新機能紹介: 節点データ並べ替え処理を用いた FrontISTRの性能評価

2017年6月9日 第36回FrontISTR研究会

<計算事例/Ver.4.6の新機能/Ver.5.0の公開に向けた取り組み>

謝辞

本研究開発は、文部科学省ポスト「京」重点課題® 「近未来型ものづくりを先導する革新的設計・製造プロセスの 開発」の一環として実施したものである

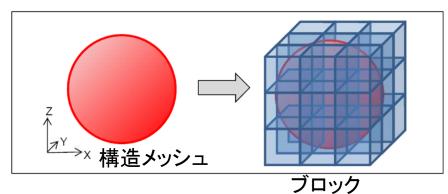
文部科学省 フラッグシップ2020プロジェクトポスト「京」重点課題®「近未来型ものづくりを先導する革新的設計・製造プロセスの開発」サブ課題E「新材料に対応した高度成形・溶接シミュレータの研究開発」

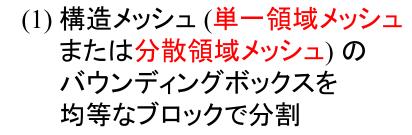
プロジェクト概要

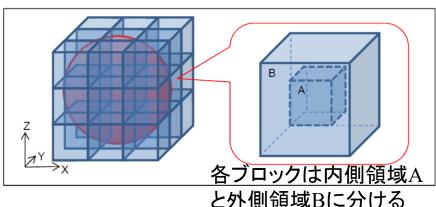
高度溶接シミュレーション技術を開発し、溶接工程における溶接順序探索および逆ひずみ量推定の高精度化・高速化を行うことが目的である。平成29年度までの達成目標として、入熱による熱弾塑性解析の計算精度を検証し、数mmのオーダーの溶融条件を考慮した大規模並列計算性能を検証する。平成31年度までの達成目標として、ターゲット問題における部品規模の溶接解析の計算精度を従来アプリと比較し、開発するアプリの優位性を示す。そして、全体規模の溶接解析結果を実験値と比較し、開発するアプリの予測精度を検証する。

ポスト「京」重点課題®サブ課題Eのホームページ http://www.multi.k.u-tokyo.ac.jp/PostK-8E/

節点データ並べ替え処理 (リオーダリング処理)



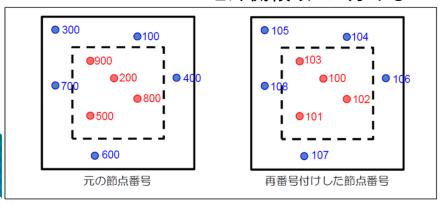




(2) ブロック順に、各ブロックに対して 節点番号のリナンバリング

<u>リナンバリングのルール</u>

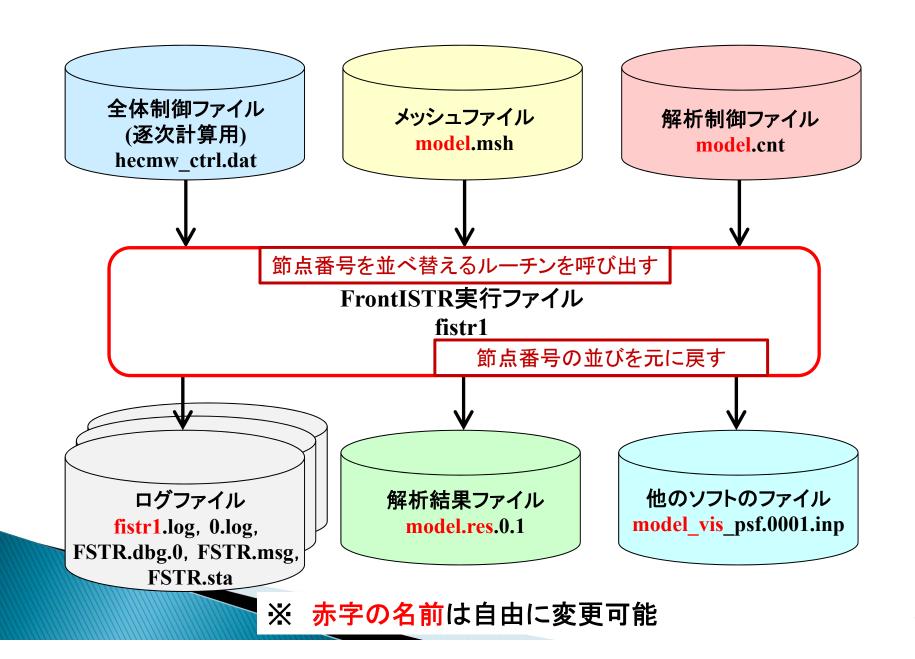
- (a) 節点座標の値と無関係に リナンバリング
- (b) 各ブロックに対して, まず内側領域Aの番号を詰め, 次に外側領域Bの番号を詰める ようにリナンバリング



