

核種生成・変換計算に関する ROM

北海道大学 千葉 豪

今回のセミナーのテーマ「Reduced order model」であるが、名大・遠藤先生がテキストで述べられているように「工学的知見に基づいた縮約モデル」と広く捉えるのであれば、それこそ様々なモデルが存在する。本稿のテーマ「核種生成・変換計算に関する Reduced order model」については、主に回帰モデルと固有直交分解 (POD) に関するものとなる。これらを理解するためには、特異値分解、擬似逆行列といった線形代数の基本的な知識が必要となるので、最初にそれらについての簡単な解説を行った上で、本題に入る流れとする。

読み飛ばしても問題ないものについては、このように枠で囲んで示すこととする。

1 行列の特異値分解と擬似逆行列

1.1 行列の特異値分解

行列の特異値分解については膨大な量のテキストがあり、身近なものとしても、第 50 回炉物理夏期セミナーテキストでの名大・山本先生のものや今回のセミナーでの遠藤先生のものがある。そこで、本稿では、行列の特異値分解については、最低限の解説を行う。

ランクが r の m 行 n 列の実数行列 \mathbf{A} ($\mathbf{A} \in \mathbb{R}^{m \times n}$ と書ける) は、直交行列 (正規直交行列とも呼ばれる) $\mathbf{U}_{m \times m}$ 、 $\mathbf{V}_{n \times n}$ と対角行列 $\mathbf{D}_{r \times r} = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_r)$ を用いて以下のように分解できる。

$$\mathbf{A} = \mathbf{U} \begin{pmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}_{m \times n} \mathbf{V}^T \quad (1)$$

なお、 σ_i は正の実数であり、降順に並ぶものとする。これらの σ_i を行列 \mathbf{A} の非ゼロの特異値と呼び、このような操作を行列の特異値分解と呼ぶ。 $\mathbf{U} = (\mathbf{u}_1 \ \mathbf{u}_2 \ \dots \ \mathbf{u}_m)$ 、 $\mathbf{V} = (\mathbf{v}_1 \ \mathbf{v}_2 \ \dots \ \mathbf{v}_n)$ と表すならば、式 (1) は以下のようにも書ける。

$$\mathbf{A} = \sum_{i=1}^r \mathbf{u}_i \sigma_i \mathbf{v}_i^T \quad (2)$$

行列 \mathbf{A} の j 番目の列ベクトル \mathbf{A}_{*j} を取り出すために、 j 番目のみの要素が 1 でそれ以外が 0 である、サイズが m の列ベクトル \mathbf{e}_j を式 (2) の両辺に右から乗ずると以下が得られる。

$$\mathbf{A}_{*j} = \mathbf{A} \mathbf{e}_j = \sum_{i=1}^r \mathbf{u}_i \sigma_i \mathbf{v}_i^T \mathbf{e}_j = \sum_{i=1}^r \mathbf{u}_i \sigma_i v_{i,j} \quad (3)$$

ここで、 $v_{i,j}$ は \mathbf{V} の i 列 j 行の要素に対応することに留意されたい。これより、行列 \mathbf{A} を構成する列ベクトルは、正規直交基底を構成するベクトル \mathbf{u}_i の重ね合わせにより記述され、例えば j 番目の列ベクトル \mathbf{A}_{*j} をこの正規直交基底で記述した場合、 i 番目の基底ベクトルに対応する係数が $\sigma_i v_{i,j}$ として与えられることが分かる。従って、例えばベクトルのセットが与えられたときに、それらが張る空間の正規直交基底を求めたいならば、それらを列ベクトルとした行列を構成し、それに対して特異値分解を施すことにより、求めたい正規直交基底が \mathbf{U} の (非ゼロの特異値に対応する成分) 列ベクトルとして得られることが分かる。これが後述する固有直交分解の基本的な考え方となる。

ここで、行列の固有値と固有ベクトルについて復習しておこう。 $\mathbf{A} \in \mathbb{R}^{m \times m}$ について、 $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$ が成り立つとき、 \mathbf{x} 、 λ をそれぞれ行列 \mathbf{A} の固有ベクトル、固有値と呼ぶ。固有ベクトルと固有値のセットはこの場合最大で m 個存在するので、それらを区別するために $\mathbf{A}\mathbf{x}_i = \lambda_i\mathbf{x}_i$ と表記する。仮にこれらのセットが m 個存在するとき、まとめて以下のように記述できる。

$$\mathbf{A} (\mathbf{x}_1 \ \mathbf{x}_2 \ \cdots \ \mathbf{x}_m) = (\mathbf{x}_1 \ \mathbf{x}_2 \ \cdots \ \mathbf{x}_m) \text{diag} (\lambda_1, \lambda_2, \cdots, \lambda_m) \quad (4)$$

従って、 $\mathbf{X} = (\mathbf{x}_1 \ \mathbf{x}_2 \ \cdots \ \mathbf{x}_m)$ 、 $\mathbf{D} = \text{diag} (\lambda_1, \lambda_2, \cdots, \lambda_m)$ のように定義すると、式 (4) は以下のように書き直せる。

$$\mathbf{A}\mathbf{X} = \mathbf{X}\mathbf{D} \quad (5)$$

さらに両辺の右から \mathbf{X}^{-1} を乗じると以下を得る。

$$\mathbf{X} = \mathbf{X}\mathbf{D}\mathbf{X}^{-1} \quad (6)$$

これらの形の式は以降に頻出するので、ここで慣れておくとよいだろう。

行列 $\mathbf{A}\mathbf{A}^T$ は $m \times m$ の正方行列となる。 \mathbf{D} は対角行列なので $\mathbf{D}^T = \mathbf{D}$ であることから、

$$\mathbf{A}\mathbf{A}^T = \mathbf{U} \begin{pmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}_{m \times n} \mathbf{V}^T \mathbf{V} \begin{pmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}_{n \times m} \mathbf{U}^T \quad (7)$$

が得られる。 \mathbf{U} 、 \mathbf{V} は直交行列なので、 $\mathbf{U}\mathbf{U}^T = \mathbf{U}^T\mathbf{U} = \mathbf{I}_{m \times m}$ 、 $\mathbf{V}\mathbf{V}^T = \mathbf{V}^T\mathbf{V} = \mathbf{I}_{n \times n}$ が成り立つ (\mathbf{I} は単位行列を示す)。従って、式 (7) は以下のように書き直せる。

$$\mathbf{A}\mathbf{A}^T = \mathbf{U} \begin{pmatrix} \mathbf{D}^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}_{m \times m} \mathbf{U}^T \quad (8)$$

ここで、 $(\mathbf{D}^2)_{r \times r} = \text{diag} (\sigma_1^2, \sigma_2^2, \dots, \sigma_r^2)$ である。式 (8) の右から \mathbf{U} を乗じると以下が得られる。

$$\mathbf{A}\mathbf{A}^T\mathbf{U} = \mathbf{U} \begin{pmatrix} \mathbf{D}^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}_{m \times m} \quad (9)$$

これより、行列 $\mathbf{A}\mathbf{A}^T$ の i 番目 (ただし $i \leq r$) の固有値と固有ベクトルが σ_i^2 、 \mathbf{u}_i であることが分かる。

同様に、行列 $\mathbf{A}^T\mathbf{A}$ は $n \times n$ の正方行列となり、上と同様の手続きを行うことによって以下が得られる。

$$\mathbf{A}^T\mathbf{A}\mathbf{V} = \mathbf{V} \begin{pmatrix} \mathbf{D}^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}_{n \times n} \quad (10)$$

これより、行列 $\mathbf{A}^T\mathbf{A}$ の i 番目 (ただし $i \leq r$) の固有値と固有ベクトルが σ_i^2 、 \mathbf{v}_i であることが分かる。

実対称行列の特異値分解と固有値分解の関係について考えよう。

ランクが m の対称行列 $A \in \mathbb{R}^{m \times m}$ を考える。これを特異値分解すると $A = UDV^T$ が得られ、これを転置することにより $A^T = VDU^T$ が得られる。対称行列であることから $A = A^T$ なので、これより $U = V$ となることが分かり、 $A = VDV^T$ が得られる。 $V^T = V^{-1}$ なので、 $A = VDV^{-1}$ と書き直せるが、これは行列 A の固有値分解に相当することが分かる。つまり、行列 A の固有値は D の対角成分である σ_i となるが、特異値は正の実数となることが分かっているので、固有値もそうならなければならない。従って、実対称行列の固有値は全て正でなければならないということになるが、決してそのようなことは無い。では、これまでの議論のどこに瑕疵があったのだろうか？

$UDV^T = VDU^T$ において、 D が対角行列であることに着目すると、この式は実は $U_{*i} = -V_{*i}$ であるときにも成り立つ。仮に固有値が全て負となる場合を考えると、 $U = -V$ とすれば、 $A = -VDV^T = V(-D)V^T$ となり、 $AV = V(-D)$ より、固有値が全て負になることが分かる。つまり、正の固有値のみを持つ実対称行列を特異値分解したときには U と V として同一のものが得られるが、負の固有値をもつ行列の場合には、 U と V の列ベクトルの符号が異なるものが得られることになる。

1.2 擬似逆行列

逆行列は正則行列（非特異行列、可逆行列とも呼ばれる）に対して定義されるが、正則行列に限らない、任意の大きさの行列に対して擬似逆行列というものが定義できる。以降では、行列 A の擬似逆行列を A^\dagger のように表記する。擬似逆行列は以下の性質を持つ（実数行列の擬似逆行列が満足する Moore-Penrose 条件に相当する）。

$$AA^\dagger A = A, \quad A^\dagger AA^\dagger = A^\dagger, \quad (AA^\dagger)^T = AA^\dagger, \quad (A^\dagger A)^T = A^\dagger A \quad (11)$$

加えて、以下も成り立つ。

$$(A^T A)^\dagger = A^\dagger (A^T)^\dagger, \quad (AA^T)^\dagger = (A^T)^\dagger A^\dagger \quad (12)$$

A^\dagger は行列 A の特異値分解の結果得られる U 、 V 、 D を用いて以下のようにも定義できる。

$$A^\dagger = V \begin{pmatrix} D^{-1} & 0 \\ 0 & 0 \end{pmatrix} U^T \quad (13)$$

注意しなければならないのが、 $A^\dagger A = I$ が成り立たない点である。このことを式 (13) を用いて実際に確かめてみよう。

$$AA^\dagger = U \begin{pmatrix} D & 0 \\ 0 & 0 \end{pmatrix} V^T V \begin{pmatrix} D^{-1} & 0 \\ 0 & 0 \end{pmatrix} U^T = U \begin{pmatrix} D & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} D^{-1} & 0 \\ 0 & 0 \end{pmatrix} U^T = U \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} U^T \quad (14)$$

