

作成日：2024 年 8 月 30 日

利用可能な電子ファイルは、次の URL アドレスで入手できます：

印刷されたコピーは、次の場所から入手可能です：

- NASA 航空宇宙情報センター
800 エルクリッジ・ランディング・ロード
リンシカム・ハイツ, MD 21090-2934
(301) 621-0390
- 国家技術情報サービス (NTIS)
5285 ポートロイヤル・ロード
スプリングフィールド, VA 22161-2171
(703) 487-4650

概要

本論文は、入力および出力データのシフトに基づく自己相関行列および相互相関行列から構成される情報行列の一般化バージョンを紹介するものである。データの相関の概念に基づき、入力および出力データから直接モデルを構築する新しいシステム実現アルゴリズムを開発した。このアルゴリズムは、情報行列から導出された特殊な相関行列の計算から始まる。この特殊な相関行列は、システムの可観測性行列および状態ベクトルの相関に関する情報を提供する。その後、可観測性行列と他の代数的操作を組み合わせてシステムモデルを構築する。このアプローチは、システムモデルを表現するために使用されるシステム行列の計算に対するいくつかの異なるアルゴリズムを導く。また、新しいアルゴリズムと、時間および周波数領域における他の実現アルゴリズムとの関係が、情報行列の行列因数分解を通じて確立される。いくつかの例を用いて、これらの新しいアルゴリズムの有効性と有用性を示す。

1. はじめに

近年、柔軟な構造物の能動制御、例えば音響ノイズの低減、ジッター誘発振動の抑制、および宇宙機アンテナの精密指向性制御などにおいて、システム同定が大きな支持を得ている。実際には、制御設計に基づく解析モデルが初めて使用される場合、解析モデルが指定された性能要求を満たすのに十分な精度を持たないことが多く、制御設計がうまく機能しない。この結果、多くのエンジニアは、制御パラメータの調整や入力および出力データからの正確な数学モデルの同定を行うための実験を行う。また、システムモデルの同定に加えて、ほとんどのロバスト制御手法では、モデル不確実性に関する情報も必要である。

システム同定は、多くのアプローチ、視点、および技術を含み [?, ?, ?, ?, ?, ?, ?, ?]、大きく 2 つのグループに分けられる。1 つのグループは、非パラメトリックアプローチ [?, ?, ?, ?, ?, ?, ?, ?] を使用し、最小二乗法を用いて入力-出力マップを決定する。入力-出力マップは、必ずしも物理的な意味を持たないシステムモデルで特徴付けられる。これらの技術は一般にブラックボックスアプローチと呼ばれ、反復処理を行わない。もう 1 つのグループは、パラメトリックアプローチ [?] を使用してシステムモデルパラメータを決定する。パラメータは、構造の剛性や質量などの物理量を表す場合がある。非線形数学的最適化技術を使用して、各パラメータの最適値を探索する。

パラメトリックおよび非パラメトリックアプローチの両方が多くの成功例を持つ。個々の技術の開発と強化が続けられている一方で、異なる手法を包括的かつ一貫した形で統一するための枠組みが依然として必要とされている。過去 10 年間で、研究者たちはシステム実現理論 [?] を通じて、異なる技術を統一するための枠組みを成功裏に提供してきた。

本論文は、参考文献 [?, ?, ?] で議論されたようなシステム同定技術の改善の必要性に基づいている。測定データが多い場合、計算時間および数値精度が問題となる。本論文の第 3 章では、参考文献 [?, ?] で提示されたデータ相関の概念を用いて、より効率的にシステム同定を行うための代替手順を説明する。“情報行列を用いたシステム実現” (SRIM) と呼ばれる新しいアルゴリズムが開発された。情報行列は、参考文献 [?] で定義された行列に類似している。周波数領域での解析は、時間領域または周波数領域でのシフトされた入力・出力データの相関に基づく一般的な形式を持つ。特別な相関行列が導入され、情報行列から計算される。この特別な相関行列は、出力がパルス入力によって生成された自由減衰応答である場合、パルス応答のシフトされたデータ相関に還元される。離散時間モデルを形成するために、SRIM アルゴリズムは、状態行列、入力行列、出力行列、および直接伝達行列を計算するための複数の異なる手法を含んでいる。

データ相関を用いた固有システム実現アルゴリズム (ERA/DC) は、パルス応答のシフトされたデータ相関を利用し、特異値分解を通じて相関行列を因数分解してシステム行列を得る。パルス応答は、パルス入力を用いるか、またはオブザーバ/Kalman フィルタ同定法 (OKID) を使用して入力と出力データから計算される。このため、SRIM アルゴリズムは、ERA/DC を拡張して、入力および出力データから直接システム同定を行うものと見なすことができる。

本論文の第 4 章では、SRIM アルゴリズムに基づいて開発された数学的枠組みが、異なる実現技術間の関係を確立

するためにどのように使用されるかを説明し、情報行列から異なるアルゴリズムが導出される方法を示す。情報行列の列空間および零空間の基底ベクトルは、異なる実現アルゴリズム間のリンクを提供する重要な要素である。情報行列の行列因数分解は、多くの異なる技術を統合するための理論的發展と計算手続きの中で最も重要なステップとして提示される。

本論文の各章は、背景情報から始まり、数値的および実験的な例で終わる。これらの例は、論文で提示されたアルゴリズムの有効性と有用性を示している。また、システム行列の計算における異なる技術の有効性を示すために、アルゴリズムの比較も行っている。

記号

- A : 状態行列, $n \times n$
- A_c : 連続時間状態行列, $n \times n$
- B : 入力影響行列, $n \times r$
- B_e : 連続時間入力影響行列
- B_i : B の i 番目の列, $n \times 1$
- C : 出力影響行列, $m \times n$
- C_a : 加速度計出力影響行列
- D : 直接伝達行列, $m \times r$
- D_i : D の i 番目の列, $m \times 1$
- 周波数応答関数, $m \times r$ (周波数変数 ω における)
- I : 次数 i の単位行列
- K : 剛性行列
- k : 時間インデックス
- k_i : i 番目のばね定数
- L : データ長
- M : 質量行列
- m : 質量
- n : システムの次数
- r : 入力数
- p : 出力数
- O : 観測可能性行列, $\mathbb{R}^{p \times n}$
- U : 左特異行列, $\mathbb{R}^{p \times p}$
- V : 右特異行列, $\mathbb{R}^{r \times r}$
- S : 特異値行列, $\mathbb{R}^{p \times r}$
- T_{xx} : 状態ベクトルの自己相関行列, $\mathbb{R}^{n \times n}$
- T_{uu} : 入力ベクトルの自己相関行列, $\mathbb{R}^{r \times r}$
- T_{xy} : 出力と入力の相互相関行列, $\mathbb{R}^{p \times r}$
- T_{yx} : 出力と状態の相互相関行列, $\mathbb{R}^{p \times n}$
- T_{yy} : 出力ベクトルの自己相関行列, $\mathbb{R}^{p \times p}$
- W : プロセスノイズベクトル, $\mathbb{R}^{n \times 1}$
- $X(k)$: 時刻 k における状態ベクトル, $\mathbb{R}^{n \times 1}$
- $U(k)$: 時刻 k における入力ベクトル, $\mathbb{R}^{r \times 1}$
- $Y(k)$: 時刻 k における出力ベクトル, $\mathbb{R}^{p \times 1}$
- Toeplitz 行列: Toeplitz 行列, $\mathbb{R}^{p \times r}$
- $U_p(k)$: 入力行列, $U(k)$ から $U(k + N - 1)$ までの行列, $\mathbb{R}^{r \times N}$
- $X(k, N)$: 状態ベクトルの行列, $X(k)$ から $X(k + N - 1)$ までの行列, $\mathbb{R}^{n \times N}$
- $Y_p(k)$: 出力行列, $Y(k)$ から $Y(k + N - 1)$ までの行列, $\mathbb{R}^{p \times N}$
- N : システムの最大次数を決定する整数
- k : 時刻インデックス
- $v(k)$: 時刻 k における特異値に対応するベクトル
- $i(k)$: 時刻 k における i 番目の入力
- $y(k)$: 時刻インデックス k における出力測定ベクトル, サイズ $m \times 1$
- $y_p(k)$: $y(k)$ から $y(k + p - 1)$ までを含むベクトル, サイズ $pm \times 1$
- Z_k : 周波数領域変数
- F : パラメータ行列
- E : 作業用行列

- O_i : 第 i の ARX 係数行列、サイズ $m \times m$ (出力ベクトル $y(k+i)$ に関連)
- $O_{i \times j}$: サイズ $i \times j$ のゼロ長方形行列
- $O_{i \times j}$: 第 i の ARX 係数行列、サイズ $m \times r$ (入力ベクトル $u(k+i)$ に関連)
- $O_{i \times j}(Z_k)$: 周波数変数 Z_k における入力ベクトル $u(Z_k)$ に関連する ARX 係数行列、サイズ $m \times r$
- σ_i : 第 i の特異値
- $O_{i \times j}(k)$: 周波数変数 Z_k における出力ベクトル $y(Z_k)$ に関連する第 i の ARX 係数行列、サイズ $m \times m$
- t : 作業用行列
- 0_i : 階数 i のゼロ正方形行列
- $0_{i \times j}$: サイズ $i \times j$ のゼロ長方形行列
- O^+ : 疑似逆行列

略語一覧:

- ARX: 自己回帰外因性モデル
- ERA/DC: 固有系実現アルゴリズム (データ相関付き)
- FRF: 周波数応答関数
- IDM: 間接法
- OEM: 出力誤差最小化法
- OKID: オブザーバ/カルマンフィルタ識別法
- SMI: 部分空間モデル同定法
- SRIM: 情報行列を用いたシステム実現法
- SV: 特異値

情報行矩陣を用いたシステム実現法 (SRIM)

この節では、入力から出力へのマップを特徴づけるシステム行列を計算するためのシステム実現アルゴリズムを示します。システム行列は状態行列 A 、入力行列 B 、出力行列 C 、および直接伝達行列 D で構成されます。

1 SRIM アルゴリズム

SRIM アルゴリズムは、シフトされた入力データと出力データの自己相関および相互相関行列から成る情報行列の計算を含みます。

2 状態空間モデルとアルゴリズムの開発

第3節では、離散時間状態空間モデルの記述から始まり、可観測性行列やシステム行列から形成されるトープリッツ行列などの重要な定義が提供されます。次に、システム行列 $[A, B, C, D]$ の計算に用いる状態空間モデルの実現の開発が続きます。その後、アルゴリズムの計算手順が示されます。最後に、シミュレーションと実験例が記述されます。

2.1 状態空間モデル

決定論的な線形時不変システムは、次の離散時間状態空間モデルで一般的に表現されます：

$$x(k+1) = Ax(k) + Bu(k), \quad (1)$$

$$y(k) = Cx(k) + Du(k), \quad (2)$$

ここで、 $x(k)$ は時刻インデックス k における $n \times 1$ の状態ベクトル、 $u(k)$ は r 個の入力に対応する $r \times 1$ の入力ベクトル、そして $y(k)$ は m 個のセンサ測定に関連する $m \times 1$ の出力ベクトルです。システム行列 A, B, C 、および D は未知であり、与えられた入力および出力データ、すなわち $k = 0, 1, 2, \dots, \ell$ における $u(k)$ と $y(k)$ から決定されます。

式 (2) は、さまざまな時間シフトに対して次のような行列形式で表現できます：

$$\begin{bmatrix} y(0) & y(1) & \cdots & y(\ell) \end{bmatrix} = \begin{bmatrix} \mathcal{O} & \mathcal{C}_u \end{bmatrix} \begin{bmatrix} x(0) \\ u(0) & u(1) & \cdots & u(\ell) \end{bmatrix}, \quad (3)$$

もし $k = 0, 1, \dots, \ell$ の $u(k)$ 、 $y(k)$ および $k = 0, 1, \dots, \ell+1$ の $x(k)$ が既知であれば、最小二乗法を用いて式 (3) から A, B, C 、および D を決定することができます。しかし、一般に状態ベクトル $x(k)$ の時系列は未知であり、推

定するか、可能であれば測定する必要があります。実際には、通常は $k = 1, \dots, \ell$ における入力系列および出力系列、つまり $u(k)$ と $y(k)$ のみが利用可能です。したがって、システム同定に使用するための他の形態のシステム方程式を見つける必要があります。

いくつかの代数的操作により、式 (2) は次のように導かれます：

$$\begin{bmatrix} y(k+1) \\ y(k+2) \\ \vdots \\ y(k+p) \end{bmatrix} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{p-1} \end{bmatrix} x(k) + \begin{bmatrix} D & CB & CAB & \cdots & CA^{p-2}B \\ 0 & D & CB & \cdots & CA^{p-3}B \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & D & CB \end{bmatrix} \begin{bmatrix} u(k) \\ u(k+1) \\ \vdots \\ u(k+p-1) \end{bmatrix}.$$

3 状態空間モデルの実現

ここでの p の選択については後で示すが、 p はシステムモデルのサイズ（行列 A の次元）に依存する整数である。 $y_p(k)$ 、 \mathcal{O}_p 、 $u_p(k)$ 、および \mathcal{T}_p を以下のように定義する。

$$y(k) = \begin{bmatrix} y(k) \\ y(k+1) \\ \vdots \\ y(k+p-1) \end{bmatrix} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{p-1} \end{bmatrix} [x(k)]$$

よって、式 (3) は次のようになる。

$$y_p(k) = \mathcal{O}_p x(k) + \mathcal{T}_p u_p(k)$$

ここで、 \mathcal{O}_p は次元 $pm \times n$ の可観測行列であり、 \mathcal{T}_p は次元 $pm \times pr$ の一般化されたトープリッツ行列である。行列 A 、 B 、 C 、および D が一意でなくとも、システムのマルコフパラメータ D および $CA^k B$ が一意であるため、 \mathcal{T}_p も一意であることに注意されたい。式 (3) を使用してシステムモデルを開発する方法は、セクション 3.2 で示されている。