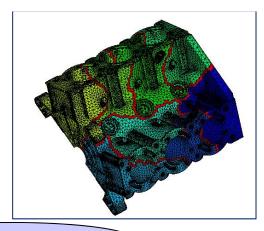
(1) と (2) をFrontISTRプログラム の中で行う

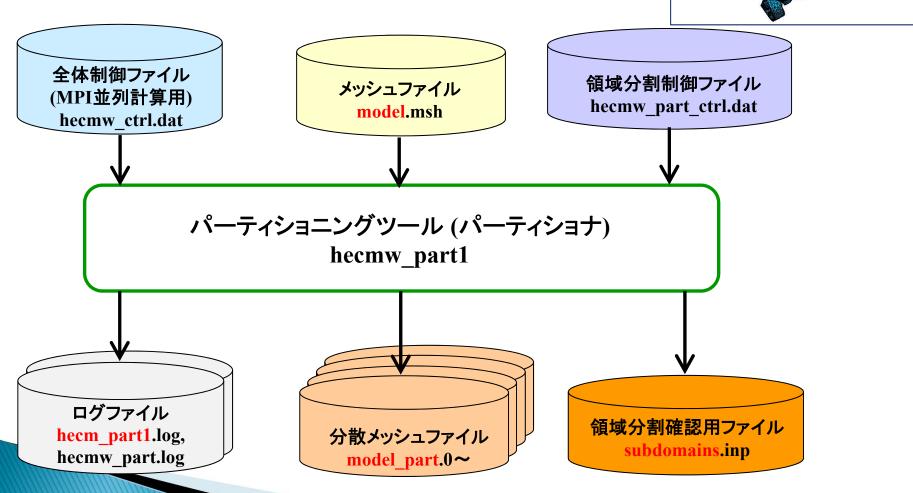
内側領域A. 外側領域Bの順に節点番号を詰める

節点並べ替え処理を移植したFrontISTRの逐次計算の流れ



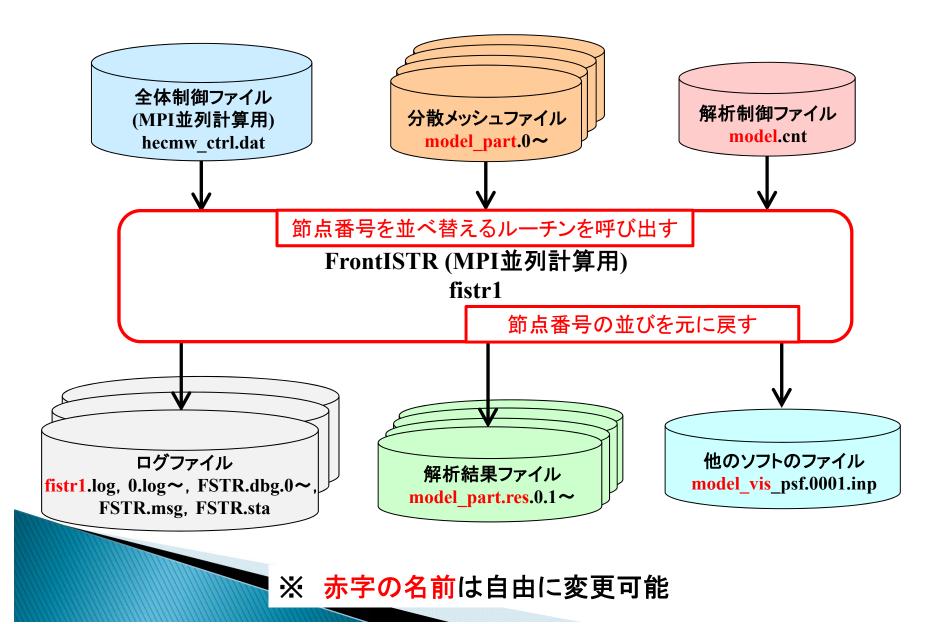
節点並べ替え処理を移植した FrontISTRの並列計算の流れ (1/2)





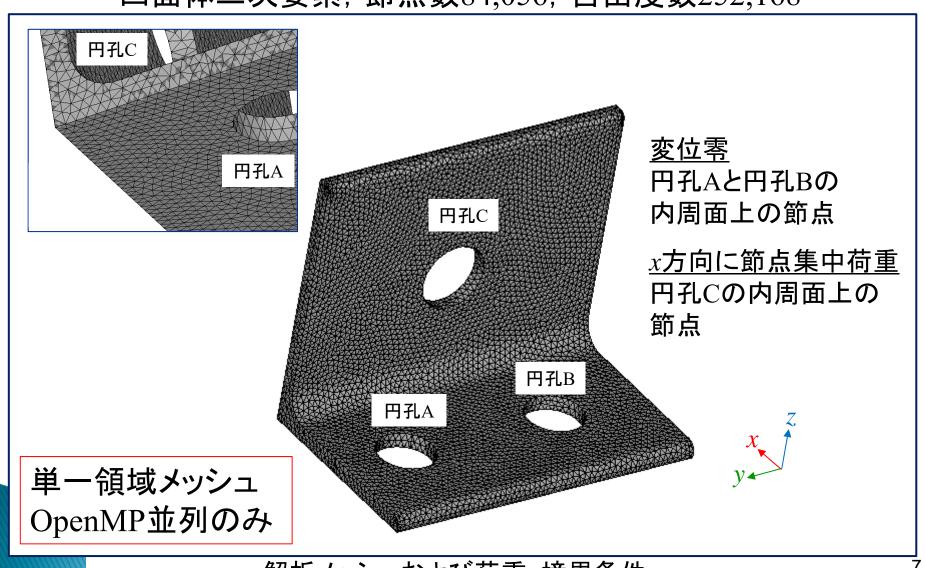
※ 赤字の名前は自由に変更可能

節点並べ替え処理を移植したFrontISTRの並列計算の流れ (2/2)



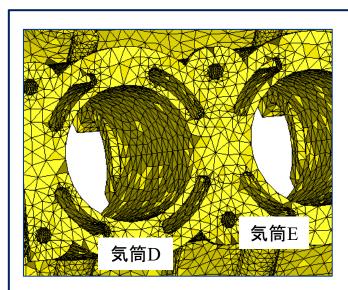
計測に使用する解析モデル (1/3)

四面体二次要素, 節点数84,056, 自由度数252,168



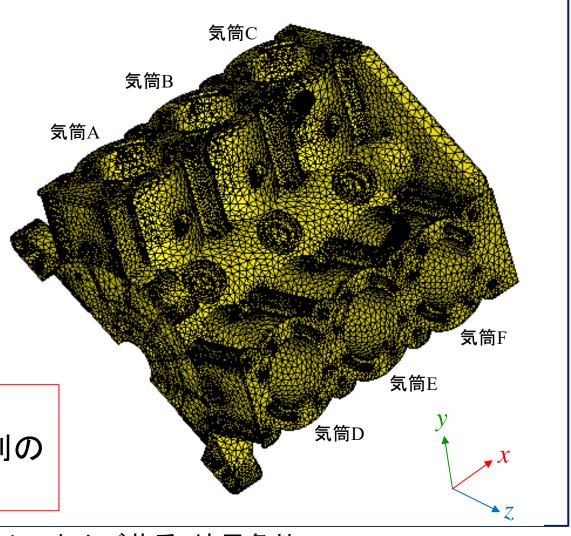
計測に使用する解析モデル (2/3)

