Vol.2011-HPC-130 No.5
2011/7/27

ヘテロ環境を目指した拡張階層型領域間分割に基づく高次フィルイン付き前処理手法の高速化

林雅江, 大島聡史, 中島研吾[†1]

拡張階層型領域間分割は，領域外からの高次フィルインを考慮可能とする並列化手法であり，分散データの局所性も高いことから，メニーコア環境での効率的な並列化手法として期待される．本研究では，物性値分布に不均質性をもつことから悪条件となる三次元静弾性問題に対し，拡張階層型領域間分割に基づく高次フィルイン前処理付き反復解法を適用する．本報告では，T2K(東大) を利用し，マルチコア環境における本並列実装プログラムの収束性および高次フィルイン付き前処理の並列性能についてマルチカラー法との比較に基づき評価する．

Parallel ILU Preconditioner Based on Extended Hierarchical Interface Decomposition for Heterogeneous environments

Masae HAYASHI, Satoshi OHSHIMA,
Kengo NAKAJIMA[†1]

Extended version of Hierarchical Interface Decomposition(HID) is developed as an effective parallelization method for Finite Element Method(FEM) on multi/many-core environments for its high locality of distributed mesh data. And thichker separators introduced in Extended HID allow us to take into account high level fill-ins in parallel ILU preconditioners. We implemented Extended HID to OpenMP parallel FEM base simulation of linear elasticity problem with heterogeneous property. The developed code has been tested on the T2K Open Super Computer(T2K/Tokyo) using 1 node, 16 cores to evaluate the robustness and the scalability of parallel ILU decomposition based on the comparison with MC ordering.

1. はじめに

反復法の収束特性は係数行列の固有値分布に依存するため，実用的な問題に適するためには前処理を施すことが一般的であり，悪条件問題になれば前処理は不可欠である．Block Jacobi 型前処理は並列化において，領域外からの影響を無視することで，高い並列性が得られる．しかしながら，本来の大域的な処理を無視するため，領域数の増加とともに前処理としての効果が弱まり，反復回数が著しく増加する場合がある[3),8)]．階層型領域間境界分割 (Hierarchical Interface Decomposition, HID)[4)] は，不完全 LU 分解 (ILU)，不完全 Cholesky 分解 (IC) に基づく並列前処理手法を安定に収束させ，高い並列性能を得る事ができる領域分割手法であるが，領域外の節点からの高次 fill-in を考慮することができず，悪条件問題では必ずしも安定ではない．拡張階層型領域間分割 (Extended HID) は，領域外節点からの高次 fill-in の影響を，HID に基づく並列 ILU/IC 前処理手法に導入可能な手法として提案された[5)]．Extended HID はセパレータ領域を拡張し，その厚みを増やすことで領域外にある節点からの高次の fill-in を考慮可能とする．本研究では，局所的な不均質性を考慮した有限要素法 (以下 FEM) による三次元静的弾性力学問題に Extended HID を適用し，OpenMP によるマルチコア環境向け並列コードを実装した．解法には 2 次までの fill-in が考慮可能な ILU 前処理付き GPBiCG 法を用いる．T2K オープンスパコン (東大) の 1 ノード 16 コアを利用し，その収束性と並列性能について評価した．並列性能については特に，高次 fill-in 付き不完全 ILU 分解の並列実装部分に注目する．また，評価する上で共有メモリ型並列計算機における ILU 前処理等の並列化に用いられてきた従来手法の一つ，MultiColor(MC) ordering を用いて並列化した場合と比較する．本論文は以下，2 節でアプリケーションと Extended HID 法の概要について述べ，3 節に実験環境，4 節にて収束性の評価を，5 節にて不完全 ILU 分解における並列性能評価を示し，6 節にて考察，7 節にてまとめと今後の課題を述べる．

2. アプリケーションと Extended HID 法の概要

2.1 不均質場における三次元弾性問題

図 1 に示す不均質物性分布をもつ立方体形状における三次元静的弾性問題を有限要素法

†1 東京大学 情報基盤センター
Infomation Technology Center, The University of Tokyo

Vol.2011-HPC-130 No.5
2011/7/27

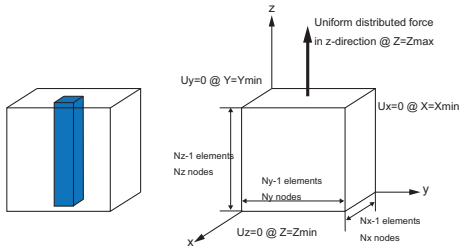

図 1 Simple cube geometries with heterogeneity for 3D linear elastic problems.

