

# シンプレックス法を用いた非負のモデルパラメータを持つ 並列処理時間モデルの予測力向上

折居 茂夫<sup>1,a)</sup> 山本 義郎<sup>1</sup>

受付日 2014年8月27日, 採録日 2015年3月4日

**概要:** 並列処理時間の回帰モデルの予測力向上を図る計算法を提案する. 与えられた線形基底関数モデルのモデルパラメータに非負制約を課し, 観測データの揺らぎがモデル化されることを抑制して予測力を向上する. 非負制約を課したモデルパラメータは, 連立残差不等式を有理数演算のシンプレックス法で解いて決定する. 観測データの揺らぎをモデル化していた基底関数のあるものは, 非負制約を課すことにより残差を小さくすることができなくなり, シンプレックス法はそのモデルパラメータを零値とする. このようにして, 揺らぎのモデル化により生じた過剰適合の一部を自動的に抑制することができる. モデル化の事例において, 非負制約によりモデルの予測力が向上することを確認した. 問題の規模とプロセッサ数を変数とする 2, 3 変数の並列処理時間のモデル化の事例では, 提案計算法でのみで予測モデルが構築できることを確認した.

**キーワード:** 並列処理, 性能予測, 予測モデル, 過剰適合, 変数選択

## Improving Predictive Power of Non-Negative Parameter Models of Parallel Processing Timing by Using Simplex Method

SHIGEO ORII<sup>1,a)</sup> YAMAMOTO YOSHIRO<sup>1</sup>

Received: August 27, 2014, Accepted: March 4, 2015

**Abstract:** We propose a novel method improving predictive power of regression models for parallel processing time. Non-negative constraint for model parameters of a given model function can improve model predictive power because the constraint suppresses modeling of fluctuation of observed data. The non-negative model parameters are obtained as a solution of simultaneous inequalities for residuals solved by the simplex method with rational arithmetic. Some bases-functions that model the fluctuation cannot reduce residual by adding the non-negative constraint and the simplex method outputs zero value for their model parameters. In this way, some overfitting which occurs due to the modeling of fluctuation can be suppressed automatically. In examples for modeling, the non-negative constraint effects to improve predictive power on models. In examples of modeling of parallel processing time having two or three variables consisting of problem size and number of processors, only the proposed method is effective to make a predictive model.

**Keywords:** parallel processing, performance prediction, prediction model, overfitting, variable selection

### 1. はじめに

並列処理は, 処理を複数のプロセッサに分配して処理時間の短縮を図る. 一方分配できない処理や分配時に発生する新たな処理, すなわち並列オーバーヘッドが時間短縮を妨

げる. 並列処理の性能はこの短縮効果とオーバーヘッドの結果として得られるため, 処理時間をモデル化して性能予測することが行われてきた. プロセッサ数の増加により処理時間が短縮される並列化部と短縮されないか増加する並列オーバーヘッドをモデル化して並列処理時間モデルを構築する方法が提案され [1], [2], [3], [5], [6], [8], [12], プログラムと計算機を別々にモデル化しそれらを統合することによ

<sup>1</sup> 東海大学  
Tokai University, Hiratsuka, Kanagawa 259-1292, Japan  
<sup>a)</sup> GDE00740@nifty.com

りモデル化する方法が提案された [4], [7], [9]. 性能が異なるプロセッサによる並列処理のモデル化方法も提案された [14]. 実際の数値シミュレーションという複雑な並列処理の時間をモデル化する方法 [10], [13], [16], 複数の変数を扱うモデル化方法 [11], [15], モデル式を動的に補完する方法 [17] 等が提案されている. これらの提案は主に, 並列処理の手続きとその処理時間の関係をより詳細に記述することを通してモデルの精度を向上するという観点で行われている. これに対し本論文では, 基底関数の候補を想定したモデル式のモデルパラメータの決定方法を工夫してモデルの予測力を向上する方法について提案する. 並列処理時間モデルは当初プロセッサ数を変数とした 1 変数モデルであったが, Gustafson [18] が問題のプロセッサ数の増加とともに問題を大きくすると並列処理の性能が維持できることを指摘して以来プロセッサ数と問題の大きさを変数とするモデルが重要となり, 現在ではこの 2 変数モデルをベースにしたモデル化が一般的である. さらに実際の並列処理のモデル化では使用するモデルパラメータ数が増える. これを最小二乗法で決定すると, 外挿領域の予測が大きく外れることをしばしば経験する. この原因としては, 用いた基底関数がモデル化に使った観測データの振舞いをうまく記述できない, 使用した観測データが外挿領域で優勢となる振舞いの情報をあまり含んでいない, 観測データ中のノイズがモデル化に不要な基底関数によりモデル化される等がある. 特に予測に不要な基底関数の中に冪乗等の急激に増大するものがあると, データを外挿した領域での予測値は実際とは大きく異なることになる. ゆえに, モデルによる予測ではこのような事態を回避する工夫が重要な課題であると考えられる.

そこで我々は予測に不要なノイズのモデル化を抑制してモデルパラメータを推定する方法を研究してきた. 本論文では非負制約を課したモデルパラメータを持つ残差不等式を有理数演算のシンプレックス法で解いて, ノイズの主要因の 1 つである観測の揺らぎのモデル化を抑制してモデルパラメータを推定する計算法を提案する. 解法にシンプレックス法を用いると, モデル化に必要な基底関数のモデルパラメータは零となりその基底変数が用いられないため, 予測を阻害する過剰適合を自動的に抑制しながら回帰モデルの予測力向上を図ることができる. 2 章では本論文で扱う課題について述べる. 3 章で課題を解決する基本的アイデアを示す. 4 章でこのアイデアをシンプレックス法で解く方法を提案する. 5 章では, 1 変数モデルに関するシミュレーションにより, 提案手法の基本的アイデアの検証と比較に用いた手法の妥当性を確認する. 次にモデル変数の数が 2 つと 3 つの場合についての事例を用いて, 提案手法により並列処理時間モデルの予測精度が向上することを確認する. 6 章で提案方法の適用性を考察し, 7 章でまとめを行う.

## 2. 予測モデル構築における課題

最小二乗法 (LSM) で予測モデルを作る際の課題を, 正規方程式を使って説明する. 目的変数  $y$  を説明変数  $x$  の  $M$  個の基底関数  $f_k(x)$  で説明する回帰モデル  $\hat{y}(x) = a_0 + \sum_{k=1}^M a_k f_k(x)$  ( $k = 1, \dots, M$ ) を考える. 係数  $a_k$  はモデルパラメータである. このパラメータを  $(x_i, y_i)$  ( $i = 1, \dots, N$ ) の観測データで推定する. LSM では式 (1) の残差平方和  $E (= \sum \{y_i - \hat{y}(x_i)\}^2)$  を最小にする  $a_k$  を推定する. 式を簡単にするため  $f_0(x_i) = 1$  とすると  $E$  は式 (1) となる.

$$E = \sum_{i=1}^N \left\{ y_i - \sum_{k=0}^M a_k f_k(x_i) \right\}^2 \quad (1)$$

$E$  を最小にするため, 両辺をモデルパラメータ  $a_k$  で偏微分して  $\partial E / \partial a_k = 0$  ( $k = 0, \dots, M$ ) とすると式 (2) を得る.

$$\begin{aligned} \frac{\partial E}{\partial a_k} &= -2 \sum_{i=1}^N \left[ f_k(x_i) \left\{ y_i - \sum_{k=0}^M a_k f_k(x_i) \right\} \right] \\ &= \sum_{i=1}^N [f_k(x_i) y_i - f_k(x_i) (a_0 f_0(x_i) + a_1 f_1(x_i) + \dots \\ &\quad + a_M f_M(x_i))] = 0 \\ a_0 \sum_{i=1}^N f_k(x_i) + a_1 \sum_{i=1}^N f_k(x_i) f_1(x_i) + \dots \\ &\quad + a_M \sum_{i=1}^N f_k(x_i) f_M(x_i) = \sum_{i=1}^N f_k(x_i) y_i \end{aligned} \quad (2)$$

ここで式を簡潔にするため, 観測データ  $y_i$  を  $\mathbf{y} = [y_1, y_2, \dots, y_N]^T$ , モデルパラメータ  $\mathbf{a} = [a_0, a_1, \dots, a_M]^T$ ,  $\mathbf{f} = \begin{bmatrix} f_1(x_1) & f_1(x_2) & \dots & f_1(x_N) \\ \vdots & \vdots & & \vdots \\ f_M(x_1) & f_M(x_2) & \dots & f_M(x_N) \end{bmatrix}$  とベクトルと行列で表記すと式 (2) は式 (3) と表現でき, モデルパラメータ  $\mathbf{a}$  はこの正規方程式の解として決定することができる.

$$\mathbf{f} \mathbf{f}^T \mathbf{a} = \mathbf{f} \mathbf{y} \quad (3)$$

ここでノイズのモデル化を具体的に考察する. 回帰モデル  $f_k(x)$  において  $k = 1$  の基底関数  $f_1$  が過剰適合項の場合を考え,  $\mathbf{a}' = [a'_0, 0, a'_2, \dots, a'_M]^T$ , 観測データが持つ測定誤差等のノイズを表す定数を  $\mathbf{d} = [d_1, d_2, \dots, d_N]^T$  とし, このような観測データを式 (4) で生成する.

$$\mathbf{y} = \mathbf{f}^T \mathbf{a}' + \mathbf{d} \quad (4)$$

式 (4) を式 (3) に代入すると,  $\mathbf{f} \mathbf{f}^T \mathbf{a} = \mathbf{f} \mathbf{f}^T \mathbf{a}' + \mathbf{f} \mathbf{d}$  となり式 (5) を得る.

$$\mathbf{f} \mathbf{f}^T (\mathbf{a} - \mathbf{a}') = \mathbf{f} \mathbf{d} \quad (5)$$

式 (5) は, 観測データに  $d_i \neq 0$  が 1 点でもあると右辺が非零となり  $\mathbf{a} - \mathbf{a}' \neq \mathbf{0}$  すなわち  $a_1 \neq 0$  となることを示す.

