Vol.2012-HPC-133 No.4
2012/3/26

# 時間モデルを用いた並列性能予測の誤差を検討する一方法

折居茂夫† 穴井宏和††

小規模データの処理時間を用いて構築した時間モデルの予測誤差を検討する方法を提案する．時間モデルと測定時間の残差程度の大きさの閾値を設け，この閾値に相当するモデルパラメータの変化分から予測誤差を推定する．予測誤差の推定時に出現する連立不等式を数式処理のアルゴリズムの一つ限量記号消去法により簡単に解く方法を提案する．時間モデルとそれを構築するのに使用した時間データがあれば，モデル化方法によらず本提案が適用できることを例示する．またこの方法が持つ最適化の機能を使い，時間モデルの精度向上ができる場合があることを例示する．本方法では，フィッティングデータの取捨選択による予測誤差縮減の定量的把握が可能となる．

# An Approach to Estimate Prediction Error of Parallel Performance for Run Time Models

SHIGEO ORII† HIROKAZU ANAI††

We propose an approach to estimate prediction error of parallel performance for run time models obtained from small-scale input data. The prediction error is estimated from a threshold selected from the range of residual between the model and measured data. We also propose a technique to solve simultaneous inequalities emerged from the estimation. The simultaneous inequalities can be easily solved with one of symbolic computation algorithms, Quantifier Elimination. The approach can be applied to a set of a run time model and measured times regardless of modeling method. An example indicates that optimization ability of the approach will refine run time models. The method enables us to grasp reduction of the prediction error quantitatively by the choice of fitting data.

## 1. はじめに

ペタフロプスの計算能力を持つ数万台規模のプロセッサを用いた並列計算機が実用化され，高並列処理が日常的な時代になった．このような処理では，少しの並列化阻害要因が並列性能を大きく下げる．そこで並列化阻害要因の影響を解析するための高並列処理に適応した並列効率メトリックを提案した[1]．また並列処理システムのプロセッサ数と問題の大きさに関する性能評価を行うため，並列処理時間から時間モデルを構築する方法を提案してきた[2, 3, 4].

時間モデルを構築すると，並列化阻害要因の影響を系統的に解析することができる．また時間モデルには，小規模データ，即ち小規模なプロセッサ数と問題の大きさでモデルを構築し大規模問題に外挿することによる，性能予測が期待される．

構築した時間モデルが本当に性能予測に使えるかを調べるには，小規模データで時間モデルを構築し大規模データの処理時間が予測できることを確認する，validation test を行う．しかし一般の並列処理においてオーダーの異なる小規模データと大規模データを準備して validation test を行うことは容易でない．

ここで「工学シミュレーションの品質マネジメント」の validation の定義[5]に基づいて validation test における確認内容をまとめると次の 2 つから成ると考えられる．

1) 並列システムの相似性の確認（小規模データにおける処理と大規模データの処理が，プロセッサ数と問題の規模を変数とするある関数で表せることの確認.）
2) 相似性があるとして，使用に耐える精度で外挿できるかの確認.

1)の相似性については，プロセッサ数と問題の大きさに依存して計算や通信の方法が変わる場合があり，そのような場合は大規模問題における処理時間による確認が必要と考えられる．一方 2)についてはモデルを構築した小規模データとモデルの残差に相当するモデルパラメータの変化分が予測に影響を及ぼすととらえることで，予測誤差を推定できると考えられる．

そこで本研究では上記 1)の相似性は保たれていると仮定し，時間モデルと残差からモデルパラメータの変化を求め予測誤差を検討する方法を提案する．またこの方法で用いる連立不等式を数式処理の解法の一つ限量記号消去法(Quantifier Elimination, 以後 QE と記す)で解くことを提案する．数式処理 QE を用いる利点は，解が等式，不等式の形で得られることにある．よってそれらの式で表された範囲として予測誤差の検討を行うことができる．

2 章では時間モデルと残差から予測誤差を検討するベーシックアイディアを提案する．3 章ではベーシックアイディアを定式化する．4 章では適用事例を示す．5 章でまとめと議論を行う．

† 富士通（株） †† (株) 富士通研究所／九州大学
Fujitsu Ltd. FUJITSU LABORATORIES Ltd./ Kyushu University

## 2. ベーシックアイディア

プロセッサ数 $p$ を変えて並列処理の時間を測定した $m$ 個の測定値 $di(i=1,2,\ldots,m)$を図 1 (a)の×印で示す．これを関数 $T(c, p)$で表わせるものとし，最小自乗法でモデルパラメータ $c$ の値 C を求めると $T(\mathrm{C}, p)$を得る．$T(\mathrm{C}, p)$と測定値 $di$ の残差 $ei$（図中水色の双方向矢印）はモデルがこの位の大きさの誤差を持つことを示唆し，外挿領域でもこの誤差に起因した予測値の変動があると考えられる.そこで全ての残差がある閾値 $e'$ (≦ $Max(e_i)$)以下の場合の「予測誤差を含む時間モデル $T(\mathrm{C}+\triangle c, p)$」を「時間モデル $T(\mathrm{C}, p)$」から求め，これら 2 つのモデルを図 1 (b)のように外挿して比較することにより，予測誤差を検討することを提案する.