4 状態空間モデルの実現

状態空間システム同定の目的は、未知の行列 A 、 B 、 C 、および D を求めることであり、それらは \mathcal{O}_p および \mathcal{T}_p の中に埋め込まれている。ひとつのアプローチは、既知の入力および出力データから \mathcal{O}_p および \mathcal{T}_p を計算することから始まる。 \mathcal{O}_p が計算されると、出力行列 C は \mathcal{O}_p の最初の m 行である。 $\mathcal{O}_p(m+1:pm, :)$ を、 \mathcal{O}_p の $(m+1)$ 行目から pm 行目までの最後の $(p-1)m$ 行と全ての列で構成される行列と定義する。同様に、 $\mathcal{O}_p(1:(p-1)m, :)$ を、 \mathcal{O}_p の最初の $(p-1)m$ 行と全ての列で構成される行列と定義する。

$$\mathcal{O}_p(m+1:pm, :) = \begin{bmatrix} CA^{p-2}B & \cdots & CB \\ CA^{p-3}B & \cdots & 0 \\ \vdots & \ddots & \vdots \\ CAB & \cdots & 0 \end{bmatrix}$$

次の等式を形成する。

$$\mathcal{O}_p(1:(p-1)m, :) = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{p-1} \end{bmatrix}$$

その後、状態行列 A を次のように計算できる。

$$A = \mathcal{O}_p(1:(p-1)m, :)\mathcal{O}_p(m+1:pm, :)$$

ここで、 t は擬似逆行列を意味する。整数 p は、次の条件を満たすように選ばなければならない。すなわち、次のようにして、次元が $(p-1)m \times n$ の行列 $\mathcal{O}_p(rn+1:pr, :)$ の階数が n 以上であることが必要である。

$$(p-1)m \geq n \quad \Rightarrow \quad p \geq \frac{n}{m} + 1$$

ここで、 n はシステムの次数を表す。同様に、行列 T_p (式 (4)) の最初の m 行と最初の r 列は、直接伝達行列 D を構成する。 $T(m+1:(p-1)m, 1:r)$ を、 T_p の最初の m 行と最初の r 列を削除して形成される行列として定義する。

$$T(m+1:(p-1)m, 1:r) = \text{行列}$$

式 (10) は次のようになる。

$$\begin{bmatrix} CB \\ CAB \\ \vdots \\ CA^{p-2}B \end{bmatrix} = O_p(1:(p-1)m, :)B$$

$$B = O_p^\dagger(1:(p-1)m, :)T(m+1:(p-1)m, 1:r)$$

O_p と T_p を決定するために、まずベクトル方程式 (式 (5)) を次のように行列方程式に拡張する。

$$Y_p(k) = O_p X(k) + \Theta U_p(k)$$

ここで、

$$X(k) = [x(k) \quad x(k+1) \quad \dots \quad x(k+N-1)]$$

$$Y_p(k) = \begin{bmatrix} y_p(k) & y_p(k+1) & \dots & y_p(k+N-1) \\ y(k+1) & y(k+2) & \dots & y(k+N) \\ \vdots & \vdots & \dots & \vdots \\ y(k+p-1) & y(k+p) & \dots & y(k+p+N-2) \end{bmatrix}$$

$$U_p(k) = \begin{bmatrix} u_p(k) & u_p(k+1) & \dots & u_p(k+N-1) \\ u(k+1) & u(k+2) & \dots & u(k+N) \\ \vdots & \vdots & \dots & \vdots \\ u(k+p-1) & u(k+p) & \dots & u(k+p+N-2) \end{bmatrix}$$

整数 N は、 $Y_p(k)$ および $U_p(k)$ の階数が少なくとも O_p の階数に等しくなるように十分に大きくなければならない。式 (12) は O_p および Θ_p を求めるための主要な方程式であり、データ行列 $Y_p(k)$ および $U_p(k)$ にのみ依存する。これらの行列から、システム行列 A 、 B 、 C 、および D を決定するために必要な情報を抽出する必要がある。

以下の量が定義される：

$$T_{lrp}(k)X_r(k)$$

$$N = \mathcal{L} - p, \quad (4)$$

ここで、 \mathcal{L} はデータの長さ、 p はデータのシフト量を表す。 T_{yy} 、 T_{uu} 、および T_{xx} は対称行列である。正方行列 T_{yy} ($rap \times rap$)、 T_{uu} ($rp \times rp$)、および T_{zz} ($n \times n$) は、それぞれ出力データ y 、入力データ u 、および状態ベクトル x の時間シフトを考慮した自己相関を表す。長方形行列 T_{yu} ($rap \times rp$)、 T_{yz} ($rap \times n$)、および T_{zu} ($n \times rp$) は、それぞれ出力データ y と入力データ u 、出力データ y と状態ベクトル x 、および状態ベクトル x と入力データ u の相互相関を表す。整数 N が十分に大きい場合、式 (14) で定義された量は、入力データと出力データがエルゴード性を満たす定常過程であれば、統計的な意味での期待値に近似される。

式 (14) を考慮すると、式 (12) に $U^T(k)$ を右から乗じ、その結果を N で除算することにより、

$$T_{yu} = \Theta_p T_{zu} + T_{uu}$$

$$T_p = (T_{yy} - \Theta_p T_{uu})^{-1} \quad (5)$$

行列 T_{uu}^{-1} が存在するのは、 p と N が選ばれ、 T_{uu} が少なくともランク rp を持つ場合に限る。同様に、式 (12) に $Y^T(k)$ を右から乗じると、

$$T_{yy} = \Theta_p T_{yu} + T_{yy}$$

また、式 (12) に $X^T(k)$ を右から乗じると、

$$T_{yx} = \Theta_p T_{xu} + T_{xx}$$

式 (16) で T_p を式 (17) および (18) に代入し、次に結果として得られる T_{yx} を式 (17) に代入すると、

$$T_{yy} - T_{yu}T_{uu}^{-1}T_{yu}^T = \Theta_p T_{xx} \Theta_p^T - \Theta_p T_{xu}T_{uu}^{-1}T_{zu}^T$$

ここで、以下を定義する：

$$\hat{T}_{hh} = T_{yy} - T_{yu}T_{uu}^{-1}T_{yu}^T$$

および

$$\hat{T}_{xx} = T_{zz} - T_{xz}T_{zu}^{-1}T_{zu}^T$$

式 (19) は次のようになります。

$$T_{hh} = \Phi T_{xx} - T \quad (22)$$

式 (22) は、システム行列 A と C を決定するための主要な式です。 T_{hh} は出力自己相関行列 T_{yy} から、入力自己相関行列 T_{uu} の逆行列で重み付けされた相互相関行列 T_{yu} の積を引いたものとして決定されます。 T_{hh} は、入力自己相関行列 T_{uu} が可逆である場合にのみ存在します。対称行列 T_{uu} は、入力信号 $u(k)$ が豊富で持続的である場合、すなわち行列 $\Phi_p(k)$ が満秩である場合に可逆です。このとき、 k 時点での状態ベクトル $x(k)$ と入力信号 $u(i)$ が $i \geq k$ において相関がないと仮定します。すなわち、現在および将来の入力データは現在の状態と相関がありません。この場合、相互相関行列 Φ_{zu} は $n \times r_p$ の零行列となり、行列 Φ_{zz} (式 (21)) は Φ_{xz} になります。例えば、入力 u が零平均の白色ランダムガウス信号である場合、データ長が十分に長いとき (理論上は $N \rightarrow \infty$)、 $\Phi_{xz} = \Phi_{zx}$ となります。

Φ を次のように定義します。

$$\Phi = L\Phi(k)$$

ここで、行列 Φ は情報行列として定義され、入力と出力データの相関行列 Φ_{yy} 、 Φ_{yu} 、および T_{uu} から形成されます。情報行列は、システム行列 A 、 B 、 C 、および D を計算するために必要なすべての情報を含んでいます。 Φ の因数分解により次が得られます。

$$\Phi = L\Phi T = L\Omega_{xp}, L\Omega_{p, xp}, \dots, \Omega_{p, yu}, T_{uu}^{-1}$$

ここで、 I_{pm} (または I_{pr}) は秩 prm (または pr) の単位行列であり、 $O_{pm \times pr}$ (または $O_{pr \times pm}$) は $prm \times pr$ (または $pr \times pm$) の零行列です。行列とその転置の積は、行列自体の秩に応じて、正半定または正定行列になります。したがって、式 (24) の左側の行列積は、正半定または正定行列です。式 (24) の右側の行列三重積では、左側の行列とその転置 (すなわち右側の行列) はいずれも満秩であるため、 T_{hh} も正半定または正定行列でなければなりません。

$$T_{hh} \quad (25)$$

$T_{uu} > 0$ (正定) である場合、 T^{-1} が存在するために必要です。

式 (22) の左側の対称行列 T_{hh} は、入力および出力データから知られていますが、右側は次元 $mp \times n$ の長方形行列 Φ_p 、次元 $n \times n$ の対称行列 T_{gzz} および Φ_p の転置の積で形成されます。行列 T_{ghh} を 3 つの行列に因数分解して可観測性行列 Φ_p を解く必要があることは明らかであり、このアプローチは第 3.2.1 節で採用されています。

3.2.1 A と C の計算

第 3.2.1 節では、 A および C を計算する 2 つの方法を示します。1 つの方法は、行列 T_{hh} を完全に因数分解するものであり、これを完全因数分解法と呼びます。もう 1 つの方法は、 T_{ghh} の一部を因数分解するものであり、部分因数分解法と呼びます。

3.2.1.1 完全因数分解法

入力および出力データから計算された行列 T_{hh} に対して、行列因数分解法は T_{hh} を 3 つの行列の積に因数分解することから始まります。特異値分解は、行列の因数分解を実行するための明らかな選択肢です。

対称行列 T_{Chh} の特異値分解を行うと、

$$T_{Ghh} = U_2 U^T \begin{bmatrix} \Lambda_n & 0 \\ 0 & 0 \end{bmatrix} U^T \quad (26)$$

となる。ここで、整数 $n_0 = pm - n$ は T_{Chh} の従属列の数を示し、 $O_{n \times n_0}$ は $n \times n_0$ のゼロ行列、 O_{n_0} は次元 n_0 の正方ゼロ行列である。 $pm \times n$ 行列 U_n は対角行列 Λ_n の n 個の非ゼロ特異値に対応し、 $pm \times n_0$ 行列 U_{n_0} は n_0 個のゼロ特異値に対応する。式 (22) と (26) を組み合わせると、

$$O \sim T_{hh}^T = U_n \Lambda_n U_n^T \quad (27)$$

となる。この最後の等式から、 O_p と O_z に対してそれぞれ 1 つの解が得られる：

$$O_p = U_n \quad (28)$$

および

$$O_z = \Lambda_n \quad (29)$$

式 (28) は、相関行列 T_{Lhh} から計算される $pm \times n$ 行列 U_n が、可観測性行列 O_p の表現であり、式 (8) を用いて出力行列 C と状態行列 A を解くために使用できることを示している。 U_n の最初の m 行が出力行列 C を構成する。

式 (29) は、相関 O_z (式 (21)) を相関行列 T_{hh} の特異値行列 Λ^2 として示す。入力 u がゼロ平均の白色雑音列である場合、相関 T_{zz} は O_z (式 (14)) に簡約され、これは状態ベクトル x の相関を表す。 Λ^2 の対角性は、状態ベクトル x のすべての個々の要素が線形に独立し、かつ直交していること（結合していないこと）を意味する。状態ベクトル $x(k)$ の各個別要素は 1 つの座標を表す。それぞれの座標の重要性は、対応する特異値の大きさによって測定される。

対角行列 Λ_n を次のように表す：

$$\Lambda_n = \text{diag}[\sigma_1, \sigma_2, \dots, \sigma_n] = \text{diag}[\sigma_1^2, \sigma_2^2, \dots, \sigma_n^2] \quad (30)$$

ここで、 σ_i ($i = 1, 2, \dots, n$) は単調に減少し、 $\sigma_1 > \sigma_2 > \dots > \sigma_n > 0$ を満たす。従って、状態ベクトル $x(k)$ の各要素（座標）の強度は特異値によって定量化される。特異値 $\sigma_{i+1}, \dots, \sigma_n$ が比較的小さく、系の情報よりも雑音情報を多く含むと仮定する。その結果、特異値 $\sigma_{i+1}, \dots, \sigma_n$ に対応する座標は他の座標と比べて無視できる。システムの次元は n から i に減少し、特異値 $\sigma_{i+1}, \dots, \sigma_n$ は削除される。

実際には、システムの不確実性や測定雑音の影響で、いずれの特異値もゼロにはならないため、 $n_0 = 0$ 、 $n = pm$ 、 $U_{n_0} = \emptyset$ (空集合) となり、特異値の切り捨てが行われない。可観測性行列 O_p (式 (28)) は正方行列となり、式 (27) で得られた U_n は $pm \times pm$ 行列となる。これにより、式 (9) の等式が破られ、式 (8) を用いて状態行列 A を解くことができないことを示している。

特異値がゼロである場合、 m 以上の最小特異値をゼロとみなし、完全分解法を使用する前に考慮する必要があります。言い換えれば、 \mathbf{D}_n の最後の m 列を切り取り、 \mathbf{D}_r として扱う必要があります。この問題を解決するために、別の方法が 3.2.1.2 節で提示されています。

3.2.1.2 部分分解法

整数 p がどのように選ばれたとしても、最小の n_o の値は m (出力の数) でなければならず、 $n < pr_o$ という関係を満たすために、これにより方程式 (27) の等式制約が満たされます。特異値の切り捨てを避ける唯一の方法は、 $pm \times pm$ の正方行列 \mathbf{T}_{4hh} の特異値分解を行う代わりに、次のように部分的に行列を因数分解することです。

$$\mathbf{n}_{hh}(:, 1:(p-1)m) = \mathbf{U}_2 \mathbf{V}^T = [\mathbf{U}'' \quad \mathbf{O}_{m \times n_o}] \mathbf{V} \quad (31)$$

