\boldsymbol{y}(t)$］



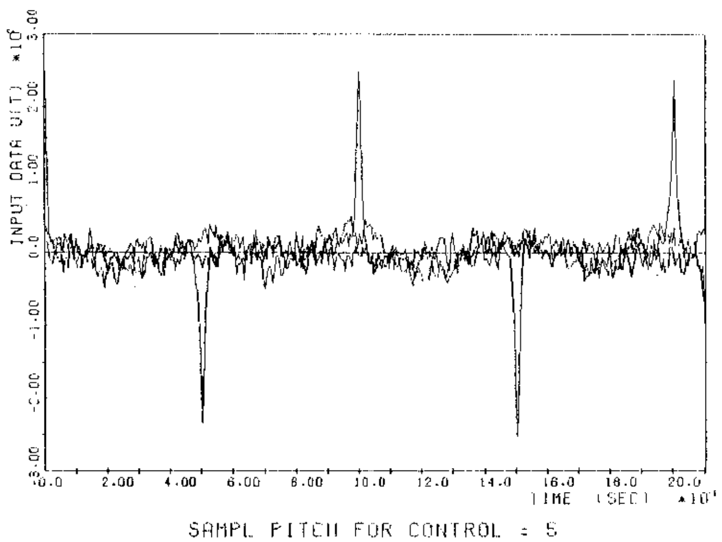
定常 Riccati 方程式の解を用いたトラッキング問題［システムの入力：$\boldsymbol{u}(t)$］



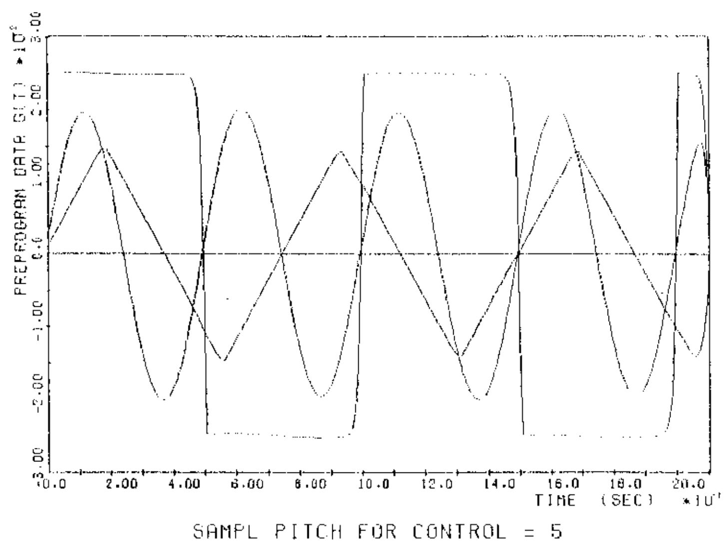
定常 Riccati 方程式の解を用いたトラッキング問題［プレプロデータ：$\boldsymbol{g}(t)$］



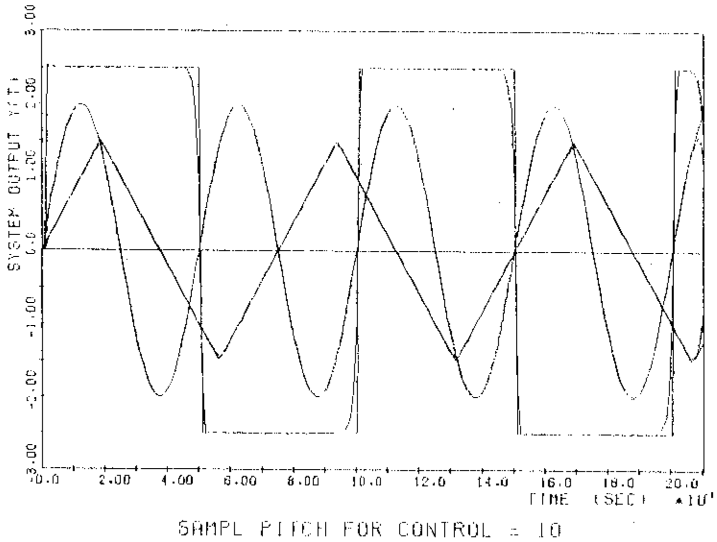
定常 Riccati 方程式の解を用いたトラッキング問題［目標値とシステムの出力：$\boldsymbol{y}(t)$］



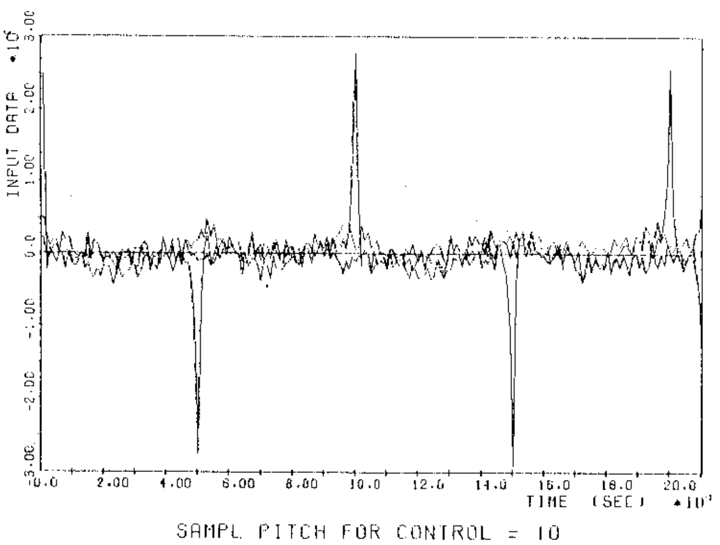
定常 Riccati 方程式の解を用いたトラッキング問題［システムの入力：$\boldsymbol{u}(t)$］



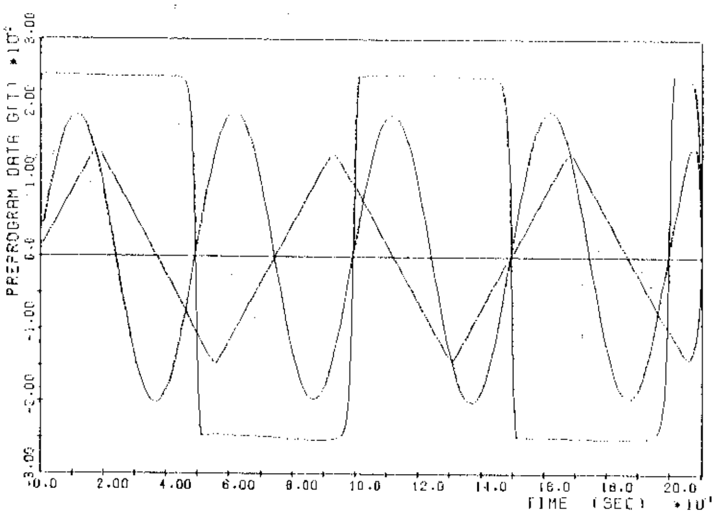
定常 Riccati 方程式の解を用いたトラッキング問題［プレプロデータ：$\boldsymbol{g}(t)$］



定常Riccati方程式の解を用いたトラッキング問題［目標値とシステムの出力：$\boldsymbol{y}(t)$］



定常Riccati方程式の解を用いたトラッキング問題［システムの入力：$\boldsymbol{u}(t)$］



定常Riccati方程式の解を用いたトラッキング問題［プレプロデータ：$\boldsymbol{g}(t)$］

〔2〕 トラッキング問題の求解（非定常Riccati方程式）の例

```
 /DPEX/ TIME-15:50:03 CPU-00:00:10
TRACK/T TEST3                                                                    ①
 SPECIFY DIGITALIZED PITCH & NUMBER OF POINTS. DT,NZ :  0.001,2100
 INPUT -MODE OF ( 3, 3) MATRIX    F      : D      ②
 (ONLY DIAGONAL PART)
 ( 1) ROW
0.01
 ( 2) ROW
0.01
 ( 3) ROW
0.01
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3, 3) MATRIX    Q      : I      ②
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3, 3) MATRIX    R      : D      ②
 (ONLY DIAGONAL PART)
 ( 1) ROW
0.0001
 ( 2) ROW
0.0001
 ( 3) ROW
0.0001
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3,2100) TIME SERIES   ZINP   :  F   ③
 INPUT -FILENAME :   TRACKR
 HOW MANY POINTS OF THE RICCATI EQU. SOLUTIONS ARE OUTPUT ?  3   ④
 SPECIFY    3 OUTPUT POINTS :  100 2000 2100
 OUTPUT-MODE OF ( 3, 3) MATRIX  K(2100) : _
 --- ( 3, 3) MATRIX  K(2100) ---
 ( 1)  1.00000000D-02  0.0             0.0
 ( 2)  0.0             1.00000000D-02  0.0
 ( 3)  0.0             0.0             1.00000000D-02
 FILENAME : _
 OUTPUT-MODE OF ( 3, 3) MATRIX  K(2000) : _
 --- ( 3, 3) MATRIX  K(2000) ---
 ( 1)  1.00999837D-02  1.00952775D-04  5.13992584D-07
 ( 2)  1.00952775D-04  1.03075022D-02 -1.01959231D-04
 ( 3)  5.13992584D-07 -1.01959231D-04  1.02014659D-02
 FILENAME : _
 OUTPUT-MODE OF ( 3, 3) MATRIX  K( 100) : _
 --- ( 3, 3) MATRIX  K( 100) ---
 ( 1)  1.00999903D-02  1.00959437D-04  5.14685352D-07
 ( 2)  1.00959437D-04  1.03075262D-02 -1.01966335D-04
 ( 3)  5.14685352D-07 -1.01966335D-04  1.02014800D-02
 FILENAME : _
 TIME-15:52:49 CPU-00:00:22 SIMULATION. ? (Y OR N) :  Y    ⑤
 SPECIFY NOISE-APPLIED PART.(IN/OBS/BOTH/NO) :  BOTH   ⑥
 INPUT -MODE OF ( 3, 3) MATRIX  I-NOISE : D   ⑦
 (ONLY DIAGONAL PART)
 ( 1) ROW
0.1
 ( 2) ROW
0.1
 ( 3) ROW
0.1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3, 3) MATRIX  O-NOISE : D   ⑦
```

```
 (ONLY DIAGONAL PART)
 ( 1) ROW
0.1
 ( 2) ROW
0.1
 ( 3) ROW
0.1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3, 1) MATRIX   X(0)    : Z  ⑧
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3,2100) TIME SERIES  NOISE   :  F    ⑨
 INPUT -FILENAME :   GAUSS
 SPECIFY SAMPL PITCH FOR CONTROL ( INTEGER, MULT BY DT ) :  5  ⑩
 OUTPUT-MODE OF ( 3, 420) TIME SERIES   X(T)   :  F
 OUTPUT-FILENAME :   X
 OUTPUT-MODE OF ( 3, 420) TIME SERIES   U(T)   :  F
 OUTPUT-FILENAME :   U
 OUTPUT-MODE OF ( 3, 420) TIME SERIES   Y(T)   :  F
 OUTPUT-FILENAME :   Y
 OUTPUT-MODE OF ( 3, 420) TIME SERIES   G(T)   :  F
 OUTPUT-FILENAME :   G
 TIME-15:56:33 CPU-00:00:36 SIMULATION. ? (Y OR N) :  N  ⑪
 /DPEX/ TIME-15:56:39 CPU-00:00:36
```

【説明】

① トラッキングさせる連続系システムの離散化ピッチタイム及び目標時系列データのデータ点数を入力する。

② 重み係数行列$\boldsymbol{F}$，$\boldsymbol{Q}$（準正定マトリックス），$\boldsymbol{R}$（正定マトリックス）の入力

③ 目標時系列データの入力

④ 非定常リカッチ方程式の解の個数の出力指示（0≦個数≦10）
0を入力した場合は出力しない。また出力は，時刻の後の方が先に出力される。

⑤ シミュレーションの実行指示

⑥ ノイズの入力部分の指示
（ＩＮ：入力部，ＯＢＳ：観測部，ＢＯＴＨ：両方，ＮＯ：無し）

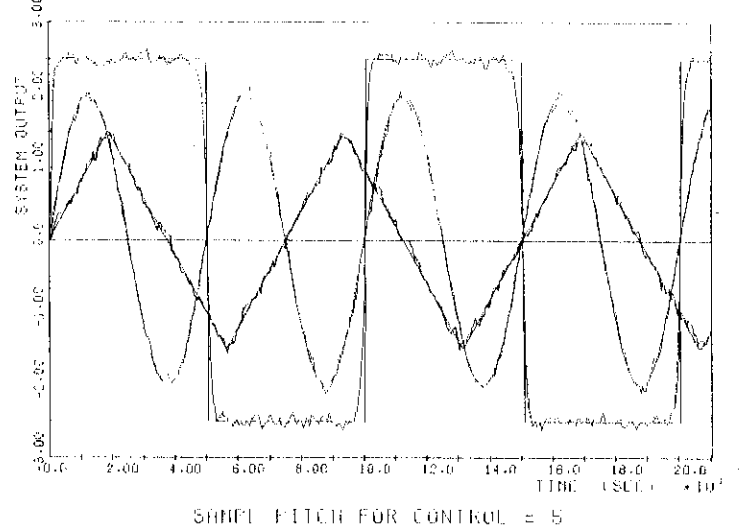
⑦ ノイズの重み係数行列の入力

⑧ シミュレーション時の状態変数ベクトルの初期値入力

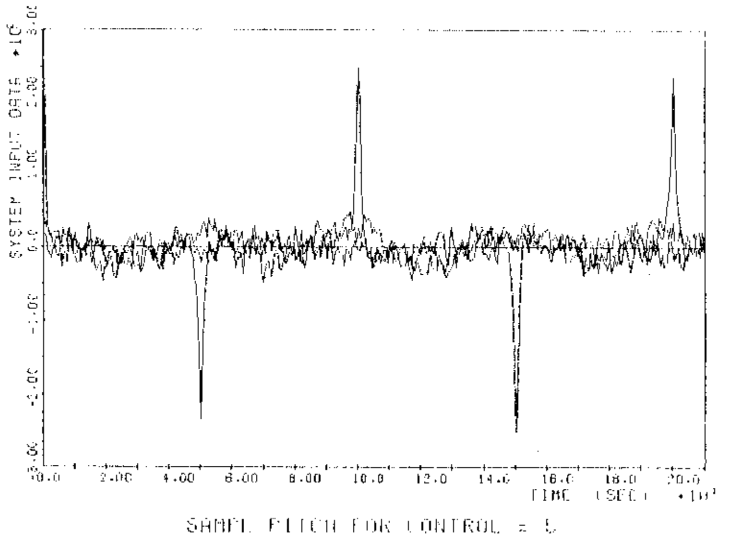
⑨ ノイズの時系列データ入力

⑩ シミュレーション時の離散化サンプルピッチの入力
離散化ピッチタイムの倍数を入力する。

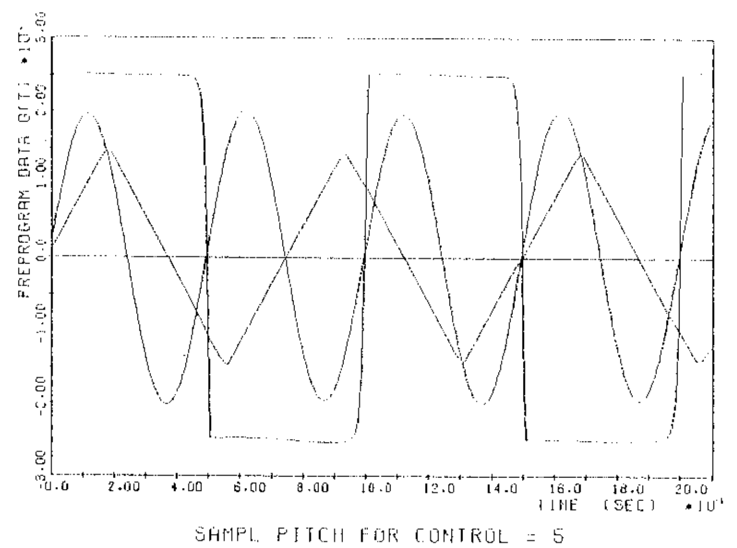
⑪ シミュレーションの実行指示（Yまたは↵で⑤以降の処理に続く。）



非定常Riccati方程式の解を用いたトラッキング問題［目標値とシステムの出力：$\boldsymbol{y}(t)$］



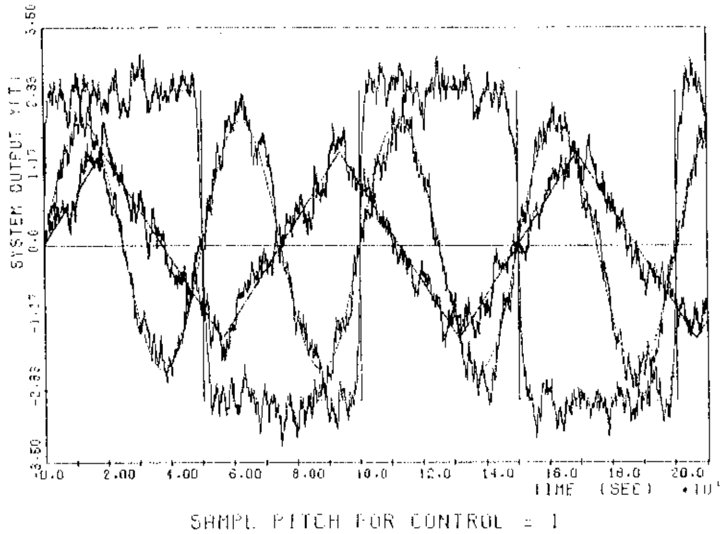
非定常Riccati方程式の解を用いたトラッキング問題［システムの入力：$\boldsymbol{u}(t)$］



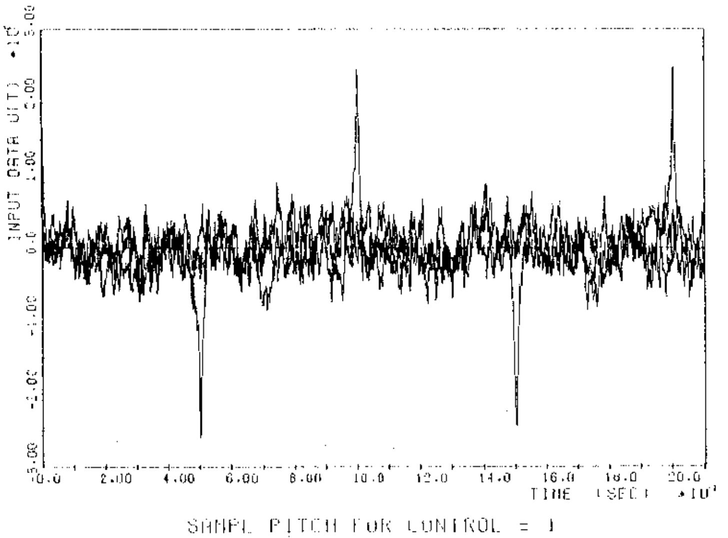
非定常Riccati方程式の解を用いたトラッキング問題［プレプロデータ：$\boldsymbol{g}(t)$］

〔3〕 トラッキング問題の求解（非定常Riccati方程式）のリスタートの例

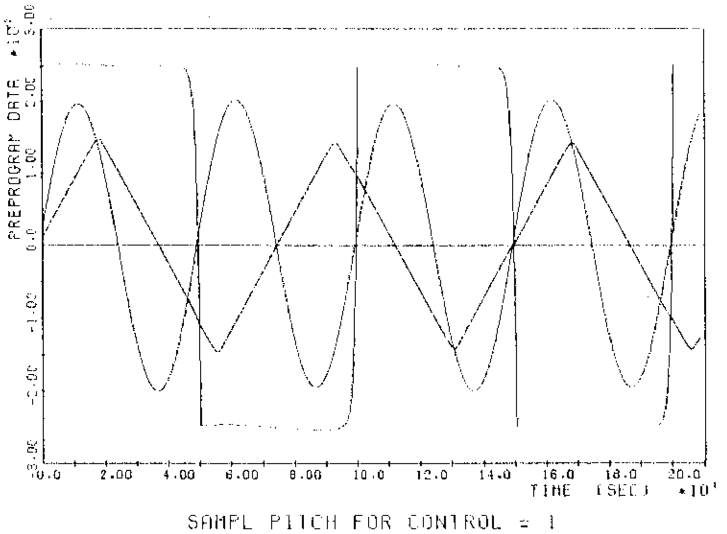
```
 /DPEX/ TIME-16:13:46 CPU-00:00:01
TRACK/T/R TEST3
 TIME-16:14:16 CPU-00:00:04 SIMULATION. ? (Y OR N) :  Y
 SPECIFY NOISE-APPLIED PART.(IN/OBS/BOTH/NO) :  BOTH
 INPUT -MODE OF ( 3, 3) MATRIX  I-NOISE : D
 (ONLY DIAGONAL PART)
 ( 1) ROW
0.1
 ( 2) ROW
0.1
 ( 3) ROW
0.1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3, 3) MATRIX  O-NOISE : D
 (ONLY DIAGONAL PART)
 ( 1) ROW
0.1
 ( 2) ROW
 ***
0.1
 ( 3) ROW
0.1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3, 1) MATRIX   X(O)   : Z
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3,2100) TIME SERIES  NOISE   :  F
 INPUT -FILENAME :   GAUSS
 SPECIFY SAMPL PITCH FOR CONTROL ( INTEGER, MULT BY DT ) :  1
 OUTPUT-MODE OF ( 3,2100) TIME SERIES   X(T)   :  F
 OUTPUT-FILENAME :   X
 OUTPUT-MODE OF ( 3,2100) TIME SERIES   U(T)   :  F
 OUTPUT-FILENAME :   U
 OUTPUT-MODE OF ( 3,2100) TIME SERIES   Y(T)   :  F
 OUTPUT-FILENAME :   Y
 OUTPUT-MODE OF ( 3,2100) TIME SERIES   G(T)   :  F
 OUTPUT-FILENAME :   G
 TIME-16:17:24 CPU-00:00:17 SIMULATION. ? (Y OR N) :  N
 /DPEX/ TIME-16:17:33 CPU-00:00:17
```



非定常Riccati方程式の解を用いたトラッキング問題［目標値とシステムの出力：$\boldsymbol{y}(t)$］



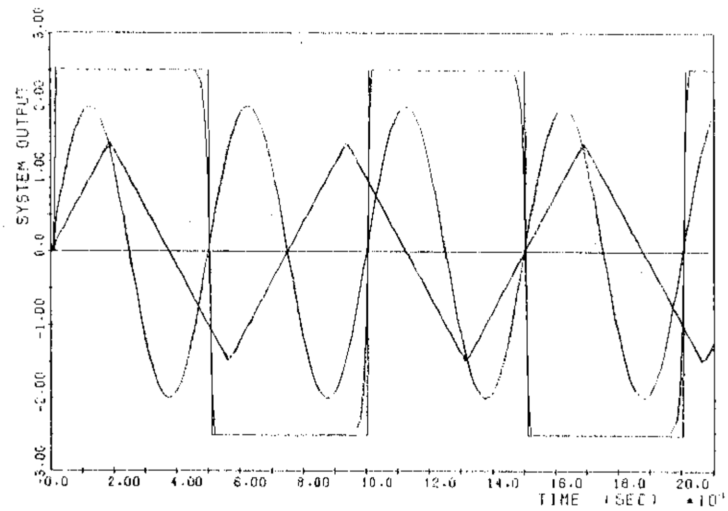
非定常Riccati方程式の解を用いたトラッキング問題［システムの入力：$\boldsymbol{u}(t)$］



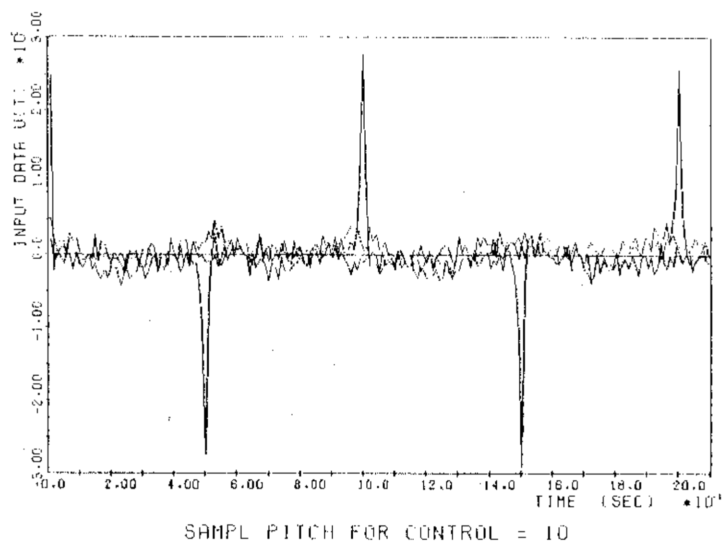
非定常Riccati方程式の解を用いたトラッキング問題［プレプロデータ：$\boldsymbol{g}(t)$］

〔4〕 トラッキング問題の求解（非定常Riccati方程式）のリスタートの例

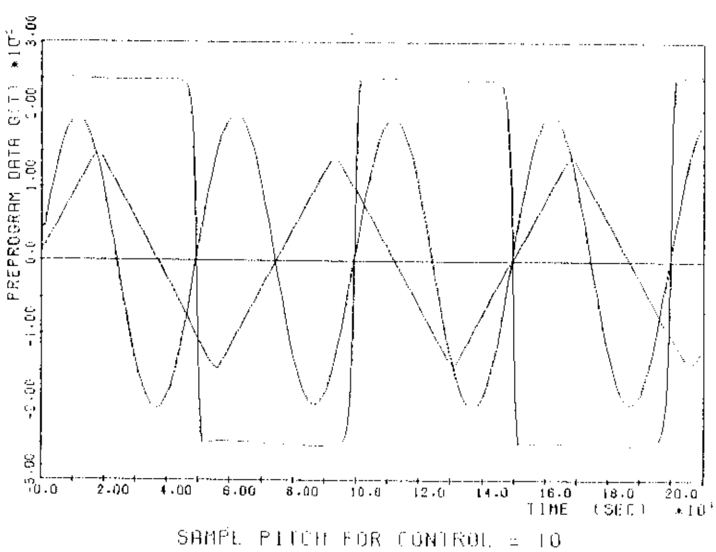
```
 /DPEX/ TIME-11:38:51 CPU-00:00:45
TRACK/T/R TEST3
 TIME-11:39:17 CPU-00:00:48 SIMULATION. ? (Y OR N) :  Y
 SPECIFY NOISE-APPLIED PART.(IN/OBS/BOTH/NO) :  BOTH
 INPUT -MODE OF ( 3, 3) MATRIX  I-NOISE : D
 (ONLY DIAGONAL PART)
 ( 1) ROW
0.1
 ( 2) ROW
0.1
 ( 3) ROW
0.1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3, 3) MATRIX  O-NOISE : D
 (ONLY DIAGONAL PART)
 ( 1) ROW
0.1
 ( 2) ROW
0.1
 ( 3) ROW
0.1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3, 1) MATRIX   X(0)    : Z
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3,2100) TIME SERIES  NOISE    :  F
 INPUT -FILENAME :   GAUSS
 SPECIFY SAMPL PITCH FOR CONTROL ( INTEGER, MULT BY DT ) :  10
 OUTPUT-MODE OF ( 3, 210) TIME SERIES   X(T)    :  F
 OUTPUT-FILENAME :   XT
 OUTPUT-MODE OF ( 3, 210) TIME SERIES   U(T)    :  F
 OUTPUT-FILENAME :   UT
 OUTPUT-MODE OF ( 3, 210) TIME SERIES   Y(T)    :  F
 OUTPUT-FILENAME :   YT
 OUTPUT-MODE OF ( 3, 210) TIME SERIES   G(T)    :  F
 OUTPUT-FILENAME :   GT
 TIME-11:42:22 CPU-00:00:59 SIMULATION. ? (Y OR N) :  N
 /DPEX/ TIME-11:42:25 CPU-00:01:00
```



非定常Riccati方程式の解を用いたトラッキング問題［目標値とシステムの出力：$\boldsymbol{y}$(t)］



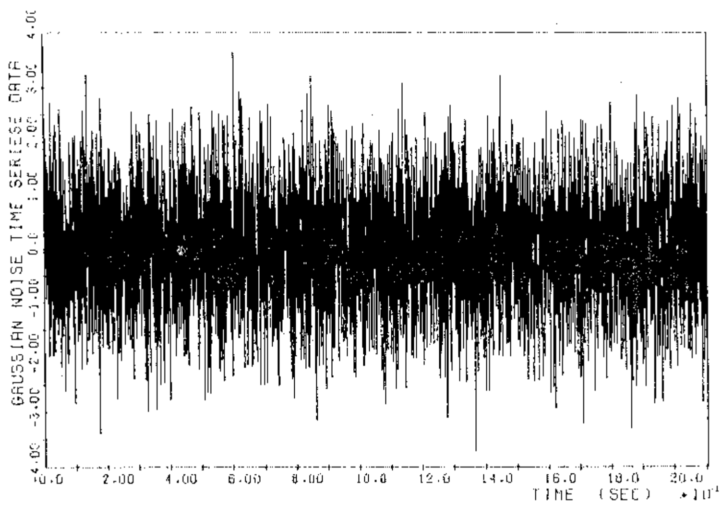
非定常Riccati方程式の解を用いたトラッキング問題［システムの入力：$\boldsymbol{u}$(t)］



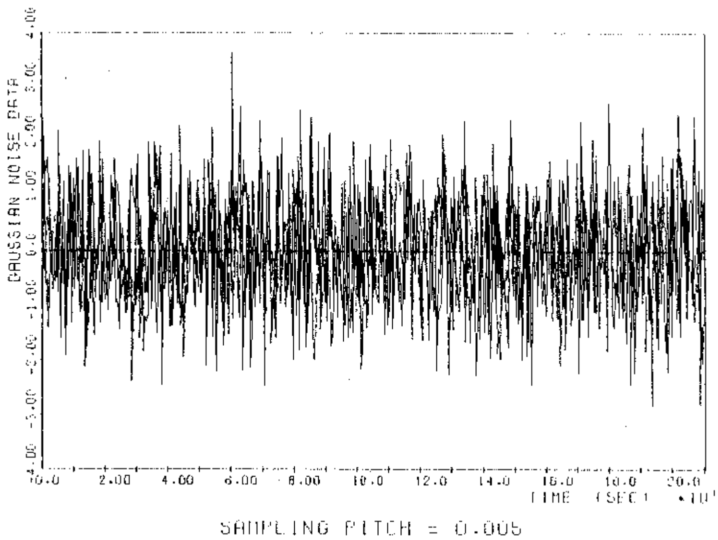
非定常Riccati方程式の解を用いたトラッキング問題［プレプロデータ：$\boldsymbol{g}$(t)］

＜トラッキング問題で使用した系＞

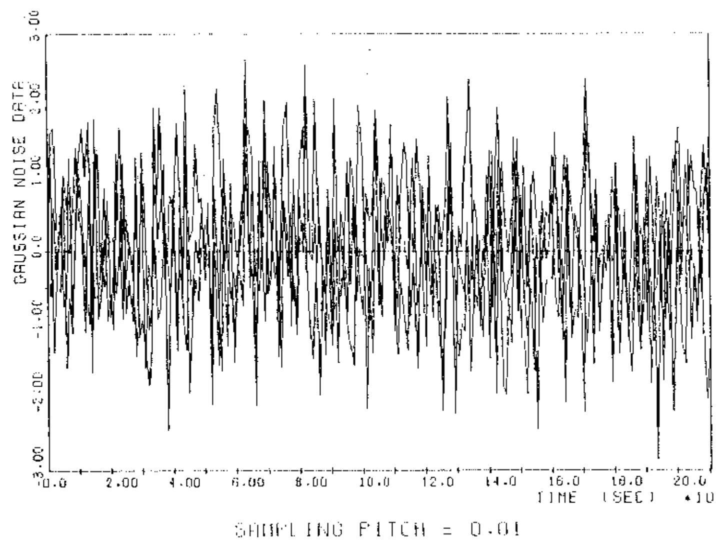
```
 /DPACS/ TIME-15:35:25 CPU-00:00:27
TYPE/S TEST3
 TO TYPE OUT SYSTEMDATA
 SYSTEMDATA TEST3.SO      ( 3, 3, 3)

 --- ( 3, 3) MATRIX   A        ---
 ( 1)  1.00000000D+00  2.00000000D+00  0.0
 ( 2)  0.0             3.00000000D+00  0.0
 ( 3)  0.0            -2.00000000D+00  2.00000000D+00
 FILENAME : _
 --- ( 3, 3) MATRIX   B        ---
 ( 1)  1.00000000D+00  0.0             0.0
 ( 2)  0.0             1.00000000D+00  0.0
 ( 3)  0.0             0.0             1.00000000D+00
 FILENAME : _
 --- ( 3, 3) MATRIX   C        ---
 ( 1)  1.00000000D+00  0.0             0.0
 ( 2)  0.0             1.00000000D+00  0.0
 ( 3)  0.0             0.0             1.00000000D+00
 FILENAME : _
 --- ( 3, 3) MATRIX   D        ---
 ( 1)  0.0             0.0             0.0
 ( 2)  0.0             0.0             0.0
 ( 3)  0.0             0.0             0.0
 FILENAME : _
 /DPACS/ TIME-15:36:37 CPU-00:00:32

 /DPACS/ TIME-15:40:01 CPU-00:00:38
POLE TEST3
 TO CALCULATE SYSTEM'S POLES
 POLES OF CONTINUOUS SYSTEM TEST3.SO

 -------------------------------------------------------------
  NO.      REAL                IMAGINARY           ABSOLUTE
 -------------------------------------------------------------
 ( 1)   1.00000000D+00      0.0                1.00000000D+00
 ( 2)   3.00000000D+00      0.0                3.00000000D+00
 ( 3)   2.00000000D+00      0.0                2.00000000D+00
 -------------------------------------------------------------
 FILENAME : _
 CONTINUOUS SYSTEM TEST3.SO      IS UNSTABLE
 /DPACS/ TIME-15:41:06 CPU-00:00:43
```



トラッキング問題で使用したノイズデータ（サンプルピッチ＝ 0.001（sec))



トラッキング問題で使用したノイズデータ（サンプルピッチ＝ 0.005（sec))



トラッキング問題で使用したノイズデータ（サンプルピッチ＝ 0.01（sec))

3.8 ＳＹＳサブシステムの理論概要・処理機能・利用例

Ⅰ. システム解析

状態空間表現の可制御性・可観測性行列の取得	SY. I. 1
V [／@] @＝S, T, M, D	

機　　能

システムデータの形式に従い可制御性・可観測性行列の計算を行う。

◆第1スイッチが／Sの場合

状態空間表現形式システムから可制御性行列または可観測性行列を得る。

◆第1スイッチが／Tの場合

伝達関数行列表現形式システムから対応する一つの状態空間表現モデルの可制御性行列を求める。

◆第1スイッチが／Mの場合

マルコフパラメータ表現形式システムから，ハンケル行列を得る。

◆第1スイッチが／Dの場合

微分（差分）方程式表現形式システムから，対応する一つの状態空間表現形式システムの可制御性行列を得る。

理論概要

◆／S：状態空間システムΣ_Sに対し下を計算。（$k \geqq n$）

可制御性行列　：　$\boldsymbol{V}_S = [\boldsymbol{B},\ \boldsymbol{A}\boldsymbol{B},\ \cdots\cdots,\ \boldsymbol{A}^{k-1}\boldsymbol{B}] \in \boldsymbol{R}^{n \times km}$

可観測性行列　：　$\boldsymbol{N}_S = {}^t[\boldsymbol{C},\ \boldsymbol{C}\boldsymbol{A},\ \cdots\cdots,\ \boldsymbol{C}\boldsymbol{A}^{k-1}] \in \boldsymbol{R}^{kp \times n}$

ここでのkは，最低nであれば，上の両行列のランク判定から，可制御，可観測を判定できる。kをnより大きくしても，1次独立でない行列を元の行列に加えていくだけであるからランク判定には影響を与えない。また，$\boldsymbol{A}$の固有値のうち1より小さい固有値の影響は，kを大きくしていくにつれ0に近づく。

◆／T：伝達関数モデルΣ_Tに対し1つの状態空間表現は

$$
\left\{
\begin{aligned}
\sigma \bar{\boldsymbol{x}} &= \begin{bmatrix} & & & -q_{n_T}\boldsymbol{I}_p \\ \boldsymbol{I}_p & \ddots & \boldsymbol{0} & \vdots \\ \boldsymbol{0} & \ddots & \boldsymbol{I}_p & -q_1\boldsymbol{I}_p \end{bmatrix} \bar{\boldsymbol{x}} + \begin{bmatrix} \boldsymbol{P}_{n_T} \\ \vdots \\ \boldsymbol{P}_1 \end{bmatrix} \boldsymbol{u} \\
\boldsymbol{y} &= [\boldsymbol{0} \cdots\cdots \boldsymbol{0}\ \boldsymbol{I}_p]\ \bar{\boldsymbol{x}}
\end{aligned}
\right.
$$

上式のk次の可制御性行列は次式で与えられる。

$$\boldsymbol{V}_T = \begin{bmatrix} \boldsymbol{V}_{11} & \cdots & \boldsymbol{V}_{1k} \\ \vdots & \ddots & \vdots \\ \boldsymbol{V}_{n_T 1} & \cdots & \boldsymbol{V}_{n_T k} \end{bmatrix}$$

ただし，$\boldsymbol{V}_{ij} \in \boldsymbol{R}^{p\times m}$

$$\begin{cases} \boldsymbol{V}_{i1} = \boldsymbol{P}_{n_T-i+1} & (\, i=1,\sim, n_T\,) \\ \boldsymbol{V}_{1j} = -q_{n_T}\boldsymbol{V}_{n_T j-1} & \\ \boldsymbol{V}_{ij} = \boldsymbol{V}_{i-1\, j-1} - q_{n_T-i+1}\boldsymbol{V}_{n_T j-1} & (\, j=2,\sim, k\,) \end{cases}$$

◆／M：マルコフパラメータ表現モデルから，ハンケル行列を得る。Σ_Mに対し，k×r 次のハンケル行列は次式で与えられる。

$$\boldsymbol{V}_M = \begin{bmatrix} \boldsymbol{V}_{11} & \cdots & \boldsymbol{V}_{1r} \\ \vdots & \ddots & \vdots \\ \boldsymbol{V}_{k1} & \cdots & \boldsymbol{V}_{kr} \end{bmatrix}$$

ただし，$\boldsymbol{V}_{ij} \in \boldsymbol{R}^{p\times m}$，$\boldsymbol{V}_{ij} = \boldsymbol{M}_{i+j-1} \quad \begin{pmatrix} i=1,\sim, k \\ j=1,\sim, r \end{pmatrix}$

◆／D：微分（差分）方程式表現モデルΣ_Dに対し，1つの状態空間表現は

$$\begin{cases} \sigma\bar{\boldsymbol{x}} = \begin{bmatrix} & \boldsymbol{0} & -\boldsymbol{T}_{n_D} \\ \boldsymbol{I}_p & \ddots & \vdots \\ \boldsymbol{0} & \boldsymbol{I}_p & -\boldsymbol{T}_1 \end{bmatrix}\bar{\boldsymbol{x}} + \begin{bmatrix} \boldsymbol{U}_{n_D} \\ \vdots \\ \boldsymbol{U}_1 \end{bmatrix}\boldsymbol{u} \\ \boldsymbol{y} = [\,\boldsymbol{0} \cdots\cdots \boldsymbol{0}\ \boldsymbol{I}_p\,]\,\bar{\boldsymbol{x}} \end{cases}$$

上式のk次の可制御性行列は，次式で与えられる。

$$\boldsymbol{V}_D = \begin{bmatrix} \boldsymbol{V}_{11} & \cdots & \boldsymbol{V}_{1k} \\ \vdots & \ddots & \vdots \\ \boldsymbol{V}_{n_D 1} & \cdots & \boldsymbol{V}_{n_D k} \end{bmatrix}$$

ただし，$\boldsymbol{V}_{ij} \in \boldsymbol{R}^{p\times m}$ は，

$$\begin{cases} \boldsymbol{V}_{i1} = \boldsymbol{U}_{n_D-i+1} \\ \boldsymbol{V}_{1j} = -\boldsymbol{T}_{n_D}\cdot\boldsymbol{V}_{n_D j-1} \\ \boldsymbol{V}_{ij} = \boldsymbol{V}_{i-1\, j-1} - \boldsymbol{T}_{n_D-i+1}\cdot\boldsymbol{V}_{n_D j-1} \end{cases}$$

実行例

〔1〕 状態空間表現形式システムから可制御性行列または可観測性行列を得る例

```
/SYS/ TIME-22:37:23 CPU-00:00:04
V/S
 TO MAKE MATRIX (1) VS OR (2) NS
 WHICH MATRIX ? :  1
 SYSTEM NAME :    EX1
 SYSTEM EX1       HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0       (    3) ;SYSIN   /S /C /  /   (87/ 7/ 3) :EX1
  2) EX1.S1       (    3) ;CMR     /T /C /2 /   (87/ 9/17) :AB2
  3) EX1.S2       (    3) ;CMR     /M /C /2 /   (87/ 9/17) :AB2
  4) EX1.S3       (    3) ;COOD    /1 /  /  /   (87/ 9/25) :
  5) EX1.S4       (    3) ;COOD    /2 /  /  /   (87/ 9/25) :COOD/2
 WHICH SYSTEMDATA IS USED ? :  1
 SPECIFY NO. OF COLUMN BLOCKS (1- 4) :   4 ①
 OUTPUT-MODE OF ( 3, 8) MATRIX  VS        :
 --- ( 3, 8) MATRIX  VS       ---
 ( 1)  0.0              0.0              1.00000000D+00 -2.00000000D+00
 ( 2)  1.00000000D+00 -2.00000000D+00 -1.00000000D+00  4.00000000D+00
 ( 3)  0.0              1.00000000D+00  0.0              1.00000000D+00
 ( 1) -1.00000000D+00  4.00000000D+00  2.00000000D+00 -4.00000000D+00
 ( 2)  2.00000000D+00 -4.00000000D+00  3.00000000D+00 -2.00000000D+00
 ( 3)  3.00000000D+00 -5.00000000D+00  0.0              7.00000000D+00
 FILENAME :  VS
/SYS/ TIME-22:42:17 CPU-00:00:12
```

【説明】

① 理論概要に述べた，kの入力要求である。尚，kを大きくすると$\boldsymbol{A}$の固有値の1より大きい要素がk乗されることにより，オーバーフローが起こる可能性があるので注意が必要である。

〔2〕 伝達関数行列表現形式システムから対応する一つの状態空間表現モデルの可制御性行列を求める例

```
/SYS/ TIME-22:37:28 CPU-00:00:04
V/T
 TO MAKE MATRIX VT
 SYSTEM NAME :   EX1
 SPECIFY NO. OF COLUMN BLOCKS (1- 7) :   7
 OUTPUT-MODE OF ( 6,14) MATRIX  VT        :
 --- ( 6,14) MATRIX  VT        ---
 ( 1) -1.00000000D+00  4.00000000D+00  0.0              0.0
 ( 2)  3.00000000D+00 -7.00000000D+00  5.00000000D+00 -5.00000000D+00
 ( 3)  1.00000000D+00 -2.00000000D+00 -1.00000000D+00  4.00000000D+00
 ( 4) -1.00000000D+00  5.00000000D+00  5.00000000D+00 -9.00000000D+00
 ( 5)  0.0             0.0             1.00000000D+00 -2.00000000D+00
 ( 6)  1.00000000D+00 -1.00000000D+00 -1.00000000D+00  5.00000000D+00
 ( 1)  5.00000000D+00 -1.00000000D+01 -5.00000000D+00  2.00000000D+01
 ( 2) -5.00000000D+00  2.50000000D+01  2.50000000D+01 -4.50000000D+01
 ( 3)  2.00000000D+00 -4.00000000D+00  3.00000000D+00 -2.00000000D+00
 ( 4)  3.00000000D+00  5.00000000D+00  5.00000000D+00  7.00000000D+00
 ( 5) -1.00000000D+00  4.00000000D+00  2.00000000D+00 -4.00000000D+00
 ( 6)  5.00000000D+00 -9.00000000D+00  3.00000000D+00  5.00000000D+00
 ( 1)  1.00000000D+01 -2.00000000D+01  1.50000000D+01 -1.00000000D+01
 ( 2)  1.50000000D+01  2.50000000D+01  2.50000000D+01  3.50000000D+01
 ( 3) -1.00000000D+00  1.20000000D+01  1.60000000D+01 -2.40000000D+01
***
 ( 4)  3.10000000D+01 -3.50000000D+01  2.50000000D+01  3.90000000D+01
 ( 5)  3.00000000D+00 -2.00000000D+00 -1.00000000D+00  1.20000000D+01
 ( 6)  5.00000000D+00  7.00000000D+00  3.10000000D+01 -3.50000000D+01
 ( 1) -5.00000000D+00  6.00000000D+01
 ( 2)  1.55000000D+02 -1.75000000D+02
 ( 3)  1.30000000D+01  1.40000000D+01
 ( 4)  8.70000000D+01 -3.50000000D+01
 ( 5)  1.60000000D+01 -2.40000000D+01
 ( 6)  2.50000000D+01  3.90000000D+01
 FILENAME :   VT
/SYS/ TIME-22:42:18 CPU-00:00:12
```

〔3〕 マルコフパラメータ表現形式システムから，ハンケル行列を得る例

```
/SYS/ TIME-22:37:29 CPU-00:00:05
V/M
 TO MAKE MATRIX VM
 SYSTEM NAME :   EX1
 SPECIFY NO. OF LOW & COLUMN BLOCKS (1-50),(1-50) :  3,7
 OUTPUT-MODE OF ( 6,14) MATRIX  VM        : _
 --- ( 6,14) MATRIX  VM       ---
 ( 1)  0.0              0.0              1.00000000D+00 -2.00000000D+00
 ( 2)  1.00000000D+00 -1.00000000D+00 -1.00000000D+00  5.00000000D+00
 ( 3)  1.00000000D+00 -2.00000000D+00 -1.00000000D+00  4.00000000D+00
 ( 4) -1.00000000D+00  5.00000000D+00  5.00000000D+00 -9.00000000D+00
 ( 5) -1.00000000D+00  4.00000000D+00  2.00000000D+00 -4.00000000D+00
 ( 6)  5.00000000D+00 -9.00000000D+00  3.00000000D+00  5.00000000D+00
 ( 1) -1.00000000D+00  4.00000000D+00  2.00000000D+00 -4.00000000D+00
 ( 2)  5.00000000D+00 -9.00000000D+00  3.00000000D+00  5.00000000D+00
 ( 3)  2.00000000D+00 -4.00000000D+00  3.00000000D+00 -2.00000000D+00
 ( 4)  3.00000000D+00  5.00000000D+00  5.00000000D+00  7.00000000D+00
 ( 5)  3.00000000D+00 -2.00000000D+00 -1.00000000D+00  1.20000000D+01
 ( 6)  5.00000000D+00  7.00000000D+00  3.10000000D+01 -3.50000000D+01
 ( 1)  3.00000000D+00 -2.00000000D+00 -1.00000000D+00  1.20000000D+01
 ( 2)  5.00000000D+00  7.00000000D+00  3.10000000D+01 -3.50000000D+01
 ( 3) -1.00000000D+00  1.20000000D+01  1.60000000D+01 -2.40000000D+01
***
 ( 4)  3.10000000D+01 -3.50000000D+01  2.50000000D+01  3.90000000D+01
 ( 5)  1.60000000D+01 -2.40000000D+01  0.0              0.0
 ( 6)  2.50000000D+01  3.90000000D+01  0.0              0.0
 ( 1)  1.60000000D+01 -2.40000000D+01
 ( 2)  2.50000000D+01  3.90000000D+01
 ( 3)  0.0              0.0
 ( 4)  0.0              0.0
 ( 5)  0.0              0.0
 ( 6)  0.0              0.0
 FILENAME :   VM
/SYS/ TIME-22:42:20 CPU-00:00:12
```

〔4〕 微分（差分）方程式表現形式システムから，対応する一つの状態空間表現形式システムの可制御性行列を得る例

```
/SYS/ TIME-22:37:31 CPU-00:00:05
V/D
 TO MAKE MATRIX VD
 SYSTEM NAME :    EX1
 SPECIFY NO. OF COLUMN BLOCKS (1- 7) :    7
 OUTPUT-MODE OF ( 6,14) MATRIX  VD       :
 --- ( 6,14) MATRIX  VD       ---
 ( 1) -1.00000000D+00  4.00000000D+00  0.0              0.0
 ( 2)  3.00000000D+00 -7.00000000D+00  5.00000000D+00 -5.00000000D+00
 ( 3)  1.00000000D+00 -2.00000000D+00 -1.00000000D+00  4.00000000D+00
 ( 4) -1.00000000D+00  5.00000000D+00  5.00000000D+00 -9.00000000D+00
 ( 5)  0.0             0.0             1.00000000D+00 -2.00000000D+00
 ( 6)  1.00000000D+00 -1.00000000D+00 -1.00000000D+00  5.00000000D+00
 ( 1)  5.00000000D+00 -1.00000000D+01 -5.00000000D+00  2.00000000D+01
 ( 2) -5.00000000D+00  2.50000000D+01  2.50000000D+01 -4.50000000D+01
 ( 3)  2.00000000D+00 -4.00000000D+00  3.00000000D+00 -2.00000000D+00
 ( 4)  3.00000000D+00  5.00000000D+00  5.00000000D+00  7.00000000D+00
 ( 5) -1.00000000D+00  4.00000000D+00  2.00000000D+00 -4.00000000D+00
 ( 6)  5.00000000D+00 -9.00000000D+00  3.00000000D+00  5.00000000D+00
 ( 1)  1.00000000D+01 -2.00000000D+01  1.50000000D+01 -1.00000000D+01
 ( 2)  1.50000000D+01  2.50000000D+01  2.50000000D+01  3.50000000D+01
 ( 3) -1.00000000D+00  1.20000000D+01  1.60000000D+01 -2.40000000D+01
***
 ( 4)  3.10000000D+01 -3.50000000D+01  2.50000000D+01  3.90000000D+01
 ( 5)  3.00000000D+00 -2.00000000D+00 -1.00000000D+00  1.20000000D+01
 ( 6)  5.00000000D+00  7.00000000D+00  3.10000000D+01 -3.50000000D+01
 ( 1) -5.00000000D+00  6.00000000D+01
 ( 2)  1.55000000D+02 -1.75000000D+02
 ( 3)  1.30000000D+01  1.40000000D+01
 ( 4)  8.70000000D+01 -3.50000000D+01
 ( 5)  1.60000000D+01 -2.40000000D+01
 ( 6)  2.50000000D+01  3.90000000D+01
 FILENAME :   VD
/SYS/ TIME-22:42:21 CPU-00:00:12
```