このようにモデル化に必要な基底関数があると、観測データのノイズによりそのモデルパラメータが値を持つようになる。これが過剰適合である。観測データにはつねにノイズが存在するため、 $f_1$ のような必要ない基底関数がモデル中に含まれていれば、過剰適合はつねに発生していると考えられる。もしこの  $f_1$  が  $x$  の冪乗等の急激に増加する関数であれば、観測データ  $\mathbf{y}$  の外挿領域においてその項の影響が大きくなり、結果モデルの予測力が低下する。したがってモデルパラメータの決定時に過剰適合となっている基底関数を見つけて削除することが、予測モデル構築における重要な課題の1つとなる。

### 3. 基本的アイデア

2章で述べたノイズ  $\mathbf{d}$  を観測における「揺らぎ」に限定し、揺らぎがモデル化されて生じる過剰適合を抑制する方法の基本的アイデアを述べる。

図 1 の  $\times$  印は  $\mathbf{d}$  を「揺らぎ」と想定したシミュレーションデータ  $\mathbf{y}$  を観測データとしたときの一例で、真の値  $y^0 = 0.1$ ,  $d_i$  ( $i = 1, \dots, N$ ) を平均 0, 標準偏差  $\sigma = 0.01$  の正規分布に従う乱数で生成した、 $\mathbf{y} = [y^0, y^0, \dots, y^0]^T + [d_1, d_2, \dots, d_N]^T$  で、ここに  $N = 8$  である。

このシミュレーションデータを多項式  $f(x) = a_0 + a_1x$  ( $M = 1$ ) を用い LSM でモデル化すると、図 1 の破線で示すモデルとなり、 $x > 3.1$  での外挿は負の方向に真の値  $y^0 = 0.1$  から外れていく。モデルパラメータは  $\mathbf{a} = [0.127, -0.0101]^T$  で、この過剰適合は  $a_1$  項により生じる。次に  $f(x) = a_0 + a_1x + a_2x^2 + a_3x^3$  ( $M = 3$ ) を用い LSM でモデル化すると、図の実線で示すようにデータの内挿区間で波打ったように見える曲線となる。モデルパラメータ値は  $\mathbf{a} = [0.0194, 0.161, -0.0840, 0.0130]^T$  である。この曲線を観測データの範囲外  $x > 3.1$  に外挿すると今度は正の方向に真値  $y^0 = 0.1$  から外れていく。このように揺らぎをモデル化して生ずる過剰適合により、これらのモデルでは外挿時の当てはまりが悪い。シミュレーションデータを複数回作成し同様の  $M = 1$  と  $M = 3$  のモデル化を行うと、そのすべてにおいて外挿時の当てはまりが悪いモデル化が見られる。

このような揺らぎのモデル化を、モデルパラメータに非負制約を課して抑制することを考える。まず  $M = 1$  では図 1 のような負の傾きになるモデル化が抑制できる。次に  $M = 3$  では、上述の  $\mathbf{a}$  の要素のように正負のモデルパラメータを持った基底関数項の重ね合わせで波打ったように見える揺らぎがモデル化されることが抑制できる。

ここで本研究のモデル化対象である並列計算機の処理時間のモデル化を考えると、上記で用いたような多項式を用いてモデル化する場合も多く、LSM を用いれば上記のように揺らぎがモデル化される可能性がある。また並列処理

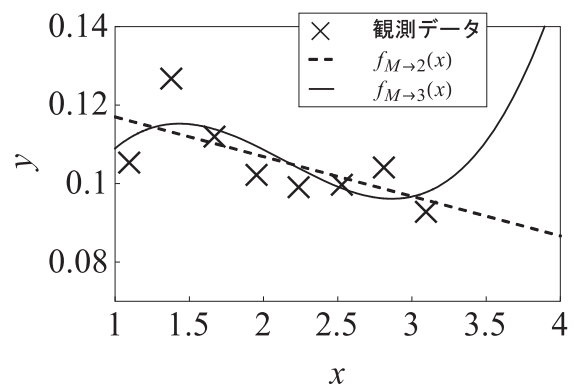


図 1 シミュレーションデータの多項式によるモデル化  
Fig. 1 Modeling of simulation data with polynomial functions.

時間のモデル化でよく使われる  $\log_2(x)$  のようなテラー展開で正負のモデルパラメータを持つ関数が基底関数で使われると、多項式と  $\log_2(x)$  の重ね合わせで  $\log_2(x)$  由来の観測データがモデル化される可能性が生じる。

幸いなことに並列計算機の処理時間のモデル化において観測データは時間であり非負である。そこで基底関数を演算時間、通信時間等の実際の処理に対応するようにモデル化すると、基底関数を非負に設定できる。非負制約を課することができれば、図 1 で示した例のような揺らぎがモデル化される過剰適合を抑制し外挿時に予測を大きく外さない「予測モデル」の構築が可能と考えた。

上述の  $f(x)$  のモデルパラメータに非負制約を課すと各基底関数は単調増加関数となる。したがって基底関数の中には残差を小さくする働きができなくなるものが出ると考えられる。その場合、残差を小さくしてモデルパラメータを決定する計算法では、その基底関数のモデルパラメータは零に漸近すると考えられる。このモデルパラメータの値を 4 章で述べるシンプレックス法で零値にすることにより、過剰適合を引き起こす基底関数が取り除かれたモデルを構築する。これにより並列処理の時間モデルの予測力向上を図ることが本提案の基本的アイデアである。

### 4. シンプレックス法によるモデルパラメータの決定

3章で述べた非負制約を課したモデルパラメータの推定には、ラグランジュ乗数が最大になる基底ベクトルを探す非負制約最小二乗法 [19] を適用することができる。この計算法を用いると、5.1 節の説明変数が 1 変数の例示のように外挿ができるモデル「予測モデル」が得られる場合がある。一方 5.2 節の 2 変数モデル、5.3 節の 3 変数モデルで例示するようにこの計算法ではモデル化できない場合もある。並列処理の処理時間のモデルは、1 章で述べたようにウィークスケーリング等ができることが求められるので、説明変数がプロセッサ数と問題の大きさの、2 変数以上で構成されるモデルのモデルパラメータを決定できる計算法

が必要となる。

そこで我々は、2変数3変数のモデルパラメータが決定できる計算法として、解の候補をすべて列挙して選択する数式処理のアルゴリズムの1つ限量記号消去法(QE)を用いる方法[20]を提案してきたが、本論文ではQEより多くの変数や入力データが扱え、実行可能解から解を選択する線形計画法の解法の1つシンプレックス法を用いた計算法を提案する。

シンプレックス法を適用するためには、LSMのように残差平方和を最小化するのではなく、残差  $e_i (= y_i - f(x_i))$  ( $i = 1, \dots, N$ ) の最大値を最小化する。具体的に話しを進めるためにまず式(6)の連立残差方程式を考える。ここに  $\mathbf{e} = [e_1, e_2, \dots, e_N]^T$  である。

$$\mathbf{e} = \mathbf{y} - \mathbf{f}^T \mathbf{a} \quad (\mathbf{a} \geq 0) \tag{6}$$

残差の最大値を  $\varepsilon (= \text{Max } e_i)$  とし、要素数  $N$  のベクトルを  $\boldsymbol{\varepsilon} = [\varepsilon, \dots, \varepsilon]^T$  として、式(6)から式(7)の連立残差不等式を得る。

$$\varepsilon \geq |\mathbf{y} - \mathbf{f}^T \mathbf{a}| \quad (\mathbf{a} \geq 0) \tag{7}$$

式(7)を解いて得られた  $\varepsilon$  の最小値 ( $e_i$  の最大値でもある) を  $\varepsilon_{\min}$ 、要素数  $N$  のベクトルを  $\boldsymbol{\varepsilon}_{\min} = [\varepsilon_{\min}, \dots, \varepsilon_{\min}]^T$  と置くと、モデルパラメータは式(8)の連立残差不等式を解くことで決定できる。

$$\boldsymbol{\varepsilon}_{\min} \geq |\mathbf{y} - \mathbf{f}^T \mathbf{a}| \quad (\mathbf{a} \geq 0) \tag{8}$$

次に式(7)と式(8)を線形計画法の問題として記述する。まず式(7)の絶対値を外し、式(9)の連立残差不等式とする。

$$-\boldsymbol{\varepsilon} \leq \mathbf{y} - \mathbf{f}^T \mathbf{a} \leq \boldsymbol{\varepsilon} \quad (\mathbf{a} \geq 0) \tag{9}$$

さらに  $\mathbf{f}$  に  $x_i$  の値を代入した  $\mathbf{F}$  を用いて式(9)を線形化し式(10)とする。

$$-\boldsymbol{\varepsilon} \leq \mathbf{y} - \mathbf{F}^T \mathbf{a} \leq \boldsymbol{\varepsilon} \quad (\mathbf{a} \geq 0) \tag{10}$$

式(10)を制約条件とし  $\varepsilon$  を最小化する線形計画法の問題として記述すると式(11)となり、式(7)の解  $\varepsilon_{\min} = \text{Min}(\varepsilon)$  とそれを用いた式(8)の解  $\mathbf{a}$  を同時に得ることができる。なお表示の簡略化のため説明変数が1変数の基底関数の場合で話しを進めてきたが、 $x_i$  を代入したことにより  $\mathbf{F}$  はモデルパラメータ  $a_k$  を変数とする線形基底関数となるので、説明変数  $x_i$  が  $x_{1i}, x_{2i}, \dots$  と複数となっても式(11)を適用できる。

$$\begin{aligned} & \text{Min } \varepsilon \\ & \text{s.t. } \varepsilon + \mathbf{F}^T \mathbf{a} \geq \mathbf{y} \\ & \quad \varepsilon - \mathbf{F}^T \mathbf{a} \geq -\mathbf{y} \\ & \quad \mathbf{a} \geq 0 \end{aligned} \tag{11}$$

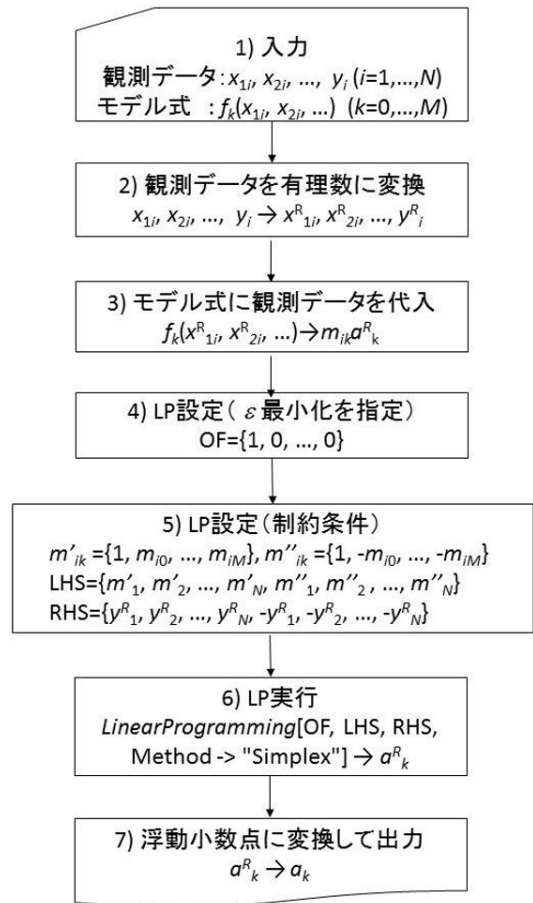


図2 モデルパラメータ推定手順

Fig. 2 The procedure of model parameter estimation.