四面体二次要素, 節点数459,292, 自由度数1,377,876



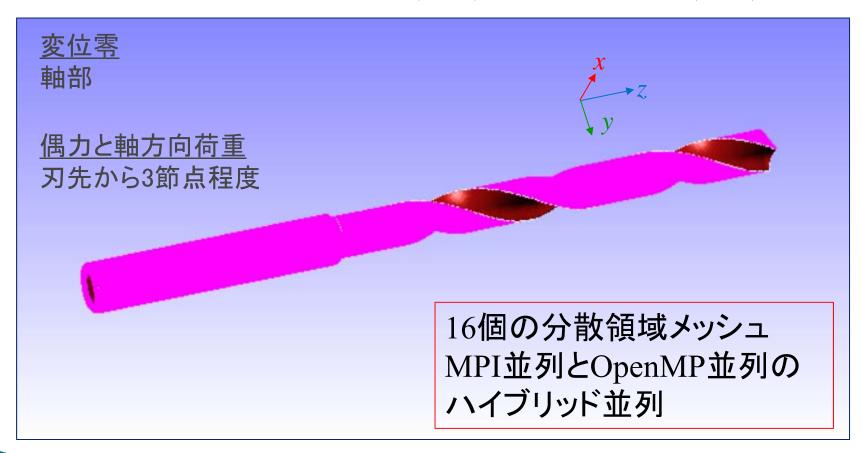
<u>圧力</u> 気筒内面

4個の分散領域メッシュ MPI並列とOpenMP並列の ハイブリッド並列



計測に使用する解析モデル (3/3)

四面体一次要素、節点数1,700,262、自由度数5,100,786



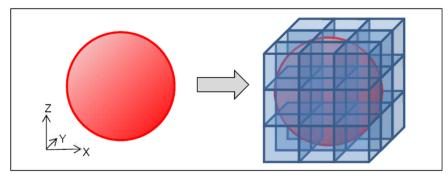
解析メッシュおよび荷重・境界条件

使用した計算機

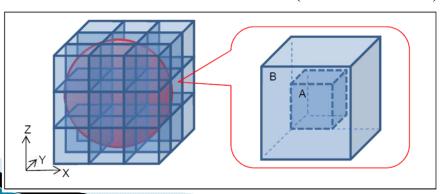
- ・ 京コンピュータ理化学研究所計算科学研究機構・・・ MPIの1プロセスあたり8スレッド並列
- Oakleaf-FX (FX10)
 東京大学情報基盤センター
 MPIの1プロセスあたり16スレッド並列
- FX100名古屋大学情報基盤センター・・・・ MPIの1プロセスあたり32スレッド並列
- Oakforest-PACS
 筑波大学計算科学研究センター/東京大学情報基盤センター
 ・・・ MPIの1プロセスあたり68スレッド並列

ブロック分割と節点番号リオーダリング

- 節点データ並べ替え処理のブロック分割数 N^3 • • 10³, 20³, 30³, 40³, 50³, 60³, 70³
- 今回は、節点番号リオーダリングにおいて、外側領域B は設定しない (FX10でのhingeモデルの計算において、 計算速度の違いがほとんどなかったため)



ブロック(1辺の分割数N)



反復法線形ソルバー

- 前処理付きCG法を使用
- 前処理には、ブロックSSOR前処理を使用 hingeとv6engineでは一回、drillでは二回前処理 参考のため、ブロック対角スケーリング前処理も使用
- CG法における残差の収束判定値は1.0×10-8に設定

時間を計測したプログラムの範囲

- 前処理 (Precond)
 hecmw1/src/solver/solver_33/hecmw_precond_33.f90
 call hecmw_precond_SSOR_33_apply(ZP) の前後
- 行列ベクトル積 (SpMV)
 hecmw1/src/solver/solver_33/hecmw_solver_las_33.f90
 call hecmw_matvec_33_inner(hecMESH, hecMAT, X, Y, Tcomm) の前後

- ※ ブロックSSOR前処理付きCG法とブロック対角スケーリング 前処理付きCG法のSpMVの計測時間はほとんど同じことを確認
- ※ ブロック対角スケーリング前処理付きCG法のPrecondの計算時間は非常に短いことを確認 (ブロックSSOR前処理付きCG法のPrecondの1/10~1/20)

評価する値

• オリジナル (リオーダリング無し) に対する各反復での 実行時間に対する高速化率

(節点リオーダリングを行わない各反復での実行時間) /(節点リオーダリングした各反復での実行時間)

 詳細プロファイラで求める値 浮動小数点演算ピーク比, FLOPS, メモリスループット, メモリビジー率, L2エンジンビジー率, L1エンジンビジー率, L1Dミス率, L1Dミスdm率, 浮動小数点ロードキャッシュアクセス待ち

京コンピュータでの高速化率 (1/3)

Table Speed-up factors for SpMV and Precond (hinge model)

The number of blocks	Computation time (SpMV) [s]	Computation time (Precond) [s]	Iteration	Speed-up (SpMV)	Speed-up (Precond)
Original	18.28801	23.05571	2969		
10^{3}	18.10317	21.89459	2960	1.00714	1.04984
20^{3}	17.56169	21.20926	2923	1.02522	1.07021
30^{3}	17.44582	21.22267	2931	1.03485	1.07246
40^{3}	17.63312	21.72549	2914	1.01792	1.04157
50^{3}	17.07438	21.37279	2923	1.05448	1.06202
60^{3}	17.25029	21.22111	2909	1.03873	1.06449
-0^{3}	17.55793	21.48429	2914	1.02228	1.05326

- ・ MPIのランク割り当ては1次元
- 計測回数は詳細プロファイラExcel対応のデフォルト値である7回
- ・ プロセス0のプロファイル情報を出力

京コンピュータでの高速化率 (2/3)