なお、これに右から A を乗じると、以下のように A に戻ることが確認できる。

$$U \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} U^T U \begin{pmatrix} D & 0 \\ 0 & 0 \end{pmatrix} V^T = U \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} D & 0 \\ 0 & 0 \end{pmatrix} V^T = U \begin{pmatrix} D & 0 \\ 0 & 0 \end{pmatrix} V^T \quad (15)$$

行列 $A \in \mathbb{R}^{m \times m}$ を考え、そのランクが m であるときには、式 (14) 中の部分的に単位行列となっている行列が単位行列となるため $A^\dagger A = I$ が成り立つ。この場合、 $A^\dagger = A^{-1}$ 、つまり擬似逆行列が逆行列と一致する。

2 回帰の基礎

2.1 最小二乗法による線形回帰

パラメータ x に依存するパラメータ y があり、 $y = ax + d$ なる関係があるとする。測定により、 I 個の異なる x の値に対して y の値が得られたものとし、そのセットを $(x_1, y_1), (x_2, y_2), \dots, (x_I, y_I)$ と記述する。測定値には測定に伴う誤差が混入するため、個々のデータセットの関係は $y_i = ax_i + d + u_i$ と記述される。なお、測定誤差（擾乱項とも呼ばれる） u_i は当然のことながら未知である。

この I 個のデータセットから係数 a 、 d を求めるとき、最小二乗法では、以下で定義される G を最小化するという条件を課す。

$$G = \sum_{i=1}^I \{(ax_i + d) - y_i\}^2 \quad (16)$$

G が最小値をとる場合には、以下の式が満足される。

$$\frac{\partial G}{\partial a} = 0, \quad (17)$$

$$\frac{\partial G}{\partial d} = 0 \quad (18)$$

これらに式 (16) を実際に代入することにより、以下を得る。

$$\frac{\partial G}{\partial a} = \sum_{i=1}^I 2(ax_i + d - y_i) \cdot x_i = 2 \left\{ a \sum_{i=1}^I x_i^2 + d \sum_{i=1}^I x_i - \sum_{i=1}^I x_i y_i \right\} = 0, \quad (19)$$

$$\frac{\partial G}{\partial d} = \sum_{i=1}^I 2(ax_i + d - y_i) = 2 \left\{ a \sum_{i=1}^I x_i + d \sum_{i=1}^I 1 - \sum_{i=1}^I y_i \right\} = 0 \quad (20)$$

式 (19)、(20) は以下のようにまとめられる。

$$a \sum_{i=1}^I x_i^2 + d \sum_{i=1}^I x_i = \sum_{i=1}^I x_i y_i, \quad (21)$$

$$a \sum_{i=1}^I x_i + d \sum_{i=1}^I 1 = \sum_{i=1}^I y_i \quad (22)$$

2 個の未知数 a 、 d に対して 2 本の方程式が定義されるため、未知数 a 、 d が一意に決まることが分かるであろう¹。

もし、 $y = ax + d$ ではなく、 $y = d$ のように y が x に依存しない振る舞いをするとう仮定するならばどうなるであろうか。この場合は、式 (20) において $a = 0$ とすればよく、以下の式が得られる。

$$d \sum_{i=1}^I 1 - \sum_{i=1}^I y_i = 0 \quad (23)$$

従って、未知数 d は以下のように決まる。

$$d = \frac{\sum_{i=1}^I y_i}{\sum_{i=1}^I 1} = \frac{\sum_{i=1}^I y_i}{I} \quad (24)$$

これは、 I 個の y_i の平均をとっていることに他ならず、直感と一致するであろう。

¹ そうとはならない場合もあるがここでは省略する。

式 (21)、(22) は行列形式で以下のように書ける。

$$\begin{pmatrix} \sum_{i=1}^I x_i^2 & \sum_{i=1}^I x_i \\ \sum_{i=1}^I x_i & \sum_{i=1}^I 1 \end{pmatrix} \begin{pmatrix} a \\ d \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^I x_i y_i \\ \sum_{i=1}^I y_i \end{pmatrix} \quad (25)$$

これを行列 M 、ベクトル \mathbf{b} 、 \mathbf{n} を用いて以下のように記述する。

$$M\mathbf{b} = \mathbf{n} \quad (26)$$

未知数 a 、 d を要素に持つベクトル \mathbf{b} は、行列 M に逆行列が存在すれば (M が正則行列であれば)、以下のように一意に決まる。

$$\mathbf{b} = M^{-1}\mathbf{n} \quad (27)$$

ここで、以下の行列、ベクトルを定義する。

$$\mathbf{X} = \begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_I & 1 \end{pmatrix}, \quad \mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_I \end{pmatrix} \quad (28)$$

これら \mathbf{X} 、 \mathbf{y} を用いると、式 (16) は以下のように書くことができる。

$$G = \sum_{i=1}^I \{(ax_i + d) - y_i\}^2 = (\mathbf{X}\mathbf{b} - \mathbf{y})^T (\mathbf{X}\mathbf{b} - \mathbf{y}) \quad (29)$$

これより、最小二乗法は「 $\mathbf{X}\mathbf{b}$ (モデルによる予測値) と \mathbf{y} (実測値) の差分に対応するベクトル $(\mathbf{X}\mathbf{b} - \mathbf{y})$ の 2-ノルムを最小化する方法」とも言い換えられる。

式 (26) で定義した M 、 \mathbf{n} は \mathbf{X} 、 \mathbf{y} を用いて以下のように記述できる。

$$M = \mathbf{X}^T \mathbf{X} = \begin{pmatrix} x_1 & x_2 & \dots & x_I \\ 1 & 1 & \dots & 1 \end{pmatrix} \begin{pmatrix} x_1 & 1 \\ x_2 & 1 \\ \vdots & \vdots \\ x_I & 1 \end{pmatrix}, \quad (30)$$

$$\mathbf{n} = \mathbf{X}^T \mathbf{y} = \begin{pmatrix} x_1 & x_2 & \dots & x_I \\ 1 & 1 & \dots & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_I \end{pmatrix} \quad (31)$$

従って、最小二乗法で定義された式 (26) ($M\mathbf{b} = \mathbf{n}$) は、 \mathbf{X} 、 \mathbf{y} を用いて以下のように書き換えられる。

$$\mathbf{X}^T \mathbf{X} \mathbf{b} = \mathbf{X}^T \mathbf{y} \quad (32)$$

前述したように、個々のデータセット $(x_1, y_1), (x_2, y_2), \dots, (x_I, y_I)$ は以下の方程式を満足する。

$$y_1 = ax_1 + d + u_1, \quad (33)$$

$$y_2 = ax_2 + d + u_2, \quad (34)$$

\vdots

$$y_I = ax_I + d + u_I \quad (35)$$

これらの式は、 \mathbf{X} 、 \mathbf{y} を用いて行列形式で以下のように書ける。

$$\mathbf{y} = \mathbf{X}\mathbf{b} + \mathbf{u} \quad (36)$$

ここで、 $\mathbf{u} = (u_1 \ u_2 \ \dots \ u_I)^T$ である。この式について、 $\mathbf{X}\mathbf{b}$ を左辺に移し、両辺の左から \mathbf{X}^T を乗ずると以下を得る。

$$\mathbf{X}^T\mathbf{y} - \mathbf{X}^T\mathbf{X}\mathbf{b} = \mathbf{X}^T\mathbf{u} \quad (37)$$

式 (32) から明らかなように、最小二乗法では式 (37) がゼロとなるように \mathbf{b} を決めていることになる。言い換えれば、最小二乗法では「モデルによる予測値と実測値の差分 ($\mathbf{y} - \mathbf{X}\mathbf{b}$) に \mathbf{X}^T を作用させたものがゼロとなるように (差分のベクトルと \mathbf{X} を構成する列ベクトルが直交するように) \mathbf{b} を決めている」と理解することも出来る。

G の b_i (この例では a 、 d) に対する偏微分がゼロになる条件は以下のように書ける。

$$\frac{\partial G}{\partial b_i} = (\mathbf{X}_{*i})^T (\mathbf{X}\mathbf{b} - \mathbf{y}) + (\mathbf{X}\mathbf{b} - \mathbf{y})^T \mathbf{X}_{*i} = 0 \quad (38)$$

この偏微分がスカラー量であることを考えると、 $(\mathbf{X}_{*i})^T (\mathbf{X}\mathbf{b} - \mathbf{y})$ を転置すると $(\mathbf{X}\mathbf{b} - \mathbf{y})^T \mathbf{X}_{*i}$ が得られることから、両者は同一であることが分かる。従って、

$$(\mathbf{X}_{*i})^T (\mathbf{X}\mathbf{b} - \mathbf{y}) = 0 \quad (39)$$

が得られる。そして、これを全ての i についてまとめると以下の式が得られ、これは式 (32) と一致する。

$$\mathbf{X}^T (\mathbf{X}\mathbf{b} - \mathbf{y}) = 0 \quad (40)$$

これより、予測と実測の差分のベクトルと \mathbf{X} を構成する列ベクトルが直交するように \mathbf{b} を決めることは、式 (29) で定義される G を最小化する条件 $\frac{\partial G}{\partial \mathbf{b}} = 0$ と等価であることが分かる。

これまでの議論では、入出力パラメータの関係を $y = ax + d$ と記述していたが、これを一般化して $y = \sum_{j=1}^J a_j f_j(x)$ のもとで I 個の (x_i, y_i) のデータセットから a_j を求める問題に置き換えることが出来る。このときの \mathbf{X} 、 \mathbf{b} はそれぞれ以下のように定義される。

$$\mathbf{X} = \begin{pmatrix} f_1(x_1) & f_2(x_1) & \dots & f_J(x_1) \\ f_1(x_2) & f_2(x_2) & \dots & f_J(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ f_1(x_I) & f_2(x_I) & \dots & f_J(x_I) \end{pmatrix}, \quad \mathbf{b} = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_J \end{pmatrix} \quad (41)$$

式 (32) から、 \mathbf{b} は以下のように求められる。

$$\mathbf{b} = (\mathbf{X}^T\mathbf{X})^{-1} \mathbf{X}^T\mathbf{y} \quad (42)$$

これより、 \mathbf{X} と \mathbf{y} さえ計算してしまえば、あとは行列とベクトルの簡単な演算により \mathbf{b} が求められると考えるかもしれないが、これをそのまま数値的に計算するのが危険であることは線形代数のテキストでは常識とされている。後述するが、このことを知った上でも甘く見て、このような計算をして \mathbf{b} を求めると、後でしっぺ返しを喰らうことになる。

\mathbf{b} を安定に求める方法として QR 分解を使う方法が挙げられる。QR 分解により $\mathbf{X} = \mathbf{Q}\mathbf{R}$ とする ($\mathbf{Q}^T = \mathbf{Q}^{-1}$ を満足し、 \mathbf{R} は正の対角成分を持つ上三角行列である) と以下が得られる。

$$\mathbf{X}^T\mathbf{X} = (\mathbf{Q}\mathbf{R})^T \mathbf{Q}\mathbf{R} = \mathbf{R}^T \mathbf{Q}^T \mathbf{Q}\mathbf{R} = \mathbf{R}^T\mathbf{R} \quad (43)$$