この$\triangle c$ は $T(\mathrm{C}+\triangle c,\ p)\text{-}di \leqq e'$から求める．この不等式は数式処理のアルゴリズムの一つ限量記号消去(Quantifier Elimination: QE)を用いると容易に解くことができるのでこれを用いることを提案する.

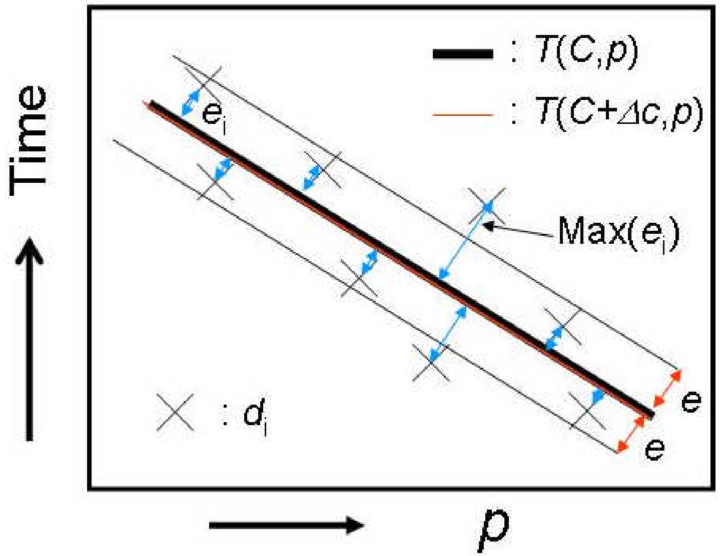


図 1(a)　時間モデル $T(\mathrm{C}, p)$及び
$T(\mathrm{C}, p)$とある $e'$に対する定時間モデル $T(\mathrm{C}+\triangle c, p)$

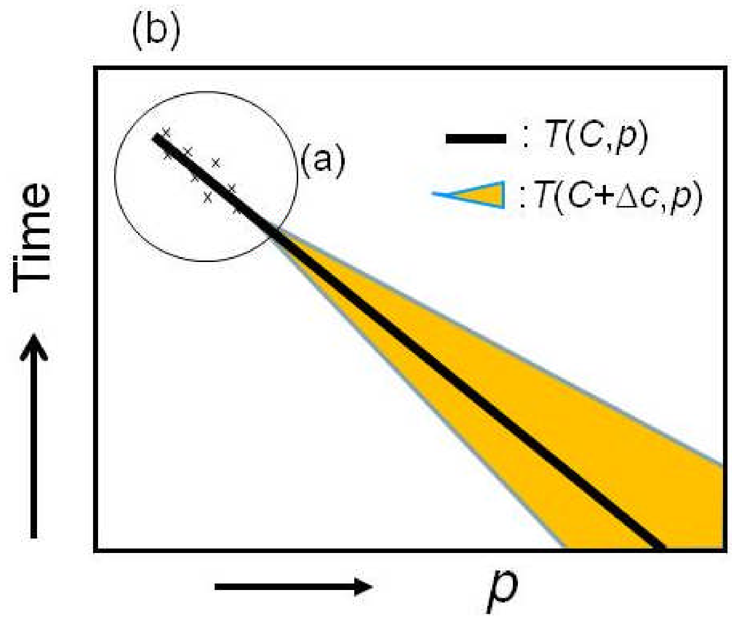


図 1(b)　外挿領域の $T(\mathrm{C},p)$と $T(\mathrm{C}+\triangle c, p)$

## 3. 予測誤差を含む時間モデルの導出方法

２章で述べた予測誤差を含む時間モデル $T(\mathrm{C}+\triangle c, p)$を導出する方法，即ち指定した $e'$に対する$\triangle c$ を求める方法を示す．まず $i=1,2,\ldots,m$ 個の測定時間 $d_i$で，$j$ 個のパラメータがある式(1)に最小自乗法を適用して「時間モデル」を決定する場合を考える．ここに $c_j$ は最小自乗法により求める未知パラメータである.

$$T(c_j, p) - d_i = e_i \tag{1}$$

求めたモデルパラメータ $\mathrm{C}_j$ を用いて，式(1)の残差 $e_i$ をある $e$ にした「予測誤差を含む時間モデル」を求める式を式(2)に表す．ここに $p_i$,はデータ $i$ のプロセッサ数 $p$ の値である．$m$ 個あった $e_i$ を，ある $e$ で代表した結果，式(2)は不等式となる．ここに $\mathrm{C}_j$ は最小自乗法により求めたパラメータ値である.