Table Speed-up factors for SpMV and Precond (v6engine model)

The number of blocks	Computation time (SpMV) [s]	Computation time (Precond) [s]	Iteration	Speed-up (SpMV)	Speed-up (Precond)
Original	1.146401	1.235759	1014		
10^{3}	1.143144	1.240854	1010	0.99889	0.99196
20^{3}	1.127818	1.193071	1009	1.01146	1.03067
30^{3}	1.139960	1.181571	1002	0.99374	1.03348
40^{3}	1.157218	1.224151	1008	0.98479	1.00350
50^{3}	1.154322	1.201551	1004	0.98334	1.01832
60^{3}	1.154717	1.212101	1003	0.98202	1.00845
-70^{3}	1.132946	1.189572	1008	1.00588	1.03267

- MPIのランク割り当ては1次元
- 計測回数は詳細プロファイラExcel対応のデフォルト値である7回
- プロセス0のプロファイル情報を出力

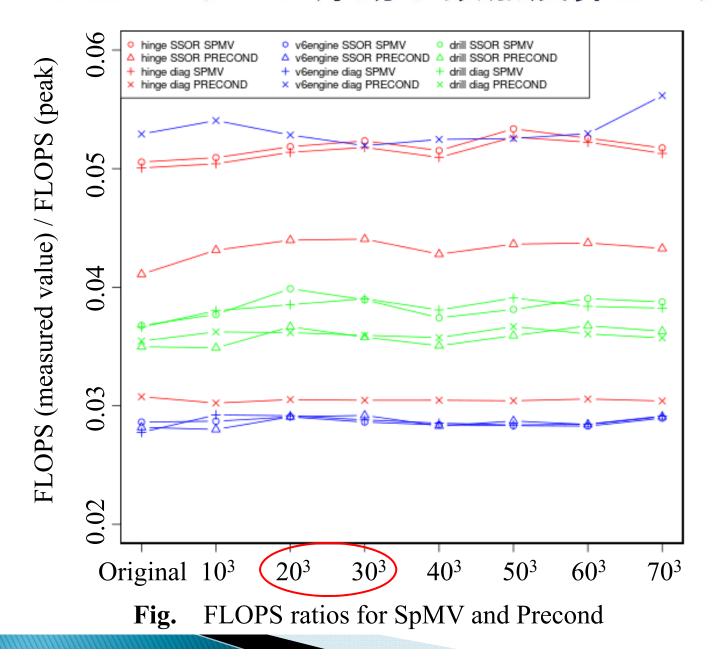
京コンピュータでの高速化率 (3/3)

Table Speed-up factors for SpMV and Precond (drill model)

The number of blocks	Computation time (SpMV) [s]	Computation time (Precond) [s]	Iteration	Speed-up (SpMV)	Speed-up (Precond)
Original	76.05670	84.71761	6097		
10^{3}	70.85679	80.92576	5850	1.02990	1.00444
20^{3}	69.19652	79.39498	5848	1.05425	1.02346
30^{3}	71.95737	82.52394	6103	1.05800	1.02759
40^{3}	75.12283	84.60778	6121	1.01641	1.00523
50^{3}	72.29470	81.00781	6039	1.04202	1.03584
60^{3}	72.26197	81.13552	6130	1.05821	1.04980
70^{3}	71.96623	80.96424	6113	1.05961	1.04910