本論文では式(11)をシンプレックス法で解くことを提案する。シンプレックス法を用いることにより、 $\varepsilon$  を減らすモデル化に必要なモデルパラメータは基底変数として選択され、 $\varepsilon$  を増やす不要なものは非基底変数となり零値を得る。この零値により、小さな値の項がモデル化に必要なかを検討する手間が省略できる。またこの零値により、予測に不要な基底関数の一部が削除されたモデルが自動的に構築できる。さらに、シンプレックス法を有理数演算することを提案する。有理数演算を用いると厳密な計算結果が得られ、計算誤差による誤ったモデルパラメータ選択が回避できる。たとえば5.3節の事例を有理数演算と浮動小数点演算で行い比べると、5ケースすべてで値が異なり、4ケースで異なった変数選択となった。その詳細は議論を参照されたい。

図2に線形計画法の関数を適用して式(11)を解きモデルパラメータを推定する手順を示す。始めにモデル化に用いる観測データとモデル式を入力する。2)番目に入力した観測データを有理数に変換する。3)番目に観測データをモデル式に代入して、 $\mathbf{F}^T$ の要素  $m_{ik}$  を求める。4)番目に目的関数を設定する。最小化するのは  $\varepsilon$  なので1、その他のモデルパラメータは最小化対象外なので  $M+1$  個の0をセットする。5)番目にLPを適用するための制約条件を

セットする。  $m_{ik}'$ ,  $m_{ik}''$  の要素の先頭に加えた 1 は式 (11) の制約式の  $\varepsilon$  に対応する。 6) 番目にシンプレックス法を計算法とする線形計画法の関数を実行する。 この関数にはシンプレックス法を有理数演算で実行できるものが必要となる。 最後に、結果は有理数で得られるので、他の結果と比較するため浮動小数点に変換して出力する。 なお 4), 5), 6) のインプットフォームはツール依存なので、5 章で用いる Mathematica の関数 LinearProgramming のインプットフォームで例示した。

### 5. 適用事例

モデルパラメータに非負制約を課す回帰パラメータの推定に、提案手法である LP による推定により外挿時のモデルの予測精度が向上することを示す。 まず単純な 1 変数モデルに関するシミュレーションとして 2 つの事例を用いることで手法の比較の妥当性を確認し、次に並列時間モデルについて 2 つの事例で確認する。 4 章で提案したシンプレックス法を用いた計算法 (LP) の効果を確認するため、制約なしの最小二乗法、AIC に基づく変数選択をとまなう LSM [21]、非負制約付最小二乗法 (NNLS) [19] について比較する。 これは、提案手法が非負制約付 LP であるため、制約付 LSM と比較するものである。 また非負制約の結果 0 と推定されるパラメータに対する基底関数がモデルから除去されることは、重回帰分析で行われる変数選択と結果的に同等である。 そこで多くの説明変数の候補があり予測に有用な少数の説明変数を用いた重回帰分析を行うために一般的に行われる変数選択として、説明変数の組合せを変え LSM でモデル化した際にモデルの AIC (赤池情報量規準) が最小の組合せを選択する方法 (LSM 変数選択) との比較も行う。

LP のツールとして Mathematica [22] の LinearProgramming 関数を用いた。 この関数は入力を有理数で与えると有理数または整数の結果を返す。 NNLS のツールとしては MATLAB の lsqnonneg 関数 [24] を用いた。 LSM には統計解析ソフトウェア R [23] を用い、パラメータ推定は lm 関数を、AIC に基づく変数選択は step 関数を用いた。

#### 5.1 1 変数モデルによるシミュレーション

##### 5.1.1 $y^0 = x^2$ のモデル化と過剰適合の抑制

観測データとして、真のモデル  $y^0 = x^2$  を基に生成したシミュレーションデータ  $\mathbf{y}$  について、多項式関数  $f(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4 + a_5x^5$  ( $M = 5$ ) でモデル化する。 少ない観測データを用いたモデル化を考え、観測データ数  $N$  が基底関数の数  $M$  に近い観測データ数でかつ  $N \geq M$  を想定し、 $\mathbf{y}$  の要素は、1.1 から 3.1 を等間隔に 8 点 ( $N = 8$ ) で分割した  $x_i$  に対して  $y_i = x_i^2 + d_i$  ( $i = 1, \dots, 8$ ) で、平均 0、標準偏差  $\sigma = 0.01$  の正規分布に従う揺らぎ  $d_i$  を与えて生成した。

表 1 LSM によるモデルパラメータ ( $\sigma = 0.01$ )

Table 1 Model parameters obtained from LSM ( $\sigma = 0.01$ ).

パラメータ	推定値	標準誤差	t 値	p 値
$a_0$	4.30	2.00	2.16	0.164
$a_1$	-11.2	5.38	-2.07	0.174
$a_2$	12.2	5.61	2.17	0.162
$a_3$	-5.42	2.83	-1.92	0.195
$a_4$	1.29	0.691	1.86	0.204
$a_5$	-0.120	0.0658	-1.82	0.211

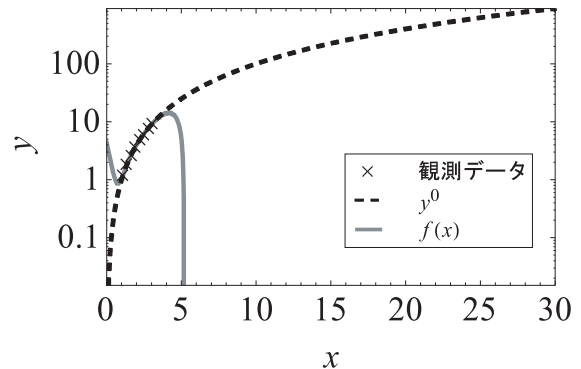


図 3 表 1 によるモデル  $f(x)$  の外挿

Fig. 3 Extrapolation of model  $f(x)$  with Table 1.

表 1 はこの  $\mathbf{y}$  と  $f(x)$  を用いて LSM で得られたモデルパラメータである。 このモデルは真のモデル  $y^0 = x^2$  とは大きく異なる。

図 3 は表 1 を用いて  $f(x)$  を外挿したモデルの予測力の評価である。 破線が  $y^0 = x^2$ 。 実線が表 1 を用いたモデル。  $\times$  印が入力データである。 実線は  $\times$  印を精度良く内挿している。 一方外挿領域は破線とまったく一致しない。 図の実線は、揺らぎ由来の過剰適合が原因で、予測ができなくなったモデルの典型的な例の 1 つである。

過剰適合となる基底関数を用いないモデルを構築するため、変数選択による最適なモデルの推定を試みた。 LSM による回帰係数の推定についてモデルの適合度指標として、 $R^2$  値 = 0.99999、修正  $R^2$  値 = 0.99997 でありモデルの適合は高い。 回帰係数  $a_1, \dots, a_5$  の  $p$  値はいずれも 0.05 より小さな値でなく係数が 0 と判断されるものはない。 またこのシミュレーションデータに対して変数選択を行い AIC を比較したところ、この 5 変数を用いたモデルが AIC 最小であった。 この分析結果からは、過剰適合項を含むことから  $x$  の値が大きい外挿時に予測が大きく外れることが分かる。 切片項の除外も含むすべての可能な組合せに対する回帰モデルの適合度を比較すると、修正  $R^2$  最大のモデルは  $a_0 + a_2x^2 + a_5x^5$  で AIC 最小のモデルは  $a_1x + a_2x^2 + a_3x^3 + a_4x^4 + a_5x^5$  であった。 このように変数選択では、必ずしも真のモデルが選択されるわけではなく、この結果のように過剰適合項が説明変数として残ることもある。 この 1 回の結果から判断することはできない

表 2 100 回の変数選択により選択されたモデル

Table 2 Models selected from 100 times of variable selection.

モデル	回数
$a_2x^2$	19
$a_1x+a_2x^2+a_3x^3+a_4x^4+a_5x^5$	19
$a_1x+a_3x^3+a_4x^4+a_5x^5$	12
$a_0+a_1x+a_2x^2+a_3x^3+a_4x^4+a_5x^5$	9
$a_2x^2+a_3x^3+a_4x^4+a_5x^5$	7
$a_2x^2+a_4x^4+a_5x^5$	7
その他	25

表 3 非負制約付モデルパラメータ ( $\sigma = 0.01$ )

Table 3 Model parameters under non-negative constraint ( $\sigma = 0.01$ ).

	$a_0$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$
LP	0	0	0.994	0.00212	0	0
NNLS	0.	0.	0.999	0.000186	0.	0.

め、切片項の除外も含む AIC に基づく変数選択を、同様に  $y_i = x_i^2 + d_i$  ( $i = 1, \dots, 8$ ) を異なる乱数に対して 100 組生成したものに対して行い、各組で選択されたモデルの回数を表 2 に示す。

真のモデル  $y = a_2x^2$  が最も多く選ばれているが 2 割程度に過ぎず、逆に外挿に対して悪影響をおよぼす  $x^4$  および  $x^5$  の項が含まれたモデルも半分以上見られた。表 1 のモデルの  $x^4$  および  $x^5$  の項に対する回帰係数の値が大きいくことから外挿として  $x$  の値が大きいくときに誤差の影響は大きくなる。

次に、表 1 と同一のデータに対し、本論文の提案である  $\mathbf{a} \geq \mathbf{0}$  の制約を課したモデル化を 4 章で提案した LP と NNLS を用いて行った。なお 5.1 節と図 2 の変数は  $x_i \rightarrow x_{1i}$ ,  $y_i \rightarrow y_i$ ,  $f(x) \rightarrow f(x_{1i})$  と対応する。モデルパラメータを表 3 に示す。両者共  $a_2$  と  $a_3$  に値を持ち、 $a_2$  値はほぼ真値 1 に近くなる。また両者とも  $a_0$ ,  $a_1$ ,  $a_4$ ,  $a_5$  が零となり、揺らぎをモデル化していた過剰適合が抑制されたことが確認できる。

表 3 の入力データと同じ方法で作った 5 組の入力データに対する LP と NNLS の予測を図 4 に示す。両者による 4 回の予測は観測データ  $x_N$  の約 10 倍の  $x = 30$  においてまた  $y_N$  の約 100 倍の  $y = 900$  においてほぼ破線の  $y^0 = x^2$  を予測する。このように非負制約  $\mathbf{a} \geq \mathbf{0}$  を課すことにより図 3 に示したような過剰適合が抑制され、モデルの予測力向上を図れることが確認できる。

さらに同様の実験を 100 組のデータに対して行った。表 4 に LP と NNLS に対する全ケースの平均値を入力データ  $x$  の最大値 3.1 の約 10 倍までのレンジ  $x = 5, 10, 20, 30$  に対して示す。表より、LP と NNLS の両方で、 $y^0(10)$  の値 100 をほぼ予測できるモデル化ができたことが確認できる。