$\mathbf{T}_{hh}(:, 1:(p-1)m)$ の次元は $pm \times (p-1)m$ であり、列よりも行が多いことを意味します。整数 n_o はゼロ特異値の数、および \mathbf{D}_o の列数を示します。 n'_o は \mathbf{U}_m の列に直交する \mathbf{D}_o の列数を表します。ノイズが含まれるデータでは、ゼロ特異値は存在せず、すなわち $n_o = 0$ です。特異値が切り捨てられなければ、 $n_o = m$ となります。いくつかの小さい特異値が切り捨てられると、 n_o は m と切り捨てられた特異値の数の合計になります。言い換えれば、 \mathbf{D}_n の列に直交する \mathbf{D}_o の列が少なくとも m あるということです。方程式 (27) から、次のように示すことが容易です。

$$\begin{aligned} \mathbf{T}_{hh}(:, 1:(p-1)m) &= \mathbf{O}_p \mathbf{T}_{4zx} \mathbf{O}_p^T - \mathbf{T}(:, 1:(p-1)m) \\ &= \mathbf{V} \mathbf{n}'_{11} - \mathbf{n}_2 \mathbf{V}^T \mathbf{n} \end{aligned}$$

これにより、次の方程式が得られます。

$$\mathbf{O}_p = \mathbf{I}_p \quad (33a)$$

および

$$\mathbf{O}_p^T \mathbf{n}_2 = \mathbf{E}_n \mathbf{V} \quad (33b)$$

方程式 (33b) は、方程式 (33a) が満たされている場合でも $\mathbf{T}_{zz} = \mathbf{E}_2$ または $\mathbf{O}_p^T \mathbf{T}(:, 1:(p-1)m) = \mathbf{V}^T$ であることを意味するものではありません。このアプローチの重要な特徴は、特異値の切り捨てがあってもなくても、 \mathbf{D}_o の列が常に十分に利用可能であり、 \mathbf{B} および \mathbf{D} を計算できることです。欠点は、特異値が状態ベクトルの相関をもはや表さないことです。

3.2.2 B および D の計算

\mathbf{A} および \mathbf{C} の計算と同様に、 \mathbf{B} および \mathbf{D} を計算するための方法は 3 つあります。最初の方法は間接法と呼ばれ、可観測性行列の列ベクトルに直交する列ベクトルを使用します。2 つ目の方法は、可観測性行列を直接利用するため、直接法と呼ばれます。3 つ目の方法は、計測された出力と再構築された出力との間の出力誤差を最小化します。再構

築された出力とは、入力履歴を使用して、計算された行列 **A**、**B**、**C**、および **D** で表される同定されたシステムモデルを駆動することによって得られる出力履歴です。

3.2.2.1 間接法

行列 **A** および **C** が既知の場合、入力行列 **B** および直接伝達行列 **D** は、トレトリッツ行列 \mathbf{T}_p (式 (4)) から計算できます。 \mathbf{T}_p を解くための方程式を定式化するには、可観測性行列 \mathbf{O}_p (式 (15)) に関連する項を除去する必要があります。

式 (26) および (28) を考慮し、式 (15) に \mathbf{U}_o^\top を左から掛け、 \mathbf{U}_n と \mathbf{U}_o の直交性を利用すると、以下の式が得られます。

上記の式に \mathbf{T}_p^{-1} を右から掛けると、次のようになります。

$$\mathbf{u}^\top \mathbf{q} - p = \mathbf{U}_o^\top \mathbf{T}_p^{-1} \quad (34)$$

式 (34) は、入力行列 **B** および直接伝達行列 **D** を解くために使用される基本的な式です。ただし、式 (34) は $\mathbf{T}_p - \mathbf{nU}_n^{-1}$ を意味するわけではありません。なぜなら、 \mathbf{U}_o^\top は次元 $n_o \times m_p$ の長方形行列であり、 $n_o < m_p$ だからです。式 (34) の右辺は既知の量であり、左辺には部分的に既知の **A** と **C** を含む \mathbf{T}_p と、部分的に未知の **B** と **D** を含む \mathbf{T}_p が含まれます。したがって、行列 \mathbf{T}_p は、**B** および **D** を抽出するために 2 つの部分に分割される必要があります。

行列 \mathbf{T}_p を以下のように分割します。

$$\mathbf{T}_p = \begin{bmatrix} \mathbf{T}_p(:, 1:r) & \mathbf{T}_p(:, r+1:2r) \\ \dots & \mathbf{T}_p(:, (p-1)r+1:pr) \end{bmatrix}$$

式 (4) (\mathbf{T}_p および \mathbf{q}_p の定義のため) および式 (28) から、次のようになります。

$$\begin{aligned} \mathbf{T}_p(:, 1:r) &= \mathbf{U}_n(1:(p-1)m, :) \mathbf{B} \\ \mathbf{T}_p(:, r+1:2r) &= [\mathbf{0}_{p \times r}] \quad \mathbf{U}_n(1:(p-2)m, :) \mathbf{B} \\ \mathbf{T}_p(:, 2r+1:3r) &= \mathbf{U}_n(1:(p-3)m, :) \mathbf{B} \\ \mathbf{T}_p(:, (p-1)r+1:pr) &= [\mathbf{0}_{(p-1) \times m}] \end{aligned} \quad (36)$$

ここで、 $\mathbf{0}_{i \times j}$ は次元 $i \times j$ の零行列です。 $\mathbf{U}_o^\top \mathbf{T}_p$ の積は次のようになります。

$$\begin{aligned} \mathbf{U}_o^\top \mathbf{T}_p(:, 1:r) &= \mathbf{U}_o^\top(:, 1:m) \mathbf{D} + \mathbf{U}_o^\top(:, m+1:pm) \mathbf{U}_n(1:(p-1)m, :) \mathbf{B} \\ \mathbf{U}_o^\top \mathbf{T}_p(:, r+1:2r) &= \mathbf{U}_o^\top(:, m+1:2m) \mathbf{D} + \mathbf{U}_o^\top(:, 2m+1:pm) \mathbf{U}_n(1:(p-2)m, :) \mathbf{B} \\ \mathbf{U}_o^\top \mathbf{T}_p(:, 2r+1:3r) &= \mathbf{U}_o^\top(:, 2m+1:3m) \mathbf{D} + \mathbf{U}_o^\top(:, 3m+1:pm) \mathbf{U}_n(1:(p-3)m, :) \mathbf{B} \\ \mathbf{U}_o^\top \mathbf{T}_p(:, (p-1)r+1:pr) &= \mathbf{U}_o^\top(:, (p-1)m+1:pm) \mathbf{D} \end{aligned} \quad (37)$$

$$\begin{bmatrix} \frac{1}{4} T : \mathbf{T}_p(:, 1:r) \\ \frac{1}{2} T : \mathbf{T}_p(:, r+1:2r) \\ \frac{U_o}{N_o} \dots \frac{U_T : \mathbf{T}_p(:, 2r+1:3r)}{U_T : \mathbf{T}_p(:, (p-1)r+1:pr)} \\ \frac{U_o^T(:, 1:m)}{U_S(:, m+1:2m)} \\ \frac{U_{lo}^T(:, 2m+1:3m)}{U_T(:, m+1:pm)} \\ \frac{U_{lo}^T(:, (p-1)m+1:pm)}{U_n(1:(p-1)m, :)} \\ \frac{U_{lo}^T(:, 2m+1:pm)}{U_n(1:(p-2)m, :)} \\ \frac{U_{lo}^T(:, 3m+1:pm)}{U_n(1:(p-3)m, :)} \end{bmatrix}$$

行列 L_{lo} の次元は $pno \times pr$ であり、行列 U_{on} の次元は $pno \times (m+n)$ です。式 (34) の右辺を

$$T^{-1} U_o T = N_o T P_{yu} U_o$$

と表すと、ここで U_{lo} は $no \times pr$ 行列です。式 (38) は、 U_o が次のように与えられることを示しています。

$$U_o = \begin{bmatrix} U_{lo}(:, 1:r) \\ \vdots \\ U_{lo}(:, (p-1)r+1:pr) \end{bmatrix}$$

さらに、行列 B と D は次のように計算されます。

$$[BD] = U_{on}U_{lo}$$

$U_{on}U_{lo}$ の最初の m 行が行列 D を形成し、最後の n 行が行列 B を形成します。

行列 B および D の式 (41) に対する最小二乗解は、行列 U_{on} が列よりも多くの行を持つ場合にのみ一意に存在します。 U_{on} の次元が $pno \times (m+n)$ であるため、整数 p は $pno > (m+n)$ を満たすように選択されなければなりません。ここで $no = pm - n$ であり、 n はシステムの次数です。例えば、 p が $(p-1)m + n$ と選ばれると、 no の最小要件は $no = m$ となり、システムの次数が $pno > (m+n)$ を満たすように決定される必要があります。特に、式 (30) の特異値 σ_n 以降、すなわち $\sigma_{n+1}, \dots, \sigma_{nm}$ がゼロでなく小さな値である場合に注意が必要です。

no が小さい場合、式 (41) から行列 B および D を計算することは時間的に効率的です。実際には、整数 no は小さな特異値を切り捨てることによって得られます。しかし、切り捨て誤差は、計算された B および D の結果にかなりの誤差をもたらす可能性があります。代替手法としては...

5 ゼロ特異値に関連する行列を使用しない B および D の計算方法

ゼロ特異値に関連する行列を使用せずに行列 B および D を計算する方法が、3.2.2.2 節に提示されている。

5.1 直接法 (3.2.2.2 節)

U_o を使用して式 (34) を導出する代わりに、直接法は可観測行列 O_p に依存して、行列 B および D を解くための式を形成する。直接法の導出方法は、間接法と同様のアプローチを使用している。

まず、式 (13) で定義された $X(k)$ と状態方程式 (式 (1)) を用いて、以下の式を形成する。

$$\begin{aligned} X(k+1) &= [x(k+1) \quad x(k+2) \quad \cdots \quad x(k+N)] \\ &= AX(k) + B\tilde{u}_N(k) \end{aligned} \quad (42) \quad (6)$$

ここで、 $\tilde{u}_N(k)$ は次のように定義される。

$$\tilde{u}(k) = [u(k) \quad u(k+1) \quad \cdots \quad u(k+N-1)] \quad (43) \quad (7)$$

式 (42) を式 (12) に代入すると、

$$\begin{aligned} Y_p(k+1) &= O_p X(k+1) + T_p U_p(k+1) \\ &= O_p AX(k) + O_p B\tilde{u}_N(k) \\ &\quad + T_p U_p(k+1) \\ &= O_p AX(k) \\ &\quad + [O_p B \quad T_p] \begin{bmatrix} \tilde{u}_N(k) \\ U_p(k+1) \end{bmatrix} \\ &= O_p AX(k) \\ &\quad + [O_p B \quad T_p] U_{p+1}(k+1) \end{aligned} \quad (44) \quad (8)$$

式 (12) から、 $X(k)$ に対する最小二乗解は次の通りである。

$$x(k) = O_p^{-1} [Y_p(k) - T_p U_p(k)] \quad (45) \quad (9)$$

式 (45) を $X(k)$ に代入すると、式 (44) は次のようになる。

$$\begin{aligned} Y_p(k+1) &- O_p A O_p^{-1} Y_p(k) \\ &- O_p A O_p^{-1} T_p U_p(k) \\ &+ [O_p B \quad T_p] U_{p+1}(k+1) \\ &= \{ [O_p B \quad T_p] \\ &\quad - [O_p A O_p^{-1} T_p \quad O_{p \times r}] \} \\ &\quad U_{p+1}(k+1) \\ &= \Phi U_{p+1}(k+1) \end{aligned} \quad (46) \quad (10)$$

ここで、 $O_{p \times r}$ は $p \times r$ のゼロ行列であり、 Φ は次のように定義される。

$$\Phi = [O_p B \quad T_p] - [O_p A O_p^{-1} T_p \quad O_{p \times r}] \quad (47) \quad (11)$$

注意すべき点として、 $O_p B$ は $p \times r$ 行列であるのに対し、 $O_p A O_p^{-1} T_p$ は $p \times pr$ 行列である。したがって、 $O_p B$ と $O_p A O_p^{-1} T_p$ の次元が異なるため、これらは直接加算することはできない。

直接一つの行列に加えることはできない。同様に、 $7p$ と $0pmxr$ を直接減算することもできない。ただし、 $[\mathbf{OpB} \ \mathbf{Tp}]$ または $[\mathbf{OpA} \ \mathbf{OtpTp} \ 0pmxr]$ は $pm \times (pr + r)$ 行列である。したがって、式 (46) に $u^{T+1}(k+1)$ を右から掛けると、

$$\begin{aligned} \mathbf{y_u}(m : (p+1)m, :) - \mathbf{OpA} \mathbf{Otpy_u}(l : pr, :) \\ = I_{uu} \end{aligned}$$

ここで、

$$I_{uu} = \frac{1_{U_{p+l}}(k+1)u^{T+1}(k+1)}{f}$$

となる。

次に、式 (49) に gtp を左から掛け、 $^{-1}$ を右から掛けると、

$$\begin{aligned} \mathbf{o_{yu}}(m+1 : (p+1)m, :) - \mathbf{AO_y}(l : pr, :) \\ = \mathbf{o_r} \end{aligned}$$

が得られる。ここで、 $OtpOp = I_n$ を適用した。入力と出力データが与えられ、行列 A と C が決定された場合、式 (50) の左側のすべての量は計算可能である。未知の行列 B と D は行列 F に埋め込まれている。式 (37) と同様に、 $//TTP$ の積を計算するために、次のように $\mathbf{O_F}$ を計算する：

$$\begin{aligned} \mathbf{OtpF} &= [\mathbf{B} \ \mathbf{otpq} - \mathbf{p}] \\ &\quad - [\mathbf{AOtpTp} \ 0pmxr] \end{aligned} \quad (51)$$

