情報処理学会論文誌　Vol. 49　No. 6　2164–2168 (June 2008)

AISM 法を用いた近似逆行列の 2 レベル並列計算

張　臨傑[†1]　森屋　健太郎[†2]　野寺　隆[†3]

大規模な線型問題に対して，2 レベルの近似逆行列計算に AISM 法を利用する新しい並列前処理の手法を提案する．SGI 社の Origin 2400 に MPI を用いて提案手法を実装した数値実験により，近似逆行列の計算は PE16 台の PE 上で最大約 136.72 倍の速度向上が得られた．

Two-level Parallel Computation for Approximate Inverse with AISM Method

Linjie Zhang,[†1] Kentaro Moriya[†2]
and Takashi Nodera[†3]

In this paper, we propose a novel strategy for parallel preconditioning of large scale linear systems by means of a two-level approximate inverse technique with AISM method. According to the numerical results on an origin 2400 by using MPI, the proposed parallel technique of computing the approximate inverse makes the speedup of about 136.72 times with 16 processors.

1. はじめに

理工学における様々な現象を記述した偏微分方程式の境界値問題を，有限要素法や有限差分法によって離散化すると，次のような大型で疎な係数行列を持つ連立 1 次方程式が得られる．

†1 慶應義塾大学大学院理工学研究科
Graduate School of Science and Technology, Keio University
†2 青山学院大学理工学部
Faculty of Science and Technology, Aoyama Gakuin University
†3 慶應義塾大学理工学部
Faculty of Science and Technology, Keio University

$$A\boldsymbol{x} = \boldsymbol{b}, \quad A \in \boldsymbol{R}^{n\times n}, \quad \boldsymbol{x}, \boldsymbol{b} \in \boldsymbol{R}^n \tag{1}$$

通常，式 (1) をクリロフ部分空間法で解く場合には，残差ノルムの収束を改善するために，式 (1) に対する行列の前処理が広く用いられている（Saad[8] [pp.102–103]）．行列の前処理とは，式 (1) に対する次式のような行列変換を行うことである．本稿では，前処理行列 $M \in \boldsymbol{R}^{n\times n}$ を計算し，式 (1) の係数行列 A の右側から掛けて近似解を計算する手法を利用する．

$$AM\boldsymbol{y} = \boldsymbol{b}, \quad \boldsymbol{x} = M\boldsymbol{y} \tag{2}$$

近年，前処理行列 M として行列 A の近似逆行列がよく用いられている．現在，近似逆行列を求める算法は数多く提案されており（e.g. Chow ら[3] [pp.997–1005]），Bru ら[2] の提案した AISM 法は，その中の 1 つの算法である．AISM 法は係数行列 A の近似逆行列を次式で計算する．

$$A^{-1} \approx \alpha^{-1} I_n - \alpha^{-2} U \Omega^{-1} V^T, \quad (\alpha > 0) \tag{3}$$

ただし，$I_n \in \boldsymbol{R}^{n\times n}$ は単位行列であり，行列 U，V の列ベクトル $\boldsymbol{u}_k$，$\boldsymbol{v}_k$ は，式 (4) で計算する．

$$\begin{aligned} \boldsymbol{u}_k &= \boldsymbol{x}_k - \sum_{i=1}^{k-1} \frac{(\boldsymbol{v}_i)_k}{\alpha r_i} \boldsymbol{u}_i, \\ \boldsymbol{v}_k &= \boldsymbol{y}_k - \sum_{i=1}^{k-1} \frac{\boldsymbol{y}_k^T \boldsymbol{u}_i}{\alpha r_i} \boldsymbol{v}_i \end{aligned} \tag{4}$$

また，式 (3) に現れる行列 Ω は対角行列であり，その対角要素は $r_k = 1 + (\boldsymbol{v}_k)_k/\alpha$ である．さらに，Bru ら[2] は，式 (3) の係数 α および式 (4) の $\boldsymbol{x}_k$，$\boldsymbol{y}_k$ について，次のような選択を提案している．

$$\alpha = 1.5\|A\|_\infty, \quad \boldsymbol{x}_k = \boldsymbol{e}_k, \quad \boldsymbol{y}_k = (\boldsymbol{a}^k - \alpha \boldsymbol{e}_k)^T$$