Vの可制御性構造の特徴抽出	S Y. I. 2
L C F	

機　能

ＳＹＳサブシステムのコマンドＶで得られた行列 $\boldsymbol{V}_{@} \in \mathbf{R}^{n \times m}$ から可制御性構造を特徴付ける次のパラメータを得る。

① Kronecker 不変量 ; $\{n_i\}_{i=1,\cdots,m}$, ν, n

② α および β パラメータ，Ｌ行列 ; $\boldsymbol{F} \in \mathbf{R}^{m \times n}$, $\boldsymbol{G} \in \mathbf{R}^{m \times m}$, $\boldsymbol{L} \in \mathbf{R}^{n \times n}$

③ Luenbergerの第１正準形 ; $\boldsymbol{A}_1 \in \mathbf{R}^{n \times n}$, $\boldsymbol{B}_1 \in \mathbf{R}^{n \times m}$, $\boldsymbol{T}_1 \in \mathbf{R}^{n \times n}$

④ Luenbergerの第２正準形 ; $\boldsymbol{A}_2 \in \mathbf{R}^{n \times n}$, $\boldsymbol{B}_2 \in \mathbf{R}^{n \times m}$, $\boldsymbol{T}_2 \in \mathbf{R}^{n \times n}$

理論概要

$\boldsymbol{V}_{@} \in \mathbf{R}^{n_v \times m_v}$, $m_v = k \cdot m$ を次のように分割する。

$$\boldsymbol{V}_{@} = [\boldsymbol{V}_j, \cdots\cdots, \boldsymbol{V}_k]$$
$$\boldsymbol{V}_j = [\boldsymbol{v}_{1j}, \cdots\cdots, \boldsymbol{v}_{mj}] \in \mathbf{R}^{n_v \times m}, \quad (j = 1, \sim, k)$$

今　$n_i\,(i=1,\sim,m)$ を $\boldsymbol{v}_{in_i}$ が

$$\boldsymbol{V}_1 \cdots\cdots \boldsymbol{V}_{n_i-1},\ \boldsymbol{v}_{1n_i} \cdots\cdots \boldsymbol{v}_{i-1\,n_i}$$

に線型従属となる最小の整数とする，このとき，

$$n = \sum_{i=1}^{m} n_i, \quad \nu = \max\{n_i\}$$

また，次式が成立する。

$$\boldsymbol{v}_{in_i} = \sum_{j=1}^{m} \alpha_{ij0}\, \boldsymbol{v}_{j1} + \sum_{j=1,\,n_j>1}^{m} \alpha_{ij1}\, \boldsymbol{v}_{j2} + \cdots\cdots + \sum_{j=1,\,n_j>n_i-1}^{m} \alpha_{ijn_i-1}\, \boldsymbol{v}_{jn_i-1}$$
$$+ \sum_{j=1,\,n_j>n_i}^{i-1} \beta_{ij}\, \boldsymbol{v}_{jni} \quad (i=1,\sim,\ m)$$

これから次の行列 $\boldsymbol{P} \in \mathbf{R}^{m \times n}$ を定義する

$$\boldsymbol{P} = [{}^t\boldsymbol{\alpha}_{ij}],\ i=1,\sim,m,\ j=1,\sim,m,\ \boldsymbol{\alpha}_{ij} \in \mathbf{R}^{n_j}$$

$$\boldsymbol{\alpha}_{ij} = \begin{cases} {}^t[\alpha_{ij0} \cdots\cdots \alpha_{ijn_j-1}] & (n_j \leq n_i) \\ {}^t[\alpha_{ij0} \cdots\cdots \alpha_{ijn_j-1}\ \beta_{ij}\ 0 \cdots\cdots 0] & (j<i,\ n_j>n_i) \\ {}^t[\alpha_{ij0} \cdots\cdots \alpha_{ijn_j-1}\ 0 \cdots\cdots 0] & (j>i,\ n_j>n_i) \end{cases}$$

・α パラメータを用いて次の行列 $\boldsymbol{F} \in \mathbf{R}^{m \times n}$ を定義する。

$$\boldsymbol{F} = [\boldsymbol{f}_1^0 \cdots\cdots \boldsymbol{f}_1^{n_1-1} \vdots \cdots\cdots \vdots \boldsymbol{f}_m^0 \cdots\cdots \boldsymbol{f}_m^{n_m-1}]$$

ただし，$\boldsymbol{f}_i^{\bar{k}} = {}^t[\ \boldsymbol{f}_{i1}^{\bar{k}} \cdots\cdots \boldsymbol{f}_{im}^{\bar{k}}\ ]$

$$\boldsymbol{f}_{ij}^{\bar{k}} = \begin{Bmatrix} \alpha_{ij\bar{k}}\ (n_j > \bar{k}) \\ \boldsymbol{0}\ \ (n_j \leq \bar{k}) \end{Bmatrix}$$

・βパラメータを用いて次の行列$\boldsymbol{G} \in \mathbf{R}^{m \times m}$を定義する。

$$\boldsymbol{G} = [\ \boldsymbol{g}_1 \cdots\cdots \boldsymbol{g}_m\ ]$$

ただし，$\boldsymbol{g}_i\ =\ {}^t[\ \boldsymbol{g}_{i1} \cdots\cdots \boldsymbol{g}_{im}\ ]$

$$\boldsymbol{g}_{ij} = \begin{cases} -\beta_{ij}\ (j<i,\ n_j > n_i),\ 1\,(j=i) \\ 0\quad (j>i \text{ or } j<i,\ n_j \leq n_i) \end{cases}$$

・αおよびβパラメータを用いて次の行列$\boldsymbol{L} \in \mathbf{R}^{n \times m}$を定義する。

$$\boldsymbol{L} = \begin{bmatrix} \boldsymbol{L}_{11} & \cdots\cdots & \boldsymbol{L}_{m1} \\ \vdots & & \vdots \\ \boldsymbol{L}_{1m} & \cdots\cdots & \boldsymbol{L}_{mm} \end{bmatrix},\quad \boldsymbol{L}_{ij} \in \mathbf{R}^{n_j \times n_i}$$

ただし，
$$\boldsymbol{L}_{ij} = -[\ \boldsymbol{S}_i a_{ij} \cdots\cdots \boldsymbol{S}_i^{n_i} a_{ii}\ ] + \begin{bmatrix} \boldsymbol{0} & & 1 \\ & \cdot^{\cdot^{\cdot}} & \\ 1 & & \boldsymbol{0} \end{bmatrix}$$
$$\boldsymbol{L}_{ij} = -[\ \boldsymbol{S}_j a_{ij} \cdots\cdots \boldsymbol{S}_j^{n_i} a_{ij}\ ]$$
$$\boldsymbol{S}_j = \begin{bmatrix} 0 & 1 & & \boldsymbol{0} \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ \boldsymbol{0} & & & 0 \end{bmatrix}$$

・Brunovskyの正準形$\boldsymbol{A}_0 \in \mathbf{R}^{n \times n}$，$\boldsymbol{B}_0 \in \mathbf{R}^{n \times m}$は次式で与えられる。

$$\boldsymbol{A}_0 = \mathrm{diag}\ \{\boldsymbol{A}_i\}_{i=1,\sim,m},\quad \boldsymbol{A}_i = \boldsymbol{S}_i \in \mathbf{R}^{n_i \times n_i}$$
$$\boldsymbol{B}_0 = \mathrm{diag}\ \{\boldsymbol{b}_i\}_{i=1,\sim,m},\quad \boldsymbol{b}_i = \boldsymbol{e}_{n_i} \in \mathbf{R}^{n_i}$$

ここで，$\boldsymbol{e}_{n_i} = {}^t(0, \cdots, 0, 1, 0, \cdots, 0)$；$n_i$番目が1の単位ベクトル。

・Luenbergerの第1正準形$\boldsymbol{A}_1 \in \mathbf{R}^{n \times n}$，$\boldsymbol{B}_1 \in \mathbf{R}^{n \times m}$および対応する可制御性空間の基底行列$\boldsymbol{T}_1 \in \mathbf{R}^{nv \times n}$は次式で与えられる。

$$\boldsymbol{A}_1 = {}^t(\boldsymbol{A}_0 + \boldsymbol{B}_0\, {}^t\boldsymbol{P}),\quad \boldsymbol{B}_1 = \mathrm{diag}\{\boldsymbol{b}_i\}_{i=1,\sim,m},\quad \boldsymbol{b}_i = \boldsymbol{e}_1 \in \mathbf{R}^{n_i}$$
$$\boldsymbol{T}_1 = [\ \boldsymbol{v}_{11} \cdots\cdots \boldsymbol{v}_{1n_1} \vdots \cdots\cdots \vdots\ \boldsymbol{v}_{m1} \cdots\cdots \boldsymbol{v}_{mn_m}\ ]$$

ここで，$\boldsymbol{e}_1 = {}^t(1, 0, 0, \cdots\cdots, 0)$ 1番目が1の単位ベクトル。

・Luenbergerの第2正準形$\boldsymbol{A}_2 \in \mathbf{R}^{n \times n}$，$\boldsymbol{B}_2 \in \mathbf{R}^{n \times m}$および対応する可制御性空間の基底行列$\boldsymbol{T}_2 \in \mathbf{R}^{nv \times n}$は次式で与えられる。

$$\boldsymbol{A}_2 = \boldsymbol{A}_0 + \boldsymbol{B}_0 \boldsymbol{G}^{-1} \boldsymbol{F}$$
$$\boldsymbol{B}_2 = \boldsymbol{B}_0 \boldsymbol{G}$$
$$\boldsymbol{T}_2 = \boldsymbol{T}_1 \cdot \boldsymbol{L}$$

実 行 例

〔1〕 V_Sの可制御性構造の特徴抽出例

```
/SYS/ TIME-22:37:32 CPU-00:00:05
LCF
TO INVESTIGATE CANONICAL STRUCTURE OF V@
SPECIFY (NV,MV,M,K) :   3,8,2,4
INPUT -MODE OF ( 3, 8) MATRIX          : : F
FILENAME :   VS
TYPE OR REVISE ? _
GIVE EPSILON, TO DETERMINE RANK OF MATRIX V@  EPS= 1.0D-10
COLUMN INDEPENDENCY OF MATRIX V@ IS INVESTIGATED
 ( 1)
  1    1
 ( 2)
  1    0
 ( 3)
  0    0

CONDITION NUMBERS ARE AS FOLLOWS
--- ( 3, 2) MATRIX CNUM      ---
( 1)   1.000000000D+00  4.142135620-01
( 2)   3.819660110-01   4.749814320-25
( 3)   1.029632600-20   1.135959700-27
CHANGE THE EPSILON ?   N
*** CANONICAL STRUCTURE OF V@ ***
1) RANK V@  =   3  (EPS =    1.000000000D-10)
2) MAXIMUM OF KRONECKER'S INVARIANTS :  2
3) KRONECKER'S INVARIANTS
      2  1
4) SITUATION OF INDIPENDENT COLUMNS IN V@( 3)
      1  1
***
      1  0
      0  0
5) ALFA-PARAMETERS
--- ( 2, 3) MATRIX  FC        ---
( 1)   7.000000000D+00 -1.000000000D+00   4.000000000D+00
( 2)   3.000000000D+00  0.0               1.000000000D+00
FILENAME : _
6) BETA-PARAMETERS
--- ( 2, 2) MATRIX  GC        ---
( 1)   1.000000000D+00  2.000000000D+00
( 2)   0.0              1.000000000D+00
FILENAME : _
7) L-MATRIX
--- ( 3, 3) MATRIX  L         ---
( 1)   1.000000000D+00  1.000000000D+00   2.000000000D+00
( 2)   1.000000000D+00  0.0               0.0
( 3)   0.0              0.0               1.000000000D+00
FILENAME : _
8) BASIS OF STATE SPACE
--- ( 3, 3) MATRIX  T         ---
( 1)  0.0              1.000000000D+00  0.0
( 2)  1.000000000D+00 -1.000000000D+00 -2.000000000D+00
( 3)  0.0              0.0              1.000000000D+00
***
FILENAME : _
9) LUENBERGER'S FIRST CANONICAL FORM
--- ( 3, 3) MATRIX  A1        ---
( 1)  0.0              7.000000000D+00  4.000000000D+00
( 2)  1.000000000D+00 -1.000000000D+00 -2.000000000D+00
( 3)  0.0              3.000000000D+00  1.000000000D+00
FILENAME : _
--- ( 3, 2) MATRIX  B1        ---
( 1)  1.000000000D+00  0.0
( 2)  0.0              0.0
( 3)  0.0              1.000000000D+00
FILENAME : _
10) LUENBERGER'S SECOND CANONICAL FORM
--- ( 3, 3) MATRIX  A2        ---
( 1)   0.0              1.000000000D+00  0.0
( 2)   1.000000000D+00 -1.000000000D+00  2.000000000D+00
( 3)   3.000000000D+00  0.0              1.000000000D+00
FILENAME : _
--- ( 3, 2) MATRIX  B2        ---
( 1)   0.0              0.0
( 2)   1.000000000D+00 -2.000000000D+00
( 3)   0.0              1.000000000D+00
FILENAME : _
/SYS/ TIME-22:42:22 CPU-00:00:12
```

〔2〕 V_Tの可制御性構造の特徴抽出例

```
/SYS/ TIME-22:37:33 CPU-00:00:05
LCF
 TO INVESTIGATE CANONICAL STRUCTURE OF V@
 SPECIFY (NV,MV,M,K) :   6,14,2,7
 INPUT -MODE OF ( 6,14) MATRIX          : : F
 FILENAME :   VT
 TYPE OR REVISE ? _
 GIVE EPSILON, TO DETERMINE RANK OF MATRIX V@  EPS= 1.0D-10
 COLUMN INDEPENDENCY OF MATRIX V@ IS INVESTIGATED
  ( 1)
   1    1
  ( 2)
   1    0
  ( 3)
   0    0

 CONDITION NUMBERS ARE AS FOLLOWS
 ---- ( 3, 2) MATRIX CNUM        ---
 ( 1)  1.00000000D+00  1.17013270D+00
 ( 2)  9.15931411D-01  5.92175494D-16
 ( 3)  2.36802512D-16  1.72910068D-16
 CHANGE THE EPSILON ?   N
 *** CANONICAL STRUCTURE OF V@ ***
 1) RANK V@   =   3  (EPS =    1.00000000D-10)
 2) MAXIMUM OF KRONECKER'S INVARIANTS :  2
 3) KRONECKER'S INVARIANTS
      2  1
 4) SITUATION OF INDIPENDENT COLUMNS IN V@( 3)
      1  1
 ***
      1  0
      0  0
 5) ALFA-PARAMETERS
 --- ( 2, 3) MATRIX  FC        ---
 ( 1)  7.00000000D+00 -1.00000000D+00  4.00000000D+00
 ( 2)  3.00000000D+00  0.0             1.00000000D+00
 FILENAME : _
 6) BETA-PARAMETERS
 --- ( 2, 2) MATRIX  GC        ---
 ( 1)  1.00000000D+00  2.00000000D+00
 ( 2)  0.0             1.00000000D+00
 FILENAME : _
 7) L-MATRIX
 --- ( 3, 3) MATRIX  L         ---
 ( 1)  1.00000000D+00  1.00000000D+00  2.00000000D+00
 ( 2)  1.00000000D+00  0.0             0.0
 ( 3)  0.0             0.0             1.00000000D+00
 FILENAME : _
 8) BASIS OF STATE SPACE
 --- ( 6, 3) MATRIX  T         ---
 ( 1) -1.00000000D+00  0.0             4.00000000D+00
 ( 2)  3.00000000D+00  5.00000000D+00 -7.00000000D+00
 ( 3)  1.00000000D+00 -1.00000000D+00 -2.00000000D+00
 ( 4) -1.00000000D+00  5.00000000D+00  5.00000000D+00
 ( 5)  0.0             1.00000000D+00  0.0
 ( 6)  1.00000000D+00 -1.00000000D+00 -1.00000000D+00
 FILENAME : _
 9) LUENBERGER'S FIRST CANONICAL FORM
 --- ( 3, 3) MATRIX  A1        ---
 ( 1)  0.0             7.00000000D+00  4.00000000D+00
 ( 2)  1.00000000D+00 -1.00000000D+00 -2.00000000D+00
 ( 3)  0.0             3.00000000D+00  1.00000000D+00
 FILENAME : _
 --- ( 3, 2) MATRIX  B1        ---
 ( 1)  1.00000000D+00  0.0
 ( 2)  0.0             0.0
 ( 3)  0.0             1.00000000D+00
 FILENAME : _
 10) LUENBERGER'S SECOND CANONICAL FORM
 --- ( 3, 3) MATRIX  A2        ---
 ( 1)  0.0             1.00000000D+00  0.0
 ( 2)  1.00000000D+00 -1.00000000D+00  2.00000000D+00
 ( 3)  3.00000000D+00  0.0             1.00000000D+00
 FILENAME : _
 --- ( 3, 2) MATRIX  B2        ---
 ( 1)  0.0             0.0
 ( 2)  1.00000000D+00 -2.00000000D+00
 ( 3)  0.0             1.00000000D+00
 FILENAME : _
 /SYS/ TIME-22:42:23 CPU-00:00:12
```

〔3〕 V_Mの可制御性構造の特徴抽出例

```
/SYS/ TIME-22:37:34 CPU-00:00:05
LCF
 TO INVESTIGATE CANONICAL STRUCTURE OF V@
 SPECIFY (NV,MV,M,K) :   6,14,2,7
 INPUT MODE OF ( 6,14) MATRIX          : : F
 FILENAME :   VM
 TYPE OR REVISE ? _
 GIVE EPSILON, TO DETERMINE RANK OF MATRIX V@  EPS= 1.0D-10
 COLUMN INDEPENDENCY OF MATRIX V@ IS INVESTIGATED
  ( 1)
   1   1
  ( 2)
   1   0
  ( 3)
   0   0

 CONDITION NUMBERS ARE AS FOLLOWS
 --- ( 3, 2) MATRIX CNUM        ---
 ( 1)  1.000000000D+00  1.683305710D+00
 ( 2)  8.471886120D-01  3.014136990D-16
 ( 3)  1.464061180D-16  3.972305740D-16
 CHANGE THE EPSILON ?   N
 *** CANONICAL STRUCTURE OF V@ ***
 1) RANK V@  =  3  (EPS =    1.000000000D-10)
 2) MAXIMUM OF KRONECKER'S INVARIANTS :  2
 3) KRONECKER'S INVARIANTS
      2  1
 4) SITUATION OF INDIPENDENT COLUMNS IN V@( 3)
      1  1
 ***
      1  0
      0  0
 5) ALFA-PARAMETERS
 --- ( 2, 3) MATRIX  FC         ---
 ( 1)  7.000000000D+00 -1.000000000D+00  4.000000000D+00
 ( 2)  3.000000000D+00  0.0              1.000000000D+00
 FILENAME : _
 6) BETA-PARAMETERS
 --- ( 2, 2) MATRIX  GC         ---
 ( 1)  1.000000000D+00  2.000000000D+00
 ( 2)  0.0              1.000000000D+00
 FILENAME : _
 7) L-MATRIX
 --- ( 3, 3) MATRIX  L          ---
 ( 1)  1.000000000D+00  1.000000000D+00  2.000000000D+00
 ( 2)  1.000000000D+00  0.0              0.0
 ( 3)  0.0              0.0              1.000000000D+00
 FILENAME : _
 8) BASIS OF STATE SPACE
 --- ( 6, 3) MATRIX  T          ---
 ( 1)  0.0              1.000000000D+00  0.0
 ( 2)  1.000000000D+00 -1.000000000D+00 -1.000000000D+00
 ( 3)  1.000000000D+00 -1.000000000D+00 -2.000000000D+00
 ( 4) -1.000000000D+00  5.000000000D+00  5.000000000D+00
 ( 5) -1.000000000D+00  2.000000000D+00  4.000000000D+00
 ( 6)  5.000000000D+00  3.000000000D+00 -9.000000000D+00
 FILENAME : _
 9) LUENBERGER'S FIRST CANONICAL FORM
 --- ( 3, 3) MATRIX  A1         ---
 ( 1)  0.0              7.000000000D+00  4.000000000D+00
 ( 2)  1.000000000D+00 -1.000000000D+00 -2.000000000D+00
 ( 3)  0.0              3.000000000D+00  1.000000000D+00
 FILENAME : _
 --- ( 3, 2) MATRIX  B1         ---
 ( 1)  1.000000000D+00  0.0
 ( 2)  0.0              0.0
 ( 3)  0.0              1.000000000D+00
 FILENAME : _
 10) LUENBERGER'S SECOND CANONICAL FORM
 --- ( 3, 3) MATRIX  A2         ---
 ( 1)  0.0              1.000000000D+00  0.0
 ( 2)  1.000000000D+00 -1.000000000D+00  2.000000000D+00
 ( 3)  3.000000000D+00  0.0              1.000000000D+00
 FILENAME : _
 --- ( 3, 2) MATRIX  B2         ---
 ( 1)  0.0              0.0
 ( 2)  1.000000000D+00 -2.000000000D+00
 ( 3)  0.0              1.000000000D+00
 FILENAME : _
 /SYS/ TIME-22:42:24 CPU-00:00:12
```

〔4〕 V_Dの可制御性構造の特徴抽出例

```
/SYS/ TIME-22:37:37 CPU-00:00:05
LCF
 TO INVESTIGATE CANONICAL STRUCTURE OF V@
 SPECIFY (NV,MV,M,K) :   6,14,2,7
 INPUT MODE OF ( 6,14) MATRIX        : : F
 FILENAME :  VD
 TYPE OR REVISE ? _
 GIVE EPSILON, TO DETERMINE RANK OF MATRIX V@  EPS= 1.0D-10
 COLUMN INDEPENDENCY OF MATRIX V@ IS INVESTIGATED
  ( 1)
   1   1
  ( 2)
   1   0
  ( 3)
   0   0

 CONDITION NUMBERS ARE AS FOLLOWS
 --- ( 3, 2) MATRIX CNUM      ---
 ( 1)  1.000000000D+00  1.170132700D+00
 ( 2)  9.159314110D-01  5.921754940D-16
 ( 3)  2.368025120D-16  1.729100680D-16
 CHANGE THE EPSILON ?   N
 *** CANONICAL STRUCTURE OF V@ ***
 1) RANK V@  =   3  (EPS =    1.000000000D-10)
 2) MAXIMUM OF KRONECKER'S INVARIANTS :  2
 3) KRONECKER'S INVARIANTS
       2  1
 4) SITUATION OF INDIPENDENT COLUMNS IN V@( 3)
       1  1
 ***
       1  0
       0  0
 5) ALFA PARAMETERS
 --- ( 2, 3) MATRIX  FC        ---
 ( 1)  7.000000000D+00 -1.000000000D+00  4.000000000D+00
 ( 2)  3.000000000D+00  0.0              1.000000000D+00
 FILENAME : _
 6) BETA-PARAMETERS
 --- ( 2, 2) MATRIX  GC        ---
 ( 1)  1.000000000D+00  2.000000000D+00
 ( 2)  0.0              1.000000000D+00
 FILENAME : _
 7) L-MATRIX
 --- ( 3, 3) MATRIX  L         ---
 ( 1)  1.000000000D+00  1.000000000D+00  2.000000000D+00
 ( 2)  1.000000000D+00  0.0              0.0
 ( 3)  0.0              0.0              1.000000000D+00
 FILENAME : _
 8) BASIS OF STATE SPACE
 --- ( 6, 3) MATRIX  T         ---
 ( 1) -1.000000000D+00  0.0              4.000000000D+00
 ( 2)  3.000000000D+00  5.000000000D+00 -7.000000000D+00
 ( 3)  1.000000000D+00 -1.000000000D+00 -2.000000000D+00
 ( 4) -1.000000000D+00  5.000000000D+00  5.000000000D+00
 ( 5)  0.0              1.000000000D+00  0.0
 ( 6)  1.000000000D+00 -1.000000000D+00 -1.000000000D+00
 FILENAME : _
 9) LUENBERGER'S FIRST CANONICAL FORM
 --- ( 3, 3) MATRIX  A1        ---
 ( 1)  0.0              7.000000000D+00  4.000000000D+00
 ( 2)  1.000000000D+00 -1.000000000D+00 -2.000000000D+00
 ( 3)  0.0              3.000000000D+00  1.000000000D+00
 FILENAME : _
 --- ( 3, 2) MATRIX  B1        ---
 ( 1)  1.000000000D+00  0.0
 ( 2)  0.0              0.0
 ( 3)  0.0              1.000000000D+00
 FILENAME : _
 10) LUENBERGER'S SECOND CANONICAL FORM
 --- ( 3, 3) MATRIX  A2        ---
 ( 1)  0.0              1.000000000D+00  0.0
 ( 2)  1.000000000D+00 -1.000000000D+00  2.000000000D+00
 ( 3)  3.000000000D+00  0.0              1.000000000D+00
 FILENAME : _
 --- ( 3, 2) MATRIX  B2        ---
 ( 1)  0.0              0.0
 ( 2)  1.000000000D+00 -2.000000000D+00
 ( 3)  0.0              1.000000000D+00
 FILENAME : _
 /SYS/ TIME-22:42:26 CPU-00:00:12
```

状態空間表現形式システム行列の取得	SY. I. 3
SMAT	

機　能

状態空間表現形式システムΣ_Sから，次の行列$\boldsymbol{S} \in \mathbf{R}^{(n+p)\times(n+m)}$を得る。

$$\text{連続系の場合}\ \boldsymbol{S} = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{D} \end{bmatrix}$$

$$\text{離散系の場合}\ \boldsymbol{S} = \begin{bmatrix} \boldsymbol{A}-\boldsymbol{I}_n & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{D} \end{bmatrix}$$

実 行 例

〔1〕　連続系状態空間表現形式システムから$\boldsymbol{S}$マトリックスを得る例

```
/SYS/ TIME-22:37:35 CPU-00:00:05
SMAT
 TO MAKE SYSTEM MATRIX
 SYSTEM NAME :    EX1
 SYSTEM EX1       HAS THE FOLLOWING SYSTEMDATAS
  1) EX1.S0       (    3) ;SYSIN    /S /C /  /    (87/ 7/ 3) :EX1
  2) EX1.S1       (    3) ;CMR      /T /C /2 /    (87/ 9/17) :AB2
  3) EX1.S2       (    3) ;CMR      /M /C /2 /    (87/ 9/17) :AB2
  4) EX1.S3       (    3) ;COOD     /1 /  /  /    (87/ 9/25) :
  5) EX1.S4       (    3) ;COOD     /2 /  /  /    (87/ 9/25) :COOD/2
 WHICH SYSTEMDATA IS USED ? :   1
 OUTPUT-MODE OF ( 5, 5) MATRIX SYSMAT    : _
 --- ( 5, 5) MATRIX SYSMAT   ---
 ( 1)   0.0              1.00000000D+00  0.0             0.0
 ( 2)   1.00000000D+00 -1.00000000D+00  2.00000000D+00  1.00000000D+00
 ( 3)   3.00000000D+00  0.0             1.00000000D+00  0.0
 ( 4)   1.00000000D+00  0.0             0.0             0.0
 ( 5)   0.0             1.00000000D+00  1.00000000D+00  0.0
 ( 1)   0.0
 ( 2)  -2.00000000D+00
 ( 3)   1.00000000D+00
 ( 4)   0.0
 ( 5)   0.0
 ***
 FILENAME :    SMAT
 /SYS/ TIME-22:42:27 CPU-00:00:12
```

可制御性・可観測性グラム行列の取得	SY. I. 4
V V	

機　能

状態空間表現モデルから次の行列を求める。

① $\boldsymbol{V}_k \cdot {}^t\boldsymbol{V}_k$ （可制御性グラム行列（k次））

② ${}^t\boldsymbol{N}_k \cdot \boldsymbol{N}_k$ （可観測性グラム行列（k次））

③ $\boldsymbol{V}_\infty \cdot {}^t\boldsymbol{V}_\infty$ （$\boldsymbol{V}_k \cdot {}^t\boldsymbol{V}_k$（$k \to \infty$））

④ ${}^t\boldsymbol{N}_\infty \cdot \boldsymbol{N}_\infty$ （${}^t\boldsymbol{N}_k \cdot \boldsymbol{N}_k$（$k \to \infty$））

理論概要

各行列は次のようにして求まる。

$$\boldsymbol{V}_k = [\ \boldsymbol{B},\ \boldsymbol{AB},\ \cdots\cdots,\ \boldsymbol{A}^{k-1}\boldsymbol{B}\ ]$$
$$\boldsymbol{N}_k = {}^t[\ \boldsymbol{C},\ \boldsymbol{AC},\ \cdots\cdots,\ \boldsymbol{A}^{k-1}\boldsymbol{C}\ ]$$

$$\boldsymbol{V}_k \cdot {}^t\boldsymbol{V}_k = \sum_{i=1}^{k} \boldsymbol{A}^{i-1}\boldsymbol{B} \cdot {}^t\boldsymbol{B}\,{}^t\boldsymbol{A}^{i-1}$$ ；可制御性グラム行列

$${}^t\boldsymbol{N}_k \cdot \boldsymbol{N}_k = \sum_{i=1}^{k} {}^t\boldsymbol{A}^{i-1}\,{}^t\boldsymbol{C} \cdot \boldsymbol{C}\,\boldsymbol{A}^{i-1}$$ ；可観測性グラム行列

$\boldsymbol{V}_\infty {}^t\boldsymbol{V}_\infty$，${}^t\boldsymbol{N}_\infty \boldsymbol{N}_\infty$は安定なシステムに対してのみ，それぞれ次のLyapunov方程式を解いて求められる。（MT.II.12参照）

$$\left.\begin{array}{l} \boldsymbol{X} = \boldsymbol{A}\,\boldsymbol{X}\,{}^t\boldsymbol{A} + \boldsymbol{B}\,{}^t\boldsymbol{B} \longrightarrow \boldsymbol{X} = \boldsymbol{V}_\infty \cdot {}^t\boldsymbol{V}_\infty \\ \boldsymbol{X} = {}^t\boldsymbol{A}\,\boldsymbol{X}\,\boldsymbol{A} + {}^t\boldsymbol{C}\,\boldsymbol{C} \longrightarrow \boldsymbol{X} = {}^t\boldsymbol{N}_\infty \cdot \boldsymbol{N}_\infty \end{array}\right\}\ \text{離散系}$$

$$\left.\begin{array}{l} \boldsymbol{X}\,\boldsymbol{A} + {}^t\boldsymbol{A}\,\boldsymbol{X} = -\boldsymbol{B}\,{}^t\boldsymbol{B} \longrightarrow \boldsymbol{X} = \int_0^\infty \exp(\boldsymbol{A}\tau)\,\boldsymbol{B}\,{}^t\boldsymbol{B}\exp({}^t\boldsymbol{A}\tau)\,d\tau = \boldsymbol{V}_\infty \cdot {}^t\boldsymbol{V}_\infty \\ \boldsymbol{X}\,{}^t\boldsymbol{A} + \boldsymbol{A}\,\boldsymbol{X} = -{}^t\boldsymbol{C}\,\boldsymbol{C} \longrightarrow \boldsymbol{X} = \int_0^\infty \exp({}^t\boldsymbol{A}\tau)\,{}^t\boldsymbol{C}\,\boldsymbol{C}\exp(\boldsymbol{A}\tau)\,d\tau = {}^t\boldsymbol{N}_\infty \cdot \boldsymbol{N}_\infty \end{array}\right\}\ \text{連続系}$$

実行例

〔1〕　3次の可制御性グラム行列算出例

```
 /SYS/ TIME-22:37:38 CPU-00:00:05
VV
 TO MAKE MATRIX (1) VV'(K) (2) N'N(K)
  (3) VV'(INFINITE) (4) N'N(INFINITE)
 WHICH MATRIX ? :  1
 SYSTEM NAME :  EX1
 SPECIFY K (N= 3) :  3
 OUTPUT-MODE OF ( 3, 3) MATRIX  VV'       : _
 --- ( 3, 3) MATRIX  VV'      ---
 ( 1)  2.200000000D+01 -2.700000000D+01 -2.500000000D+01
 ( 2) -2.700000000D+01  4.200000000D+01  2.800000000D+01
 ( 3) -2.500000000D+01  2.800000000D+01  3.600000000D+01
 FILENAME :   VVK
 /SYS/ TIME-22:42:28 CPU-00:00:12
```

〔2〕　3次の可観測性グラム行列算出例

```
 /SYS/ TIME-22:37:39 CPU-00:00:05
VV
 TO MAKE MATRIX (1) VV'(K) (2) N'N(K)
  (3) VV'(INFINITE) (4) N'N(INFINITE)
 WHICH MATRIX ? :  2
 SYSTEM NAME :  EX1
 SPECIFY K (N= 3) :  3
 OUTPUT-MODE OF ( 3, 3) MATRIX  N'N       : _
 --- ( 3, 3) MATRIX  N'N      ---
 ( 1)  8.200000000D+01  3.500000000D+01  2.200000000D+01
 ( 2)  3.500000000D+01  2.900000000D+01  1.000000000D+00
 ( 3)  2.200000000D+01  1.000000000D+00  1.500000000D+01
 FILENAME :   NNK
 /SYS/ TIME-22:42:29 CPU-00:00:12
```

〔3〕　可制御性グラム行列（k→∞）の算出例

```
 /SYS/ TIME-22:37:40 CPU-00:00:05
VV
 TO MAKE MATRIX (1) VV'(K) (2) N'N(K)
  (3) VV'(INFINITE) (4) N'N(INFINITE)
 WHICH MATRIX ? :  3
 SYSTEM NAME :  EX1
 OUTPUT-MODE OF ( 3, 3) MATRIX  VV'       :_
 --- ( 3, 3) MATRIX  VV'      ---
 ( 1)  9.527082400D-01  0.0              0.0
 ( 2) -1.845564360D-01 -3.078327250D-02 -1.426880520D-02
 ( 3) -1.808517090D+00  1.502755940D+00  6.931160660D-01
 FILENAME :  VVI
 /SYS/ TIME-22:42:35 CPU-00:00:12
```

〔4〕　可観測性グラム行列（k→∞）の算出例

```
 /SYS/ TIME-22:37:54 CPU-00:00:05
VV
 TO MAKE MATRIX (1) VV'(K) (2) N'N(K)
  (3) VV'(INFINITE) (4) N'N(INFINITE)
 WHICH MATRIX ? :  4
 SYSTEM NAME :  EX1
 OUTPUT-MODE OF ( 3, 3) MATRIX  N'N       : _
 --- ( 3, 3) MATRIX  N'N      ---
 ( 1)  8.222060550D+00 -5.797295840D+00 -1.851261880D+01
 ( 2) -2.272428490D-01  1.827906850D-01  5.102612730D-01
 ( 3)  3.213957600D-01 -2.439219930D-01 -7.224944770D-01
 FILENAME :   NNI
 /SYS/ TIME-22:42:35 CPU-00:00:12
```

定常リカッチ行列方程式の求解	SY. I. 5
RCT	

機　能

定常リカッチ行列方程式の解をPotterの方法を用いて解く。

理論概要[23),24)]

◆次の定常リカッチ方程式

$$\begin{cases} \boldsymbol{PA} + {}^t\boldsymbol{A}\ \boldsymbol{P} - \boldsymbol{PBR}^{-1}\,{}^t\boldsymbol{B}\boldsymbol{P} + {}^t\boldsymbol{HQH} = \boldsymbol{0} & \text{(連続型)} \\ \boldsymbol{P} = {}^t\boldsymbol{A}\ \boldsymbol{PA} - {}^t\boldsymbol{APB}\ (\boldsymbol{R} + {}^t\boldsymbol{BPB})\ {}^t\boldsymbol{BPA} + {}^t\boldsymbol{HQH} & \text{(離散型)} \end{cases}$$

の解$\boldsymbol{P} \in \mathbf{R}^{n\times n}$を求める。

ただし，$\boldsymbol{A} \in \mathbf{R}^{n\times n}$，$\boldsymbol{B} \in \mathbf{R}^{n\times m}$，$\boldsymbol{H} \in \mathbf{R}^{p\times n}$，$\boldsymbol{Q} \in \mathbf{R}^{p\times p}$，$\boldsymbol{R} \in \mathbf{R}^{m\times m}$，$\boldsymbol{Q}$，$\boldsymbol{R} > 0$

$$\boldsymbol{F} = \begin{cases} \boldsymbol{R}^{-1}\,{}^t\boldsymbol{BP} & \text{(連続型)} \\ (\boldsymbol{R} + {}^t\boldsymbol{B}\ \ \boldsymbol{PB})^{-1}\,{}^t\boldsymbol{BPA} & \text{(離散型)} \end{cases}$$

を求める。$\boldsymbol{F} \in \boldsymbol{R}^{m\times n}$

　$(\boldsymbol{A} - \boldsymbol{BF})$ の固有値 $\{\lambda_i\}_{i=1,\sim,n}$ も求める。

◆アルゴリズム

Step 1：次の行列$\boldsymbol{M} \in \mathbf{R}^{2n\times 2n}$を構成する。

$$\boldsymbol{M} = \begin{cases} \begin{bmatrix} \boldsymbol{A} & -\boldsymbol{BR}^{-1}\,{}^t\boldsymbol{B} \\ -{}^t\boldsymbol{HQH} & -{}^t\boldsymbol{A} \end{bmatrix} & \text{(連続型)} \\ \begin{bmatrix} \boldsymbol{A} + \boldsymbol{BR}^{-1}\,{}^t\boldsymbol{B}\,{}^t\boldsymbol{A}^{-1}\,{}^t\boldsymbol{HQH} & -\boldsymbol{BR}^{-1}\,{}^t\boldsymbol{B}\,{}^t\boldsymbol{A}^{-1} \\ -{}^t\boldsymbol{A}^{-1}\,{}^t\boldsymbol{HQH} & {}^t\boldsymbol{A}^{-1} \end{bmatrix} & \text{(離散型)} \end{cases}$$

Step 2：$\boldsymbol{M}$の固有値の中で実部が負であるものを $\{\lambda_i\}_{i=1,\sim,n}$ とし，対応する固有ベクトルを $\left\{\begin{bmatrix} \boldsymbol{v}_i \\ \boldsymbol{u}_i \end{bmatrix}\right\}_{i=1,\sim,n}$ $(\boldsymbol{u}_i \in \mathbf{R}^n$，$\boldsymbol{v}_i \in \mathbf{R}^n)$ をするとき，定常リカッチ方程式の解は次式で与えられる。

$$\boldsymbol{P} = [\boldsymbol{u}_1 \cdots\cdots \boldsymbol{u}_n]\ [\boldsymbol{v}_1 \cdots\cdots \boldsymbol{v}_n]^{-1}$$

(注) $\boldsymbol{M}$の固有値は複素平面上で実軸と虚軸に対称に分布。

実 行 例

〔1〕 定常リカッチ方程式の解の取得例

```
 /SYS/ TIME-22:37:55 CPU-00:00:05
RCT
 TO SOLVE RICCATI EQUATION
 SPECIFY (N,M,P) :   3,2,2
 (1) CONTINUOUS OR (2) DISCRETE TYPE ? :   1
 INPUT -MODE OF ( 3, 3) MATRIX   A     : _
 ( 1) ROW
0,1,0
 ( 2) ROW
1,-1,2
 ( 3) ROW
3,0,1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 3, 2) MATRIX   B     : _
 ( 1) ROW
0,0
 ( 2) ROW
1,-2
 ( 3) ROW
0,1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 3) MATRIX   H     : _
 ( 1) ROW
1,0,0
 ( 2) ROW
0,1,1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX   Q     : I
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX   R     : I
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 3, 3) MATRIX   P     : _
 --- ( 3, 3) MATRIX   P       ---
 ( 1)  4.33231300D+01  1.33502145D+01  2.53789970D+01
 ( 2)  1.33502145D+01  4.32588509D+00  8.07265264D+00
 ( 3)  2.53789970D+01  8.07265264D+00  1.59563980D+01
 FILENAME : _
 OUTPUT-MODE OF ( 2, 3) MATRIX   F     : _
 --- ( 2, 3) MATRIX   F       ---
 ( 1)  1.33502145D+01  4.32588509D+00  8.07265264D+00
 ( 2) -1.32143208D+00 -5.79117535D-01 -1.88907257D-01
 FILENAME : _
 OUTPUT-MODE OF ( 3, 2) MATRIX POLE    : _
 --- ( 3, 2) MATRIX POLE     ---
 ( 1) -1.40205713D+00  1.43824560D+00
 ( 2) -1.40205713D+00 -1.43824560D+00
 ( 3) -2.49109863D+00  0.0
 FILENAME : _

 NORM OF EQUATION ERROR :    1.24621380D-09
 /SYS/ TIME-22:42:44 CPU-00:00:12
```

入力時系列データに対する応答	SY. I. 6
$\mathrm{FILT}\begin{bmatrix} /\mathrm{S} \\ /\mathrm{D} \end{bmatrix}$	

機　能

任意の時系列データを入力し，システムの応答を求める。

◆第1スイッチが／Sの場合，離散系状態空間表現形式システムに任意の時系列データを入力し，その応答である時系列データを得る。

◆第1スイッチが／Dの場合，離散系微分（差分）方程式表現形式システムに任意の時系列データを入力し，その応答である時系列データを得る。

理論概要

◆第1スイッチが／Sの場合

$$\text{離散系のシステム}\Sigma_S \begin{cases} \boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k) \\ \boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k) + \boldsymbol{D}\boldsymbol{u}(k) \end{cases}$$

$\left\{\boldsymbol{u}(k),\ k=0,1,\cdots\cdots,\ N\right\}$と$\boldsymbol{x}(0)$を与えて$\left\{\boldsymbol{y}(k),\ k=0,1,\cdots\cdots,\ N\right\}$を計算する。

◆第1スイッチが／Dの場合

Σ_Dとしては正規化された次式で表わせる離散系システムを考える。

$$\sum_{i=0}^{n_D}\boldsymbol{T}_i\ \boldsymbol{y}(k-i) = \sum_{i=1}^{n_D}\boldsymbol{U}_i\ \boldsymbol{u}(k-i) + \sum_{i=0}^{n_D}\boldsymbol{T}_i\ \boldsymbol{D}\boldsymbol{u}(k-i)$$

これを書き換えて

$$\boldsymbol{y}(k) = -\sum_{i=1}^{n_D}\boldsymbol{T}_i\boldsymbol{y}(k-i) + \sum_{i=1}^{n_D}\boldsymbol{U}_i\boldsymbol{u}(k-i) + \sum_{i=0}^{n_D}\boldsymbol{T}_i\boldsymbol{D}\boldsymbol{u}(k-i)$$

$$\left\{\boldsymbol{u}(k),\ k=0,1,\cdots\cdots,\ N\right\}と\left\{\boldsymbol{y}(k),\ k=-1,\ -2,\cdots\cdots,\ -n_D\right\}$$

を与えて$\left\{\boldsymbol{y}(k),\ k=0,1,\cdots\cdots,\ N\right\}$を計算する。

実 行 例

〔1〕 状態空間表現形式システムの任意入力に対する応答算出例

```
/SYS/ TIME-22:37:57 CPU-00:00:05
FILT/S
TO SOLVE DIFFERENCE EQUATION
SYSTEM NAME :   EX9
HOW MANY POINTS ? :    100
INPUT -MODE OF ( 1, 100) TIME SERIES INPUT      :  F
INPUT -FILENAME :    INPUT
GIVE INITIAL STATE X(0)
INPUT -MODE OF ( 1, 1) MATRIX X(0)      : Z
TYPE OR REVISE ? _
OUTPUT-MODE OF ( 1, 100) TIME SERIES OUTPUT     :  F
OUTPUT-FILENAME : _
/SYS/ TIME-22:42:46 CPU-00:00:12
```

〔2〕 微分（差分）方程式表現形式システムの任意入力に対する応答算出例

```
/SYS/ TIME-22:37:59 CPU-00:00:05
FILT/D
TO SOLVE DIFFERENCE EQUATION
SYSTEM NAME :   EX9D
HOW MANY POINTS ? :    100
INPUT -MODE OF ( 1, 100) TIME SERIES INPUT      :  F
INPUT -FILENAME :    INPUT
GIVE INITIAL INPUT U(0) AND OUTPUT Y(0)
INPUT -MODE OF ( 1, 1) MATRIX U(0)      : Z
TYPE OR REVISE ? _
INPUT -MODE OF ( 1, 1) MATRIX Y(0)      : Z
TYPE OR REVISE ? _
OUTPUT-MODE OF ( 1, 100) TIME SERIES OUTPUT     :  F
OUTPUT-FILENAME : _
/SYS/ TIME-22:42:47 CPU-00:00:12
```

3.9 ＴＳサブシステムの理論概要・処理機能・利用例

Ⅰ．　時系列データのハンドリング

Ⅱ．　時系列データの演算

Ⅲ．　時系列データの生成

Ⅳ．　時系列データの解析

時系列データの一覧出力	ＴＳ．Ｉ．１
ＬＩＳＴ	

機　　能

時系列データ管理台帳（ＴＳＬＩＳＴ）をリスティングし，必要が有れば指定する時系列データを削除する。

実 行 例

〔１〕　時系列データの一覧出力例

```
/TS/ TIME-22:38:45 CPU-00:00:06
LIST
TOTAL NO. OF INSTALLED TIME SERIES :     6
-------------------------------
        NAME       ( N,    M)
-------------------------------
  1    CML1        ( 1,2100)
  2    CML2        ( 1,2100)
  3    SMLA        ( 1, 102)
  4    SMLB        ( 1, 102)
  5    PIP         ( 2,2100)
  6    INPUT       ( 1, 100)
-------------------------------
HOW MANY TIME SERIES ARE DELETED ? : _
/TS/ TIME-22:44:26 CPU-00:00:15
```

〔2〕 時系列データの削除例

```
 /TS/ TIME-20:45:55 CPU-00:00:01
LIST
 TOTAL NO. OF INSTALLED TIME SERIES :     8
 ------------------------------------
           NAME       ( N,    M)
 ------------------------------------
   1    CML1          ( 1,2100)
   2    CML2          ( 1,2100)
   3    SMLA          ( 1, 102)
   4    SMLB          ( 1, 102)
   5    PIP           ( 2,2100)
   6    INPUT         ( 1, 100)
   7    A             ( 2,   30)
   8    B             ( 2,   30)
 ------------------------------------
 HOW MANY TIME SERIES ARE DELETED ? :   1
 SPECIFY  1-TIME SERIES TO BE DELETED :
7
 /TS/ TIME-20:46:05 CPU-00:00:08
LIST
 TOTAL NO. OF INSTALLED TIME SERIES :     7
 ------------------------------------
           NAME       ( N,    M)
 ------------------------------------
   1    CML1          ( 1,2100)
   2    CML2          ( 1,2100)
   3    SMLA          ( 1, 102)
   4    SMLB          ( 1, 102)
   5    PIP           ( 2,2100)
   6    INPUT         ( 1, 100)
   7    B             ( 2,   30)
 ------------------------------------
 HOW MANY TIME SERIES ARE DELETED ? : _
 /TS/ TIME-20:46:20 CPU-00:00:15
```

時系列データの入出力・複写	T S. I. 2
T S	

機　　能

時系列データの入力・出力・複写を行う。

実行例

〔1〕 時系列データの入力例

```
/TS/ TIME-22:38:48 CPU-00:00:06
TS
 TO INPUT TIME SERIES A(N,M)
 SPECIFY DIMENSION N AND M OF A :   2,30
 INPUT -MODE OF ( 2,   30) TIME SERIES   A        : _
 === INITIAL DATA ===
 INPUT -MODE OF ( 1, 2) MATRIX X(0)      : _
 ( 1) ROW
0,0
 TYPE OR REVISE ?
 === INTERVAL (    1-   30) ===
 INPUT -MODE OF (30, 2) MATRIX DATA      : _
 ( 1) ROW
21,51
 ( 2) ROW
22,52
 ( 3) ROW
23,53
 ( 4) ROW
24,54
 ( 5) ROW
25,55
 ( 6) ROW
26,56
 ( 7) ROW
27,57
 ( 8) ROW
28,58
 ( 9) ROW
29,59
 (10) ROW
30,60
 (11) ROW
31,61
 (12) ROW
32,62
 (13) ROW
33,63
 (14) ROW
34,64
 (15) ROW
35,65
 (16) ROW
36,66
 (17) ROW
37,67
 (18) ROW
38,68
 (19) ROW
39,69
 (20) ROW
40,70
 (21) ROW
```

```
41,71
 (22) ROW
42,72
 (23) ROW
43,73
 (24) ROW
44,74
 (25) ROW
45,75
 (26) ROW
46,76
 (27) ROW
47,77
 (28) ROW
48,78
 (29) ROW
49,79
 (30) ROW
50,80
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 2,   30) TIME SERIES XINDATA   :  F
 OUTPUT-FILENAME :    A
 /TS/ TIME-22:44:27 CPU-00:00:15
```