同様に、 $pm \times pr$ 行列 \mathbf{Tp} (式 (36)) の別の表現を使用して、次のように得られる：

$$\begin{aligned} \mathbf{Otp}(:, 1 : r) &= -\mathbf{AOtp}(:, 1 : m)\mathbf{D} \\ &\quad + \mathbf{B} - \mathbf{AOtp}(:, m+1 : pm)\mathbf{Op}(l : (p-1)m, :)\mathbf{B} \end{aligned}$$

$$\begin{aligned} \mathbf{Otp}(:, r+1 : 2r) &= [\mathbf{otpq}(:, 1 : m) - \mathbf{AOtp}(:, m+1 : 2m)]\mathbf{D} \\ &\quad + [\mathbf{Op}(:, m+1 : pm)\mathbf{Op}(:, l : (p-1)m) - \mathbf{AOtp}(:, 2m+1 : pm)\mathbf{Op}(l : (p-2)m, :)]\mathbf{B} \end{aligned}$$

$$\begin{aligned} \mathbf{otpp}(:, 2r+1 : 3r) &= [\mathbf{otpq}(:, m+1 : 2m) - \mathbf{AOtp}(:, 2m+1 : 3m)]\mathbf{P} \\ &\quad + [\mathbf{Otp}(:, 2m+1 : pm)\mathbf{Op}(:, l : (p-2)m) - \mathbf{AOtp}(:, 3m+1 : pm)\mathbf{Op}(l : (p-3)m, :)]\mathbf{B} \end{aligned}$$

$$\mathbf{otpr}(:, pr+1 : (p+1)r) = \mathbf{Otp}(:, (p-1)m+1 : pm)\mathbf{D}$$

式 (38) と同様に、式 (52) は次の行列形式に書き換えることができる：

$$\mathbf{Opr} = \mathbf{OpA} [\mathbf{D}]$$

$$\mathbf{Opt} = \mathbf{OpA} - \mathbf{o_r}(:, l-1)$$

$$\mathbf{O}_{lpr}(:, r+1 : 2r)$$

$$\mathbf{O}_{lprV}(:, 2r+1 : 3r)$$

$$\mathbf{o}_{tpr}(:, pr+1 : (p+1)r) - \mathbf{o_r}(:, 1 : r)$$

$$[\mathbf{o}(:, 1 : m) - \mathbf{AOtp}(m+1 : 2m, :)]$$

$$[\mathbf{O}_{tp}(:, m+1:2m) - \mathbf{A}\mathbf{O}_{tp}(2m+1:3m,:)]$$

$$\mathbf{otp}(:, (p-1)m+1:pm)$$

$$[-\mathbf{A}\mathbf{O}_i(:, m+1:pm)\mathbf{Op}(1:(p-1)m,:)]$$

$$[\mathbf{Opt}(:, m+1:pm)\mathbf{Op}(:, 1:(p-1)m) - \mathbf{A}\mathbf{O}_i(:, 2m+1:pm)\mathbf{Op}(1:(p-2)m,:)]$$

$$[\mathbf{Opt}(:, 2m+1:pm)\mathbf{Op}(:, 1:(p-2)m) - \mathbf{A}\mathbf{O}_{tp}(:, 3m+1:pm)\mathbf{Op}(1:(p-3)m,:)]$$

ここで、

\mathbf{I}_n は n 次の単位行列であり、

\mathbf{O}_n は n 次の零行列です。

\mathbf{Opt} は $pn \times r$ の行列であり、

\mathbf{OpA} は $(p+1)n \times (m+n)$ の行列です。

式 (50) の左辺を次のように定義します。

$$\mathbf{opA} : \mathbf{o}_i(m+1-(i+1)m,:) - \mathbf{A}\mathbf{O}_i(l:im,:) \quad (54)$$

式 (50) から次が導かれます。

$$\mathbf{o}(:, i+1:2i)$$

$$(55) \quad \mathbf{OpT}_c(:, 1:r)$$

$$\mathbf{OpTC}(:, pr+1:(p+1)r)$$

行列 \mathbf{B} および \mathbf{D} は、次の手順で求められます。

$\mathbf{O}_i\mathbf{A}\mathbf{O}_{pr}$ の上 m 行が行列 \mathbf{D} を構成し、下 n 行が行列 \mathbf{B} を構成します。

式 (56) は、行列 \mathbf{B} および \mathbf{D} に対して一意な最小二乗解を持ちますが、

\mathbf{OpA} の行数が列数より多い場合にのみです。

\mathbf{OpA} のサイズは $pn \times (m+n)$ であるため、整数 p は次の条件を満たす必要があります。

$$pn \geq (m+n), \text{ ただし、} n \text{ はシステムの次数です。}$$

直接法による \mathbf{B} および \mathbf{D} の計算は、一般的に間接法よりも計算量が多いです。たとえば、 $n = 10$, $p = 20$ の場合、

$$\mathbf{OpA} \text{ の行数は } (p+1)n = 210 \text{ になります。しかし、}$$

\mathbf{B} および \mathbf{D} を求めるのに \mathbf{OpA} のすべての行を使用する必要はなく、必要な行数は 6 です。

行列 \mathbf{OpA} および \mathbf{Opr} の行数は、ランクが $m-k-n$ より大きくなるように十分であるべきです。また、特にシステムの不確かさが大きい場合、 \mathbf{OpA} および \mathbf{Opt} の行数を増やすことで、解が改善される可能性があることにも留意すべきです。

間接法と直接法は、それぞれ方程式 (38) および (53) の誤差を最小化します。ただし、これらは実際の出力と再構成された出力との誤差を最小化することを意味するわけではありません。ここで、再構成された出力とは、計算された行列 \mathbf{A} 、 \mathbf{B} 、 \mathbf{C} 、 \mathbf{D} によって表される同定されたシステムモデルを駆動するために用いられる入力時間履歴と対応する出力時間履歴のことです。出力誤差を最小化する別の方法は、セクション 3.2.2.3 で示されています。

3.2.2.3 出力誤差最小化法

出力誤差最小化法は、まず出力方程式（式 (5)）の再整理から始めます。この再整理は、出力ベクトルを行列 \mathbf{B} および \mathbf{D} の要素に明示的に関連付ける線形方程式を構成するために行います。その後、最小二乗解を求めることで、実際の出力と再構成された出力との誤差を最小化します。

式 (5) において $p = N$ 、 $k = 0$ として、以下の式を得ます。

$$\mathbf{u}(0) = \mathbf{o}_N \mathbf{x}(0) + \mathbf{Z} \mathbf{v}(0) \quad (12)$$

行列 \mathbf{B} および \mathbf{D} は、式 (57) の右辺の $\mathbf{V} \mathbf{U}_p(0)$ の項に埋め込まれています。次に、式 (57) から行列 \mathbf{B} および \mathbf{D} を抽出する方法を示します。

行列 \mathbf{B} および \mathbf{D} の列ベクトルを次のように表します。

$$\mathbf{B} = [\mathbf{b}_1 \ \mathbf{b}_2 \ \dots \ \mathbf{b}_r] \quad (13)$$

$$\mathbf{D} = [\mathbf{d}_1 \ \mathbf{d}_2 \ \dots \ \mathbf{d}_r] \quad (14)$$

各列ベクトル \mathbf{b}_i ($i = 1, 2, \dots, r$) は状態ベクトルの長さ n の要素を持ち、各列ベクトル \mathbf{d}_i ($i = 1, 2, \dots, r$) は出力の数 m の要素を持ちます。次に、ベクトル \mathbf{b} および \mathbf{d} を次のように定義します。

$$\mathbf{b} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \vdots \\ \mathbf{b}_r \end{bmatrix} \quad (15)$$

$$\mathbf{d} = \begin{bmatrix} \mathbf{d}_1 \\ \mathbf{d}_2 \\ \vdots \\ \mathbf{d}_r \end{bmatrix} \quad (16)$$

列ベクトル \mathbf{b} は入力行列 \mathbf{B} のすべての列ベクトルを積み重ねた結果であり、列ベクトル \mathbf{d} は伝達行列 \mathbf{D} のすべての列ベクトルを含んでいます。同様に、入力ベクトル $\mathbf{u}(k)$ を次のように明示的に記述します。

$$\mathbf{u}(k) = \begin{bmatrix} u_1(k) \\ u_2(k) \\ \vdots \\ u_r(k) \end{bmatrix} \quad (17)$$

ここで、 $u_i(k)$ ($i = 1, 2, \dots, r$) はスカラーであり、 r は入力の数です。

与えられた行列 \mathbf{b} および \mathbf{d} （式 (59)）と T_N （式 (3)）に基づき、 $\mathbf{V}_{up}(0)$ は次のように書き直すことができる。

$$\mathbf{V}_{up}(0) = \begin{bmatrix} \mathbf{D} & \mathbf{CA} & \mathbf{CA}^2 & \dots & \mathbf{CA}^{N-2} & \mathbf{CA}^{N-3} & \dots & \mathbf{Dk}(N-1) \\ \mathbf{0}_{m \times n} & \mathbf{0}_{m \times n} & \dots & \mathbf{0}_{m \times n} & \mathbf{CA}^k & \dots & \mathbf{k}(k) & \end{bmatrix}$$

$$\begin{bmatrix} \mathbf{U}_m(1) & \mathbf{CAb}(0) & \mathbf{CU}_m(1) \\ \mathbf{l}_m(2) & \mathbf{l}_m(0) & \mathbf{l}_m(0) & \mathbf{0}_{D \times n} \\ \mathbf{l}_{U(N)} & \mathbf{CA}^{k-2} \mathbf{U}_m(k) & \mathbf{U}_k = 0 \end{bmatrix}$$

ここで、 $\mathbf{0}_{m \times n}$ は $m \times n$ のゼロ行列であり、

$$\mathbf{l}_m(k) = [\mathbf{l}_{m,1}(k) \ \mathbf{l}_{m,2}(k) \ \dots \ \mathbf{l}_{m,r}(k)] \quad (61)$$

$$\mathbf{U}_n(k) = [\mathbf{U}_{n,1}(k) \ \mathbf{U}_{n,2}(k) \ \dots \ \mathbf{U}_{n,r}(k)] \quad (62)$$

また、 \mathbf{I}_m および \mathbf{I}_n はそれぞれ m 次元および n 次元の単位行列である。付録 A には、式 (61) における和の項を計算する簡単な方法が示されている。行列 $\mathbf{U}_m(k)$ のサイズは $m \times nr$ であり、 $\mathbf{U}(k)$ のサイズは $n \times nr$ である。 $\mathbf{V}_{up}(0)$ の式を書き直す目的は、未知量 \mathbf{B} と \mathbf{D} を括弧外に移すことである（式 (61)）。

式 (62) を式 (57) に代入すると、

$$\mathbf{y}_N(0) = \Phi \mathbf{e} \quad (63)$$

ここで、

$$\Phi = \begin{bmatrix} \mathbf{I}_m(0) \\ \mathbf{CA}\mathbf{I}_m(1) \\ \mathbf{CA}^2\mathbf{I}_m(2) \\ \vdots \\ \mathbf{CA}^{N-1}\mathbf{I}_m(N-1) \\ \mathbf{0}_{m \times n} \\ \mathbf{I}_m(0) \\ \mathbf{CA}\mathbf{I}_m(1) + \mathbf{C}\mathbf{I}_m(0) \\ \vdots \\ \sum_{k=0}^{N-2} \mathbf{CA}^{k-2}\mathbf{I}_m(k) \end{bmatrix} \quad (64)$$

ベクトル \mathbf{e} のサイズは $(n + mr + nr) \times 1$ であり、行列 Φ のサイズは $mN \times (n + mr + nr)$ である。未知のベクトル \mathbf{e} は次の式で解くことができる：

$$\mathbf{e} = \Phi^+ \mathbf{v}_N(0) \quad (65)$$

ここで、 $+$ は擬似逆行列を表す。最小二乗法で得られる解 \mathbf{e} は、システムに入力ノイズと出力ノイズがある場合、実際には式 (63) を満たさない。しかし、 \mathbf{e} は実際の出力ベクトル $\mathbf{y}_N(0)$ と計算された出力ベクトル $\mathbf{y}_N(0) = \Phi \mathbf{e}$ との誤差を最小化する。最小二乗法で \mathbf{e} を解くことは非常に時間がかかることがある。なぜなら、行列 Φ の行数は m とデータ長 N の積に等しいからである。例えば、出力数 $m = 5$ のシステムでデータポイントが $N = 2000$ の場合、行数は最大で 10,000 にもなる。

SRIM アルゴリズムは、システム行列 A 、 B 、 C 、および D を計算するために開発された。このセクションでは、行列 A と C を計算する 2 つの方法、および行列 B と D を計算する 3 つの方法を紹介した。SRIM アルゴリズムのプログラム手順は、セクション 3.3 に示されている。