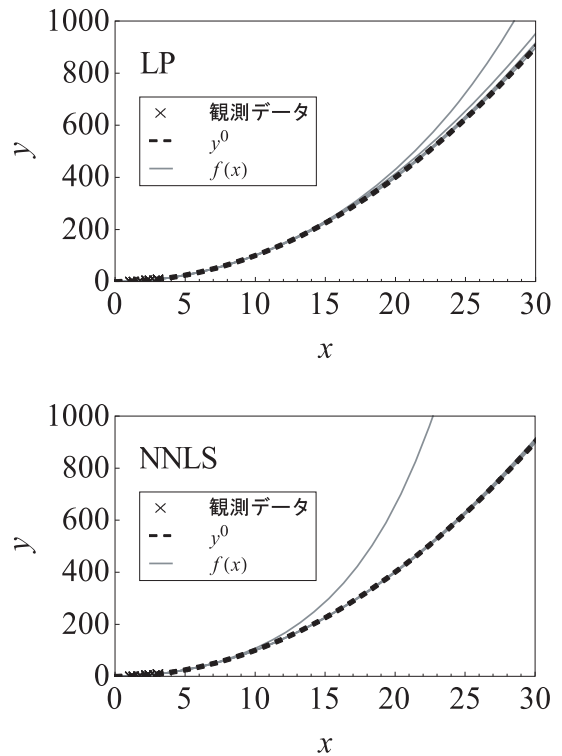


図 4  $x^2$  モデルの予測 (上: LS, 下: NNLS)

Fig. 4  $x^2$  model predictions using LP and NNLS.

表 4 100 ケースの予測値の要約 ( $\sigma = 0.01$ )

Table 4 Statistical summary of 100 predictions ( $\sigma = 0.01$ ).

$x$		5	10	20	30
$y^0(x)=x^2$		25	100	400	900
LP	平均値	25.2	106.	571.	2175.
	標準偏差	(0.235)	(10.1)	(340.)	(2604.)
NNLS	平均値	25.1	105.	548.	2006.
	標準偏差	(0.204)	(8.4)	(279.)	(2130.)

表 5 各モデルパラメータが零になる回数

Table 5 The number of times that each model parameter becomes zero.

	$a_0$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$
LP	84	40	0	56	98	70
NNLS	76	41	0	69	95	68

ところでモデルパラメータが零値ということは、自動的に過剰適合が緩和されたモデルが得られたことを意味するが、100 ケースの入力データに対してモデルパラメータが零値であったりなかったりする基底関数は、予測モデル構築の障害となる可能性が高い。そこで 100 回の各モデルパラメータが零値になるケースをカウントした。これを表 5 に示す。まず真のモデル  $y^0$  と同じ項の  $a_2$  の回数は 0 で、モデル化において正しくこの項が選択されたことが分かる。これに対し、零回になる回数が多いのは  $a_0$ ,  $a_4$ ,  $a_5$  である。このうち  $a_5$  が表 4 における  $x = 30$  の分散を大きく

表 6 LSM によるモデルパラメータ ( $\sigma = 0.01$ )

Table 6 Model parameters obtained from LSM ( $\sigma = 0.01$ ).

パラメータ	推定値	標準誤差	t 値	p 値
$a_0$	-370.	290.	-1.28	0.330
$a_1$	-178.	143.	-1.24	0.34
$a_2$	15.8	12.2	1.29	0.327
$a_3$	-0.961	0.812	-1.18	0.358
$a_4$	-79.7	62.5	-1.28	0.330
$a_5$	534.	422.	1.27	0.333

している原因なので、 $\sigma = 0.01$  の揺らぎがある観測データで 100 回モデル化すると約 70 回で真のモデル  $y^0$  に近い予測モデルが得られたことになる。

5.1.2  $y^0 = x^2 + \log_2(x)$  のモデル化と過剰適合の抑制

5.1.1 項では 1 つの関数から作ったシミュレーションデータを観測データとしたモデル化だったが、現実の並列処理時間は複数の基底関数を必要とする複雑な振舞いをする。そのような場合にも非負制約を課することにより、揺らぎのモデル化を抑制し、適切な基底関数選択ができるかを調べるため、 $x^2$  に加えて並列処理の通信時間等のモデル化でよく使われる  $\log_2(x)$  を用いて真のモデル  $y^0 = x^2 + \log_2(x)$  に対してシミュレーションデータ  $y_i = x_i^2 + \log_2(x_i) + d_i$  を作り、

$$f(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + a_4 \log_2(x) + a_5\sqrt{x}$$

( $M = 5$ )

としてモデル化した。ここにシミュレーションデータ  $x_i$ ,  $d_i$  は 5.1.1 項と同様に生成した。

$x_1 = 1.1$  から  $x_8 = 3.1$  という短い区間の  $\log_2(x)$  曲線は  $M = 3$  の多項式でもそれらしく近似できるが、外挿すると両者はまったく異なる振舞いをする。HPC ではモデル化に  $\log_2(x)$  をしばしば使うので、 $f(x)$  のように多項式と  $\log_2(x)$  の両方をモデル化に用いた場合、観測データから両者の振舞いを分離できることが重要である。したがって、予測モデルができるかということ以外に、 $a_4$  が 1 に近い値になるということがこの実験の注目点である。

表 6 は LSM で得られたモデルパラメータで、真のモデルの  $a_2 = a_4 = 1$ ,  $a_0 = a_1 = a_3 = a_5 = 0$  とは大きく異なる結果となる。また正負の推定値は揺らぎがモデル化されたことを示す。

LSM による回帰係数の推定についてモデルの適合度指標として、 $R^2$  値 = 0.999998, 修正  $R^2$  値 = 0.999993 であり、5.1.1 項の実験と同様に回帰係数  $a_1$  から  $a_5$  の  $p$  値がほぼ同様ですべてのパラメータは 0 と見なされない。また AIC に基づく変数選択で削除される項はなく、切片除去も含む変数選択において  $f(x) = a_1x + a_2x^2 + a_3x^3$  が AIC 最小であった。

同様の実験を 100 回行った際に、切片項の除外も含む変数選択で AIC 最小として選択されたモデルの個数について

表 7 100 回の変数選択により選択されたモデル

Table 7 Models selected from 100 times of variable selection.

モデル	回数
$a_1x + a_4 \log_2(x) + a_5 \sqrt{x}$	26
$a_0 + a_2x^2 + a_3x^3 + a_4 \log_2(x) + a_5 \sqrt{x}$	19
$a_1x + a_2x^2 + a_3x^3 + a_4 \log_2(x)$	13
$a_1x + a_3x^3 + a_4 \log_2(x) + a_5 \sqrt{x}$	13
$a_2x^2 + a_3x^3 + a_4 \log_2(x) + a_5 \sqrt{x}$	11
$a_1x + a_2x^2 + a_3x^3 + a_5 \sqrt{x}$	8
その他	10

表 8 非負制約付モデルパラメータ ( $\sigma = 0.01$ )

Table 8 Model parameters under non-negative constraint ( $\sigma = 0.01$ ).

	$a_0$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$
LP	0	0	0.985	$3.84 \times 10^{-3}$	1.02	0
NNLS	0.	0.	0.998	$3.01 \times 10^{-4}$	1.00	0.

て表 7 に示す。

真のモデルは 100 回中 1 回も選択されなかった。

次に表 6 と同一のシミュレーションデータに対し、 $\mathbf{a} \geq 0$  の制約を課して LP と NNLS による推定を行った結果を表 8 に示す。表は、これら 2 つの計算法の両方で  $a_2$  と  $a_4$  がほぼ 1 で選択されたことを示す。また  $a_0$ ,  $a_1$ ,  $a_5$  が零となり、揺らぎをモデル化する過剰適合が抑制された予測モデルが構築できたことを示す。

5.1.1 項と同様に、表 8 の入力データと同じ方法で作った 5 組の入力データに対する LP と NNLS の予測を図 5 に示す。破線は  $y^0 = x^2 + \log_2(x)$  である。この図より、 $\mathbf{a} \geq 0$  を用いるとモデルの予測力向上が図れることが確認できる。

同じ  $y^0$  と  $\sigma = 0.01$  に対して揺らぎが異なる入力データを 100 ケース用意し、 $\mathbf{a} \geq 0$  の効果を確認した。全ケースの  $x = 5, 10, 20, 30$  における平均値と標準偏差を表 9 に示す。結果、入力データに揺らぎがあっても LP と NNLS の両方で、入力データ  $x_N = 3.1$  の約 10 倍の  $y^0(30)$  を大きく外れずに予測するモデル化ができることが確認できた。

表 10 に 100 ケースの各モデルパラメータが零になるケースの数を示す。 $x^2$  と  $\log_2(x)$  の  $a_2$  と  $a_4$  項の回数が全ケース 0 ゆえ、これらの項が  $\sigma = 0.01$  の揺らぎデータから確実に選択できたことが確認できる。また  $a_5$  の回数もほぼ 100 で、 $\sqrt{x}$  が残差を増加させる項であることが確認できる。

以上のことから、1 変数モデルのモデルパラメータに非負制約を課することにより、揺らぎ起因の過剰適合が抑制で

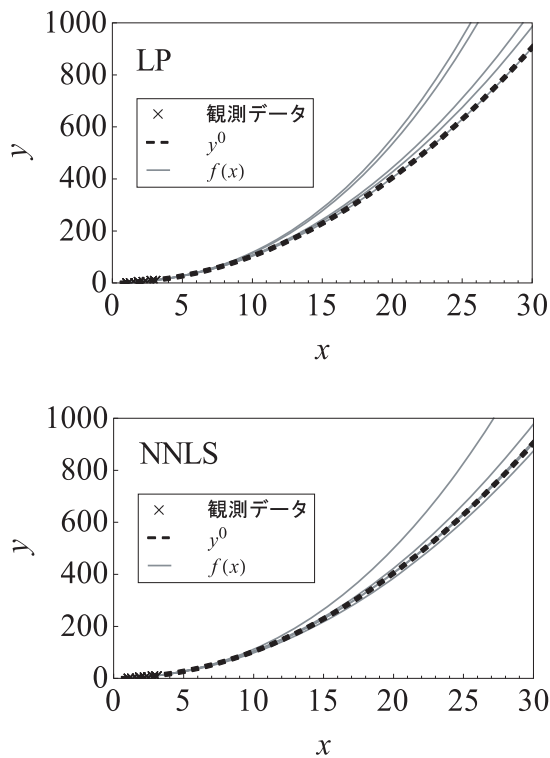


図 5  $x^2 + \log_2(x)$  モデルの予測 (上: LP, 下: NNLS)  
 Fig. 5  $x^2 + \log_2(x)$  model predictions using LP and NNLS.