ここで，$\boldsymbol{e}_k \in \boldsymbol{R}^n$ は I_n の k 番目の列ベクトルであり，$\boldsymbol{a}^k$ は行列 A の k 番目の行ベクトルである．一般的にいって，U と V は密行列になる傾向がある．U と V の疎な性質を維持するために，非ゼロ要素を切り捨てる処理を行う必要がある．

AISM 法[2] は，ベクトル $\boldsymbol{u}_k$，$\boldsymbol{v}_k$ の計算過程に依存関係が存在しているため，並列計算に不向きであり．並列計算機環境では利用されないことが多い．この問題を改善するために，森屋ら[6] は，Naik[7] の並列実装法を用いて，部分的な並列計算を可能にした AISM 法を提案した．

近年，近似逆行列を求める算法を並列化するほかに，グラフ分割法[5] や，領域分割法[8] [pp.75–84] を利用して，係数行列を並列計算向きに再構成し，2 レベルの並列計算で近似逆行列を求める解法もよく使われている．この並列手法は，もとの近似逆行列問題をいくつかの小さい近似逆行列問題に変換することである．各々の小さい近似逆行列問題は，互いに独立しているので，それら各々の問題は並列に計算できる．ただし，各小行列の近似逆行列の計算は，各小行列単位で CPU に割り当てて並列化する．この手法は，並列計算機環境で，並列計算に不向きな算法を利用する際によく使われている[1),4)]．

本稿では，2 レベルの近似逆行列計算に AISM 法を用いて，新しい前処理行列の並列計算法を提案する．最後，MPI を用いて SGI 社の Origin 2400 に実装し，提案する手法の有効性を示す．

2. 近似逆行列の 2 レベル並列計算

グラフ分割法[5] を用いて，行列の要素を次のような縁付きブロック対角行列に再構成する．

$$P^TAP = \begin{bmatrix} A_1 & & & B_1 \\ & \ddots & & \vdots \\ & & A_p & B_p \\ C_1 & \cdots & C_p & A_s \end{bmatrix} \tag{5}$$

ただし，P は置換行列である．式 (5) のような縁付きブロック行列の対角行列 A_i は，お互いに独立しているので，もとの行列 A と比べて並列計算に適している．この手法はもともと疎な行列に対するコレスキー分解の並列手法として提案されたが[4)]，Benzi ら[1)] はこの手法を利用して，AINV 法を用いた近似逆行列の 2 レベル並列計算を提案している．

まず最初に，近似逆行列の 2 レベル計算について順をおって述べることにする．本稿では，AISM 法を適用できるように，P^TAP の逆行列を次式のように分解する．

$$(P^TAP)^{-1} = \begin{bmatrix} A_1^{-1} & & & E_1 \\ & \ddots & & \vdots \\ & & A_p^{-1} & E_p \\ & & & S^{-1} \end{bmatrix} \begin{bmatrix} I_1 & & & \\ & \ddots & & \\ & & I_p & \\ F_1 & \cdots & F_p & I_s \end{bmatrix}$$

ただし， $E_i = -A_i^{-1}B_iS^{-1}$，$F_i = -C_iA_i^{-1}$，$S = A_s - \sum_{i=1}^{p} C_iA_i^{-1}B_i$ はシュールコンプリメントであり，A_s と同じ次元を持つ．I_i と I_s はそれぞれ A_i，A_s と同次元の単位行列である．右側の A_i^{-1} と S^{-1} をそれぞれの近似逆行列 M_i と M_s で近似すると，P^TAP の近似逆行列は次式のようになる．

$$(P^TAP)^{-1} \approx \begin{bmatrix} M_1 & & & \bar{E}_1 \\ & \ddots & & \vdots \\ & & M_p & \bar{E}_p \\ & & & M_s \end{bmatrix} \begin{bmatrix} I_1 & & & \\ & \ddots & & \\ & & I_p & \\ \bar{F}_1 & \cdots & \bar{F}_p & I_s \end{bmatrix} \tag{6}$$

ただし，$M_i \approx A_i^{-1}$，$M_s \approx \bar{S}^{-1}$，$\bar{E}_i = -M_iB_iM_s$，$\bar{F}_i = -C_iM_i$ である．ここで，$\bar{S} = A_s - \sum_{i=1}^{p} C_iM_iB_i$ は，近似シュールコンプリメントである．本稿では，$C_iM_iB_i$ を近似シュールコンプリメント $\bar{S}$ のローカルな部分という．