6 計算手順

SRIM アルゴリズムの計算手順を理解するために、計算ステップを以下に要約します。

1. システムの次数を n 、出力の数を m として、 $p > \frac{n}{m} + 1$ を満たす整数 p を選択します。
2. 相関行列 T_{Cyy} (次元 $pm \times pm$)、 T_{Cyu} (次元 $pm \times pr$)、および T_{Cuu} (次元 $pr \times pr$) を計算します (式 (14))。ここで、行列 $Y_p(k)$ の次元は $pm \times N$ 、 $U_p(k)$ の次元は $pr \times N$ です (式 (13))。 r は入力の数を表し、 k はシステム同定におけるデータの開始点です。 N は $L - k - p + 2 \geq N \gg \min(pm, pr)$ を満たすように選択する必要があります。ここで、 L はデータの長さです。
3. 相関行列 T_{hh} を計算します (次元 $pm \times pr$ 、式 (14))。具体的には、 $T_{hh} = T_{yy} - T_{yu}T_{uu}^{-1}T_{yu}^T$ です。
4. 完全分解法 (式 (26)) または部分分解法 (式 (31)) のために、 T_{hh} を特異値分解で分解します。
5. T_{hh} の特異値を調べ、システムの次数 n を決定し、次元 $prn \times n$ の \mathbf{U}_n (式 (26)) および次元 $pm \times no$ の \mathbf{U}_o を取得します。ここで、 $no = pm - n$ は切り捨てられた小さな特異値の数です。完全分解法の場合、 $pno \geq m + n$ を満たす必要があります。部分分解法の場合、 \mathbf{U}_n は \mathbf{U}_n (式 (31)) に置き換えられ、 no は m と切り捨てられた特異値の数の和になります。
6. $\mathbf{U}_n = \mathbf{O}_p$ または $\mathbf{U}_n = \mathbf{O}_p$ とし、状態行列 \mathbf{A} を式 (8) で求めます。出力行列 \mathbf{C} は \mathbf{U}_n の最初の m 行です。
7. 間接法の場合、 $T_{uo}^T = T_{uu}^T T_{ny} T_{ny}^{-1}$ (式 (39)) を計算し、 T_{uo} および T_{uo}^T (式 (38) および (40)) を構築します。入力行列 \mathbf{B} と直接伝達行列 \mathbf{D} を式 (41) から求めます。 \mathbf{D} は $T_{uo}T_{uo}^T$ の最初の m 行で構成され、 \mathbf{B} は最後の n 行で生成されます。
直接法の場合、 $\mathbf{O}_p \mathbf{A}$ を式 (53) から構築し、 $\mathbf{O}_p \mathbf{A} \mathbf{O}_p^T$ を計算して \mathbf{B} および \mathbf{D} を解きます。 $\mathbf{O}_p \mathbf{A} \mathbf{O}_p^T$ の最初の m 行が行列 \mathbf{D} を形成し、最後の n 行が行列 \mathbf{B} を生成します。
出力誤差最小化法の場合、 $\mathbf{Y}_N(0)$ および \mathbf{Z} を式 (64) から構築し、 $\mathbf{Z} \mathbf{Y}_N(0)$ を計算して \mathbf{B} および \mathbf{D} を解きます。 $\mathbf{Z} \mathbf{Y}_N(0)$ の最初の n 要素が初期状態ベクトル $\mathbf{x}(0)$ を形成し、次の mr 要素が \mathbf{D} の r 列ベクトルを与え、最後の nr 要素が \mathbf{B} の r 列ベクトルを生成します。
8. 実現された状態行列の固有値と固有ベクトルを求め、実現されたモデルをモーダル座標に変換して、システムの減衰率と周波数を計算します。このステップは、モーダルパラメータの同定が必要な場合のみ行います。
9. モード特異値を計算し (参考文献 1)、システムモードと雑音モードを区別します。このステップは、モーダル切り捨てによるモデル削減の手法を提供します。

計算手順は、出力データがインパルス応答時間履歴である場合、ERA/DC 法の手順に簡略化されます (参考文献 1)。システムを時間ステップゼロで励起するためにパルスを与えると仮定します。ステップ 2 では、 $k = 1$ とします。この場合、相関行列 T_{yu} と T_{uu} はゼロとなり、 $T_{hh} = T_{yy}$ が得られます。理論的には、 $T_{hh} = T_{yy} - T_{yu}T_{uu}^{-1}T_{yu}^T$ の式は、 T_{uu} が可逆でない場合、 T_{hh} の計算には使用すべきではありません。自由減衰やパルス応答などの特殊な場合には、入力信号が消滅する点で整数 k を選択すると、 T_{hh} は T_{yy} に簡略化されます。

質量 1 質量 2 質量 3

力 1 力 2 力 3

SRIM アルゴリズムは、再帰的な式を用いた計算ステップを組み込むことで、サブスペースモデル同定 (SMI) 手法よりも計算効率が高い [?, ?, ?]。SMI 手法では、大きな行列 $[U_k \ Y_k]$ の QR 分解 [?, ?]、特異値分解、その後に過剰決定系の方程式を解く必要がある。さらに、提案手法はデータ相関の概念を用いることで、SMI 手法よりも物理的な理解が深まる。

3.4. 例

SRIM アルゴリズムを説明するために、数値例と実験例を示す。数値例は 3 自由度の質量・ばね・ダンパ系を使用し、実験例はラングレー研究センターで試験されたトラス構造を使用する。システム行列の計算方法の違いについて、システムの固有周波数および減衰に関連して比較・議論する。

3.4.1. 数値例

図 1 は、質量 m_i 、減衰係数 c_i 、ばね定数 k_i ($i = 1, 2, 3$) からなる質量・ばね・ダンパ系を示している。 $w_i(t)$ はそれぞれの質量の絶対変位、 u_i は入力力である。このシステムの二次微分方程式は以下の通りである：

$$M\ddot{w} + C\dot{w} + Kw = Bu \quad (18)$$

ここで、

$$M = \begin{bmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{bmatrix}, \quad C = \begin{bmatrix} c_1 + c_2 & -c_2 & 0 \\ -c_2 & c_2 + c_3 & -c_3 \\ 0 & -c_3 & c_3 \end{bmatrix}, \quad K = \begin{bmatrix} k_1 + k_2 & -k_2 & 0 \\ -k_2 & k_2 + k_3 & -k_3 \\ 0 & -k_3 & k_3 \end{bmatrix}, \quad w = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}$$

これに対応する一階微分方程式は以下の通りである：

$$\dot{x} = A_c x + B_c u \quad (19)$$

ここで、

$$x = \begin{bmatrix} w \\ \dot{w} \end{bmatrix}, \quad A_c = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}C \end{bmatrix}, \quad B_c = \begin{bmatrix} 0 \\ M^{-1} \end{bmatrix}$$

質量 $m_1 = m_2 = m_3 = 1$ とし、ばね定数 $k_1 = 1, k_2 = 2, k_3 = 3$ 、減衰係数 $c = 0.05\sqrt{k}$ とする。ゼロ行列 $\mathbf{0}_a$ は 3 次のゼロ行列、 \mathbf{I}_3 は 3 次の単位行列、 $\mathbf{0}_{3 \times 1}$ は 3×1 のゼロベクトルである。減衰行列 $\mathbf{E} = 0.05 \times \sqrt{k}$ は、各モードの比例減衰を 0.5

$$\mathbf{Y} = \begin{bmatrix} 0 & 0 & \sqrt{2} \\ 1 & 0 & \sqrt{3} \end{bmatrix} \mathbf{W}$$

ここで、

$$\mathbf{W} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

測定ベクトル \mathbf{y} は \mathbf{W} を代入すると次のようになる。

$$\mathbf{y} = \mathbf{C}\mathbf{A}\mathbf{M}^{-1}(\mathbf{b}u - \mathbf{E}v - \mathbf{K}w) = \mathbf{C}x + \mathbf{D}u$$

ただし、

$$\mathbf{C} = -\mathbf{C}\mathbf{A}\mathbf{M}^{-1} < \mathbf{K}$$

離散状態空間モデルは次のように表される。

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}u(k) + \mathbf{u}(k)$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{D}u(k) + \phi(k)$$

ここで、

$$\mathbf{A} = 10^{-2} \times \begin{bmatrix} -4.0315 & 47.981 & 17.092 & -133.02 & 19.188 & 54.697 \\ 0.72516 & 9.6335 & 39.563 & 3.9627 & 33.292 & 62.440 \\ 47.981 & 17.092 & 59.137 & 22.195 & 3.9627 & -26.375 \\ 71.972 & 22.195 & 42.886 & 33.292 & 71.972 & 10.211 \\ 3.9627 & 33.292 & 62.440 & 19.188 & 54.697 & -4.8427 \\ 47.944 & 17.343 & -70.165 & 28.782 & 47.944 & -26.773 \\ 71.916 & 28.782 & -87.442 & 17.343 & 71.916 & 9.6095 \end{bmatrix}$$

$$\mathbf{B} = 10^{-2} \times \begin{bmatrix} -300 & 200 & -500 \\ 200 & 300 & -1.6119 \\ 0.60778 & -1.9492 & 0.60778 \\ 0.18028 & 0.91167 & 0.18028 \end{bmatrix}$$

ここで k は時間インデックスである。 $u(k)$ と $\phi(k)$ はそれぞれプロセスノイズと測定ノイズを表し、一般的にこれらはランダムな正規分布に従うと仮定される。プロセスノイズは入力外力の約 10%、測定ノイズは出力外力の約 10% の標準偏差比に設定される。初期値として p は任意に $p = 6, 12, 25, 50$, および 100 に設定され、最大システム次数 $p_{\max} = 12, 24, 100$, および 200 とされており、これは予測されるシステム次数よりも高くなっている。

$n = 6$ のシステムの例を示す。状態行列 A と出力行列 C は部分分解法によって算出され、入力行列 B と直達伝達行列 D は間接法および出力誤差最小化法を用いて算出される。

実際の応用では、正確なシステム次数は不明であり、選択された最大システム次数は上限値を提供し、解法の進行を可能にする。この上限値から、次数を 6 に削減するための 2 つの方法がある。1 つは特異値分解による方法であり、 $TChh$ の特異値を大きい順に 6 つ残すことでシステム行列を次のように得る： A は 6×6 、 B は 6×1 、 C は 2×6 、 D は 2×1 の行列となる。システム行列 A の固有値から周波数と減衰比が導かれ、行列 C と B によりモード形状とモード参加係数が推定される（文献 1 参照）。

表 ?? は、真のモード周波数と減衰比を示す。表 ?? は、異なる p 値に対して識別された減衰比を示す。識別された周波数は、真の周波数と小数点以下 3 桁まで一致しているため省略する。最後の列「エラー最大特異値」は、真の出力と同一の入力信号を用いた識別されたシステム行列から再構築された出力との誤差行列の最大特異値を示す。この誤差行列のサイズは $rn \times Z$ で、ここで m は出力の数、 Z はデータの長さを表す。

モデルの次数を削減する別の方法は、最初にフルサイズのシステム行列を取得し（すなわち特異値分解なし）、その後モードトランケーションを行う方法である。識別された各モードは、そのモードの特異値で重み付けされ、最大のモード特異値を持つ 3 つのモードがシステムを表すために選択される。表 ?? は、異なる p 値に対して識別された減衰比を示す。識別された周波数は真の周波数と小数点以下 3 桁まで一致している（表 ?? 参照）。

表 3: 部分分解法とモード削減によって得られた減衰比 B_{tio}

表 2 は、 T_{hh} の特異値を削減した際、整数 p がこの例での最小必要条件である 3 に対して十分に大きく選ばれると、すべての減衰比が過小評価されることを示している。出力誤差が最小となる整数 p が存在する。 p を最適値以上に増加させても、必ずしも出力誤差が減少するわけではない。間接法と出力誤差最小化法の両方が、出力誤差において類似の傾向を示している。出力誤差最小化法は、間接法と比較して計算に要する時間が 3 桁も長くなるが、より良い出力誤差解を提供する。

一方、表 3 は、大きな p の場合でも、モード削減により減衰比が過大評価されることがあることを示している。また、出力誤差が最小となる整数 p が存在する。表 2 と表 3 を比較すると、モード削減法が $p = 50$ の場合に、出力誤差最小化法および間接法の両方よりもやや良好な性能を示していることがわかる。モード削減法の計算時間は間接法とほぼ同じであり、出力誤差最小化法よりもはるかに短いことも注目すべき点である。出力誤差最小化法は、 $x(0)$ および行列 B 、 D （行列 A 、 C が与えられた場合）に対する出力誤差を最小化するだけであり、すべての行列 A 、 B 、 C 、 D 、および $x(0)$ に対する出力誤差をグローバルに最小化するものではない。

3.4.2 実験例

このセクションで示す実験結果は、提案された技術が実際に使用された場合の有用性を示している。図 2 は、使用された L 字型トラス構造を示している。この構造は、縦方向に 9 ベイ、横方向に 1 ベイあり、それぞれ 90 インチおよび 20 インチの長さである。短い部分は、壁にしっかりと取り付けられた鋼板にクランプされている。正方形の断面は 10 インチ \times 10 インチである。ビームの先端に位置する 2 つの冷風ジェットスラスタが、励振と制御のためのアクチュエータとして機能している。各スラスタは最大 2.2 ポンドの推力を持つ。正方形断面の一角に位置する 2 つのサーボ加速度計が、平面内の先端加速度の測定を提供している。さらに、動的結合を強化し、構造の基本周波数を低下させるために、30 ポンドのオフセットウェイトが追加されている。識別のために、トラスは両方のスラスタにランダムな入力を加えて励振される。入力信号と出力信号は 250Hz でサンプリングされ、システム識別のために記録される。2000 点のデータ記録が識別に使用される。

スラスト

重心から外れた質量による影響を考慮する。

加速度計

図 2 は、トラス構造のテスト構成を示す。

表 4 には、行列 A と C を決定するための部分分解法と出力誤差最小化法を用いて特定されたモード周波数および減衰比が示されている。初期インデックス p は、システムの最大次数を $prn = 10, 20, 30, 40, 50, 100, 200$ に設定するために任意に設定される。特異値の切り捨てがシステムモデルの次数を 6 に減少させるために使用される。出力誤差は、 p が 5 から 100 に増加するにつれて連続的に減少する。出力誤差の減少速度は、 $p = 50$ から $p = 100$ の間で遅くなる。全ての異なる p 値に対して特定された周波数は近似しており、減衰比は、第 1 モードでは 2.5% から 0.4%、第 2 モードでは 1.5% から 0.4%、第 3 モードでは 1.2% から 0.07% の範囲である。

行列 B および D を計算するための間接法と直接法の結果は示されていない。これらの方法は、表 4 に示された出力誤差よりも数桁大きい出力誤差を生じさせるからである。両方の方法は、入力および出力のノイズがホワイトでランダムなガウス分布を持ち、平均がゼロであると仮定したシミュレーションデータに対しては良好に機能する。したがって、ノイズの非線形性が行列 B および D に重大な誤差を導入する主な原因であると考えられる。この例は、行列 A および C が特異値の切り捨てによる縮減モデルによって得られる場合、行列 B および D を計算するために間接法および直接法を実際には使用すべきではないことを示している。