$$\left|T(C_j + \Delta c_j, p_i) - d_i\right| \le e \tag{2}$$

この連立不等式が成立する $e$ の下限がある．一般に $Min(e_i)$より大きいこの下限を本研究では $e$max と表記する．$e$max は式(2)が解を持つ最小の値であり，$e\max \le e \le Max(e_i)$ である．$e= Max(e_i)$とすることにより得た$\angle c_j$で，最も大きな予測誤差を検討できると考える．$e$ が小さいほど$\angle c_j$は小さく，モデルの予測誤差も小さく，$e$=$e$max として得た$\angle c_j$は式(2)が解を持つ最も小さな予測誤差となる．

式(2)から$\angle c_j$を求めることは，式(3)のように論理式を用いて記述することができる．ここに∃は限定記号である．⇒は限定記号で示した変数・パラメータを括弧内の論理式から消去する QE のオペレーション(以後 QE と記述する)を意味する．この式では $e$ にある閾値 $e'$を設定して $e$ を QE で消去し，パラメータ間の関係式 $F(\angle c_j)$が得られることを示す．

$$\bigwedge_{i=1}^{m} \exists e(-e \le T(\mathrm{C}_j + \Delta c_j, p_i, x_i) - D_i \le e \wedge e = e') \overset{\mathrm{QE}}{\Rightarrow} F(\Delta c_j) \tag{3}$$

## 4. 時間モデルの予測誤差

式(3)における $e'$を $Max(e_i)$と $e$max とし，これらに対する$\angle c_j$を計算して $T(\mathrm{C}_j+\angle c_j, p)$を求め，時間モデル $T(\mathrm{C}_j, p)$と比較して時間モデルの予測誤差を検討した結果を示す．

### 4.1 ブラックボックスアプローチで導出した時間モデルの予測誤差

式(4)はブラックボックスアプローチ BBA [4]で求めた，High Performance Linpack と HPC2500 からなる並列処理システムの時間モデルで，$a$=26002，$\mathrm{C}_1$= 0.0088823，$\mathrm{C}_2$= $1.9312\times10^{-7}$ である．

$$T(p) = a \cdot \left( \frac{1}{p} + \mathrm{C}_1 + \mathrm{C}_2 \cdot (p-1)^2 \right) \qquad \text{式(4)}$$

#### (1) 最大残差 $Max(e_i)$に対する予測誤差

式(3)で $j$=2，閾値 $e'=Max(e_i)$としたときの予測誤差を含む時間モデル $T(\mathrm{C}_1+\angle c_1, \mathrm{C}_2+\angle c_2, p)$の$\angle c_1$と$\angle c_2$を求めることは，論理式を用いて式（5）となる．

$$\bigwedge_{i=1}^{12} \exists e\left(-e \le T(\mathrm{C}_1 + \Delta c_1, \mathrm{C}_2 + \Delta c_2, p_i) - d_i \le e \wedge e = Max(e_i)\right) \overset{\mathrm{QE}}{\Rightarrow} F(\Delta c_1, \Delta c_2) \qquad \text{式(5)}$$

ここに $d_i$ は $p_i$={10,20,30,40,50,60,70,80,90,100,110,120}に対する測定時間である．QE の計算は数式処理ツール Mathematica を用い，式(5)をその関数を用いて記述すると，図２(a)となる．∃は関数 Exists で記述し，∃で指定した変数やパラメータを消去する QE は関数 Reduce を用い，∧は&&で記述した．処理は代数演算のため，測定値 $d_i$ と最大残差は Rationalize で有理数に変換する．図 2(b)は QE の実行結果である．本来この出力も有理数で表現されるが，関数//N を用いて実数表示とした．なお図中 dc1: $\angle c_1$，dc2: $\angle c_2$，である．

```
C1=Rationalize[0.0088823,0.00000001]
C2=Rationalize[1.9312*10^-7,1*10^-12]
a=26022
maxe=Rationalize[17.9745];
Reduce[Exists[{e},
    -e ≤ 14244/5-a   *(1/10+C1+dc1+(C2+dc2)*(10-1)^2)    ≤ e  &&
    -e ≤ 15477/10-a  *(1/20+C1+dc1+(C2+dc2)*(20-1)^2)    ≤ e  &&
    -e ≤ 5563/5-a    *(1/30+C1+dc1+(C2+dc2)*(30-1)^2)    ≤ e  &&
    -e ≤ 89709/100-a*(1/40+C1+dc1+(C2+dc2)*(40-1)^2)     ≤ e  &&
    -e ≤ 76531/100-a*(1/50+C1+dc1+(C2+dc2)*(50-1)^2)     ≤ e  &&
    -e ≤ 68499/100-a*(1/60+C1+dc1+(C2+dc2)*(60-1)^2)     ≤ e  &&
    -e ≤ 31219/50-a  *(1/70+C1+dc1+(C2+dc2)*(70-1)^2)    ≤ e  &&
    -e ≤ 2913/5-a    *(1/80+C1+dc1+(C2+dc2)*(80-1)^2)    ≤ e  &&
    -e ≤ 13892/25-a  *(1/90+C1+dc1+(C2+dc2)*(90-1)^2)    ≤ e  &&
    -e ≤ 13273/25-a  *(1/100+C1+dc1+(C2+dc2)*(100-1)^2) ≤ e  &&
    -e ≤ 27269/50-a  *(1/110+C1+dc1+(C2+dc2)*(110-1)^2) ≤ e  &&
    -e ≤ 10269/20-a  *(1/120+C1+dc1+(C2+dc2)*(120-1)^2) ≤ e  &&
e==maxe]]//N
```