近似逆行列の 2 レベル計算は，式 (6) に基づいて，2 レベルで P^TAP の近似逆行列を計算する．

レベル 1：$M_i \approx A_i^{-1}$（$i = 1, \ldots, p$）を計算する．
レベル 2：$M_s \approx \bar{S}^{-1}$ を計算する．

ただし，$\bar{E}_i$，$\bar{F}_i$ を陽的に計算せず，$\bar{E}_i$，$\bar{F}_i$ をベクトルに掛けるときは，$\bar{E}_i\boldsymbol{x} = -M_i(B_i(M_s\boldsymbol{x}))$，$\bar{F}_i\boldsymbol{x} = -C_i(M_i\boldsymbol{x})$ のように計算する．2 レベル計算の算法を**図 1** に示す．

2 レベル計算では，A_i（$i = 1, \ldots, p$）と $\bar{S}$ の近似逆行列を求めることによって，P^TAP の近似逆行列を計算する．つまり，次元の大きい近似逆行列問題は，$p + 1$ 個の次元の小さい近似逆行列問題に変換することになる．一般的に，2 次元問題の場合，疎な行列の近似逆行列を求めるコストは，ほぼ行列の次元の 2 乗に比例するため，いくつかの次元の小さい近似逆行列を計算することによって，総計算コストを軽減することが期待できる．ただし，

for $i = 1, \ldots, s$
　compute $M_i \approx A_i^{-1}$
　compute $\bar{S}_i = C_iM_iB_i$
end for
compute $\bar{S} = A_s - \sum_{j=1}^{p} \bar{S}_j$
compute $M_s \approx \bar{S}^{-1}$

図 1　近似逆行列の 2 レベル計算法
Fig. 1　Algorithm of the two level computation.

ステップ	PE 1	PE 2	...	PE p
1	compute $M_1 \approx A_1^{-1}$	compute $M_2 \approx A_2^{-1}$	...	compute $M_p \approx A_p^{-1}$
2	compute $\bar{S}_1 = C_1 M_1 B_1$	compute $\bar{S}_2 = C_2 M_2 B_2$	...	compute $\bar{S}_p = C_p M_p B_p$
3	receive $\bar{S}_i, (i = 2, \ldots, p)$ from other PEs	send $\bar{S}_2$ to PE 1	...	send $\bar{S}_p$ to PE 1
4	compute $\bar{S} = A_s - \sum_{j=1}^{p} \bar{S}_j$	wait	...	wait
5	compute $M_s \approx \bar{S}^{-1}$	wait	...	wait
6	send M_s to other PEs	receive M_s from PE1	...	receive M_s from PE1

図 2　2 レベル並列計算を行う様子
Fig. 2　Steps of two level parallel computation.

与えられた行列に対して p が大きくなるにつれ，A_i の次元は小さくなり，A_s の次元は大きくなる傾向がある．したがって，p の増大につれ，A_i の近似逆行列を計算する時間が短くなるものの，近似シュールコンプリメント $\bar{S}$ の近似逆行列を計算する時間は長くなる．したがって，p を増やしても，2 レベル並列計算による近似逆行列の計算時間が必ずしも減少するとは限らない．

図 1 に示した 2 レベル計算の算法において，2，3 行目の計算はそれぞれの行の処理が各 PE で独立していることが分かる．つまり，M_i（$i = 1, \ldots, p$）と近似シュールコンプリメント $\bar{S}$ のローカルの部分 $C_i M_i B_i$（$i = 1, \ldots, p$）の処理は，各 PE で独立に計算できる．すなわち，これらの計算は並列計算できることになる．本稿では，M_i（$i = 1, \ldots, p$）と $C_i M_i B_i$（$i = 1, \ldots, p$）の処理を各 PE で独立して行う 2 レベル計算を **2 レベル並列計算**と呼ぶことにする．2 レベル並列計算を行う場合の計算と通信の様子を**図 2** で示す．数値実験では，2 レベル計算と 2 レベル並列計算の両方を実装し，それぞれの速度向上を比較検討する．

3. クリロフ部分空間法への実装

2 レベル（並列）計算で計算した近似逆行列をクリロフ部分空間法の前処理行列とする実装について述べる．まずクリロフ部分空間法に現れるベクトル $\boldsymbol{x} \in \boldsymbol{R}^n$ を $(x_1, x_2, \ldots, x_p, x_s)^T$ に分割し，x_i と x_s を i 番目の PE に配置する．ただし，$x_1, x_2, \ldots, x_p, x_s$ の次元は，それぞれ $A_1, A_2, \ldots, A_p, A_s$ の次元と一致する．