＜入力データの確認＞

```
 /TS/ TIME-22:38:49 CPU-00:00:06
TS
 TO INPUT TIME SERIES A(N,M)
 SPECIFY DIMENSION N AND M OF A :  2,30
 INPUT -MODE OF ( 2,  30) TIME SERIES  A      :  F
 INPUT -FILENAME :  A2
 OUTPUT-MODE OF ( 2,  30) TIME SERIES A2      :  _
 *** ( 2,  30) TIME SERIES A2        ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 2, 1) MATRIX X(0)      ---
 ( 1)  0.0
 ( 2)  0.0
 FILENAME :_
 === INTERVAL (   1-  30) ===
 --- ( 2,30) MATRIX DATA      ---
 ( 1)  1.00000000D+00  1.02000000D+02  3.00000000D+00  4.00000000D+00
 ( 2)  1.00000000D+02  1.01000000D+02  1.03000000D+02  1.04000000D+02
 ( 1)  5.00000000D+00  6.00000000D+00  7.00000000D+00  8.00000000D+00
 ( 2)  1.05000000D+02  1.06000000D+02  1.07000000D+02  1.08000000D+02
 ( 1)  9.00000000D+00  1.00000000D+01  1.10000000D+01  1.20000000D+01
 ( 2)  1.09000000D+02  1.10000000D+02  1.11000000D+02  1.12000000D+02
 ***
 ( 1)  1.30000000D+01  1.40000000D+01  1.50000000D+01  1.60000000D+01
 ( 2)  1.13000000D+02  1.14000000D+02  1.15000000D+02  1.16000000D+02
 ( 1)  1.70000000D+01  1.80000000D+01  1.90000000D+01  2.00000000D+01
 ( 2)  1.17000000D+02  1.18000000D+02  1.19000000D+02  1.20000000D+02
 ( 1)  2.10000000D+01  2.20000000D+01  2.30000000D+01  2.40000000D+01
 ( 2)  1.21000000D+02  1.22000000D+02  1.23000000D+02  1.24000000D+02
 ( 1)  2.50000000D+01  2.60000000D+01  2.70000000D+01  2.80000000D+01
 ( 2)  1.25000000D+02  1.26000000D+02  1.27000000D+02  1.28000000D+02
 ( 1)  2.90000000D+01  3.00000000D+01
 ( 2)  1.29000000D+02  1.30000000D+02
 FILENAME : _
 OUTPUT-FILENAME : _
 /TS/ TIME-22:44:28 CPU-00:00:15
```

〔2〕 時系列データの定数値入力例

```
/TS/ TIME-22:38:51 CPU-00:00:06
TS
 TO INPUT TIME SERIES A(N,M)
 SPECIFY DIMENSION N AND M OF A :  2,30
 INPUT -MODE OF ( 2,  30) TIME SERIES   A      : C
 CONSTANT VALUE =  1
 OUTPUT-MODE OF ( 2,  30) TIME SERIES XINDATA  : F
 OUTPUT-FILENAME :  B
 /TS/ TIME-22:44:30 CPU-00:00:15
```

＜入力データの確認＞

```
/TS/ TIME-22:38:52 CPU-00:00:06
TS
 TO INPUT TIME SERIES A(N,M)
 SPECIFY DIMENSION N AND M OF A :  2,30
 INPUT -MODE OF ( 2,  30) TIME SERIES   A      : F
 INPUT -FILENAME :  A
 OUTPUT-MODE OF ( 2,  30) TIME SERIES A        : _
 *** ( 2,  30) TIME SERIES A         ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 2, 1) MATRIX X(0)      ---
 ( 1)  1.00000000D+00
 ( 2)  1.00000000D+00
 FILENAME : _
 === INTERVAL (   1-  30) ===
 --- ( 2,30) MATRIX DATA      ---
 ( 1)  1.00000000D+00  1.00000000D+00  3.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ***
 ( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00
 FILENAME : _
 OUTPUT-FILENAME : _
 /TS/ TIME-22:44:31 CPU-00:00:15
```

〔3〕 時系列データの画面表示例

```
/TS/ TIME-22:38:53 CPU-00:00:06
TS
 TO INPUT TIME SERIES A(N,M)
 SPECIFY DIMENSION N AND M OF A :   2,30
 INPUT -MODE OF ( 2,   30) TIME SERIES   A          :  F
 INPUT -FILENAME :   A
 OUTPUT-MODE OF ( 2,   30) TIME SERIES A            :  _
 *** ( 2,   30) TIME SERIES A           ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 2, 1) MATRIX X(0)      ---
 ( 1)  0.0
 ( 2)  0.0
 FILENAME : _
 === INTERVAL (    1-   30) ===
 --- ( 2,30) MATRIX DATA      ---
 ( 1)  2.10000000D+01  2.20000000D+01  2.30000000D+01  2.40000000D+01
 ( 2)  5.10000000D+01  5.20000000D+01  5.30000000D+01  5.40000000D+01
 ( 1)  2.50000000D+01  2.60000000D+01  2.70000000D+01  2.80000000D+01
 ( 2)  5.50000000D+01  5.60000000D+01  5.70000000D+01  5.80000000D+01
 ( 1)  2.90000000D+01  3.00000000D+01  3.10000000D+01  3.20000000D+01
 ( 2)  5.90000000D+01  6.00000000D+01  6.10000000D+01  6.20000000D+01
 ***
 ( 1)  3.30000000D+01  3.40000000D+01  3.50000000D+01  3.60000000D+01
 ( 2)  6.30000000D+01  6.40000000D+01  6.50000000D+01  6.60000000D+01
 ( 1)  3.70000000D+01  3.80000000D+01  3.90000000D+01  4.00000000D+01
 ( 2)  6.70000000D+01  6.80000000D+01  6.90000000D+01  7.00000000D+01
 ( 1)  4.10000000D+01  4.20000000D+01  4.30000000D+01  4.40000000D+01
 ( 2)  7.10000000D+01  7.20000000D+01  7.30000000D+01  7.40000000D+01
 ( 1)  4.50000000D+01  4.60000000D+01  4.70000000D+01  4.80000000D+01
 ( 2)  7.50000000D+01  7.60000000D+01  7.70000000D+01  7.80000000D+01
 ( 1)  4.90000000D+01  5.00000000D+01
 ( 2)  7.90000000D+01  8.00000000D+01
 FILENAME : _
 OUTPUT-FILENAME : _
/TS/ TIME-22:44:32 CPU-00:00:15
```

時系列データの修正	TS. I. 3
REVISE	

機　能

時系列データの修正を行う。

実 行 例

〔1〕 時系列データの修正例

```
 /TS/ TIME-22:38:55 CPU-00:00:06
REVISE
 TO REVISE TIME SERIES A(N,M)
 SPECIFY (N,M) :   2,30
 INPUT -FILENAME :   A
 STARTING POINT ? (0-  30) :   3
 HOW MANY POINTS ? (1-28) :   1
 === INTERVAL (    3-    3) ===
 INPUT -MODE OF ( 1, 2) MATRIX DATA       : T
 --- ( 1, 2) MATRIX DATA        ---
 ( 1)   1.000000000D+00   1.000000000D+00
 TYPE OR REVISE ?   R
 SPECIFY (I,J) :   1,1
 DATA     ( 1, 1)=   1.000000000D+00   REVISE TO    3
 END ?
 TYPE OR REVISE ?    T
 --- ( 1, 2) MATRIX DATA        ---
 ( 1)   3.000000000D+00   1.000000000D+00
 TYPE OR REVISE ?
 /TS/ TIME-22:44:33 CPU-00:00:15
```

時系列データのグラフ出力	T S. I. 4
P L O T	

機 能

時系列データのグラフをＮＬＰまたは画面に出力する。(注)

◆座標軸の選択としては，以下の２通りがある。

①Ｘ軸24分割，Ｙ軸16分割の標準型

②Ｘ軸１～24分割，Ｙ軸１～16分割のユーザ指定型

（注）図形処理端末用システムと一般の端末用システムを起動（3.5.1 ＤＰＡＣＳ／Ｊの起動方法参照）した場合で出力先がそれぞれ画面（図形処理端末）とＮＬＰ（一般の端末）とに分かれる。ＮＬＰへの出力方法については，3.5.2 ＤＰＡＣＳ／Ｊの終了方法を参照のこと。

実行例

〔1〕 時系列データのグラフ出力例（X軸，Y軸標準型）

```
 /TS/ TIME-22:38:56 CPU-00:00:06
PLOT
 TO PLOT TIME SERIES
 SPECIFY (N,M,DELT) :   3,511,1
 INPUT -MODE OF ( 3, 511) TIME SERIES   A      : F
 INPUT -FILENAME :  FFTHAM
 STARTING POINT ? (0- 510) :  0
 HOW MANY POINT ? (1- 511) :  60
 MINIMUM AND MAXIMUM OF EACH VARIABLE
  Y( 1) :  -9.92004648D+00  5.60524862D+00
  Y( 2) :  -9.01239595D+00  3.05514770D+00
  Y( 3) :   1.50863266D-01  9.92004648D+00
 HOW MANY VARIABLES ARE PLOTTED ?  3
 SPECIFY MINIMUM AND MAXIMUM OF Y-AXIS
  YMIN =< -9.92004648D+00    YMAX => 9.92004648D+00
 GIVE RANGE OF Y-AXIS :  -10,10
 SPECIFY TYPE OF LINE ?
 1. SOLID  2. DOTTED-1  3. DOTTED-2  4. CHAIN ①
 GIVE TYPES OF  3-LINES :  1,2,3
 DRAW AXIS ?
 NORMAL TYPE ?
 NAME OF X-AXIS :   TIME (SEC)
 NAME OF Y-AXIS :
 TITLE :  HAMMING WINDOW FFT PLOT
 /TS/ TIME-22:44:35 CPU-00:00:15
```

【説明】

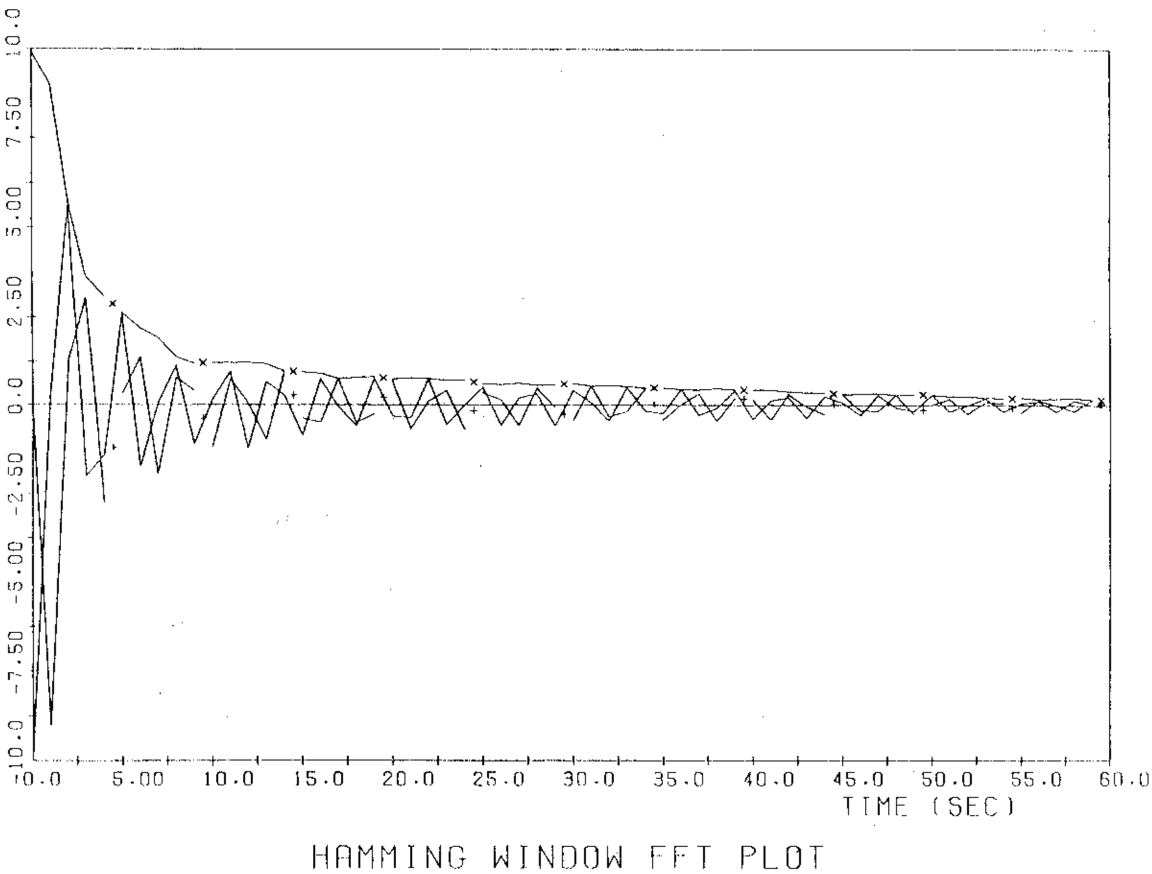
①線の種類／色を示すメッセージであり，ＤＰＡＣＳサブシステムのＳＭＬＴコマンド（ＤＰ．Ｖ．３）と同様である。

<図形処理端末用>

1. RED　2. GREEN　3. PURPLE　4. BLUE　5. PINK
6. WHITE

<一般の端末（ＮＬＰ出力）用>

1. SOLID　2. DOTTED−1　3. DOTTED−2　4. CHAIN



時系列データのグラフ出力例（X軸，Y軸標準型）

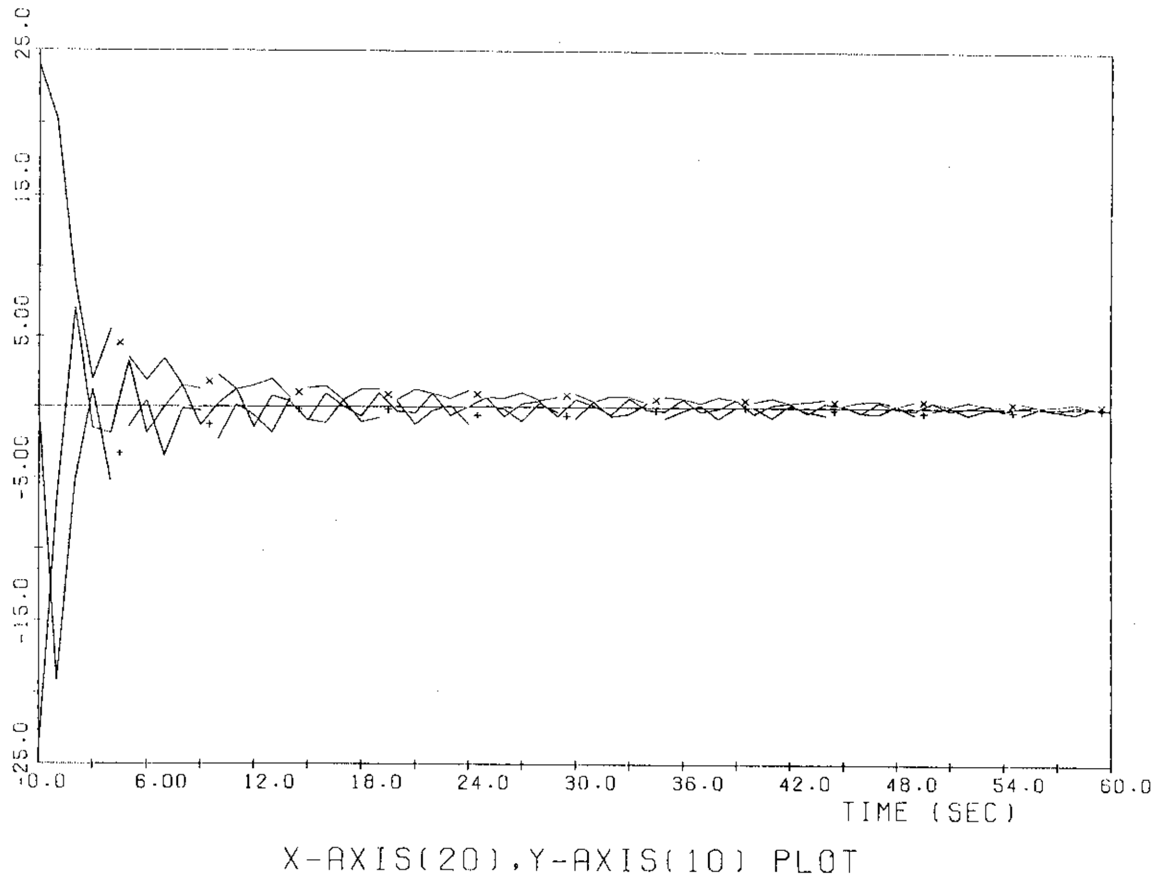
〔2〕 時系列データのグラフ出力例（X軸，Y軸の目盛り指定型）

```
/TS/ TIME-15:56:01 CPU-00:00:05
PLOT
 TO PLOT TIME SERIES
 SPECIFY (N,M,DELT) :   3,511,1
 INPUT -MODE OF ( 3, 511) TIME SERIES    A       :  F
 INPUT -FILENAME :  NOWIND
 STARTING POINT ? (0- 510) :   0
 HOW MANY POINT ? (1- 511) :   60
 MINIMUM AND MAXIMUM OF EACH VARIABLE
  Y( 1) :  -2.40953708D+01  7.08491970D+00
  Y( 2) :  -1.90837721D+01  1.25229432D+00
  Y( 3) :   8.47076701D-02  2.40953708D+01
 HOW MANY VARIABLES ARE PLOTTED ?   3
 SPECIFY MINIMUM AND MAXIMUM OF Y-AXIS
  YMIN =< -2.40953708D+01     YMAX => 2.40953708D+01
 GIVE RANGE OF Y-AXIS :  -25,25
 SPECIFY TYPE OF LINE ? _
 1. SOLID  2. DOTTED-1  3. DOTTED-2  4. CHAIN
 GIVE TYPES OF  3-LINES :  1,2,3
 DRAW AXIS ? _
 NORMAL TYPE ?   N ①
 NO. OF UNIT INTERVALS FOR X-AXIS (1-24) :   20
```

```
DRAW SCALE ? _
NAME OF X-AXIS :   TIME (SEC)
NO. OF UNIT INTERVALS FOR Y-AXIS (1-16) :   10
DRAW SCALE ? _
NAME OF Y-AXIS : _
TITLE :   X-AXIS(20),Y-AXIS(10) PLOT
DRAW FRAME ? _
DRAW ZERO LEVEL ? _
/TS/ TIME-15:57:29 CPU-00:00:08
```

【説明】

①ｘ軸，ｙ軸の目盛りの分割をユーザが指定した数で行う場合には，Ｙを入力するかリターンキーのみ押下する。Ｎを入力した場合には，自動的にｘ軸を24分割，ｙ軸を16分割し，外枠及びｙ＝０の線を引く。

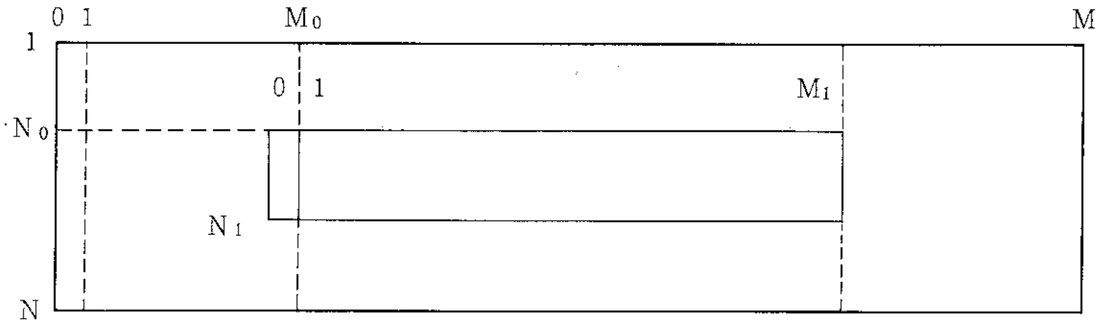


時系列データのグラフ出力例（Ｘ軸，Ｙ軸の目盛り指定型）

時系列データの一部抽出	T.S. I. 5
PICK	

機　能

時系列データを元の時系列データの一部から抽出する。



N_0, M_0：取り出す部分の最初の行と列

N_1, M_1：取り出す時系列データの大きさ

時系列データの一部抽出概念図

実 行 例

〔1〕　時系列データの一部抽出例

```
/TS/ TIME-22:38:57 CPU-00:00:06
PICK
TO PICK UP A@(N1,M1) FROM A(N,M)
SPECIFY (N,M) :   2,30
INPUT -MODE OF ( 2,  30) TIME SERIES   A       : F
INPUT -FILENAME :   A2
STARTING ROW AND COLUMN :   2,10
SPECIFY (N1,M1) :   1,10
OUTPUT-MODE OF ( 1,  10) TIME SERIES  A@       : _
*** ( 1,  10) TIME SERIES  A@        ***
ALL DATA ? _
=== INITIAL DATA ===
--- ( 1, 1) MATRIX X(0)       ---
( 1)   1.09000000D+02
FILENAME :_
=== INTERVAL (    1-   10) ===
--- ( 1,10) MATRIX DATA        ---
( 1)   1.10000000D+02  1.11000000D+02  1.12000000D+02  1.13000000D+02
( 1)   1.14000000D+02  1.15000000D+02  1.16000000D+02  1.17000000D+02
( 1)   1.18000000D+02  1.19000000D+02
FILENAME :_
/TS/ TIME-22:44:36 CPU-00:00:15
```

時系列データの分解	TS. I. 6
DCMP	

機　　能

時系列データの分解を行う。

◆行方向の分割：

$$\boldsymbol{A}(N,\ M) = \begin{bmatrix} \boldsymbol{A}_1(N_1,\ M) \\ \boldsymbol{A}_2(N_2,\ M) \end{bmatrix},\quad (N_1 + N_2 = N)$$

◆列方向の分割：

$$\boldsymbol{A}(N,\ M) = [\ \boldsymbol{A}_1(N,\ M_1),\ \boldsymbol{A}_2(N,\ M_2)\ ](M_1 + M_2 = M)$$

（注）$\boldsymbol{A}_2$の初期値は，$\boldsymbol{A}_1$の最後の列の値と同一値が入る。

実 行 例

〔1〕　時系列データの行方向の分割例

```
/TS/ TIME-22:38:59 CPU-00:00:06
DCMP
 TO DECOMPOSE A(N,M) INTO
       (1)   (A1(N1,M))
             (A2(N2,M))
       (2)   (A1(N,M1),A2(N,M2))
 WHICH DECOMPOSITION ? :    1
 SPECIFY (N,M,N1) :    2,30,1
 INPUT -MODE OF ( 2,   30) TIME SERIES   A        :  F
 INPUT -FILENAME :    A2
 OUTPUT-MODE OF ( 1,   30) TIME SERIES  A1        :  _
 *** ( 1,   30) TIME SERIES  A1        ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 1, 1) MATRIX X(0)       ---
 ( 1)  0.0
 FILENAME : __
 === INTERVAL (    1-   30) ===
 --- ( 1,30) MATRIX DATA       ---
 ( 1)  1.00000000D+00  1.02000000D+02  3.00000000D+00  4.00000000D+00
 ( 1)  5.00000000D+00  6.00000000D+00  7.00000000D+00  8.00000000D+00
 ( 1)  9.00000000D+00  1.00000000D+01  1.10000000D+01  1.20000000D+01
 ***
 ( 1)  1.30000000D+01  1.40000000D+01  1.50000000D+01  1.60000000D+01
 ( 1)  1.70000000D+01  1.80000000D+01  1.90000000D+01  2.00000000D+01
 ( 1)  2.10000000D+01  2.20000000D+01  2.30000000D+01  2.40000000D+01
 ( 1)  2.50000000D+01  2.60000000D+01  2.70000000D+01  2.80000000D+01
 ( 1)  2.90000000D+01  3.00000000D+01
 FILENAME : _
 OUTPUT-FILENAME : _
 OUTPUT-MODE OF ( 1,   30) TIME SERIES  A2        :  _
```

```
*** ( 1,  30) TIME SERIES  A2       ***
ALL DATA ? _
=== INITIAL DATA ===
--- ( 1, 1) MATRIX X(0)      ---
( 1)  0.0
FILENAME : _
=== INTERVAL (    1-  30) ===
--- ( 1,30) MATRIX DATA      ---
( 1)  1.00000000D+02  1.01000000D+02  1.03000000D+02  1.04000000D+02
( 1)  1.05000000D+02  1.06000000D+02  1.07000000D+02  1.08000000D+02
( 1)  1.09000000D+02  1.10000000D+02  1.11000000D+02  1.12000000D+02
( 1)  1.13000000D+02  1.14000000D+02  1.15000000D+02  1.16000000D+02
( 1)  1.17000000D+02  1.18000000D+02  1.19000000D+02  1.20000000D+02
( 1)  1.21000000D+02  1.22000000D+02  1.23000000D+02  1.24000000D+02
( 1)  1.25000000D+02  1.26000000D+02  1.27000000D+02  1.28000000D+02
***
( 1)  1.29000000D+02  1.30000000D+02
FILENAME : _
OUTPUT-FILENAME : _
/TS/ TIME-22:44:37 CPU-00:00:15
```

〔2〕 時系列データの列方向の分割例

```
/TS/ TIME-22:39:00 CPU-00:00:06
DCMP
 TO DECOMPOSE A(N,M) INTO
      (1)   (A1(N1,M))
            (A2(N2,M))
      (2)   (A1(N,M1),A2(N,M2))
 WHICH DECOMPOSITION ? :   2
 SPECIFY (N,M,M1) :    2,30,10
 INPUT -MODE OF ( 2,  30) TIME SERIES   A        : F
 INPUT -FILENAME :   A2
 OUTPUT-MODE OF ( 2,  10) TIME SERIES  A1        : _
 *** ( 2,  10) TIME SERIES  A1       ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 2, 1) MATRIX X(0)      ---
 ( 1)  0.0
 ( 2)  0.0
 FILENAME : _
 === INTERVAL (    1-  10) ===
 --- ( 2,10) MATRIX DATA      ---
 ( 1)  1.00000000D+00  1.02000000D+02  3.00000000D+00  4.00000000D+00
 ( 2)  1.00000000D+02  1.01000000D+02  1.03000000D+02  1.04000000D+02
 ( 1)  5.00000000D+00  6.00000000D+00  7.00000000D+00  8.00000000D+00
 ( 2)  1.05000000D+02  1.06000000D+02  1.07000000D+02  1.08000000D+02
 ( 1)  9.00000000D+00  1.00000000D+01
 ( 2)  1.09000000D+02  1.10000000D+02
 FILENAME : _
 OUTPUT-FILENAME : _
 OUTPUT-MODE OF ( 2,  20) TIME SERIES  A2        : _
 *** ( 2,  20) TIME SERIES  A2       ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 2, 1) MATRIX X(0)      ---
 ( 1)  1.00000000D+01
 ( 2)  1.10000000D+02
```

```
FILENAME : _
=== INTERVAL (    1-   20) ===
--- ( 2,20) MATRIX DATA       ---
( 1)  1.10000000D+01  1.20000000D+01  1.30000000D+01  1.40000000D+01
( 2)  1.11000000D+02  1.12000000D+02  1.13000000D+02  1.14000000D+02
( 1)  1.50000000D+01  1.60000000D+01  1.70000000D+01  1.80000000D+01
( 2)  1.15000000D+02  1.16000000D+02  1.17000000D+02  1.18000000D+02
( 1)  1.90000000D+01  2.00000000D+01  2.10000000D+01  2.20000000D+01
( 2)  1.19000000D+02  1.20000000D+02  1.21000000D+02  1.22000000D+02
( 1)  2.30000000D+01  2.40000000D+01  2.50000000D+01  2.60000000D+01
( 2)  1.23000000D+02  1.24000000D+02  1.25000000D+02  1.26000000D+02
( 1)  2.70000000D+01  2.80000000D+01  2.90000000D+01  3.00000000D+01
( 2)  1.27000000D+02  1.28000000D+02  1.29000000D+02  1.30000000D+02
FILENAME : _
OUTPUT-FILENAME : _
/TS/ TIME-22:44:38 CPU-00:00:15
```

時系列データの結合	TS. I. 7
APPEND	

機　能

時系列データの結合を行う。

◆行方向の結合：

$$\begin{bmatrix} \boldsymbol{A}_1(N_1,\ M) \\ \boldsymbol{A}_2(N_2,\ M) \end{bmatrix} = \boldsymbol{A}(N,\ M),\quad (N_1 + N_2 = N)$$

◆列方向の結合：

$$[\boldsymbol{A}_1(N,\ M_1),\ \boldsymbol{A}_2(N,\ M_2)] = \boldsymbol{A}(N,\ M),\quad (M_1 + M_2 = M)$$

（注）$\boldsymbol{A}_2$の初期値は，無視される。

実 行 例

〔1〕 時系列データの行方向の結合例

```
/TS/ TIME-22:39:02 CPU-00:00:06
APPEND
 TO APPEND A1 AND A2 AS
      (1)  A(N,M) = (A1(N1,M))
                    (A2(N2,M))
      (2)  A(N,M) = (A1(N,M1),A2(N,M2))
 WHICH COMBINATION ? :   1
 SPECIFY (N1,N2,M) :   2,2,30
 INPUT -MODE OF ( 2,  30) TIME SERIES  A1       :  F
 INPUT -FILENAME :   A
 INPUT -MODE OF ( 2,  30) TIME SERIES  A2       :  F
 INPUT -FILENAME :   A2
 OUTPUT-MODE OF ( 4,  30) TIME SERIES   A       :  _
 *** ( 4,  30) TIME SERIES   A      ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 4, 1) MATRIX X(0)      ---
 ( 1)  1.000000000D+00
 ( 2)  1.000000000D+00
 ( 3)  0.0
 ( 4)  0.0
 FILENAME : _
 === INTERVAL (   1-  30) ===
 --- ( 4,30) MATRIX DATA      ---
 ( 1)  1.00000000D+00  1.00000000D+00  3.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 3)  1.00000000D+00  1.02000000D+02  3.00000000D+00  4.00000000D+00
 ( 4)  1.00000000D+02  1.01000000D+02  1.03000000D+02  1.04000000D+02
 ( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 3)  5.00000000D+00  6.00000000D+00  7.00000000D+00  8.00000000D+00
 ( 4)  1.05000000D+02  1.06000000D+02  1.07000000D+02  1.08000000D+02
 ( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 3)  9.00000000D+00  1.00000000D+01  1.10000000D+01  1.20000000D+01
 ( 4)  1.09000000D+02  1.10000000D+02  1.11000000D+02  1.12000000D+02
 ( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 3)  1.30000000D+01  1.40000000D+01  1.50000000D+01  1.60000000D+01
 ( 4)  1.13000000D+02  1.14000000D+02  1.15000000D+02  1.16000000D+02
 ( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 3)  1.70000000D+01  1.80000000D+01  1.90000000D+01  2.00000000D+01
 ( 4)  1.17000000D+02  1.18000000D+02  1.19000000D+02  1.20000000D+02
***
 ( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 3)  2.10000000D+01  2.20000000D+01  2.30000000D+01  2.40000000D+01
 ( 4)  1.21000000D+02  1.22000000D+02  1.23000000D+02  1.24000000D+02
 ( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 3)  2.50000000D+01  2.60000000D+01  2.70000000D+01  2.80000000D+01
 ( 4)  1.25000000D+02  1.26000000D+02  1.27000000D+02  1.28000000D+02
 ( 1)  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00
 ( 3)  2.90000000D+01  3.00000000D+01
 ( 4)  1.29000000D+02  1.30000000D+02
FILENAME : _
OUTPUT-FILENAME : _
/TS/ TIME-22:44:39 CPU-00:00:15
```

〔2〕 時系列データの列方向の結合例

```
/TS/ TIME-22:39:12 CPU-00:00:06
APPEND
 TO APPEND A1 AND A2 AS
      (1)  A(N,M) = (A1(N1,M))
                    (A2(N2,M))
      (2)  A(N,M) = (A1(N,M1),A2(N,M2))
 WHICH COMBINATION ? :  2
 SPECIFY (N,M1,M2) :   2,30,30
 INPUT -MODE OF ( 2,  30) TIME SERIES  A1        :  F
 INPUT -FILENAME :   A
 INPUT -MODE OF ( 2,  30) TIME SERIES  A2        :  F
 INPUT -FILENAME :   A2
 OUTPUT-MODE OF ( 2,  60) TIME SERIES   A        :  _
 *** ( 2,  60) TIME SERIES   A        ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 2, 1) MATRIX X(0)      ---
 ( 1)   1.00000000D+00
 ( 2)   1.00000000D+00
 FILENAME : _
 === INTERVAL (    1-   50) ===
 --- ( 2,50) MATRIX DATA        ---
 ( 1)   1.00000000D+00  1.00000000D+00  3.00000000D+00  1.00000000D+00
 ( 2)   1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)   1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)   1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)   1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)   1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)   1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)   1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)   1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)   1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)   1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)   1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)   1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)   1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)   1.00000000D+00  1.00000000D+00  1.00000000D+00  1.02000000D+02
 ( 2)   1.00000000D+00  1.00000000D+00  1.00000000D+02  1.01000000D+02
 ( 1)   3.00000000D+00  4.00000000D+00  5.00000000D+00  6.00000000D+00
 ( 2)   1.03000000D+02  1.04000000D+02  1.05000000D+02  1.06000000D+02
 ( 1)   7.00000000D+00  8.00000000D+00  9.00000000D+00  1.00000000D+01
 ( 2)   1.07000000D+02  1.08000000D+02  1.09000000D+02  1.10000000D+02
 ***
 ( 1)   1.10000000D+01  1.20000000D+01  1.30000000D+01  1.40000000D+01
 ( 2)   1.11000000D+02  1.12000000D+02  1.13000000D+02  1.14000000D+02
 ( 1)   1.50000000D+01  1.60000000D+01  1.70000000D+01  1.80000000D+01
 ( 2)   1.15000000D+02  1.16000000D+02  1.17000000D+02  1.18000000D+02
 ( 1)   1.90000000D+01  2.00000000D+01
 ( 2)   1.19000000D+02  1.20000000D+02
 FILENAME : _
 === INTERVAL (   51-   60) ===
 --- ( 2,10) MATRIX DATA        ---
 ( 1)   2.10000000D+01  2.20000000D+01  2.30000000D+01  2.40000000D+01
 ( 2)   1.21000000D+02  1.22000000D+02  1.23000000D+02  1.24000000D+02
 ( 1)   2.50000000D+01  2.60000000D+01  2.70000000D+01  2.80000000D+01
 ( 2)   1.25000000D+02  1.26000000D+02  1.27000000D+02  1.28000000D+02
 ( 1)   2.90000000D+01  3.00000000D+01
 ( 2)   1.29000000D+02  1.30000000D+02
 FILENAME : _
 OUTPUT-FILENAME : _
 /TS/ TIME-22:45:02 CPU-00:00:16
```

時系列データの並べ替え	TS. I. 8
PRMT	

機　能

時系列データ：$\boldsymbol{A}$(N, M)の各行ベクトルを${}^t\boldsymbol{a}_1$, …, ${}^t\boldsymbol{a}_N$とし，この行ベクトの置換を{σ_i} i＝1, …, Nとし，以下の行列$\boldsymbol{A}$Sを計算する。

$$\boldsymbol{A}S = \begin{bmatrix} {}^t\boldsymbol{\sigma}_1 \\ \vdots \\ {}^t\boldsymbol{\sigma}_N \end{bmatrix}$$

実行例

〔1〕　時系列データの行の並べ替えの例

```
/TS/ TIME-22:39:14 CPU-00:00:06
PRMT
 TO PERMUTE ROWS OF A(N,M)
 SPECIFY (N,M) :   2,30
 INPUT -MODE OF ( 2,  30) TIME SERIES   A       : F
 INPUT -FILENAME :   A2
 SPECIFY  2-ROWS TO BE PERMUTED :   2,1
 OUTPUT-MODE OF ( 2,  30) TIME SERIES  A        : _
 *** ( 2,  30) TIME SERIES  A          ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 2, 1) MATRIX X(0)      ---
 ( 1)  0.0
 ( 2)  0.0
 FILENAME : _
 === INTERVAL (    1-  30) ===
 --- ( 2,30) MATRIX DATA      ---
 ( 1)  1.000000000D+02  1.010000000D+02  1.030000000D+02  1.040000000D+02
 ( 2)  1.000000000D+00  1.020000000D+02  3.000000000D+00  4.000000000D+00
 ( 1)  1.050000000D+02  1.060000000D+02  1.070000000D+02  1.080000000D+02
 ( 2)  5.000000000D+00  6.000000000D+00  7.000000000D+00  8.000000000D+00
 ( 1)  1.090000000D+02  1.100000000D+02  1.110000000D+02  1.120000000D+02
 ( 2)  9.000000000D+00  1.000000000D+01  1.100000000D+01  1.200000000D+01
 ( 1)  1.130000000D+02  1.140000000D+02  1.150000000D+02  1.160000000D+02
 ( 2)  1.300000000D+01  1.400000000D+01  1.500000000D+01  1.600000000D+01
 ( 1)  1.170000000D+02  1.180000000D+02  1.190000000D+02  1.200000000D+02
 ( 2)  1.700000000D+01  1.800000000D+01  1.900000000D+01  2.000000000D+01
 ( 1)  1.210000000D+02  1.220000000D+02  1.230000000D+02  1.240000000D+02
 ( 2)  2.100000000D+01  2.200000000D+01  2.300000000D+01  2.400000000D+01
 ( 1)  1.250000000D+02  1.260000000D+02  1.270000000D+02  1.280000000D+02
 ( 2)  2.500000000D+01  2.600000000D+01  2.700000000D+01  2.800000000D+01
 ( 1)  1.290000000D+02  1.300000000D+02
 ( 2)  2.900000000D+01  3.000000000D+01
 FILENAME : _
 OUTPUT-FILENAME : _
/TS/ TIME-22:45:03 CPU-00:00:16
```

時系列データのスキップ	TS. I. 9
SKIP	

機　能

時系列データ；$\boldsymbol{A}$(N, M)をk（≧1）列ずつ読み飛ばして行って$\boldsymbol{A}$!を得る。

$$\boldsymbol{A} = \left[{}^t\boldsymbol{a}_0,\ {}^t\boldsymbol{a}_1,\ \cdots,\ {}^t\boldsymbol{a}_M \right] \rightarrow \boldsymbol{A}! = \left[{}^t\boldsymbol{a}'_0,\ {}^t\boldsymbol{a}'_1,\ \cdots,\ {}^t\boldsymbol{a}'_{M'} \right]$$

$${}^t\boldsymbol{a}'_i = {}^t\boldsymbol{a}_{i\cdot(k+1)},\quad M' = [M/(k+1)]$$

実 行 例

〔1〕　時系列データのスキップ例

```
/TS/ TIME-22:39:15 CPU-00:00:06
SKIP
 TO MAKE SKIPPED TIME SERIES OF A(N,M)
 SPECIFY (N,M) :   2,30
 INPUT -MODE OF ( 2,  30) TIME SERIES   A       :  F
 INPUT -FILENAME :   A2
 NO. OF SKIPPED POINTS ? :    2
 OUTPUT-MODE OF ( 2,  10) TIME SERIES  A        :  _
 *** ( 2,  10) TIME SERIES  A        ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 2, 1) MATRIX X(0)      ---
 ( 1)  0.0
 ( 2)  0.0
 FILENAME : __
 === INTERVAL (    1-   10) ===
 --- ( 2,10) MATRIX DATA        ---
 ( 1)  3.00000000D+00  6.00000000D+00  9.00000000D+00  1.20000000D+01
 ( 2)  1.03000000D+02  1.06000000D+02  1.09000000D+02  1.12000000D+02
 ( 1)  1.50000000D+01  1.80000000D+01  2.10000000D+01  2.40000000D+01
 ( 2)  1.15000000D+02  1.18000000D+02  1.21000000D+02  1.24000000D+02
 ( 1)  2.70000000D+01  3.00000000D+01
 ( 2)  1.27000000D+02  1.30000000D+02
 FILENAME :_
 OUTPUT-FILENAME : _
/TS/ TIME-22:45:05 CPU-00:00:16
```

時系列データ同士の加算	T.S. II. 1
ADD	

機　能

時系列データ$\boldsymbol{A}$(N, M) = {a_{ij}}, $\boldsymbol{B}$(N, M) = {b_{ij}} に対して

$$\boldsymbol{A} + \boldsymbol{B} = \{ a_{ij} + b_{ij} \} \quad (i = 1, \cdots, N, j = 0, 1, \cdots, M)$$

を計算する。

実 行 例

〔1〕 時系列データ同士の加算例

```
/TS/ TIME-22:39:17 CPU-00:00:06
ADD
 TO CALCULATE A(N,M) + B(N,M)
 SPECIFY (N,M) :   2,30
 INPUT -MODE OF ( 2,  30) TIME SERIES   A      :  F
 INPUT -FILENAME :   A
 INPUT -MODE OF ( 2,  30) TIME SERIES   B      :  F
 INPUT -FILENAME :   A2
 OUTPUT-MODE OF ( 2,  30) TIME SERIES  A+B     :  _
 *** ( 2,  30) TIME SERIES  A+B     ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 2, 1) MATRIX X(0)      ---
 ( 1)  1.000000000D+00
 ( 2)  1.000000000D+00
 FILENAME : _
 === INTERVAL (    1-   30) ===
 --- ( 2,30) MATRIX DATA       ---
 ( 1)  2.000000000D+00  1.030000000D+02  6.000000000D+00  5.000000000D+00
 ( 2)  1.010000000D+02  1.020000000D+02  1.040000000D+02  1.050000000D+02
 ( 1)  6.000000000D+00  7.000000000D+00  8.000000000D+00  9.000000000D+00
 ( 2)  1.060000000D+02  1.070000000D+02  1.080000000D+02  1.090000000D+02
 ( 1)  1.000000000D+01  1.100000000D+01  1.200000000D+01  1.300000000D+01
 ( 2)  1.100000000D+02  1.110000000D+02  1.120000000D+02  1.130000000D+02
 ( 1)  1.400000000D+01  1.500000000D+01  1.600000000D+01  1.700000000D+01
 ( 2)  1.140000000D+02  1.150000000D+02  1.160000000D+02  1.170000000D+02
 ( 1)  1.800000000D+01  1.900000000D+01  2.000000000D+01  2.100000000D+01
 ( 2)  1.180000000D+02  1.190000000D+02  1.200000000D+02  1.210000000D+02
 ( 1)  2.200000000D+01  2.300000000D+01  2.400000000D+01  2.500000000D+01
 ( 2)  1.220000000D+02  1.230000000D+02  1.240000000D+02  1.250000000D+02
 ( 1)  2.600000000D+01  2.700000000D+01  2.800000000D+01  2.900000000D+01
 ( 2)  1.260000000D+02  1.270000000D+02  1.280000000D+02  1.290000000D+02
 ( 1)  3.000000000D+01  3.100000000D+01
 ( 2)  1.300000000D+02  1.310000000D+02
 FILENAME : _
 OUTPUT-FILENAME : _
/TS/ TIME-22:45:06 CPU-00:00:16
```

時系列データへの列ベクトルの加算	TS. II. 2
SADD	

機　能

時系列データ $\boldsymbol{A}$(N, M) = {a_{ij}}, オフセットベクトル $\boldsymbol{s}$(N) = {s_i} に対して

$$\boldsymbol{A} + \boldsymbol{s} = \{a_{ij} + s_i\} \quad (i = 1, \cdots, N, \ j = 0, 1, \cdots, M)$$

を計算する。

実 行 例

〔1〕 時系列データへの列ベクトルの加算例

```
 /TS/ TIME-22:39:18 CPU-00:00:06
SADD
 TO CALCULATE SCALAR(N) + A(N,M)
 SPECIFY (N,M) :   2,2100
 INPUT -MODE OF ( 2, 1) MATRIX SCALAR   :  
 ( 1) ROW
2
 ( 2) ROW
2
 TYPE OR REVISE ?
 INPUT -MODE OF ( 2,2100) TIME SERIES   A      :  F
 INPUT -FILENAME :   SQUX
 OUTPUT-MODE OF ( 2,2100) TIME SERIES  S+A     :  F
 OUTPUT-FILENAME :   SQUD
 /TS/ TIME-22:45:07 CPU-00:00:16
```

時系列データ同士の減算	TS. II. 3
SUB	

機　能

時系列データ $\boldsymbol{A}(N, M)=\{a_{ij}\}$, $\boldsymbol{B}(N, M)=\{b_{ij}\}$ に対して

$$\boldsymbol{A}-\boldsymbol{B}=\{a_{ij}-b_{ij}\} \quad (i=1, \cdots, N, \ j=0, 1, \cdots, M)$$

を計算する。

実 行 例

〔1〕　時系列データ同士の減算例

```
/TS/ TIME-22:39:19 CPU-00:00:06
SUB
 TO CALCULATE A(N,M) - B(N,M)
 SPECIFY (N,M) :   2,30
 INPUT -MODE OF ( 2,  30) TIME SERIES   A      :  F
 INPUT -FILENAME :   A
 INPUT -MODE OF ( 2,  30) TIME SERIES   B      :  F
 INPUT -FILENAME :   A2
 OUTPUT-MODE OF ( 2,  30) TIME SERIES  A-B     :
 *** ( 2,  30) TIME SERIES  A-B      ***
 ALL DATA ?
 === INITIAL DATA ===
 --- ( 2, 1) MATRIX X(0)        ---
 ( 1)   1.00000000D+00
 ( 2)   1.00000000D+00
 FILENAME :
 === INTERVAL (    1-   30) ===
 --- ( 2,30) MATRIX DATA       ---
 ( 1)  0.0              -1.01000000D+02  0.0              -3.00000000D+00
 ( 2) -9.90000000D+01 -1.00000000D+02 -1.02000000D+02 -1.03000000D+02
 ( 1) -4.00000000D+00 -5.00000000D+00 -6.00000000D+00 -7.00000000D+00
 ( 2) -1.04000000D+02 -1.05000000D+02 -1.06000000D+02 -1.07000000D+02
 ( 1) -8.00000000D+00 -9.00000000D+00 -1.00000000D+01 -1.10000000D+01
 ( 2) -1.08000000D+02 -1.09000000D+02 -1.10000000D+02 -1.11000000D+02
 ( 1) -1.20000000D+01 -1.30000000D+01 -1.40000000D+01 -1.50000000D+01
 ( 2) -1.12000000D+02 -1.13000000D+02 -1.14000000D+02 -1.15000000D+02
 ( 1) -1.60000000D+01 -1.70000000D+01 -1.80000000D+01 -1.90000000D+01
 ( 2) -1.16000000D+02 -1.17000000D+02 -1.18000000D+02 -1.19000000D+02
 ( 1) -2.00000000D+01 -2.10000000D+01 -2.20000000D+01 -2.30000000D+01
 ( 2) -1.20000000D+02 -1.21000000D+02 -1.22000000D+02 -1.23000000D+02
 ( 1) -2.40000000D+01 -2.50000000D+01 -2.60000000D+01 -2.70000000D+01
 ( 2) -1.24000000D+02 -1.25000000D+02 -1.26000000D+02 -1.27000000D+02
 ( 1) -2.80000000D+01 -2.90000000D+01
 ( 2) -1.28000000D+02 -1.29000000D+02
 FILENAME :
 OUTPUT-FILENAME :
/TS/ TIME-22:45:09 CPU-00:00:16
```

時系列データからの列ベクトルの減算	TS. II. 4
SSUB	

機　能

時系列データ $\boldsymbol{A}$(N, M) = {a_{ij}}, オフセットベクトル $\boldsymbol{s}$(N) = {s_i} に対して

$$\boldsymbol{A} - \boldsymbol{s} = \{a_{ij} - s_i\} \quad (i = 1, \cdots, N, \ j = 0, 1, \cdots, M)$$

を計算する。

実 行 例

〔1〕 時系列データからの列ベクトルの減算例

```
 /TS/ TIME-22:39:20 CPU-00:00:06
SSUB
 TO CALCULATE A(N,M) - SCALAR(N)
 SPECIFY (N,M) :   2,2100
 INPUT -MODE OF ( 2, 1) MATRIX SCALAR   : _
 ( 1) ROW
1
 ( 2) ROW
3
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2,2100) TIME SERIES   A      : F
 INPUT -FILENAME :   SINX
 OUTPUT-MODE OF ( 2,2100) TIME SERIES  A-S     : F
 OUTPUT-FILENAME :   SINM
 /TS/ TIME-22:45:10 CPU-00:00:16
```

時系列データ同士の内積および行列との乗算	TS. II. 5
MUL	

機　能

〔1〕 2つの時系列データの内積を求める。

時系列データ $\boldsymbol{A}$(N, M) = {a_{ij}}, $\boldsymbol{B}$(L, M) = {b_{ij}} に対して

$$\boldsymbol{A}\cdot\boldsymbol{B}=\{\sum_k a_{ik}\,b_{kj}\}\qquad(i=1,\cdots,N,\ j=0,1,\cdots,M)$$

を計算する。

〔2〕 時系列データと行列との積を求める。

行列データ $\boldsymbol{A}$(N, M) = {a_{ij}}, 時系列データ $\boldsymbol{B}$(L, M) = {b_{ij}} に対して

$$\boldsymbol{A}\cdot\boldsymbol{B}=\{\sum_k a_{ik}\,b_{kj}\}\qquad(i=1,\cdots,N,\ j=0,1,\cdots,M)$$