従って、式 (32) ($\mathbf{X}^T \mathbf{X} \mathbf{b} = \mathbf{X}^T \mathbf{y}$) は以下のように書ける。

$$\mathbf{R}^T \mathbf{R} \mathbf{b} = \mathbf{R}^T \mathbf{Q}^T \mathbf{y} \quad (44)$$

\mathbf{R}^T は正則行列なので、

$$\mathbf{R} \mathbf{b} = \mathbf{Q}^T \mathbf{y} \quad (45)$$

が得られ、 $\mathbf{b} = \mathbf{R}^{-1} \mathbf{Q}^T \mathbf{y}$ により \mathbf{b} が得られる。この方法では、 $\mathbf{X}^T \mathbf{X}$ ではなく \mathbf{R} の逆行列を求めることにより、数値的な安定性が増している。

最小二乗法で作成された、 \mathbf{y} を予測するモデル $\mathbf{X} \mathbf{b}$ を構成するパラメータ \mathbf{b} の共分散行列 $V(\mathbf{b})$ は以下のように得られる。

$$V(\mathbf{b}) = (\mathbf{X}^T \mathbf{X})^{-1} \sigma^2 \quad (46)$$

ここで σ^2 は線形モデルの誤差分散と呼ばれるパラメータであり、以下がそれに対する不偏推定量となっている。

$$\sigma^2 = (\mathbf{y} - \mathbf{X} \mathbf{b})^T (\mathbf{y} - \mathbf{X} \mathbf{b}) / \nu_e = S(\mathbf{b}) / \nu_e \quad (47)$$

ここで ν_e は自由度であり、一般的にはデータ数からパラメータ数を差し引いた数値となる^a。

^aNumpy の `curve_fit` 関数が返す共分散行列はこの方法によって計算されている。誤差を含むデータを `curve_fit` 関数に適用する場合、この関数が返す共分散行列はスケールされたものになっている ($S(\mathbf{b}) / \nu_e$ が乗せられている)。そのような操作が施されない共分散行列を計算したい場合には、引数の `absolute_sigma` を `True` とする必要がある。

2.2 その他の回帰手法

連立一次方程式の本数（以降ではデータの個数とも言い換える）と未知数の個数（入出力を関係付けるモデルパラメータの個数とも言い換える）が一致する場合に、個々の方程式が線形独立であれば未知数が一意に決まる。未知数を示すベクトルを $\mathbf{b} \in \mathbb{R}^I$ 、方程式の係数行列を $\mathbf{X} \in \mathbb{R}^{I \times I}$ 、定数項を $\mathbf{y} \in \mathbb{R}^I$ としたとき、この問題は以下のように記述される。

$$\mathbf{X} \mathbf{b} = \mathbf{y} \quad (48)$$

個々の方程式が線形独立であることは、 \mathbf{X} が正則行列であることと同義であるため、未知数が以下のように決まる。

$$\mathbf{b} = \mathbf{X}^{-1} \mathbf{y} \quad (49)$$

一方、データの個数がモデルパラメータの個数より大きく、個々の方程式が線形独立である場合には、式 (48) を満足する \mathbf{b} は存在しない。このような場合には、最小二乗法を適用すると以下が得られる。

$$\mathbf{b} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} \quad (50)$$

このようにして得られる \mathbf{b} は $\mathbf{X} \mathbf{b} - \mathbf{y}$ の 2-ノルムを最小化するものとして決められることは前述した。

逆に、データの個数がモデルパラメータの個数よりも小さい場合や、両者が一致していても方程式に線形独立が成り立たない場合には、式 (48) を満足する解が無限に存在することになる。従って、一意の解を得るためには何らかの束縛条件を追加する必要がある。このような場合、よく知られた擬似逆行列を用いる方法では、以下のように \mathbf{b} が得られる。

$$\mathbf{b} = \mathbf{X}^\dagger \mathbf{y} \quad (51)$$

この式は以下の手続きで得られる。 $\mathbf{X} \mathbf{b} = \mathbf{y}$ の両辺に $\mathbf{X} \mathbf{X}^\dagger$ を乗じると以下が得られる。

$$\mathbf{X} \mathbf{X}^\dagger \mathbf{X} \mathbf{b} = \mathbf{X} \mathbf{X}^\dagger \mathbf{y} \quad (52)$$

$\mathbf{X} \mathbf{X}^\dagger \mathbf{X} = \mathbf{X}$ が成り立つので

$$\mathbf{X} \mathbf{b} = \mathbf{X} (\mathbf{X}^\dagger \mathbf{y}) \quad (53)$$

より、式 (51) が得られる。このようにして定義される \mathbf{b} は式 (48) を満たすもののうち 2-ノルムを最小化するものに対応する²。

$\mathbf{X}\mathbf{b} = \mathbf{y}$ の 2-ノルム最小の解が $\mathbf{X}^\dagger\mathbf{y}$ として得られることは線形代数のテキストのどこかに記述があると思うが、ここでは K 君との議論の結果をまとめる。 $\mathbf{b}_0 = \mathbf{X}^\dagger\mathbf{y}$ としたとき、 \mathbf{b}_0 は以下のように書けることが分かっている。

$$\mathbf{b}_0 = \mathbf{V} \begin{pmatrix} \mathbf{D}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \mathbf{U}^T \mathbf{y} \quad (54)$$

これより、 \mathbf{b}_0 は $S = \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_r\}$ が張る空間に存在することが分かる。一方、 $\mathbf{X}\mathbf{b} = \mathbf{y}$ を満たす \mathbf{b} を $\mathbf{b}_0 + \boldsymbol{\epsilon}$ とするならば、 $\mathbf{X}\boldsymbol{\epsilon} = \mathbf{0}$ でなければならないので以下が成り立つ。

$$\mathbf{U} \begin{pmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \mathbf{V}^T \boldsymbol{\epsilon} = \mathbf{0} \quad (55)$$

これより、 $\mathbf{v}_i^T \boldsymbol{\epsilon} = 0$ (ただし $i \leq r$) であること、すなわち $\boldsymbol{\epsilon}$ は S に直交する ($\boldsymbol{\epsilon}$ と \mathbf{b}_0 が直交する) ことが分かる。従って、 $\|\mathbf{b}_0 + \boldsymbol{\epsilon}\|_2^2 = \|\mathbf{b}_0\|_2^2 + \|\boldsymbol{\epsilon}\|_2^2 + 2\mathbf{b}_0 \cdot \boldsymbol{\epsilon} = \|\mathbf{b}_0\|_2^2 + \|\boldsymbol{\epsilon}\|_2^2$ より $\|\mathbf{b}_0 + \boldsymbol{\epsilon}\|_2 > \|\mathbf{b}_0\|_2$ が成り立つので、 \mathbf{b}_0 が 2-ノルム最小解であることが分かる。なお、 $\|\mathbf{b}\|_i$ は \mathbf{b} の i -ノルムを示す。

以上で述べた方法では、モデルによる予測値と実測値とのずれを極力小さくするようにモデルパラメータを決定しようとしている。一方、モデルパラメータの観点から考えると、その個々の絶対値が大きくなればなるほど、入力パラメータの僅かな変動が推定結果に大きな影響を及ぼすことになるため、出来るだけ小さな値とすること、すなわち、モデルパラメータのベクトル \mathbf{b} のノルムを小さくすることが望ましいと言える。そのような考え方に基づき、式 (48) を厳密に満足させることの優先度を落とし、その代わり \mathbf{b} のノルムが小さくなるモデルを構築する方法があり、Ridge 回帰と Lasso 回帰がよく知られている。

Ridge 回帰では以下が最小化されるように \mathbf{b} を決める。

$$(\mathbf{X}\mathbf{b} - \mathbf{y})^2 + \lambda \|\mathbf{b}\|_2 \quad (56)$$

ここで、 λ は任意のパラメータ (超パラメータ) である。この \mathbf{b} は以下のように与えられる。

$$\mathbf{b} = (\mathbf{X}^T \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^T \mathbf{y} \quad (57)$$

λ をゼロとした場合には、 $\mathbf{X}^T \mathbf{X}$ が正則行列であれば最小二乗解に一致する。また、方程式の数が未知数の数よりも小さい場合などのように $\mathbf{X}^T \mathbf{X}$ が正則でないときは、逆行列ではなく擬似逆行列をとればよい。すると、

$$\mathbf{b} = (\mathbf{X}^T \mathbf{X})^\dagger \mathbf{X}^T \mathbf{y} = \mathbf{X}^\dagger \mathbf{X}^T \mathbf{X}^\dagger \mathbf{X}^T \mathbf{y} = \mathbf{X}^\dagger \mathbf{y} \quad (58)$$

となり、前述の擬似逆行列を用いる方法と一致することが分かる。なお、 λ を無限大にした場合には、 $\mathbf{b} \rightarrow \mathbf{0}$ となる。

Lasso 回帰では以下が最小化されるように \mathbf{b} を決める。

$$(\mathbf{X}\mathbf{b} - \mathbf{y})^2 + \lambda \|\mathbf{b}\|_1 \quad (59)$$

Lasso 回帰では Ridge 回帰と異なり、 \mathbf{b} は式の形式では与えられず、何らかの数値解法が必要となる。

例として、二次元平面上の点 (2,1) を通る一次直線 $y = ax + b$ の定数 a 、 b を決める問題を考えよう。この点を通る一次直線の候補は無数にある。図 1 左に、 a を 0 から 3 としたときに、 $y = ax + b$ を満足するように決めた (a, b) の 2-ノルムを示す。擬似逆行列を用いて (a, b) を決めた場合には $(a, b) = (0.4, 0.1)$ が得

²勿論、最小化するものを 2-ノルムと限定する必要はない。例えば、1-ノルムを最小化したときの検討例については、炉物理分野では JAEA 渡邊氏が学生時代に発表した論文がある (*Trans. Am. Nucl. Soc.*, 111, p.1391 (2014).)。

られるが、その解の 2-ノルムが最小となっていることが確認できる。次に、同図右に、Ridge 回帰で λ を 0 から 3 の値としたときの (a, b) の 2-ノルムと回帰式の誤差を示す。 λ をゼロとした場合には、回帰式の誤差がゼロとなり、 (a, b) のノルムは擬似逆行列を用いた場合のものと同一となるが、 λ を大きくすることにより、回帰式の誤差が大きくなる一方で、 (a, b) のノルムは擬似逆行列を用いた場合と比べてさらに小さくなる様子が分かる。