- MPIのランク割り当ては1次元
- 計測回数は詳細プロファイラExcel対応のデフォルト値である7回
- プロセス0のプロファイル情報を出力

京コンピュータでの浮動小数点演算ピーク比



京コンピュータでのL1Dミスdm率

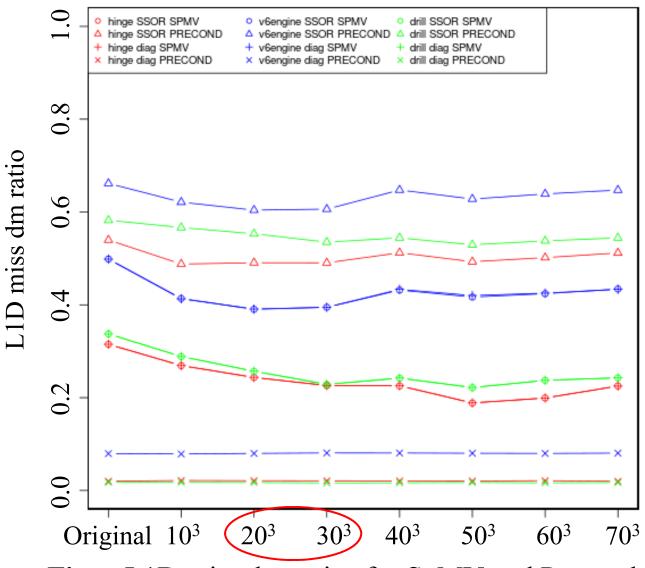


Fig. L1D miss dm ratios for SpMV and Precond

京コンピュータでの浮動小数点ロードキャッシュアクセス待ち

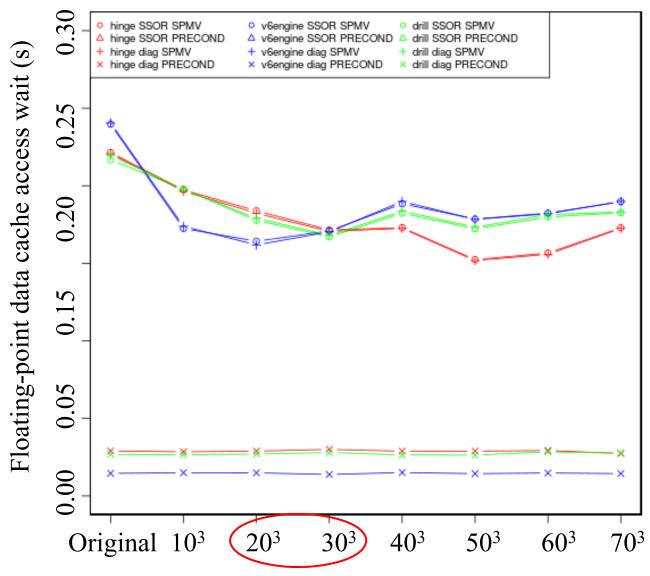


Fig. Floating-point data cache access waits for SpMV and Precond 20

京コンピュータでのメモリースループット

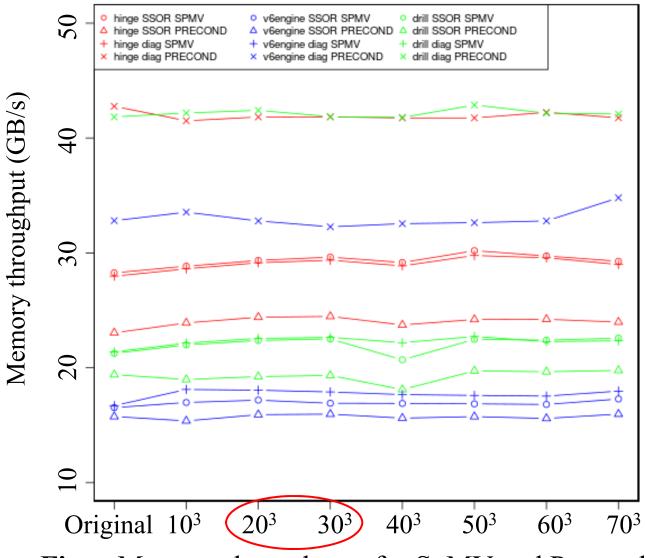


Fig. Memory throughputs for SpMV and Precond

Oakleaf-FX (FX10) での高速化率 (1/3)

Speed-up factors for SpMV and Precond (hinge model) **Table**

The number of blocks	Computation time (SpMV) [s]	Computation time (Precond) [s]	Iteration	Speed-up (SpMV)	Speed-up (Precond)
Original	13.18413	15.47871	2969		
10^{3}	12.97384	14.89413	2960	1.01312	1.03609
20^{3}	12.70465	14.67830	2923	1.02166	1.03819
30^{3}	12.65766	14.58390	2931	1.02826	1.04777
40^{3}	12.48992	14.60519	2916	1.03673	1.04089
50^{3}	12.42498	14.63354	2923	1.04465	1.04136
60^{3}	12.51178	14.68243	2909	1.03244	1.03292
-0^{3}	12.72032	14.94967	2914	1.01726	1.01620

- MPIのランク割り当ては1次元
- 計測回数は詳細プロファイラExcel対応のデフォルト値である7回プロセス0のプロファイル情報を出力

Oakleaf-FX (FX10) での高速化率 (2/3)

Table Speed-up factors for SpMV and Precond (v6engine model)

The number of blocks	Computation time (SpMV) [s]	Computation time (Precond) [s]	Iteration	Speed-up (SpMV)	Speed-up (Precond)
Original	1.041230	0.9512710	1013		
10^{3}	1.008217	0.9375578	1009	1.02866	1.01062
20^{3}	1.005861	0.9059998	1008	1.03005	1.04478
30^{3}	0.996272	0.8866609	1002	1.03377	1.06121
40^{3}	1.009953	0.8935171	1003	1.02079	1.05412
50^{3}	1.012137	0.9033293	1004	1.01960	1.04371
60^{3}	1.017749	0.9108773	1006	1.01600	1.03712
-20^{3}	1.020417	0.9233970	1009	1.01636	1.02611

- MPIのランク割り当ては1次元
- 計測回数は詳細プロファイラExcel対応のデフォルト値である7回
- ・ プロセス0のプロファイル情報を出力

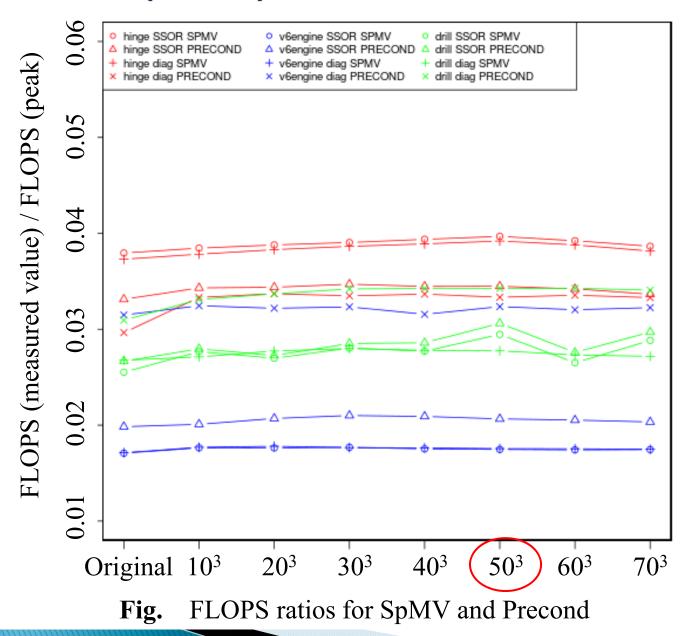
Oakleaf-FX (FX10) での高速化率 (3/3)

Table Speed-up factors for SpMV and Precond (drill model)

The number of blocks	Computation time (SpMV) [s]	Computation time (Precond) [s]	Iteration	Speed-up (SpMV)	Speed-up (Precond)
Original	56.28657	56.96856	6084		
10^{3}	53.57943	55.97189	5937	1.02514	0.99321
20^{3}	54.03896	56.58810	6101	1.04450	1.00953
30^{3}	54.32028	56.41487	6152	1.04777	1.02110
40^{3}	54.48738	55.79104	6147	1.04371	1.03167
50^{3}	51.63326	52.43852	6105	1.09388	1.09013
60^{3}	55.01965	55.89051	6123	1.02958	1.02582
$_{-}$ 70 ³	52.57910	53.85520	6136	1.07966	1.06685