図２(a)　QE により最大残差 $Max(e_i)$に対する$\angle c_1$，$\angle c_2$ の値を求めるプログラム

```
(7.96273*10^-8+dc2==0&&dc1==0.000946051)||
(dc2==4.14635*10^-8&&0.000115456+dc1==0)||
(1.22436*10^-10+dc1+11881.dc2≥0&&((-7.96273*10^-8<dc2≤-6.84759*10^-8&&dc1+2401
.dc2≤0.000754866)||(-6.84759*10^-8<dc2≤-4.87413*10^-8&&dc1+6241.dc2≤0.000491919)||
(-4.87413*10^-8<dc2≤9.49981*10^-9&&dc1+9801.dc2≤0.0003184)))||
(0.000112098+dc1+81.dc2≥0&&((dc2>9.49981*10^-9&&dc1+9801.dc2≤0.0003184&&dc2≤
3.51625*10^-8)||(dc2>3.51625*10^-8&&dc2<4.14635*10^-8&&dc1+14161.dc2≤0.00047170
8)))
```

図 2(b)　QE の実行結果 $F(\angle c_1, \angle c_2)$

図２(b)の QE の実行結果は，$\Delta c_1$ と$\Delta c_2$ の関係式は等式と不等式で表現される．これらの関係を可視化すると図３となり，解が存在する領域は線分で囲まれた部分となる．図２(b)の１行目の式は図３左上の■である．図２(b)の２行目の式は図３の右下の■である．図２(b)の３から５行目及び，６から８行目は，各々黒の実線で囲まれた範囲，破線で囲まれた範囲に対応する．そこで左上と右下の■の$\Delta c_1$ と$\Delta c_2$ を用いて予測誤差の上限と下限を検討する．

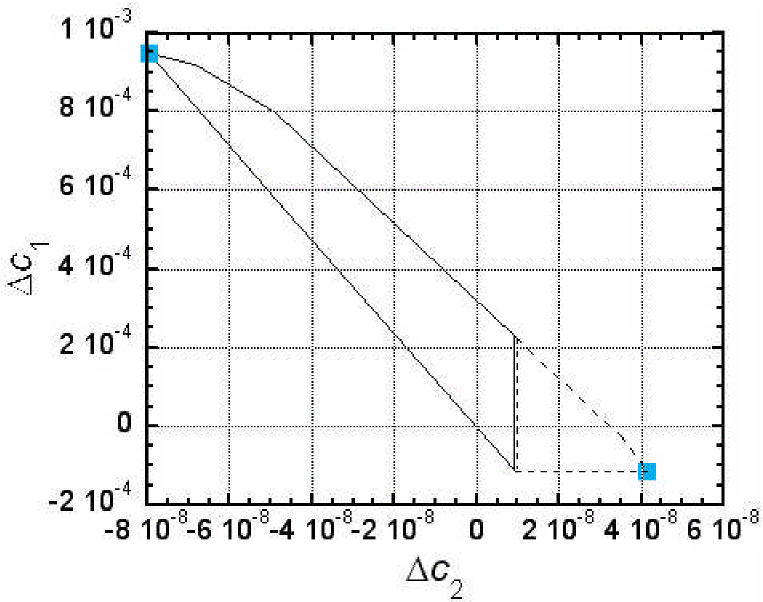


図３　QE 実行結果 $F(\Delta c_1, \Delta c_2)$の可視化

図４は時間モデルと時間モデルを外挿し，予測誤差を含む時間モデルと比較した図である．この図において，黒の実線は時間モデル式(4)である．青線は図３の左上の■を$\Delta c_1$,と$\Delta c_2$ に用いたもので，予測誤差の下限である．赤線は図３の右下の■を用いたもので，予測誤差の上限である．式(5)の解は図３の線で囲まれた範囲なので，図４の予測誤差範囲はこれら赤と青が囲む範囲である．この範囲が最大残差 $Max(e_i)$から推定した最大の予測誤差である．