表 9 100 ケースの予測値 ( $\sigma = 0.01$ )

Table 9 Statistical summary of 100 predictions ( $\sigma = 0.01$ ).

X		5	10	20	30
$y^0(x)=x^2+\log_2(x)$		27.3	103.	404.	905.
LP	平均値	27.5	107.	451.	1076.
	(標準偏差)	(0.239)	(4.54)	(50.2)	(186.)
NNLS	平均値	27.5	107.	443.	1048.
	(標準偏差)	(0.211)	(3.95)	(43.5)	(161.)

表 10 各モデルパラメータが零になる回数

Table 10 The number of times that each model parameter becomes zero.

	$a_0$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$
	71	65	0	30	0	99
NNLS	68	54	0	27	0	100

ることがシミュレーションにより確認できた。またこの過剰適合抑制は LP と NNLS のどちらでも確認できたが、LSM の変数選択では必ずしも良いモデルは選択されなかった。このことから次節以降の実際の並列処理時間のモデル化において、非負制約付きの既存の計算法として提案手法に対して NNLS を比較対象とすることにした。

また変数選択の考え方としての比較として、真のモデルを選択する可能性もあるため、非負制約を課さない LSM による変数選択についても比較対象とすることにした。

## 5.2 2変数モデルの場合

5.1 節の 2 つの例は 1 変数モデルであったが、導入で述べたように並列処理の性能のスケールアップには、少なくともプロセッサ数と問題の大きさの 2 変数からなる並列処理時間モデルが必要となる。そこで分子動力学プログラム [25] を並列計算機 IBM SP2 で実行した処理時間をこれら 2 変数でモデル化したモデル [26] を使い、LP によりモデル化した結果、過剰適合が緩和されて予測力が向上し、外挿ができる「予測モデル」が構築できることを確認した。一方比較のため行った LSM 変数選択と NNLS によるモデル化では、外挿に強い予測モデルの構築はできなかった。

並列処理時間のモデル化では、日々の並列処理の際に採取できる観測データからモデル化できることが望ましい。そこで観測データは、簡単に採取できるウォークロックタイム (処理を投入してから終了するまでの時間) のみとした。次に日々の処理計算では同じ問題を解かないことから、問題の大きさが異なる観測データを用いるモデル化を想定した。またモデル化が必要となる時は、並列処理するようになってから日が浅い場合が多いと考えられるので、観測データが少ないことを想定した。

以上の条件から観測データ  $y$  はプロセッサ数  $p = \{2, 4, 6, 8, 10\}$  と、問題の大きさ  $n = \{3200, 7200, 12800, 20000\}$  という 2 つの変数の組合せで 20 通り測定したウォークロックタイム中の 10 点を用いてモデル化した。また乱数を用いてこの 10 点の観測データの組合せを 5 ケース作り、任意の観測データから予測モデルが構築できるかを確認した。

### (1) 処理時間モデルの基底関数の抽出

モデルの基底関数はプログラムから抽出した。モデル化した分子動力学プログラムは、ファンデルワールス力と重力の影響下で時間発展するアルゴン原子の巨視的振舞いをシミュレーションする分子動力学プログラムである。

このような並列計算されるシミュレーションプログラムの処理時間をモデル化する場合、プログラムのすべての処理をモデル化するのではなく、処理時間の大部分を費やすカーネルを見つけてその部分の処理を式化し、それらをアセンブルしてプログラムの処理時間モデルを構築する方法が一般的である。そこで時間全体の 99% の時間を占めるサブルーチンの処理時間を式化した [26]。具体的には、サブルーチン中のカーネル (do ループ, MPI 通信部) の処理時間を、問題の大きさ  $n$  の関数であるループ回転数に比例するものとし、次にその処理時間はプロセッサ数  $p$  に反比例するとして式化した。付録 A.1 にサブルーチンごとの処理時間のモデルを示す。表で示したように、時間全体の 99% の時間を占めるサブルーチンは 5 つあり、そのうち 3 つのサブルーチンで並列処理が行われる。  $T_{\text{force}}$  は粒子間力の計算時間である、  $T_{\text{tabe}}$  は遮蔽距離内に在る粒子の表を作る、  $T_{\text{propcel}}$  は粒子の物理データをメッシュに落として物理量を計算する、  $T_{\text{cross}}$  は粒子の境界での反射



表 11 LP により推定された式 (12) のモデルパラメータ  
Table 11 Estimated model parameters of Eq. (12) using LP.

No.	$a_0$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$	$a_7$
1	0	0	$3.11 \times 10^{-6}$	$3.54 \times 10^{-2}$	0.307	$6.55 \times 10^{-4}$	0	3.53
2	0	0	$3.77 \times 10^{-6}$	$3.75 \times 10^{-2}$	0.296	$4.95 \times 10^{-4}$	0	3.56
3	0	0	$3.52 \times 10^{-6}$	$4.11 \times 10^{-2}$	0.293	$1.52 \times 10^{-4}$	0	3.21
4	0	0	$3.01 \times 10^{-6}$	$4.00 \times 10^{-2}$	0.301	0	0	3.64
5	4.34	$2.91 \times 10^{-7}$	$1.51 \times 10^{-6}$	$4.17 \times 10^{-2}$	0.302	0	0	1.21

を処理し,  $T_{moves}$  では粒子の移動を処理する. 表中  $I_s$  は時間の繰返し数,  $N_{sample}$  は観測の回数, そして  $n_{table}$  は時間積分の繰返しの何回目に遮蔽距離内に在る粒子の表を更新するかを決める間隔である.  $t_u$  はカーネルの逐次時間の計算時間,  $t_{0u}$  はその立ち上がり時間,  $a_u$  は論理性能  $r_a$  (flops) により規格化した効率である. 付録 A.1 中の  $\sigma$  は MPLALLTOALL で行った総和計算をモデル化した通信時間であり,  $t_{0c}$  はその立ち上がり時間,  $e_c$  は通信の論理性能  $r_c$  (MB/s) で規格化した効率である. また各プロセッサの総和計算時間は  $t_{0sum}$  とした.  $t_p$  は  $t_u$  と  $\sigma$  の和からサブルーチンの並列処理時間で,  $t_{0p}$  は立ち上がり時間,  $e_p$  はカーネルの並列処理の効率である.

これらの処理時間の総和  $T_{MD}$  ( $= T_{force} + T_{tabe} + T_{propcel} + T_{cross} + T_{moves}$ ) に対し, 変数  $p$  と  $n$  で式をまとめると, プログラム全体の処理時間を式 (12) のように, 8つのモデルパラメータを持つ 2 変数  $p, n$  のモデルとして記述することができる.

モデルパラメータと式化したカーネルモデルの係数の関係を付録 A.2 に示す. 各パラメータは主に立ち上がり時間  $t_0$  に関するもの, 理論性能に対する効率である  $r_a, r_c, e_p$  に関するものおよび, 立ち上がり時間とこれら割合の混合からなる. これら立ち上がり時間と効率は原理的には正であると考えられるので, 付録 A.2 の 1 列目に示したモデルパラメータ, すなわち式 (12) のモデルパラメータはすべて非負値と想定し, 4 章で提案した LP を適用した.

$$T_{MD}(p, n) = a_0 + a_1 n^2 + a_2 \frac{n^2}{p} + a_3 n + a_4 \frac{n}{p} + a_5 n p + \frac{a_6}{p} + a_7 p \quad (12)$$

(2) LP によるモデル化と予測力向上

式 (12) のモデルパラメータを  $\mathbf{a} \geq \mathbf{0}$  とし,  $p \rightarrow x_{1i}, n \rightarrow x_{2i}, y \rightarrow y_i, T_{MD}(p, n) \rightarrow f(x_{1i}, x_{2i})$  として図 2 を適用して, モデルパラメータを推定した. 表 11 の No.1~5 に冒頭で述べたように乱数で観測データを 5 ケース選んだ推定値を示す. 表で値が 0 の項は, 自動的に削除された過剰適合項である. 全ケースで零となる  $a_6$  はどの観測データもモデル化にこの項を必要としないことを示す.

この表のモデルパラメータを用いて問題の大きさ  $n = 96800$  の処理時間の予測を確認した結果を図 6 に

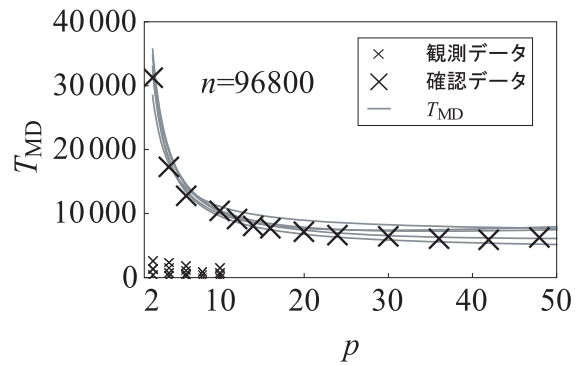


図 6 LP で構築した MD 処理時間モデルの予測  
Fig. 6 Predictions of MD processing time with LP-based models.

示す. 図はプロセッサ数  $p$  に対する経過時間  $T_{MD}$  である. 大きな  $\times$  印は予測確認のための観測データである. 実線が LP を用いた 5 ケースのモデルの予測で, データの組合せが変わっても  $\times$  印の値と傾向が予測できた. 図の小さい  $\times$  印はモデル化に用いた観測データで, 用いたプロセッサ数の最大値は  $p = 10$ , 問題の規模の最大値は  $n = 20000$  である. 大きな  $\times$  印の確認データのプロセッサ数の最大値は  $p = 48$ , 問題の規模の最大値は  $n = 96800$  であるので, 図は  $p$  と  $n$  で各々約 5 ( $\sim 48/10$ ) 倍, 5 ( $\sim 96800/20000$ ) 倍の領域を予測できたことを示す.

表 12 は LSM により AIC に基づいて変数選択した結果推定されたモデルパラメータである. 表中「削除」と書かれているモデルパラメータは, 変数選択で削除されたことを示す. 表は入力データにより削除される項が異なることを示す. また負の値を持つモデルパラメータが存在していることから, 基底関数の合成により揺らぎ等のノイズをモデル化する現象が生じていることが分かる. この表のモデルパラメータを用いた予測を図 7 に示す. 図中の実線が予測で, プロセッサ数の増加をまったく予測できない場合が 1 ケースあることを示す.

LP と同じ観測データと式 (12) に対して  $\mathbf{a} \geq \mathbf{0}$  の条件を付与した NNLS でモデルパラメータを決定すると, 5 ケースの番号順に  $a_1 = \{6.50 \times 10^{-6}, 5.42 \times 10^{-6}, 6.01 \times 10^{-6}, 3.34 \times 10^{-6}, 3.90 \times 10^{-6}\}$ , 他のモデルパラメータはすべて 0. となった. 結果, 得られたモデルは図 8 の実線となり, 予測モデルを作ることができなかった.