を計算する。

実 行 例

〔1〕 時系列データと行列データとの積を算出する例

```
/TS/ TIME-22:39:21 CPU-00:00:06
MUL
 TO CALCULATE
      (1)  MAT(N,M) * A(M,L)
      (2)  A(N,M) * B'(L,M)
 WHICH CALCULATION ? :   1
 SPECIFY (N,M,L) :   3,2,30
 INPUT -MODE OF ( 3, 2) MATRIX  MAT      : _
 ( 1) ROW
1,0
 ( 2) ROW
0,1
 ( 3) ROW
1,-1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2,  30) TIME SERIES   A       :  F
 INPUT -FILENAME :   A2
 OUTPUT-MODE OF ( 3,  30) TIME SERIES MAT*A     :  _
 *** ( 3,  30) TIME SERIES MAT*A    ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 3, 1) MATRIX X(0)       ---
 ( 1)  0.0
 ( 2)  0.0
 ( 3)  0.0
 FILENAME : _
 === INTERVAL (    1-  30) ===
 --- ( 3,30) MATRIX DATA       ---
```

```
( 1)   1.00000000D+00  1.02000000D+02  3.00000000D+00  4.00000000D+00
( 2)   1.00000000D+02  1.01000000D+02  1.03000000D+02  1.04000000D+02
( 3)  -9.90000000D+01  1.00000000D+00 -1.00000000D+02 -1.00000000D+02
( 1)   5.00000000D+00  6.00000000D+00  7.00000000D+00  8.00000000D+00
( 2)   1.05000000D+02  1.06000000D+02  1.07000000D+02  1.08000000D+02
( 3)  -1.00000000D+02 -1.00000000D+02 -1.00000000D+02 -1.00000000D+02
( 1)   9.00000000D+00  1.00000000D+01  1.10000000D+01  1.20000000D+01
( 2)   1.09000000D+02  1.10000000D+02  1.11000000D+02  1.12000000D+02
( 3)  -1.00000000D+02 -1.00000000D+02 -1.00000000D+02 -1.00000000D+02
( 1)   1.30000000D+01  1.40000000D+01  1.50000000D+01  1.60000000D+01
( 2)   1.13000000D+02  1.14000000D+02  1.15000000D+02  1.16000000D+02
( 3)  -1.00000000D+02 -1.00000000D+02 -1.00000000D+02 -1.00000000D+02
( 1)   1.70000000D+01  1.80000000D+01  1.90000000D+01  2.00000000D+01
( 2)   1.17000000D+02  1.18000000D+02  1.19000000D+02  1.20000000D+02
( 3)  -1.00000000D+02 -1.00000000D+02 -1.00000000D+02 -1.00000000D+02
( 1)   2.10000000D+01  2.20000000D+01  2.30000000D+01  2.40000000D+01
( 2)   1.21000000D+02  1.22000000D+02  1.23000000D+02  1.24000000D+02
( 3)  -1.00000000D+02 -1.00000000D+02 -1.00000000D+02 -1.00000000D+02
( 1)   2.50000000D+01  2.60000000D+01  2.70000000D+01  2.80000000D+01
( 2)   1.25000000D+02  1.26000000D+02  1.27000000D+02  1.28000000D+02
( 3)  -1.00000000D+02 -1.00000000D+02 -1.00000000D+02 -1.00000000D+02
***
( 1)   2.90000000D+01  3.00000000D+01
( 2)   1.29000000D+02  1.30000000D+02
( 3)  -1.00000000D+02 -1.00000000D+02
FILENAME :_
OUTPUT-FILENAME : _
/TS/ TIME-22:45:12 CPU-00:00:16
```

〔2〕 時系列データ同士の内積を求める例

```
 /TS/ TIME-22:39:23 CPU-00:00:06
MUL
 TO CALCULATE
       (1)  MAT(N,M) * A(M,L)
       (2)  A(N,M) * B'(L,M)
 WHICH CALCULATION ? :   2
 SPECIFY (N,M,L) :    2,30,2
 INPUT -MODE OF ( 2,  30) TIME SERIES   A      :  F
 INPUT -FILENAME :   A
 INPUT -MODE OF ( 2,  30) TIME SERIES   B      :  F
 INPUT -FILENAME :   A2
 OUTPUT-MODE OF ( 2, 2) MATRIX A*B'        :  _
 --- ( 2, 2) MATRIX A*B'     ---
 ( 1)   5.71000000D+02  3.66900000D+03
 ( 2)   5.65000000D+02  3.46300000D+03
 FILENAME : _
 /TS/ TIME-22:45:11 CPU-00:00:16
```

時系列データ同士の乗算	TS. II. 6
TSMUL	

機能

〔1〕 時系列データ$\boldsymbol{A}$, $\boldsymbol{B}$が同一の次元を持つ場合の2つの時系列データの積を求める。
時系列データ$\boldsymbol{A}(N, M) = \{a_{ij}\}$, $\boldsymbol{B}(N, M) = \{b_{ij}\}$に対して

$$\{a_{ij} b_{ij}\} \quad (i = 1, \cdots, N,\ j = 0, 1, \cdots, M)$$

を計算する。

〔2〕 時系列データ$\boldsymbol{B}$が(1, M)の次元を持つ場合の2つの時系列データの積を求める。
時系列データ$\boldsymbol{A}(N, M) = \{a_{ij}\}$, $\boldsymbol{B}(1, M) = \{b_j\}$に対して

$$\{a_{ij} b_j\} \quad (i = 1, \cdots, N,\ j = 0, 1, \cdots, M)$$

を計算する。

実行例

〔1〕 時系列データ同士の乗算例(1)

```
/TS/ TIME-22:39:25 CPU-00:00:07
TSMUL
 TO CALCULATE
    (1) A(N,M) * B(N,M)
    (2) A(N,M) * B(1,M)
    WHICH TYPE ? 1
 SPECIFY (N,M) :   2,2100
 INPUT -MODE OF ( 2,2100) TIME SERIES   A       :  F
 INPUT -FILENAME :   SINX
 INPUT -MODE OF ( 2,2100) TIME SERIES   B       :  F
 INPUT -FILENAME :   SQUX
 OUTPUT-MODE OF ( 2,2100) TIME SERIES   A*B     :  F
 OUTPUT-FILENAME :   SINSQU
 /TS/ TIME-22:45:15 CPU-00:00:16
```

〔2〕 時系列データ同士の乗算例（2）

```
 /TS/ TIME-22:39:24 CPU-00:00:06
TSMUL
 TO CALCULATE
    (1) A(N,M) * B(N,M)
    (2) A(N,M) * B(1,M)
    WHICH TYPE ? 2
 SPECIFY (N,M) :   2,2100
 INPUT -MODE OF ( 2,2100) TIME SERIES   A      :  F
 INPUT -FILENAME :   SINX
 INPUT -MODE OF ( 1,2100) TIME SERIES   B      :  F
 INPUT -FILENAME :   TRI
 OUTPUT-MODE OF ( 2,2100) TIME SERIES  A*B     :  F
 OUTPUT-FILENAME :   TSMUL2
 /TS/ TIME-22:45:14 CPU-00:00:16
```

〔3〕 時系列データ同士の乗算例（定数値入力）

```
 /TS/ TIME-14:00:01 CPU-00:00:01
TSMUL
 TO CALCULATE
    (1) A(N,M) * B(N,M)
    (2) A(N,M) * B(1,M)
    WHICH TYPE ? 2
 SPECIFY (N,M) :   2,2100
 INPUT -MODE OF ( 2,2100) TIME SERIES   A      :  C
 CONSTANT VALUE =   -4
 INPUT -MODE OF ( 1,2100) TIME SERIES   B      :  F
 INPUT -FILENAME :   SIN
 OUTPUT-MODE OF ( 2,2100) TIME SERIES  A*B     :  F
 OUTPUT-FILENAME :   SIN4
 /TS/ TIME-14:00:15 CPU-00:00:14
```

時系列データと列ベクトルとの乗算	TS. II. 7
SMUL	

機　能

時系列データ $\boldsymbol{A}$(N, M) = {a_{ij}}, オフセットベクトル $\boldsymbol{s}$(N) = {s_i} に対して

$$\boldsymbol{A} \times \boldsymbol{s} = \{a_{ij} \times s_i\} \quad (i = 1, \cdots, N,\ j = 0, 1, \cdots, M)$$

を計算する。

実 行 例

〔1〕　時系列データと列ベクトルの乗算例

```
 /TS/ TIME-14:00:28 CPU-00:00:01
SMUL
 TO CALCULATE SCALAR(N) * A(N,M)
 SPECIFY (N,M) :   2,2100
 INPUT -MODE OF ( 2, 1) MATRIX  SCALAR  : _
 ( 1) ROW
2
 ( 2) ROW
3
 TYPE OR REVISE ?_
 INPUT -MODE OF ( 2,2100) TIME SERIES   A       :  F
 INPUT -FILENAME :   SINX
 OUTPUT-MODE OF ( 2,2100) TIME SERIES  S*A      :  F
 OUTPUT-FILENAME :   SINES1
 /TS/ TIME-14:00:39 CPU-00:00:13
```

時系列データ同士の除算	TS. II. 8
TSDIV	

機　能

〔1〕 時系列データ$\boldsymbol{A}$，$\boldsymbol{B}$が同一の次元を持つ場合の2つの時系列データの商を以下の様に求める。

時系列データ$\boldsymbol{A}(N, M) = \{a_{ij}\}$，$\boldsymbol{B}(N, M) = \{b_{ij}\}$に対して

$$\boldsymbol{A} / \boldsymbol{B} = \{a_{ij} / b_{ij}\} \quad (i = 1, \cdots, N,\ j = 0, 1, \cdots, M)$$

を計算する。但し，$b_{ij} \neq 0$

〔2〕 時系列データ$\boldsymbol{B}$が$(1, M)$の次元を持つ場合の2つの時系列データの商を以下の様に求める。

時系列データ$\boldsymbol{A}(N, M) = \{a_{ij}\}$，$\boldsymbol{B}(1, M) = \{b_j\}$に対して

$$\boldsymbol{A} / \boldsymbol{B} = \{a_{ij} / b_j\} \quad (i = 1, \cdots, N,\ j = 0, 1, \cdots, M)$$

を計算する。但し，$b_j \neq 0$

実 行 例

〔1〕 時系列データ同士の除算例（1）

```
 /TS/ TIME-14:00:54 CPU-00:00:01
TSDIV
 TO CALCULATE
    (1) A(N,M) / B(N,M)
    (2) A(N,M) / B(1,M)
    WHICH TYPE ? 1
 SPECIFY (N,M) :   2,2100
 INPUT -MODE OF ( 2,2100) TIME SERIES   A       :  F
 INPUT -FILENAME :    SINX
 INPUT -MODE OF ( 2,2100) TIME SERIES   B       :  F
 INPUT -FILENAME :    SQUD
 OUTPUT-MODE OF ( 2,2100) TIME SERIES  A/B      :  F
 OUTPUT-FILENAME :    TSDIV
 /TS/ TIME-14:01:10 CPU-00:00:15
```

〔2〕 時系列データ同士の除算例（2）

```
 /TS/ TIME-14:01:28 CPU-00:00:01
TSDIV
 TO CALCULATE
    (1) A(N,M) / B(N,M)
    (2) A(N,M) / B(1,M)
    WHICH TYPE ? 2
 SPECIFY (N,M) :   2,2100
 INPUT -MODE OF ( 2,2100) TIME SERIES   A        :  F
 INPUT -FILENAME :   SINX
 INPUT -MODE OF ( 1,2100) TIME SERIES   B        :  F
 INPUT -FILENAME :   TRID
 OUTPUT-MODE OF ( 2,2100) TIME SERIES  A/B       :  F
 OUTPUT-FILENAME :   TSDIV2
 /TS/ TIME-14:01:43 CPU-00:00:20
```

〔3〕 時系列データ同士の除算例（3）零割発生の例

```
 /TS/ TIME-15:29:07 CPU-00:00:01
TSDIV
 TO CALCULATE
    (1) A(N,M) / B(N,M)
    (2) A(N,M) / B(1,M)
    WHICH TYPE ? 1
 SPECIFY (N,M) :   1,2000
 INPUT -MODE OF ( 1,2000) TIME SERIES   A        :  F
 INPUT -FILENAME :   SIN
 INPUT -MODE OF ( 1,2000) TIME SERIES   B        :  F
 INPUT -FILENAME :   SQU                               ①
 @@ ZERO DIVIDE OCCURS AT ROW =     1  COLUMN =      0
 ERROR IN XDIVF  (K039  ) :  209
 ERROR RETURN AT  1 IN TS39F  (X039  )
 /TS/ TIME-15:30:59 CPU-00:00:06
```

【説明】

①零割の発生した行と列を示す。尚，列の値は初期値ベクトルの場合0としている。

時系列データの列ベクトルによる除算	TS. II. 9
SDIV	

機　能

時系列データの行毎に一定値で除算する。

時系列データ $\boldsymbol{A}(N, M)=\{a_{ij}\}$, $\boldsymbol{s}(N)=\{s_i\}$ に対して

$$\boldsymbol{A}/\boldsymbol{s}=\{a_{ij}/s_i\} \quad (i=1, \cdots, N, \ j=0, 1, \cdots, M)$$

を計算する。但し，$s_i \neq 0$

実 行 例

〔1〕　時系列データと列ベクトルの除算例

```
 /TS/ TIME-14:02:07 CPU-00:00:01
SDIV
 TO CALCULATE A(N,M) / SCALAR(N)
 SPECIFY (N,M) :   2,2100
 INPUT -MODE OF ( 2, 1) MATRIX  SCALAR  : _
 ( 1) ROW
2
 ( 2) ROW
3
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2,2100) TIME SERIES   A      :  F
 INPUT -FILENAME :   SINX
 OUTPUT-MODE OF ( 2,2100) TIME SERIES  A/S     :  F
 OUTPUT-FILENAME :   SINXS1
 /TS/ TIME-14:02:15 CPU-00:00:08
```

時系列データの差分値算出	TS. Ⅱ. 10
DIF	

機　能

サンプル周期Tの時系列データ$\boldsymbol{A}(N, M) = \{a_{ij}\}$に対して

$$\{(a_{ij+1} - a_{ij}) / T\} \quad (i = 1, \cdots, N, \ j = 0, 1, \cdots, M-1)$$

を計算する。

実 行 例

〔1〕　時系列データの差分時系列データを計算する例

```
/TS/ TIME-14:02:29 CPU-00:00:01
DIF
 TO MAKE DIFFERENTIAL TIME SERIES OF A(N,M)
 SPECIFY (N,M,DELT) :   2,30,1
 INPUT -MODE OF ( 2,  30) TIME SERIES   A       :  F
 INPUT -FILENAME :   A2
 ORDER OF DIFFERENTIATION ? :   1
 OUTPUT-MODE OF ( 2,  29) TIME SERIES DIF(A)    :  _
 *** ( 2,  29) TIME SERIES DIF(A)    ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 2, 1) MATRIX X(0)      ---
 ( 1)  1.00000000D+00
 ( 2)  1.00000000D+02
 FILENAME : _
 === INTERVAL (   1-  29) ===
 --- ( 2,29) MATRIX DATA      ---
 ( 1)  1.01000000D+02 -9.90000000D+01  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  2.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)  1.00000000D+00
 ( 2)  1.00000000D+00
 FILENAME : _
 OUTPUT-FILENAME : _
 /TS/ TIME-14:02:40 CPU-00:00:06
```

時系列データの積分値算出	TS. II. 11
INT	

機　　能

サンプル周期Tの時系列データ $\boldsymbol{A}(N, M) = \{a_{ij}\}$ に対して，
初期値 $\boldsymbol{x}(0) = \{x_{i0}\}(i=1, \cdots, N)$ を与えて

$$\{(x_{ij-1} + T \cdot a_{ij-1}\}\ (i=1, \cdots, N,\ j=1, \cdots, M+1)$$

を計算する。

実 行 例

〔1〕　時系列データの積分計算時系列データを算出する例

```
/TS/ TIME-14:02:57 CPU-00:00:01
INT
 TO INTEGRATE A(N,M)
 SPECIFY (N,M,DELT) :   2,30,1
 INPUT -MODE OF ( 2,   30) TIME SERIES   A       :  F
 INPUT -FILENAME :   A2
 SPECIFY INITIAL DATA
 INPUT -MODE OF ( 1, 2) MATRIX X(0)       : Z
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 2,   31) TIME SERIES INT(A)    :  _
 *** ( 2,  31) TIME SERIES INT(A)    ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 2, 1) MATRIX X(0)       ---
 ( 1)  0.0
 ( 2)  0.0
 FILENAME : _
 === INTERVAL (    1-   31) ===
 --- ( 2,31) MATRIX DATA       ---
 ( 1)  0.0               1.000000000D+00  1.030000000D+02  1.060000000D+02
 ( 2)  0.0               1.000000000D+02  2.010000000D+02  3.040000000D+02
 ( 1)  1.100000000D+02   1.150000000D+02  1.210000000D+02  1.280000000D+02
 ( 2)  4.080000000D+02   5.130000000D+02  6.190000000D+02  7.260000000D+02
 ( 1)  1.360000000D+02   1.450000000D+02  1.550000000D+02  1.660000000D+02
 ( 2)  8.340000000D+02   9.430000000D+02  1.053000000D+03  1.164000000D+03
 ( 1)  1.780000000D+02   1.910000000D+02  2.050000000D+02  2.200000000D+02
 ( 2)  1.276000000D+03   1.389000000D+03  1.503000000D+03  1.618000000D+03
 ( 1)  2.360000000D+02   2.530000000D+02  2.710000000D+02  2.900000000D+02
 ( 2)  1.734000000D+03   1.851000000D+03  1.969000000D+03  2.088000000D+03
 ( 1)  3.100000000D+02   3.310000000D+02  3.530000000D+02  3.760000000D+02
 ( 2)  2.208000000D+03   2.329000000D+03  2.451000000D+03  2.574000000D+03
 ( 1)  4.000000000D+02   4.250000000D+02  4.510000000D+02  4.780000000D+02
 ( 2)  2.698000000D+03   2.823000000D+03  2.949000000D+03  3.076000000D+03
 ( 1)  5.060000000D+02   5.350000000D+02  5.650000000D+02
 ( 2)  3.204000000D+03   3.333000000D+03  3.463000000D+03
 FILENAME : _
 OUTPUT-FILENAME : _
/TS/ TIME-14:03:11 CPU-00:00:09
```

時系列データの自然対数の算出	T S. Ⅱ. 1 2
L O G	

機　能

時系列データ $\boldsymbol{A}(N, M)=\{a_{ij}\}$ に対して,

$$\{\log_e(a_{ij}+m_i)\} \quad (i=1, \cdots, N,\ j=0, \cdots, M)$$

を計算する。但し, $m_i > -\{\min_j(a_{ij})\}$

実 行 例

〔1〕 時系列データの自然対数の算出例

```
 /TS/ TIME-14:03:24 CPU-00:00:01
LOG
 TO MAKE LOGARITHMIC TIME SERIES OF A(N,M)
      ( LOG(A)(I,J) = LN( A(I,J) + M(I) ) )
 SPECIFY (N,M) :   2,30
 INPUT -MODE OF ( 2,  30) TIME SERIES   A       :  F
 INPUT -FILENAME :   A2
 MINIMUM OF VARIABLES
 --- ( 1, 2) MATRIX MINI        ---
 ( 1)  0.0               0.0
 INPUT -MODE OF ( 1, 2) MATRIX   M        :  _
 ( 1) ROW
1,1
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 2,  30) TIME SERIES LOG(A)    : _
 *** ( 2,  30) TIME SERIES LOG(A)     ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 2, 1) MATRIX X(0)        ---
 ( 1)  0.0
 ( 2)  0.0
 FILENAME : _
 === INTERVAL (    1-   30) ===
 --- ( 2,30) MATRIX DATA        ---
 ( 1)  6.93147181D-01  4.63472899D+00  1.38629436D+00  1.60943791D+00
 ( 2)  4.61512052D+00  4.62497281D+00  4.64439090D+00  4.65396035D+00
 ( 1)  1.79175947D+00  1.94591015D+00  2.07944154D+00  2.19722458D+00
 ( 2)  4.66343909D+00  4.67282883D+00  4.68213123D+00  4.69134788D+00
 ( 1)  2.30258509D+00  2.39789527D+00  2.48490665D+00  2.56494936D+00
 ( 2)  4.70048037D+00  4.70953020D+00  4.71849887D+00  4.72738782D+00
 ( 1)  2.63905733D+00  2.70805020D+00  2.77258872D+00  2.83321334D+00
 ( 2)  4.73619845D+00  4.74493213D+00  4.75359019D+00  4.76217393D+00
 ( 1)  2.89037176D+00  2.94443898D+00  2.99573227D+00  3.04452244D+00
 ( 2)  4.77068462D+00  4.77912349D+00  4.78749174D+00  4.79579055D+00
 ( 1)  3.09104245D+00  3.13549422D+00  3.17805383D+00  3.21887582D+00
 ( 2)  4.80402104D+00  4.81218436D+00  4.82028157D+00  4.82831374D+00
 ( 1)  3.25809654D+00  3.29583687D+00  3.33220451D+00  3.36729583D+00
 ( 2)  4.83628191D+00  4.84418709D+00  4.85203026D+00  4.85981240D+00
 ( 1)  3.40119738D+00  3.43398720D+00
 ( 2)  4.86753445D+00  4.87519732D+00
 FILENAME : _
 OUTPUT-FILENAME : _
 /TS/ TIME-14:03:41 CPU-00:00:12
```

正規雑音時系列データの生成	T S. Ⅲ. 1
G A U S S	

機　能

白色雑音時系列データを生成する。必要が有れば，平均値並びに共分散行列を指定して白色雑音時系列データを生成する。

理論概要

平均値 0，標準偏差 σ_i（i＝1，…，N），共分散行列 $\boldsymbol{I}_N$ を持つ時系列 $\boldsymbol{x}_j \in \mathbf{R}^N$（j＝1，…，M）を，平均値 $\boldsymbol{m}=\{m_i\}\in\mathbf{R}^N$，共分散行列 $\boldsymbol{V}\in\mathbf{R}^{N\times N}$ を持つ時系列 $\bar{\boldsymbol{x}}_j\in\mathbf{R}^N$（j＝1，…，M）に変換することを考える。ただし $\boldsymbol{V}=\{v_{ij}\}$ は

$$\begin{cases} v_{ii}=1 \\ v_{ij}=v_{ji},\ |v_{ij}|\leqq 1 \quad (i\neq j) \end{cases}$$

であるとする。$\boldsymbol{V}$ を特異値分解することにより

$$\boldsymbol{V}=\boldsymbol{U}\boldsymbol{W}\,{}^t\boldsymbol{U}$$

と表わすことができる。ただし

$$\begin{aligned} &{}^t\boldsymbol{U}\boldsymbol{U}=\boldsymbol{I}, \quad \boldsymbol{U}\in\mathbf{R}^{N\times N} \\ &\boldsymbol{W}=\mathrm{diag}\{w_i\}\in\mathbf{R}^{N\times N}, \quad w_i>0 \end{aligned}$$

この時 $\bar{\boldsymbol{x}}_j$ は次のような変換で得られる。

$$\bar{\boldsymbol{x}}_j=\boldsymbol{U}\boldsymbol{W}^{1/2}\boldsymbol{x}_j+\boldsymbol{m} \qquad (j=1,\cdots,M)$$

ただし

$$\boldsymbol{W}^{1/2}=\mathrm{diag}\{\sqrt{w_i}\}\in\mathbf{R}^{N\times N}$$

こうして得られる $\bar{\boldsymbol{x}}_j$ の平均値 $\bar{\boldsymbol{m}}$　共分散行列 $\bar{\boldsymbol{V}}$ を求めてみると

$$\begin{aligned} \bar{\boldsymbol{m}} &= \frac{1}{M}\sum_j \bar{\boldsymbol{x}}_j=\frac{1}{M}\sum_j(\boldsymbol{U}\boldsymbol{W}^{1/2}\boldsymbol{x}_j+\boldsymbol{m})=\boldsymbol{U}\boldsymbol{W}^{1/2}\left(\frac{1}{M}\sum_j\boldsymbol{x}_j\right)+\boldsymbol{m}=\boldsymbol{m} \\ \bar{\boldsymbol{V}} &= \frac{1}{M}\sum_j(\bar{\boldsymbol{x}}_j-\bar{\boldsymbol{m}})\,{}^t(\bar{\boldsymbol{x}}_j-\bar{\boldsymbol{m}}) \\ &= \frac{1}{M}\sum_j\boldsymbol{U}\boldsymbol{W}^{1/2}\boldsymbol{x}_j\,{}^t\boldsymbol{x}_j\boldsymbol{W}^{1/2}\,{}^t\boldsymbol{U} \\ &= \boldsymbol{U}\boldsymbol{W}^{1/2}\left(\frac{1}{M}\sum_j\boldsymbol{x}_j\,{}^t\boldsymbol{x}_j\right)\boldsymbol{W}^{1/2}\,{}^t\boldsymbol{U} \\ &= \boldsymbol{V} \end{aligned}$$

となり確かに平均値 $\boldsymbol{m}$，共分散行列 $\boldsymbol{V}$ となっている。

実 行 例

〔1〕 正規雑音時系列データの生成例

```
 /TS/ TIME-14:03:56 CPU-00:00:01
GAUSS
 TO MAKE GAUSSIAN TIME SERIES
 SPECIFY (N,M) :   2,100
 SPECIFY MEAN VALUES AND COVARIANCE MATRIX ? _
 INPUT -MODE OF ( 2, 1) MATRIX MEAN      : Z
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX  COV      : _
 ( 1) ROW
1,0.5
 ( 2) ROW
0.5,1
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 2, 100) TIME SERIES GAUSS     : _
 *** ( 2, 100) TIME SERIES GAUSS      ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 2, 1) MATRIX X(0)       ---
 ( 1)   9.27949976D-01
 ( 2)   1.36007942D+00
 FILENAME : _
 === INTERVAL (    1-   50) ===
 --- ( 2,50) MATRIX DATA       ---
 ( 1) -8.95042231D-02 -1.03023585D+00  4.41628968D-01 -8.16062290D-01
 ( 2) -1.05346983D+00 -5.40876433D-01 -8.35425300D-02  2.07244634D-02
 ( 1)  1.81357814D+00 -1.84331489D-01  1.99208483D-01  7.99899187D-02
 ( 2)  2.37915350D+00 -9.02424909D-01  4.20447296D-01  7.98141434D-01
 ( 1) -2.02015866D+00 -1.81495939D-01  4.01987503D-01 -4.25300626D-01
 ( 2) -8.29438407D-01 -5.56604265D-01 -3.91539627D-02  1.14213405D+00
 ( 1)  4.66131248D-01  1.15260766D+00  1.46614952D+00 -2.71980005D-01
 ( 2)  7.12561278D-01 -1.50366711D-01  1.97629812D+00 -2.11328386D-02
 ( 1)  1.91399437D+00 -8.34525934D-01 -8.49091087D-01 -6.28860181D-01
 ( 2)  3.56660000D-01  2.17493459D-01 -3.76521124D-01 -1.61926025D+00
 ( 1) -3.35585413D-01  5.38010004D-01 -1.38893015D+00  9.36608027D-01
 ( 2) -4.71197434D-01  1.56593897D-01 -1.70346608D-01  1.39651193D+00
 ( 1)  1.39178771D+00 -3.39821519D-01 -6.55532628D-01  6.01928481D-01
 ( 2)  3.63345731D-01 -1.63939145D+00  2.40117661D-01  1.94898500D-01
 ( 1)  8.81489632D-01  1.21945608D+00  8.52221470D-01 -2.91855991D-01
 ( 2)  4.80626333D-01  1.19572494D+00  8.78452601D-01  4.83153230D-01
 ( 1)  7.52308216D-01 -4.49802170D-01  1.54066084D+00  2.69455355D-01
 ( 2) -2.90606840D-01  5.00090378D-02  4.84478507D-02  1.01668780D+00
 ( 1) -9.71759973D-01  6.14399376D-01 -6.80600884D-01 -1.60753428D-01
 ( 2) -3.41396280D-01  1.31416742D+00 -1.06782202D+00  4.65097196D-01
 ***
 ( 1)  9.68617374D-01 -1.61303979D+00  4.31461877D-01  5.72262860D-01
 ( 2)  5.47551259D-01 -2.33189483D-01  1.42012951D+00  4.74337560D-01
 ( 1)  4.07854262D-01 -6.27655782D-01 -7.33246829D-01 -7.03592675D-01
 ( 2)  2.54561072D+00 -9.08886853D-01 -2.07533259D+00  1.73852294D+00
 ( 1) -3.16350369D-01 -2.15928441D-01
 ( 2) -2.99558022D-01  5.43064177D-02
 FILENAME :_
```

```
=== INTERVAL (   51- 100) ===
--- ( 2,50) MATRIX DATA      ---
( 1)   9.40893846D-01 -5.23157395D-01  8.33139770D-01 -1.89310347D+00
( 2)  -4.11265122D-02 -1.03597310D+00 -8.65857737D-01  6.98555566D-02
( 1)  -1.56675956D-02 -1.18593293D+00  1.02517266D-01 -1.57306544D+00
( 2)   8.25657172D-01 -1.53244125D+00  5.52865098D-01 -1.47241306D+00
( 1)  -1.00701989D+00 -1.08621218D+00 -1.99079117D+00 -5.32439377D-01
( 2)   2.68390665D-01 -1.53963465D-01 -1.69100187D+00  8.29586456D-01
( 1)  -2.99992243D-01 -9.45130154D-01 -9.97870672D-01 -3.38794036D-01
( 2)  -4.57294304D-01 -1.25546382D+00 -9.38582835D-01 -5.38003691D-01
( 1)  -6.22230555D-01 -1.01643530D+00  8.73224526D-01 -1.12821000D+00
( 2)  -1.81582694D-01 -3.49479819D-01  1.18702891D+00  1.71744685D+00
( 1)  -1.56412119D+00 -3.80718147D-01  1.31964890D+00 -3.27711545D-01
( 2)  -2.22399937D-01 -4.00391071D-01  3.42948766D-01 -1.02737527D+00
( 1)   5.19285210D-01 -1.48130972D+00  5.05943486D-01  1.20113979D+00
( 2)   7.12664275D-01 -3.60317651D-01 -9.71321286D-01  5.44679361D-01
***
( 1)   4.81994321D-01 -2.39223070D+00  9.99342303D-02  1.93981337D+00
( 2)  -4.99923457D-01  5.62930700D-01  6.22005642D-01  1.46879104D+00
( 1)   2.60949997D+00  8.37323629D-02 -1.52149661D+00  2.39088891D-01
( 2)  -1.77208370D-01  8.07778361D-01 -1.41522726D+00  1.02445040D+00
( 1)   1.64301455D-01 -4.90745855D-02 -3.17927815D-01  1.28315800D+00
( 2)  -2.84230212D-01 -5.21778032D-03  4.17362167D-01  2.69956002D+00
( 1)  -1.59613733D-01  5.47106356D-01  8.62041922D-01  8.67192428D-01
( 2)   2.04594294D-02 -5.47088886D-01  2.10613617D+00  9.68002574D-01
( 1)   6.38874382D-01  7.49216401D-01 -1.05895197D+00 -1.01419874D+00
( 2)   1.26044147D+00  4.25317197D-01 -1.21630890D+00 -1.12190192D+00
( 1)  -2.65956307D-01 -5.26619922D-01
( 2)  -4.92545251D-01 -4.30664067D-01
FILENAME : _
OUTPUT-FILENAME :   G
/TS/ TIME-14:04:03 CPU-00:00:10
```

最大周期時系列データ（M－系列）の生成	T S. Ⅲ. 2
M S E Q	

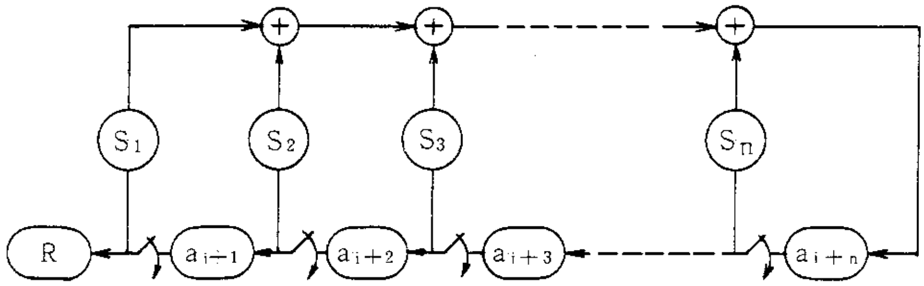
機　能

指定する2値を取るM－系列データを生成する。その際，最大周期（2^i-1）を規定するビット量 i を指定する。

理論概要 [25)]

M系列（Maximum length sequence)は，外観上不規則に見える確定的時系列データである。特に，取り得る値が2値の場合を，本コマンドで用いている2レベルのM系列と称する。以下はこのM系列について述べる。

今，取り得る値を0か1とし（0，1である必要はない），下図に示すn段のフィードバック付サンプルホールド型ロジック回路を構成すると，レジスタ R に入る値の系列がM系列となる。



$\oplus$ ：Exclusive OR，$S_i=0$ 又は1の固定値

—⁄— ；1時間ステップ毎に矢印の方向にデータを1つ順送りすることを意味する

これは，最大の周期（2^n-1）を持つ時系列データとなる。漸化式は，$a_{i+n+1}=\sum_{j=1}^{n}S_j a_{i+j}$（modulus 2の加算）。ここで $S_{n+1}=1$ と置くと漸化式は

$$\sum_{j=1}^{n+1} S_j a_{i+j}=0$$

となり，さらに $a_{i+j}=x^j a_i$ なる遅延演算子 x を導入すると $(\sum_{j=1}^{n+1} S_j x^j)\,a_i=0$ と書き直せる。ここで特性多項式 $S(x)=\sum_{j=1}^{n+1} S_j x^j$ を定義すると，n次多項式 S(x)が原始多項式である時，系列 a_i がM系列となる。原始多項式であるとは（x^k-1）／S(x) なる演算において，$0<k<2^n-1$ のとき割り切れず，$k=2^n-1$ のときのみ割り切れる，ことを意味する。

M系列信号は，不規則性を有していることから真に不規則な信号の代用として同定の際などに使用できる。しかも，本当に不規則な信号であると有限時間平均の統計的ばらつきは避けられないが，M系列の場合，それがないという有利な点を持っている。

実 行 例

〔1〕 最大周期時系列データの生成例

```
/TS/ TIME-14:04:12 CPU-00:00:01
MSEQ
 TO MAKE M-SEQUENCE TIME SERIES
 SPECIFY (N,M) :   2,100
 NO. OF BITS (2-20) ? :   5
 LOW VALUE AND HIGH VALUE ? :   -1,1
 OUTPUT-MODE OF ( 2, 100) TIME SERIES MSEQ      : _
 *** ( 2, 100) TIME SERIES MSEQ      ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 2, 1) MATRIX X(0)      ---
 ( 1)   1.00000000D+00
 ( 2)   1.00000000D+00
 FILENAME : _
 === INTERVAL (    1-   50) ===
 --- ( 2,50) MATRIX DATA      ---
 ( 1)   1.00000000D+00 -1.00000000D+00  1.00000000D+00 -1.00000000D+00
 ( 2)   1.00000000D+00 -1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)  -1.00000000D+00  1.00000000D+00 -1.00000000D+00 -1.00000000D+00
 ( 2)  -1.00000000D+00 -1.00000000D+00 -1.00000000D+00  1.00000000D+00
 ( 1)  -1.00000000D+00 -1.00000000D+00  1.00000000D+00 -1.00000000D+00
 ( 2)   1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ***
 ( 1)   1.00000000D+00 -1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  -1.00000000D+00 -1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)   1.00000000D+00 -1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  -1.00000000D+00  1.00000000D+00 -1.00000000D+00 -1.00000000D+00
 ( 1)  -1.00000000D+00 -1.00000000D+00 -1.00000000D+00  1.00000000D+00
 ( 2)   1.00000000D+00 -1.00000000D+00 -1.00000000D+00 -1.00000000D+00
 ( 1)   1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  -1.00000000D+00  1.00000000D+00 -1.00000000D+00  1.00000000D+00
 ( 1)  -1.00000000D+00 -1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  -1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)  -1.00000000D+00  1.00000000D+00 -1.00000000D+00 -1.00000000D+00
 ( 2)  -1.00000000D+00  1.00000000D+00  1.00000000D+00 -1.00000000D+00
 ( 1)   1.00000000D+00 -1.00000000D+00 -1.00000000D+00 -1.00000000D+00
 ( 2)  -1.00000000D+00 -1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 1)  -1.00000000D+00  1.00000000D+00 -1.00000000D+00  1.00000000D+00
 ( 2)   1.00000000D+00  1.00000000D+00  1.00000000D+00 -1.00000000D+00
 ( 1)  -1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
 ( 2)  -1.00000000D+00  1.00000000D+00  1.00000000D+00 -1.00000000D+00
 ( 1)  -1.00000000D+00  1.00000000D+00
 ( 2)   1.00000000D+00 -1.00000000D+00
 FILENAME :
```

```
=== INTERVAL (   51- 100) ===
--- ( 2,50) MATRIX DATA      ---
***
( 1)  1.00000000D+00 -1.00000000D+00 -1.00000000D+00 -1.00000000D+00
( 2) -1.00000000D+00  1.00000000D+00 -1.00000000D+00 -1.00000000D+00
( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
( 2) -1.00000000D+00 -1.00000000D+00  1.00000000D+00 -1.00000000D+00
( 1)  1.00000000D+00 -1.00000000D+00 -1.00000000D+00  1.00000000D+00
( 2)  1.00000000D+00 -1.00000000D+00  1.00000000D+00  1.00000000D+00
( 1)  1.00000000D+00 -1.00000000D+00  1.00000000D+00 -1.00000000D+00
( 2)  1.00000000D+00 -1.00000000D+00  1.00000000D+00  1.00000000D+00
( 1) -1.00000000D+00  1.00000000D+00 -1.00000000D+00 -1.00000000D+00
( 2) -1.00000000D+00 -1.00000000D+00 -1.00000000D+00  1.00000000D+00
( 1) -1.00000000D+00 -1.00000000D+00  1.00000000D+00 -1.00000000D+00
( 2)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
( 1)  1.00000000D+00 -1.00000000D+00  1.00000000D+00  1.00000000D+00
( 2) -1.00000000D+00 -1.00000000D+00  1.00000000D+00  1.00000000D+00
( 1)  1.00000000D+00 -1.00000000D+00  1.00000000D+00  1.00000000D+00
( 2) -1.00000000D+00  1.00000000D+00 -1.00000000D+00 -1.00000000D+00
( 1) -1.00000000D+00 -1.00000000D+00 -1.00000000D+00  1.00000000D+00
( 2)  1.00000000D+00 -1.00000000D+00 -1.00000000D+00 -1.00000000D+00
( 1)  1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
( 2) -1.00000000D+00  1.00000000D+00 -1.00000000D+00  1.00000000D+00
( 1) -1.00000000D+00 -1.00000000D+00  1.00000000D+00  1.00000000D+00
( 2) -1.00000000D+00  1.00000000D+00  1.00000000D+00  1.00000000D+00
( 1) -1.00000000D+00  1.00000000D+00 -1.00000000D+00 -1.00000000D+00
***
( 2) -1.00000000D+00  1.00000000D+00  1.00000000D+00 -1.00000000D+00
( 1)  1.00000000D+00 -1.00000000D+00
( 2) -1.00000000D+00 -1.00000000D+00
FILENAME : _
OUTPUT-FILENAME :   M
/TS/ TIME-14:04:27 CPU-00:00:11
```

正弦波時系列データの生成	TS. Ⅲ. 3
SIN	

機　能

正弦波時系列データ$\boldsymbol{A}(1, m)$を生成する。

$$\boldsymbol{A}(1, m) = \{a_j\} = W \cdot \sin\{\omega(j \cdot T - \theta)\}, \quad (j = 1, \cdots, m)$$

但し，T：サンプル周期，θ：開始時刻，W：振幅，ω：$2\pi / T_p$，T_p：周期

＜注意事項＞

本コマンドにて作成される時系列データは，指定した開始時刻に係わらず，時刻$t = 0$からの時系列データを作成する。従って，指定した開始時刻$\theta > 0$の場合には，時刻$t = 0$から時刻$t = \theta$の間のデータは零が補われている。また，指定した開始時刻$\theta < 0$の場合には開始時刻$t = \theta$から時刻$t = 0$の間のデータは欠落している。

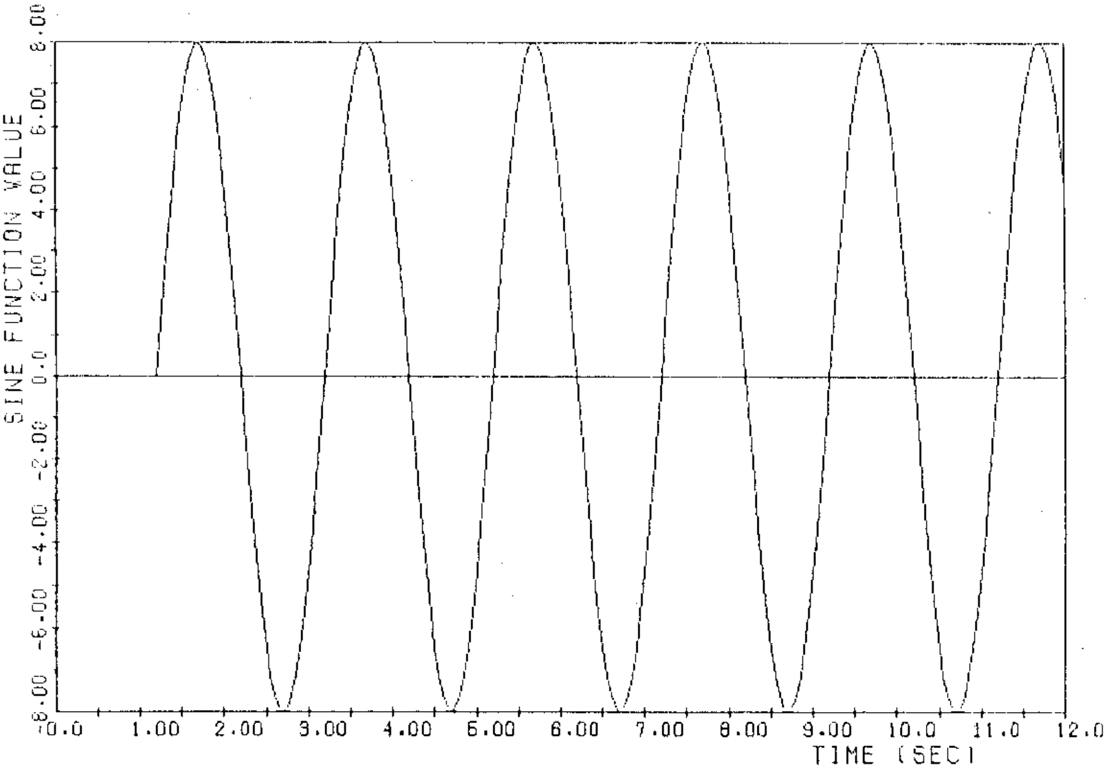
実 行 例

〔1〕　正弦波時系列データの生成例（1）

```
 /TS/ TIME-14:00:01 CPU-00:00:01
SIN
 TO CALCULATE SIN(1,M) TIME SERIES  DATA
 SPECIFY M :   2100
 GIVE AMPLITUDE, PERIOD, DELTAT, START TIME  8,2,0.05,1.2
 OUTPUT-MODE OF ( 1,2100) TIME SERIES  SIN      : F
 OUTPUT-FILENAME :   SIN
 /TS/ TIME-14:00:15 CPU-00:00:14
```

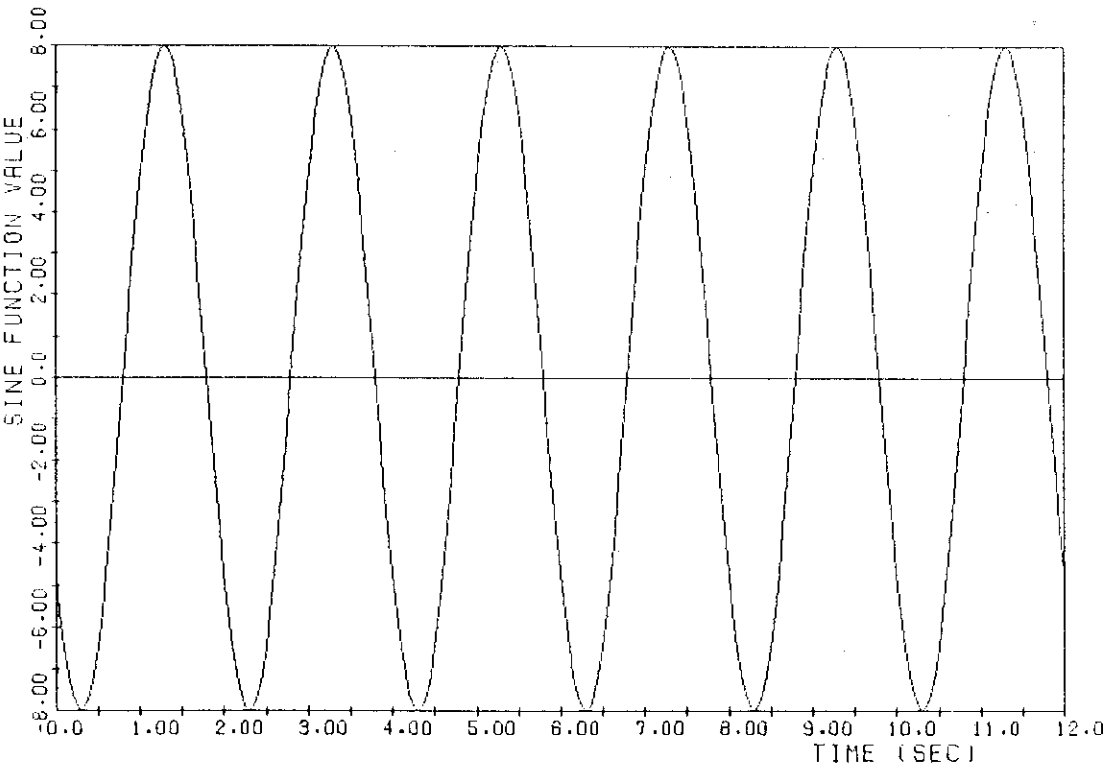
〔2〕　正弦波時系列データの生成例（2）

```
 /TS/ TIME-14:00:28 CPU-00:00:01
SIN
 TO CALCULATE SIN(1,M) TIME SERIES  DATA
 SPECIFY M :   2100
 GIVE AMPLITUDE, PERIOD, DELTAT, START TIME  8,2,0.05,-1.2
 OUTPUT-MODE OF ( 1,2100) TIME SERIES  SIN      : F
 OUTPUT-FILENAME :   SIND
 /TS/ TIME-14:00:39 CPU-00:00:13
```



START TIME=1.2 PERIOD=2 INTERVAL TIME=0.05

正弦波時系列データ（開始時刻≧０）の例



START TIME=-1.2 PERIOD=2 INTERVAL TIME=0.05

正弦波時系列データ（開始時刻＜０）の例

方形波時系列データの生成	TS. Ⅲ. 4
SQU	

機　能

方形波時系列データ $\boldsymbol{A}(1, m)$ を生成する。

①$j \cdot T < \theta$ の場合　　$a_j = 0$

②$\theta + (n-1)T_p \leqq j \cdot T < \theta + (2n-1)T_p/2$ の場合　　$a_j = W$

③$\theta + (2n-1)T_p/2 \leqq j \cdot T < \theta + nT_p$ の場合　　$a_j = -W$

但し，T：サンプル周期，θ：開始時刻，W：振幅，T_p：周期，n：正整数

＜注意事項＞

本コマンドにて作成される時系列データは，指定した開始時刻に係わらず，時刻 $t = 0$ からの時系列データを作成する。従って，指定した開始時刻 $\theta > 0$ の場合には，時刻 $t = 0$ から時刻 $t = \theta$ の間のデータは零が補われている。また，指定した開始時刻 $\theta < 0$ の場合には開始時刻 $t = \theta$ から時刻 $t = 0$ の間のデータは欠落している。

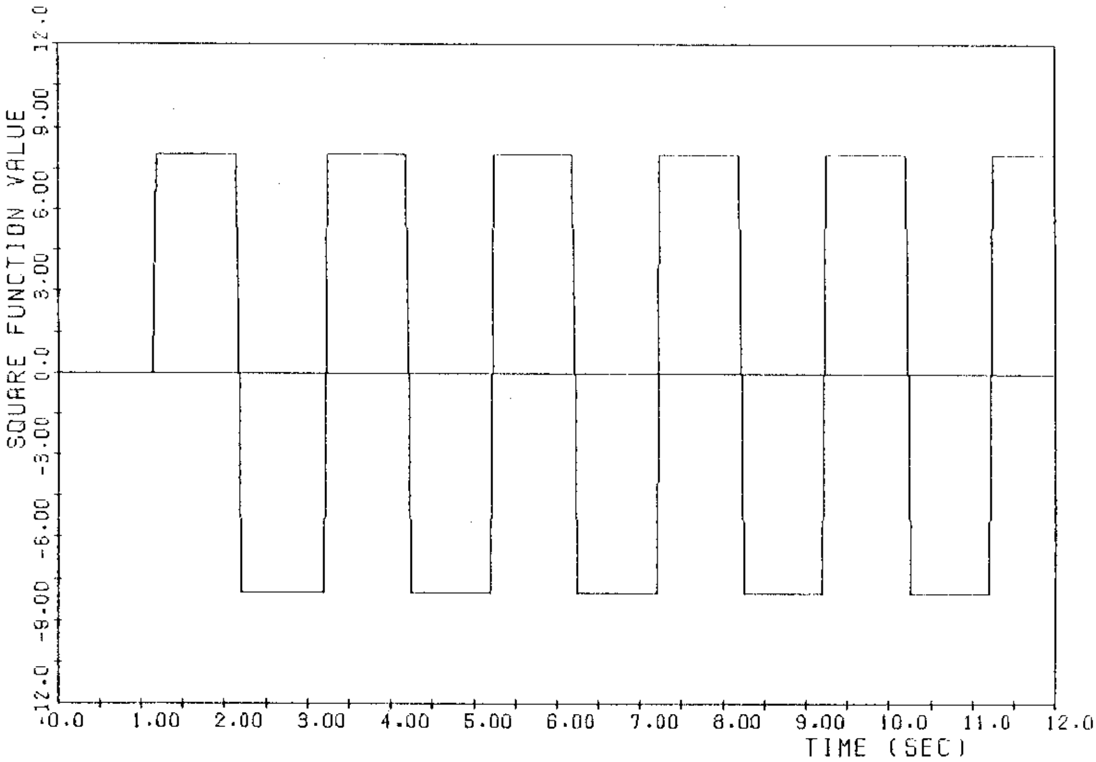
実 行 例

〔1〕　方形波時系列データの生成例（1）

```
 /TS/ TIME-14:00:54 CPU-00:00:01
SQU
 TO CALCULATE SQU(1,M) TIME SERIES  DATA
 SPECIFY M :   2100
 GIVE AMPLITUDE, PERIOD, DELTAT, START TIME  8,2,0.05,1.2
 OUTPUT-MODE OF ( 1,2100) TIME SERIES  SQU      :  F
 OUTPUT-FILENAME :   SQU
 /TS/ TIME-14:01:10 CPU-00:00:15
```