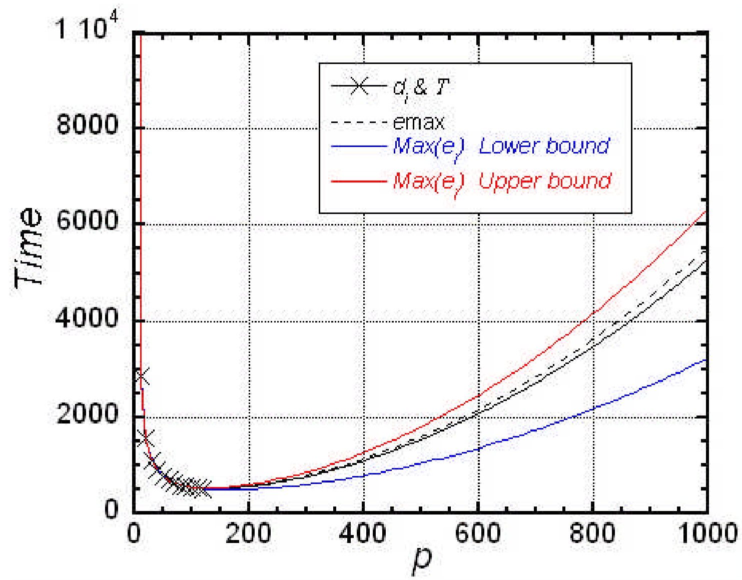


図４　時間モデルの予測と残差 $Max(e_i)$と emax から推定した予測誤差

(2)　**式(5)に解がある最小残差 *emax* に対する予測誤差**

次に連立不等式が成り立つ一番小さな残差値 *emax* を閾値としたときの予測誤差を検討する．*emax* は式（6）の QE 計算で得ることができる[6]．この QE は Mathematica では図５のように記述でき，実行結果として emax≥ 1101248/81125 を得る．

$$\bigwedge_{i=1}^{12} \exists \Delta c_1 \exists \Delta c_2 \left( -e\max \le T(C_1+\Delta c_1, C_2+\Delta c_2, p) - d_i \le e\max \right) \overset{\text{QE}}{\Rightarrow} F(\Delta c_1, \Delta c_2) \quad \text{式(6)}$$

```
c1=Rationalize[0.0088823,0.00000001]
c2=Rationalize[1.9312*10^-7,1*10^-12]
a=26022
Reduce[Exists[{dc1,dc2},
     -emax ≤ 14244/5-    a*(1/10+C1+dc1+(C2+dc2)*(10-1)^2)    ≤ emax    &&
     -emax ≤ 15477/10-   a*(1/20+C1+dc1+(C2+dc2)*(20-1)^2)    ≤ emax    &&
     -emax ≤ 5563/5-     a*(1/30+C1+dc1+(C2+dc2)*(30-1)^2)    ≤ emax    &&
```

-emax ≤ 89709/100- a*(1/40+C1+dc1+(C2+dc2)*(40-1)^2) ≤ emax &&
-emax ≤ 76531/100- a*(1/50+C1+dc1+(C2+dc2)*(50-1)^2) ≤ emax &&
-emax ≤ 68499/100- a*(1/60+C1+dc1+(C2+dc2)*(60-1)^2) ≤ emax &&
-emax ≤ 31219/50- a*(1/70+C1+dc1+(C2+dc2)*(70-1)^2) ≤ emax &&
-emax ≤ 2913/5- a*(1/80+C1+dc1+(C2+dc2)*(80-1)^2) ≤ emax &&
-emax ≤ 13892/25- a*(1/90+C1+dc1+(C2+dc2)*(90-1)^2) ≤ emax &&
-emax ≤ 13273/25- a*(1/100+C1+dc1+(C2+dc2)*(100-1)^2) ≤ emax &&
-emax ≤ 27269/50- a*(1/110+C1+dc1+(C2+dc2)*(110-1)^2) ≤ emax &&
-emax ≤ 10269/20- a*(1/120+C1+dc1+(C2+dc2)*(120-1)^2) ≤ emax
]]

図５　式(5) に解がある最小残差値 *emax* を求める QE プログラム

次に得られた *e*=*emax* とした式(7)を解くことにより，閾値 *emax* に対する$\triangle c_1$と$\triangle c_2$即ち dc1= 0.000056212, dc2=9.4998×$10^{-9}$ を得る．これに対する予測誤差を含む時間モデルを図４の破線で示す．破線は式(3)のように問題を定式化したときの最小誤差である．

$$\bigwedge_{i=1}^{12} \exists e\left(-e \le T(C_1+\Delta c_1, C_2+\Delta c_1, p)-d_i \le e \wedge e = e\max\right) \overset{\mathrm{QE}}{\Rightarrow} F(\Delta c_1, \Delta c_2)$$

式(7)

### 4.2　ホワイトボックスアプローチで導出した時間モデルの予測誤差

提案した予測誤差の検討方法は，モデル式と測定データがあれば，モデリングの方法に関係なく適用することができる. 式(8)は MD コードと SP2 からなる並列システムで実行したときのホットスポットカーネルをモデル化した，即ちホワイトボックスアプローチ(WBA)で導出した時間モデルである[7].