表 12 LSM で AIC に基づく変数選択により推定されたモデルパラメータ

Table 12 Estimated model parameters using LSM based on AIC variable selection.

No.	$a_0$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$	$a_7$
1	削除	削除	$2.13 \times 10^{-6}$	$3.55 \times 10^{-2}$	0.328	$9.00 \times 10^{-4}$	-93.1	削除
2	74.6	$-13.3 \times 10^{-7}$	$11.6 \times 10^{-6}$	$3.18 \times 10^{-2}$	0.244	$2.87 \times 10^{-3}$	削除	-8.56
3	72.7	$-5.08 \times 10^{-7}$	$6.29 \times 10^{-6}$	$3.93 \times 10^{-2}$	0.264	$1.46 \times 10^{-3}$	削除	-10.9
4	48.8	削除	$2.83 \times 10^{-6}$	$2.50 \times 10^{-2}$	0.334	$1.93 \times 10^{-3}$	15.8	削除
5	-594.	$-11.5 \times 10^{-7}$	$9.02 \times 10^{-6}$	$12.2 \times 10^{-2}$	削除	$-4.06 \times 10^{-3}$	1918.	40.4

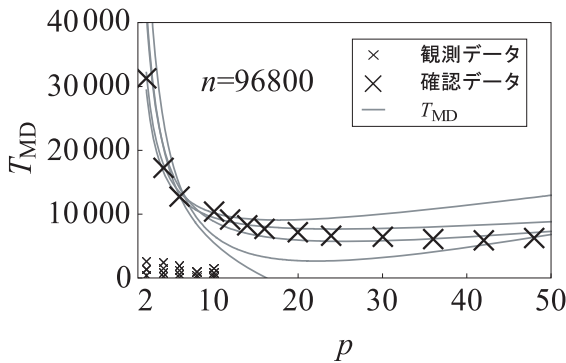


図 7 LSM で AIC に基づく変数選択により構築した MD 処理時間モデルの予測

Fig. 7 Predictions of MD processing time models with LSM based on AIC variable selection.

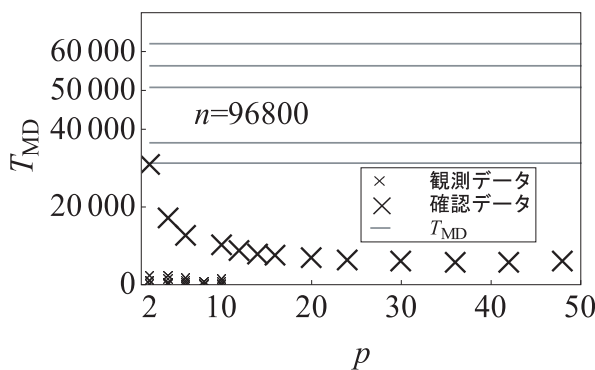


図 8 NNLS で構築した MD 処理時間モデルの予測

Fig. 8 Predictions of MD processing time with NNLS-based models.

表 13 MD モデルの残差平方和 E

Table 13 Residual sum of squares for MD models.

データ No.	1	2	3	4	5
LP	1862.	787.	972.	2289.	784.
NNLS ( $\times 10^6$ )	5.66	3.09	5.44	11.3	3.26

この原因を探るため、LP、NNLS の残差平方和  $E$  を式 (1) で決定して比較した。これを表 13 に示す。すべての観測データにおいて  $E_{NNLS} \gg E_{LP}$  であり、NNLS の残差平方和の最小化が十分でないことが分かる。

以上のことから、2 変数モデルの予測力向上を図るのに、LP が有効である場合があることが確認できた。残差平方

和の桁が 2-3 桁異なることから予測誤差について外挿時に数桁の予測誤差の違いとなるため、LP が外挿時の予測精度が高いことが分かる。

### 5.3 3 変数モデルの場合

5.2 節では 2 変数モデル構築において有理演算 LP を用いた提案方法が並列処理時間の予測モデル構築に有効であることを確認したが、並列処理時間モデルではモデルが 3 変数になる場合もある。そこで提案したモデル化方法を、3 変数モデルとしてポピュラーな密行列式を解く性能ベンチマークコード High Performance Linpack [27] (以後 HPL) を並列計算機 FUJITSU HPC2500 [28] で実行した処理時間をモデル化した。

#### (1) 処理時間モデル

HPL は次元数  $N$  の密行列計算を  $P \times Q$  の 2 次元プロセス格子に分割し、並列処理するプログラムである。その処理時間モデルは計算量が処理時間に比例すると仮定して得られる [27]。このとき panel factorization と trailing sub matrix 計算で生じる通信のモデルパラメータが異なることを想定して各々の通信時間を  $t_{cf} = \alpha_f + \beta_f L$ ,  $t_{cm} = \alpha_m + \beta_m L$  とする。ここに  $L$  は通信量である。次に HPC2500 がクロスパーネットワークで通信することを考慮し、立ち上がり時間  $\alpha_f$ , と  $\alpha_m$  が  $P$  に比例すると仮定して  $\alpha = aP + b$  として処理時間を求めると、処理時間モデルは式 (13) のように  $P, Q, N$  の 3 変数 7 パラメータモデルとなる。ここに  $NB$  はプロセス格子のブロックサイズ、 $\gamma, \beta_f, \beta_m, a_f, a_m, b_f, b_m$  がモデルパラメータである。

$$\begin{aligned}
 T_{HPL}(P, Q, N) &= \gamma \frac{2N^3}{3PQ} + \beta_f \frac{N^2}{2P} + \beta_m \frac{3N^2}{2Q} \\
 &+ a_f P \frac{N(1 + NB \log_2(P))}{NB} + a_m P \frac{N(P-1 + \log_2(P))}{NB} \\
 &+ b_f \frac{N(1 + NB \log_2(P))}{NB} + b_m \frac{N(P-1 + \log_2(P))}{NB}
 \end{aligned} \tag{13}$$

観測データであるウォールクロックタイム  $y$  は前節と同様になるべく少なくし、 $NB = 100$  に対して問題の大きさ  $N = \{10000, 15000, 20000, 25000, 30000, 35000\}$  とプロセス数  $Q = \{10, 20, 30, 40\}$  と  $P = 1$  の組合せからラン

表 14 LP により推定された式 (13) のモデルパラメータ  
Table 14 Estimated model parameters of Eq. (13) using LP.

No.	$\gamma$	$\beta_f$	$\beta_m$	$a_f$	$a_m$	$b_f$	$b_m$
1	$4.1 \times 10^{-10}$	$2.61 \times 10^{-7}$	$1.23 \times 10^{-7}$	$2.25 \times 10^{-5}$	0	0	0
2	$4.08 \times 10^{-10}$	$2.73 \times 10^{-7}$	$1.30 \times 10^{-7}$	$2.26 \times 10^{-5}$	0	0	0
3	$4.17 \times 10^{-10}$	$2.53 \times 10^{-7}$	$1.10 \times 10^{-7}$	0	$2.38 \times 10^{-4}$	$2.18 \times 10^{-4}$	0
4	$4.22 \times 10^{-10}$	$2.45 \times 10^{-7}$	$0.866 \times 10^{-7}$	$1.08 \times 10^{-5}$	0	$1.36 \times 10^{-4}$	$7.96 \times 10^{-3}$
5	$4.23 \times 10^{-10}$	$2.35 \times 10^{-7}$	$1.26 \times 10^{-7}$	$1.97 \times 10^{-5}$	$0.178 \times 10^{-4}$	0	0

ダムに選んだ4点,  $N$  とプロセッサ数  $P = \{10, 20, 30, 40\}$  と  $Q = 1$  の組合せからランダムに選んだ4点と全組合せの中から重複しない1点の計9点を1ケースとし, 前節と同様5ケース用意した. 同じ問題を2度計算しないという観点から, ランダムに選ぶ  $N$  と  $Q$  および  $N$  と  $P$  の各組合せにおいて,  $N$  が重複する組合せを避けた. 提案方法の有効性を調べるため, LSM 変数選択と NNLS で同様のモデル化を行い比較した.

(2) LP によるモデル化と予測力向上

式 (13) と図 2 のタームを  $P \rightarrow x_{1i}, Q \rightarrow x_{2i}, N \rightarrow x_{3i}, y \rightarrow y_i, T_{HPL3D}(P, Q, N) \rightarrow f(x_{1i}, x_{2i}, x_{3i})$  と対応付けて, 推定したモデルパラメータを表 14 の No.1~5 に示す. なお図中のモデルパラメータとは  $\gamma \rightarrow a_1, a_2 \rightarrow \beta_f, a_3 \rightarrow \beta_m, a_4 \rightarrow a_f, a_5 \rightarrow a_m, a_6 \rightarrow b_f, a_7 \rightarrow b_m$  と対応する. 表で値が零の項は, 過剰適合項として自動的に削除されたことを示す. 5 ケースの  $\gamma, \beta_f, \beta_m$  の値はほぼ同じで, 5 ケースの入力データに対してパラメータ決定が安定して行われていることが確認できる.  $a_m, b_f, b_m$  に零値が出現し, これらの項がその入力モデル化に必要な可能性が高いことが分かる.

図 9 の実線は式 (13) と表 14 のモデルパラメータを用いたモデルの LP の予測の確認テストである. 大きな  $\times$  印は確認のための測定値である. 小さい  $\times$  印はモデル化に用いた観測データである.  $P = 1, N = 80000$  のケースでは問題の大きさにして約 12 ( $= 80000^3/35000^3$ ) 倍の領域で  $P = 40 \sim 120$  の予測ができることが確認できる.  $Q = 1, N = 45000$  の場合では, 問題の大きさにして約 2 ( $= 45000^3/35000^3$ ) 倍の領域で  $T_{HPL}$  が最小値になるまでの予測がほぼできることが確認できる.

結果, 3 変数 7 パラメータの並列処理時間の予測モデルがウォールクロックタイムのみの入力データから構築できることが確認できた.