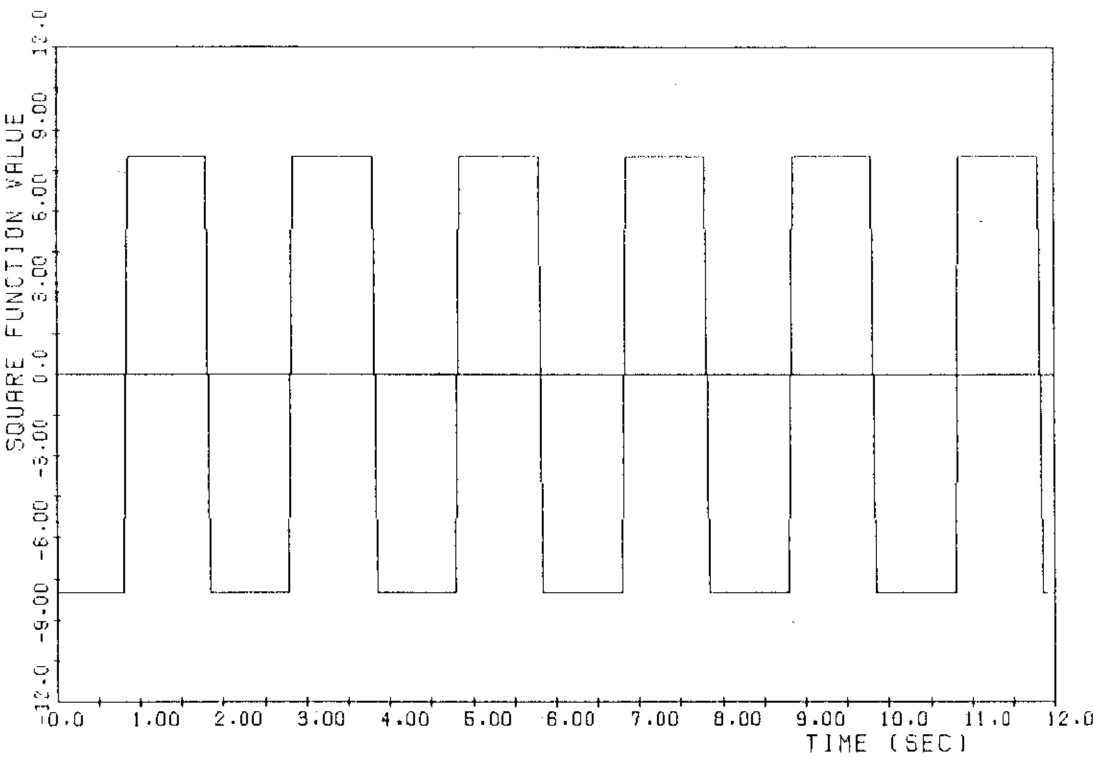
〔2〕　方形波時系列データの生成例（2）

```
 /TS/ TIME-14:01:28 CPU-00:00:01
SQU
 TO CALCULATE SQU(1,M) TIME SERIES  DATA
 SPECIFY M :   2100
 GIVE AMPLITUDE, PERIOD, DELTAT, START TIME  8,2,0.05,-1.2
 OUTPUT-MODE OF ( 1,2100) TIME SERIES  SQU      :  F
 OUTPUT-FILENAME :   SQUD
 /TS/ TIME-14:01:43 CPU-00:00:20
```



START TIME=1.2 PERIOD=2 INTERVAL TIME=0.05

方形波時系列データ（開始時刻≧０）の例



START TIME=-1.2 PERIOD=2 INTERVAL TIME=0.05

方形波時系列データ（開始時刻＜０）の例

鋸波時系列データの生成	T S. Ⅲ. 5
S A W	

機　能

鋸形波時系列データ $\boldsymbol{A}(1, m)$ を生成する。

①$j \cdot T < \theta$ の場合　　　　　　$a_j = 0$

②$\theta + (n-1) T_p \leqq j \cdot T < \theta + n T_p$ の場合

$$a_j = W (j \cdot T - (n-1) T_p - \theta) / T_p$$

但し，T：サンプル周期，θ：開始時刻，W：振幅，T_p：周期，n：正整数

<注意事項>

本コマンドにて作成される時系列データは，指定した開始時刻に係わらず，時刻 $t = 0$ からの時系列データを作成する。従って，指定した開始時刻 $\theta > 0$ の場合には，時刻 $t = 0$ から時刻 $t = \theta$ の間のデータは零が補われている。また，指定した開始時刻 $\theta < 0$ の場合には開始時刻 $t = \theta$ から時刻 $t = 0$ の間のデータは欠落している。

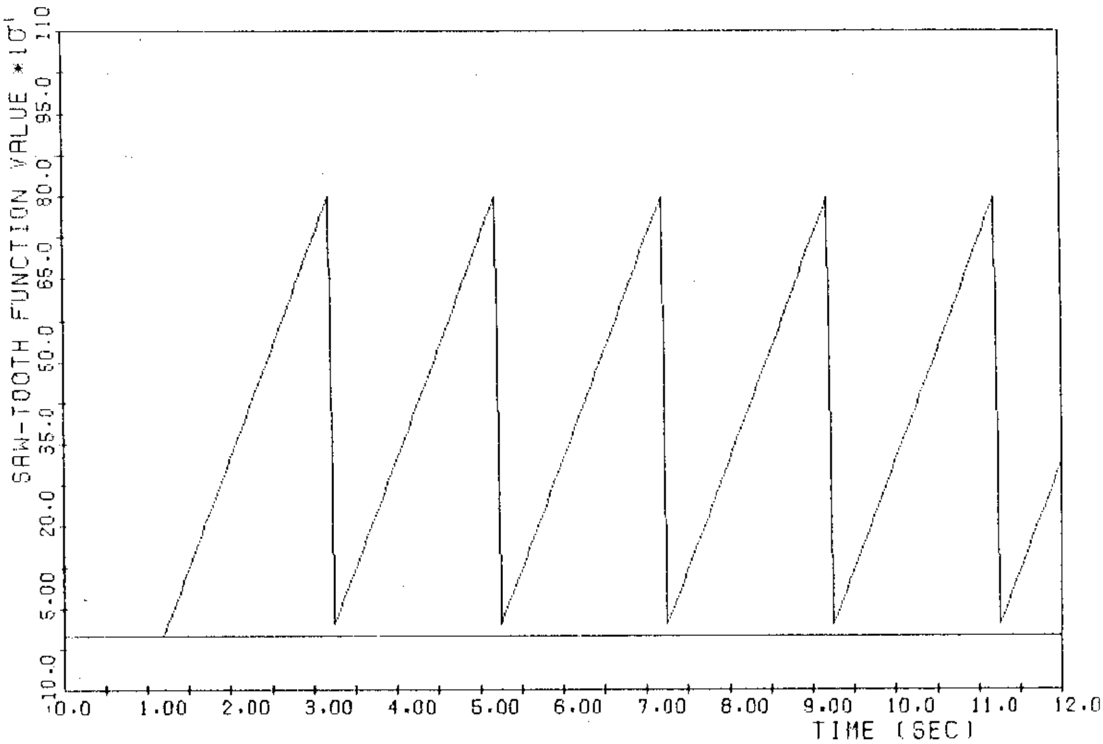
実 行 例

〔1〕　鋸波時系列データの生成例（1）

```
 /TS/ TIME-14:02:07 CPU-00:00:01
SAW
 TO CALCULATE SAW(1,M) TIME SERIES  DATA
 SPECIFY M :   2100
 GIVE AMPLITUDE, PERIOD, DELTAT, START TIME  8,2,0.05,1.2
 OUTPUT-MODE OF ( 1,2100) TIME SERIES  SAW      :  F
 OUTPUT-FILENAME :   SAW
 /TS/ TIME-14:02:15 CPU-00:00:08
```

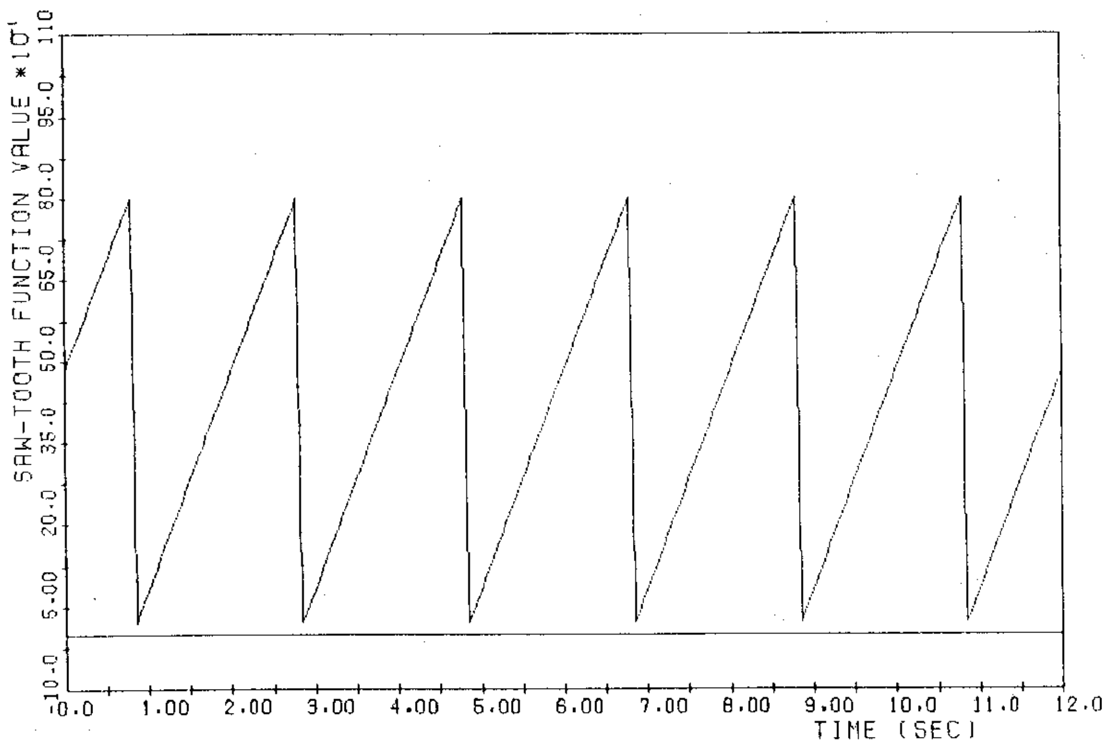
〔2〕　鋸波時系列データの生成例（2）

```
 /TS/ TIME-14:02:29 CPU-00:00:01
SAW
 TO CALCULATE SAW(1,M) TIME SERIES  DATA
 SPECIFY M :   2100
 GIVE AMPLITUDE, PERIOD, DELTAT, START TIME  8,2,0.05,-1.2
 OUTPUT-MODE OF ( 1,2100) TIME SERIES  SAW      :  F
 OUTPUT-FILENAME :   SAWD
 /TS/ TIME-14:02:40 CPU-00:00:06
```



START TIME=1.2 PERIOD=2 INTERVAL TIME=0.05

鋸波時系列データ（開始時刻≧0）の例



START TIME=-1.2 PERIOD=2 INTERVAL TIME=0.05

鋸波時系列データ（開始時刻＜0）の例

三角波時系列データの生成	TS. III. 6
TRI	

機　能

三角形波時系列データ $\boldsymbol{A}$（1，m）を生成する。

①$j \cdot T < \theta$ の場合　$a_j = 0$

②$\theta + (n - 5/4) T_p \leqq j \cdot T < \theta + (n - 3/4) T_p$ の場合

$$a_j = 4W (j \cdot T - (n-1) T_p - \theta) / T_p - W$$

③$\theta + (n - 3/4) T_p \leqq j \cdot T < \theta + (n - 1/4) T_p$ の場合

$$a_j = 4W (1 - j \cdot T + (2n-1) T_p / 2 + \theta) / T_p - W$$

但し，T：サンプル周期，θ：開始時刻，W：振幅，T_p：周期，n：正整数

<注意事項>

本コマンドにて作成される時系列データは，指定した開始時刻に係わらず，時刻 t = 0 からの時系列データを作成する。従って，指定した開始時刻 $\theta > 0$ の場合には，時刻 t = 0 から時刻 t = θ の間のデータは零が補われている。また，指定した開始時刻 $\theta < 0$ の場合には開始時刻 t = θ から時刻 t = 0 の間のデータは欠落している。

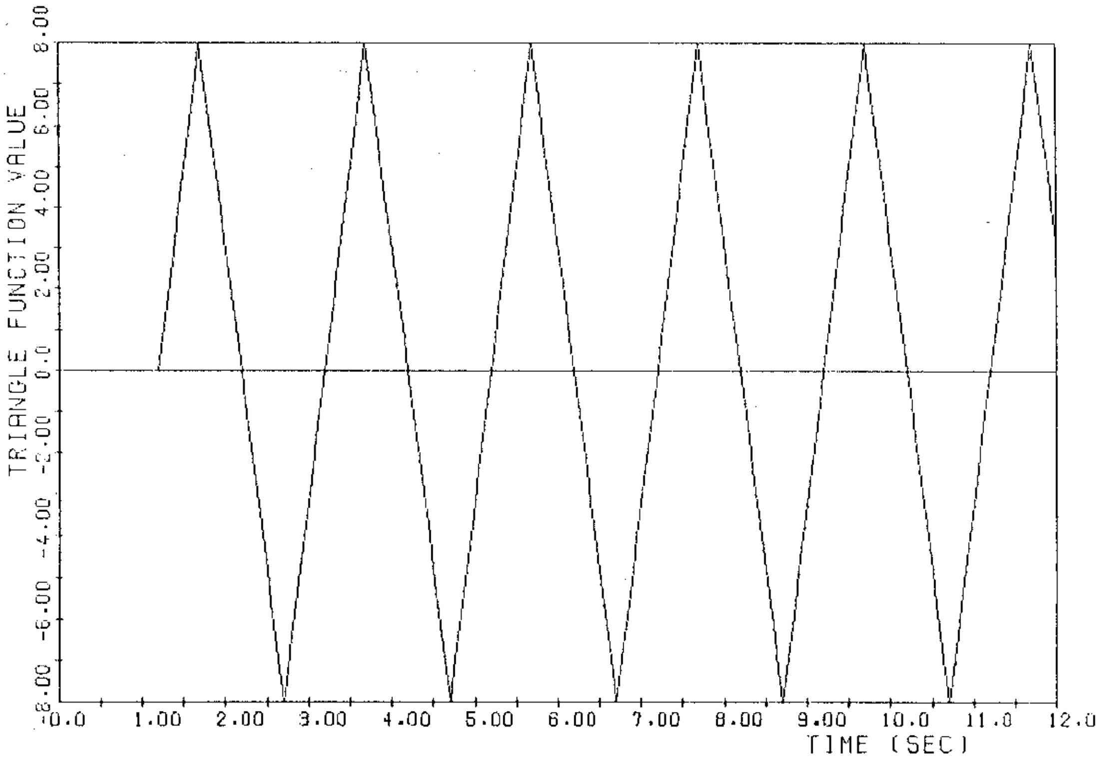
実 行 例

〔1〕　三角波時系列データの生成例（1）

```
 /TS/ TIME-14:02:57 CPU-00:00:01
TRI
 TO CALCULATE TRI(1,M) TIME SERIES  DATA
 SPECIFY M :   2100
 GIVE AMPLITUDE, PERIOD, DELTAT, START TIME  8,2,0.05,1.2
 OUTPUT-MODE OF ( 1,2100) TIME SERIES  TRI       : F
 OUTPUT-FILENAME :   TRI
 /TS/ TIME-14:03:11 CPU-00:00:09
```

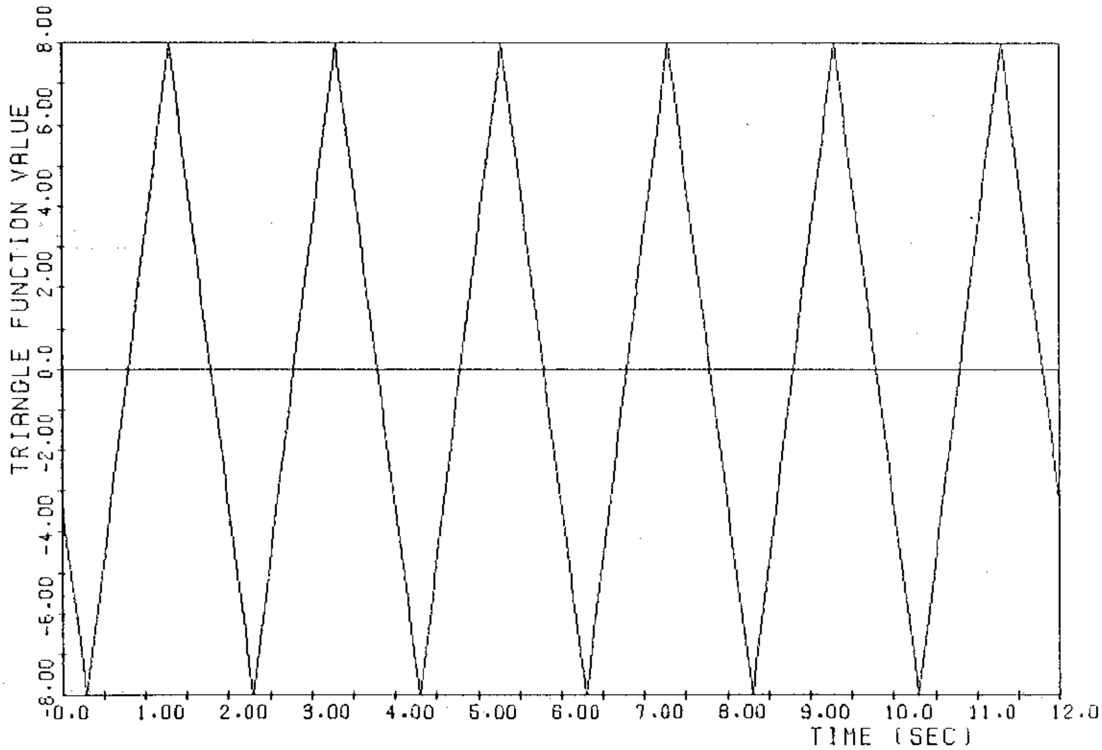
〔2〕　三角波時系列データの生成例（2）

```
 /TS/ TIME-14:03:24 CPU-00:00:01
TRI
 TO CALCULATE TRI(1,M) TIME SERIES  DATA
 SPECIFY M :   2100
 GIVE AMPLITUDE, PERIOD, DELTAT, START TIME  8,2,0.05,-1.2
 OUTPUT-MODE OF ( 1,2100) TIME SERIES  TRI       : F
 OUTPUT-FILENAME :   TRID
 /TS/ TIME-14:03:41 CPU-00:00:12
```



START TIME=1.2 PERIOD=2 INTERVAL TIME=0.05

三角波時系列データ（開始時刻≧０）の例



START TIME=-1.2 PERIOD=2 INTERVAL TIME=0.05

三角波時系列データ（開始時刻＜０）の例

時系列データの各種統計量の計算	TS. Ⅳ. 1
STAT	

機　能

〔1〕 時系列データの各種統計量の計算を行う。
計算する量としては，①平均値，②共分散行列，③標準偏差，④相関係数の4つである。

〔2〕 時系列データを指定する平均値及び標準偏差を持つ時系列データに変換する。

理論概要

◆時系列データ $\boldsymbol{A}(N, M) = \{a_{ij}\}$ について次の値を計算する。

平均値：$m_i = \dfrac{1}{M+1}\sum_{j=0}^{M} a_{ij} \quad (i=1, \cdots, N)$

共分散行列：$\boldsymbol{V} = \{v_{ij}\} \in \mathbf{R}^{N \times N}$

$$v_{ij} = \frac{1}{M+1}\left\{\sum_{k=0}^{M} a_{ik} \cdot a_{jk} - \frac{1}{M+1}\sum_{k=0}^{M} a_{ik}\sum_{k=0}^{M} a_{jk}\right\}$$

標準偏差：$\sigma_i = \sqrt{v_{ii}} \quad (i=1, \cdots, N)$

相関係数：$\boldsymbol{R} = \{r_{ij}\} \in \mathbf{R}^{N \times N}$

$$r_{ij} = v_{ij} / \sigma_i \sigma_j$$

◆平均値 m_i，標準偏差 σ_i $(i=1, \cdots, N)$ を持つ時系列データ $\boldsymbol{A}(N, M) = \{a_{ij}\}$ から平均値 $\overline{m}_i$，標準偏差 $\overline{\sigma}_i$ を持つ時系列データ $\overline{\boldsymbol{A}}(N, M) = \{\overline{a}_{ij}\}$ への変換は

$$\overline{a}_{ij} = \frac{\overline{\sigma}_i}{\sigma_i}(a_{ij} - m_i) + \overline{m}_i$$

により行なう。

実 行 例

〔1〕 時系列データの各種統計量の計算例

```
/TS/ TIME-14:03:56 CPU-00:00:01
STAT
TO CALCULATE STATISTICS OF A(N,M)
SPECIFY (N,M) :    2,30
INPUT -MODE OF ( 2,  30) TIME SERIES   A        :  F
INPUT -FILENAME :   A2
*** STATISTICS OF ( 2,  30) TIME SERIES A2          ***
1) MEAN VALUES
--- ( 2, 1) MATRIX MEAN      ---
( 1)   1.82258065D+01
( 2)   1.11709677D+02
```

```
FILENAME : _
2) COVARIANCE MATRIX
--- ( 2, 2) MATRIX  COV      ---
( 1)   3.08303850D+02  9.47107180D+01
( 2)   9.47107180D+01  4.90335068D+02
FILENAME : _
3) STANDARD DEVIATIONS
--- ( 2, 1) MATRIX  SD       ---
( 1)   1.75585834D+01
( 2)   2.21435107D+01
FILENAME : _
4) CORRELATION COEFFICIENTS
--- ( 2, 2) MATRIX  COR      ---
( 1)   1.00000000D+00  2.43592088D-01
( 2)   2.43592088D-01  1.00000000D+00
FILENAME : _
NORMALIZE ? _
SPECIFY NEW MEAN VALUES AND STANDARD DEVIATIONS ? _
INPUT -MODE OF ( 2, 1) MATRIX MEAN       : Z
TYPE OR REVISE ? _
INPUT -MODE OF ( 2, 1) MATRIX  SD        : I
TYPE OR REVISE ? _
OUTPUT-MODE OF ( 2,  30) TIME SERIES NRMZ(A)  :   _
*** ( 2,  30) TIME SERIES NRMZ(A)  ***
ALL DATA ? _
=== INITIAL DATA ===
--- ( 2, 1) MATRIX X(0)       ---
( 1) -1.03799982D+00
( 2)  0.0
FILENAME : _
=== INTERVAL (    1-   30) ===
--- ( 2,30) MATRIX DATA       ---
( 1) -9.81047621D-01  4.77112485D+00 -8.67143216D-01 -8.10191013D-01
( 2)  0.0             0.0             0.0             0.0
( 1) -7.53238810D-01 -6.96286608D-01 -6.39334405D-01 -5.82382202D-01
( 2)  0.0             0.0             0.0             0.0
( 1) -5.25429999D-01 -4.68477797D-01 -4.11525594D-01 -3.54573391D-01
( 2)  0.0             0.0             0.0             0.0
( 1) -2.97621188D-01 -2.40668986D-01 -1.83716783D-01 -1.26764580D-01
( 2)  0.0             0.0             0.0             0.0
( 1) -6.98123775D-02 -1.28601748D-02  4.40920279D-02  1.01044231D-01
( 2)  0.0             0.0             0.0             0.0
( 1)  1.57996433D-01  2.14948636D-01  2.71900839D-01  3.28853042D-01
( 2)  0.0             0.0             0.0             0.0
( 1)  3.85805244D-01  4.42757447D-01  4.99709650D-01  5.56661852D-01
( 2)  0.0             0.0             0.0             0.0
( 1)  6.13614055D-01  6.70566258D-01
( 2)  0.0             0.0
FILENAME : _
OUTPUT-FILENAME : _
/TS/ TIME-14:04:03 CPU-00:00:10
```

時系列データの相関関数の算出	TS. Ⅳ. 2
CORF	

機　能

〔1〕 時系列データの自己相関関数ならびに相互相関関数の計算を行う。

理論概要

平均値 m_x，標準偏差 σ_x の時系列 x_i（$i=1, \cdots, M$）が与えられた時

$$\phi_{xx}(\tau)=\frac{1}{N-\tau}\cdot\frac{1}{\sigma_x^2}\sum_{i=1}^{N-\tau}(x_i-m_x)\cdot(x_{i+\tau}-m_x) \qquad (\tau\geq 0)$$

を自己相関関数という。

さらに平均値 m_y，標準偏差 σ_y の時系列 y_i（$i=1, \cdots, M$）が与えられた時

$$\phi_{xy}(\tau)=\frac{1}{N-\tau}\cdot\frac{1}{\sigma_x\sigma_y}\sum_{i=1}^{N-\tau}(x_i-m_x)\cdot(y_{i+\tau}-m_y) \qquad (\tau\geq 0)$$

を相互相関関数という。

2つの時系列データ $\boldsymbol{A}(N_1, M_1)$，$\boldsymbol{B}(N_2, M_2)$ について

$$M=\min(M_1, M_2)$$

として処理する。また a_i（$i=1, \cdots, N_1$）と b_j（$j=1, \cdots, N_2$）の相関関数 $\phi_{a_ib_j}$ は結果の時系列データの第 $\{N_1\cdot(i-1)+j\}$ 行に得られる。ただし $N_1\cdot N_2>20$ の場合には相関関数を求める変数の対を最大20組まで指定する。（0, 0 の入力により指定は終了する。）

実 行 例

〔1〕 2つの時系列データ同士の相互相関関数の算出例

```
/TS/ TIME-14:04:12 CPU-00:00:01
CORF
 TO CALCULATE CORRELATION FUNCTION OF A(N1,M1) AND B(N 2,M2)
 SPECIFY (N1,M1,N2,M2) :   2,100,2,100
 INPUT -MODE OF ( 2, 100) TIME SERIES   A       :  F
 INPUT -FILENAME :   G
 INPUT -MODE OF ( 2, 100) TIME SERIES   B       :  F
 INPUT -FILENAME :   M
 MAXIMUM LAG ? :   5
 OUTPUT-MODE OF ( 4,    5) TIME SERIES COR(A,B) : _
 *** ( 4,    5) TIME SERIES COR(A,B) ***
 ALL DATA ? _
 === INITIAL DATA ===
 --- ( 4, 1) MATRIX X(0)      ---
 ( 1)   1.65180102D-01
 ( 2)  -2.15202547D-02
 ( 3)   4.75179484D-02
 ( 4)  -4.30407377D-02
 FILENAME : _
 === INTERVAL (    1-    5) ===
 --- ( 4, 5) MATRIX DATA      ---
 ( 1) -9.70644988D-02 -7.07446821D-02  1.25380952D-01  2.93341779D-02
 ( 2)  6.27058137D-02 -3.40273952D-02 -1.64731234D-01  1.05142013D-01
 ( 3)  1.36619348D-02 -6.98903476D-02  1.46200081D-01 -4.63803133D-02
 ( 4)  4.58383910D-02  5.00427142D-02  1.64600284D-01  2.65241102D-01
 ( 1) -8.52448418D-02
 ( 2) -4.07858522D-02
 ( 3) -1.77380727D-01
 ( 4) -2.33375586D-01
 FILENAME : _
 OUTPUT-FILENAME : _
 /TS/ TIME-14:04:27 CPU-00:00:11
```

時系列データの高速フーリエ変換の実行	T S. Ⅳ. 3
F F T	

機　　能

時系列データの指定する行に対して高速フーリエ変換を行う。

理論概要 [26)]

ある時系列データ $\{x(j),\ j=0,\ 1,\ \cdots,\ N-1\}$ とその複素FFT成分 $\{X(k),\ k=0,\ 1,\ \cdots,\ \frac{N}{2}-1,\ \frac{N}{2},\ \cdots,\ N-1\}$ 対応は，下式で表わせる。（Tはサンプル周期）

$$X(k)=T\sum_{j=0}^{N-1} x(j)\exp\left(-i\,2\pi\frac{kj}{N}\right)$$

尚，逆変換は，$x(j)=\dfrac{2}{N\cdot T}\displaystyle\sum_{k=0}^{\frac{N}{2}-1} X(k)\exp\left(i\,2\pi\frac{jk}{N}\right)$

これを用いると生のFFTスペクトル（周波数のパワースペクトル）は下式で表わせる。

$$\underline{P}(k)=\frac{X(k)\cdot X^{*}(k)}{N\cdot T},\quad (*;\text{複素共役})$$

今　$\underline{P}(k)=T\cdot C(k)\cdot C^{*}(k)$と置くと

$$C(k)=\frac{1}{\sqrt{N}}\sum_{j=0}^{N-1} x(j)\ \exp\left(-i\,2\pi\frac{kj}{N}\right)$$

が定義できる。本コマンドで求める複素フーリエ係数は，この$C(k)$である。結果は，3×（N－1）の時系列データの形で得られ，第1行に$C(k)$の実部，第2行に虚部，第3行には絶対値が入る。

ウィンドー（時間領域での荷重度，ラグウィンドーとも言う）をかけずに（無意識に箱型ウィンドーになっているが）パワースペクトルを求めた時，周波数の高い所で負のスペクトル値が現われることがある。これは，スペクトルウィンドー（周波数領域での荷重度）の裾が広がり過ぎて負の値を持つことによる。従って，スペクトルウィンドーを，中心（f＝0 Hz）に集中させようとすればラグウィンドーは箱型となり，一方スペクトルウィンドーの裾を広げず高さもおさえようとすれば，ラグウィンドーを滑かにしなければならない。この２つの矛盾する要求をできるだけ満足するようにラグウィンドーを決めなければならない訳である。（ラグウィンドーとスペクトルウィンドーとは互いにフーリエ変換の関係にある。）

以上の理由から，３種のウィンドーが指定できる。これは，$C(k)$を計算する際に，$x(j)$の代わりに，荷重度をかけたものを用いることを意味する。

① Hamming Window　　$x(j)\rightarrow x(j)\{0.54+0.46\sin(\frac{2\pi i}{N})\}$

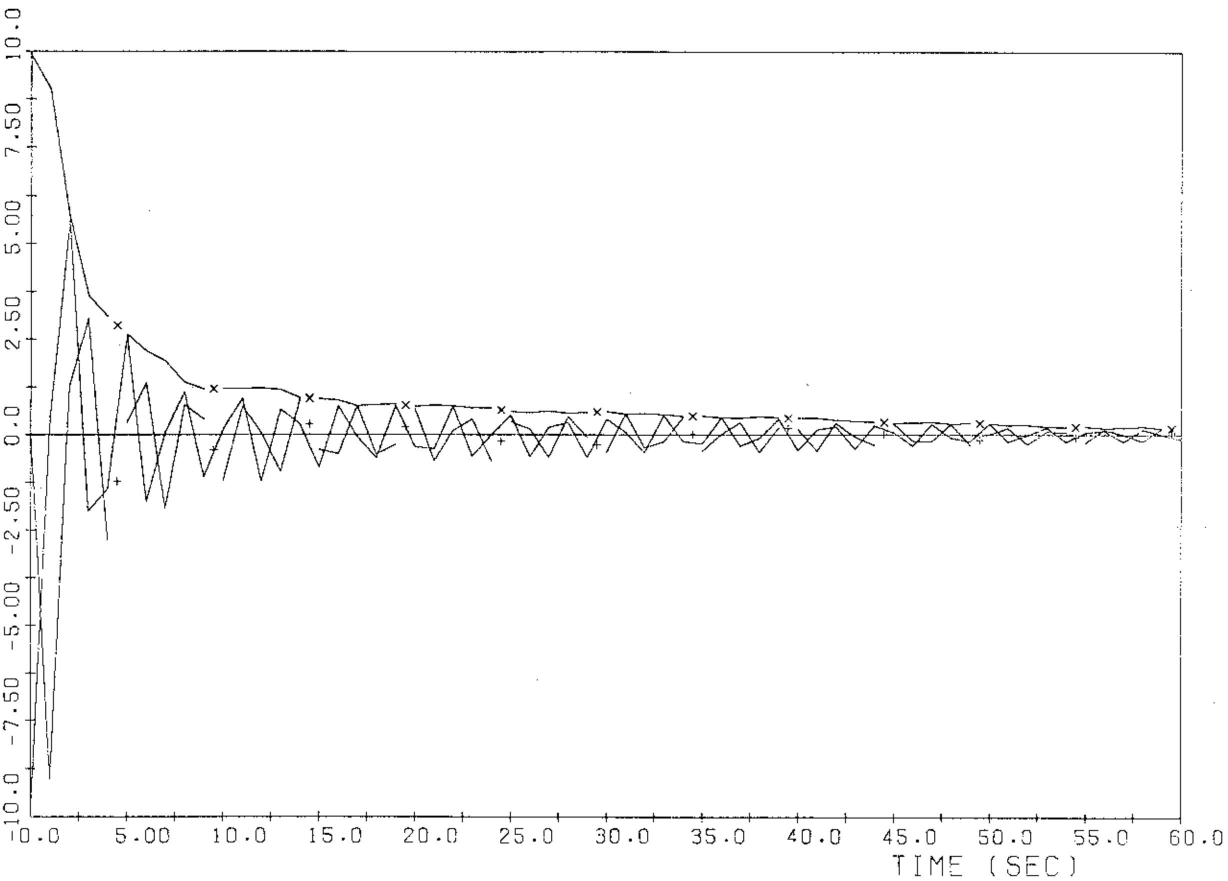
② Hanning Window　　$x(j)\rightarrow x(j)\{0.5+0.5\sin(\frac{2\pi i}{N})\}$

③ Gaussian Window　　$x(j)\rightarrow x(j)\ \exp\{-18(\frac{i-\frac{N}{2}}{N})^{2}\}$

実 行 例

〔1〕 時系列データの高速フーリエ級数変換（ハミングウインドー）の例

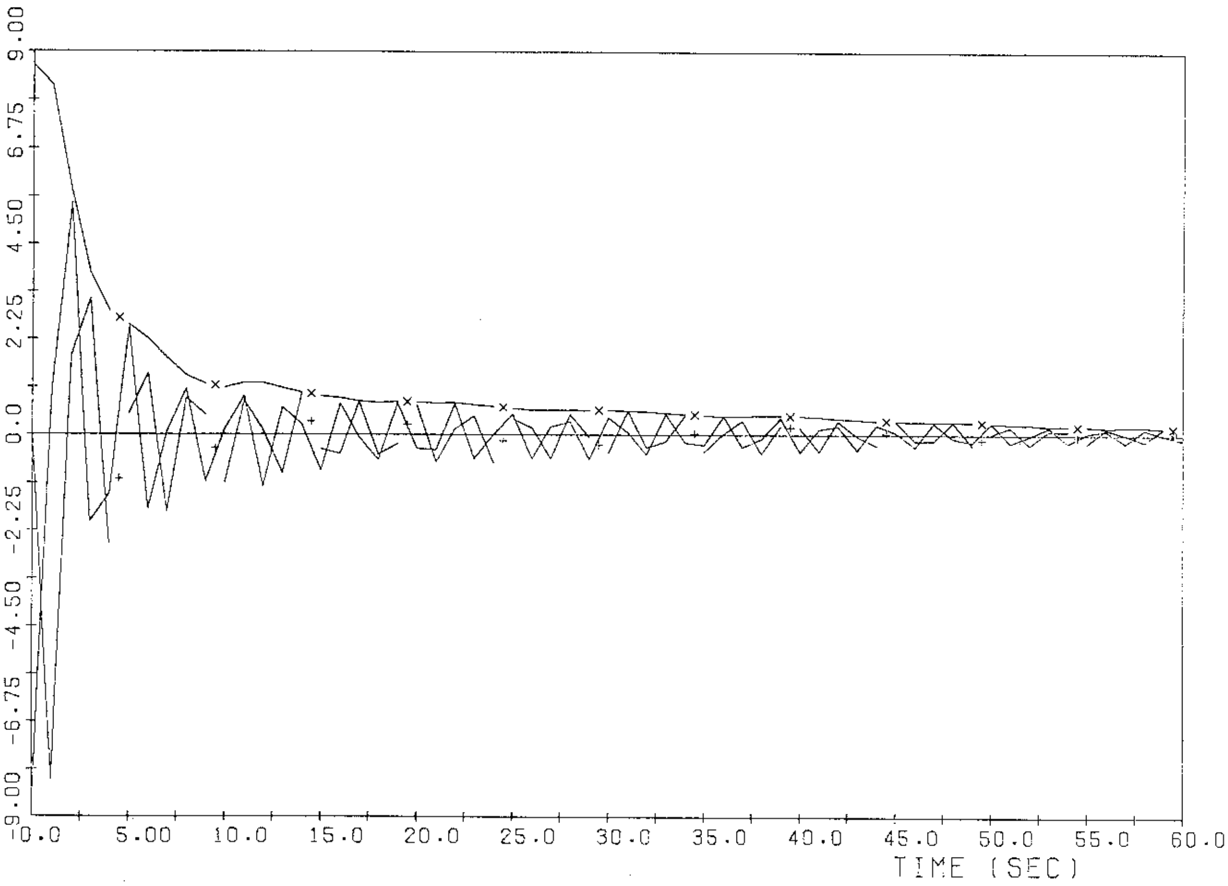
```
 /TS/ TIME-09:43:14 CPU-00:00:01
FFT
 TO EXECUTE FFT
 SPECIFY (N,M) :    2,2100
 INPUT -MODE OF ( 2,2100) TIME SERIES   A       :  F
 INPUT -FILENAME :   PIP
 ON WHICH VARIABLE ? :   1
 HOW MANY POINTS ? :   512
 STARTING POINT ? (0-1589) :    30
 USE WINDOW ? _
 (1) HAMMING WINDOW  (2) HANNING WINDOW  (3) GAUSSIAN  WINDOW
 WHICH WINDOW IS USED ? :   1
 OUTPUT-MODE OF ( 3, 511) TIME SERIES FFT       :  F
 OUTPUT-FILENAME :   FFTEND
 /TS/ TIME-09:45:32 CPU-00:00:11
```



HAMMING WINDOW FFT PLOT

〔2〕　時系列データの高速フーリエ級数変換（ハニングウインドー）の例

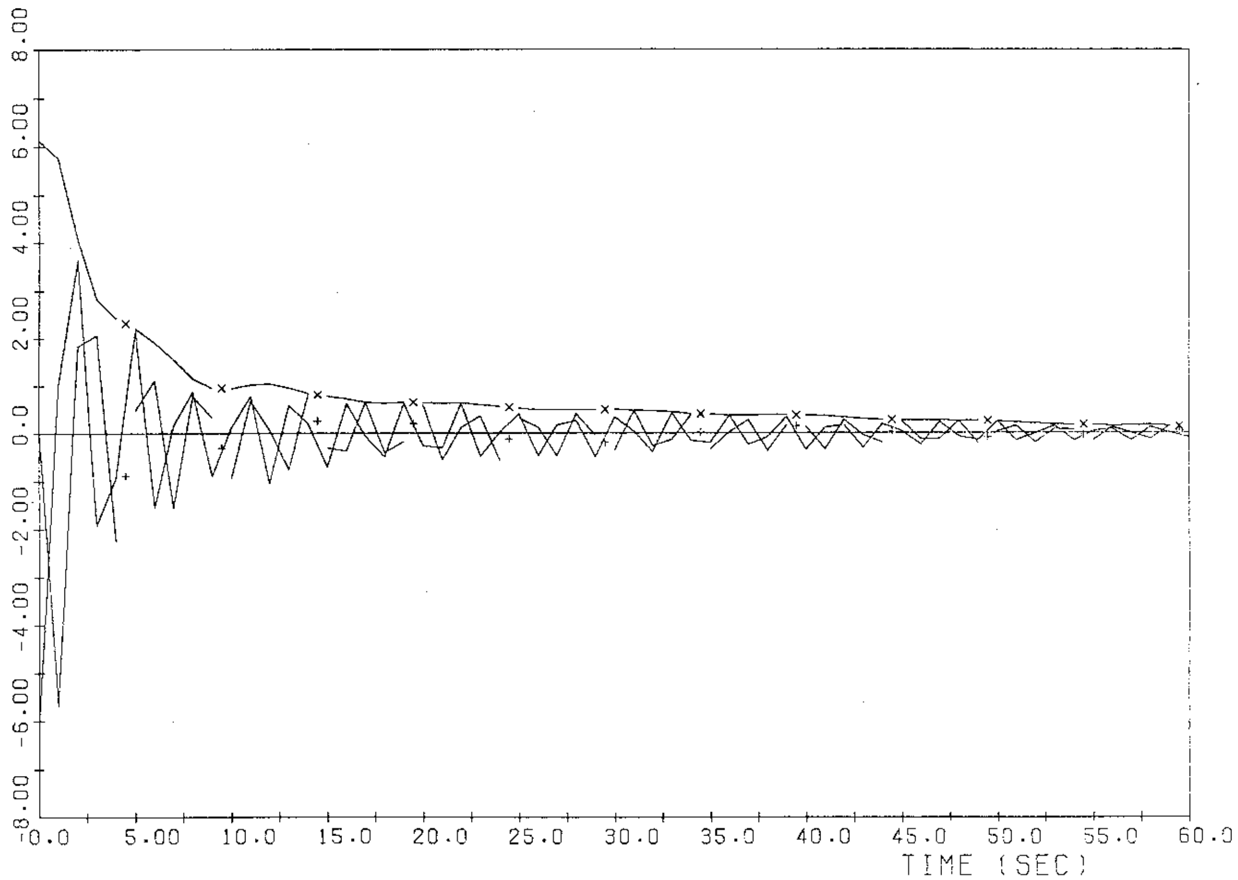
```
/TS/ TIME-09:43:17 CPU-00:00:01
FFT
 TO EXECUTE FFT
 SPECIFY (N,M) :   2,2100
 INPUT -MODE OF ( 2,2100) TIME SERIES   A       :  F
 INPUT -FILENAME :   PIP
 ON WHICH VARIABLE ? :   1
 HOW MANY POINTS ? :   512
 STARTING POINT ? (0-1589) :    30
 USE WINDOW ? _
 (1) HAMMING WINDOW  (2) HANNING WINDOW  (3) GAUSSIAN  WINDOW
 WHICH WINDOW IS USED ? :   2
 OUTPUT-MODE OF ( 3, 511) TIME SERIES FFT       :  F
 OUTPUT-FILENAME :   FFTHAN
 /TS/ TIME-09:45:34 CPU-00:00:11
```



MANNING WINDOR FFT PLOT

〔3〕 時系列データの高速フーリエ級数変換（ガウスウインドー）の例

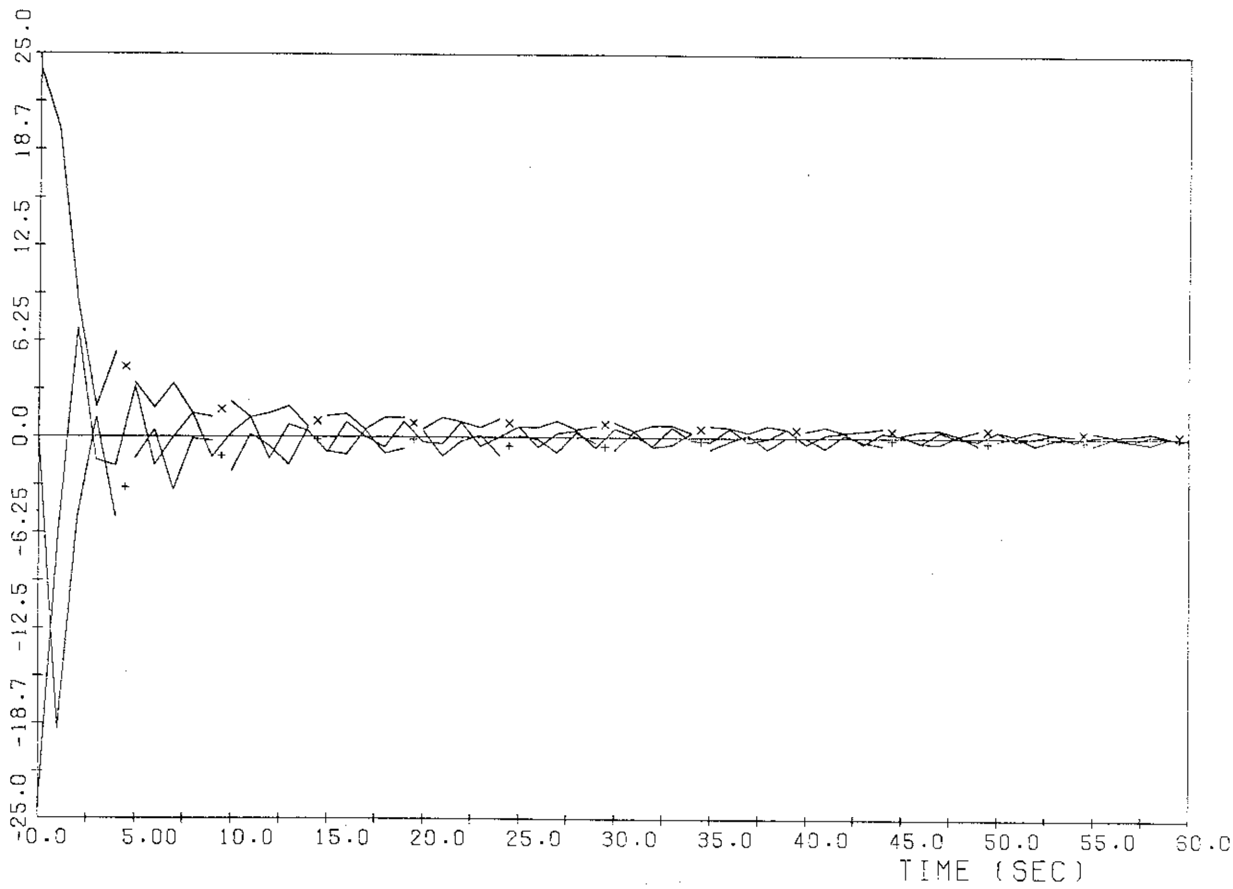
```
 /TS/ TIME-09:43:19 CPU-00:00:01
FFT
 TO EXECUTE FFT
 SPECIFY (N,M) :   2,2100
 INPUT -MODE OF ( 2,2100) TIME SERIES   A       :  F
 INPUT -FILENAME :   PIP
 ON WHICH VARIABLE ? :   1
 HOW MANY POINTS ? :   512
 STARTING POINT ? (0-1589) :    30
 USE WINDOW ? _
 (1) HAMMING WINDOW  (2) HANNING WINDOW  (3) GAUSSIAN  WINDOW
 WHICH WINDOW IS USED ? :   3
 OUTPUT-MODE OF ( 3, 511) TIME SERIES FFT       :  F
 OUTPUT-FILENAME :   FFTGAU
 /TS/ TIME-09:45:36 CPU-00:00:11
```



GAUSSIAN WINDOW FFT PLOT

〔4〕　時系列データの高速フーリエ級数変換（時間ウインドー不使用）の例

```
 /TS/ TIME-09:43:23 CPU-00:00:01
FFT
 TO EXECUTE FFT
 SPECIFY (N,M) :   2,2100
 INPUT -MODE OF ( 2,2100) TIME SERIES   A       :  F
 INPUT -FILENAME :    PIP
 ON WHICH VARIABLE ? :    1
 HOW MANY POINTS ? :    512
 STARTING POINT ? (0-1589) :    30
 USE WINDOW ?   N
 OUTPUT-MODE OF ( 3, 511) TIME SERIES FFT       :  F
 OUTPUT-FILENAME :    NOWIND
 /TS/ TIME-09:45:37 CPU-00:00:11
```



NOWIND FFT PLOT

3.10 MATサブシステムの理論概要・処理機能・利用例

Ⅰ. 行列データのハンドリング
Ⅱ. 行列データの演算
Ⅲ. 行列データの解析

行列データの一覧出力	MT. I. 1
LIST	

機　能

行列データ管理台帳（MATLIST）をリスティングし，必要が有れば指定する行列データを削除する。

実行例

〔1〕 行列データの一覧出力例

```
/MAT/ TIME-22:36:17 CPU-00:00:03
LIST
TOTAL NO. OF INSTALLED MATRICES :     9
----------------------------------
          NAME        ( N,    M)
----------------------------------
   1    A             ( 2,    3)
   2    S             ( 5,    3)
   3    FX            ( 3,    6)
   4    FY            ( 4,    2)
   5    MODEL2A       ( 6,    6)
   6    MODEL2B       ( 6,    3)
   7    M2OPTA        ( 3,    6)
   8    VS            ( 3,    8)
   9    VT            ( 6,   14)
----------------------------------
HOW MANY MATRICES ARE DELETED ? : _
/MAT/ TIME-22:40:52 CPU-00:00:10
```