Msnap × mtime × ((10 × *n*)/(au10 × ra)+t0u10)+msnap × mtime × ((9 × *n*)/(au300 × ra)+t0u300) + msnap × mtime × (2 × ((8 × nxy × (*p*-1))/(ec303 × *p* × rc)+*p* × t0c303)+t0p302+t0p303+t0sum303+((10 × *n*)/(au302 × ra)+t0u302) × ep302/*p*+((7 × *n*)/(au303 × ra)+t0u303) × ep303/*p*) + mtable × (((n-1) × (((5+57.6/*n*) × *n*)/(2 × au100 ra)+t0u100)+t0u101) × ep101/*p*+t0p101[*n*]) + (1+msnap × mtime) × ((t0u270+(*n*-1) (525.9/(au271 × ra)+t0u271)) × ep270/*p*+2 × (t0sum+2 × ((8 × *n* × (*p*-1))/(ec270 × *p* × rc)+*p* t0c270[*n*]))+t0p270[*n*])

式(8)

式中，*p* はプロセッサ数，*n* は問題の大きさを表す粒子数，その他はアプリケーションプログラム，並列計算機，入力値に関連したモデルパラメータである．この式に文献[7]に記載したモデルパラメータを代入し，*n*=96800 とすると時間モデルは式(9)となる．ここに a=54642，$C_1$=0.05922，$C_2$=0.0007528 である．

$$T(p) = a \cdot \left(\frac{1}{p} + \mathrm{C}_1 + \mathrm{C}_2 \cdot p\right)$$

式(9)

式(9)を式(5)，(6)，(7)に代入すると，4.1 節の図４と同様な図６を得る．

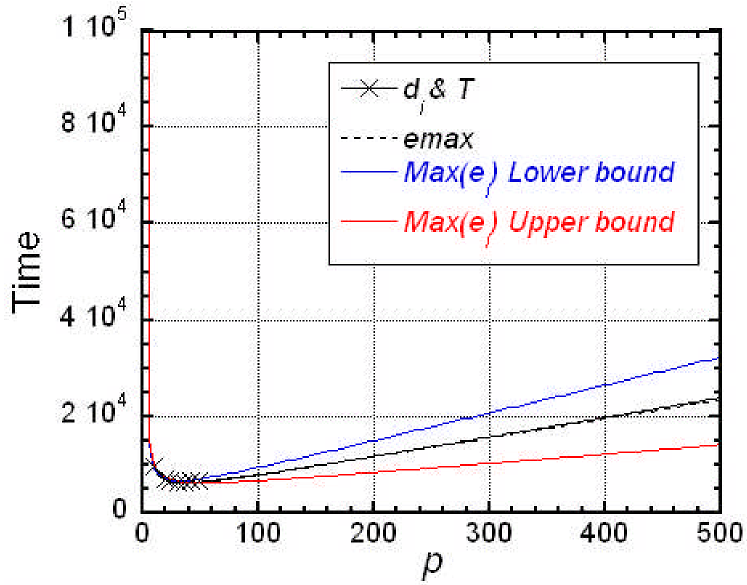


図６　WBA で導出した時間モデルの予測誤差

### 4.3　時間モデルの精度向上（最適化）

4.2 節で示した時間モデルを測定値のレンジでプロットすると，図７(a)のように測定×に対し実線で示した時間モデルは小さくモデル化されていることがわかる．これはモデル化したホットスポットの処理時間が実アプリケーションプログラム全体の処理時間に足りていないためである．図はまた式(5)，(7)に式(9)を代入して得た T($C_1$+$\triangle c_1$, $C_2$+$\triangle c_2$)の点線が測定点上に乗っていることを示す．残差の二乗平均平方根で実線と点線比較すると，各々268.1 と 110.0 である．そこでこの T($C_1$+$\triangle c_1$ と $C_2$+$\triangle c_2$ , *p*)を時間モデルとすると，*Max*($e_i$)に対する予測誤差は図７(b)となり，図６と比べると予測精度が向上していることがわかる．