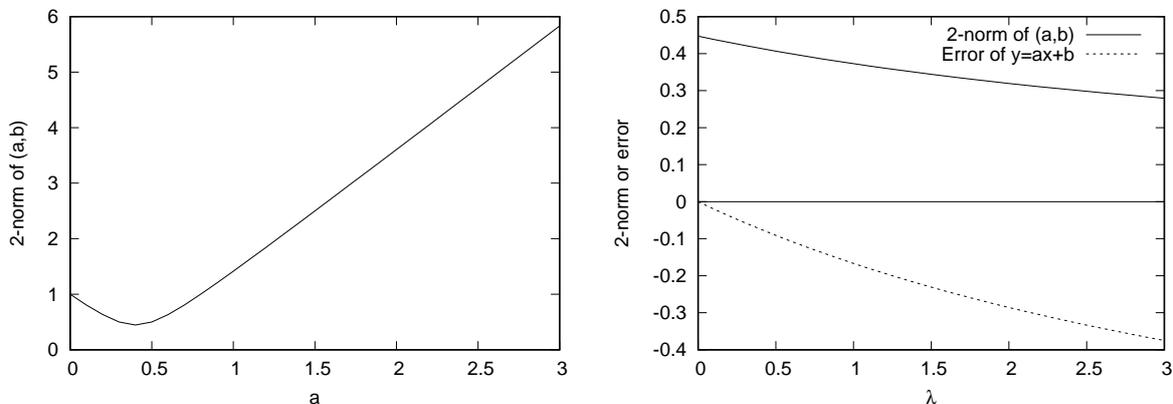


図 1: 簡易な回帰問題の例

3 線形回帰モデルの例

3.1 四因子公式の予測モデル

核燃料の燃焼により、燃料中の核種の生成・変換が起こり、燃料の炉物理特性は大きく変化する。ここでは、四因子公式中の 2 つの因子について、軽水炉の燃料ピンセルでの燃焼中の変化を予測するための回帰モデルについて説明する。

軽水炉ピンセルについて、初期燃料の U-235 濃縮度をいくつかの値に変化させた燃焼計算を行い、四因子の熱中性子利用率 f と中性子再生率 η の積について、初期燃料の U-235 濃縮度と燃焼度に対する依存性を得た。結果を図 2 に示すが、燃焼度の増加及び濃縮度の減少に伴い、核分裂性物質の減少、核分裂生成物の蓄積等により、 $f\eta$ が減少していく様子が分かる。

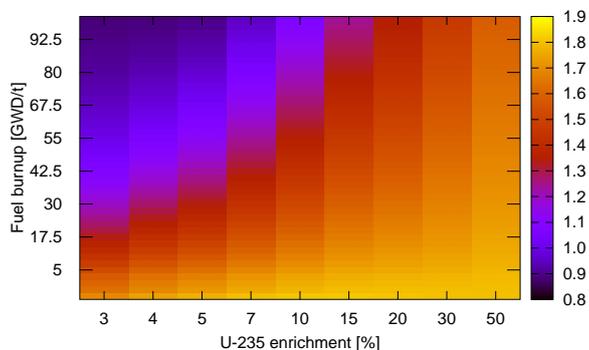


図 2: $f\eta$ の濃縮度、燃焼度に対する依存性

この $f\eta$ について、濃縮度を x と燃焼度を y としたときの線形回帰モデルを上述の方法により作成した。回帰式を以下に示す。

$$f\eta = \sum_{i=0}^I \sum_{j=0}^{j=i} \alpha_{i,j} x^j y^{i-j} \quad (60)$$

実質的には、 x に関する I 次の多項式と y に関する I 次の多項式の積で記述したものと考えればよい。回帰式の作成に用いたデータは、濃縮度については 9 点、燃焼度については 43 点であるので、この回帰式で設定できる I の最大値は 8 となる。

このようにして作成した回帰モデルについて、回帰モデルの作成に用いたデータに対する再現精度を評価した。ここで、濃縮度については 3 から 50 wt% まで異なる間隔で (値が小さいほど細かく) データ点を採っていることから、濃縮度をそのまま x として扱ったもの (Case 1) に加えて、その対数を x としたもの (Case 2) で回帰モデルを作成した。これらの方法に基づく回帰モデルの再現誤差を Fig. 3 に示す。濃縮度について対数をとったほうが再現精度が向上することが分かる。

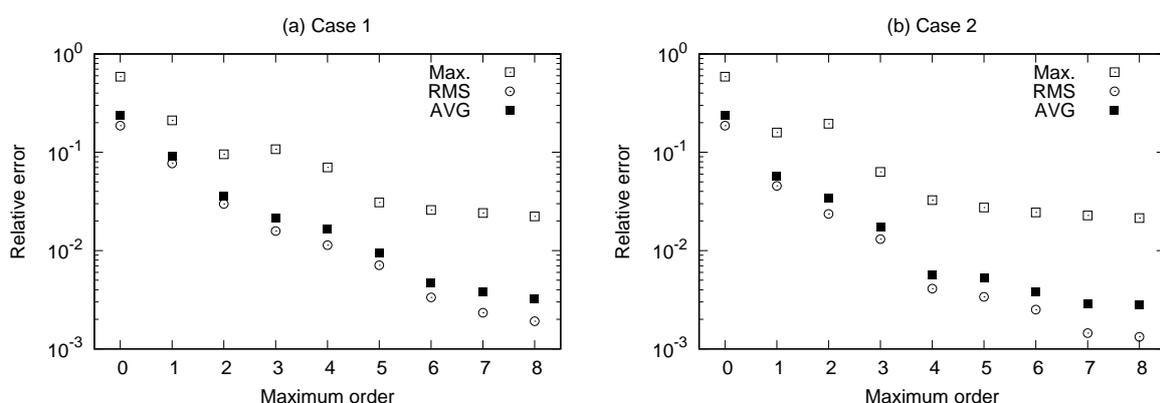


図 3: $f\eta$ に対する回帰モデルの再現誤差 (濃縮度をそのまま扱った場合と対数をとった場合の比較)

次いで、 x と y のクロスタームを無視して回帰モデルを作成した場合 (Case 3) の結果を Case 1 と比較したものを図 4 に示す。クロスタームを無視した場合は、回帰式は x と y の多項式の単純和となるため、最大次数を大きくしたときの精度の改善が見られない。

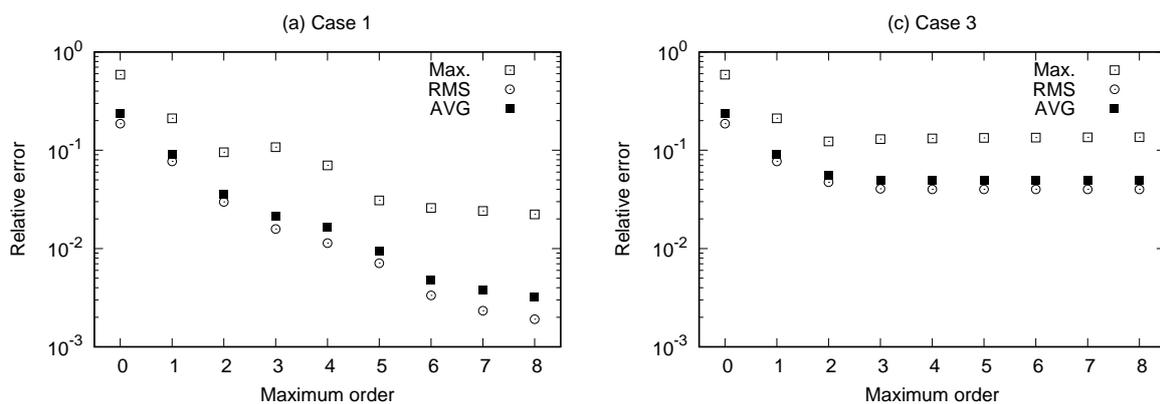


図 4: $f\eta$ に対する回帰モデルの再現誤差 (クロスタームを考慮した場合と無視した場合の比較)

最後に、QR 分解ではなく、 $A^T A$ の逆行列を用いて回帰式を作成した場合 (Case 4) の結果を Case 1 と比較したものを図 5 に示す。最大次数を 8 としたときに、QR 分解を用いない場合には大幅な誤差の拡大が観察される。

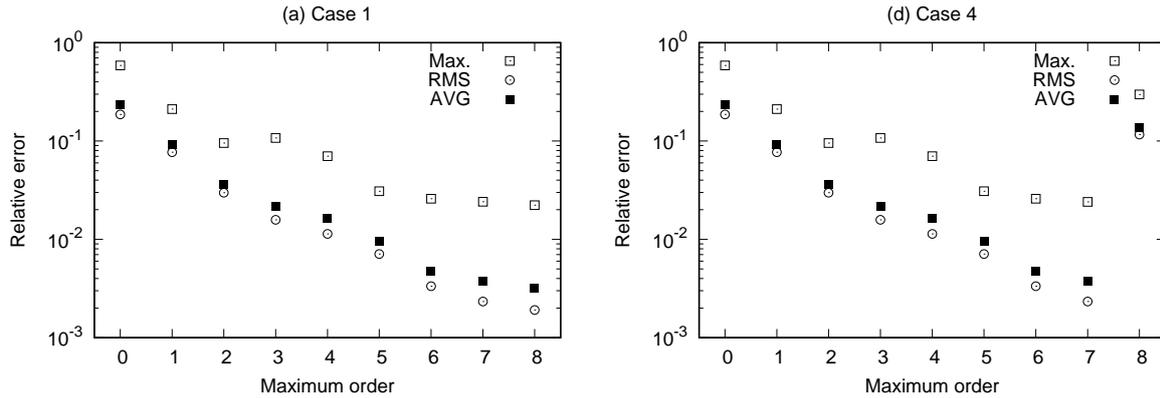


図 5: f_η に対する回帰モデルの再現誤差 (QR 分解を用いた場合と用いない場合の比較)

図 5 では、QR 分解を用いない場合の回帰モデルの予測精度の悪化の一例を示したが、ここでは、濃縮度の値を固定し、43 点の燃焼度のデータから燃焼度のみ依存する回帰式を作成したときの、QR 分解の適用の影響を示す。このときにはデータ点数が多いことから、多項式の次数をより大きく設定することが可能となる。得られた回帰式の再現誤差を Fig. 6 に示す。QR 分解を適用した場合には最大次数が 20 までは安定的に良好な回帰モデルによる予測が行えていることが分かる。理論上は 42 次まで良好に予測できるものと考えられるが、そうならず 21 次以降で再現誤差が急拡大するのは数値誤差によるものと考えられるが定かではない。一方、QR 分解を適用しない場合には 11 次程度から誤差の拡大が観察される。