〔2〕 行列データの削除例

```
 /MAT/ TIME-22:36:20 CPU-00:00:03
LIST
 TOTAL NO. OF INSTALLED MATRICES :    10
 ---------------------------------
           NAME        ( N,    M)
 ---------------------------------
    1    A             ( 2,    3)
    2    S             ( 5,    3)
    3    FX            ( 3,    6)
    4    FY            ( 4,    2)
    5    MODEL2A       ( 6,    6)
    6    MODEL2B       ( 6,    3)
    7    M2OPTA        ( 3,    6)
    8    VS            ( 3,    8)
    9    VT            ( 6,   14)
   10    ABC           ( 2,    2)
 ---------------------------------
 HOW MANY MATRICES ARE DELETED ? :   3
 SPECIFY  3-MATRICES TO BE DELETED :
1,2,10
 /MAT/ TIME-22:40:56 CPU-00:00:10

 /MAT/ TIME-22:40:57 CPU-00:00:10
LIST
 TOTAL NO. OF INSTALLED MATRICES :     7
 ---------------------------------
           NAME        ( N,    M)
 ---------------------------------
    1    FX            ( 3,    6)
    2    FY            ( 4,    2)
    3    MODEL2A       ( 6,    6)
    4    MODEL2B       ( 6,    3)
    5    M2OPTA        ( 3,    6)
    6    VS            ( 3,    8)
    7    VT            ( 6,   14)
 ---------------------------------
 HOW MANY MATRICES ARE DELETED ? : _
 /MAT/ TIME-22:41:20 CPU-00:00:18
```

行列データの入出力・修正・複写	MT. I. 2
MAT	

機　能

行列データの入力・出力・修正・複写を行う。

実行例

〔1〕 行列データの入力を行毎に行う例

```
 /MAT/ TIME-22:36:22 CPU-00:00:03
MAT
 TO INPUT MATRIX A(N,M)
 SPECIFY DIMENSION N AND M OF A :   5,3
 INPUT -MODE OF ( 5, 3) MATRIX    A        : _
 ( 1) ROW
1,6,11
 ( 2) ROW
2,7,11
 ( 3) ROW
3,8,13
 ( 4) ROW
4,9,14
 ( 5) ROW
5,10,15
 TYPE OR REVISE ?   T
 --- ( 5, 3) MATRIX    A       ---
 ( 1)   1.00000000D+00   6.00000000D+00   1.10000000D+01
 ( 2)   2.00000000D+00   7.00000000D+00   1.10000000D+01
 ( 3)   3.00000000D+00   8.00000000D+00   1.30000000D+01
 ( 4)   4.00000000D+00   9.00000000D+00   1.40000000D+01
 ( 5)   5.00000000D+00   1.00000000D+01   1.50000000D+01
 ***
 TYPE OR REVISE ?   R
 SPECIFY (I,J) :    2,11
 SPECIFY (I,J) :    2,3
   A       ( 2, 3)=   1.10000000D+01   REVISE TO    12
 END ? _
 TYPE OR REVISE ?   T
 --- ( 5, 3) MATRIX    A       ---
 ( 1)   1.00000000D+00   6.00000000D+00   1.10000000D+01
 ( 2)   2.00000000D+00   7.00000000D+00   1.20000000D+01
 ( 3)   3.00000000D+00   8.00000000D+00   1.30000000D+01
 ( 4)   4.00000000D+00   9.00000000D+00   1.40000000D+01
 ( 5)   5.00000000D+00   1.00000000D+01   1.50000000D+01
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 5, 3) MATRIX    A        :    F
 FILENAME :   S
 /MAT/ TIME-22:40:59 CPU-00:00:10
```

〔2〕 行列データの修正例

```
 /MAT/ TIME-22:36:23 CPU-00:00:03
MAT
 TO INPUT MATRIX A(N,M)
 SPECIFY DIMENSION N AND M OF A :   2,3
 INPUT -MODE OF ( 2, 3) MATRIX   A        : _
 ( 1) ROW
1,2,3
 ( 2) ROW
4,4,5
 TYPE OR REVISE ?   R
 SPECIFY (I,J) :   2,2
   A      ( 2, 2)=  4.000000000D+00  REVISE TO   5
 END ?   N
 SPECIFY (I,J) :   2,3
   A      ( 2, 3)=  5.000000000D+00  REVISE TO   6
 END ?   _
 TYPE OR REVISE ?   T
 --- ( 2, 3) MATRIX   A       ---
 ( 1)  1.000000000D+00  2.000000000D+00  3.000000000D+00
 ( 2)  4.000000000D+00  5.000000000D+00  6.000000000D+00
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 2, 3) MATRIX   A       :   F
 FILENAME :   A
 /MAT/ TIME-22:41:00 CPU-00:00:10
```

〔3〕 行列データの（i, i）要素に1を入力する例

```
 /MAT/ TIME-22:36:24 CPU-00:00:03
MAT
 TO INPUT MATRIX A(N,M)
 SPECIFY DIMENSION N AND M OF A :   2,3
 INPUT -MODE OF ( 2, 3) MATRIX   A        : I
 TYPE OR REVISE ?   T
 --- ( 2, 3) MATRIX   A       ---
 ( 1)  1.000000000D+00  0.0              0.0
 ( 2)  0.0              1.000000000D+00  0.0
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 2, 3) MATRIX   A       :   F
 FILENAME :_
 /MAT/ TIME-22:41:05 CPU-00:00:10
```

〔4〕 行列データの左上対称min（N，M）次小行列の左下三角部だけを入力し，右上三角部は対称にし，その他の要素は，0にする例

```
 /MAT/ TIME-22:36:25 CPU-00:00:03
MAT
 TO INPUT MATRIX A(N,M)
 SPECIFY DIMENSION N AND M OF A :   2,3
 INPUT -MODE OF ( 2, 3) MATRIX   A       : S
 (ONLY LOWER TRIANGULAR PART)
 ( 1) ROW
1
 ( 2) ROW
2,3
 TYPE OR REVISE ?   T
 --- ( 2, 3) MATRIX   A        ---
 ( 1)  1.00000000D+00  2.00000000D+00  0.0
 ( 2)  2.00000000D+00  3.00000000D+00  0.0
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 2, 3) MATRIX   A       :   F
 FILENAME : _
 /MAT/ TIME-22:41:06 CPU-00:00:10
```

〔5〕 行列データの画面表示例

```
 /MAT/ TIME-22:36:27 CPU-00:00:03
MAT
 TO INPUT MATRIX A(N,M)
 SPECIFY DIMENSION N AND M OF A :   2,3
 INPUT -MODE OF ( 2, 3) MATRIX   A       : F
 FILENAME :   A
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 2, 3) MATRIX A         : _
 --- ( 2, 3) MATRIX A          ---
 ( 1)  1.00000000D+00  2.00000000D+00  3.00000000D+00
 ( 2)  4.00000000D+00  5.00000000D+00  6.00000000D+00
 FILENAME : _
 /MAT/ TIME-22:41:07 CPU-00:00:10
```

〔6〕 ファイルから行列データを符号を反転して読み込む例

```
 /MAT/ TIME-22:36:30 CPU-00:00:03
MAT
 TO INPUT MATRIX A(N,M)
 SPECIFY DIMENSION N AND M OF A :   2,3
 INPUT -MODE OF ( 2, 3) MATRIX   A         : -F
 FILENAME :   A
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 2, 3) MATRIX A           : _
 --- ( 2, 3) MATRIX A          ---
 ( 1) -1.000000000D+00 -2.000000000D+00 -3.000000000D+00
 ( 2) -4.000000000D+00 -5.000000000D+00 -6.000000000D+00
 FILENAME : _
 /MAT/ TIME-22:41:09 CPU-00:00:10
```

〔7〕 ファイルから行列データを転置して読み込む例

```
 /MAT/ TIME-22:36:28 CPU-00:00:03
MAT
 TO INPUT MATRIX A(N,M)
 SPECIFY DIMENSION N AND M OF A :   3,2
 INPUT -MODE OF ( 3, 2) MATRIX   A         : F'
 FILENAME :   A
 TYPE OR REVISE ?   T
 --- ( 3, 2) MATRIX   A        ---
 ( 1)   1.000000000D+00  4.000000000D+00
 ( 2)   2.000000000D+00  5.000000000D+00
 ( 3)   3.000000000D+00  6.000000000D+00
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 3, 2) MATRIX A           : _
 --- ( 3, 2) MATRIX A          ---
 ( 1)   1.000000000D+00  4.000000000D+00
 ( 2)   2.000000000D+00  5.000000000D+00
 ( 3)   3.000000000D+00  6.000000000D+00
 FILENAME : _
 /MAT/ TIME-22:41:10 CPU-00:00:10
```

〔8〕 行列データの複写例

```
 /MAT/ TIME-22:36:31 CPU-00:00:03
MAT
 TO INPUT MATRIX A(N,M)
 SPECIFY DIMENSION N AND M OF A :   3,3
 INPUT -MODE OF ( 3, 3) MATRIX   A         : F
 FILENAME :   VVK
 TYPE OR REVISE ?_
 OUTPUT-MODE OF ( 3, 3) MATRIX VVK         : _
 --- ( 3, 3) MATRIX VVK        ---
 ( 1)  2.200000000D+01 -2.700000000D+01 -2.500000000D+01
 ( 2) -2.700000000D+01  4.200000000D+01  2.800000000D+01
 ( 3) -2.500000000D+01  2.800000000D+01  3.600000000D+01
 FILENAME :   VVKD
 /MAT/ TIME-22:41:11 CPU-00:00:10
```

行列データの分解	MT. I. 3
DCMP	

機　　能

行列データの分解を行う。

◆行方向の分割：

$$\boldsymbol{A}(n,\ m)=\begin{bmatrix}\boldsymbol{A}_1\ (n_1,\ m)\\ \boldsymbol{A}_2\ (n_2,\ m)\end{bmatrix},\quad (n_1+n_2=n)$$

◆列方向の分割：

$$\boldsymbol{A}(n,\ m)=[\boldsymbol{A}_1\ (n,\ m_1),\ \boldsymbol{A}_2\ (n,\ m_2)],\ (m_1+m_2=m)$$

実 行 例

〔1〕　行方向の分割例

```
 /MAT/ TIME-14:42:37 CPU-00:00:02
DCMP
 TO DECOMPOSE A(N,M) INTO
        (1)   (A1(N1,M))
              (A2(N2,M))
        (2)   (A1(N,M1),A2(N,M2))
 WHICH DECOMPOSITION ? :   1
 SPECIFY (N,M,N1) :   3,1,2
 INPUT -MODE OF ( 3, 1) MATRIX   A       : _
 ( 1) ROW
1
 ( 2) ROW
2
 ( 3) ROW
3
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 2, 1) MATRIX   A1      :
 --- ( 2, 1) MATRIX   A1      ---
 ( 1)   1.00000000D+00
 ( 2)   2.00000000D+00
 FILENAME : _
 OUTPUT-MODE OF ( 1, 1) MATRIX   A2      : _
 --- ( 1, 1) MATRIX   A2      ---

 ( 1)   3.00000000D+00
 FILENAME : _
 /MAT/ TIME-14:43:39 CPU-00:00:02
```

〔2〕 列方向の分割例

```
 /MAT/ TIME-22:36:34 CPU-00:00:03
DCMP
 TO DECOMPOSE A(N,M) INTO
       (1)   (A1(N1,M))
             (A2(N2,M))
       (2)   (A1(N,M1),A2(N,M2))
 WHICH DECOMPOSITION ? :   2
 SPECIFY (N,M,M1) :   1,3,1
 INPUT -MODE OF ( 1, 3) MATRIX   A          : _
 ( 1) ROW
1,2,3
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 1, 1) MATRIX   A1         : _
 --- ( 1, 1) MATRIX   A1      ---
 ( 1)  1.00000000D+00
 FILENAME : _
 OUTPUT-MODE OF ( 1, 2) MATRIX   A2         : _
 --- ( 1, 2) MATRIX   A2      ---
 ( 1)  2.00000000D+00  3.00000000D+00
 FILENAME : _
 /MAT/ TIME-22:41:16 CPU-00:00:10
```

行列データの結合	MT. I. 4
APPEND	

機　能

行列データの結合を行う。

◆行方向の結合：

$$\begin{bmatrix} \boldsymbol{A}_1(n_1,\ m) \\ \boldsymbol{A}_2(n_2,\ m) \end{bmatrix} = \boldsymbol{A}(n,\ m),\ (n_1+n_2=n)$$

◆列方向の結合：

$$[\boldsymbol{A}_1(n,\ m_1),\ \boldsymbol{A}_2(n,\ m_2)] = \boldsymbol{A}(n,\ m),\ (m_1+m_2=m)$$

実 行 例

〔1〕　行方向の結合例

```
 /MAT/ TIME-11:53:44 CPU-00:00:01
APPEND
 TO APPEND A1 AND A2 AS
      (1)   A(N,M) = (A1(N1,M))
                     (A2(N2,M))
      (2)   A(N,M) = (A1(N,M1),A2(N,M2))
 WHICH COMBINATION ? :  1
 SPECIFY (N1,N2,M) :   1,1,1
 INPUT -MODE OF ( 1, 1) MATRIX   A1      : _
 ( 1) ROW
1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 1, 1) MATRIX   A2      : _
 ( 1) ROW
2
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 2, 1) MATRIX   A       : _
 --- ( 2, 1) MATRIX   A       ---
 ( 1)  1.00000000D+00
 ( 2)  2.00000000D+00
 FILENAME : _
 /MAT/ TIME-11:54:28 CPU-00:00:02
```

〔2〕 列方向の結合例

```
 /MAT/ TIME-11:55:12 CPU-00:00:04
APPEND
 TO APPEND A1 AND A2 AS
      (1)   A(N,M) = (A1(N1,M))
                     (A2(N2,M))
      (2)   A(N,M) = (A1(N,M1),A2(N,M2))
 WHICH COMBINATION ? :   2
 SPECIFY (N,M1,M2) :   1,1,1
 INPUT -MODE OF ( 1, 1) MATRIX   A1      : _
 ( 1) ROW
1
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 1, 1) MATRIX   A2      : _
 ( 1) ROW
2
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 1, 2) MATRIX   A       : _
 --- ( 1, 2) MATRIX   A      ---
 ( 1)  1.000000000D+00  2.000000000D+00
 FILENAME : _
 /MAT/ TIME-11:55:43 CPU-00:00:05
```

行列データの並べ替え	MT. I. 5
PRMT	

機　能

行列データの並べ替えを行う。

◆行ベクトルの並べ替えを行う。

行列データ：$\boldsymbol{A}$（n，m）の各行ベクトルを${}^t\boldsymbol{a}_1$，…${}^t\boldsymbol{a}_n$とし，この行ベクトルの置換を$\{\sigma_i\}$ i＝1，…，nとすれば，$\boldsymbol{A}$＄を得る。

$$\boldsymbol{A}\$ = \begin{bmatrix} {}^t\sigma_1 \\ \vdots \\ {}^t\sigma_n \end{bmatrix}$$

◆列ベクトルの並べ替えを行う。

行列データ：$\boldsymbol{A}$（n，m）の各列ベクトルを$\boldsymbol{a}_1$，…，$\boldsymbol{a}_m$とし，この列ベクトルの置換を$\{\sigma_i\}$，i＝1，…，mとすれば，$\boldsymbol{A}$＄を得る。

$$\boldsymbol{A}\$ = [\sigma_1, \cdots, \sigma_m]$$

実 行 例

〔1〕　行ベクトルの並べ替えの例

```
/MAT/ TIME-13:22:49 CPU-00:00:01
PRMT
TO PERMUTE (1) COLUMNS OR (2) ROWS OF A(N,M)
WHICH PERMUTATION ? :   1
SPECIFY (N,M) :   1,3
INPUT -MODE OF ( 1, 3) MATRIX   A      : _
( 1) ROW
10,20,30
TYPE OR REVISE ? _
SPECIFY PERMUTATION :   3,1,2
OUTPUT-MODE OF ( 1, 3) MATRIX   A$     : _
--- ( 1, 3) MATRIX   A$     ---
( 1)  3.000000000D+01  1.000000000D+01  2.000000000D+01
FILENAME : _
/MAT/ TIME-13:23:37 CPU-00:00:02
```

〔2〕 列ベクトルの並べ替え

```
 /MAT/ TIME-13:25:40 CPU-00:00:04
PRMT
 TO PERMUTE (1) COLUMNS OR (2) ROWS OF A(N,M)
 WHICH PERMUTATION ? :   2
 SPECIFY (N,M) :   3,1
 INPUT -MODE OF ( 3, 1) MATRIX   A       : _
 ( 1) ROW
10
 ( 2) ROW
20
 ( 3) ROW
30
 TYPE OR REVISE ?__
 SPECIFY PERMUTATION :   3,1,2
 OUTPUT-MODE OF ( 3, 1) MATRIX   A$      : _
 --- ( 3, 1) MATRIX   A$     ---
 ( 1)  3.00000000D+01
 ( 2)  1.00000000D+01
 ( 3)  2.00000000D+01
 FILENAME : _
 /MAT/ TIME-13:26:09 CPU-00:00:04
```

行列データの各種演算を連続に行う	ＭＴ．Ⅱ．１
ＭＴＣＡＬ	

機　　能

以下に示す２つの行列データ同士の演算を連続操作で行う。

① 加算　$\boldsymbol{M}=\boldsymbol{A}+\boldsymbol{B}$　（ＡＤＤコマンド：ＭＴ．Ⅱ．２参照）

② 減算　$\boldsymbol{M}=\boldsymbol{A}-\boldsymbol{B}$　（ＳＵＢコマンド：ＭＴ．Ⅱ．３参照）

③ 乗算（右）　$\boldsymbol{M}=\boldsymbol{A}*\boldsymbol{B}$　（ＭＵＬコマンド：ＭＴ．Ⅱ．４参照）

④ 乗算（左）　$\boldsymbol{M}=\boldsymbol{B}*\boldsymbol{A}$　（ＭＵＬコマンド：ＭＴ．Ⅱ．４参照）

⑤ 逆行列　$\boldsymbol{M}=\boldsymbol{A}^{-1}$　（ＩＮＶコマンド：ＭＴ．Ⅱ．６参照）

⑥ 逆行列との積　$\boldsymbol{M}=\boldsymbol{A}*\boldsymbol{B}^{-1}$

⑦ スカラー倍　$\boldsymbol{M}=\alpha*\boldsymbol{A}$　（ＳＭＵＬコマンド：ＭＴ．Ⅱ．５参照）

⑧ 転置　$\boldsymbol{M}={}^{t}\boldsymbol{A}$　（ＴＲＮＳコマンド：ＭＴ．Ⅱ．７参照）

⑨ 中間結果の表示

ここで，$\boldsymbol{M}$は中間結果を表す。$\boldsymbol{A}$は被演算行列であり前回の演算結果が使用される。$\boldsymbol{B}$は演算行列であり，新たに入力される行列である。また，$\alpha\in\boldsymbol{R}$である。

このコマンドが実行されたときに，最初に中間結果行列データ$\boldsymbol{M}$の入力が要求される。

実 行 例

〔1〕　行列データの各種演算を連続に行った例

```
/MAT/ TIME-15:03:28 CPU-00:00:02
MICAL
 ORIGINAL MATRIX SPECIFY (N,M) :  3,3
 INPUT -MODE OF ( 3, 3) MATRIX   A       :  _
 ( 1) ROW
3,4,5
 ( 2) ROW
2,3,4
 ( 3) ROW
1,2,3
 TYPE OR REVISE ? _
 MATCAL SEL 1.+B, 2,-B, 3.*B, 4.INV, 5.*INV B, 6.S*, 7.B*, 8.TRN, 9.DIS, 10.END
  KEY-IN SELECT NO. :  1
 INPUT -MODE OF ( 3, 3) MATRIX   B       : 1
 TYPE OR REVISE ? _
 MATCAL SEL 1.+B, 2,-B, 3.*B, 4.INV, 5.*INV B, 6.S*, 7.B*, 8.TRN, 9.DIS, 10.END
  KEY-IN SELECT NO. :  9
 OUTPUT-MODE OF ( 3, 3) MATRIX RESULT-V : _
 --- ( 3, 3) MATRIX RESULT-V ---
 ( 1)  4.000000000D+00  4.000000000D+00  5.000000000D+00
 ( 2)  2.000000000D+00  4.000000000D+00  4.000000000D+00
 ( 3)  1.000000000D+00  2.000000000D+00  4.000000000D+00
 FILENAME : _

 MATCAL SEL 1.+B, 2,-B, 3.*B, 4.INV, 5.*INV B, 6.S*, 7.B*, 8.TRN, 9.DIS, 10.END
  KEY-IN SELECT NO. :  2
 INPUT -MODE OF ( 3, 3) MATRIX   B       : _
 ( 1) ROW
1,2,3
 ( 2) ROW
2,3,4
 ( 3) ROW
1,0,1
 TYPE OR REVISE ? _
 MATCAL SEL 1.+B, 2,-B, 3.*B, 4.INV, 5.*INV B, 6.S*, 7.B*, 8.TRN, 9.DIS, 10.END
  KEY-IN SELECT NO. :  9
 OUTPUT-MODE OF ( 3, 3) MATRIX RESULT-V : _
 --- ( 3, 3) MATRIX RESULT-V ---
 ( 1)  3.000000000D+00  2.000000000D+00  2.000000000D+00
 ( 2)  0.0              1.000000000D+00  0.0
 ( 3)  0.0              2.000000000D+00  3.000000000D+00
 FILENAME : _
 MATCAL SEL 1.+B, 2,-B, 3.*B, 4.INV, 5.*INV B, 6.S*, 7.B*, 8.TRN, 9.DIS, 10.END
  KEY-IN SELECT NO. :  3
 SPECIFY (M,L) 3,3
 INPUT -MODE OF ( 3, 3) MATRIX   B       : _
 ( 1) ROW
 ***
1,0,2
 ( 2) ROW
2,1,1
 ( 3) ROW
3,3,1
 TYPE OR REVISE ? _
 MATCAL SEL 1.+B, 2,-B, 3.*B, 4.INV, 5.*INV B, 6.S*, 7.B*, 8.TRN, 9.DIS, 10.END
  KEY-IN SELECT NO. :  9
 OUTPUT-MODE OF ( 3, 3) MATRIX RESULT-V : _
 --- ( 3, 3) MATRIX RESULT-V ---
 ( 1)  1.300000000D+01  8.000000000D+00  1.000000000D+01
 ( 2)  2.000000000D+00  1.000000000D+00  1.000000000D+00
 ( 3)  1.300000000D+01  1.100000000D+01  5.000000000D+00
 FILENAME : _
 MATCAL SEL 1.+B, 2,-B, 3.*B, 4.INV, 5.*INV B, 6.S*, 7.B*, 8.TRN, 9.DIS, 10.END
  KEY-IN SELECT NO. :  4
 EPS =  1D-10
 MATCAL SEL 1.+B, 2,-B, 3.*B, 4.INV, 5.*INV B, 6.S*, 7.B*, 8.TRN, 9.DIS, 10.END
```

```
 KEY-IN SELECT NO. :  9
OUTPUT-MODE OF ( 3, 3) MATRIX RESULT-V : _
--- ( 3, 3) MATRIX RESULT-V ---
( 1) -1.66666667D-01  1.94444444D+00 -5.55555556D-02
( 2)  8.33333333D-02 -1.80555556D+00  1.94444444D-01
***
( 3)  2.50000000D-01 -1.08333333D+00 -8.33333333D-02
FILENAME : _
MATCAL SEL 1.+B, 2,-B, 3.*B, 4.INV, 5.*INV B, 6.S*, 7.B*, 8.TRN, 9.DIS, 10.END
 KEY-IN SELECT NO. :  5
 SPECIFY (M,L)  3,3
INPUT -MODE OF ( 3, 3) MATRIX   B       : _
( 1) ROW
2,2,1
( 2) ROW
3,2,1
( 3) ROW
2,2,3
TYPE OR REVISE ? _
EPS =  1D-10
MATCAL SEL 1.+B, 2,-B, 3.*B, 4.INV, 5.*INV B, 6.S*, 7.B*, 8.TRN, 9.DIS, 10.END
 KEY-IN SELECT NO. :  9
OUTPUT-MODE OF ( 3, 3) MATRIX RESULT-V : _
--- ( 3, 3) MATRIX RESULT-V ---
( 1)  3.59722222D+00 -2.11111111D+00 -5.13888889D-01
( 2) -3.34027778D+00  1.88888889D+00  5.48611111D-01
( 3) -2.10416667D+00  1.33333333D+00  2.29166667D-01
FILENAME : _
MATCAL SEL 1.+B, 2,-B, 3.*B, 4.INV, 5.*INV B, 6.S*, 7.B*, 8.TRN, 9.DIS, 10.END
***
 KEY-IN SELECT NO. :  6
 SCALAR =     3
MATCAL SEL 1.+B, 2,-B, 3.*B, 4.INV, 5.*INV B, 6.S*, 7.B*, 8.TRN, 9.DIS, 10.END
 KEY-IN SELECT NO. :  9
OUTPUT-MODE OF ( 3, 3) MATRIX RESULT-V :
--- ( 3, 3) MATRIX RESULT-V ---
( 1)  1.07916667D+01 -6.33333333D+00 -1.54166667D+00
( 2) -1.00208333D+01  5.66666667D+00  1.64583333D+00
( 3) -6.31250000D+00  4.00000000D+00  6.87500000D-01
FILENAME : _
MATCAL SEL 1.+B, 2,-B, 3.*B, 4.INV, 5.*INV B, 6.S*, 7.B*, 8.TRN, 9.DIS, 10.END
 KEY-IN SELECT NO. :  7
SPECIFY (L,N) 3,3

INPUT -MODE OF ( 3, 3) MATRIX   B       : _
( 1) ROW
2,3,1
( 2) ROW
2,2,3
( 3) ROW
2,-1,-3
TYPE OR REVISE ? _
MATCAL SEL 1.+B, 2,-B, 3.*B, 4.INV, 5.*INV B, 6.S*, 7.B*, 8.TRN, 9.DIS, 10.END
***
 KEY-IN SELECT NO. :  9
OUTPUT-MODE OF ( 3, 3) MATRIX RESULT-V : _
--- ( 3, 3) MATRIX RESULT-V ---
( 1) -1.47916667D+01  8.33333333D+00  2.54166667D+00
( 2) -1.73958333D+01  1.06666667D+01  2.27083333D+00
( 3)  5.05416667D+01 -3.03333333D+01 -6.79166667D+00
FILENAME : _
MATCAL SEL 1.+B, 2,-B, 3.*B, 4.INV, 5.*INV B, 6.S*, 7.B*, 8.TRN, 9.DIS, 10.END
 KEY-IN SELECT NO. :  8
MATCAL SEL 1.+B, 2,-B, 3.*B, 4.INV, 5.*INV B, 6.S*, 7.B*, 8.TRN, 9.DIS, 10.END
 KEY-IN SELECT NO. :  9
OUTPUT-MODE OF ( 3, 3) MATRIX RESULT-V : _
--- ( 3, 3) MATRIX RESULT-V ---
( 1) -1.47916667D+01 -1.73958333D+01  5.05416667D+01
( 2)  8.33333333D+00  1.06666667D+01 -3.03333333D+01
( 3)  2.54166667D+00  2.27083333D+00 -6.79166667D+00
```

```
FILENAME : _
MATCAL SEL 1.+B, 2,-B, 3.*B, 4.INV, 5.*INV B, 6.S*, 7.B*, 8.TRN, 9.DIS, 10.END
 KEY-IN SELECT NO. :  10
OUTPUT-MODE OF ( 3, 3) MATRIX RESULT-M : _
--- ( 3, 3) MATRIX RESULT-M ---
( 1) -1.47916667D+01 -1.73958333D+01  5.05416667D+01
( 2)  8.33333333D+00  1.06666667D+01 -3.03333333D+01
***
( 3)  2.54166667D+00  2.27083333D+00 -6.79166667D+00
FILENAME : _
/MAT/ TIME-15:10:31 CPU-00:00:05
```

行列データ同士の加算	MT. II. 2
ADD	

機　　能

行列データ $\boldsymbol{A}(n, m) = \{a_{ij}\}$, $\boldsymbol{B}(n, m) = \{b_{ij}\}$に対して

$$\boldsymbol{A} + \boldsymbol{B} = \{a_{ij} + b_{ij}\} \quad (i = 1, \cdots, n, \ j = 1, \cdots, m)$$

を計算する。

実 行 例

〔1〕　行列データ同士の加算例

```
/MAT/ TIME-22:36:39 CPU-00:00:03
ADD
 TO CALCULATE A(N,M)+B(N,M)
 SPECIFY (N,M) :    1,2
 INPUT -MODE OF ( 1, 2) MATRIX    A        : _
 ( 1) ROW
1,2
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 1, 2) MATRIX    B        : _
 ( 1) ROW
3,4
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 1, 2) MATRIX  A+B        : _
 --- ( 1, 2) MATRIX  A+B      ---
 ( 1)   4.00000000D+00  6.00000000D+00
 FILENAME : _
 /MAT/ TIME-22:41:18 CPU-00:00:10
```

行列データ同士の減算	MT. II. 3
SUB	

機　能

行列データ $\boldsymbol{A}(n, m) = \{a_{ij}\}$, $\boldsymbol{B}(n, m) = \{b_{ij}\}$ に対して

$$\boldsymbol{A} - \boldsymbol{B} = \{a_{ij} - b_{ij}\} \quad (i = 1, \cdots, n, \ j = 1, \cdots, m)$$

を計算する。

実 行 例

〔1〕　行列データ同士の減算例

```
 /MAT/ TIME-22:36:41 CPU-00:00:03
SUB
 TO CALCULATE A(N,M)-B(N,M)
 SPECIFY (N,M) :   1,2
 INPUT -MODE OF ( 1, 2) MATRIX   A        : _
 ( 1) ROW
3,4
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 1, 2) MATRIX   B        : _
 ( 1) ROW
1,2
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 1, 2) MATRIX  A-B       : _
 --- ( 1, 2) MATRIX  A-B      ---
 ( 1)  2.00000000D+00  2.00000000D+00
 FILENAME : _
 /MAT/ TIME-22:41:20 CPU-00:00:10
```

行列データ同士の乗算	MT. Ⅱ. 4
MUL	

機　　能

行列データ $\boldsymbol{A}(n, m) = \{a_{ij}\}$，$\boldsymbol{B}(m, r) = \{b_{ij}\}$ に対して

$$\boldsymbol{A} \cdot \boldsymbol{B} = \{\sum_{k=1}^{m} a_{ik} b_{kj}\} \quad (i = 1, \cdots, n, j = 1, \cdots, r)$$

を計算する

実 行 例

〔1〕　行列データ同士の乗算例

```
 /MAT/ TIME-22:36:42 CPU-00:00:03
MUL
 TO CALCULATE A(N,M)*B(M,L)
 SPECIFY (N,M,L) :   2,1,2
 INPUT -MODE OF ( 2, 1) MATRIX   A        : _
 ( 1) ROW
1
 ( 2) ROW
2
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 1, 2) MATRIX   B        : _
 ( 1) ROW
1,-2
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 2, 2) MATRIX  A*B       : _
 --- ( 2, 2) MATRIX  A*B     ---
 ( 1)  1.000000000D+00 -2.000000000D+00
 ( 2)  2.000000000D+00 -4.000000000D+00
 FILENAME : _
 /MAT/ TIME-22:41:21 CPU-00:00:10
```

行列データのスカラー倍	MT. II. 5
SMUL	

機　能

行列データ$\boldsymbol{A}$(n, m) = $\{a_{ij}\}$, スカラー量αに対して

$$\alpha \boldsymbol{A} = \{\alpha a_{ij}\} \quad (i = 1, \cdots, n, \ j = 1, \cdots, m)$$

を計算する。

実 行 例

〔1〕　行列データのスカラー倍の例

```
 /MAT/ TIME-22:36:43 CPU-00:00:03
SMUL
 TO CALCULATE SCALAR*A(N,M)
 SPECIFY (N,M) :   2,3
 SCALAR =   -2
 INPUT -MODE OF ( 2, 3) MATRIX   A       : F
 FILENAME :   A
 TYPE OR REVISE ?   T
 --- ( 2, 3) MATRIX   A      ---
 ( 1)  1.000000000D+00  2.000000000D+00  3.000000000D+00
 ( 2)  4.000000000D+00  5.000000000D+00  6.000000000D+00
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 2, 3) MATRIX  S*A      : _
 --- ( 2, 3) MATRIX  S*A      ---
 ( 1) -2.000000000D+00 -4.000000000D+00 -6.000000000D+00
 ( 2) -8.000000000D+00 -1.000000000D+01 -1.200000000D+01
 FILENAME : _
 /MAT/ TIME-22:41:22 CPU-00:00:11
```

行列データの（擬似）逆行列の算出	MT. II. 6
INV	

機　能

Moore－Penrose の擬似逆行列を求める。

理論概要

行列 $\boldsymbol{A} \in \mathbf{R}^{n\times n}$ に対し，次の特異値分解を考える。

$$\boldsymbol{A} = \boldsymbol{U} \cdot \mathrm{diag}\{\sigma_1 \cdots \sigma_m\}\, {}^t\boldsymbol{V}$$

$${}^t\boldsymbol{U}\boldsymbol{U} = \boldsymbol{V}\,{}^t\boldsymbol{V} = \boldsymbol{I}_m$$

$$\{\sigma_i\} \ (i=1,\ 2,\ \cdots,\ m) \ (\sigma_i \in \mathbf{R}),\ \boldsymbol{U} \in \mathbf{R}^{n\times m},\ \boldsymbol{V} \in \mathbf{R}^{m\times m}$$

このとき，十分に小さい $\varepsilon > 0$ に対して，$\boldsymbol{A}$ の擬似逆行列 $\boldsymbol{A}^+ \in \mathbf{R}^{m\times n}$ が次式で与えられる。

$$\sigma_1 \geqq \sigma_2 \geqq \cdots \geqq \sigma_k > \varepsilon > \sigma_{k+1} \geqq \cdots \geqq \sigma_m$$

を満たす整数 k が定まる。今，$\boldsymbol{U} \in \mathbf{R}^{n\times m}$ と $\boldsymbol{V} \in \mathbf{R}^{m\times n}$ を

$$\boldsymbol{U} = [\underset{k}{\boldsymbol{U}_1} \vdots \underset{m-k}{\boldsymbol{U}_2}],\ \boldsymbol{V} = [\underset{k}{\boldsymbol{V}_1} \vdots \underset{m-k}{\boldsymbol{V}_2}]$$

の様に分割するとき，行列 $\boldsymbol{A}$ に対応する線形写像に関して，

$$\boldsymbol{A}^+ = \boldsymbol{V} \cdot \mathrm{diag}\{\bar{\sigma}_1 \cdots \bar{\sigma}_m\}\,{}^t\boldsymbol{U}$$

$$\bar{\sigma}_i \equiv \begin{cases} \sigma_{i^{-1}} & (\sigma_i \geqq \varepsilon) \\ 0 & (\sigma_i < \varepsilon) \end{cases}$$

尚，$\boldsymbol{A}^+$は，以下の条件を満足している。

$$\boldsymbol{A}\boldsymbol{A}^+\boldsymbol{A} = \boldsymbol{A}$$

$$\boldsymbol{A}^+\boldsymbol{A}\boldsymbol{A}^+ = \boldsymbol{A}^+$$

$\boldsymbol{A}\boldsymbol{A}^+$, $\boldsymbol{A}^+\boldsymbol{A}$は対称

実 行 例

〔1〕　擬似逆行列の算出例

```
/MAT/ TIME-22:36:45 CPU-00:00:03
INV
 TO CALCULATE PSEUDO INVERSE A-(M,N) OF A(N,M)
 SPECIFY (N,M) :   5,3
 INPUT -MODE OF ( 5, 3) MATRIX   A       : F
 FILENAME :   S
 TYPE OR REVISE ? _
 EPS =   1.0D-10
 OUTPUT-MODE OF ( 3, 5) MATRIX   A-      : _
 --- ( 3, 5) MATRIX   A-      ---
 ( 1) -2.46666667D-01 -1.33333333D-01 -2.00000000D-02  9.33333333D-02
 ( 2) -6.66666667D-02 -3.33333333D-02  9.54097912D-18  3.33333333D-02
 ( 3)  1.13333333D-01  6.66666667D-02  2.00000000D-02 -2.66666667D-02
 ( 1)  2.06666667D-01
 ( 2)  6.66666667D-02
 ( 3) -7.33333333D-02
 FILENAME : _
/MAT/ TIME-22:41:24 CPU-00:00:11
```

行列データの転置行列の算出	MT. II. 7
TRNS	

機　能

行列データ $\boldsymbol{A}(n, m) = \{a_{ij}\}$のとき

$^t\boldsymbol{A} = \{a_{ji}\}$　$(i = 1, \cdots, n,\ j = 1, \cdots, m)$

を計算する。

実 行 例

〔1〕　転置行列の算出例

```
 /MAT/ TIME-22:36:46 CPU-00:00:03
TRNS
 TO TRANSPOSE A(N,M)
 SPECIFY (N,M) :   1,2
 INPUT -MODE OF ( 1, 2) MATRIX   A       : _
 ( 1) ROW
1,2
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 2, 1) MATRIX   A'      : _
 --- ( 2, 1) MATRIX   A'      ---
 ( 1)  1.00000000D+00
 ( 2)  2.00000000D+00
 FILENAME :_
 /MAT/ TIME-22:41:25 CPU-00:00:11
```

多項式関数値の算出	MT. Ⅱ. 8
FUNC	

機　　能

行列データ $\boldsymbol{A}(n, m) = \{a_{ij}\}$ と多項式 P(s)

$$P(s) = \begin{cases} s^m + C_1 s^{m-1} + \cdots + C_{m-1}s + C_m & \text{または} \\ (s-\lambda_1)(s-\lambda_2)\cdots(s-\lambda_m) & \end{cases}$$

に対し，$P(\boldsymbol{A}) \in \mathbf{R}^{n\times n}$ を計算する。

実 行 例

〔1〕　多項式関数値の算出（係数入力）例

```
 /MAT/ TIME-22:36:47 CPU-00:00:03
FUNC
 TO CALCULATE P(A(N,N)),
 WHERE (1) P(S)=S**M+C(1)*S**(M-1)+...+C(M-1)*S+C(M)
       (2) P(S)=(S-R(1))*(S-R(2))*...*(S-R(M))
 WHICH TYPE ? :   1
 SPECIFY M :   2
 SPECIFY COEFFICIENTS
 INPUT -MODE OF ( 1, 2) MATRIX  COEF     : _
 ( 1) ROW
2,1
 TYPE OR REVISE ? _
 SPECIFY N :   1
 INPUT -MODE OF ( 1, 1) MATRIX   A       : _
 ( 1) ROW
5
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 1, 1) MATRIX  P(A)     : _
 --- ( 1, 1) MATRIX  P(A)    ---
 ( 1)  3.60000000D+01
 FILENAME : _
 /MAT/ TIME-22:41:26 CPU-00:00:11
```

〔2〕 多項式関数値の算出（根入力）例

```
/MAT/ TIME-22:41:27 CPU-00:00:11
FUNC
TO CALCULATE P(A(N,N)),
WHERE (1) P(S)=S**M+C(1)*S**(M-1)+...+C(M-1)*S+C(M)
      (2) P(S)=(S-R(1))*(S-R(2))*...*(S-R(M))
WHICH TYPE ? :   2
SPECIFY M :   2
SPECIFY ROOTS
INPUT -MODE OF ( 2, 2) MATRIX ROOT      : _
( 1) ROW
1,1
( 2) ROW
1,-1
TYPE OR REVISE ? _
SPECIFY N :   1
INPUT -MODE OF ( 1, 1) MATRIX   A       : _
( 1) ROW
5
TYPE OR REVISE ? _
OUTPUT-MODE OF ( 1, 1) MATRIX  P(A)     : _
--- ( 1, 1) MATRIX  P(A)    ---
( 1)  1.70000000D+01
FILENAME : _
/MAT/ TIME-22:36:48 CPU-00:00:03
```

指数関数値の算出	MT. II. 9
EXP	

機　能

行列データ$\boldsymbol{A}$(n, m) = {a_{ij}}とサンプル周期Tに対してexp($\boldsymbol{A}$T)を計算する。

$$\exp(\boldsymbol{A}T) = \boldsymbol{I}_n + \boldsymbol{A}T + \frac{1}{2!}(\boldsymbol{A}T)^2 + \cdots = \sum_{k=1}^{\infty}\frac{1}{k!}(\boldsymbol{A}T)^k$$

実 行 例

〔1〕　指数関数値の算出例

```
/MAT/ TIME-22:36:50 CPU-00:00:03
EXP
 TO CALCULATE EXP(A(N,N)*DELT)
 SPECIF=Y :   1
 INPUT -MODE OF ( 1, 1) MATRIX   A       : I
 TYPE OR REVISE ? _
 DELT =    1
 EPS  =    1.0D-10
 OUTPUT-MODE OF ( 1, 1) MATRIX EXP(A)    : _
 --- ( 1, 1) MATRIX EXP(A)    ---
 ( 1)   2.71828183D+00
 FILENAME : _
 /MAT/ TIME-22:41:28 CPU-00:00:11
```

指数関数の積分値の算出	MT. Ⅱ. 10
IEXP	

機　能

行列データ $\boldsymbol{A}$(n, m) = {a_{ij}} とサンプル周期Tに対して $\int_0^T \exp(\boldsymbol{A}\tau)\,d\tau$ を計算する。

$$\int_0^T \exp(\boldsymbol{A}\tau)\,d\tau = T\left(\boldsymbol{I}_n + \frac{1}{2!}\boldsymbol{A}T + \frac{1}{3!}(\boldsymbol{A}T)^2 + \cdots\right) = \sum_{k=1}^{\infty}\frac{1}{k!}(\boldsymbol{A}T)^{k-1}T$$

実 行 例

〔1〕　指数関数値の積分値の算出例

```
/MAT/ TIME-22:36:51 CPU-00:00:03
IEXP
TO CALCULATE INTEGRAL OF EXP(A(N,N)*T) IN (0,DELT)
SPECIFY N :   1
INPUT -MODE OF ( 1, 1) MATRIX    A       : I
TYPE OR REVISE ? _
DELT =    1
EPS  =    1.0D-10
OUTPUT-MODE OF ( 1, 1) MATRIX IEXP(A)   : _
--- ( 1, 1) MATRIX IEXP(A)   ---
( 1)   1.71828183D+00
FILENAME :_
/MAT/ TIME-22:41:30 CPU-00:00:11
```

リゾルベント行列の算出	MT. Ⅱ. 11
RSLV	

機　　能

行列データ $\boldsymbol{A}$（n，n）＝｛a_{ij}｝に対して次のリゾルベント行列を計算する。

$$(s\boldsymbol{I}-\boldsymbol{A})^{-1}=\frac{\boldsymbol{P}(s)}{Q(s)}$$

但し，$\boldsymbol{P}(s)=\boldsymbol{P}_1 s^{n-1}+\boldsymbol{P}_2 s^{n-2}+\cdots+\boldsymbol{P}_{n-1}s+\boldsymbol{P}_n$

$Q(s)=s^n+q_1 s^{n-1}+\cdots+q_{n-1}s+q_n$

ここで，$\boldsymbol{P}_1=\boldsymbol{I}_n$ とおくとき，次式を順次計算（k＝1，…，n）してq_k，$\boldsymbol{P}_k$ を求める。

$q_k=-\mathrm{tr}\ (\boldsymbol{A}\boldsymbol{P}_k)/k,\ \boldsymbol{P}_{k+1}=\boldsymbol{A}\boldsymbol{P}_k+q_k\boldsymbol{I}_n$

実 行 例

〔1〕　リゾルベント行列算出例

```
 /MAT/ TIME-22:36:52 CPU-00:00:04
RSLV
 TO CALCULATE RESOLVENT MATRIX P(S)/Q(S) OF A(N,N),
 WHERE P(S)=S**(N-1)*I+S**(N-2)*P(2)+...+S*P(N-1)+P(N)
 AND   Q(S)=S**N+Q(1)*S**(N-1)+...+Q(N-1)*S+Q(N)
 SPECIFY N :   2
 INPUT -MODE OF ( 2, 2) MATRIX   A        : _
 ( 1) ROW
1,2
 ( 2) ROW
3,4
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 2, 2) MATRIX  P2        : _
 --- ( 2, 2) MATRIX  P2      ---
 ( 1) -4.00000000D+00  2.00000000D+00
 ( 2)  3.00000000D+00 -1.00000000D+00
 FILENAME : _
 OUTPUT-MODE OF ( 2, 2) MATRIX  P3        : _
 --- ( 2, 2) MATRIX  P3      ---
 ( 1)  0.0              0.0
 ( 2)  0.0              0.0
 FILENAME : _
 OUTPUT-MODE OF ( 1, 2) MATRIX   Q        : _
 --- ( 1, 2) MATRIX   Q      ---
 ( 1) -5.00000000D+00 -2.00000000D+00
 FILENAME : _
 /MAT/ TIME-22:41:31 CPU-00:00:11
```

リヤプノフ方程式の求解	MT. II. 12
LYP	

機　能

次のリヤプノフ方程式

$\boldsymbol{XA} + {}^t\boldsymbol{AX} = -\boldsymbol{Q}$　　連続系

$\boldsymbol{X} = {}^t\boldsymbol{AXA} + \boldsymbol{Q}$　　離散系

の解$\boldsymbol{X} \in \mathbf{R}^{n\times n}$を求める。但し，$\boldsymbol{A}$，$\boldsymbol{Q} \in \mathbf{R}^{n\times n}$ $(\boldsymbol{Q} \geqq 0)$

理論概要

◆連続系 Lyapunov 方程式（$\boldsymbol{XA} + {}^t\boldsymbol{AX} = -\boldsymbol{Q}$）の場合

$\boldsymbol{A}$ の固有値と固有ベクトルを$\{\lambda_i\}_{i=1,\cdots,n}$，$\{\boldsymbol{v}_i\}_{i=1,\cdots,n}$ $(\lambda_i \in \mathbf{C}, \boldsymbol{v}_i \in \mathbf{C}^n)$ で表わすとき，次の仮定をおく。

1. $\mathrm{Re}(\lambda_i) < 0$　　$(i = 1, \cdots, n)$
2. $\{\boldsymbol{v}_i\}$ $i = 1, \cdots, n$　一次独立

このとき，解は次式で求められる。

$$\boldsymbol{X} = [\boldsymbol{u}_1 \cdots \boldsymbol{u}_n][\boldsymbol{v}_1 \cdots \boldsymbol{v}_n]^{-1}$$

ここで，$\{\boldsymbol{u}_i\}$ $i = 1, \cdots, n$ は次式で与えられる。

$$\left\{\begin{array}{l} \lambda_i \text{が実数のとき：} \boldsymbol{u}_i = \boldsymbol{U}_i\boldsymbol{Q}\boldsymbol{v}_i \\ \qquad\qquad\qquad \boldsymbol{U}_i = ((-{}^t\boldsymbol{A}) - \lambda_i\boldsymbol{I}_n)^{-1} \\ \lambda_i \text{が実数でないとき：} \lambda_i = \lambda_{iR} + j\lambda_{iI},\ \lambda_{i+1} = \lambda_{iR} - j\lambda_{iI} \end{array}\right.$$

$$\begin{bmatrix} \boldsymbol{u}_i \\ \boldsymbol{u}_{i+1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{U}_i & -\boldsymbol{U}_{i+1} \\ \boldsymbol{U}_{i+1} & \boldsymbol{U}_i \end{bmatrix} \begin{bmatrix} \boldsymbol{Q}\boldsymbol{v}_{iR} \\ \boldsymbol{Q}\boldsymbol{v}_{iI} \end{bmatrix}$$

$$\left\{\begin{array}{l} \boldsymbol{U}_i = \{((-{}^t\boldsymbol{A}) - \lambda_{iR}\boldsymbol{I}_n)^2 + \lambda_{iI}^2\boldsymbol{I}_n\}^{-1}((-{}^t\boldsymbol{A}) - \lambda_{iR}\boldsymbol{I}_n) \\ \boldsymbol{U}_{i+1} = \{((-{}^t\boldsymbol{A}) - \lambda_{iR}\boldsymbol{I}_n)^2 + \lambda_{iI}^2\boldsymbol{I}_n\}^{-1}(\lambda_{iI}\boldsymbol{I}_n) \end{array}\right.$$

$$(\boldsymbol{v}_i = \boldsymbol{v}_{iR} + j\boldsymbol{v}_{iI})$$

◆離散系 Lyapunov 方程式（$\boldsymbol{X} = {}^t\boldsymbol{AXA} + \boldsymbol{Q}$）の場合

$\boldsymbol{A}$ の固有値と固有ベクトルを$\{\lambda_i\}_{i=1,\cdots,n}$，$\{\boldsymbol{v}_i\}_{i=1,\cdots,n}$，$(\lambda_i \in \mathbf{C}, \boldsymbol{v}_i \in \mathbf{C}^n)$ で表わすとき，次の仮定をおく。

1. $|\lambda_i| < 1$，$\lambda_i \neq 0$　　$(\forall i)$
2. $\{\boldsymbol{v}_i\}_{i=1,\cdots,n}$　　一次独立

このとき，解は次式で与えられる。

$$\boldsymbol{X} = [\boldsymbol{u}_1 \cdots \boldsymbol{u}_n][\boldsymbol{v}_1 \cdots \boldsymbol{v}_n]^{-1}$$