によって解く．一次補間関数に基づく六面体アイソパラメトリック要素を使用し，各要素は辺の長さ＝1の立方体である．ポアソン比は全要素で0.30とし，ヤング率については，図1に青色で示す中心部分にある要素にはその周囲と異なる値を与え不均質性をもたせる．具体的には，青色で示す部分の要素には，$E = 1000$ を与え，それ以外の部分には $E = 1.00$ を与える．こうした問題では，領域分割による並列化を行うことで，中心部分が領域境界によって分断されてしまい，収束性が悪化することが知られている．こうした問題においては，対角スケーリングやILU(0)等の簡便な前処理手法で収束を得る事は難しく，高次のfill-inが考慮可能なILU前処理など，より強力な前処理が必要となる．本プログラムコードは．GeoFEM[9]の並列有限要素法フレームワークに基づき，GeoFEMの節点ベース局所データ構造を使用している．各点における(u,v,w)の3変位成分を同時に処理する3x3ブロック化がなされている．反復解法には，GPBiCG法(Generalized Product-type method based on BiCG)[7]を使用した．

2.2 階層型領域間境界分割 (HID) による並列化

HIDは，Nested Deissectionをより厳密に行うものと解釈され，第一段階として，有限要素メッシュの節点とコネクティビティ情報から構成されるグラフを互いに共通部分を持たない節点の集合(connector)とそれを隔てるseparatorに分解する[4]．この段階でできる節点の集合を「レベル1のコネクタ (C^1)」と呼ぶ．各 C^1 は各部分領域に当たる．HIDでは，C^1 に含まれなかった残りの節点(separator)に対しても再帰的に分割処理を実施し，separatorとそれによって隔てられるconnectorへと分割する．こうして分割を繰り返してゆく際，各節点にレベルが割り振られる．レベルは，あるコネクタに属する節点がいくつの部分領域と共有節点を持つかを表す．すなわち，レベル k の connector(C^k, $k > 1$) は，k 個の部分領域と共有節点を持つ．また，同レベルのconnectorに属する節点同士が共有節点を持つことはなく，あるレベル k のconnectorはより高いレベル $l(> k)$ のconnectorによって隔てられる．図2左は，二次元9点差分格子を4領域に分割する場合の例である．HIDでは，各節点が低いレベルに属する節点から順に番号付けがなされる．その結果，並び替えられた節点のコネクテビティを元に生成された全体マトリクスは図2右に示すようにブロック構造を有する．同レベルに属するコネクタ同士は直接隣接しないため，このような構造によってILU(0)/IC(0)前処理やガウスザイデル前処理等の計算プロセスを並列化することができ，レベル順に処理してゆく事でBlockJacobi型の局所並列前処理と比較してより完全な前進後退代入が実現できる．

2.3 拡張階層型領域間境界分割法 (Extended HID)

HIDによるレベル順のリオーダリングによって，ILU(0)/IC(0)前処理を並列化ができる．ただし，このままでは，領域外の節点からの高次fill-inを考慮する事ができないため，悪条件問題では必ずしも安定ではない[4]．そこで，HID法を基にした並列ILU/IC前処手法において，領域外の節点からの高次fill-inの影響を考慮可能としたものがExtended HID(以下ExHID)である[5]．ExHID法は，領域外からの節点のセパレータの厚みを拡張することで考慮可能とする．図3に示すようにseparator領域を3層にすることで，隣接領域からの2次のfill-inが考慮可能となる．fill-inレベルに応じて，separatorの厚みを調節する．

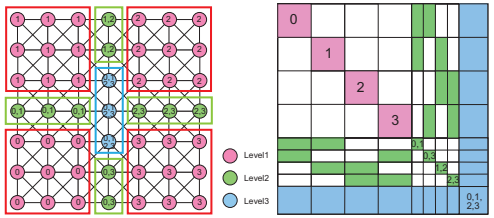

図 2 Domain decomposition of FE mesh accoding to HID(left) and reordered matrix(right). reordering.