- MPIのランク割り当ては1次元
- 計測回数は詳細プロファイラExcel対応のデフォルト値である7回
- プロセス0のプロファイル情報を出力

Oakleaf-FX (FX10)での浮動小数点演算ピーク比



Oakleaf-FX (FX10) でのL1Dミスdm率

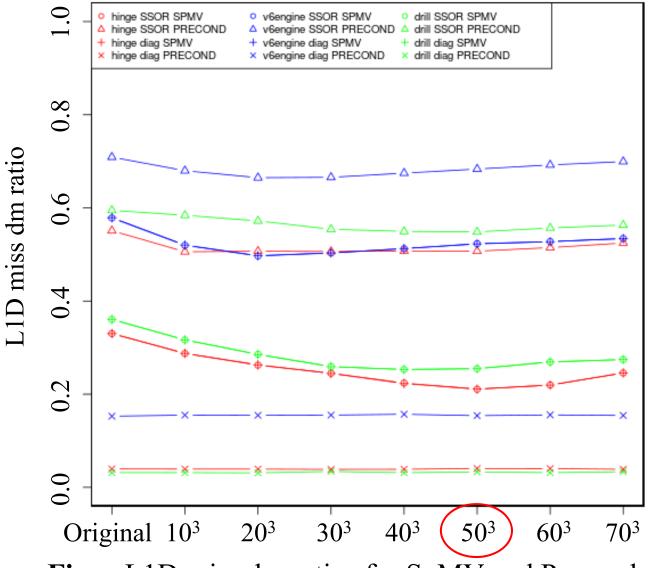


Fig. L1D miss dm ratios for SpMV and Precond

Oakleaf-FX (FX10) での浮動小数点ロードキャッシュアクセス待ち

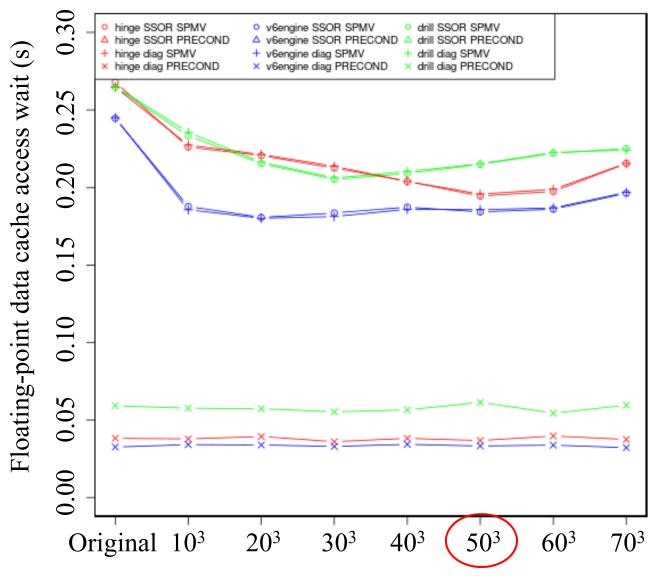
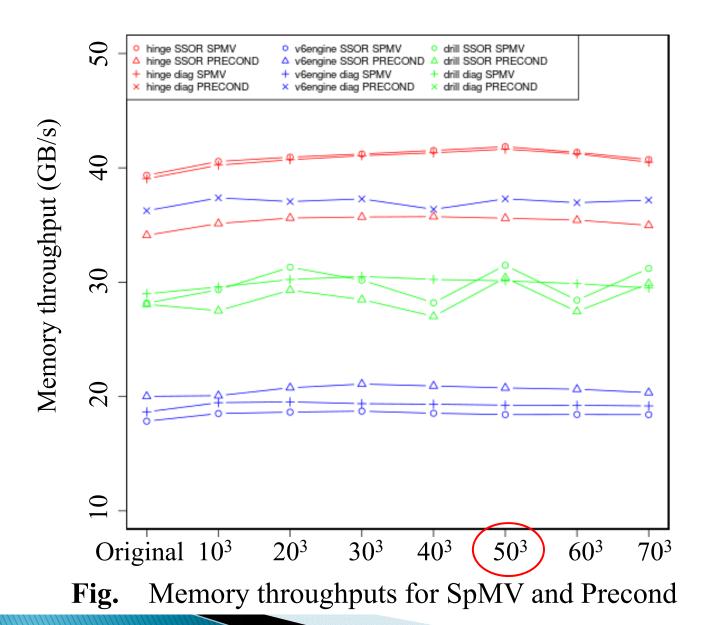


Fig. Floating-point data cache access waits for SpMV and Precond 27

Oakleaf-FX (FX10) でのメモリースループット



28

FX100での高速化率 (1/3)

Table Speed-up factors for SpMV and Precond (hinge model)

The number of blocks	Computation time (SpMV) [s]	Computation time (Precond) [s]	Iteration	Speed-up (SpMV)	Speed-up (Precond)
Original	8.570488	8.142538	2969		
10^{3}	8.565090	7.814994	2960	0.99759	1.03875
20^{3}	8.420197	7.684985	2923	1.00207	1.04312
30^{3}	8.448958	7.727957	2931	1.00140	1.04016
40^{3}	8.402116	7.767484	2916	1.00183	1.02957
50^{3}	8.429548	7.799688	2923	1.00096	1.02778
60^{3}	8.386102	7.856517	2909	1.00133	1.01546
-70^{3}	8.402823	7.956407	2914	1.00105	1.00443

- MPIのランク割り当ては1次元
- 計測回数は詳細プロファイラExcel対応のデフォルト値である11回
- プロセス0のプロファイル情報を出力

FX100での高速化率 (2/3)

Table Speed-up factors for SpMV and Precond (v6engine model)