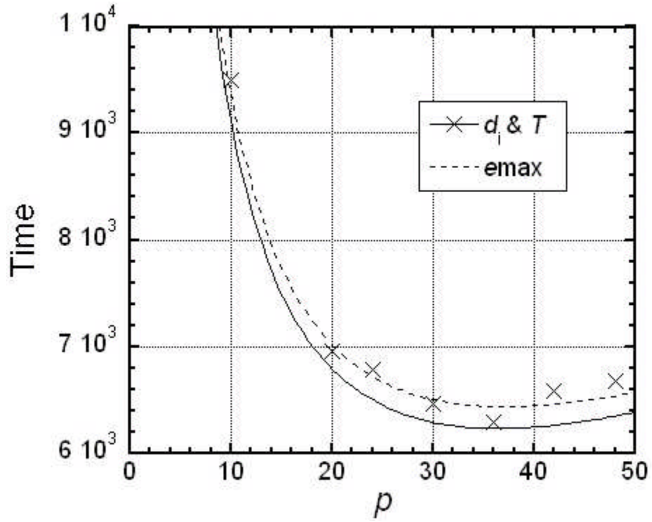


図 7 (a)　 WBA で導出した時間モデルの誤差

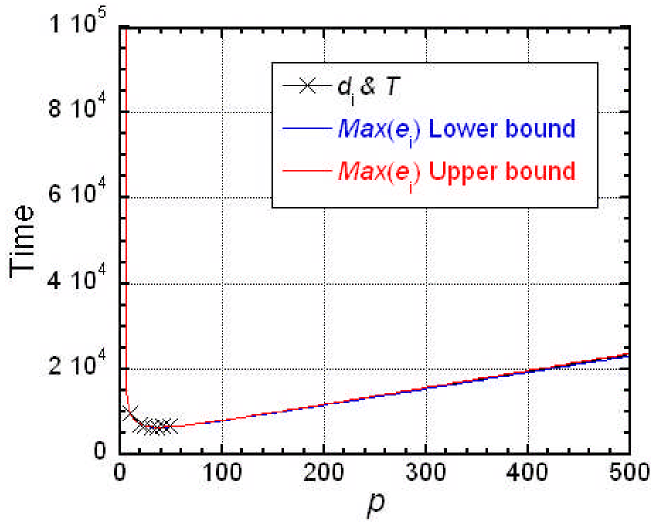


図 7 (b)　 再最適化した時間モデルの予測誤差

### 4.4　時間モデルの精度向上（予測誤差の縮減の定量的検討）

提案した予測誤差検討方法により，フィッティングする測定データを取捨選択して縮減した予測誤差を，定量的に検討できることを示す．

4.1 節(1)の BBA 時間モデル式(4)は, モデル式と時間をフィッティングして構築するのではなく，式(10)のように並列オーバーヘッド $X$ にプロセッサ数 $p$ をかけた式でフィッティングして構築する[4]．具体的には左辺を測定値から決定し，右辺のモデル式とフィッティングする．このフィッティングの残差を測定データに対してプロットしたものを図８に示す．図は残差が p=110 において突出していることを示す．そこでこの測定値を削除して時間モデルを作り，式(11)で$\triangle c_1$ と$\triangle c_2$ を求めその予測誤差を検討した．

$$\frac{1-\varepsilon'_p}{\varepsilon'_p} \equiv \frac{1}{a}(p \cdot \mathrm{X}) \tag{10}$$

ここに $\varepsilon'_p = \dfrac{a}{p \cdot \tau}$，　$X = a \cdot \left( \dfrac{c_0}{p} + |c_1| + \sum_{j=2} c_j (p-1)^{j-1} \right)$ である．

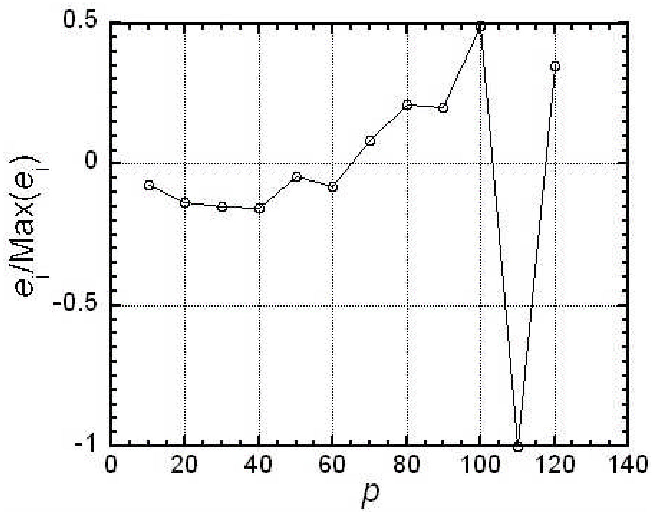


図８　 フィッティング残差

$$LHS(\Delta c_1, \Delta c_2, p_i) := -e \le T(C_1 + \Delta c_1, C_2 + \Delta c_2, p_i) - d_i \le e \wedge e = Max(e_i)$$

$$\bigwedge_{i=1}^{12} \exists e\left(\neg LHS(\Delta c_1, \Delta c_2, p=110) \wedge LHS(\Delta c_1, \Delta c_2, p_i)\right) \overset{\mathrm{QE}}{\Rightarrow} F(\Delta c_1, \Delta c_2) \tag{11}$$