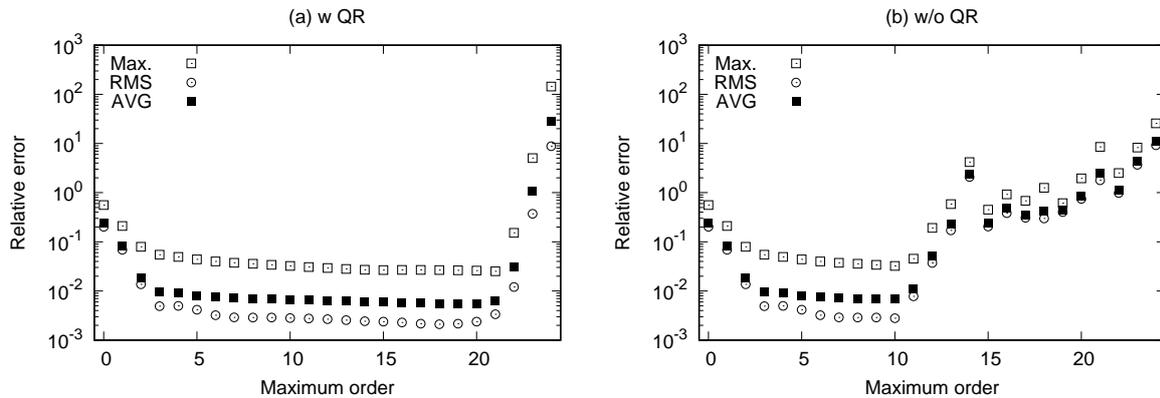


図 6: f_η に対する回帰モデルの再現誤差 (燃焼度のみを説明変数とした場合)

3.2 移行期高速炉の炉心特性の予測モデル

高速炉の炉心計算を行う際には、炉心に装荷する核燃料について、原子炉での燃焼、冷却、再処理、燃料加工を複数回繰り返した状況 (多重リサイクル) を想定し平衡状態となった TRU 組成を用いる。このような扱いは、複数の高速炉が建設され、大規模エネルギー源として使われること (いわゆる「平衡期」) を想定したものであるが、そのような時期が到来する前には、軽水炉から高速炉に切り替わる過渡期 (いわゆる「移行期」) が存在し、そこでは軽水炉の使用済み燃料から取り出された TRU が燃料として装荷される。移行期高速炉の炉心計算を行う際には、代表的な条件で得られた TRU 組成を想定することが考えられるが、移行期をより現実的に考えるとすれば、軽水炉からの使用済み燃料は様々な燃料仕様や運転方法に基づいて発生するため、移行期の高速炉は多様な TRU 組成の燃料を受け入れることになる。移行期を想定し

た高速炉の炉心設計では、想定し得る範囲の TRU 組成条件について計算を行い、設計条件が満足されることを確認する必要がある。

移行期のための高速炉の炉心設計をするにあたって、移行期が現実的にイメージされる時期に差し掛かった場合には、装荷する燃料の TRU 組成の情報も具体化されるものと考えられる。つまり、高速炉燃料の基となる軽水炉使用済み燃料の照射・冷却の履歴が明確化される、ということである。このような場合には、様々な軽水炉の燃料仕様や運転条件に基づく燃焼計算を行い、その結果に基づいて移行期高速炉の燃料組成を決めなければならないものと考えられる。また、様々な組成の TRU 組成を受け入れた移行期高速炉は、その炉心特性や、使用済み燃料中の核種インベントリ（廃棄物特性）も、燃料の TRU 組成に依存して異なる。

このように、移行期高速炉が受け入れる TRU の基となる使用済み燃料を生み出す軽水炉の燃料仕様、運転条件が与えられたときの、移行期高速炉の炉心特性や廃棄物特性を評価したい場合には、軽水炉燃料の燃焼計算と高速炉燃料の燃焼計算を連続して行う必要がある。前者は燃料ピンセルモデルで代用できるが、後者については、全炉心モデルでの計算が基本となるので、こういった評価を種々の条件について直接的に行うことは現実的ではない。従って、入力条件と出力パラメータの関係を、想定しうる複数の条件について予め得ておき、それに基づいた回帰モデルを作成し利用することは有益と考えられる。

そこで、当研究室では、移行期高速炉の燃料として受け入れることが想定される軽水炉の使用済み燃料について、軽水炉の燃料仕様や運転条件が入力情報として与えられたとき、その使用済み燃料を受け入れた移行期高速炉の炉心特性と廃棄物特性を予測するモデルを開発した。

入力情報としては以下を想定した。

- 炉型 (PWR/BWR)
- U-235 濃縮度
- 燃焼度
- 水素対重元素比 (H/HM 比)
- 全 TRU に対する Pu-239 の同位体比 (MOX 燃料の場合)
- 全 TRU に対する Pu-240 の同位体比 (MOX 燃料の場合)
- 軽水炉から取り出して再処理した後に、高速炉に装荷されるまでの冷却時間

また、出力情報として想定したものは以下とした。

- 炉心特性 (Pu 富化度、燃焼反応度、冷却材ボイド反応度、ドップラー反応度、実効遅発中性子割合)
- 取り出し燃料中の 21 の重核種 (U、Np、Pu、Am、Cm) のインベントリ

これらの入出力情報を直接関連付ける回帰モデルを作成することも考えられるが、回帰モデルの作成にあたっては、入力パラメータ数が 1 増えるだけでモデルが複雑化することが予想される。上記の評価の流れとしては、軽水炉燃料の燃焼計算、燃焼後の冷却計算、高速炉の炉心計算が想定されるが、このうち燃焼後の冷却計算については回帰モデルを採用するまでもなく、比較的容易に行うことが出来るので、図 7 に示すように、軽水炉燃料の燃焼計算と高速炉の炉心計算のそれぞれに対して回帰モデルを作成することとし、冷却計算については解析的に行う方針を採った。この場合は、前段のモデルでは 16 の TRU の組成比が出力パラメータとして得られ、それらが後段のモデルの入力パラメータとなる。

高速炉炉心計算の回帰モデルでは、16 の TRU の組成比が入力パラメータ、21 の重核種のインベントリが出力パラメータとなり、入出力パラメータの個数が大きい。そこで、第 50 回炉物理夏期セミナーテキストの名大・山本先生の解説に述べられている方法でデータの次元削減を行った。以下、簡単に説明する。

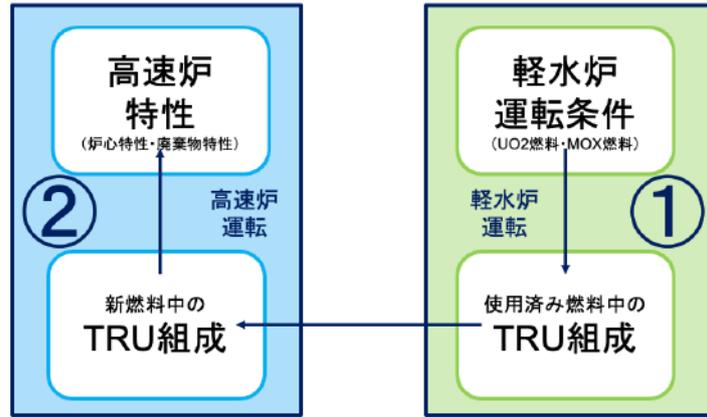


図 7: 軽水炉の燃料仕様、運転条件から高速炉の炉心特性、廃棄物特性を推定するモデルの概要

総数がそれぞれ I 、 J の入出力パラメータを考え、 i 番目の入力パラメータを x_i 、 j 番目の出力パラメータを y_j とする。次元削減を行わない場合、 y_j の回帰モデルは以下のように書ける。

$$y_j = \sum_{k=1}^K a_{j,k} f_k(x_1, x_2, \dots, x_I) \quad (61)$$

ここで、 K は回帰関数の個数を示す。回帰モデルの係数 $a_{j,k}$ はこれまでに述べた方法で容易に得られるが、回帰関数が各々の入力パラメータに依存する多項式関数の積で記述されるとするならば、 K は I が大きくなるに従い爆発的に大きくなる。係数 $a_{j,k}$ は $J \times K$ だけ存在するので、 I と J が大きい場合にはこの係数の個数もそれに従って大きくなる。

そこで、入力パラメータ、出力パラメータのいずれに対しても次元削減を行う。入出力パラメータのセットとして L 個得られているものとし、その l 番目のものを x_l 、 y_l とする。そして、それぞれが構成する snapshot 行列を以下のように特異値分解し、低ランク近似を施す。

$$(x_1 \ x_2 \ \dots \ x_L) = U_x \begin{pmatrix} D_x & 0 \\ 0 & 0 \end{pmatrix} V_x^T \approx (u_{x,1} \ u_{x,2} \ \dots \ u_{x,r_x}) (c_{x,1} \ c_{x,2} \ \dots \ c_{x,L}), \quad (62)$$

$$(y_1 \ y_2 \ \dots \ y_L) = U_y \begin{pmatrix} D_y & 0 \\ 0 & 0 \end{pmatrix} V_y^T \approx (u_{y,1} \ u_{y,2} \ \dots \ u_{y,r_y}) (c_{y,1} \ c_{y,2} \ \dots \ c_{y,L}) \quad (63)$$

ここで、 $c_{x,l}$ 、 $c_{y,l}$ は入出力パラメータについて、 l 番目のデータの POD 基底ベクトルの展開係数を示し、 r_x 、 r_y は POD 基底の次元数を示す。このようにしたときの回帰モデルは、 r_x 個のパラメータ c_x から r_y 個のパラメータ c_y を予測するものとして作成される³。

回帰モデルによる予測の手順は以下となる。

1. 予測を行う入力パラメータ x が与えられたとする。
2. $U_x^T x$ より c_x を求める。
3. c_x から回帰式により c_y を推定する。
4. $U_y c_y$ より y の予測値を得る。

多項式を回帰関数として作成した回帰モデルについて、入出力パラメータの総数と削減後の数、回帰関数における最大次数を表 1 に示す。軽水炉の取り出し燃料中の 16 の TRU の組成比については、6 から 8 程度まで次元削減が行える一方、高速炉の取り出し燃料中の 21 の重核種インベントリについては次元削減の程度が小さいという結果が得られた。また、回帰関数の最大次数としては、最大で 5 まで考慮する必要があった。

³特異値 σ と v から構成される c ではなく、 v_x から v_y を予測する回帰モデルを作成することも出来るが、等価と考えられる。

表 1: 回帰モデルにおける入出力パラメータ数の削減の程度など

Model	Number of input parameters	Number of output parameters	Maximum order of regression Eq.
LWR-UO2 → TRU in SF	5	16 → 6	4
LWR-MOX → TRU in SF	7	16 → 8	5
TRU in fuel → FR core parameters	16 → 8	4 → 3	2
TRU in fuel → FR SF inventories	16 → 8	21 → 17	3

4 核種生成・変換計算への POD-GP の適用

4.1 背景

核種の生成・変換計算では、核種数に応じた大きさを持つ行列を扱う必要があるが、核分裂生成物の数は 1,000 を超える。また、実際の原子炉においては、装荷される燃料棒毎、さらには軸方向位置毎に燃料中の核種生成・変換計算を行う必要があり、考慮しなければならない媒質数は膨大となる。このように、膨大な数の媒質について 1,000 を超える核種の数密度を数値計算により追跡することは現実的とは言えない。そのため、(100 程度の) 重要な核種のみを考慮する低次元の核種生成・変換チェーンモデルを用いるのが一般的であるが、この方法ではチェーンモデルに含まれない核種の数密度を得ることは出来ない。