3. 実験環境

本研究では，実験環境にT2Kオープンスパコン(東大)(T2K(東大))を使用した．T2K(東大)は筑波大，東大，京大の3大学で定められたT2Kオープンスパコン仕様に基づき日立製作所が製作した952ノード，約15,000コア，ピーク性能140TFLOPSのクラスタ型コ

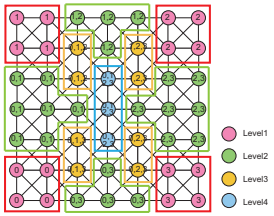

図 3　Extended HID with thicker level-2 separators.

ンピュータシステムである．各ノードは cc-NUMA(Cache-Coherent NUMA) アーキテクチャに基づき AMD quad-core Opteron(2.3GHz) 4 ソケット，合計 16 コアから構成されている．ノード当たりの記憶容量は 32GB(一部 128GB) である．

4. 収束性に関する実験

4.1 概　　要

ExHID による並列化と MC ordering(以下 MC) による並列化を行った場合のそれぞれの収束性について評価する．ExHID を用いた場合は，fill-in なしの時には separator の厚みはオリジナルの HID と同様に 1 層とし，1 次，2 次の fill-in の考慮する時には，separator の厚みを 3 層または 5 層とする．従って，fill-in レベル l と separator の厚み t の組み合わせより前処理を指定し，ILU(l=0,t=1), ILU(1,3), ILU(1,5), ILU(2,5) の 4 通りを利用する．一方の MC では separator の厚みを設定する代わりに，fill-in レベルに応じて色塗りのルールを厳しくしてゆく．fill-in レベルが 1 の時は隣接節点の隣接まで，fill-in レベル 2 の時は隣接節点の隣接の隣接まで含めて色分けルールを適用する．従って，fill-in レベルが上がるほど必要な色数は多くなる．本実験では，各 fill-in レベルにおける最少の色数を用いることとし，ILU(0) では 8 色,ILU(1) では 27 色,ILU(2) では 64 色となっている．問題サイズは 50^3 要素，397,953 自由度である．T2K1 ノード，16 スレッドで並列計算したときの収束までにかかった反復回数を比較する．

4.2 実 験 結 果

図 4 に MC による前処理 3 通りで要した反復回数と ExHID による前処理 4 通りで要した反復回数を示す．両手法とも fill-in レベルを上げることで反復回数の減少が得られ，ExHID においてはさらに，セパレータの厚みを上げることでも反復回数の減少が得られていることがわかる．また，ExHID と MC を比べると，ExHID の方がより短い反復回数で収束が得られており，ILU(2,5) の時，最少の 103 反復で収束が得られた．

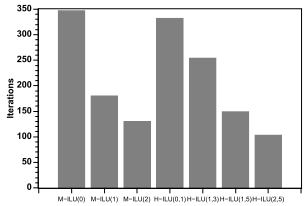

図 4　Iterations when increasing fill-in level using 16 cores on T2K/Tokyo for the 3D linear elasticity model with 125,000 elements, 397,953 DOF.