行列 $P^T AP$ とベクトルの積は，次式で計算するものとする．

$$\begin{bmatrix} A_1 & & & B_1 \\ & \ddots & & \vdots \\ & & A_p & B_p \\ C_1 & \cdots & C_p & A_s \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_p \\ x_s \end{bmatrix} = \begin{bmatrix} A_1 x_1 + B_1 x_s \\ \vdots \\ A_p x_p + B_p x_s \\ \sum_{i=1}^{p} C_i x_i + A_s x_s \end{bmatrix}$$

ここで，$A_i x_i + B_i x_s$（$i = 1, \ldots, s$）と $C_i x_i$（$i = 1, \ldots, s$）は，お互いに独立に並列計算できる．

なお，前処理行列 M とベクトルの積は，次式のように計算すればよい．

$$\begin{bmatrix} M_1 & & & \bar{E}_1 \\ & \ddots & & \vdots \\ & & M_p & \bar{E}_p \\ & & & M_s \end{bmatrix} \begin{bmatrix} I_1 & & & \\ & \ddots & & \\ & & I_p & \\ \bar{F}_1 & \cdots & \bar{F}_p & I_s \end{bmatrix} \times \begin{bmatrix} x_1 \\ \vdots \\ x_p \\ x_s \end{bmatrix} = \begin{bmatrix} M_1 x_1 + \bar{E}_1 z_s \\ \vdots \\ M_p x_1 + \bar{E}_p z_s \\ M_s z_s \end{bmatrix}$$

ただし，$z_s = \sum_{i=1}^{p} \bar{F}_i x_i + x_s$ である．また，$\bar{F}_i x_i$（$i = 1, \ldots, s$）と $M_i x_i$（$i = 1, \ldots, s$）の計算は独立に行えるので，並列計算が可能となる．

ベクトルの内積は，$\alpha = \sum_{i=1}^{p} < x_i, y_i > + < x_s, y_s >$ で計算する．よって，$< x_i, y_i >$（$i = 1, \ldots, s$）は，お互いに独立に並列計算することができる．

4. 数値実験

MPI を用いて SGI 社の共有メモリ型並列計算機 Origin 2400 を利用し，以下の条件で数値実験を行った．

- CPU の周波数と個数：300 MHz × 16 個

- 通信ライブラリ：MPI-1.2
- プログラム言語：C 言語
- 計算精度：倍精度
- ノードの割当て：1 CPU が 1PE を担当

グラフ分割法による行列の再構成には，Karypis ら[5] の開発した関数 pmetis を利用する．近似解を求めるために，クリロフ部分空間法の 1 つである GMRES(m) 法[9] を利用する．残差ノルムの収束評価に関しては，連立 1 次方程式の初期近似解を零ベクトルとし，GMRES(m) 法の収束判定条件を

$$\|\boldsymbol{r}_k\|_2 \,/\, \|\boldsymbol{b}\|_2 \leq 1.0 \times 10^{-12}$$

とする．ただし，$\boldsymbol{r}_k$ は k 回目の反復における GMRES(m) 法の残差ベクトルである．GMRES(m) 法の反復回数は，クリロフ部分空間の次元が 1 つ増加するたびに 1 回と数える．また，リスタート周期 m を 50 とする．計算時間については，MPI_Wtime() 関数で求めた値を秒単位で示す．計算時間と速度向上とも小数点以下 2 桁まで表示する．

AISM 法のパラメータ α は，Bru ら[2] の提案に基づき $\alpha = 1.5\times \|A\|_\infty$ の値を使用する．行列 U，V の非ゼロ要素の切り捨て処理を行う閾値は 0.1 とする．近似シュールコンプリメントの計算では，非ゼロ要素の切り捨て処理は行わない．計算した近似逆行列を GMRES(m) 法の右前処理行列として適用し，近似解を計算する．

［数値例］ 正方領域 $\Omega = [0,\ 1] \times [0,\ 1]$ における次の偏微分方程式の境界値問題を考える．

$$-u_{xx} - u_{yy} + D\left\{\left(y - \frac{1}{2}\right)u_x + \left(x - \frac{1}{3}\right)\left(x - \frac{2}{3}\right)u_y\right\} - 43\pi^2 u = G(x,\ y),$$
$$u(x,\ y)|_{\partial\Omega} = 1 + xy$$