低次元のチェーンモデルを用いる方法と異なるアプローチとして、当研究室ではこれまでに固有直展開 (POD) とガラーキン射影 (Galerkin projection) を用いた、核種生成・変換計算に関する ROM の検討を進めてきた。これは、全ての核種を考慮した計算結果に基づいて、核種数密度ベクトルの正規直交基底 (POD 基底) を求め、その基底が張る低次元空間に問題を変換することで、解くべき問題の縮約を図る、というものである。ただし、この方法 (以下、POD-GP と呼称する) では、縮約後の問題において、燃焼行列に対応する (核種遷移) 行列が正の固有値実数成分をもち、安定に解を得られないことがあることがこれまでの検討で分かっている。本稿では、核種生成・変換計算への POD-GP の適用例とその問題点、さらにはその対処法について述べる。

4.2 理論の概要

I 個の核種の数密度の時間変動を考える。時刻 t における数密度ベクトルを $\mathbf{n}(t)$ としたとき、Bateman 方程式は以下のように行列形式で記述できる。

$$\frac{d\mathbf{n}(t)}{dt} = \mathbf{A}\mathbf{n}(t) \quad (64)$$

この式を解いて、異なる L 個の時間点 t_l で数密度ベクトル $\mathbf{n}(t_l)$ が得られたものとする。この L 個のベクトルが構成する行列 $\mathbf{N} = (\mathbf{n}(t_1) \ \mathbf{n}(t_2) \ \cdots \ \mathbf{n}(t_L))$ に対して特異値分解を施すことにより、ベクトルのセット $\mathcal{Q} = \{\mathbf{n}(t_1), \mathbf{n}(t_2), \dots, \mathbf{n}(t_L)\}$ が張る空間の正規直交基底ベクトルのセット \mathcal{S} を得ることが出来る。 \mathcal{Q} に I 個の線形独立なベクトルが含まれていれば、 \mathcal{S} が張る空間の次元数は I となる。また、 \mathcal{S} に対して特異値が小さい成分を無視することによって、その次元を小さくすることが出来る。 \mathcal{S} が張る空間の次元数を J とし ($J \leq I$)、 $\mathcal{S} = \{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_J\}$ と記述したとき、サイズが I のベクトル $\mathbf{n}(t)$ を \mathcal{S} のベクトルを用いて以下のように近似する。

$$\mathbf{n}(t) \approx \sum_{j=1}^J c_j(t) \mathbf{u}_j \quad (65)$$

この式から分かるように、任意時刻 t の数密度ベクトル $\mathbf{n}(t)$ を \mathbf{u}_j の線形和で近似していることから、各々の \mathbf{u}_j に乗られている $c_j(t)$ は、「数密度ベクトルを \mathcal{S} で近似したときの成分 \mathbf{u}_j に対応する係数 (展開係数)」と考えることが出来る。

式 (65) を式 (64) に代入すると以下を得る。

$$\sum_{j=1}^J \frac{dc_j(t)}{dt} \mathbf{u}_j - \sum_{j=1}^J c_j(t) \mathbf{A} \mathbf{u}_j = \boldsymbol{\epsilon}(t) \quad (66)$$

ここで、左辺がゼロとならない、すなわち

$$\sum_{j=1}^J \frac{dc_j(t)}{dt} \mathbf{u}_j = \sum_{j=1}^J c_j(t) \mathbf{A} \mathbf{u}_j \quad (67)$$

とならないのは、式 (65) が近似であることが理由である。そのため、式 (67) の両辺の差を、式 (66) ではサイズが I の誤差ベクトル $\boldsymbol{\epsilon}(t)$ としている。 $J = I$ であるときは式 (65) は厳密となるため、式 (67) は成り立つ。

ここまでの議論では、 $c_j(t)$ は任意である。この $c_j(t)$ を一意に決めるために、誤差ベクトル $\boldsymbol{\epsilon}(t)$ が S を構成する全てのベクトルに直交するという条件、すなわち、

$$\mathbf{u}_j^T \boldsymbol{\epsilon}(t) = 0 \quad (68)$$

を全ての j に対して課すものとする（これを Galerkin projection と呼ぶ）。このことは、式 (66) の両辺の左から $\mathbf{u}_{j'}$ の転置を乗じたものをゼロとすることと同義である。すると、以下が得られる。

$$\sum_{j=1}^J \frac{dc_j(t)}{dt} \mathbf{u}_{j'}^T \mathbf{u}_j - \sum_{j=1}^J c_j(t) \mathbf{u}_{j'}^T \mathbf{A} \mathbf{u}_j = 0 \quad (69)$$

さらに、 S が正規直交基底であることから以下の式が得られる。

$$\frac{dc_{j'}(t)}{dt} = \sum_{j=1}^J c_j(t) \mathbf{u}_{j'}^T \mathbf{A} \mathbf{u}_j \quad (70)$$

これは全ての j' に対して成り立つので、まとめて以下のように書ける。

$$\frac{d}{dt} \begin{pmatrix} c_1(t) \\ c_2(t) \\ \vdots \\ c_J(t) \end{pmatrix} = \begin{pmatrix} \mathbf{u}_1^T \mathbf{A} \mathbf{u}_1 & \mathbf{u}_1^T \mathbf{A} \mathbf{u}_2 & \cdots & \mathbf{u}_1^T \mathbf{A} \mathbf{u}_J \\ \mathbf{u}_2^T \mathbf{A} \mathbf{u}_1 & \mathbf{u}_2^T \mathbf{A} \mathbf{u}_2 & \cdots & \mathbf{u}_2^T \mathbf{A} \mathbf{u}_J \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{u}_J^T \mathbf{A} \mathbf{u}_1 & \mathbf{u}_J^T \mathbf{A} \mathbf{u}_2 & \cdots & \mathbf{u}_J^T \mathbf{A} \mathbf{u}_J \end{pmatrix} \begin{pmatrix} c_1(t) \\ c_2(t) \\ \vdots \\ c_J(t) \end{pmatrix} \quad (71)$$

さらに、 $\mathbf{U} = (\mathbf{u}_1 \ \mathbf{u}_2 \ \cdots \ \mathbf{u}_J)$ とすると

$$\mathbf{U}^T \mathbf{A} \mathbf{U} = \begin{pmatrix} \mathbf{u}_1^T \mathbf{A} \mathbf{u}_1 & \mathbf{u}_1^T \mathbf{A} \mathbf{u}_2 & \cdots & \mathbf{u}_1^T \mathbf{A} \mathbf{u}_J \\ \mathbf{u}_2^T \mathbf{A} \mathbf{u}_1 & \mathbf{u}_2^T \mathbf{A} \mathbf{u}_2 & \cdots & \mathbf{u}_2^T \mathbf{A} \mathbf{u}_J \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{u}_J^T \mathbf{A} \mathbf{u}_1 & \mathbf{u}_J^T \mathbf{A} \mathbf{u}_2 & \cdots & \mathbf{u}_J^T \mathbf{A} \mathbf{u}_J \end{pmatrix} \quad (72)$$

であるので、 $\mathbf{c}(t) = (c_1(t) \ c_2(t) \ \cdots \ c_J(t))^T$ 、 $\hat{\mathbf{A}} = \mathbf{U}^T \mathbf{A} \mathbf{U}$ とすると、以下の式を得る。

$$\frac{d\mathbf{c}(t)}{dt} = \hat{\mathbf{A}} \mathbf{c}(t) \quad (73)$$

これより、サイズが I の数密度ベクトルに対する微分方程式が、サイズが J の展開係数ベクトルに対する微分方程式に変換されていることが分かる。 $J < I$ とした場合には、解くべき方程式の本数が低減されるため、計算負荷を低減出来る。

4.3 瞬時核分裂後の FP 数密度の計算への POD-GP の適用

JENDL/FPD-2011、FPY-2011 にデータが与えられている 1,400 の FP を対象として、U-235 が熱中性子によって瞬時に核分裂した後の数密度変動に対して、POD を適用した。はじめに、1,400 よりも十分に大きい数の時間点での FP の数密度ベクトルを計算し、それに基づいて構築した数密度行列に対して特異値分解を適用し、POD 基底を計算した。その後、この数密度行列がどの程度の次元数の POD 基底で再現可能であるかを評価した。得られた結果を図 8 左に示すが、100 を超える程度の次元数の POD 基底により、良好な精度で FP 数密度を予測することが分かる。そこで次に、 $\mathbf{n}(t)$ を POD 基底が張る部分空間に射影した $\mathbf{c}(t)$ に対する式 (73) を解いて得た結果の再現誤差を図 8 右に示すが、100 次元程度までは POD 基底の次元数の増加とともに誤差が低減する傾向が見られるものの安定した解が得られていないこと、100 次元程度を超えたあたりから解が発散すること、1,100 次元程度を超えたあたりから 100 次元程度と同様の結果が得られることが分かる。

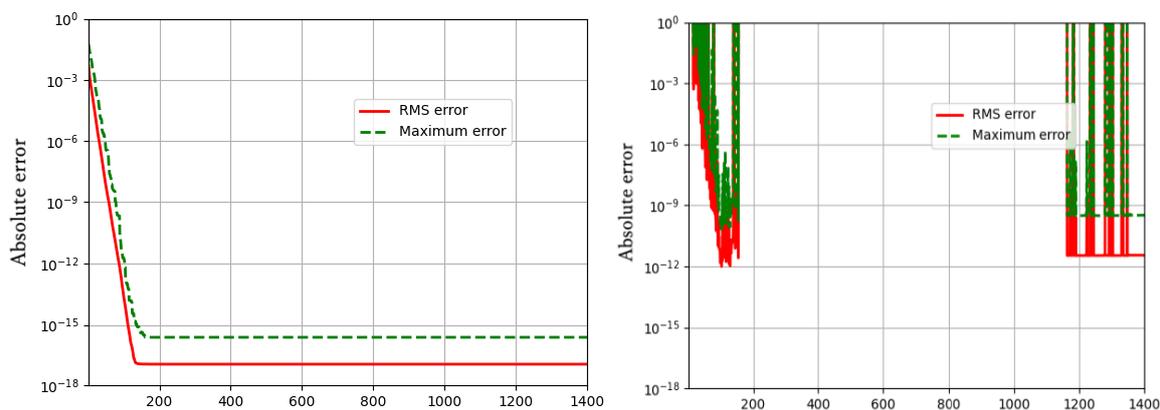


図 8: 瞬時核分裂後の FP の数密度の POD による予測値の誤差 (横軸は POD 基底の次元数を示す)