前節と同様に AIC による変数選択で決定したモデルパラメータを表 15 に, 予測の確認を図 10 に示す. 入力データは表 14 と同じである. 図中の実線は式 (13) の予測で, 同上図  $P = 1, N = 80000$  では全入力データの予測が大きい  $\times$  の確認データと一致していることを示す. 一方同下図の  $Q = 1, N = 45000$  に対しては, 1 つの入力データで確認データとまったく一致せず, 残りの 4 つのデータも確認

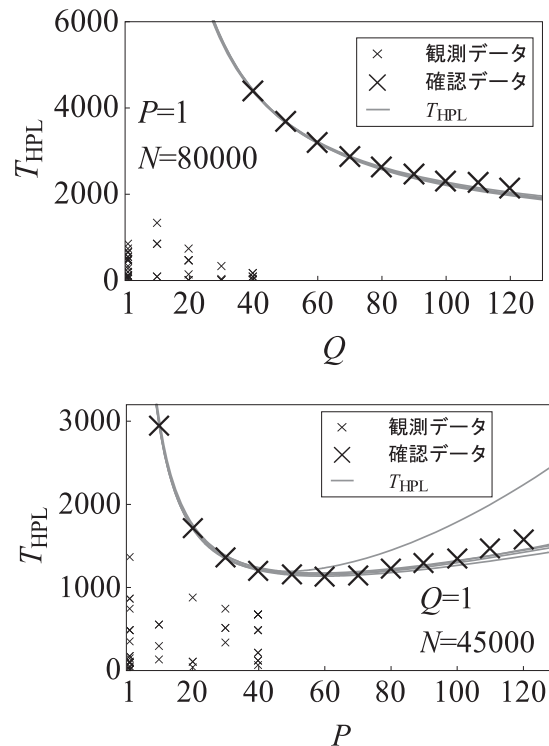


図 9 LP で構築した HPL 処理時間モデルの予測  
Fig. 9 Predictions of HPL processing time with LP-based models.

データの傾向をとらえていないことを示す.

モデルパラメータを NNLS で計算した結果, データ番号順に  $\gamma = \{5.94 \times 10^{-10}, 5.79 \times 10^{-10}, 6.51 \times 10^{-10}, 5.33 \times 10^{-10}, 6.14 \times 10^{-10}\}$ , その他のモデルパラメータはすべて零となり図 11 となった. 図はモデルが  $P = 1$  の場合の予測ができることを示す. 一方  $P > 1$  の場合に生じる,  $P$  に対して凹となる現象がモデル化できなかったことを示す.

表 16 に LP と NNLS の残差平方和  $E$  を示す. LP の  $E$  は  $10^0 \sim 10^2$  のオーダーで, NNLS は  $10^5$  のオーダーである. これらのことから 5.2 節と同様, LP の外挿時の予測精度が高いことが分かる.

以上のことから 3 変数モデルでは, LP でのみ良い予測モデル構築が可能であることが確認できた.

6. 議論

シンプレックス法を有理数演算するメリットを考える.

表 15 LSM で AIC に基づく変数選択により推定されたモデルパラメータ  
 Table 15 Estimated model parameters using LSM based on AIC variable selection.

No.	切片	$\gamma$	$\beta_f$	$\beta_m$	$a_f$	$a_m$	$b_f$	$b_m$
1	3.23	$4.10 \times 10^{-10}$	$2.83 \times 10^{-7}$	$1.26 \times 10^{-7}$	$-4.29 \times 10^{-3}$	$1.10 \times 10^{-2}$	$-3.86 \times 10^{-2}$	2.10
2	0.660	$4.20 \times 10^{-10}$	$2.43 \times 10^{-7}$	$1.25 \times 10^{-7}$	$-1.08 \times 10^{-4}$	削除	$-2.10 \times 10^{-3}$	0.0876
3	0.121	$4.14 \times 10^{-10}$	$2.54 \times 10^{-7}$	$1.15 \times 10^{-7}$	$-2.53 \times 10^{-5}$	$3.09 \times 10^{-4}$	削除	0.0115
4	1.03	$4.23 \times 10^{-10}$	$2.42 \times 10^{-7}$	$0.790 \times 10^{-7}$	削除	削除	$-4.66 \times 10^{-5}$	0.0162
5	0.202	$4.19 \times 10^{-10}$	$2.46 \times 10^{-7}$	$1.22 \times 10^{-7}$	$-1.01 \times 10^{-4}$	削除	$-2.18 \times 10^{-3}$	0.0855

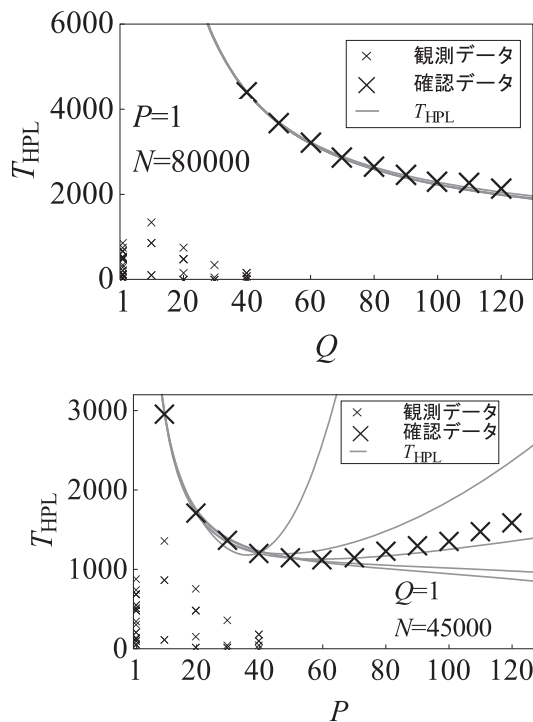


図 10 LSM で AIC に基づく変数選択により構築した HPL 処理時間モデルの予測

Fig. 10 Predictions of HPL processing time models with LSM based on AIC variable selection.

5.3 節の事例を有理数演算と浮動小数点演算で行い比べると、5 ケースすべてで値が異なり、4 ケースで異なった変数選択となった。たとえば  $\mathbf{a}$  の要素を  $\{\varepsilon, \gamma, \beta_f, \beta_m, a_f, a_m, b_f, b_m\}$  とすると、ケース 1 は、有理数演算で  $\{0.843, 4.10 \times 10^{-10}, 2.61 \times 10^{-7}, 1.23 \times 10^{-7}, 2.25 \times 10^{-5}, 0, 0, 0\}$ 、浮動小数点演算で  $\{0, 4.16 \times 10^{-10}, 2.51 \times 10^{-7}, 1.16 \times 10^{-7}, 2.02 \times 10^{-5}, 3.38 \times 10^{-5}, 0, 0\}$  となる。この問題は解法に改訂シンプレックス法を用いることにより解決でき、浮動小数点で表した有理数演算のシンプレックス法の結果と比較すると有効数字 6 桁で一致する。この差異は有理数演算のシンプレックス法と改訂シンプレックス法では生じなかった。また適当なピボットが見つからない等、シンプレックス法が解を決定できない場合を除くと、有理数演算の結果はモデル化に用いた観測データに対する厳密解である。したがってモデルパラメータ決定に利用した計算法の計算誤差を、他の計算法ほど気にしないで済むものとする。

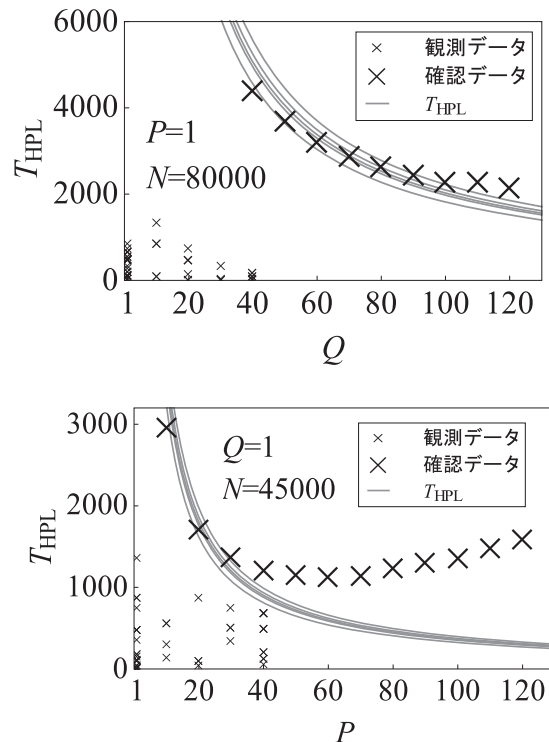


図 11 NNLS で構築した HPL 処理時間モデルの予測

Fig. 11 Predictions of HPL processing time with NNLS-based models.

表 16 HPL モデルの残差平方和  $E$   
 Table 16 Residual sum of squares for HPL models.

No.	1	2	3	4	5
LP	4.53	275.	4.02	5.95	27.5
NNLS( $\times 10^5$ )	0.429	1.39	1.28	1.92	2.12

提案した計算法 LP ではモデルパラメータに非負制約が付与できることが前提である。もし非負制約が適用できないモデル化に LP を適用すると、たとえば真のモデル  $y^0 = -0.5 + x^2$  に対して揺らぎ  $d_i$  ( $i = 1, \dots, N$ ) を加え、シミュレーションデータ  $y_i = -0.5 + x_i^2 + d_i$  ( $i = 1, \dots, 8$ ) を生成しモデル化を行うと、 $f(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + a_4x^4 + a_5x^5$  に対しては  $a_0 = 0, a_1 = 0, a_2 = 0.613, a_3 = 0.113, a_4 = 0, a_5 = 0$  となり、モデル化に必要な  $a_0$  項は過剰適合項として削除され、真のモデル  $y^0$  とは異なるモデルが得られる。

並列処理時間のモデル化においても、付録 A.1 のような個々のカーネルをモデル化すると、モデルパラメータの1つ立ち上がり時間  $t_{0u}$  が負になる場合がある [26]. 問題の大きさを変えたときキャッシュの効果が変わることを見ると、複雑な現象を線形近似でモデル化したためにそのしわ寄せが  $t_{0u}$  に生じたと推察する. これを確かめるには非負や非正、或いはそれらの条件が混在したモデルパラメータ決定が必要と思われる. それは限量記号消去法 (QE) [20] により可能であるが、非負で抑制した観測データの揺らぎのモデル化が復活する可能性があり、議論は本論文の範囲を超えるので将来の研究課題としたい.

## 7. まとめ

並列処理時間の回帰モデルの予測力向上のため、モデルパラメータに非負の条件を付与することで観測の揺らぎがモデル化される過剰適合の抑制を試みた. モデルパラメータを決定し、非負制約を課したことにより過剰適合項となった基底関数のモデルパラメータを零値にするために、シンプレックス法を用いた計算法 (LP) を提案した. また LP の計算誤差を抑制するために有理数演算にすることを提案した. 適用事例から次の効果を確認した.

- 1) モデルパラメータに非負制約を課すことにより、複数の基底関数の重ね合わせで揺らぎがモデル化される過剰適合が抑制でき、モデルの予測力が向上する.
- 2) 非負制約を課すことにより残差を増やすことしかできなかった基底関数のモデルパラメータが零になり、過剰適合を起す基底関数のいくつかを抑制したモデルが自動的に構築できる.
- 3) 説明変数が1変数のモデル化では、LP と既存の非負制約付き計算法である非負最小二乗法 (NNLS) で上記 1), 2) の効果が確認できた. 一方 LSM 変数選択では確認できなかった.
- 4) 説明変数が2, 3変数のモデル化では、各々5ケースの観測データを用いた LP, LSM 変数選択, NNLS の比較では、LP でのみで 1), 2) の効果が観測できた. また異なる5つの観測データの組合せで予測確認データの値とほぼ一致する予測モデルが構築できた.