表 1 特異値の切り捨てを伴う部分分解法によって得られたモードパラメータの特定結果

モード	OEM P	モード 1		モード 2		モード 3	
		周波数 (Hz)	減衰率 (%)	周波数 (Hz)	減衰率 (%)	周波数 (Hz)	減衰率 (%)
5	10	5.89	2.48	7.29	1.54	48.5	1.19
15		5.88	2.08	7.30	0.99	48.2	0.71
20		5.88	1.10	7.30	0.51	48.0	0.99
25		5.88	0.62	7.29	0.38	47.5	2.06
50		5.87	0.46	7.29	0.42	47.4	2.44
100		5.86	0.44	7.29	0.41	48.4	0.46
200		5.85	0.43	7.28	0.43	48.6	0.07

表 2 モード間引き手法を用いたモードパラメータの同定結果

モード	モード 1		モード 2		モード 3		誤差,	p
	周波数, Hz	減衰率, %	周波数, Hz	減衰率, %	周波数, Hz	減衰率, %	最大 SV	
5	5.89	3.50	7.28	2.30	49.0	1.13	817.99	5
10	5.87	0.65	7.29	0.47	48.6	0.74	262.81	10
15	5.85	0.40	7.28	0.41	48.6	0.46	197.32	15
20	5.85	0.37	7.28	0.41	48.7	0.44	216.79	20
25	5.85	0.38	7.28	0.42	48.7	0.64	203.56	25
50	5.85	0.38	7.28	0.44	48.6	0.47	198.69	50
100	5.85	0.40	7.28	0.45	48.5	0.30	194.58	100

表 5 には、部分分解法によって同定されたモード周波数と減衰比が示されています。この方法では、特異値分解による行列の間引きを行わず、行列 A および C を決定し、行列 B および D は間接的な方法で算出されました。フルサイズのモデルは、その後、6 次元（対象となるモードのみを含む）に縮小され、出力誤差の計算に使用されました。出力誤差は、 p が 5 から 10 に増加すると急速に減少し、 $p = 15$ で最小に達します。その後、出力誤差はわずかに増加し、再び減少して $p = 100$ で別の最小値に達します。しかし、 $p = 100$ での最小出力誤差値 194.58 は、 $p = 15$ での最小出力誤差値 197.32 とほとんど差がありません。表 4 に示された結果と同様に、異なる p 値に対して同定された周波数は非常に近く、減衰比はモード 1 で 3.5 % から 0.4 %、モード 2 で 2.3 % から 0.45 %、モード 3 で 1.13 % から 0.3 % の範囲であることがわかります。

表 4 と表 5 のモードパラメータの類似性は、両者が特異値分解前の同じ可観測性行列を共有しているため、驚くべ

きことではありません。表 4 は特異値の間引き後に計算されたモードパラメータを示しており、つまり小さな特異値に対応する可観測性行列の列が間引かれています。一方、表 5 はフルサイズの可観測性行列から計算されたモードパラメータを示しています。表 5 に示されたモードパラメータは、システムを表すものとして選ばれました。

ここで、出力誤差最小化手法を用いて行列 B および D を再計算した場合、表 5 の出力誤差がさらに低減されるかどうかという疑問が生じるかもしれません。表 5 の最後の列がこの疑問に答えています。実際、すべてのケースで出力誤差が多少改善されています。表 5 における $p = 15$ の出力誤差値 171.33 は、表 4 での $p = 50$ における出力誤差値 174.25 よりも良好です。この結果は、特異値の間引きと出力誤差最小化の組み合わせが、任意の p 値での全体最小値を生成しない可能性があることを示しています。その結果、モード間引きと出力誤差最小化手法の組み合わせがモデル縮小には効果的であることがわかります。

3.5. 第 3 章のまとめ

新しいシステム実現アルゴリズムが開発され、データ相関行列を用いて特異値分解による可観測性行列を計算しました。データ相関行列は、シフトされた出力データの自己相関行列から、シフトされた入力および出力データ間の相互相関を、シフトされた入力データの自己相関行列の逆で重み付けして減算することにより形成されます。その後、この可観測性行列を使用して状態行列 A と出力行列 C が計算されます。

二つの計算手法が提示されており、それには完全分解法と部分分解法が含まれる。行列 A と C を決定するためには、部分分解法の方が完全分解法よりも扱いやすいと考えられる。これは、部分分解法が特異値の切り捨てを必要としないためである。実際には、データ列がどれほどクリーンであっても、ゼロの特異値は存在しない。どの特異値を切り捨てるかを決定するには、エンジニアリングの判断や感度解析などの特殊な技術が必要である。

計算された行列 A および C に基づき、入力行列 B 、直接伝達行列 D 、および初期状態ベクトル $x(0)$ を計算するための三つの手法が説明されている。間接法は、可観測性行列に直交する列ベクトルを持つ行列とデータ相関方程式を使用して、行列 B および D を抽出する。直接法は、可観測性行列と出力方程式を使用して、行列 B および D を解くための方程式を構成する。出力誤差最小化法は、テスト出力と計算された（すなわち再構築された）出力との誤差を最小化することにより、行列 B および D 、ならびに初期状態ベクトル $x(0)$ を決定する。入力と出力のノイズがホワイト、ガウス、およびゼロ平均の場合、この要約で述べられているいずれの手法の組み合わせでも、行列 A 、 B 、 C 、 D および初期状態ベクトル $x(0)$ の計算に優れた性能を示す。他のノイズの場合でも、特異値の切り捨てが行われない限り、いずれの組み合わせも良好に動作する。モデル縮減のために特異値の切り捨てが行われる場合、部分分解アルゴリズムと出力誤差最小化の組み合わせは、他の手法よりも優れており、モード縮減技術に匹敵する。

4. SRIM と他の手法の統一化

多くのシステム実現アルゴリズムは、状態空間の離散時間線形モデルから始まり、データ相関に基づいてシステム行列を計算するための基本的な方程式を形成する。一方、他のアルゴリズムは、有限差分モデルとデータ相関を用いてシステム行列を解く。これら二つのアプローチは異なるタイプのモデルを使用するため、根本的に異なるように見える。異なるモデルは、最小化される方程式誤差の違いにより、識別結果に顕著な差異をもたらす可能性がある。その結果、識別結果を解釈し、問題に最も適した技術を選択することが非常に難しい。このため、異なる技術を包括的かつ一貫性のある形で統一する必要がある。

本論文の第 4 章では、セクション 3 で提示された SRIM と他の実現手法との関係が確立される。SRIM は、状態空間の離散時間線形方程式を用いて導出され、システム実現のための特別なデータ相関を形成する。他の実現アルゴリズム、例えばオブザーバ/カルマンフィルタ識別 (OKID) 技術 (文献 11 および 12) では、入力・出力データ相関の情報を必要とする有限差分方程式の係数行列を計算することから始まる。データ相関の使用における類似性から、SRIM と OKID 法の関係が確立される。本章で提示されるアプローチは、他の技術、例えば OKID や他の研究者が開発したサブスペース識別アプローチ (文献 6 および 7) を理解し、解釈するためのより良い方法を提供する。

4.1. 時間領域 ARX モデル

有限差分モデルは、制御分野では一般に自己回帰外因性 (ARX) モデルと呼ばれる。ARX の係数行列は、入力と出力データから出力方程式の誤差（すなわち、実際の出力と推定された出力の差）を最小化することで計算できる (文献 11)。

離散時間 ARX モデルは、通常次のように表記される。

$$\alpha_{p-1}y(k+p-1) + \alpha_{p-2}y(k+p-2) + \cdots + \alpha_0y(k) = \beta_{p-1}u(k+p-1) + \beta_{p-2}u(k+p-2) + \cdots + \beta_0u(k) \quad (20)$$

ここで、 c_i は $i = 1, 2, \dots, p-1$ に対して $m \times rm$ 行列であり、 β_i は $i = 1, 2, \dots, p-1$ に対して $m \times r$ 行列である。 $y(k)$ は時刻 k における $m \times 1$ の出力ベクトルであり、 $u(k)$ は時刻 k における $r \times 1$ の入力ベクトルである。この方程式は、出力列 $y(k)$ を p 時間ステップまでの入力列 $u(k)$ に関連付ける。

式 (66) は次の行列等式を導出する:

$$\mathcal{O}_Y^p(k) = \mathcal{B}_U^p(k) \quad (67)$$

ここで、

$$\begin{aligned} \mathcal{O} &= [\mathcal{O}_0 \quad \mathcal{O}_1 \quad \dots \quad \mathcal{O}_{p-1}], \quad \mathcal{Z} = [\mathcal{Z}_0 \quad \mathcal{Z}_1 \quad \dots \quad \mathcal{Z}_{p-1}], \\ \mathcal{Y}^p(k) &= [y(k) \quad y(k+1) \quad \dots \quad y(k+N-1)], \quad \mathcal{Y}_p(k) = \begin{bmatrix} y(k+1) & y(k+2) & \dots & y(k+N) \\ \vdots & \vdots & \ddots & \vdots \\ y(k+p-1) & y(k+p) & \dots & y(k+p+N-2) \end{bmatrix}, \\ \mathcal{U}^p(k) &= [u(k) \quad u(k+1) \quad \dots \quad u(k+N-1)], \quad \mathcal{U}_p(k) = \begin{bmatrix} u(k+1) & u(k+2) & \dots & u(k+N) \\ \vdots & \vdots & \ddots & \vdots \\ u(k+p-1) & u(k+p) & \dots & u(k+p+N-2) \end{bmatrix} \end{aligned} \quad (68)$$

式 (67) に $u^T(k)$ を後乗し、式 (14) における R_{yu} と R_{uu} の定義を考慮すると、

ここで、 R_{uu}^{-1} の存在が仮定される。同様に、式 (67) に $v^T(k)$ を後乗し、式 (14) における R_{vy} と R_{yu} の定義を考慮すると、

$$\mathcal{O}^T R_{yy} = \beta^T R_{yu}^T \quad (70)$$

式 (69) の β を式 (70) に代入すると次のようになる:

$$R_{yy}^{-1} [R_{yu}^T R_{uu}^{-1} R_{yu}] \mathcal{O}^T R_{yu} = 0$$

または

$$\mathcal{O} \mathcal{T}_{hh} = 0 \quad (71)$$

ここで、

$$\mathcal{T}_{hh} = R_{yy} - R_{yu} R_{uu}^{-1} R_{yu}^T$$

これは式 (20) と同一である。式 (71) は、 $m \times rm$ のパラメータ行列 \mathcal{O} が $rm \times pm$ 行列 \mathcal{T}_{hh} の零空間にあることを意味する。言い換えれば、式 (26) または式 (31) の \mathcal{O} から任意の m 列を使用して、次のように行列 \mathcal{O} を構築できる:

$$\mathcal{O} = \text{任意の } m \text{ 列を } \mathcal{O}^T \text{ または } \mathcal{O}^{*T} \text{ から選ぶ} \quad (72)$$

結果として、 $m \times pr$ 行列 \mathbf{F}_l は以下のように解くことができる。

$$\mathbf{f}_l = \text{任意の } r_n \text{ 行を } \mathbf{U}_o^T \mathbf{P}_{yy} \mathbf{U}_{uu}^{-1} \text{ から選択} \quad (21)$$

実際には、入力および出力のノイズなどのシステム不確実性のために、行列 \mathbf{U}_o を形成するための零特異値が存在しない可能性がある。この場合、 m の列に対応する最小の特異値を選択して \mathbf{U}_o を形成する。もう一つの方法として、式 (??) と (??) を組み合わせて $\mathbf{c}_t /$ を計算すると、次のようになる。

$$\mathbf{O}_m = \mathbf{0}_{m \times p(m+r)} \quad (22)$$

または

$$\mathbf{O}_m = \mathbf{0}_{m \times p(m+r)} \quad (23)$$

ここで $\mathbf{0}_{m \times p(m+r)}$ は $m \times p(m+r)$ の零行列であり、

$$\mathbf{O} = [-\mathbf{a} \quad \mathbf{b}], \quad \mathbf{T}_m = \begin{bmatrix} \mathbf{T}_{yy} & \mathbf{T}_{yu} \\ \mathbf{T}_{yu} & \mathbf{T}_{uu} \end{bmatrix} \quad (24)$$

行列 \mathbf{O} のサイズは $m \times p(m+r)$ であり、 \mathbf{n} のサイズは $p(m+r) \times p(m+r)$ である。式 (??) は、未知の行列 \mathbf{O} が行列 \mathbf{T} のヌル列空間に存在することを示唆している。 \mathbf{T} のヌル列空間の基底ベクトルから生成された任意の m 列が、 \mathbf{O}^T の解と見なされる。理論的には、 p の値を十分に大きく選ぶ必要があり、 $p(m+r) \times p(m+r)$ 行列 \mathbf{T} がランク落ちするようにする。入力相関行列 \mathbf{U}_{uu} は pr のフルランクを持つ必要があるため、 \mathbf{T} のランク落ちから、 \mathbf{T}_{yy} のランクは pr 未満でなければならない。具体的には、 \mathbf{T}_{yy} のランクは少なくとも m 個の独立した列を持ち、 \mathbf{T} のヌル列空間の次元が rn となるように、 $pm - rn$ 以下でなければならない。

6.1 オブザーバ／カルマンフィルタ識別 (OKID) アルゴリズム

式 (??) の一つの解は、 $m \times m$ 行列 \mathbf{a}^{-1} を式 (??) に前置することで次のように得られる。

$$\mathbf{O}_{\text{len}} = \mathbf{0}_{m \times p(m+r)} \quad (25)$$

または、式 (??) と (24) から

$$\mathbf{O}_{\text{len}} = \mathbf{0}_{m \times p(m+r)} \quad (26)$$

次に、 \mathbf{T} の m 行を $(p-1)m+1$ から pm まで取り出し、次のように式 (26) の右側に移動させる。

$$\mathbf{L}_n((pmT)_{1+(p-1)(p+r)m,:}) = \mathbf{T}((p-1)m+l'pm,:) \quad (27)$$

ここで

$$\mathbf{O} = [-\mathbf{o}_{p11} \quad \mathbf{O}_{11} \quad \cdots \quad \mathbf{O}_{p-11} \quad \mathbf{O}_{p20} \quad \mathbf{O}_{p11}\mathbf{n} \quad \mathbf{O}_{p11}\mathbf{n} \quad \cdots \quad \mathbf{O}_{p11}\mathbf{n}] \quad (28)$$