Bateman 方程式 (64) における係数行列 A の固有値は複素平面上で負の実軸付近に存在する (固有値の実部は負で、僅かな虚数成分を持つ場合がある) ことが分かっている。一方、縮約後の方程式 (73) における係数行列 \hat{A} については、その限りではない。POD 基底が張る空間の次元数 J がオリジナルの空間の次元数 I と一致する場合、 $\hat{A} = U^T A U$ における行列 U はユニタリ行列となることから、 $U^T = U^{-1}$ である。従って、 A と \hat{A} は相似な行列となり、固有値が一致することが保証される。一方、 $J < I$ の場合には、 U はユニタリ行列とはならず、 A と \hat{A} の固有値が一致することは保証されない。文献 (Roberts, *et al.*, *Nucl. Sci. Eng.*, **193**, pp. 1371) の記述に従うならば、「the leading J eigenvalues of \hat{A} are inferred from the eigenvalue of A 」と解釈される。 \hat{A} の固有値に実部が正のものが含まれる場合には、それは $t \rightarrow \infty$ で解が発散することを意味し、適切な解が得られないことが予想される。これが、Bateman 方程式に POD-GP を適用した際に安定に解を得ることができない主な理由と考えられる。

4.4 15 核種チェーン問題での検討

4.4.1 問題設定

ここでは 15 核種からなる簡易なチェーンを用いて行った検討結果について紹介する。チェーンの構成を図 9 に、チェーンを構成する核種の崩壊定数と初期数密度を表 2 に、それぞれ示す。

計算では、時間をステップで分割し、ステップが進む毎にステップの幅が 2 倍になるものとした。また、各時間ステップはサブステップに等分割した。本検討では、最初のステップの時間幅を 0.01 秒とし、サブステップ数を 40 とした。なお、数密度に対する計算誤差は全て絶対値で評価した。計算は全て Python で行い、オリジナルモデル、ROM の燃焼計算はいずれも α モード分解法により行った。

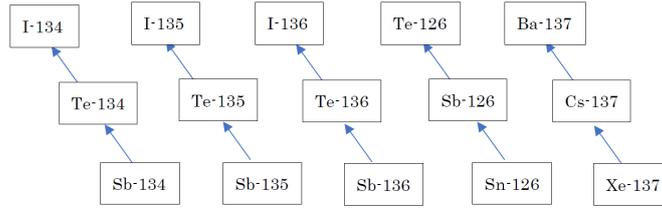


図 9: 15 核種チェーン

表 2: 15 核種チェーンにおける定数と初期値

核種 No.	核種名	崩壊定数 [1/s]	初期数密度
0	Sb-134	8.88462E-01	2.29585E-03
1	Te-134	2.76096E-04	6.73523E-02
2	I-134	0	8.58291E-03
3	Sb-135	4.05263E-01	1.46106E-03
4	Te-135	3.64737E-02	3.22482E-02
5	I-135	0	2.93524E-02
6	Sb-136	8.45121E-01	1.15358E-04
7	Te-136	3.96000E-02	1.33578E-02
8	I-136	0	7.70678E-03
9	Sn-126	2.19582E-13	5.60411E-04
10	Sb-126	6.43454E-07	7.71267E-06
11	Te-126	0	9.13459E-06
12	Xe-137	3.02488E-03	6.14115E-02
13	Cs-137	7.30319E-10	6.01601E-04
14	Ba-137	0	1.87798E-06

α モード分解法について簡単に説明する。式 (64) の行列 \mathbf{A} に対して $\mathbf{A} = \mathbf{XDX}^{-1}$ のように固有値分解を施し、これを原式に代入すると以下を得る。

$$\frac{d\mathbf{n}(t)}{dt} = \mathbf{XDX}^{-1}\mathbf{n}(t) \quad (74)$$

この両辺に左から \mathbf{X}^{-1} を乗じると以下を得る。

$$\frac{d\mathbf{X}^{-1}\mathbf{n}(t)}{dt} = \mathbf{DX}^{-1}\mathbf{n}(t) \quad (75)$$

$\hat{\mathbf{n}}(t) = \mathbf{X}^{-1}\mathbf{n}(t)$ とすると、以下が得られる。

$$\frac{d\hat{\mathbf{n}}(t)}{dt} = \mathbf{D}\hat{\mathbf{n}}(t) \quad (76)$$

\mathbf{n} のサイズ I と同数の固有値が存在するとした場合、 $\mathbf{D} = \text{diag}(\alpha_1, \alpha_2, \dots, \alpha_I)$ と書けるので、 $\hat{n}_i(t) = \hat{n}_i(0) \exp(\alpha_i t)$ が容易に得られる。これらの個々を本稿では α モードと呼称している。 $\mathbf{n}(t) = \mathbf{X}\hat{\mathbf{n}}(t)$ より、 $\mathbf{n}(t)$ は複数の α モードの線形和で記述されることが分かる。

ステップ数を 15 とした場合（最終時間点は 300 秒程度）のオリジナルモデルでの結果を図 10 に示す。不安定核種の数密度はゼロに漸近するが、数値的に計算可能な範囲（ 10^{-200} より大きい）であることが分かる。 α モード固有値と、各モード成分の初期値を表 3 に示す。安定核の個数が 5 であることから固有値がゼロのモードが 5 つ存在すること、非ゼロの固有値は全て負の実数であることが分かる。また、数密度の各 α モード成分の時間変化を図 11 に示す。それぞれの α モードは単一の指数関数に従って変動するため、

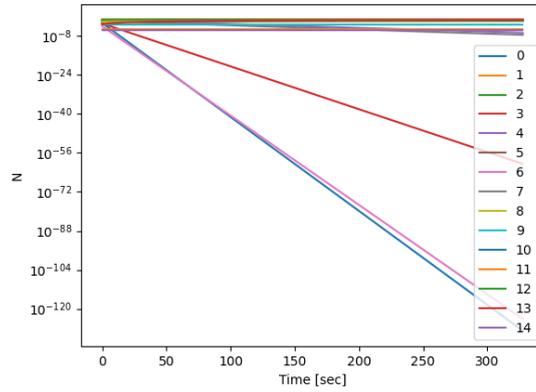


図 10: オリジナルモデルでの数密度計算結果 (総ステップ数 15)

数密度の各成分も同様の振る舞いを示す。

表 3: α モード固有値と各モードの初期値 (計算結果の出力をそのまま貼り付けたものである)

Index	Re(α)	Im(α)	Initial value
1	0.0	0.0	0.07823106
2	-0.000276096	0.0	0.09849836761135064
3	-0.888462	0.0	0.0032473269683862536
4	0.0	0.0	0.06306165999999999
5	-0.0364737	0.0	0.04787644733566365
6	-0.405263	0.0	0.0021756381782958357
7	0.0	0.0	0.021179938
8	-0.0396	0.0	0.019061942894703317
9	-0.845121	0.0	0.00016729515351185373
10	0.0	0.0	0.00057725826
11	-6.43454e-07	0.0	1.0907092057362963e-05
12	-2.19582e-13	0.0	0.0007925409719324758
13	0.0	0.0	0.06201497898
14	-7.30319e-10	0.0	0.08769978944760394
15	-0.00302488	0.0	0.08684898666997291

15 の α モードのうち、いずれかの 1 つを無視した場合の数密度の最大誤差と RMS 誤差を図 12 に示す。初期値が小さいモードであるほど無視することによる誤差が小さく、この例では 11 番目のモードを無視した場合の誤差が最小となり 10^{-6} 程度であった。

α モード固有値と各モード成分の初期値の分布、及び図 12 の結果から、この問題に対して POD-GP を適用するにあたって、以下のことが予想される。

- 固有値がゼロのモードが 5 つ、絶対値が極めて小さいモードが 2 つ、それぞれ含まれていることから、これらは時間に対して一定となる成分と考えてよい。おそらくそれが POD 基底の第一成分となるであろう。このことは 15 モードのうち 6 モードは除去できる可能性を示している。
- 数密度誤差として 10^{-6} 程度が許容されるとするならば、 α モードの 11 番目の成分を無視することが出来る。これにより 15 モードのうちさらに 1 モードは除去可能となるものと考えられる。

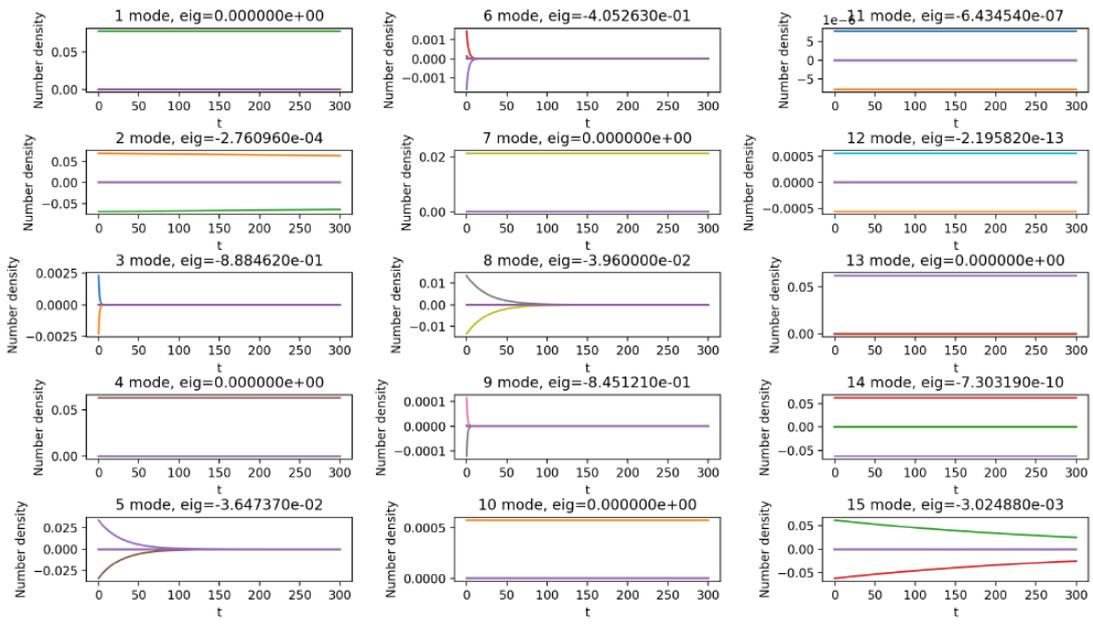


図 11: 核種数密度における α モード毎の寄与

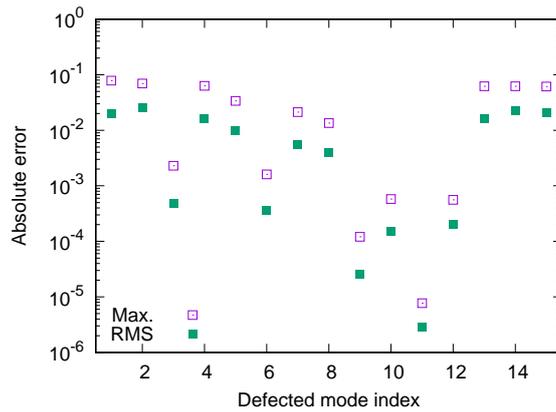


図 12: 1 つのモードを除去したときの α モード展開による数密度の誤差

以上より、POD-GP により最低でも 8 次元程度まで縮約が可能となるものと予想される。

4.4.2 オリジナルモデルと同一次元の POD 基底が張る空間に射影したときの数密度の再現精度

ステップ数を 15 とした場合について、オリジナルモデルと、オリジナルモデルと同一の次元の POD 基底が張る空間に射影した場合 (4.2 節の説明において $J = I$ とした場合) の結果を図 13 に示す。両者は理論的には等価であるが、特異値分解における数値誤差の影響で、後者では 10^{-15} 未満の小さい数密度を適切に評価できないことが分かる。また、数密度の各 POD モード成分の時間変化を図 14 に示す⁴。第一基底成分が平均的な振る舞いを示し、第二、第三となるに従い、変動の周期が短くなっていくことが分かる。両計算ケースの α 固有値が一致することが理論的に保証されるが、POD モード成分別の数密度に見られる振動的な振る舞いは、複数の指数関数の重ね合わせで記述されるものではないため、数値誤差の範囲と言えるであろう。