The number of blocks	Computation time (SpMV) [s]	Computation time (Precond) [s]	Iteration	Speed-up (SpMV)	Speed-up (Precond)
Original	0.8248747	0.5990109	1014		
10^{3}	0.7999466	0.5997377	1009	1.02607	0.99386
20^{3}	0.8092155	0.5864903	1010	1.01532	1.01731
30^{3}	0.8147693	0.5719682	1002	1.00042	1.03488
40^{3}	0.8252256	0.5763482	1003	0.98873	1.02804
50^{3}	0.8111453	0.5940188	1005	1.00789	0.99945
60^{3}	0.8161315	0.5897645	1003	0.99974	1.00465
-0^{3}	0.8205803	0.5964112	1009	1.00027	0.99940

- MPIのランク割り当ては1次元
- 計測回数は詳細プロファイラExcel対応のデフォルト値である11回
- ・ プロセス0のプロファイル情報を出力

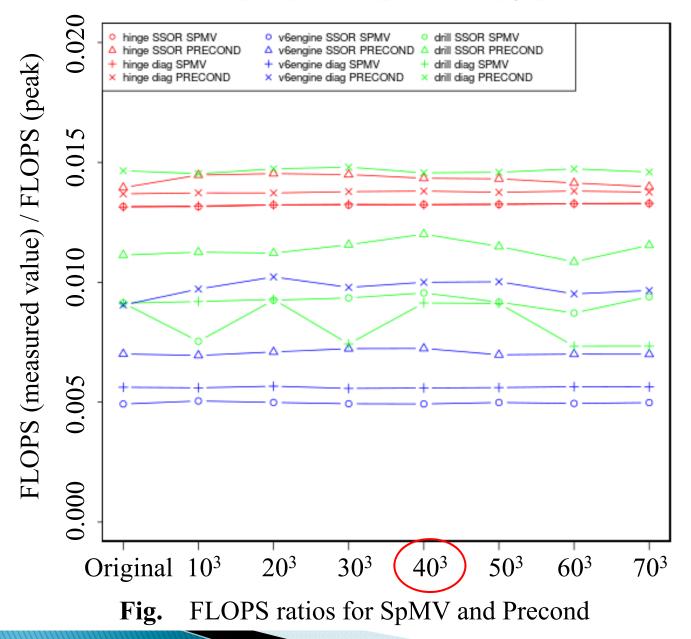
FX100での高速化率 (3/3)

Table Speed-up factors for SpMV and Precond (drill model)

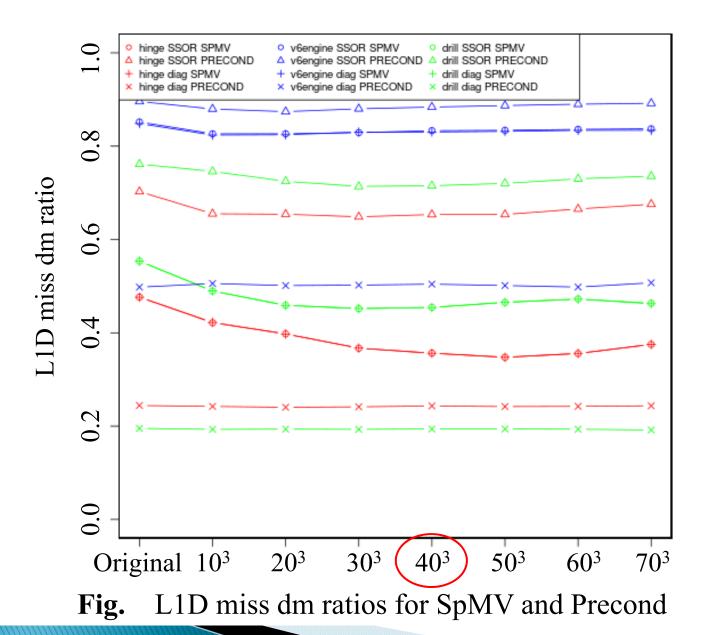
The number of blocks	Computation time (SpMV) [s]	Computation time (Precond) [s]	Iteration	Speed-up (SpMV)	Speed-up (Precond)
Original	38.06782	31.92686	6004		
10^3	45.73346	31.26471	5880	0.81519	1.00008
20^{3}	36.08755	30.38933	5850	1.02781	1.02364
30^{3}	37.64582	31.07888	6109	1.02889	1.04525
40^{3}	36.57927	29.72731	6119	1.06062	1.09456
50^{3}	36.57025	29.78124	5835	1.01164	1.04187
60^{3}	39.84197	32.66498	5844	0.93000	0.95135
70^{3}	38.44460	31.91789	6141	1.01279	1.02310