次に示すように、の次元は $m \times ((2-1)m+pr)$ であり、これは単位行列 I_{rn} を除去したために e より m 列短くなっています。式 (78) は、次のように @ に対する最小二乗解を持つべきです。

$$= T_{(p-1)m+1:pm,:} [T_g(pm+n(l:(p-1)m,:), \\ 1:(p-t-r)m,:)]_{(80)}$$

ここで、 t は擬似逆行列を意味します。擬似逆行列を避けるためには、 $T_{(p-1)rn+1:n(l:(-1)m,:)}$ の $(p-1)m+1$ から pm までの rn 列を削除し、対応する m 列を $LT_{pm+1:(p+r)m,:}$ から削除します。

式 (80) から得られるパラメータ行列は、OKID 手法（文献 1 および 7）で使用されるシステム行列 A, B, C, D およびオブザーバゲイン G を特定するために使用されるパラメータ行列と同一です。相関行列 T は情報行列と呼ばれ、入力データと出力データ間の相関情報を含んでいます。

理論上、同じ相関行列で開始するすべての方法は、同じ同定結果を生成するべきです。実際には、システムの不確かさが存在するため、同定結果は多少異なる場合があります。たとえば、式 (80) によって解かれる行列 I は、測定された出力と ARX で計算された出力の残差を最小化します（文献 1 および 7）。一方、他の方法では、測定された出力と同定された状態空間モデルから計算された出力との誤差を最小化します。最小化のために異なる誤差基準が使用されるため、同定結果は多少異なると予想されます。それにもかかわらず、同定されたモデルが特異値の切り捨てやモードの切り捨てによって大幅に削減されない限り、結果に大きな違いはないはずです。

6.2 実験例

このセクションでは、図 2 のトラス構造から取得されたデータを用いて説明を行います。OKID 手法を適用してシステム行列 A, B, C, D を決定します。計算ステップは 2 つ必要です。最初のステップは、式 (80) を使用して ARX 係数行列を計算し、システムのマルコフパラメータ（パルス応答）を決定します。次に、計算されたパルス応答から固有系実現アルゴリズム（ERA）を使用して、同時に行列 A, B, C, D を実現します。この例では、ERA で特異値の切り捨ては行っていません。ERA で同定されたフルサイズのモデルを、興味のあるモードのみを含む 6 次の次元に削減します。削減されたモデルを用いて出力誤差を計算します。

表 6 には、OKID 手法で同定されたモード周波数と減衰比が示されています。出力誤差は、 p が 5 から 10 に増加する際に急速に減少し、 $p = 15$ で最小に達します。その後、出力誤差は再びわずかに増加し、 $p = 50$ で再び最小になります。表 5 と 6 のモーダルパラメータは同一です。両方の表の出力誤差は非常に近いですが、 $p = 100$ では、表 6 の出力誤差値 882.26 は、表 5 の出力誤差値 194.58 の 4.5 倍です。両方の表でモーダルパラメータが同一であるため、この差異の原因は行列 C, B, D にある可能性があります。実際に、行列 B と D を出力誤差最小化法で再計算すると、出力誤差は両方の表で同じレベルに戻ります。

表 3 OKID により同定されたモードパラメータとモード切断

モード	1	モード	2	モード	3	OKID OEM	周波数、減衰	誤差、誤差、P
ttz	周波数 (Hz)	減衰 (%)	周波数 (Hz)	減衰 (%)	周波数 (Hz)	減衰 (%)	最大 SV	最大 SV
5	5.89	10	5.87	15	5.85	20	5.85	
25	5.85	50	5.85	100	5.85	3.50	0.65	
0.40		0.37		0.38		0.38		
0.40		7.28	7.29	7.28	7.28	7.28	7.28	
2.30		0.47		0.41		0.41		
0.42		0.44		0.45		49.0	48.6	
48.6		48.7		48.7		48.6	48.5	
1.13		0.74		0.46		0.44		
0.64		0.47		0.30		818.05	262.81	
197.22	215.97	203.21	196.70	882.26	735.99	202.84	171.33	
174.46	174.06	175.10	174.02					

4.2. 周波数領域の ARX モデル

周波数領域の ARX モデルは、時間領域の ARX モデルに z 変換を適用して生成されます。周波数領域の ARX モデルの係数行列は、周波数応答データを用いて、対象の周波数点における実際の伝達関数と推定された伝達関数との誤差を最小化することによって求められます。理論上、周波数領域アプローチで得られた ARX 係数行列は、時間領域アプローチで得られる行列と同一であるべきです。

$G(z_k)$ を式 (1) で記述されたシステムの伝達関数行列とします。左行列分数（参考文献 1）を考えます。

$$O(z_k) = a^{-1}(z_k)\phi(z_k) \quad (81)$$

ここで、

$$\phi(z_k) = \phi_0 + \phi_1 z_k^{-1} + \cdots + \phi_{p-1} z_k^{-(p-1)} \quad (82)$$

および

$$\Phi(z_k) = \Phi_0 + \Phi_1 z_k^{-1} + \cdots + \Phi_{p-1} z_k^{-(p-1)}$$

は行列多項式です。各 ϕ_i は $m \times m$ の実数正方行列で、各 Φ_i は $m \times r$ の実数長方形行列です。 $G(z_k)$ が実験から得られた周波数応答関数 (FRF) を表す場合、変数 $z_k = e^{j(2\pi k/L\Delta t)}$ ($k = 0, 1, \dots, L-1$) は $2\pi k/(L\Delta t)$ での周波数点に対応し、 Δt はサンプリング時間間隔、 L はデータの長さです。式 (81) の因数分解は一意ではありません。簡便さと単純さのために、両多項式の次数は $p-1$ とすることが出来ます。

式 (81) に $\Phi(z_k)$ を前掛けすると、

$$\Phi(z_k)a(z_k) = \Psi(z_k) \quad (83)$$

となり、これを

$$\phi_0 G(z_k) + \phi_1 G(z_k) z_k^{-1} + \cdots + \phi_{p-1} G(z_k) z_k^{-(p-1)} = \Phi_0 + \Phi_1 z_k^{-1} + \cdots + \Phi_{p-1} z_k^{-(p-1)} \quad (84)$$

と整理できます。式 (84) は式 (66) の z 変換です。 $G(z_k)$ と z_k^{-1} が既知であるため、式 (84) は線形方程式です。 $G(z_k)$ が $z_k = e^{j(2\pi k/L\Delta t)}$ ($k = 0, \dots, L-1$) で既知であるため、 L 個の方程式が得られます。これら L 個の方程式をまとめると、

$$e_\Psi = \Phi_{r-1} e_\Phi \quad (85)$$

となります。

以下の行列を考える：

$$\begin{array}{cccc} G(z_0) & G(z_1) & \cdots & G(z_{L-1}) \\ G(z_0)z_0^l & G(z_1)z_1^{l-1} & \cdots & G(z_{L-1})z_{L-1}^{l-1} \\ \vdots & \vdots & \ddots & \vdots \\ z_0^{l-1}I & z_1^{l-1}I & \cdots & z_{L-1}^{l-1}I \\ z_0I & z_1I & \cdots & z_{L-1}I \\ z_0^{l-1}h & z_1^{l-1}h & \cdots & z_{L-1}^{l-1}h \end{array}$$

$$\Phi = \begin{bmatrix} -\mathbf{O}_0 & -\mathbf{O}_1 & \cdots & -\mathbf{O}_{p-1} \\ \mathbf{I}_0 & \mathbf{I}_1 & \cdots & \mathbf{I}_{p-1} \end{bmatrix} \quad (86)$$

ここで、 Φ は $(m+r)p \times (rL)$ の行列であり、 Ψ は $m \times (m+r)p$ の行列である。式 (85) は線形代数方程式であり、パラメータ行列 \mathbf{O} が Φ のヌル列空間に属することを示唆している。

Φ のヌル列空間は $\Phi\Phi^*$ のヌル列空間と同一である。ここで $*$ は複素共役転置を表す。式 (85) にポストマルチプライすると、

$$\mathbf{O} = \mathbf{O}_1 \Psi_{(m+r)} \quad (87)$$

式 (87) において、 $m \times p(m+r)$ 次元の零行列 $\mathbf{O}_{m \times p(m+r)}$ は、 $m \times t$ 次元の零行列 $\mathbf{O}_{m \times t}$ よりも一般にかなり小さい。 $\Phi\Phi^*$ を計算する際、行列 Φ を完全に形成せずに計算することは、その特別な構成により容易である。また、式 (87) から Ψ を求める方が、式 (85) から求めるよりもはるかに簡単である。

式 (87) は、式 (75) と同形式であるが、式 (87) の Φ は複素行列であり、式 (75) の \mathbf{T}_Φ は実行列である。両方の式 (75) および (87) は、ARX 係数行列 \mathbf{C} および β を求めるために導出される。

\mathbf{C} および β はともに実行列であるため、 Φ と \mathbf{T}_Φ は共通の性質を持つべきである。 \mathbf{T}_Φ はエルミート行列であるため、 $\mathbf{T}_\Phi = \mathbf{T}_\Phi^*$ 、 \mathbf{T}_Φ の実部は対称行列であり、虚部は反対称行列である。

$$\mathbf{T}_\Phi = \text{実部}[\mathbf{T}_\Phi] = \text{実部}[\Phi\Phi^*] \quad (89)$$

行列 \mathbf{T}_Φ (式 (89)) は、ARX 係数行列を計算するために構築されているが、状態行列 \mathbf{A} および出力行列 \mathbf{C} を計算するためにも使用できる。式 (76) で示されたような分割と同様に、 \mathbf{T}_Φ を次のように 4 つの部分に分割する：

$$\mathbf{T}_\Phi = \begin{bmatrix} \mathbf{T}_{\Phi_{11}} & \mathbf{T}_{\Phi_{12}} \\ \mathbf{T}_{\Phi_{21}} & \mathbf{T}_{\Phi_{22}} \end{bmatrix} \quad (90)$$

ここで、 $\mathbf{T}_{\Phi_{11}}$ は $pm \times pm$ の正方行列であり、 $\mathbf{T}_{\Phi_{12}}$ は $pm \times pr$ の長方形、 $\mathbf{T}_{\Phi_{22}}$ は $pr \times pr$ の正方行列である。次のような概念的な等式を考慮する：

$$\mathbf{T}_{\Phi_{11}} = \mathbf{T}_{\Phi_{yy}}, \quad \mathbf{T}_{\Phi_{12}} = \mathbf{T}_{\Phi_{yx}}, \quad \mathbf{T}_{\Phi_{22}} = \mathbf{T}_{\Phi_{xx}} \quad (91)$$

7 行列の定義と計算方法

方程式 (20) で定義された $prn \times pm$ 行列 T_{Chh} は以下のように記述されます。

$$T_{hh} = T_{ii} - T_{Q2}T_{217}T_{lT2} \quad (92)$$

行列 T_{hh} は、完全分解法または部分分解法のいずれかを使用して、行列 A および C を決定するために使用されます。これに対し、行列積 $T_{12}T_{2-21}$ は、直接法または間接法を用いて、入力行列 B と伝達行列 D を決定するために使用できます。

7.1 伝達関数誤差最小化法

行列 B および D を計算するための直接法および間接法は、小さな特異値の切り捨てが行われる場合、実際の伝達関数と推定された伝達関数との間の誤差を最小化するとは限りません。時間領域における出力誤差最小化法に類似して、伝達関数誤差最小化法は、伝達関数を行列 B および D に明示的に関連付ける方程式を形成します。

$m \times r$ の伝達関数 $G(z_k)$ (方程式 (81)) は、システム行列によって次のように表現されます：

$$G(z_k) = D + C(z_k I_n - A)^{-1} B \quad (93)$$

ここで、 $z_k (k: 0, 1, \dots, \ell - 1)$ はすべての z_k についてです。方程式 (93) は次を導きます：

$$G(z_0) = I_m + C(z_1 I_n - A)^{-1} B \quad (94)$$

$$I_m + C(z_0 I_n - A)^{-1} \\ + C(z_{e-1} I_n - A)^{-1}$$

ここで、 I_m および I_n はそれぞれ次数 m および n の単位行列です。行列 A 、 C および $G(z_k)$ が与えられた場合、行列 B および D は次のようにして決定できます：

$$I_m + C(z_0 I_n - A)^{-1} G(z_0) = I_m + C(z_1 I_n - A)^{-1} \quad (95)$$

ノイズのあるシステムでは、行列 B および D の解は方程式 (94) を満たすのではなく、方程式 (94) の左右の項の間の誤差を最小二乗法で最小化します。実際には、行列 B および D を決定するためには、方程式 (94) の実部のみが必要です。

7.2 実験例

この例では、大型トラス型構造から得られた実験データを使用します。図 3 は、制御構造相互作用問題を研究するために使用される Langley 試験台 (参考文献 13) を示しています。システムには、制御用の 8 組のコロケートされた入力および出力があります。入力は空気...