5. 並列性能の調査

5.1 概　　要

ExHID と MC を用いた場合の計算時間の比較および並列性能について評価する．問題規模は前節と同様の 125,000 要素，397,953 自由度とし，スレッド数を 4,8,16 で並列実行する．NUMA アーキテクチャでは「最初にそのバッファにアクセスしたプロセッサのメモリ上にバッファの内容がアサインされる」というファーストタッチ (FT) ルールがある．そのため，本研究で開発したソースコードにおいては，初期化等の処理において各プロセスが各自の部分領域のデータを FT するように実装した上で，実行時には各スレッドが割り当てられたコアに属するローカルなソケット上のメモリを使用する　numactl ポリシ「–cpunodebind=$SOCKET –localalloc」を適用している．

5.2 実 験 結 果

スレッド数を 4,8,16 とした場合の fill-in 付き ILU 分解と反復計算 (収束までに要した全反復計算) に要した時間を計測した結果を示す．図 5 は，反復計算に要した計算時間で，左側が MC を用いた場合，右側が ExHID を用いた場合である．図 6 は ILU 分解に要した計算時間で，同じく左側が MC，右側が ExHID を用いた場合である．fill-in レベルが上がるほど，ILU 分解に要する時間が大きくなり，反復計算と同等の大きさになることがわかる．

次に，手法間を比較すると，反復計算，ILU 分解ともに，Extended HID を用いた場合の方が全体的により短い計算時間となることがわかる．反復回数が最少となった Extended HID の ILU(2,5) では 24.9 秒で，MC による ILU(2) はその約 1.8 倍となっている．

次に，ILU 分解に要した計算時間について，4 スレッド実行時の計算時間を 1 とした時の

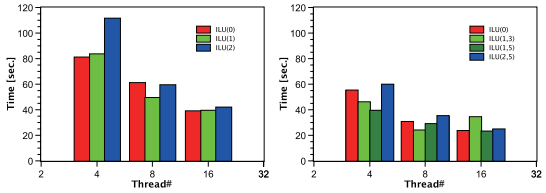

図 **5** Elapsed time for iterations using 4 to 16 cores by MC ordering(left) and Extended HID(right) on T2K/Tokyo for the 3D linear elasticity model with 125,000 elements, 397,953 DOF.

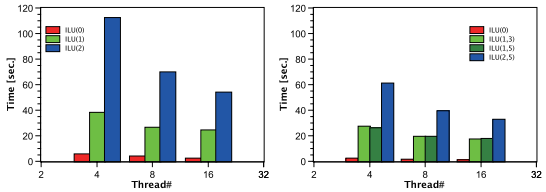

図 **6** Elapsed time for ILU factorization using 4 to 16 cores by MC ordering(left) and Extended HID(right) on T2K/Tokyo for the 3D linear elasticity model with 125,000 elements, 397,953 DOF.

Speed-up を図 7 で比べる．右側が MC，左側が ExHID による結果である．Speed-up については手法間に大きな違いはなく，8 スレッドで約 1.3-1.8 倍，16 スレッドで約 1.6-2.2 倍となった．

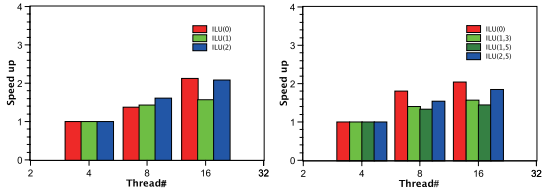

図 **7** Speed-up for ILU factorization using 4 to 16 cores by MC ordering(left) and Extended HID(right) on T2K/Tokyo for the 3D linear elasticity model with 125,000 elements, 397,953 DOF.

6. 考　　察

6.1 fill-in 数について

各 fill-in レベルでの fill-in の数を ExHID と MC で比較した．ExHID では fill-in の数が ILU(1,3) で 2,308,285,ILU(1,5) で 2,356,327 ILU(2,5) で 5,657,618 となり，MC では ILU(1) で，2,986,540 で ILU(2) で 7,780,420 となり，ExHID を用いた場合がいずれの fill-in レベルでも fill-in 数が少ない事がわかる．このことから，ExHID の方が ILU 分解に要する計算時間がより短くなったと考えられる．また，fill-in 数の減少に加え，ExHID では反復回数もより少ない回数で収束が得られていたたことから，反復計算時間の減少にも結びついたと考えられる．