$p$=110 の値を除去すると，式(4)のモデルパラメータは $C_1$=0.008981，$C_2$=1.630×$10^{-7}$ となる．この時間モデルと残差指定時間モデルを図 9 に示す．スケールが同じ図 4 と比較すると，閾値を最大残差 $Max(e_i)$に対する誤差の範囲が顕著に縮減されており，例えば 2 つの図の $p$=1000 の誤差を比較すると，時間モデルの利用の可否が変わるほどの誤差の縮減が達成されている．

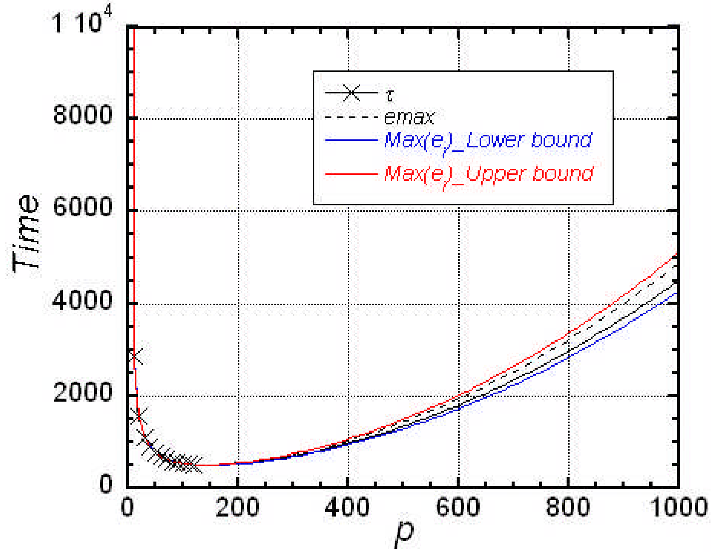


図 9　突出した残差($p$=110)の測定値を削除して構築した時間モデルの予測誤差

## 5. まとめと議論

式で記述された時間モデルと測定値の残差が，ある閾値内に入ると仮定して予測誤差を定量的に検討する方法を提案した．この方法で出現する連立不等式を，数式処理の一アルゴリズム，限量記号消去で解くことを提案した．数式として得られる解により，予測誤差を定量的に検討することができることを示した．この方法が時間モデルのモデリング方法によらず適用できることを，ホワイトボックスとブラックボックスアプローチで導出した時間モデルを用いて示した．この方法で得られる閾値 emax を使って時間モデルを再最適化できる場合があることを示した．またフィッティングデータの取捨選択により縮減した予測誤差を定量的に把握できることを示した．

4.3 節で述べた最適化ができるか否かの判断は図 7 (a)の測定値，時間モデル，閾値を emax と指定した時間モデルを比較して行ったが，定量的目安として残差の二乗平均平方根 RMS と emax の値の比較が考えられる．即ち RMS> emax であれば最適化可能．RMS< emax であれば不可．4.3 節では RMS=268.1，emax=141.7 で前者．4.1 節は RMS=9.485，emax=13.57 で後者である．

本論文では 1 章で述べた「並列システムの相似性の確認」ができたものと仮定した．実際は並列計算機やアプリケーションプログラムが変わるごとにこの確認を実施しなくても済む工夫が必要である．並列計算機に起因する部分の確認はそれようのベンチマークテストを工夫することが考えられるが，アプリケーションに起因する部分も含めて今後の研究課題と考える．

**謝辞**　富士通株式会社テクニカルコンピューティング･ソリューション事業本部の奥田基エグゼクティブアーキテクト，樋口哲二シニアマネージャの援助に感謝します．本研究は科研費（21340025）の助成を受けたものである．

## 参考文献

1) S. Orii: Metrics for evaluation of parallel efficiency toward highly parallel processing, Parallel Comput, 36 (2010) 16-25.
2) 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，Vo.39, No.3 (1998).
3) 折居茂夫：高並列処理におけるスケーラビリティ評価方法（Ⅱ），情報処理学会研究報告, 2010-HPC-126, Vol.2010, No.44 (2010).
4) 折居茂夫 : 時間モデルを用いた並列処理の性能評価 - 並列化部に隠れた並列オーバーヘッド-, 情報処理学会研究報告, 2011-HPC-130, Vol.2011, No.1 (2011).
5) 白鳥正樹，他：日本計算工学会標準　工学シミュレーションの品質マネジメント，JSCES S-HQC001 (2011).
6) Shigeo Orii, Hirokazu Anai: Application of Quantifier Eliminator to Symbolic-Numeric Optimization in Biochemical Model, Research Communications in Biochemistry and Cell & Molecular Biology, Vol. 12, No. (1&2) (2008) pp.73-89.
7) 折居茂夫，レベル 1･2 並列ベンチマーク仕様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).