- 2 台
- 1 台
- 8 台の比例制御および双方向スラスタ
- 8 台のサーボモータ (DC)
- 加速度計
- 876

スラスタと出力は加速度計です。図??は、入力と出力のペアの位置を示しています。この構造物は、位置 1、2、6、7 にある 4 つのスラスタにランダムな入力信号を与えて励振されました。入力信号と出力信号は、低周波域にエネルギーを集中させるため、ナイキスト周波数 (12.8 Hz) の 78% に設定されたデジタルローパスフィルタでフィルタリングされました。各センサから得られた 2048 のデータポイントを、サンプリング周波数 25.6 Hz で同定に使用しました。この例では、位置 1 および 2 にある入力と出力のペアから得られる 4 つの周波数応答関数 (FRF) を用いて、シ

システムの状態空間モデルが同定されました。

整数 $p = 50$ は、最大 50 のモード（システム次元 100 の場合）を同定するのに十分です。周波数領域版でシステムの次元を特異値分解によって 80 に切り詰め、不安定モードを 1 つ除去してさらに 78 に減少させた状態空間モデルが得られました。図??は、実験データ（実線）と再構成された周波数応答データ（破線）を比較したものです。

図??は、入力 1 に対する出力 1 の周波数応答を示しており、強い信号を表しています。同定されたシステム行列 A 、 B 、 C 、および D を用いて再構成された FRF が得られます。ここで行列 C および D は、間接法によって算出されました。1% 未満の減衰を持つモードが 33 個（66 の状態に対応）あります。実験データと再構成 FRF 間の伝達関数誤差の最大特異値は 128.89 です。図??を注意深く観察すると、振幅および位相プロットの右端近くに顕著な不一致があることが分かります。この不一致を修正する一つの方法は、伝達関数誤差最小化法を用いて行列 B および D を再計算することです。

- 103
- 102
- 10

大きさ (Magnitude)

- 10^0
- 10^{-1}
- 10^{-2}
- 200
- 150
- 100
- 50

位相 (Phase)

- 角度, 0°
- -50°
- -100°
- -150°
- -200°

周波数応答 (Frequency Response)

テスト信号 Rons ct L

- 2 4 6 8 10 12 周波数, Hz
- ...X-
- テスト
- -----再構成済み
- ! . . . l . . . ϕ . i . I . . . I . $^\circ$ | s

0 2 4 6 8 10 12 周波数, Hz

図 5 は、新たに計算された行列 B および D に基づく、入力 1 に対する出力 1 の周波数応答を示しています。伝達関数誤差は 128.89 から 65.184 に低減しました。明らかにトレードオフが存在し、テストと再構成された FRF の差異は、約 12 Hz から約 8 Hz に移動しています（図 5）。他の入出力ペアについても同様の結果が得られましたが、この論文では示していません。入力 1 に対する出力 2 の周波数応答は弱い信号を表しています。信号が弱いのは、センサー 2 が入力 1 に対して直交しているためです。結果は、強い信号に対しての一致がより良好であることを示しています。

結論

34

103

102

101

振幅

100

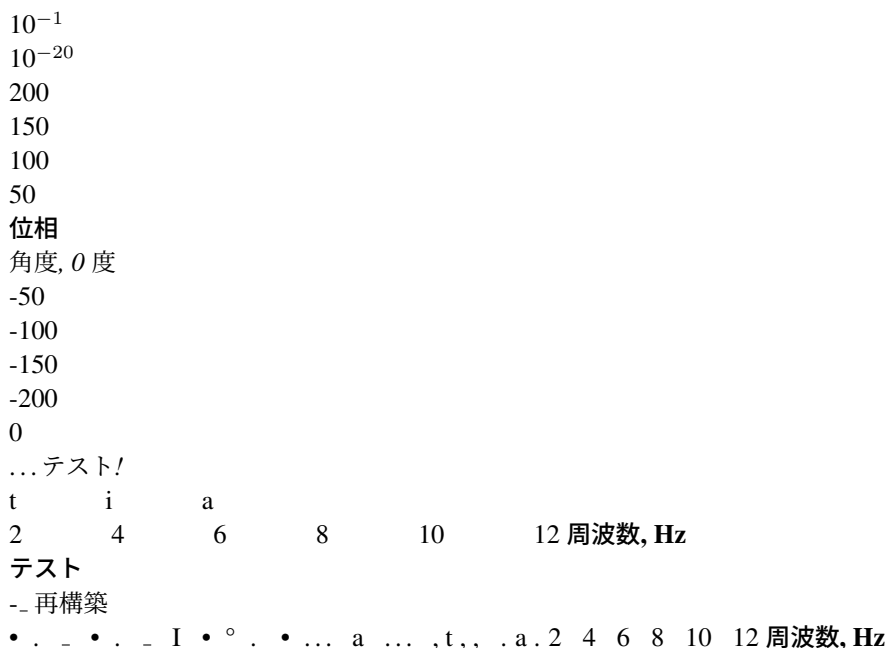


図 5. テストと再構築された入力-i/出力-1 FRF の比較

テストデータおよび行列 B と D から再構築された入力-i/出力-1 FRF データ。

4.3. 第 4 章まとめ

データ相関の概念は、本論文で開発されたシステム実現方法と、時間領域および周波数領域における離散時間有限差分モデルに基づく他の方法との関連を導き出します。この新しい状態空間アプローチは、情報行列を使用することで、これらの方法をより良く理解し解釈する手段を提供します。実際、情報行列はシステム行列を計算するための共通の基盤です。本論文で開発された方法では、情報行列の列空間の基底ベクトルを使用してシステム行列を計算します。一方、既存の他の方法では、情報行列の零列空間の基底ベクトルを使用してシステム行列を計算します。理論的には、両アプローチは同一の結果を生み出すはずですが、システムの不確実性や計測ノイズにより、実際の結果は大きく異なる可能性があります。本論文で提示された数学的統一は、異なる方法から得られた識別結果をユーザーが解釈するのに役立つでしょう。

8 結論

本論文には、3つの主要な技術的貢献が含まれています。第一に、シフトされた入力データおよび出力データの相関行列から構成される一般化情報行列が導入されました。第二に、情報行列を基盤とした新しいシステム実現アルゴリズムが導出されました。第三に、いくつかのシステム実現アルゴリズムが、システム情報行矩陣を通じて統一されました。パルス応答や自由減衰応答の場合、情報行列はシフトされた出力データの相関行列に帰着します。したがって、パルス応答に基づく古典的なシステム実現アルゴリズムは、本論文で紹介されたアルゴリズムの特別なケースと見なすことができ、システム実現アルゴリズムの統一が古典的な実現手法を含むことを意味します。

NASA ラングレー研究センター
ハンプトン、VA 23681-0001
1996 年 12 月 2 日

付録 A

出力誤差最小化法の効率的計算

任意の整数 N に対して、次式を計算することが可能です。

$$\sum_{k=0}^{N-2} \text{CAN}_{k-2} \ln(k)$$

この計算は、次のように初期条件がゼロの仮想的な線形システムを用いることで、次元が $m \times nr$ の行列として実現できます。

$$\begin{aligned} x_{ij}(N+1) &= Ax_{ij}(N) + c_i u_j(N), & x_{ij}(0) &= 0_{n \times 1}, \\ z_{ij}(N) &= Cx_{ij}(N), & i &= 1, 2, \dots, n, \quad j = 1, 2, \dots, r \end{aligned} \quad (\text{A1}) \quad (33)$$

ここで、 e_i は n 次元単位行列の第 i 列を表し、 $u_j(k)$ は式 (60) で定義されたベクトル $u(k)$ の第 j 成分です。任意の時刻 N において、次式は次元 $m \times r$ の行列 $Z(N)$ を生成します。

$$Z(N) = \begin{bmatrix} z_{11}(N) & z_{21}(N) & \dots & z_{n1}(N) \\ \vdots & \vdots & \ddots & \vdots \\ z_{1r}(N) & z_{2r}(N) & \dots & z_{nr}(N) \end{bmatrix} \quad (\text{A2})$$

これにより、次のようになります。

$$\sum_{k=0}^{N-2} \text{CAN}_{k-2} \ln(k) = Z(N-1), \quad N > 1 \quad (\text{A3})$$

付録 B

情報行列の効率的な計算方法

$\mathbf{Y}_p(k)$ および $\mathbf{U}_p(k)$ の生成に使用されるデータシフトの性質から、相関行列 \mathbf{T}_{uu} 、 \mathbf{T}_{yu} 、および \mathbf{T}_u を効率的に計算する方法が存在します。一般性を失わずに簡単にするために、 \mathbf{T}_{yu} の計算を考えます。 $\mathbf{Y}_p(k)\mathbf{U}^T(k)$ の積は次のように書くことができます。

$$\begin{aligned} & \sum_{r=k}^{k+N} \mathbf{Y}(r)\mathbf{u}^T(r), \quad \sum_{r=k}^{k+N} \mathbf{Y}(r)\mathbf{u}^T(r+1), \quad \sum_{r=k}^{k+N} \mathbf{Y}(r)\mathbf{u}^T(r+2), \dots \\ &= \sum_{r=k+1}^{k+N+1} \mathbf{Y}(r)\mathbf{u}^T(r-1), \quad \sum_{r=k+1}^{k+N+1} \mathbf{Y}(r)\mathbf{u}^T(r), \quad \sum_{r=k+1}^{k+N+1} \mathbf{Y}(r)\mathbf{u}^T(r+1), \dots \\ &= \sum_{r=k+2}^{k+N+2} \mathbf{Y}(r)\mathbf{u}^T(r-2), \quad \sum_{r=k+2}^{k+N+2} \mathbf{Y}(r)\mathbf{u}^T(r-1), \quad \sum_{r=k+2}^{k+N+2} \mathbf{Y}(r)\mathbf{u}^T(r), \dots \end{aligned} \quad (\text{B1})$$

式 (B1) に現れるパターンから、ユーザーは表示されていないすべての要素を埋めるのに問題がないはずです。たとえば、すべての対角要素はその上下限を除いて同一であり、すべての副対角要素についても同じことが言えます。その結果、2 番目の対角要素は、最初の対角要素から次のように計算できます。

$$\sum_{r=k+1}^{k+N} \mathbf{Y}(r)\mathbf{u}^T(r) + \mathbf{y}(k+N)\mathbf{u}^T(k+N) - \mathbf{y}(k)\mathbf{u}^T(k) \quad (\text{B2})$$

帰納法により、対角 $m \times r$ サブマトリックスは次の再帰式で計算できます。

$$Z = \sum_{r=k+i+1}^{k+N+i} \mathbf{y}(r)\mathbf{u}^T(r) + \mathbf{y}(k+N+i)\mathbf{u}^T(k+N+i) - \mathbf{y}(k+i)\mathbf{u}^T(k+i) \quad (\text{B3})$$

ただし、 $i = 0, 1, \dots, p-1$ 。この再帰式は、各量が前の量から計算されることを示しています。同様に、最初の上部副対角サブマトリックスは、2 番目の m 行上の最初の上部副対角サブマトリックスから次のように計算できます。

$$\sum_{r=k+1}^{k+N} \mathbf{Y}(r)\mathbf{u}^T(r+1) = \sum_{r=k+1}^{k+N} \mathbf{y}(r)\mathbf{u}^T(r+1) + \mathbf{y}(k+N)\mathbf{u}^T(k+N+1) - \mathbf{y}(k)\mathbf{u}^T(k+1) \quad (\text{B4})$$

したがって、最初の上部副対角 $m \times r$ サブマトリックスは、次の再帰式で計算できます。

$$\sum_{r=k+i+1}^{k+N+i} \mathbf{y}(r)\mathbf{u}^T(r+1) = \sum_{r=k+i+1}^{k+N+i} \mathbf{Y}(r)\mathbf{u}^T(r+1) + \mathbf{y}(k+N+i)\mathbf{u}^T(k+N+i+1) - \mathbf{y}(k+i)\mathbf{u}^T(k+i+1) \quad (\text{B5})$$

まず、 $i = 0, 1, \dots, p-2$ に対して、二番目の上位オフダイアゴナルの $m \times r$ サブマトリックスは次のように再帰

的に計算されます：

$$\sum_{v=k+i+1}^{k+N+i} y(r)u^T(r+2) + y(k+N+i)u^T(k+N+i+2) - y(k+i)u^T(k+i+2) \quad (\text{B6})$$

ここで、 $i = 0, 1, \dots, p-3$ です。同様に、三番目、四番目、さらには $(p-1)$ 番目までの上位オフダイアゴナル要素は、最初の上位オフダイアゴナル要素が既知である場合に再帰的に計算することができます。最初の上位オフダイアゴナル要素は、積 $Y_p(k)U^T(k)$ の最初の m 行です。

同様に、下位の列も最初の m 列が計算され次第、計算が可能です。例えば、最初の下位オフダイアゴナル $m \times r$ サブマトリックスは次のように再帰的に計算されます：

$$\sum_{v=k+i+1}^{k+N+i} y(r)u^T(r-1) - y(k+N+i)u^T(k+N+i-1) + y(r)u^T(r-1) - y(k+i)u^T(k+i-1) \quad (\text{B7})$$

ここで、 $i = 1, \dots, p-1$ です。第二の下位オフダイアゴナル $m \times r$ サブマトリックスは次のように再帰的に計算されます：

$$\sum_{v=k+i+1}^{k+N+i} y(r)u^T(r-2) = \sum_{v=k+i+1}^{k+N+i-1} y(r)u^T(r-2) + y(k+N+i)u^T(k+N+i-2) - y(k+i)u^T(k+i-2) \quad (\text{B8})$$

ここで、 $i = 2, \dots, p-1$ です。帰納法により、三番目、四番目、さらには $(p-1)$ 番目までの下位オフダイアゴナル要素も、最初の下位オフダイアゴナル要素が決定され次第、再帰的に計算できます。式 (B8) における最初の下位オフダイアゴナル要素は、積 $Y_p(k)u^T(k)$ の最初の m 列です。

積 $Y_p(k)u^T(k)$ の計算は、最初の m 行および最初の m 列が計算され次第、再帰的に実行されます。長いデータ記録、すなわち $N \gg 1$ の場合、この再帰的手法は非常に効率的な計算時間を持ちます。 $Y_p(k)Y^T(k)$ および $U_p(k)U^T(k)$ の計算は、 $Y_p(k)u^T(k)$ の計算と類似しています。 $Y_p(k)Y^T(k)$ と $U_p(k)U^T(k)$ の両者は対称行列であるため、上半分または下半分のみを計算すれば十分です。