ただし，右辺 $G(x,\ y)$ は真の解が $u = 1 + xy$ になるように決定する．領域 Ω を格子点数 192×192 に区切り，5 点中心差分で離散化すると得られる連立 1 次方程式の次元は 36,864 となる．Dh の値は 2^{-7} とする．ただし，h はきざみ幅であり $h = 1/193$ である．

最初に，1 つの PE で 2 レベル計算を行わずに，AISM 法で近似逆行列を計算する．近似逆行列の計算に，約 3,799.65 秒かかった．次に，p を 2，4，6，8，20，12，14，16 に設定し，グラフ分割を行い，係数行列を式 (5) に変換する．この処理にかかった時間は 1 秒未満であった．グラフによる分割の結果から，pmetis はグラフをほぼ均等に分割していることが分かる．つまり，行列 $A_i, (i = 1, \ldots, p)$ の次元は，ほぼ同じであった．また，p の増加につれ A_i の次元が減少し，A_s の次元が増大することが確認できた．$p = 2$ の場合，A_1，A_2 と A_s の次元はそれぞれ 18,239，18,239，386 であった．$p = 12$ になると，A_s と A_i の次元はほぼ同じになった．p を 14 以上に設定すると，A_s の次元は A_i を上回り，$p = 16$ の場合，A_i の平均次元は約 2,150 に対し，A_s の次元は 2,460 まで増大した．

表 1　2 レベル（並列）計算を行う場合の近似逆行列の計算時間と速度向上
Table 1　Computation time and Speedup of 2 level (parallel) computation for an approximate inverse.

	2 レベル計算		2 レベル並列計算	
p	計算時間	速度向上	計算時間	速度向上
2	1,686.56	2.25	850.19	4.46
4	818.30	4.64	206.91	18.36
6	551.11	6.89	95.15	39.93
8	426.68	8.90	58.06	65.44
10	360.54	10.53	42.60	89.18
12	315.92	12.02	33.65	112.90
14	295.39	12.86	30.70	123.76
16	277.31	13.70	27.78	136.72

ここで，1 つの PE を用いる 2 レベル計算と複数の PE を用いる 2 レベル並列計算を行い，それぞれの速度向上を**表 1** に示した．この表 1 から，1 つの PE で，2 レベル計算を行うだけで，最大約 13.70 倍の速度向上が得られたことが分かる．さらに，2 レベル並列計算を行うと，16 台の PE で最大約 136.72 倍の速度向上得られた．2 レベル計算と 2 レベル並列計算とも，p の増大につれ，得られた速度向上が大きくなった．ただし，$p = 14$ 以降，速度向上の増大率は小さくなった．それは，p を 14 以上に設定すると，A_s の次元が A_i よりも大きくなり，近似シュールコンプリメント $\bar{S}$ の近似逆行列を計算する時間が長くなったためである．

最後に，2 レベル並列計算で計算した近似逆行列を GMRES(50) 法の前処理行列として適用し近似解を計算した．式 (7) を満たすまでに要した反復回数と計算時間を**表 2** に示す．ただし，T_{pre} は近似逆行列の計算時間を表している．I と T_{gmres} は GMRES(50) 法の反復回数と計算時間を表しているが，T_{gmres} には前処理時間である T_{pre} は含まれていない．T_{total} は総処理時間であり，$T_{\mathrm{total}} = T_{\mathrm{pre}} + T_{\mathrm{gmres}}$ の関係がある．また，E_{total} は，近似解を計算することにおいて，1 つの PE で 2 レベル計算を行わない場合の総処理時間に対する速度向上の値である．表 2 から，$p = 14$ まで，近似解を計算する総処理も高い速度向上が得られたことが分かる．ただし，$p = 16$ の場合，速度向上は 10.04 まで落ちた．これは，3 章で述べた実装法で GMRES(m) 法を並列化する場合に，A_s の次元が大きくなると，GMRES(m) 法の並列効果が落ちるためである．

表 2　GMRES(50) 法の数値実験の結果
Table 2　Numerical results of GMRES(50).