以上のことから並列処理時間の「予測モデル」構築に LP が有効であると考えられる. ウォークロックタイムを観測データとし、少ないデータで「予測モデル」を構築できることが期待できる提案した計算法 LP が、多くの並列計算機とそれに関わる資源の有効利用に役立つものと期待する.

## 参考文献

[1] Buzbee, L.B.: Plasma simulation and fusion calculation, LA-UR-83-1633, Los Alamos National Laboratory (1983).  
 [2] Adams, M.L. and Crockett, W.T.: Modeling algorithm execution time on processor arrays, *IEEE Computer*,

Vol.17, No.7, pp.38–43 (1984).  
 [3] Moler, C.: Matrix Computation on Distributed Memory Multiprocessors, Heath, M. (Ed.) Hypercube Multiprocessors, SIAM, pp.181–195 (1986).  
 [4] Thomasian, A.: Analytic queuing network models for parallel processing of task systems, *IEEE Trans. Computers*, Vol.C-35, No.12, pp.1045–1054 (1986).  
 [5] Hockney, W.R.: Parameterization of computer performance, *Parallel Comput.*, Vol.5, pp.97–103 (1987).  
 [6] Clement, J.M. and Quinn, J.M.: Analytical performance prediction on multicomputers, *Proc. Supercomputing '93*, pp.886–894 (1993).  
 [7] Menasce, D., Noh, H.S. and Tripathi, K.S.: A methodology for the performance prediction of massively parallel applications, *Proc. IEEE Symposium on Parallel and Distributed Processing*, pp.250–257 (1993).  
 [8] Crovella, E.M. and LeBlanc, J.T.: Parallel performance prediction using lost cycle analysis, *Proc. Supercomputing '94*, pp.600–610 (1994).  
 [9] Xu, Z., Zhang, X. and Sun, L.: Semi-empirical multiprocessor performance predictions, *J. Parallel Distrib. Comput.*, Vol.39, pp.14–28 (1996).  
 [10] Xu, Z. and Hwang, K.: Early prediction of MPP performance: The SP2, T3D, and Paragon experiences, *Parallel Comput.*, Vol.22, pp.917–942 (1996).  
 [11] Clement, J.M. and Quinn, J.M.: Automated performance prediction for scalable parallel computing, *Parallel Comput.*, Vol.23, pp.1405–1420 (1997).  
 [12] 古市実裕, 永松礼夫, 出口光一郎: 高並列計算機の性能評価のための挙動予測モデル, 情報処理学会論文誌, Vol.38, No.9, pp.1736–1744 (1997).  
 [13] 折居茂夫: 数値計算のための並列計算機性能評価方法, 情報処理学会論文誌, Vol.39, No.3, pp.529–541 (1998).  
 [14] Kishimoto, Y. and Ichikawa, S.: Optimizing the configuration of a heterogeneous cluster with multiprocessing and execution-time estimation, *Parallel Comput.*, Vol.31, pp.697–710 (2005).  
 [15] Barnes, J.B., Rountree, B., Lowenthal, K.D., Reeves, J., de Supinski, B. and Schulz, M.: A regression-based approach to scalability prediction, *Proc. 22nd Annual International Conference on Supercomputing (ICS 2008)*, pp.368–377 (2008).  
 [16] Mathis, M.M., Kerbyson, J.D. and Hoisie, A.: A performance model of non-deterministic particle transport on large-scale systems, *Future Generation Comput. Systems*, Vol.22, pp.324–335 (2006).  
 [17] Raghavachar, K., Mahinthakumar, K., Worley, P., Zechman, E. and Ranjithan, R.: Parallel performance modeling using a genetic programming-based error correction procedure, *SIMULATION*, Vol.83, No.7, pp.515–527 (2007).  
 [18] Gustafson, L.J.: Re-evaluating Amdahl's Law, *CACM*, Vol.31, No.5, pp.532–533 (1988).  
 [19] Lawson, L.C. and Hanson, J.R.: Solving Least Squares Problems, Prentice-Hall, Chapter 23, p.160 (1974).  
 [20] 折居茂夫, 山本義郎: 限量記号消去法を用いた回帰モデルの予測力向上, 情報処理学会研究報告, 2013-HPC-139, No.1 (2013).  
 [21] 赤池弘次, 甘利俊一, 北川源四郎, 樺島祥介, 下平英寿: 赤池情報量規準 AIC—モデリング・予測・知識発見, 共立出版 (2007).  
 [22] Mathematica, available from (<http://reference.wolfram.com/mathematica/guide/Mathematica.en.html>).  
 [23] R, available from (<http://cran.r-project.org/>).  
 [24] MATLAB, available from (<http://www.mathworks.co>

jp/jp/help/matlab/ref/lsqnonneg.html).

[25] Watanabe, T. and Kaburaki, H.: Increase in chaotic motions of atoms in a large-scale self-organized motion, *Phys. Rev. E*, Vol.54, pp.1504-1509 (1996).

[26] 折居茂夫: レベル 1・2 並列ベンチマーク使用及びそれに基づくスカラ並列計算機 SP2 のベンチマークテスト, JAERI-Data/Code 98-020 (1998).

[27] Petitet, A., Whaley, R.C., Dongarra, J. and Cleary, A.: HPL: A Portable Implementation of the High-Performance Linpack Benchmark for Distributed-Memory Computers, available from (<http://www.netlib.org/benchmark/hpl/>).

[28] Matsuo, Y., Tsuchiya, M., Aoki, M., Sueyasu, N., Inari, T. and Yazawa, K.: Early Experience with Aerospace CFD at JAXA on the Fujitsu PRIMEPOWER HPC2500, *Proc. SC2004* (2004).

## 付 録

### 付録 A.1 ホットスポットカーネルの処理時間モデル [26]

$$T_{force} = n_{sample} I_s t_p$$

$$t_u = t_{0u270} + n \left( t_{0u271} + \frac{n}{2} C_1 \frac{5 + 13C_2}{r_a \cdot a_{u271}} \right)$$

$$\sigma_{270} = 2n \left( 2 \left( t_{0c270p} + \frac{X_8}{r_c e_{c270}} \right) + t_{0sum270} \right)$$

$$t_p = t_{0p270} n + \frac{t_u}{pe_{p270}} + \sigma_{270}$$


---


$$T_{table} = n_{sample} I_s / n_{table} t_p$$

$$t_u = t_{0u101} + n \left( t_{0u100} + \frac{n}{2} \frac{5 + C_1}{r_a a_{u100}} \right)$$

$$t_p = t_{0p100} n^2 + \frac{t_u}{pe_{p101}}$$


---


$$T_{prepcel} = n_{sample} I_s t_p$$

$$t_{u302} = t_{0u302} + \frac{10n}{r_a a_{u302}}$$

$$t_{u303} = t_{0u303} + \frac{7n}{r_a a_{u303}}$$

$$t_{p302} = t_{0p302} + t_{u302} / (pe_{p302})$$

$$t_{p303} = t_{0p303} + t_{u303} / (pe_{p303})$$

$$t_p = t_{p302} + t_{p303} + \sigma_{303}$$

$$\sigma_{303} = n_x n_y \left( 2 \left( t_{0c303p} + \frac{X_8}{r_c e_{c303}} \right) + t_{0sum303} \right)$$


---


$$T_{pcross} = n_{sample} I_s t_u, \quad t_u = t_{0u10} + \frac{10n}{r_a a_{u10}}$$


---


$$T_{moves} = n_{sample} I_s t_u, \quad t_u = t_{0u300} + \frac{9n}{r_a a_{u300}}$$

$\alpha_{cut} = 4, r_{cut} = 3, n_{sample} = 6, I_s = 2000, n_{table} = 14.16\alpha_{cut}r_{cut}, n_x = 40, n_y = 20, X_8 = 8, X_4 = 4, C_1 = \pi(\alpha_{cut}r_{cut})^2 / \text{高さ} \times \text{幅} = \pi(\alpha_{cut}r_{cut})^2 / (2.5n), C_2 = 1/\alpha_{cut}, C_3 = 1/I_s, C_4 = 0.5$  ( $x$  方向で粒子速度が正の確

率),  $C_5 = 0.5$  ( $y$  方向で粒子速度が正の確率),  $C_6 \sim 1$  (境界に接しない粒子の確率)

### 付録 A.2 モデルパラメータとカーネル係数の関係

( $\times n_{sample} I_s$  秒)

$a_0$	$t_{0p302} + t_{0p303} + t_{0u10} + t_{0u300} + n_x n_y t_{0sum303} + \frac{16n_x n_y}{r_c e_{c303}}$
$a_1$	$\frac{1}{n_{table}} (t_{0p101})$
$a_2$	$\frac{1}{n_{table}} \left( \frac{5}{2a_{u100} r_a e_{p101}} \right)$
$a_3$	$t_{0p270} + 2t_{0sum270} + \frac{10}{a_{u10} r_a} + \frac{9}{a_{u300} r_a} + \frac{32}{r_c e_{c270}}$
$a_4$	$\frac{1}{n_{table}} \left( \frac{t_{0u100} + \frac{90.48}{a_{u100} r_a}}{e_{p101}} \right) + \frac{1}{r_a} \left( \frac{10}{a_{u302} e_{p302}} + \frac{7}{a_{u303} e_{p303}} \right) + \frac{t_{0u271} + \frac{525.9}{a_{u271} r_a}}{e_{p270}}$
$a_5$	$4t_{0c270}$
$a_6$	$\frac{1}{n_{table}} \left( \frac{t_{0u101}}{e_{p101}} \right) + \frac{t_{0u270}}{e_{p270}} + \frac{t_{0u302}}{e_{p302}} + \frac{t_{0u303}}{e_{p303}}$
$a_7$	$2t_{0c303}$



折居 茂夫 (正会員)

昭和 56 年 3 月東海大学大学院理学研究科博士課程退学。東海大学非常勤講師、日本原子力研究所核融合特別研究生を経て、昭和 58 年 4 月に富士通株式会社に入社、スーパーコンピュータで実行するアプリケーションプログラムの処理速度向上と性能評価研究等に従事し、平成 24 年 6 月退職、現在に至る。日本物理学会、日本計算工学会の会員。



山本 義郎 (正会員)

平成 10 年 3 月岡山大学大学院自然科学研究科博士後期課程修了。博士(理学)。北海道大学大学院工学研究科助手, 多摩大学経営情報学部助教授を経て, 現在, 東海大学理学部数学科教授。計算機統計学, データマイニング, 統計データの視覚化の研究に従事。日本計算機統計学会, 日本統計学会等の会員。