ここで，$\{\boldsymbol{u}_i\}_{i=1,\dots,n}$は次式で与えられる。

$$\begin{cases} \lambda_i \text{が実数のとき：} \boldsymbol{u}_i = \boldsymbol{U}_i {}^t\boldsymbol{A}^{-1}\boldsymbol{Q}\boldsymbol{v}_i \\ \qquad\qquad\qquad \boldsymbol{U}_i = ({}^t\boldsymbol{A}^{-1} - \lambda_i \boldsymbol{I}_n)^{-1} \\ \lambda_i \text{が実数でないとき：} \lambda_i = \lambda_{iR} + j\lambda_{iI},\ \lambda_{i+1} = \lambda_{iR} - j\lambda_{iI} \end{cases}$$

$$\begin{bmatrix} \boldsymbol{u}_i \\ \boldsymbol{u}_{i+1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{U}_i & -\boldsymbol{U}_{i+1} \\ \boldsymbol{U}_{i+1} & \boldsymbol{U}_i \end{bmatrix} \begin{bmatrix} {}^t\boldsymbol{A}^{-1}\boldsymbol{Q}\boldsymbol{v}_{iR} \\ {}^t\boldsymbol{A}^{-1}\boldsymbol{Q}\boldsymbol{v}_{iI} \end{bmatrix}$$

$$\begin{cases} \boldsymbol{U}_i = \{(({}^t\boldsymbol{A}^{-1} - \lambda_{iR}\boldsymbol{I}_n)^2 + \lambda_{iI}^2 \boldsymbol{I}_n\}^{-1} ({}^t\boldsymbol{A}^{-1} - \lambda_{iR}\boldsymbol{I}_n) \\ \boldsymbol{U}_{i+1} = \{(({}^t\boldsymbol{A}^{-1} - \lambda_{iR}\boldsymbol{I}_n)^2 + \lambda_{iI}^2 \boldsymbol{I}_n\}^{-1} (\lambda_{iI}\boldsymbol{I}_n) \end{cases}$$

$$(\boldsymbol{v}_i = \boldsymbol{v}_{iR} + j\boldsymbol{v}_{iI})$$

実 行 例

〔1〕 連続系のリヤプノフ方程式の解の算出例

```
/MAT/ TIME-14:00:03 CPU-00:00:02
LYP
 TO SOLVE LYAPUNOV EQUATION,
 (1) X*A+A'X=-Q, (2) X=A'*X*A+Q
 WHICH TASK ? :  1
 SPECIFY N :  2
 INPUT -MODE OF ( 2, 2) MATRIX   A        : _
 ( 1) ROW
0,1
 ( 2) ROW
-2,-3
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX   Q        : I
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 2, 2) MATRIX   X        : _
 --- ( 2, 2) MATRIX    X     ---
 ( 1)  1.25000000D+00  2.50000000D-01
 ( 2)  2.50000000D-01  2.50000000D-01
 FILENAME : _
 VERIFY ?
 NORM OF EQUATION ERROR =  1.05471187D-15
   /MAT/ TIME-14:00:23 CPU-00:00:03
```

〔2〕 離散系のリヤプノフ方程式の解の算出例

```
/MAT/ TIME-14:00:20 CPU-00:00:03
LYP
 TO SOLVE LYAPUNOV EQUATION,
 (1) X*A+A'X=-Q, (2) X=A'*X*A+Q
 WHICH TASK ? :  2
 SPECIFY N :  2
 INPUT -MODE OF ( 2, 2) MATRIX   A        : _
 ( 1) ROW
6.00423599D-1,2.32544158D-1
 ( 2) ROW
-4.65088316D-1,-9.72088747D-2
 TYPE OR REVISE ?  T
 --- ( 2, 2) MATRIX    A     ---
 ( 1)  6.00423599D-01  2.32544158D-01
 ( 2) -4.65088316D-01 -9.72088747D-02
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX   Q        : I
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 2, 2) MATRIX   X        : _
 --- ( 2, 2) MATRIX    X     ---
 ( 1)  1.71667959D+00  2.47796104D-01
 ( 2)  2.47796104D-01  1.09194791D+00
 FILENAME : _
 VERIFY ? _
 NORM OF EQUATION ERROR =  2.02615702D-15
 /MAT/ TIME-14:00:40 CPU-00:00:05
```

固有値問題の求解	MT. Ⅲ. 1
E I G	

機　　能

行列 $\boldsymbol{A} \in \mathbf{R}^{n \times n}$ に対し，

$$\boldsymbol{A}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i \qquad (\lambda_i \in \mathbf{C}\ ,\ \boldsymbol{v}_i \in \mathbf{C}^n)$$

を満足する $\{\lambda_i\}$，$\{\boldsymbol{v}_i\}$ （i ＝1，2，…，n）を求める[27,28,29]。

実 行 例

〔1〕　固有値問題の求解例

```
 /MAT/ TIME-14:00:43 CPU-00:00:05
EIG
  TO SOLVE EIGEN-PROBLEM :
  A(N,N)*V(N)=(RR+J*RI)*V(N)
  SPECIFY N :    3
  INPUT -MODE OF ( 3, 3) MATRIX    A       : _
  ( 1) ROW
1,2,3
  ( 2) ROW
4,5,6
  ( 3) ROW
7,8,9
  TYPE OR REVISE ? _
 ---------------------------------------------------------------------
   NO.      REAL                  IMAGINARY                ABSOLUTE
 ---------------------------------------------------------------------
  ( 1)    1.61168440D+01        0.0                    1.61168440D+01
  ( 2)   -1.11684397D+00        0.0                    1.11684397D+00
  ( 3)   -3.35578952D-16        0.0                    3.35578952D-16
 ---------------------------------------------------------------------
 ***
 FILENAME : _
 OUTPUT-MODE OF ( 3, 3) MATRIX    VR       : _
 --- ( 3, 3) MATRIX    VR      ---
 ( 1)   2.83349452D-01  8.19142369D-01 -3.33333333D-01
 ( 2)   6.41674726D-01  9.04288154D-02  6.66666667D-01
 ( 3)   1.00000000D+00 -6.38284739D-01 -3.33333333D-01
 FILENAME : _
 OUTPUT-MODE OF ( 3, 3) MATRIX    VI       : _
 --- ( 3, 3) MATRIX    VI      ---
 ( 1)   0.0             0.0             0.0
 ( 2)   0.0             0.0             0.0
 ( 3)   0.0             0.0             0.0
 FILENAME : _
 /MAT/ TIME-14:01:02 CPU-00:00:07
```

一般化固有値問題の求解	MT. Ⅲ. 2
GEIG	

機　　能

行列 $\boldsymbol{A}$, $\boldsymbol{B} \in \mathbf{R}^{n \times n}$ に対し，

$$\beta_i \boldsymbol{A} \boldsymbol{v}_i = (\alpha_{iR} + j\alpha_{iI}) \boldsymbol{B} \boldsymbol{v}_i \qquad (\alpha_{iR},\ \alpha_{iI},\ \beta_i \in \mathbf{R},\ \boldsymbol{v}_i \in \mathbf{C}^n)$$

を満足する $\{\alpha_{iR} + j\alpha_{iI}\}$, $\{\beta_i\}$, $\{\boldsymbol{v}_i\}$　($i = 1, 2, \cdots, n$) を求める。

実 行 例

〔1〕　一般化固有値問題の求解例

```
 /MAT/ TIME-14:01:05 CPU-00:00:07
GEIG
 TO SOLVE GENERALIZED EIGEN-PROBLEM :
 A(N,N)*V(N)=(RR+J*RI)*B(N,N)*V(N)
 SPECIFY N :   2
 INPUT -MODE OF ( 2, 2) MATRIX   A        : _
 ( 1) ROW
1,2
 ( 2) ROW
3,4
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX   B        : I
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 1, 2) MATRIX ALFAR      : _
 --- ( 1, 2) MATRIX ALFAR    ---
 ( 1)  5.37228132D+00 -3.72281323D-01
 FILENAME :_
 OUTPUT-MODE OF ( 1, 2) MATRIX ALFAI      : _
 --- ( 1, 2) MATRIX ALFAI    ---
 ( 1)  0.0             0.0
 FILENAME :_
 OUTPUT-MODE OF ( 1, 2) MATRIX BETA       : _
 --- ( 1, 2) MATRIX BETA     ---
 ( 1)  1.00000000D+00  1.00000000D+00
 FILENAME :_
 EPS =   1.0D-10
 *** FLAG ON EIGENVALUES ***
   0  0
 --------------------------------------------------------------
  NO.      REAL                 IMAGINARY           ABSOLUTE
 --------------------------------------------------------------
 ( 1)   5.37228132D+00       0.0                5.37228132D+00
 ( 2)  -3.72281323D-01       0.0                3.72281323D-01
 --------------------------------------------------------------
 FILENAME :_
 OUTPUT-MODE OF ( 2, 2) MATRIX   V        : _
 --- ( 2, 2) MATRIX   V      ---
 ( 1)  4.57427108D-01  1.00000000D+00
 ( 2)  1.00000000D+00 -6.86140662D-01
 FILENAME :_
 /MAT/ TIME-14:01:24 CPU-00:00:09
```

特異値分解の実行	MT. Ⅲ. 3
SVD	

機　能

行列$\boldsymbol{A} \in \mathbf{R}^{n \times n}$に対し，

$$\boldsymbol{A} = \boldsymbol{U} \cdot \mathrm{diag}\{\sigma_1 \cdots \sigma_m\}\, {}^t\boldsymbol{V}$$

$${}^t\boldsymbol{U}\boldsymbol{U} = {}^t\boldsymbol{V}\boldsymbol{V} = \boldsymbol{I}_m$$

を満足する$\{\sigma_i\}$ $(i = 1, 2, \cdots, m)(\sigma_i \in \mathbf{R})$, $\boldsymbol{U} \in \mathbf{R}^{n \times m}$, $\boldsymbol{V} \in \mathbf{R}^{m \times m}$を求める。

ここで，σ_iを$\boldsymbol{A}$のi番目の特異値と呼び，第1式の右辺を$\boldsymbol{A}$の特異値分解と呼ぶ。[30]

実 行 例

〔1〕　特異値分解の実行例

```
 /MAT/ TIME-14:01:24 CPU-00:00:09
SVD
 TO OBTAIN SINGULAR VALUE DECOMPOSITION :
 A(N,M)=U(N,M)*DIAG(S(M))*V'(M,M)
 SPECIFY (N,M) :   5,3
 INPUT -MODE OF ( 5, 3) MATRIX   A      : F
 FILENAME :   S
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 1, 3) MATRIX   S      : _
 --- ( 1, 3) MATRIX   S       ---
 ( 1)  3.51272233D+01  2.46539670D+00  7.45735190D-16
 FILENAME : _
 OUTPUT-MODE OF ( 5, 3) MATRIX   U      : _
 --- ( 5, 3) MATRIX   U       ---
 ( 1) -3.54557057D-01 -6.88686644D-01  5.32043519D-01
 ( 2) -3.98696370D-01 -3.75554529D-01 -7.56878438D-01
 ( 3) -4.42835683D-01 -6.24224150D-02  1.77413834D-01
 ( 4) -4.86974996D-01  2.50709699D-01 -2.12366432D-01
 ( 5) -5.31114309D-01  5.63841814D-01  2.59787516D-01
 FILENAME : _
 OUTPUT-MODE OF ( 3, 3) MATRIX   V      : _
 --- ( 3, 3) MATRIX   V       ---
 ( 1) -2.01664911D-01  8.90317133D-01 -4.08248290D-01
 ( 2) -5.16830501D-01  2.57331627D-01  8.16496581D-01
 ( 3) -8.31996092D-01 -3.75653879D-01 -4.08248290D-01
 FILENAME : _
 /MAT/ TIME-14:01:36 CPU-00:00:10
```

像空間、零化空間の基底の算出	MT. Ⅲ. 4
MAP	

機　能

行列 $\boldsymbol{A} \in \mathbf{R}^{n\times m}$ に対し，次の特異値分解を考える。

$$\boldsymbol{A} = \boldsymbol{U} \cdot \mathrm{diag}\{\sigma_1 \cdots \sigma_m\}\ {}^t\boldsymbol{V}\ (\sigma_1 \geqq \sigma_2 \geqq \cdots \geqq \sigma_m)$$

$${}^t\boldsymbol{U}\boldsymbol{U} = {}^t\boldsymbol{V}\boldsymbol{V} = \boldsymbol{I}_m$$

$$\{\sigma_i\}\ (i = 1,\ 2,\ \cdots,\ m)\quad (\sigma_i \in \mathbf{R}^{n\times m}),\ \boldsymbol{U} \in \mathbf{R}^{n\times m},\ \boldsymbol{V} \in \mathbf{R}^{m\times m}$$

このとき，十分に小さい $\varepsilon > 0$ に対して

$$\sigma_1 \geqq \sigma_2 \geqq \cdots \geqq \sigma_k > \varepsilon > \sigma_{k+1} \geqq \cdots \geqq \sigma_m$$

を満たす整数 k が定まる。今，$\boldsymbol{U} \in \mathbf{R}^{n\times m}$ と $\boldsymbol{V} \in \mathbf{R}^{m\times n}$ を

$$\boldsymbol{U} = [\boldsymbol{U}_1 \vdots \boldsymbol{U}_2],\ \boldsymbol{V} = [\boldsymbol{V}_1 \vdots \boldsymbol{V}_2]$$

の様に分割するとき，行列 $\boldsymbol{A}$ に対応する線形写像に関して，以下の行列を計算する。

$$\boldsymbol{I}_m\boldsymbol{A} = \mathrm{span}\ \{\boldsymbol{U}_1\}$$

$$\mathrm{Ker}\boldsymbol{A} = \mathrm{span}\ \{\boldsymbol{V}_2\}$$

$$(\mathrm{Ker}\boldsymbol{A})^{\perp} = \mathrm{span}\ \{\boldsymbol{V}_1\}$$

実 行 例

〔1〕　像空間，零化空間の基底の算出例

```
/MAT/ TIME-14:01:37 CPU-00:00:10
MAP
 TO OBTAIN BASIS MATRICES OF THE FOLLOWING SUBSPACES
 ON MAP CORRESPONDING TO MATRIX A(N,M) :
 (1) IMAGE, (2) KERNEL, (3) (KERNEL)
 SPECIFY (N,M) :   5,3
 INPUT -MODE OF ( 5, 3) MATRIX   A        : F
 FILENAME :   S
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 1, 3) MATRIX   S        : _
 --- ( 1, 3) MATRIX   S       ---
 ( 1)   3.51272233D+01   2.46539670D+00   7.45735190D-16
 FILENAME : _
 EPS =   1.0D-10
 OUTPUT-MODE OF ( 5, 2) MATRIX  IM         : _
 --- ( 5, 2) MATRIX  IM       ---
 ( 1) -3.54557057D-01 -6.88686644D-01
 ( 2) -3.98696370D-01 -3.75554529D-01
 ( 3) -4.42835683D-01 -6.24224150D-02
 ( 4) -4.86974996D-01  2.50709699D-01
 ( 5) -5.31114309D-01  5.63841814D-01
 FILENAME : _
 OUTPUT-MODE OF ( 3, 1) MATRIX  KER        : _
 --- ( 3, 1) MATRIX  KER      ---
 ( 1) -4.08248290D-01
 ( 2)  8.16496581D-01
 ( 3) -4.08248290D-01
 FILENAME : _
 OUTPUT-MODE OF ( 3, 2) MATRIX  KER'       : _
 --- ( 3, 2) MATRIX  KER'     ---
 ( 1) -2.01664911D-01  8.90317133D-01
 ( 2) -5.16830501D-01  2.57331627D-01
 ( 3) -8.31996092D-01 -3.75653879D-01
 FILENAME : _
/MAT/ TIME-14:01:49 CPU-00:00:12
```

最小２乗解の算出	ＭＴ．Ⅲ．５
ＬＳＱ	

機　　能

行列$\boldsymbol{A} \in \mathbf{R}^{n \times m}$と$\boldsymbol{B} \in \mathbf{R}^{n \times k}$ に対し，次の方程式

$$\boldsymbol{A}\boldsymbol{X} = \boldsymbol{B}$$

に関する最小２乗問題の解$\boldsymbol{X} \in \mathbf{R}^{m \times k}$は，次式で与えられる。

$$\boldsymbol{X} = \boldsymbol{A}^{+}\boldsymbol{B}$$

ここで，$\boldsymbol{A}^{+} \in \mathbf{R}^{m \times n}$は，擬似逆行列である。$\boldsymbol{x}_i \in \mathbf{R}^{m}$と$\boldsymbol{b}_i \in \mathbf{R}^{n}$でそれぞれ$\boldsymbol{X}$と$\boldsymbol{B}$の第 i列ベクトルを表すと，上記の$\boldsymbol{x}_i$はノルム$\| \boldsymbol{A}\boldsymbol{x}_i - \boldsymbol{b}_i \|$を最小にする$\boldsymbol{x}_i$となっている。

実 行 例

〔１〕　最小２乗解の算出例

```
 /MAT/ TIME-14:01:50 CPU-00:00:12
LSQ
 TO SOLVE A(N,M)*X(M,L)=B(N,L)
 SPECIFY (N,M,L) :   5,3,1
 INPUT -MODE OF ( 5, 3) MATRIX   A      : F
 FILENAME :   S
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 5, 1) MATRIX   B      : _
 ( 1) ROW
5
 ( 2) ROW
5
 ( 3) ROW
5
 ( 4) ROW
5
 ( 5) ROW
5
 TYPE OR REVISE ? _
 EPS =   1.0D-10
 OUTPUT-MODE OF ( 3, 1) MATRIX   X      : _
 --- ( 3, 1) MATRIX   X      ---
 ( 1) -5.00000000D-01
 ( 2)  1.52655666D-16
 ( 3)  5.00000000D-01
 FILENAME : _
 /MAT/ TIME-14:02:20 CPU-00:00:15
```

座標変換の実行	MT. Ⅲ. 6
COOD	

機　能

正方行列$\boldsymbol{A} \in \mathbf{R}^{n\times n}$と正則行列$\boldsymbol{T} \in \mathbf{R}^{n\times n}$に対し，

$$\widehat{\boldsymbol{A}} = \boldsymbol{T}^{-1}\boldsymbol{A}\boldsymbol{T}$$

を計算する。

実 行 例

〔1〕　座標変換の実行例

```
 /MAT/ TIME-14:02:21 CPU-00:00:15
COOD
 TO OBTAIN A¬(N,N)=T(N,N)**(-1)*A(N,N)*T(N,N)
 SPECIFY N :    2
 INPUT -MODE OF ( 2, 2) MATRIX   A       : _
 ( 1) ROW
1,2
 ( 2) ROW
3,4
 TYPE OR REVISE ? _
 INPUT -MODE OF ( 2, 2) MATRIX   T       : _
 ( 1) ROW
0,1
 ( 2) ROW
1,0
 TYPE OR REVISE ? _
 EPS =    1.00-10
 OUTPUT-MODE OF ( 2, 2) MATRIX   A¬      : _
 --- ( 2, 2) MATRIX   A¬     ---
 ( 1)   4.00000000D+00   3.00000000D+00
 ( 2)   2.00000000D+00   1.00000000D+00
 FILENAME : _
 /MAT/ TIME-14:02:40 CPU-00:00:17
```

ベクトル列の一次独立性の検証	MT. Ⅲ. 7
VEC	

機　能

一次独立なベクトル$\{\boldsymbol{a}_i\}$ i＝1，…，m（$\boldsymbol{a}_i \in \mathbf{R}^n$）が与えられるとき，あるベクトル$\boldsymbol{b} \in \mathbf{R}^n$ がそれらに一次独立であるか否か調査する。行列［$\boldsymbol{a}_1 \cdots \boldsymbol{a}_m \vdots \boldsymbol{b}$］$\in \mathbf{R}^{n \times (m+1)}$ の特異値がすべて非零のものであるか否かにより判断できる。

実 行 例

〔1〕　ベクトル列の一次独立性の検証例

```
/MAT/ TIME-14:02:42 CPU-00:00:17
VEC
 TO INVESTIGATE COLUMN-INDEPENDENCY OF A(N,M)
 SPECIFY (N,M) :   5,3
 INPUT -MODE OF ( 5, 3) MATRIX   A      : F
 FILENAME :   S
 TYPE OR REVISE ? _
 EPS =   1.0D-10
 *** COLUMN-INDEPENDENCY OF A ***
   1   1  -1
/MAT/ TIME-14:03:05 CPU-00:00:19
```

ノルム、行列式、トレースの算出	MT. Ⅲ. 8
NORM	

機　能

行列$\boldsymbol{A} \in \mathbf{R}^{n \times m}$に対し，

① Euclidean Norm　：　$x_1 = \max\{\|\boldsymbol{A}\boldsymbol{v}\| : \|\boldsymbol{v}\| = 1\}$

② Frobenius Norm　：　$x_2 =（\Sigma\Sigma a_{ij}^2）$

③ |Determinant|　：　$x_3 = |\det \boldsymbol{A}|$　（n＝mのとき）

④ Trace　：　$x_4 = \mathrm{tr}(\boldsymbol{A})$　（n＝　のとき）

を求める。

①～③は，$\boldsymbol{A}$ の特異値を$\{\sigma_i\}$ i ＝ 1，…とするとき，次式で表わせる。

$x_1 = \max\{\sigma_i\}$

$x_2 = \sqrt{(\Sigma \sigma_i^2)}$

$x_3 = |\Pi \sigma_i|$

実行例

〔1〕 Euclidean Norm算出例

```
/MAT/ TIME-14:03:07 CPU-00:00:19
NORM
TO CALCULATE THE FOLLOWING VALUES ON A(N,M)
(1) EUCLIDEAN NORM, (2) FROBENIUS NORM
(3) DETERMINANT (N=M), (4) TRACE (N=M)
WHICH VALUE ?  :  1
SPECIFY (N,M) :  5,3
INPUT -MODE OF ( 5, 3) MATRIX    A        : F
FILENAME :   S
TYPE OR REVISE ?   T
--- ( 5, 3) MATRIX    A       ---
( 1)  1.00000000D+00  6.00000000D+00  1.10000000D+01
( 2)  2.00000000D+00  7.00000000D+00  1.20000000D+01
( 3)  3.00000000D+00  8.00000000D+00  1.30000000D+01
( 4)  4.00000000D+00  9.00000000D+00  1.40000000D+01
( 5)  5.00000000D+00  1.00000000D+01  1.50000000D+01
TYPE OR REVISE ?_
OUTPUT-MODE OF ( 1, 1) MATRIX    A*       : _
--- ( 1, 1) MATRIX    A*      ---
( 1)  3.51272233D+01
***
FILENAME : _
/MAT/ TIME-14:03:20 CPU-00:00:20
```

〔2〕 Frobenius Norm算出例

```
/MAT/ TIME-14:03:23 CPU-00:00:20
NORM
TO CALCULATE THE FOLLOWING VALUES ON A(N,M)
(1) EUCLIDEAN NORM, (2) FROBENIUS NORM
(3) DETERMINANT (N=M), (4) TRACE (N=M)
WHICH VALUE ?  :  2
SPECIFY (N,M) :  5,3
INPUT -MODE OF ( 5, 3) MATRIX    A        : F
FILENAME :   S
TYPE OR REVISE ?_
OUTPUT-MODE OF ( 1, 1) MATRIX    A*       : _
--- ( 1, 1) MATRIX    A*      ---
( 1)  3.52136337D+01
FILENAME : _
/MAT/ TIME-14:03:37 CPU-00:00:25
```

〔3〕 |Determinant| 算出例

```
/MAT/ TIME-14:03:38 CPU-00:00:25
NORM
 TO CALCULATE THE FOLLOWING VALUES ON A(N,M)
 (1) EUCLIDEAN NORM, (2) FROBENIUS NORM
 (3) DETERMINANT (N=M), (4) TRACE (N=M)
 WHICH VALUE ?  :   3
 SPECIFY N :   2
 INPUT -MODE OF ( 2, 2) MATRIX   A       : _
 ( 1) ROW
1,2
 ( 2) ROW
3,4
 TYPE OR REVISE ?_
 OUTPUT-MODE OF ( 1, 1) MATRIX   A*      : _
 --- ( 1, 1) MATRIX   A*     ---
 ( 1)  2.00000000D+00
 FILENAME : _
 /MAT/ TIME-14:03:48 CPU-00:00:27
```

〔4〕 Trace 算出例

```
/MAT/ TIME-14:03:48 CPU-00:00:27
NORM
 TO CALCULATE THE FOLLOWING VALUES ON A(N,M)
 (1) EUCLIDEAN NORM, (2) FROBENIUS NORM
 (3) DETERMINANT (N=M), (4) TRACE (N=M)
 WHICH VALUE ?  :   4
 SPECIFY N :   2
 INPUT -MODE OF ( 2, 2) MATRIX   A       : _
 ( 1) ROW
1,2
 ( 2) ROW
3,4
 TYPE OR REVISE ? _
 OUTPUT-MODE OF ( 1, 1) MATRIX   A*      : _
 --- ( 1, 1) MATRIX   A*     ---
 ( 1)  5.00000000D+00
 FILENAME : _
 /MAT/ TIME-14:03:59 CPU-00:00:29
```

4. ＤＰＡＣＳ／Ｊバッチ処理版（ＤＰＡＣＳＢ）の機能・利用編

4.1 機能

本パッケージは，DPACS／JのＤＰＡＣＳサブシステムの一機能であるシステム同定機能についてのみバッチ化したものである。本来のDPACS／Jの機能から言えば，会話的に使用できるところに，その特徴の１つが有るのであるが，システム同定機能を実行する際に，同定する次元や繰り返しの数が多かったり，システム同定の可観測指数を高くしたりすると，非常に時間がかかり（１入力１出力，時系列データ点数＝１５００，可観測指数＝８で10分程度），現在の日本原子力研究所那珂研究所の計算機利用における端末上での実行時間の制約（ＣＰＵ時間で２分）を越えてしまうため，システム同定機能に関してのみバッチ化することとなった。

ただし，基礎となるＤＰＡＣＳサブシステムの基本思想はそのまま受け継ぎ，コマンドならびにパラメータをカード・イメージで入力することにより，システム同定を実行する。しかも，コマンドやパラメータの入力については，画面からの入力がＪＣＬ中のＳＹＳＩＮデータセットからの入力になっただけのことであり，基本的な違いは無い。また，本パッケージにて同定されたシステムデータは，そのままDPACS／Jパッケージのシステムとして登録されるので，その後の解析・設計作業は，DPACS／Jにて行うことができる。

従って，ここではこのパッケージのコマンドについての説明は改めて行わず，実行例ならびに，実行ＪＣＬについてのみ掲載することにする。

DPACSBパッケージには，DPACS／JのＤＰＡＣＳサブシステムのシステム同定機能と同様に，以下の４つのコマンドがある。

第4.1表　ＤＰＡＣＳＢコマンド一覧

機能分類	コマンド名	機能概要
ＤＰＡＣＳＢ	ＧＬＳ	一般化最小２乗推定
Ｉ	ＣＭＬ	条件付き最尤推定
システム同定	ＳＭＬ	条件付き最尤推定
	ＬＩ	制御情報下の最尤推定

4.2 利用例

以下にＧＬＳ（一般化最小２乗法）をバッチ処理版で実行した際のＪＣＬと実行結果リストを示す。

＜実行ＪＣＬ例＞

```
//JCLG JOB
//JCLG EXEC JCLG
//SYSIN DD DATA,DLM='++'
// JUSER 99999999,XX.XXXXX,9999.99
   T.3 W.3 I.4 C.5
  OPTP PASSWORD=PPPPP,NOTIFY=XXXX,MSGLEVEL=(1,1,2)
// EXEC PGM=KEQEFT01,DYNAMNBR=20
//SYSTSPRT DD SYSOUT=*,
//            DCB=(RECFM=FBA,LRECL=137,BLKSIZE=19043)
//FT06F001 DD SYSOUT=*,
//            DCB=(RECFM=FBA,LRECL=137,BLKSIZE=19043)
//CMDIN     DD DUMMY
//CMDOUT    DD DUMMY
//FT01F001 DD DSN=&&WORK,SPACE=(TRK,(5,1)),UNIT=TSSWK,
//            DCB=(LRECL=80,BLKSIZE=3120,RECFM=FB)
//FT99F001 DD DUMMY
//SYSPRINT DD SYSOUT=*,
//            DCB=(RECFM=FBA,LRECL=137,BLKSIZE=19043)
//SYSTSIN  DD  *
CALL 'J3294.DPACS4J.LOAD(MAIN)'
/*
//FT05F001 DD  *
GLS EXIDN
2
2
5
496
1
0.05,0.05
3
1
2
```

１０行の空白行

システムのコメント入力行

```
BYE
/*
++
//
```

＜出力結果例＞

```
/DPACS/ 20: 4:27
TO IDENTIFY SYSTEM BY G.L.S.
----------------------------------------------------------------------------
   G.L.S. DRIVING PARAMETERS FOR SYSTEMDATA EXIDN.IO
----------------------------------------------------------------------------

1)  1 OBSERVABILITY INDICES :     2
2)  1 DEGREES IN EACH ROW OF F(.) :     2
3)  STARTING DATA-POINT :     5
4)  DATA-LENGTH USED FOR G.L.S. :   496
5)  TYPE OF FILTER :  INFINITY FILTER
6)  CONVERGENCE CRITERION (EPSA,EPSB) :     5.00000000D-02  5.00000000D-02
7)  NO. OF ITERATION OF OPTIMIZATION :     3
8)  COEFFICIENT PATERN
     1-TH ROW OF T :     2  3
     1-TH ROW OF U :     2  3
----------------------------------------------------------------------------
( 1)-TH OUTPUT-SUBSYSTEM'S ESTIMATION
----------------------------------------------------------------------------
----------------------------------------------------------------------------
(  1)-TH ITERATION FOR 1-TH OUTPUT-SUBSYSTEM
----------------------------------------------------------------------------
--- ( 1, 1) MATRIX  TO       ---
( 1)  1.00000000D+00
--- ( 1, 1) MATRIX  T1       ---
( 1) -1.46575983D+00
--- ( 1, 1) MATRIX  T2       ---
( 1)  5.25862225D-01
--- ( 1, 1) MATRIX  UO       ---
( 1)  0.0
--- ( 1, 1) MATRIX  U1       ---
( 1)  3.11857957D-03
--- ( 1, 1) MATRIX  U2       ---
( 1)  1.01145327D+00
--- ( 1, 2) MATRIX FILTER   ---
( 1)  9.06331618D-01  4.31199198D-01
--- ( 2, 2) MATRIX  COR.    ---
( 1)  6.21084960D-02 -3.91407288D-02
( 2)  1.00000000D+00 -6.30199269D-01

D(ALFA) =  3.00619390D+00     D(BETA) =  1.33753082D+00

--- ( 1, 4) MATRIX A.VAR.   ---
( 1)  1.20276823D-03  6.13858626D-02  4.52340687D-02  4.43632776D-02
--- ( 1, 2) MATRIX B.VAR.   ---
( 1)  4.43472976D-01  1.95497714D+00
----------------------------------------------------------------------------
(  2)-TH ITERATION FOR 1-TH OUTPUT-SUBSYSTEM
----------------------------------------------------------------------------
--- ( 1, 1) MATRIX  TO       ---
( 1)  1.00000000D+00
--- ( 1, 1) MATRIX  T1       ---
( 1) -1.49461319D+00
--- ( 1, 1) MATRIX  T2       ---
( 1)  5.54791230D-01
--- ( 1, 1) MATRIX  UO       ---
( 1)  0.0
--- ( 1, 1) MATRIX  U1       ---
( 1) -4.68823984D-03
--- ( 1, 1) MATRIX  U2       ---
```

```
( 1)  1.01112563D+00
--- ( 1, 4) MATRIX FILTER   ---
( 1)  1.21617563D+00  9.77699452D-01  3.74397613D-01  1.14560498D-01
--- ( 2, 2) MATRIX  COR.    ---
( 1)  2.78191615D-02 -6.40656730D-03
( 2)  1.00000000D+00 -2.30293328D-01

D(ALFA) =  6.59168128D-02     D(BETA) =  7.62007976D-01

--- ( 1, 4) MATRIX A.VAR.   ---
( 1)  1.41268020D-05  7.38739280D-04  1.36422192D-03  5.20890900D-03
--- ( 1, 2) MATRIX B.VAR.   ---
( 1)  8.14796941D-02  1.30242824D-01
-------------------------------------------------------------------------
(  3)-TH ITERATION FOR 1-TH OUTPUT-SUBSYSTEM
-------------------------------------------------------------------------
--- ( 1, 1) MATRIX  T0      ---
( 1)  1.00000000D+00
--- ( 1, 1) MATRIX  T1      ---
( 1) -1.49777944D+00
--- ( 1, 1) MATRIX  T2      ---
( 1)  5.57863327D-01
--- ( 1, 1) MATRIX  U0      ---
( 1)  0.0
--- ( 1, 1) MATRIX  U1      ---
( 1) -3.93801080D-03
--- ( 1, 1) MATRIX  U2      ---
( 1)  1.00753513D+00
--- ( 1, 6) MATRIX FILTER   ---
( 1)  1.29297656D+00  1.14489524D+00  5.39230315D-01  2.15461338D-01
( 1)  3.64260381D-02  8.45369018D-03
--- ( 2, 2) MATRIX  COR.    ---
( 1)  2.57373045D-02 -1.09630877D-03
( 2)  1.00000000D+00 -4.25960987D-02

D(ALFA) =  1.05790812D-02     D(BETA) =  4.24929547D-01

--- ( 1, 4) MATRIX A.VAR.   ---
( 1)  2.61866176D-06  1.54013246D-04  3.88126088D-04  2.51121913D-03
--- ( 1, 2) MATRIX B.VAR.   ---
( 1)  5.06928891D-02  4.65502601D-02
-------------------------------------------------------------------------
   PARAMETERS IDENTIFIED BY G.L.S.
-------------------------------------------------------------------------
--- ( 1, 1) MATRIX  T0      ---
( 1)  1.00000000D+00
--- ( 1, 1) MATRIX  T1      ---
( 1) -1.49777944D+00
--- ( 1, 1) MATRIX  T2      ---
( 1)  5.57863327D-01
--- ( 1, 1) MATRIX  U0      ---
( 1)  0.0
--- ( 1, 1) MATRIX  U1      ---
( 1) -3.93801080D-03
--- ( 1, 1) MATRIX  U2      ---
( 1)  1.00753513D+00
--- ( 1, 6) MATRIX FILTER   ---
( 1)  1.29297656D+00  1.14489524D+00  5.39230315D-01  2.15461338D-01
( 1)  3.64260381D-02  8.45369018D-03
--- ( 1, 1) MATRIX SIGMA    ---
( 1)  2.57373045D-02
--- ( 1, 1) MATRIX  DALF    ---
```

```
( 1)   1.05790812D-02
--- ( 1, 1) MATRIX  DBET     ---
( 1)   4.24929547D-01
--- ( 1, 4) MATRIX A.VAL.    ---
( 1)   2.61866176D-06   1.54013246D-04   3.88126088D-04   2.51121913D-03

GIVE COMMENT FOR SYSTEMDATA :

/DPACS/ 20: 9:50
```

4.3 利用上の注意事項

本パッケージは，入力パラメータにエラーが存在した場合，そこで処理を中断する。こうしたエラーが発生した場合，再度パラメータの妥当性についてチェックを行う必要がある。

エラーが発生した場合，カードイメージのデータを見直すことは勿論，必要に応じて会話版（ＤＰＡＣＳ）を実行し，エラーメッセージを画面上にて確認することが間違い探しの早道の場合もある。

5. データ編集・登録プログラム（EDITJ）の機能・利用編

DPACS／Jに対して外部から離散型の時系列データを取り込む目的で開発されたコードでである。

5.1 機能

EDITJは，外部の時系列データの内，JT－60実験データベースの時系列データ（結果データ，放電条件データ）をPID（Point Identification）番号と放電番号をキーにしてバッファに取り込み，編集した後DPACS／Jの時系列データファイルに登録する機能，及びニコレ社製オシロスコープからの出力のフォーマットである第5.1図に示すフォーマットのデータを取り込み，編集した後DPACS／Jの時系列データファイルに登録する機能の2つである。後者の場合は，第5.1図のフォーマットに従っていれば何でも取り込めるわけであり汎用性がある。

具体的な機能の特徴は次の通りである。

① 原時系列データの一部または全部を抽出することができる。

② その際サンプリング・ピッチを変更できるが，必要に応じて3次元スプライン補間を実行する。

③ 原時系列データが第5.1図からの入力の場合，本当の物理値に直すための演算（ax＋b）のaとbを指定できる。

④ バッファに複数の時系列データを取り込み，①，②，③の編集を経た後，並びを指定することにより，ベクトルの時系列データを作成して，DPACS／Jの時系列データファイルに登録する。

⑤ バッファをディスクファイルに書き出し保存することができる。

⑥ 漢字端末または通常の英字端末からユーザフレンドリに操作できる。それぞれの端末の＜起動画面＞を参照のこと。

タイトル（A32）							
^NWAVE=^^^	波形数(I10)						
^NDATA=^^^	ﾃﾞｰﾀ数n(I10)	^CHANNO=^^	ﾁｬﾝﾈﾙNO(I10)	^TSTART=^^	T0 (1PE10.3)	^TSTEP=^^^	TS (1PE10.3)
V0 (1PE10.3)	V1 (1PE10.3)	V2 (1PE10.3)	V3 (1PE10.3)	V4 (1PE10.3)	V5 (1PE10.3)	V6 (1PE10.3)	V7 (1PE10.3)
V9 (1PE10.3)	. . . .						. . . .
. . . .							. . . .
Vn-3	Vn-2	Vn-1	0.0	← レコードの最後まで0.0が入る →			0.0
^NDATA=^^^	. . . .						. . . .

波形数分繰返す

注意点　a）ファイル属性は　DCB=(RECFM=FB, BLKSIZE=3200, LRECL=80, DSORG=PS)，文字データ(EBCDIC)である。
b）T0；V0データの時刻（SEC ），TS；サンプリングピッチ（SEC ），^ ；blank

第5.1図　　EDITJが続み取れるデータFORMAT

5.2 利用例

画面のオペレーションフローを第5.2図に示す。それぞれの画面のオペレーションを引続き示す。

<起動画面>

ＥＤＩＴＪには，漢字端末用の画面（日本語出力）とキャラクタ端末用の画面（英語出力）がある。起動時の操作方法が異なるだけで，その後の操作は全く同一である。（起動後のキャラクタ端末用ＥＤＩＴＪに関しての説明は省略する。

・漢字端末用ＥＤＩＴＪの起動方法

```
EDITJ
 TIME-10:23:25 CPU-00:00:00 SERVICE-1276 SESSION-00:00:12 JUNE 14,1989
 *********************************************************
 ***** JT-60 DATA EDITION START  (DATABASE TO DPACS) *****
 *********************************************************
 *** _
```

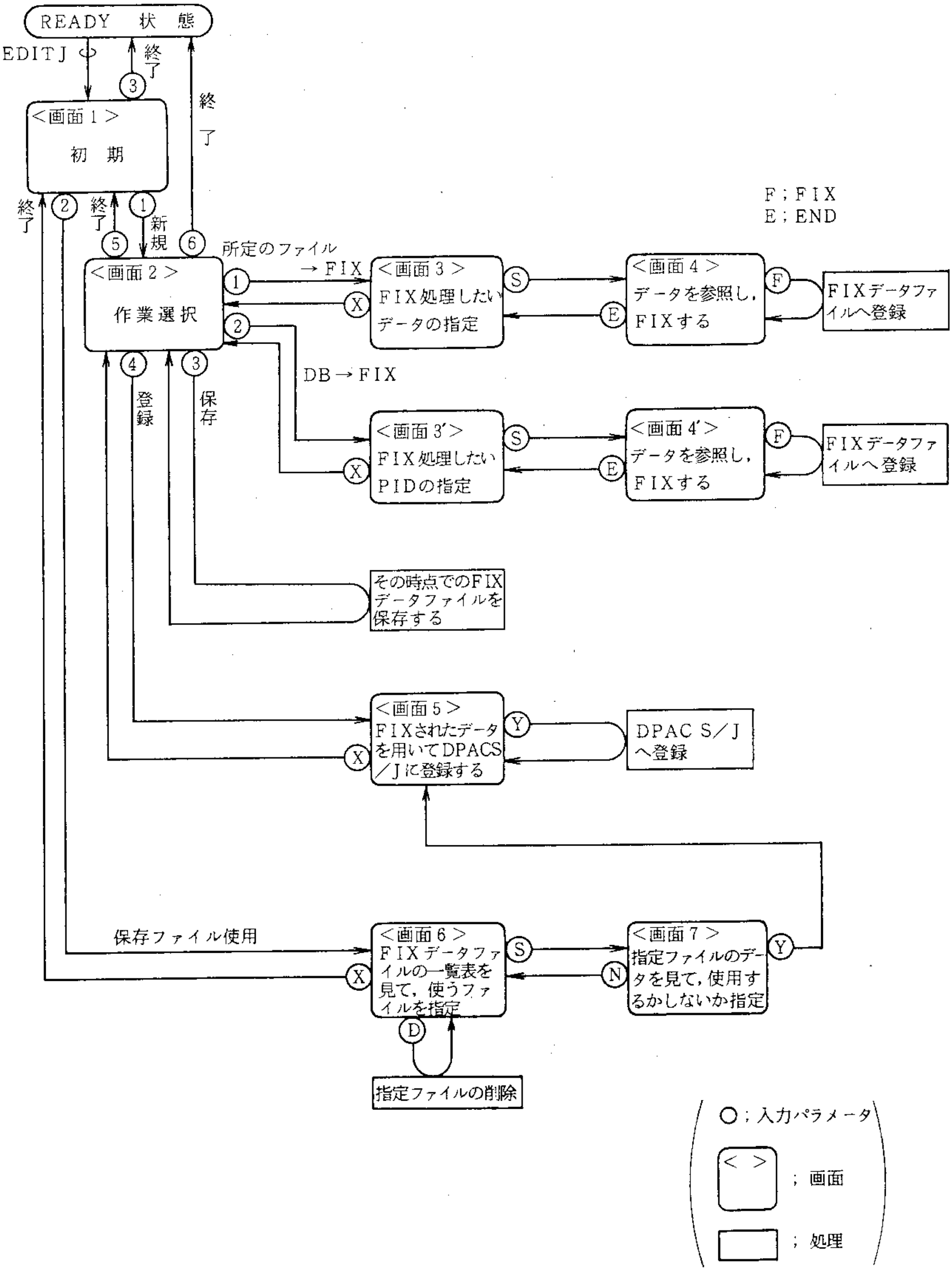
・キャラクタ端末用ＥＤＩＴＪの起動方法

```
EDITJ CHR
 TIME-10:24:26 CPU-00:00:02 SERVICE-31253 SESSION-00:01:13 JUNE 14,1989
 *********************************************************
 ***** JT-60 DATA EDITION START  (DATABASE TO DPACS) *****
 *********************************************************
 *** _
```

◆上記の両画面共，フルスクリーン表示となるので，ここで，リターンキーを押下。

<画面１>　初期選択

```
*****  ＪＴ－６０データ編集コード  ***  画面1  **

1. 新規登録
2. 保存 FIX DATA を使用しての登録 (RESTART)
3. 終了

        作業番号を指定してください。 □
```



第5.2図　EDITJ画面遷移と処理内容

◆作業選択を行う。

＜終了画面＞

```
**********************************************************
*****           JT-60 DATA EDITION END              *****
**********************************************************
TIME-10:28:19 CPU-00:00:02 SERVICE-30432 SESSION-00:00:40 JUNE 14,1989
READY
```

＜画面２＞　作業選択

```
  ＊＊＊＊＊  ＪＴ－６０データ編集コード  ＊＊＊  画面２  ＊＊

1. 指定ファイルからの FIX処理
       指定ファイル名    = [J3388.A399FVH.DATA]

2. DBからの FIX処理
       DB情報

3. FIX FILE の保存
       保存情報名        = [A399FVHD]

4. DPACS/J への登録

5. 終了

6. 完全終了

     作業番号を指定してください。 [ ]
```

◆作業選択を行う。ファイル名の指定を必要とする場合には，そのファイル名をＩＤ番号から入力する。指定したファイルが存在しない場合は，ＸＸ……Ｘが表示される。

＜画面３＞　所定のデータファイル→ＦＩＸ

＊＊＊＊＊　ＪＴ－６０データ編集コード　＊＊＊　画面３　＊＊

ファイル名 [J3388.A399FVH.DATA]　　波形数 [4]

FIX処理を行なうデータ名称の MARK欄に文字を入れて下さい。　Ｘ：終了
他：ＦＩＸ処理

MARK	NO.	データ名称	データ点数	チャンネル又は,PID NO.	ショットNO.	データ収集開始時間	サンプリングピッチ	変換係数 A	変換係数 B	単位略称
S	1	A399FVH F-DATA	3968	0		1.990D+00	1.000D-04	1.5	0.5	A
	2		3968	2		1.990D+00	1.000D-04			
	3		3968	1		1.990D+00	1.000D-04			
	4		3968	3		1.990D+00	1.000D-04			

◆所定のデータファイルからの入力の場合，データ名称，変換係数A，B（Ax＋B にて工学値計算），単位略称を本画面に入力する。

◆次にＭＡＲＫの欄に“＊”又は“Ｘ”以外の英数字を入れることにより，そのデータの確認とＦＩＸ処理を行う。<画面４>に移る。

◆<画面４>に於てＦＩＸ処理を行ったデータについてはＭＡＲＫ欄に“＊”が表示される。

◆ＭＡＲＫ欄に“Ｘ”を入力することにより本画面を終了する。

◆スクロールは，上（ＰＦ７），下（ＰＦ８），左（ＰＦ10），右（ＰＦ11）により行なわれる。

<画面４>　ＦＩＸ処理

＊＊＊＊＊　ＪＴ－６０データ編集コード　＊＊＊　画面４　＊＊

ファイル名 J3388.A399FVH.DATA

データ名称	データ点数	チャンネル又は,PIDNO.	ショットNO.	データ収集開始時間	サンプリングピッチ	変換係数 A	変換係数 B	単位略称
A399FVH F-DATA	3968	0		1.990D+00	1.000D-04	1.500D+00	5.000D-01	A

データ収集開始時間 ①　データ収集終了時間 ②　サンプリングピッチ ③

処理コマンドを指定して下さい。 FIX ④

時刻	データ内容							
1.990D+00	6.2D-01	6.0D-01	5.8D-01	5.6D-01	5.5D-01	5.8D-01	6.3D-01	6.2D-01
1.991D+00	5.9D-01	5.7D-01	5.9D-01	6.3D-01	6.5D-01	6.3D-01	6.1D-01	5.9D-01
1.992D+00	5.6D-01	6.7D-01	6.8D-01	6.6D-01	6.4D-01	6.1D-01	6.2D-01	6.5D-01
1.992D+00	6.9D-01	6.8D-01	6.6D-01	6.3D-01	6.2D-01	6.1D-01	6.5D-01	6.3D-01
1.993D+00	6.1D-01	5.8D-01	6.0D-01	6.2D-01	5.9D-01	5.7D-01	5.4D-01	5.4D-01
1.994D+00	5.6D-01	6.2D-01	6.0D-01	5.8D-01	5.5D-01	5.5D-01	5.0D-01	5.0D-01
1.995D+00	5.7D-01	5.9D-01	5.7D-01	5.7D-01	5.3D-01	5.7D-01	6.3D-01	6.2D-01
1.996D+00	6.0D-01	5.8D-01	5.6D-01	6.0D-01	6.5D-01	6.4D-01	6.2D-01	6.0D-01
1.996D+00	5.7D-01	5.4D-01	5.9D-01	6.5D-01	6.3D-01	6.0D-01	5.9D-01	5.4D-01
1.997D+00	5.8D-01	6.5D-01	6.5D-01	6.2D-01	6.0D-01	6.0D-01	6.4D-01	6.9D-01

◆<画面３>にて指定されたデータ内容が表示される。

◆①データ収集開始，②終了時間，③サンプリングピッチを設定できる。外挿は行なわないが，内挿はスプライン補間を行う。設定しない場合は，オリジナルと同じ値となる。なお，入力は浮動小数点形式（0.025 や 2.5 D－2）で行う。

◆④に処理コマンドを入力できる。

ＦＩＸ：ＦＩＸデータファイルへ登録する。

␣␣␣：①，②，③の指定に従って数値表示する。

ＥＮＤ：本画面を終了する。

◆入力数値が異常と見做された場合“ＥＲＲＯＲ…”を表示。

◆ＦＩＸ処理は，最大40であるのでこれを越えた時点で処理コマンド入力部に“ＭＡＸ”と表示され，以後ＦＩＸ処理は不可。

◆スクロールは上（ＰＦ７），下（ＰＦ８），左（ＰＦ１），右（ＰＦ11）により行なわれる。

<画面3'> DB→FIX

＊＊＊＊＊ ＪＴ－６０データ編集コード ＊＊＊ 画面３ ＊＊

ファイル名 ［　　　　］　　波形数 ［　　］

FIX処理を行なうデータ名称の MARK欄に文字を入れて下さい。　Ｘ：終了　他：ＦＩＸ処理

MARK	NO.	データ 名称	データ 点数	① チャンネル 又は,PID NO.	② ショット NO.	データ収集 開始時間	サンプリング ピッチ	変換係数 A	変換係数 B	単位 略称
*		PLSM CURR (FB)	5141	EB1625 I001	E004615	0.0	2.000E-03	1.582D+03	0.0	A

◆ＤＢからの入力の場合，この画面にて①にＰＩＤ番号，②にショット番号を設定してリターンキーを押下するとＤＢからバッファへの入力処理が行なわれ，図のような他の情報が表示される。

◆次にＭＡＲＫの欄に"＊"又は"Ｘ"以外の英数字を入れることにより，そのデータの確認とＦＩＸ処理を行う<画面4'>に移る。

◆<画面4'>に於てＦＩＸ処理を行ったデータについてはＭＡＲＫ欄に"＊"が表示される。

◆プレプログラム波形の場合に限り，サンプリングピッチにデータを設定することができる。default 値は 0.01 secである。

◆ＭＡＲＫ欄に"Ｘ"を入力することにより本画面を終了する。

◆スクロールは，上（ＰＦ７），下（ＰＦ８），左（ＰＦ10），右（ＰＦ11）により行なわれる。

<画面4'> ＦＩＸ処理

＊＊＊＊＊ ＪＴ－６０データ編集コード ＊＊＊ 画面４ ＊＊

ファイル名 ［　　　　］

データ 名称	データ 点数	チャンネル 又は,PIDNO.	ショット NO.	データ収集 開始時間	サンプリング ピッチ	変換係数 A	変換係数 B	単位 略称
PLSM CURR (FB)	5141	EB1625 I001	E004615	0.0	2.000E-03	1.582D+03	0.0	A