⁴なお、ここで用いた POD 基底は、一定のタイムステップ 0.1 秒での snapshot から作成したものである。

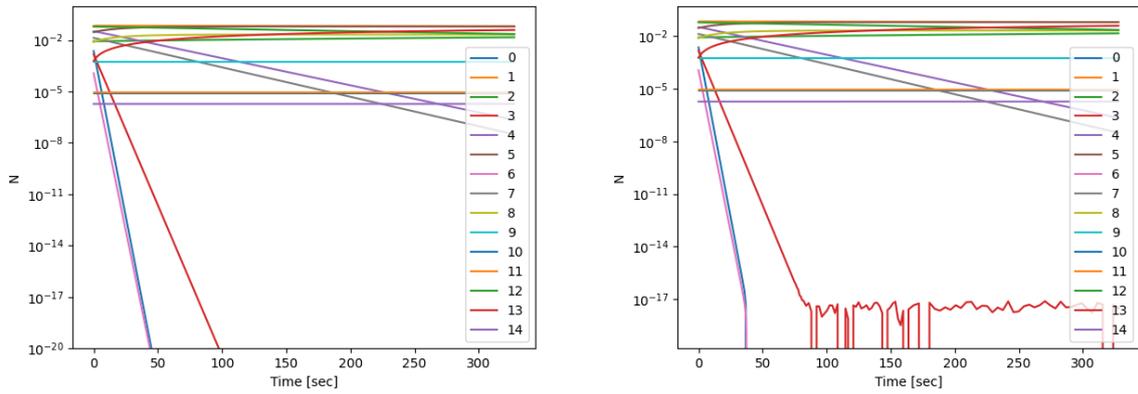


図 13: オリジナルモデルによる数密度 (左) と同一次元の POD 基底空間に射影して得られた数密度 (右) (総ステップ数 15)

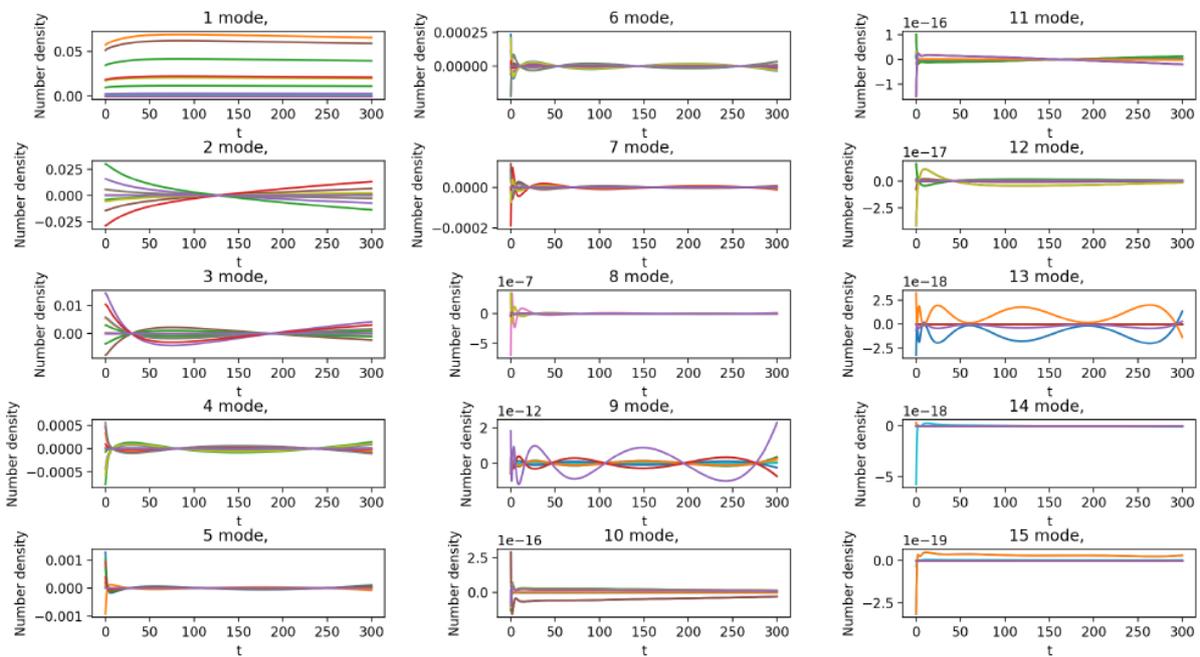


図 14: POD モード毎の数密度成分

ステップ数を 25 とした場合 (最終時間点は 3.88 日程度) について同様の計算を行った結果を図 15 に示す。不安定核種の数密度は数値計算で考慮できる範囲よりも小さい値となることが分かる。一方、POD 基底空間に射影したときの結果は、上述の問題と同様、 10^{-15} 未満では意味のあるものとはならなかった。

4.4.3 POD-GP による計算

数密度の snapshot 行列から作成した低次元の POD 基底が張る空間に数密度を射影した問題を解き、数密度を計算した。総ステップ数を 15、25 としたときの最大誤差と RMS 誤差を図 16 に示す。総ステップ数を 15 とした場合、次元数の増加に伴い誤差は低下するが、次元数が 13 と 15 のときに特異な振る舞いを示すことが分かる。これらに共通するのは ROM における遷移行列 \hat{A} の固有値に複素共役のものが含まれる点であった (なお、次元数が 2、3 のときも複素共役の固有値が含まれていた)。また、総ステップ数を 25

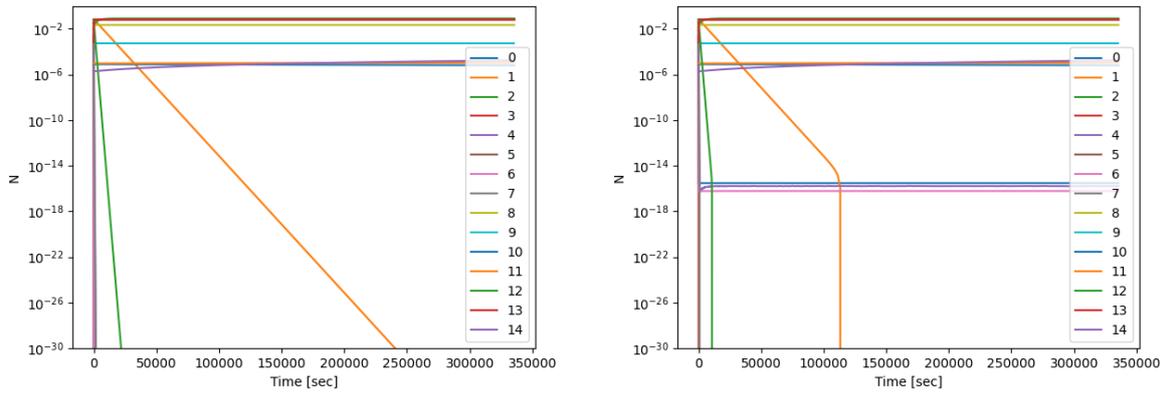


図 15: オリジナルモデルによる数密度 (左) と同一次元の POD 基底空間に射影して得られた数密度 (右) (総ステップ数 25)

とした場合、次元数が 4 以下のときには、 $\hat{\mathbf{A}}$ が 10^{-5} を超える正の実数の固有値を持つため、解が発散傾向となった。次元数が 6 のときにも正の実数の固有値が現れたが、値が 6.8×10^{-7} 程度であったため、着目した時間範囲では解は発散傾向を示さなかった。次元数の増加に伴い誤差は低下するが、次元数が 8、11、12 のときに特異な振る舞いを示した。次元数が 8 のときには固有値として 10^{-5} オーダーのものが現れたことが原因と考えられる。また、次元数が 11、12 のときには、ステップ数を 15 とした場合と同様に、複素共役の固有値が現れた (実数部は 10^{-17} のオーダー)。複素共役の固有値は次元数が 14、15 のときにも現れたが、実数部のオーダーが 10^{-28} であることにより、誤差の拡大が抑制された可能性がある。

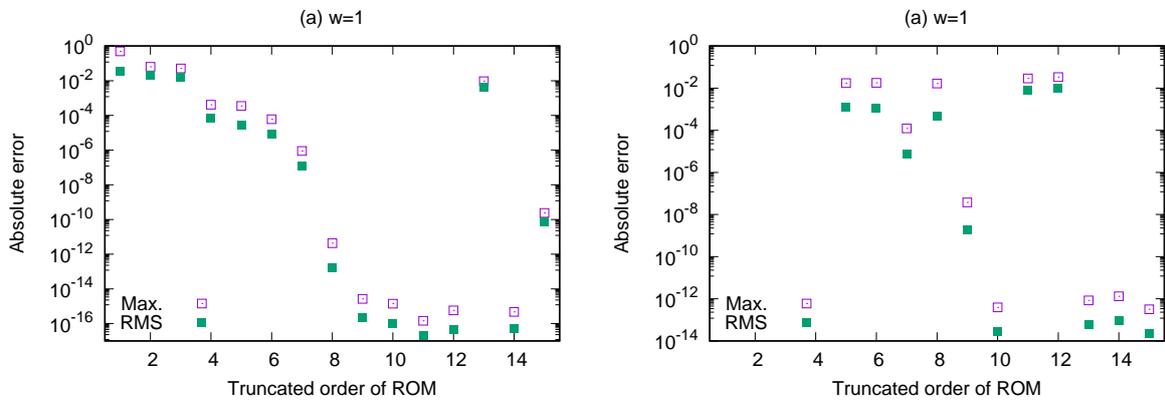


図 16: POD-GP による数密度誤差 (左: 総ステップ数 15、右: 総ステップ数 25)