6.2 色数について

MC の色数については各 fill-in レベルに対し，最少の色数を指定していた．そこで，MC の ILU(2) を行う際の色数を最大 5000 色まで増加させていった場合の収束性および，計算時間 (反復計算のみ，反復計算+ILU 分解) を追試調査した．図 8 左に収束までにかかった反復回数，図 8 右に計測時間を示す．収束までに要する反復回数は，約 500 色から顕著に減少し，ExHID の ILU(2,5) とした時とほぼ同等の反復回数 (108 反復) で収束し，約 1000 色ではそれを下回る 64 反復で収束が得られた．一般には色数か増えるとスレッド間で同期をとるタイミングが増えるため性能劣化につながると懸念されるが，本実験においては計算時間についても約 1000 色の場合に計算時間が最少の約 20 秒となり，ExHID の ILU(2,5) の結果 (24 秒) を下回る結果を得た．

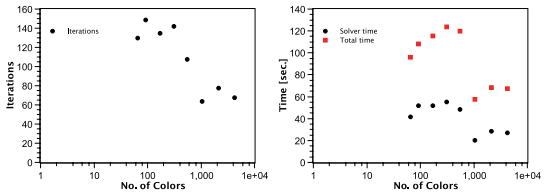

図 **8** Increase in the number of colors by Multicolor ordering according to the level of fill-ins of ILU preconditioner

7. まとめと今後の課題

ExHID を局所的な不均質性を考慮した有限要素法による三次元静的弾性力学問題に適用し，2 次の fill-in まで考慮可能な ILU 前処理付きの OpenMP 並列コードを実装した．T2K/Tokyo の 1 ノード 16 コアを利用し，収束性および計算時間を MC との比較を交え評価した．その結果，MC の色数が少ない場合には，ExHID を用いた方がよりよい収束性を得ることや，fill-in レベルを上げた場合にも生じる fill-in 数が少ないことから，結果的に ILU 分解および反復計算部分にかかる計算時間も最短となった．ただし，MC の色数を上げていった場合には，一般に収束性が向上し，追試実験においては収束性，計算時間ともに ExHID の結果を下回る結果が得られ，MC との比較については色数を含めた更なる調査が必要といえる．本並列実装を GPU 利用に展開してゆき，高コストな高次 fill-in 付き ILU 分解や前処理部分の更なる高速化を目指してゆく事が今後の課題である．

参 考 文 献

1) K. Nakajimal :Parallel Iterative Solvers of GeoFEM with Selective Blocking Preconditioning for Nonlinear Contact Problems on the Earth Simulator, ACM. IEEE Proceedings of SC2003, 2003.
2) K. Nakajima :Strategies for Preconditioning Methods of Parallel Iterative Solvers in Finite-Element Applications in Geophysics, Advances in Geocomputing, Lecture Notes in Earth Science, Vol.119, pp.65-118, 2009.
3) K. Nakajima, H. Nakamura and T. Tanahashi: Parallel Iterative Solvers with Localized ILU Preconditioning, Lecture Notes in Computer Sience, Vol.1225, pp.342-350, 1997.
4) P. Henon and Y. Saad :A Parallel Multistage ILU Factorization based on a Hierarchical Graph decomposition, SIAM Journal for Scientific Computing, Vol.28, pp.2266-2293, 2007.
5) K. Nakajima :Parallel Multistage Preconditioners based on a Hierarchical Graph Decomposition for SMP Cluster Architectures with a Hybrid Parallel Programming Model, Lecture Notes in Computer Science, Vol.4782, pp.384-395, 2007.
6) Information Technology Center, the University of Tokyo: http://www.cc.u-tokyo.ac.jp/
7) 藤野清次，張紹良「反復法の数理」，朝倉書店，1996.
8) K. Garatani, H.Nakamura, H.Okuda and G.Yagawa :GeoFEM:High Performance Parallel FEM for Solid Earth, Lecture Notes in Computer Sicence, Vol.1593, pp.132-140, 1999.
9) GeoFEM: http://geofem.tokyo.rist.or.jp/
10) 岩下武史，高橋康人，中島浩: 代数ブロック化多色順序付け法による並列化 ICCG ソルバの性能評価，情報処理学会研究報告 (HPC121-11)，2009.