p	T_{pre}	I	T_{gmres}	T_{total}	E_{total}
2	850.19	10,195	5,173.39	6,023.58	1.90
4	206.91	7,297	1,743.88	1,950.79	5.88
6	95.15	9,006	1,486.89	1,581.97	7.25
8	58.06	8,207	1,101.35	1,159.40	9.89
10	42.60	9,595	1,170.40	1,213.01	9.45
12	33.65	8,244	932.81	966.40	11.87
14	30.70	6,991	789.02	819.69	13.99
16	27.78	8,395	1,114.45	1,142.13	10.04

5. 結　　論

近似逆行列の 2 レベル並列計算に AISM 法を適用し，SGI 社の Origin 2400 に算法の並列実装を行った．数値実験により，近似逆行列の計算は，最大約 136.72 倍の速度向上が得られた．近似解を求める総処理も最大 13.99 倍の速度向上が得られた．したがって，提案する手法は計算機環境で AISM 法を利用する場合の有効な並列手法の 1 つになりうるといえる．

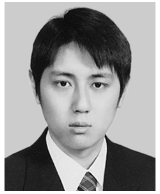
参 考 文 献

1) Benzi, M., Marín, J. and Tůma, M.: A Two-level Parallel Preconditioner Based on Sparse Approximate Inverse, *IMACS Series in Computational and Applied Mathematics*, Vol.5, pp.167–178, IMACS (1999).
2) Bru, R., Cerdán, J., Marín, J. and Mas, J.: Preconditioning Sparse Nonsymmetric Linear Systems with the Sherman-Morrison Formula, *SIAM J. Sci. Comput.*, Vol.25, No.2, pp.701–715 (2003).
3) Chow, E. and Saad, Y.: Approximate Inverse Preconditioners via Sparse-sparse Iterations, *SIAM J. Sci. Comput.*, Vol.19, No.3, pp.995–1023 (1998).
4) Heath, M.T., Ng, E. and Peyton, B.W.: Parallel Algorithms for Sparse Linear Systems, *SIAM Review*, Vol.33, No.3, pp.420–460 (1990).
5) Karypis, G. and Kumar, V.: A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs, *SIAM J. Sci. Comput.*, Vol.20, No.1, pp.359–392 (1998).
6) 森屋健太郎，張　臨傑，野寺　隆：Sherman-Morrison 法の並列化による近似逆行列の計算，情報処理学会論文誌，Vol.48, No.3, pp.1462–1478 (2007).
7) Naik, V.K.: A Scalable Implementation of the NAS Benchmark BT on Distributed Memory Systems, *IBM Systems Journal*, Vol.34, No.2, pp.273–291 (1995).
8) Saad, Y.: *Iterative Methods for Sparse Linear Systems*, PSW Publishing Co., Boston (1966).
9) Saad, Y. and Schultz, M.H.: GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems, *SIAM J. Sci. Statist. Comput.*, Vol.7, No.3, pp.856–869 (1986).

(平成 19 年 12 月 28 日受付)
(平成 20 年 3 月 4 日採録)

張　　臨傑

1997 年中国西安交通大学コンピュータ学部コンピュータ科学 & 工学学科卒業．2004 年 3 月慶應義塾大学大学院理工学研究科基礎理工学専攻博士前期課程修了．2007 年 9 月同大学大学院理工学研究科基礎理工学専攻博士後期課程修了．工学博士．現在，中国海洋大学助教．並列計算機による大規模線形計算に興味を持つ．



森屋健太郎（正会員）

1999 年富士通株式会社入社．2003 年 4 月より青山学院大学理工学部助手に就任．2007 年 4 月助教となり同年 6 月退職．同年 7 月 IT 関連企業入社，現在に至る．その間，2000 年 9 月慶應義塾大学大学院理工学研究科後期博士課程（基礎理工学専攻）に入学し，2002 年 3 月同大学院同研究科後期博士課程修了．工学博士．並列計算機による大規模計算に興味を持つ．

野寺　　隆（正会員）

1982 年慶應義塾大学大学院工学研究科博士課程（数理工学専攻）修了．現在，同大学教授．その間，1986 年より 1 年間米国スタンフォード大学客員教授．大規模な行列計算の算法の研究開発に従事．ハイパフォーマンス・コンピューティングや文書処理に興味を持つ．著書に『楽々LaTeX』（共立出版）等がある．工学博士．エッセイスト．SIAM，日本応用数理学会各会員．