データ収集開始時間 ［　　］①　データ収集終了時間 ［　　］②　サンプリングピッチ ［　　］③

処理コマンドを指定して下さい。［　］

時刻	データ内容							
0.0	0.0	0.0	0.0	0.0	-1.6D+03	0.0	-1.6D+03	-1.6D+03
1.600D-02	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3.200D-02	2.5D+04	5.1D+04	6.5D+04	7.8D+04	9.2D+04	1.1D+05	1.2D+05	1.4D+05
4.800D-02	1.6D+05	1.8D+05	1.9D+05	2.1D+05	2.3D+05	2.5D+05	2.7D+05	2.9D+05
6.400D-02	3.1D+05	3.3D+05	3.6D+05	3.7D+05	4.0D+05	4.2D+05	4.4D+05	4.7D+05
8.000D-02	5.0D+05	5.2D+05	5.5D+05	5.8D+05	6.0D+05	6.2D+05	6.5D+05	6.8D+05
9.600D-02	7.1D+05	7.2D+05	7.5D+05	7.8D+05	8.1D+05	8.3D+05	8.4D+05	8.5D+05
1.120D-01	8.7D+05	9.0D+05	9.3D+05	9.5D+05	9.5D+05	9.6D+05	9.7D+05	9.9D+05
1.280D-01	9.9D+05	9.7D+05	9.7D+05	9.5D+05	9.4D+05	9.4D+05	9.3D+05	9.2D+05
1.440D-01	9.2D+05	9.1D+05	9.2D+05	9.2D+05	9.2D+05	9.3D+05	9.3D+05	9.3D+05

◆<画面3'>にて指定されたデータ内容が表示される。

◆①データ収集開始，②終了時間，③サンプリングピッチを設定できる。外挿は行なわないが，内挿はスプライン補間を行う。設定しない場合は，オリジナルと同じ値となる。なお，入力は浮動小数点形式（0.025 や 2.5 D－2）で行う。

◆④に処理コマンドを入力できる。

ＦＩＸ：ＦＩＸデータファイルへ登録する。

␣␣␣：①，②，③の指定に従って数値表示する。

ＥＮＤ：本画面を終了する。

◆入力数値が異常とみなされた場合"ＥＲＲＯＲ…"を表示。

◆ＦＩＸ処理は，最大40であるのでこれを越えた時点で処理コマンド入力部に"ＭＡＸ"と表示され，以後ＦＩＸ処理は不可。

◆スクロールは上（ＰＦ７），下（ＰＦ８），左（ＰＦ10），右（ＰＦ11）により行なわれる。

＜画面５＞　ＤＰＡＣＳ／Ｊ登録処理

＊＊＊＊＊　ＪＴ－６０データ編集コード　＊＊＊　画面５　＊＊

登録をする時はＹ、そうでない時はＸ　を指定して下さい。□

FIXED SCQ	編集領域	ARRAY 名	データ名称	データ点数	データ収集開始時間	データ収集終了時間	サンプリングピッチ
1			PLSM CURR (FB)	2100	0.0	4.198D+00	2.000D-03
2			PLSM HORI.POSI (FB)	2100	0.0	4.198D+00	2.000D-03

◆本画面にて，ＤＰＡＣＳ／Ｊに登録するデータを指定する。１回の登録では，１ベクトル量の時系列のみの登録となる。その際，ＡＲＲＡＹ名は，１種類とし，ベクトルの何番目にどのデータを割り付けるかを編集領域に指定する。さらに，上の指定欄にＹを入れてリターンキー押下することで登録を行う。同じデータを何度でも使用できる。

◆登録完了時には，該当する編集領域の欄に"＊＊"が入る。

◆編集領域欄に"？␣"が表示された場合には，次のどれかを意味する。

(i)　編集領域に入力した番号の並びがおかしい。

(ii)　編集領域に入力した番号が数値以外。

(iii)　同ＡＲＲＡＹ名の情報において，データ点数・サンプリングピッチが同一ではない。

◆スクロールは，上（ＰＦ７），下（ＰＦ８），左（ＰＦ10），右（ＰＦ11）により行なわれる。

＜画面６＞　ＦＩＸデータファイル名一覧表示

＊＊＊＊＊　ＪＴ－６０データ編集コード　＊＊＊　画面６　＊＊

FIX情報保存名に対応するMARK欄にＳ、又はＤを指定して下さい。　Ｓ：表示
Ｄ：削除

MARK	FIX FILE 名	保存年月日	保存時刻
S	SAVEA399	88-01-05	20:43

◆既に保存されているＦＩＸデータファイル名の一覧表を出力する。

◆ＤＰＡＣＳ／Ｊへの登録作業に使うもののＭＡＲＫ欄に“Ｓ”を入力する。ＭＡＲＫ欄に“Ｄ”を入力すれば，そのＦＩＸデータファイルは削除される。

◆ＭＡＲＫ欄に“Ｘ”を入力すれば，本画面を終了する。

◆スクロールは，上（ＰＦ７），下（ＰＦ８）で行なわれる。

＜画面７＞　保存ＦＩＸデータ確認

＊＊＊＊＊　ＪＴ－６０データ編集コード　＊＊＊　画面７＊＊

この内容でよければY，そうでなければN，削除の場合はD，及びNO.を指定して下さい。 D

FIX保存情報名 ： TEST　　NO. 02

データ名称	データ点数	チャンネル又は，PID NO.	ショットNO.	データ収集開始時間	サンプリングピッチ	変換係数 A	変換係数 B	単位略称	ファイル名	NO.
PLSM CURR (FB)	2100	EB1625 I001	E004615	0.0	2.000D-03	1.582D+03	0.0	A		1

◆＜画面６＞にて選択したＦＩＸデータファイルの内容が表示される。この画面にて，内容が良ければ，“Ｙ”，そうでなければ“Ｎ”を入力する。また，削除したいＰＩＤがあれば，“Ｄ”及び番号を入力する。

◆スクロールは，上（ＰＦ７），下（ＰＦ８），左（ＰＦ10），右（ＰＦ11）により行なわれる。

6. 導　入　編

ファイルを中心とした「制御システム設計・解析プログラム群」(DPACS/J)を導入するに当たって，予めいくつかの事柄について，その環境を整える必要がある。本章では，このプログラム群を導入する際の必要事項について記述する。

6.1 必要となるファイル

(1) DPACS/Jにおいて必要となるファイル

DPACS/Jにおいて必要となるファイルは，以下の4つである。

① 各サブシステム起動用「コマンド・プロシジャ・ファイル」

② 各サブシステムの「コマンド・ファイル」

③ 「計算・作図結果ファイル」

④ 「計算・作図結果出力用JCLファイル」

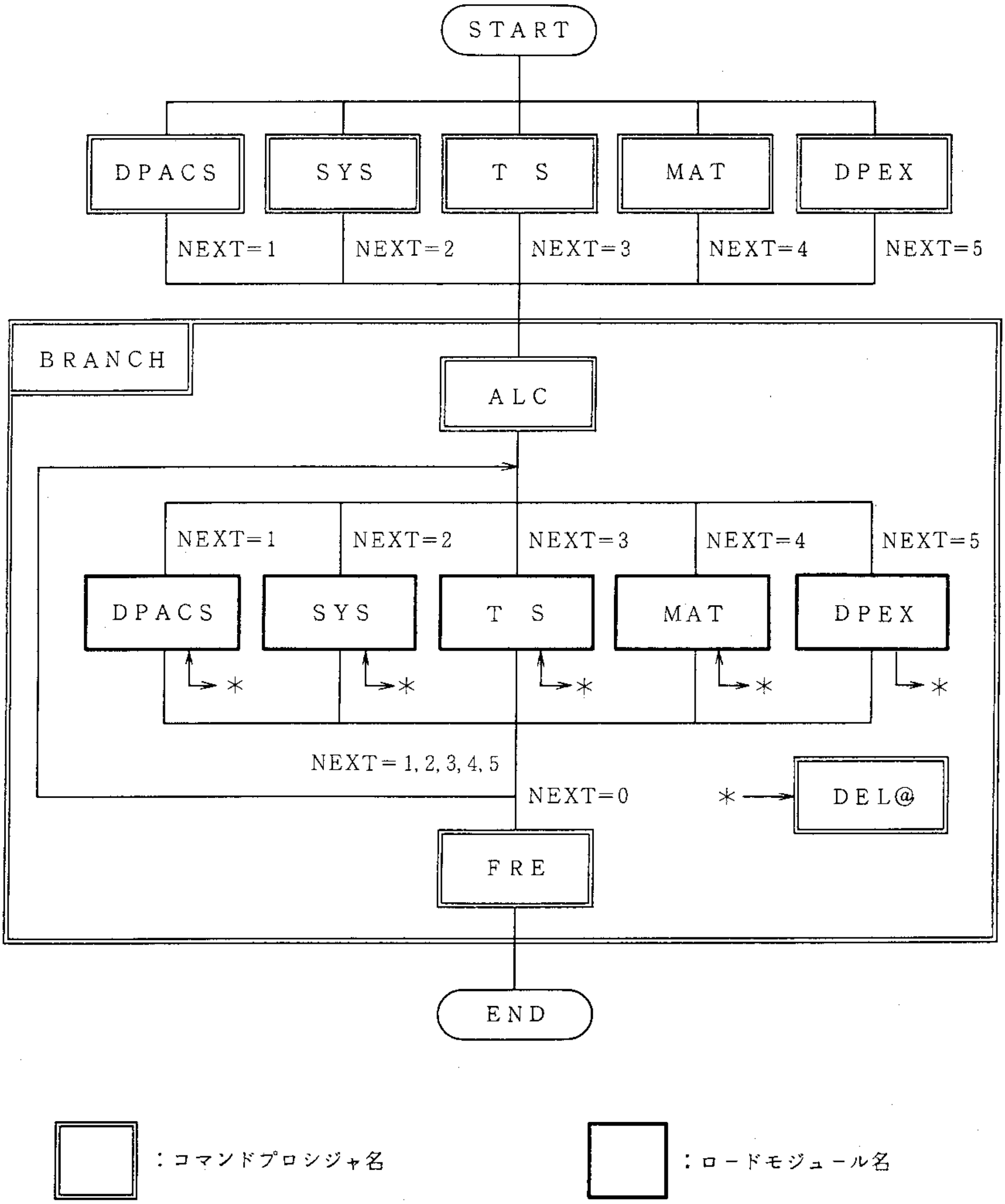
① コマンド・プロシジャ・ファイル

各サブシステムを起動するための一連のTSSコマンドを格納するファイルであり各ユーザの"TSSMAC.CLIST"ファイルのメンバーとして登録されている必要がある。

第6.1表にコマンド・プロシジャ・ファイルの構成，第6.1図にコマンド・プロシジャ間の関係，さらにそれ以降に各コマンド・プロシジャのコマンド列リストを示す。

第6.1表　コマンド・プロシジャ・ファイル

No.	メンバー名	機　能
1	DPACS	DPACSサブシステム起動
2	DPEX	DPEXサブシステム起動
3	SYS	SYSサブシステム起動
4	TS	TSサブシステム起動
5	MAT	MATサブシステム起動
6	ALC	共通ファイル割当
7	BRANCH	次サブシステム選択
8	FRE	共通ファイル解放・終了処理
9	@DEL	ファイル削除用(各サブシステム内で呼ばれる)



第 6.1 図　　コマンドプロシジャの体系

・各サブシステム起動用コマンドプロシジャ

```
  MEMBER NAME  DPACS

/**********************************************************/
/*** DESIGN PACKAGE FOR CONTROL SYSTEM / JAERI (DPACS/J) ***/
/*** DPACS SUBSYSTEM COMMAND PROCEDURE    (1989.06.01)   ***/
/**********************************************************/
PROC 0 KNJ DBG
GLOBAL NEXT PLT PIP
  IF &DBG = DBG THEN DO
      CONTROL MSG NOFLUSH LIST
    END
  ELSE DO
      CONTROL NOMSG NOFLUSH
    END
  FREEALL
  SET &NEXT = 1
  EXEC 'J3294.TSSMAC.CLIST(BRANCH)' '&KNJ &DBG'
EXIT

  MEMBER NAME  SYS

/**********************************************************/
/*** DESIGN PACKAGE FOR CONTROL SYSTEM / JAERI (DPACS/J) ***/
/*** SYS   SUBSYSTEM COMMAND PROCEDURE    (1989.06.01)   ***/
/**********************************************************/
PROC 0 KNJ DBG
GLOBAL NEXT PLT PIP
  IF &DBG = DBG THEN DO
      CONTROL MSG NOFLUSH LIST
    END
  ELSE DO
      CONTROL NOMSG NOFLUSH
    END
  FREEALL
  SET &NEXT = 2
  EXEC 'J3294.TSSMAC.CLIST(BRANCH)' '&KNJ &DBG'
EXIT

  MEMBER NAME  TS

/**********************************************************/
/*** DESIGN PACKAGE FOR CONTROL SYSTEM / JAERI (DPACS/J) ***/
/*** MAT   SUBSYSTEM COMMAND PROCEDURE    (1989.06.01)   ***/
/**********************************************************/
PROC 0 KNJ DBG
GLOBAL NEXT PLT PIP
  IF &DBG = DBG THEN DO
      CONTROL MSG NOFLUSH LIST
    END
  ELSE DO
      CONTROL NOMSG NOFLUSH
    END
  FREEALL
  SET &NEXT = 3
  EXEC 'J3294.TSSMAC.CLIST(BRANCH)' '&KNJ &DBG'
EXIT
```

MEMBER NAME MAT

```
/***********************************************************/
/*** DESIGN PACKAGE FOR CONTROL SYSTEM / JAERI (DPACS/J) ***/
/*** MAT   SUBSYSTEM COMMAND PROCEDURE    (1989.06.01)   ***/
/***********************************************************/
PROC 0 KNJ DBG
GLOBAL NEXT PLT PIP
  IF &DBG = DBG THEN DO
      CONTROL MSG NOFLUSH LIST
    END
  ELSE DO
      CONTROL NOMSG NOFLUSH
    END
  FREEALL
  SET &NEXT = 4
  EXEC 'J3294.TSSMAC.CLIST(BRANCH)' '&KNJ &DBG'
EXIT
```

MEMBER NAME DPEX

```
/***********************************************************/
/*** DESIGN PACKAGE FOR CONTROL SYSTEM / JAERI (DPACS/J) ***/
/*** DPEX  SUBSYSTEM COMMAND PROCEDURE    (1989.06.01)   ***/
/***********************************************************/
PROC 0 KNJ DBG
GLOBAL NEXT PLT PIP
  IF &DBG = DBG THEN DO
      CONTROL MSG NOFLUSH
    END
  ELSE DO
      CONTROL NOMSG NOFLUSH
    END
  FREEALL
  SET &NEXT = 5
  EXEC 'J3294.TSSMAC.CLIST(BRANCH)' '&KNJ &DBG'
  EXIT
```

・次サブシステム選択コマンドプロシジャ

```
/***********************************************************/
/*** DESIGN PACKAGE FOR CONTROL SYSTEM / JAERI (DPACS/J) ***/
/*** SELECT A NEXT SUBSYSTEM PROCEDURE    (1989.06.01)   ***/
/***********************************************************/
PROC 0 KNJ DBG
GLOBAL NEXT PLT PIP
IF  &DBG = DBG  THEN  DO
   CONTROL MSG NOFLUSH LIST
   END
ELSE DO
   CONTROL NOMSG NOFLUSH
END
/*** COMMON FILE ALLOCATE ***/
EXEC 'J3294.TSSMAC.CLIST(ALC)' '&KNJ &DBG'
```

```
/********************/
/****  SUB SYSTEM ***/
/********************/
START: +
DO WHILE &NEXT > 0 AND &NEXT < 6
/**********************/
/***  EXECUTE DPACS  ***/
/**********************/
   IF &NEXT  = 1 AND &STR(&KNJ) = &STR(KNJ) THEN DO
      CALL 'J3294.DPACS4.LOAD(MAIN)' 'FLIB(PROMPT=NO)'
      SET &NEXT = &LASTCC
      SET &PLT = &STR(PLT)
      SET &PIP = &STR(PIP)
   END
   IF &NEXT = 1 AND &STR(&KNJ) ¬= &STR(KNJ) THEN DO
      CALL 'J3294.DPACS4T.LOAD(MAIN)' 'FLIB(PROMPT=NO)'
      SET &NEXT = &LASTCC
      SET &PIP = &STR(PIP)
   END
/********************/
/***  EXECUTE SYS  ***/
/********************/
   IF &NEXT  = 2 THEN DO
      CALL 'J3294.SYS3.LOAD(MAIN)' 'FLIB(PROMPT=NO)'
      SET &NEXT = &LASTCC
   END
/*******************/
/***  EXECUTE TS  ***/
/*******************/
   IF &NEXT = 3 AND &STR(&KNJ) = &STR(KNJ) THEN DO
      CALL 'J3294.TS3.LOAD(MAIN)' 'FLIB(PROMPT=NO)'
      SET &NEXT = &LASTCC
      SET &PLT = &STR(PLT)
   END
   IF &NEXT = 3 AND &STR(&KNJ) ¬= &STR(KNJ) THEN DO
      CALL 'J3294.TS3T.LOAD(MAIN)' 'FLIB(PROMPT=NO)'
      SET &NEXT = &LASTCC

   END
/********************/
/***  EXECUTE MAT  ***/
/********************/
   IF &NEXT  = 4 THEN DO
      CALL 'J3294.MAT3.LOAD(MAIN)' 'FLIB(PROMPT=NO)'
      SET &NEXT = &LASTCC
   END
/*********************/
/***  EXECUTE DPEX  ***/
/*********************/
   IF &NEXT  = 5 THEN DO
      ALLOC DD(FT93F001) DA(DPEX.RSRT.DATA) SHR
      IF &LASTCC > 0 THEN DO
         ALLOC DD(FT93F001) DA(DPEX.RSRT.DATA) NEW CAT +
               SPA(10 1) TRA USING(C@)
         IF &LASTCC > 0 THEN DO
            WRITE F(FT93F001) DA(DPEX.RSRT.DATA) ALLOCATION FAILED
            SET &NEXT = 9
         END
      END
      ALLOC DD(FT96F001) DUMMY
      CALL 'J3294.DPEX2.LOAD(MAIN)','FLIB(PROMPT=NO)'
      SET &NEXT = &LASTCC
   END
```

```
END
/************************/
/***  END PROCEDURE  ***/
/************************/
FRE: +
   EXEC 'J3294.TSSMAC.CLIST(FRE)' '&KNJ &DBG'
ABNML: +
   IF &NEXT  > 5 THEN DO
      WRITE **** DPACS/J ABNOMAL END **** &LASTCC &MAXCC
   END
   IF &NEXT  < 0 THEN DO
      WRITE **** YOU CAN NOT EXECUTE BRANCH ! ****
   END
OWARI:EXIT Q
```

・共通ファイル割当・解放コマンドプロシジャ

```
  MEMBER NAME  ALC

/**************************************************************/
/*** DESIGN PACKAGE FOR CONTROL SYSTEM / JAERI (DPACS/J) ***/
/*** FILE ALLOCATION SUB-COMMAND PROCEDURE (1989.06.01)   ***/
/**************************************************************/
PROC 0 KNJ DBG
GLOBAL NEXT PLT PIP
IF &DBG = DBG THEN DO
   CONTROL MSG NOFLUSH LIST
END
ELSE DO
   CONTROL NOMSG NOFLUSH
END
   DEL CMDOUT.DATA
   DEL CMDIN.DATA
   DEL XRFXWF.DATA
   DEL E@1.DATA
   DEL E@2.DATA
   ATTR A@ BLKSIZE(3120) LRECL(80) RECFM(F B) DSORG(PS)
   ATTR D@ BLKSIZE(3510) LRECL(90) RECFM(F B) DSORG(PS)
   ATTR B@ RECFM(V B S)
   ATTR C@ BLKSIZE(6000) LRECL(150) RECFM(F B A) DSORG(PS)
   ALLOC DA(XRFXWF.DATA) DD(FT60F001) SP(2 1) T NEW CAT US(B@)
   ALLOC DA(CMDOUT.DATA) DD(FT91F001) SP(1 1) T NEW CAT US(D@)
   ALLOC DA(CMDIN.DATA) DD(FT90F001) SP(1 1) T NEW CAT US(D@)
   ALLOC DA(E@1.DATA) SP(2 1) T NEW CAT US(A@)
   ALLOC DA(E@2.DATA) SP(2 1) T NEW CAT US(A@)
   ALLOC DD(FT89F001) DA(GRDATA.DATA) SHR
   ALLOC DD(FT99F001) DA(PIPACKLP.DATA) OLD
EXIT

  MEMBER NAME  FRE

/**************************************************************/
/*** DESIGN PACKAGE FOR CONTROL SYSTEM / JAERI (DPACS/J) ***/
/*** FILE FREE SUB-COMMAND PROCEDURE        (1989.06.01)   ***/
/**************************************************************/
PROC 0 KNJ DBG
GLOBAL NEXT PLT PIP
IF  &DBG = DBG  THEN  DO
    CONTROL MSG NOFLUSH LIST
    END
```

```
 ELSE DO
   CONTROL NOMSG NOFLUSH
   END
FREEALL
SUBLIB
DEL  CMDOUT.DATA
DEL  CMDIN.DATA
DEL  XRFXWF.DATA
DEL  E@1.DATA
DEL  E@2.DATA
IF  &STR(&PLT) = &STR(PLT) THEN DO

        WRITENR ***OUTPUT PLOT TO NLP? (Y/N) ==>
        READ &YY
        IF &STR(&YY) = &STR(Y) THEN DO
           SUBMIT DPACS.CNTL(PLOT)
        END
    END
    IF  &STR(&PIP) = &STR(PIP) THEN DO
      WRITENR ***OUTPUT PIPK TO NLP? (Y/N) ==>
      READ &YY
        IF &STR(&YY) = &STR(Y) THEN DO
           SUBMIT DPACS.CNTL(COMPACT)
        END
    END
    LIB
    EXIT QUIT
```

・ファイル削除用コマンドプロシジャ

```
MEMBER NAME  DEL@

PROC 1 DSN
CONTROL NOMSG
DELETE &DSN
EXIT
```

② コマンド・ファイル

DPACS／JならびにDPACSBは各サブシステムにおいて使用できるコマンドを予め所定のコマンド・ファイルに格納している。各サブシステム毎のコマンド・ファイルを列挙すると以下の様になる。

第6.2表　コマンド・ファイル

No.	コマンド・ファイル名	使用サブシステム	REC	BLK	編成	スペース(TRK)
1	DCMDF. DATA	DPACS (PIPACK) DPACSB	80	3120	PS	1
2	MCMDF. DATA	MAT	80	3120	PS	1
3	SCMDF. DATA	SYS	80	3120	PS	1
4	XCMDF. DATA	TS	80	3120	PS	1
5	NCMDF. DATA	DPEX	80	3120	PS	1

③ 計算・作図結果格納ファイル

DPACS／JならびにDPACSBは，計算結果ならびに作図結果を格納するファイルとして，以下の2つのファイルを用意しておく必要がある。

第6.3表　計算・作図結果格納ファイル

No.	ファイル名	サブシステム	REC	BLK	編成	スペース(TRK)
1	GRDATA. DATA	DPACS TS	80	3200	PS	11
2	PIPACKLP. DATA	DPACS DPACSB	137	19043	PS FBA	初期量：30 増分量：10

④ 計算・作図結果出力JCLファイル

DPACS／JならびにDPACSBにおいて，計算結果ならびに作図結果を格納するファイルとして，先の2つのファイルを述べておいたが，これらを出力するためのJCLファイルも予め用意しておく必要がある。このJCLは，JCL．CNTLファイル（レコード長80バイトの区分データセット）の1メンバーとして登録されている必要がある。

第6.4表　計算・作図結果出力JCL

No.	メンバー名	出力対象ファイル名
1	PLOT	GRDATA. DATA
2	COMPACT	PIPACKLP. DATA

・計算結果出力JCL（COMPACT）の例

```
          MEMBER NAME  COMPACT

   T(2) W(4) C(2) I(3) NLP NOTIFY(JXXXX) MSGLEVEL(2,1,2),MSGCLASS=R
 /*JOBPARM K=0
//COMPACT   EXEC PGM=JRQCPRT,PARM='TYPE2'
//UTYIN    DD DSN=JXXXX.PIPACKLP.DATA,DISP=SHR
//UTYNLP   DD SYSOUT=*
//UTYLIST  DD SYSOUT=*
//
```

・作図結果出力ＪＣＬ（ＰＬＯＴ）の例

```
          MEMBER NAME  PLOT

 T(2) W(2) C(2) I(2) OPN GRP NOTIFY(JXXXX) MSGLEVEL(1,1,2) MSGCLASS(R)
// EXEC LMGO,LM='J1622.GROUT',PNM=GRPNL,SYSOUT=R
// EXPAND DISKTO,DDN=FT89F001,DSN='JXXXX.GRDATA'
// EXPAND GRNLP,SYSOUT=R
//SYSIN DD *
/*
//
```

（注意）　上記２つのＪＣＬをマニュアルにて実行させる場合には，ＰＦＤのＥＤＩＴ（データ編集）画面で，ＳＵＢＭＩＴ（ＳＵＢ）コマンドを発行する前にSUBLIBコマンドを発行しておく必要がある。尚，通常のDPACS／Ｊの終了処理に於ける実行の場合にはこれらの操作は行う必要がない。

(2) ＤＰＡＣＳＢで必要となるファイル

ＤＰＡＣＳＢにて必要となるファイルは起動用ＪＣＬファイルを除けば，DPACS／ＪのＤＰＡＣＳＢサブシステムにて用意してあるファイルで十分であるので，特に改めて用意する必要はない。従って，ここでは起動用ＪＣＬファイルについてのみ説明する。

この起動用ＪＣＬは，ＪＣＬ．ＣＮＴＬファイル（レコード長80バイトの区分データセット）の１メンバーとして登録されている方がユーザの管理上容易と思われる。なお，ＪＣＬのＬＩＳＴについては，4.2に掲載してあるので，そちらを参照されたい。

(3) ＥＤＩＴＪで必要となるファイル

ＥＤＩＴＪで必要となるファイルは，以下の２つである。

① ＥＤＩＴＪ起動用コマンドプロシジャ・ファイル

② 時系列データ管理ファイル（TSLISTF. DATA）

上記以外のファイルもＥＤＩＴＪにおいて使用しているが，最初にＥＤＩＴＪを動作させた場合に自動的に作成されるので利用者は特に意識する必要がない。

・ＥＤＩＴＪ起動用コマンドプロシジャ

```
  MEMBER NAME  EDITJ

PROC 0 CHR DBG
IF  &DBG ¬= DBG THEN DO
    CONTROL MSG FLUSH
END
ELSE DO
    CONTROL MSG LIST FLUSH
END
TIME
CONTROL NOMSG FLUSH
WRITE ************************************************************
WRITE ***** JT-60 DATA EDITION START  (DATABASE TO DPACS) *****
WRITE ************************************************************
FREEALL
DEL E@1.DATA
DEL E@2.DATA
ATTR A@ BLKSIZE(3120) LRECL(80) RECFM(F B) DSORG(PS)
ALLOC DA(E@1.DATA) SP(2 1) T NEW CAT US(A@)
ALLOC DA(E@2.DATA) SP(2 1) T NEW CAT US(A@)
IF  &CHR = CHR THEN DO
    ALLOC DD(MENULIB) DA('J3294.MNTE.DATA') SHR REU
END
ELSE DO
    ALLOC DD(MENULIB) DA('J3294.MNT.DATA') SHR REU
END
ALLOC DA(EDIT60LP.DATA) DD(FT62F001) SHR REU
CALL 'J3294.EDITJ.LOAD' 'FLIB(DFB=YES)'
WRITE ************************************************************
WRITE *****          JT-60 DATA EDITION END                  *****
WRITE ************************************************************
FREEALL
DEL  E@1.DATA
DEL  E@2.DATA
CONTROL MSG
TIME
EXIT
```

6.2 各パッケージの動作環境

DPACS／Jを端末上にて実行する場合に必要となる実行リージョンサイズは，約3.3Mバイトである。現在，一般ユーザに解放されているリージョンサイズが，2Mバイトであることから，DPACS／Jを使用するに当たっては，予め「TSSリージョン拡張願い」を，計算機室長に提出し，リージョンサイズが，4.0Mバイトまで使用できる様に許可を受ける必要がある。なお，EDITJの実行リージョン・サイズは約2Mバイトであるので，これだけを使用する場合には，拡張申請は必要ない。

また，DPACSBは実行並びにCPU時間の観点から，夜間JOBとなる可能性が大きいので，その実行に際しては，注意する必要がある。

LOGON時のリージョンサイズの指定方法

① DPACS／Jの場合

JCET010 SYSTEM READY
LOGON TSS JXXXX/YYYY S(4096)

② EDIT／Jの場合

JCET010 SYSTEM READY
LOGON TSS JXXXX/YYYY S(2048)

(注) JXXXXはユーザID，YYYYはパスワードを示す。

また，ファイルの初期化等については，全く不用であるので6.1節で述べたファイルを用意さえすれば直ちに利用可能状態となる。

7. 結　言

ＤＰＡＣＳ／Ｊは，その構想の開始された，昭和57年（1982年）から７年の時間を経て整備されたことになる。その間ＪＴ－60の建設とその完成，実験へと装置側のフェーズは変化したが，一方その解析手法や方法論について進歩したかというとどうもそのようではない。本コードの整備により，「はじめに」でも述べたように，プラズマ制御の方法論のうちの一部にある思想，即ち多変数制御理論に基礎を置く手法を提供できるものと確信している。しかし，プラズマ制御手法の解明は，まさにその出発点に立ったばかりであり，今後，プラズマそのものを表現する方法等を始め，研究しなければならないことは山積している。これらも含め全てを解明し，核融合発電の実現を志向しつつ結びとする。

謝　辞

ＤＰＡＣＳ／Ｊの核となるコードを提供して下さった東京工業大学教授　古田勝久氏，同助手小菅一弘氏に感謝の意を表します。また，原研在職中に，一連の研究の示唆を与えて下さった高エネルギー物理学研究所助教授　小方厚氏にお礼申し上げます。コードの導入に際し様々な角度から助言をして下さったＪＴ－60試験部ＪＴ－60第２試験室室長　近藤育朗氏，核融合研究部プラズマ実験研究室室長　前田彦祐氏，同部加熱工学第２研究室　志甫諒氏に感謝申し上げます。最後に，本報告書を仕上げるに当たり多大な便宜を図って下さった，ＪＴ－60試験部ＪＴ－60第１試験室室長　白形弘文氏，ＪＴ－60試験部部長　飯島勉氏，臨界プラズマ研究部部長　田村早苗氏にお礼申し上げます。

7. 結　言

DPACS／Jは，その構想の開始された，昭和57年（1982年）から7年の時間を経て整備されたことになる。その間JT－60の建設とその完成，実験へと装置側のフェーズは変化したが，一方その解析手法や方法論について進歩したかというとどうもそのようではない。本コードの整備により，「はじめに」でも述べたように，プラズマ制御の方法論のうちの一部にある思想，即ち多変数制御理論に基礎を置く手法を提供できるものと確信している。しかし，プラズマ制御手法の解明は，まさにその出発点に立ったばかりであり，今後，プラズマそのものを表現する方法等を始め，研究しなければならないことは山積している。これらも含め全てを解明し，核融合発電の実現を志向しつつ結びとする。

謝　辞

DPACS／Jの核となるコードを提供して下さった東京工業大学教授　古田勝久氏，同助手小菅一弘氏に感謝の意を表します。また，原研在職中に，一連の研究の示唆を与えて下さった高エネルギー物理学研究所助教授　小方厚氏にお礼申し上げます。コードの導入に際し様々な角度から助言をして下さったJT－60試験部JT－60第2試験室室長　近藤育朗氏，核融合研究部プラズマ実験研究室室長　前田彦祐氏，同部加熱工学第2研究室　志甫諒氏に感謝申し上げます。最後に，本報告書を仕上げるに当たり多大な便宜を図って下さった，JT－60試験部JT－60第1試験室室長　白形弘文氏，JT－60試験部部長　飯島勉氏，臨界プラズマ研究部部長　田村早苗氏にお礼申し上げます。

参 考 文 献

(1) 相川裕史，都築直久，木村豊秋，小方厚，二宮博正，鈴木康夫：「プラズマ位置・形状制御系における制御性能の検討」，JAERI－M7845（1978）

(2) A. Ogata, H. Ninomiya:"Optimal Feedback Control of Plasma Parameters in a Tokamak", Japanese Journal of Applied Physics, vol. 18 No. 4 (1979)

(3) R. E. Kalman, T. S. Englar:"A User's Manual for The Automatic Synthesis Program", NASA-CR-475 (1965)

(4) 「プラズマ制御コード開発に関する概念設計書」，(原研（担当筆者ら）が行った作業の報告書)（1983）

(5) 古田勝久，梶原宏之：「制御系のためのＣＡＤ」，計測と制御，vol.18 No.9（1979）

(6) 栗原研一，伊藤康浩，木村豊秋：「ＪＴ－60実験データベース構築(I)」，JAERI－M 87－097（1987）
伊藤康浩，栗原研一，木村豊秋：「ＪＴ－60実験データベース構築(II)」，JAERI－M 87－098（1987）

(7) 有本卓：「線型システム理論」，数理解析とその周辺1，産業図書（1979）

(8) 古田勝久：「線型システムの観測と同定」，コロナ社（1976）

(9) H. H. Rosenbrock:"State-Spase and Multivariable Theory", Nelson Press (1970)

(10) E. J. Davison:"A Method for Simplifying Linear Dynamic Systems", IEEE Transaction on Automatic Control, AC-11, pp. 93/101 (1966)

(11) 疋田，小山，三浦：「極配置問題におけるフィードバックゲインの自由度と低ゲインの導出」，計測自動制御学会論文集，11－5，556／560（1975）

(12) H. Kimura:"Pole Assignment by Gain Output Feedback", IEEE Transaction on Automatic Control, AC-13 pp. 748/749 (1968)

(13) M. Athans, P. L. Falb:"Optimal Control", McGraw-Hill Book Company (1966)

(14) K. Furuta, S. Kamiyama:"State Feedback and Inverse System", International Journal of Control, vol. 25 pp. 229/241 (1977)

(15) B. Gopinath:"On the Control of Linear Multiple Iuput-Output Systems", Bell System Technical Journal, vol. 50 pp. 1063/1081 (1971)

(16) R. A. Miller:"Specific Optimal Control of Linear Regulator Using a Minimal Order Observer", International Journal of Control, vol. 18, pp. 139/159 (1973)

(17) H. Kajiwara, K. Furuta:"Canonical Realization of Observers with Arbitrarily Assigned Poles for Linear Functions of the State", International Journal of Control, vol. 29, pp. 457/469 (1979)

(18) 有本卓：「カルマン・フィルター」，システムサイエンスシリーズ，産業図書（1976）

(19) E. J. Davison, A. Goldenberg:"Robust Control of a General Servomechanism Problem: The Servo Compensator", Automatica, vol. 11, pp. 461/471 (1975)

(20) K. Furuta, T. Nomura, H. Kajiwara:"Interactive Simulation Method for Integrated Linear Multivariable Systems", IEEE International Systems and Circuit Symposium, Tokyo (1979)

(21) H. H. Rosenbrock:"Computer-Aided Control System Design", Academic Press (1974)

(22) K. Furuta, et al. :"Structural Identification and Software Package for Linear Multivariable Systems", R. Isermann (Ed.), Identification and System Parameter Estimation, Pergamon Press, pp. 415/422, (1980)

(23) J. E. Potter:"Matrix Quardratic Solution", SIAM Journal on Applied Mathematics, vol. 14, pp. 496/501 (1966)

(24) D. R. Vaughan:"A Nonrecursive Algebraic Solution for Discrete Riccati Equation Computation", IEEE Transaction Automatic Control, AC-13, pp. 114/115 (1970)

(25) 柏木濶：「M系列再発見」，計測と制御 vol. 20, No. 2

(26) 日野幹雄：「スペクトル解析」，朝倉書店（1977）

(27) B. T. Smith, et al. :"Matrix Eigensystem Routines EISPACK Guide 2nd ed.", Lecture Notes in Computer Science, 6, Springer-Verlag (1976)

(28) 藤村統一郎，筒井恒夫：「EISPACK－J：固有値問題を解く副プログラムパッケージ」，JAERI－M8253（1979）

(29) B. S. Garbow, et al. :"Matrix Eigensystem Routines-EISPACK Guide Extension", Springer-Verlag, (1977)

(30) 児玉慎三，須田信英：「システム制御のためのマトリクス理論」，計測自動制御学会（1981）

(31) R. Guidorzi:"Canonical Structure in Identification of Multivariable Systems", Automatica, vol. 11, pp. 361/374 (1975)

(32) J. E. Ackerman:"On the Synthesis of Linear Control Systems with Specified Characteristics", Automatica, vol. 13, pp. 89/94 (1977)

(33) 明石一，今井弘之：「詳解制御工学演習」，共立出版（1981）

付録　コマンド入力形式一覧表、コマンド索引表

コマンド入力形式	機　　能	参　照
LIST {␣SN1 {␣SN2 {・・・} } }	システムおよびシステムデータの一覧出力	DP. I. 1
DELETE␣SN1 {␣SN2 {・・・} }	システムおよびシステムデータの削除	DP. I. 2
RENAME␣SN␣OSN	システムの名称変更	DP. I. 3
PROTECT [# ／U] ␣SN1 {␣SN2 {・・・} }	システムの保護	DP. I. 4
SYSIN ([／@]) [# ／C] ␣OSN　@=S, T, M, D, I	システムデータの入力	DP. I. 5
TYPE ([／@]) ([／N ／O]) ␣SN　@=S, T, M, D, I	システムデータの出力	DP. I. 6
REVISE ([／@]) ␣SN　@=S, T, M, D, I	システムデータの修正	DP. I. 7
COPY ([／@]) ␣SN␣OSN　@=S, T, M, D, I	システムデータの複写	DP. I. 8
PICK ([／@]) [# ／C] ␣OSN　@=S, T, M	空間の座標変換および入出力変数の選出	DP. I. 9
DUAL ([／@]) ␣SN␣OSN　@=S, T, M	双対システムの作成	DP. I. 10
COOD ([／1 ／2]) ␣SN	状態空間表現形式システムの正則変換	DP. II. 1
CMR ([／@ ([／C ／O]) ([／1 ／2]) ／D (∠C) ([／1 ／2])]) ␣SN　@=S, T, M	外部記述の正準最小実現の取得	DP. II. 2
TF (／S) ␣SN	伝達関数行列表現形式システムの取得	DP. II. 3
MRKF ([／@]) ␣SN　@=S, T, M	マルコフパラメータ表現形式システムの取得	DP. II. 4
DIF (／S) ␣SN	微分（差分）方程式表現形式システムの取得	DP. II. 5

コマンド入力形式	機　　　能	参　　照
IO([/@])␣SN　　@=S, D	入出力データ表現形式システムの取得	DP. II. 6
DIGIT(/D) $\begin{bmatrix} \# \\ /L \end{bmatrix}$ ␣SN␣OSN	連続系状態空間表現形式システムの離散系化	DP. II. 7
CONT␣SN␣OSN	離散系状態空間表現形式システムの連続系化	DP. II. 8
POLE $\begin{bmatrix} \# \\ /V \end{bmatrix}$ ␣SN	システムの極による安定判別	DP. III. 1
ZERO␣SN	システムの零点の算出	DP. III. 2
CCCO($\begin{bmatrix} /C \\ /O \end{bmatrix}$)␣SN	システムの可制御性・可観測性判別	DP. III. 3
RDCT $\begin{bmatrix} /E\left(\begin{bmatrix} /N \\ /P \end{bmatrix}\right) \\ /D \end{bmatrix}$ ␣SN␣OSN	システムの低次元化	DP. III. 4
EMR␣SN␣OSN	状態空間表現形式システムのε－最小実現取得	DP. III. 5
PLOC($\begin{bmatrix} /X \\ /Y \end{bmatrix}$)␣SN	極配置問題の求解	DP. IV. 1
OPT $\begin{bmatrix} \# \\ /S \begin{bmatrix} \# \\ /I \end{bmatrix} \end{bmatrix}$ ␣SN	最適制御問題の求解	DP. IV. 2
DCPL␣SN	非干渉化問題の求解	DP. IV. 3
OBS($\begin{bmatrix} /G \\ /M \\ /F \\ /K \end{bmatrix}$)␣SN␣OSN	オブザーバの構成	DP. IV. 4
MFS␣SN1␣SN2	モデル追従型サーボ問題の求解	DP. IV. 5
OSV␣SN	最適サーボ問題の求解	DP. IV. 6

コマンド入力形式	機　　能	参　照
LINK [#／N] [#／R] ␣SN1 {␣SN2 {・・・}} ␣OSN	複合線型（非線型）系の構成	DP. V. 1
CLPS ([／X／Y]) ␣SN␣OSN	閉ループ系の構成	DP. V. 2
SMLT [#／N] [#／U] ␣SN	シミュレーションの実行	DP. V. 3
NYQST [#／I] ␣SN	（逆）ナイキスト線図の作図	DP. VI. 1
BODE␣SN	ボード線図の作図	DP. VI. 2
LOCI␣SN	根軌跡の作図	DP. VI. 3
BAND [#／I] ␣SN	（逆）ゲシュゴリン帯の作図	DP. VI. 4
GLS␣SN	一般化最小2乗推定	DP. VII. 1
CML␣SN	条件付き最尤推定	DP. VII. 2
SML␣SN	対称型条件付き最尤推定	DP. VII. 3
LI␣SN	制限情報下最尤推定	DP. VII. 4
LIST {␣SN1 {␣SN2 {・・・}}}	システムおよびシステムデータの一覧出力	DX. I. 1
DELETE␣SN1 {␣SN2 {・・・}}	システムおよびシステムデータの削除	DX. I. 2
RENAME␣SN␣OSN	システムの名称変更	DX. I. 3
PROTECT [#／U] ␣SN1 {␣SN2 {・・・}}	システムの保護	DX. I. 4
INIT␣SN1␣SN2	状態変数ベクトル初期値算出	DX. II. 1
TRACK ([／S／T]) [#／R] ␣SN	トラッキング問題（任意目標入力）の求解	DX. II. 2

コマンド入力形式	機　能	参　照
V [／@]　　@=S, T, M, D	状態空間表現の可制御性・可観測性行列の取得	SY. I. 1
LCF	Vの可制御性構造の特徴抽出	SY. I. 2
SMAT	状態空間表現形式システム行列の取得	SY. I. 3
VV	可制御性・可観測性グラム行列の取得	SY. I. 4
RCT	定常リカッチ行列方程式の求解	SY. I. 5
FILT [／S, ／D]	入力時系列データに対する応答	SY. I. 6
LIST	時系列データの一覧出力	TS. I. 1
TS	時系列データの入出力・複写	TS. I. 2
REVISE	時系列データの修正	TS. I. 3
PLOT	時系列データのグラフ出力	TS. I. 4
PICK	時系列データの一部抽出	TS. I. 5
DCMP	時系列データの分解	TS. I. 6
APPEND	時系列データの結合	TS. I. 7
PRMT	時系列データの並べ替え	TS. I. 8
SKIP	時系列データのスキップ	TS. I. 9
ADD	時系列データ同士の加算	TS. II. 1
SADD	時系列データへの列ベクトルの加算	TS. II. 2
SUB	時系列データ同士の減算	TS. II. 3
SSUB	時系列データからの列ベクトルの減算	TS. II. 4
MUL	時系列データ同士の内積および行列との乗算	TS. II. 5
TSMUL	時系列データ同士の乗算	TS. II. 6
SMUL	時系列データと列ベクトルとの乗算	TS. II. 7

コマンド入力形式	機　　能	参　照
TSDIV	時系列データ同士の除算	TS. II. 8
SDIV	時系列データの列ベクトルによる除算	TS. II. 9
DIF	時系列データの差分値算出	TS. II. 10
INT	時系列データの積分値算出	TS. II. 11
LOG	時系列データの自然対数の算出	TS. II. 12
GAUSS	正規雑音時系列データの生成	TS. III. 1
MSEQ	最大周期時系列データ（M－系列）の生成	TS. III. 2
SIN	正弦波時系列データの生成	TS. III. 3
SQU	方形波時系列データの生成	TS. III. 4
SAW	鋸波時系列データの生成	TS. III. 5
TRI	三角波時系列データの生成	TS. III. 6
STAT	時系列データの各種統計量の計算	TS. IV. 1
CORF	時系列データの相関関数の算出	TS. IV. 2
FFT	時系列データの高速フーリエ変換の実行	TS. IV. 3
LIST	行列データの一覧出力	MT. I. 1
MAT	行列データの入出力・修正・複写	MT. I. 2
DCMP	行列データの分解	MT. I. 3
APPEND	行列データの結合	MT. I. 4
PRMT	行列データの並べ替え	MT. I. 5
MTCAL	行列データの各種演算を連続に行う	MT. II. 1
ADD	行列データ同士の加算	MT. II. 2
SUB	行列データ同士の減算	MT. II. 3
MUL	行列データ同士の乗算	MT. II. 4

コマンド入力形式	機　　能	参　照
SMUL	行列データのスカラー倍	MT. Ⅱ. 5
INV	行列データの（擬似）逆行列の算出	MT. Ⅱ. 6
TRNS	行列データの転置行列の算出	MT. Ⅱ. 7
FUNC	多項式関数値の算出	MT. Ⅱ. 8
EXP	指数関数値の算出	MT. Ⅱ. 9
IEXP	指数関数の積分値の算出	MT. Ⅱ. 10
RSLV	リゾルベント行列の算出	MT. Ⅱ. 11
LYP	リヤプノフ方程式の求解	MT. Ⅱ. 12
EIG	固有値問題の求解	MT. Ⅲ. 1
GEIG	一般化固有値問題の求解	MT. Ⅲ. 2
SVD	特異値分解の実行	MT. Ⅲ. 3
MAP	像空間、零化空間の基底の算出	MT. Ⅲ. 4
LSQ	最小2乗解の算出	MT. Ⅲ. 5
COOD	座標変換の実行	MT. Ⅲ. 6
VEC	ベクトル列の一次独立性の検証	MT. Ⅲ. 7
NORM	ノルム、行列式、トレースの算出	MT. Ⅲ. 8

コマンド索引表

ADD	TS. II. 1
ADD	MT. II. 2
APPEND	TS. I. 7
APPEND	MT. I. 4
BAND	DP. VI. 4
BODE	DP. VI. 2
CCCO	DP. III. 3
CLPS	DP. V. 2
CML	DP. VII. 2
CMR	DP. II. 2
CONT	DP. II. 8
COOD	DP. II. 1
COOD	MT. III. 6
COPY	DP. I. 8
CORF	TS. IV. 2
DCMP	TS. I. 6
DCMP	MT. I. 3
DCPL	DP. IV. 3
DELETE	DP. I. 2
DELETE	DX. I. 2
DIF	DP. II. 5
DIF	TS. II. 10
DIGIT	DP. II. 7
DUAL	DP. I. 10
EIG	MT. III. 1
EMR	DP. III. 5
EXP	MT. II. 9
FFT	TS. IV. 3
FILT	SY. I. 6
FUNC	MT. II. 8
GAUSS	TS. III. 1
GEIG	MT. III. 2
GLS	DP. VII. 1
IEXP	MT. II. 10
INIT	DX. II. 1
INT	TS. II. 11
INV	MT. II. 6
IO	DP. II. 6
LCF	SY. I. 2
LI	DP. VII. 4
LINK	DP. V. 1
LIST	DP. I. 1
LIST	DX. I. 1
LIST	TS. I. 1
LIST	MT. I. 1
LOCI	DP. VI. 3
LOG	TS. II. 12
LSQ	MT. III. 5
LYP	MT. II. 12
MAP	MT. III. 4
MAT	MT. I. 2
MFS	DP. IV. 6
MRKF	DP. II. 4
MSEQ	TS. III. 2
MTCAL	MT. II. 1
MUL	TS. II. 5
MUL	MT. II. 4
NORM	MT. III. 8
NYQST	DP. VI. 1
OBS	DP. IV. 4
OPT	DP. IV. 2
OSV	DP. IV. 7
PICK	DP. I. 9
PICK	DP. I. 5
PLOC	DP. IV. 1
PLOT	TS. I. 4
POLE	DP. III. 1
PRMT	TS. I. 8
PRMT	MT. I. 5
PROTECT	DP. I. 4
PROTECT	DX. I. 4
RCT	SY. I. 5
RDCT	DP. III. 4
RENAME	DP. I. 3
RENAME	DX. I. 3
REVISE	DP I. 7
REVISE	TS. I. 3
RLSV	MT. II. 11
SADD	TS. II. 2
SAW	TS. III. 5
SDIV	TS. II. 9
SIN	TS. III. 3
SKIP	TS. I. 9
SMAT	SY. I. 3
SML	DP. VII. 3
SMLT	DP. V. 3
SMUL	TS. II. 7
SMUL	MT. II. 5
SQU	TS. III. 4
SSUB	TS. II. 4
STAT	TS. IV. 1
SUB	TS. II. 3
SUB	MT. II. 3
SVD	MT. III. 3
SYSIN	DP. I. 5
TF	DP. II. 3
TRACK	DX. II. 2
TRI	TS. III. 6
TRNS	MT. II. 7
TS	TS. I. 2
TSDIV	TS. II. 8
TSMUL	TS. II. 6
TYPE	DP. I. 6
V	SY. I. 1
VEC	MT. III. 7
VV	SY. I. 4
ZERO	DP. III. 2