- MPIのランク割り当ては1次元
- 計測回数は詳細プロファイラExcel対応のデフォルト値である11回
- プロセス0のプロファイル情報を出力

FX100での浮動小数点演算ピーク比



FX100でのL1Dミスdm率



33

FX100での浮動小数点ロードキャッシュアクセス待ち

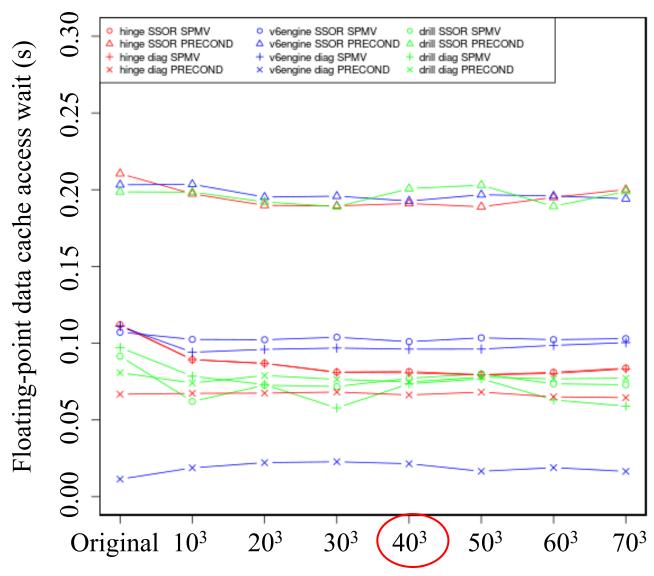


Fig. Floating-point data cache access waits for SpMV and Precond 34

FX100でのメモリースループット

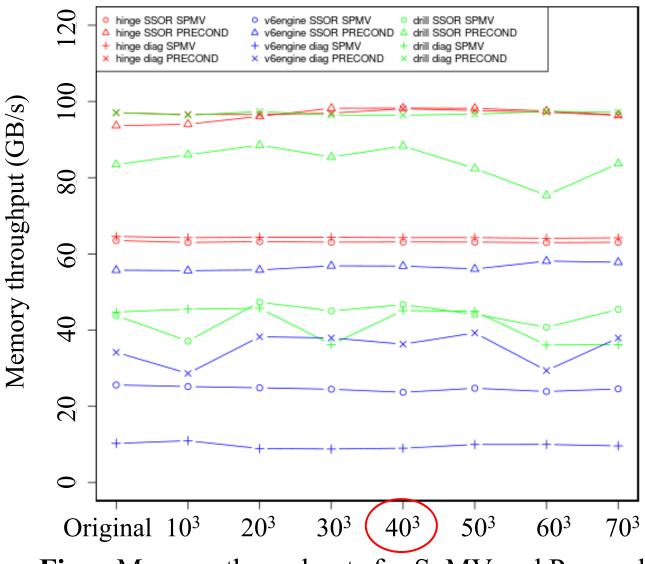


Fig. Memory throughputs for SpMV and Precond

Oakforest-PACSでの計測

- ジョブ実行リソースグループはregular-cacheとし、MCDRAMをキャッシュとして利用した
- 環境変数KMP_AFINITYはデフォルトのcompactとしたが、 scatterによる予備計算では性能の変化は見られなかった
- プロファイラとしてIntel Vtuneを用いた
- コンパイルオプションは、-O2 -qopenmp -g -axMIC-AVX512とした
- 実行コマンドは

mpiexec.hydra -np 4 -gtool "/opt/intel/vtune_amplifier_xe/bin64/amplxe-cl -finalization-mode=defferred -collect hpc-performance -knob collect-memory-bandwidth=true -r result:0" ./fistr1

として一回計測を行い、

amplxe-cl -report summary -r result.*.ofp

としてプロファイル情報の出力を行った

- Vtuneでは、SpMVとPrecondを分けて計算時間を計測することができないため、 プログラム全体での計測となった
- メモリバンド幅は、マニュアルによれば上記コマンドラインの-knob collect-memory-bandwidth=true で計測できることになっているが、出力は常に0となり計測できなかった

Oakforest-PACSでの高速化率 (1/3)

Table Speed-up factors for SpMV and Precond (hinge model)

The number of blocks	Computation time (SpMV) [s]	Computation time (Precond) [s]	Iteration	Speed-up (SpMV)	Speed-up (Precond)
Original	20.72320	36.06093	2969		
10^{3}	19.64763	35.82028	2963	1.05261	1.00468
20^{3}	20.49588	34.48725	2926	0.99644	1.03048
30^{3}	18.27435	32.51077	2931	1.11949	1.09500
40^{3}	20.07422	34.98708	2924	1.01668	1.01507
50^{3}	19.52406	35.80756	2918	1.04318	0.98977
60^{3}	18.86606	32.25540	2909	1.07624	1.09538
70^{3}	20.69605	35.36050	2915	0.98309	1.00126

- MPIのランク割り当ては1次元
- 計測回数は詳細プロファイラExcel対応のデフォルト値である11回
- ・ プロセス0のプロファイル情報を出力

Oakforest-PACSでの高速化率 (2/3)

Table Speed-up factors for SpMV and Precond (v6engine model)

The number of blocks	Computation time (SpMV) [s]	Computation time (Precond) [s]	Iteration	Speed-up (SpMV)	Speed-up (Precond)
Original	1.70737	2.63099	1012		
10^{3}	1.72820	2.57663	1013	0.98892	1.02211
20^{3}	1.64421	2.66779	1007	1.03328	0.98133
30^{3}	1.63229	2.32805	1002	1.03566	1.11896
40^{3}	1.69662	2.58516	1001	0.99540	1.00667
50^{3}	1.69001	2.40481	1004	1.00229	1.08541
60^{3}	1.65224	2.63259	1003	1.02418	0.99051
-0^{3}	1.66075	2.45627	1010	1.02604	1.06902

- MPIのランク割り当ては1次元
- 計測回数は詳細プロファイラExcel対応のデフォルト値である11回
- ・ プロセス0のプロファイル情報を出力

Oakforest-PACSでの高速化率 (3/3)

Table Speed-up factors for SpMV and Precond (drill model)

The number of blocks	Computation time (SpMV) [s]	Computation time (Precond) [s]	Iteration	Speed-up (SpMV)	Speed-up (Precond)
original	42.52692	136.66149	6079		
10^{3}	40.25543	127.69605	6022	1.04652	1.06017
20^{3}	38.17061	122.81535	5845	1.07124	1.06991
30^{3}	38.81141	122.38792	6146	1.10781	1.12893
40^{3}	40.01771	126.95975	6310	1.10308	1.11732
50^{3}	39.89915	125.75523	6114	1.07200	1.09298
60^{3}	41.70168	129.72520	6069	1.01811	1.05174
70^{3}	40.50043	127.16233	5829	1.00685	1.03050

- MPIのランク割り当ては1次元
- 計測回数は詳細プロファイラExcel対応のデフォルト値である11回
- ・ プロセス0のプロファイル情報を出力

現状のまとめ

▶ 演算性能向上の評価について

- ・京コンピュータ、Oakleaf-FX (FX10)、FX100、Oakforest-PACSにおいて、hinge、v6engine、drillモデルを用いて、節点データ並べ替え処理に対する反復法線形ソルバーの前処理および行列ベクトル積の性能を調査した
- L1D ミスdm 率 (1次データキャッシュミス数に対する, ロード・ストア 命令による1次データキャッシュミスの割合) が削減され, 浮動小数点 ロードキャッシュアクセス待ちも削減され, 結果としてメモリスループット が向上することを確認した

▶ ブロック分割数の調査について

- ・節点番号並べ替え処理における最適なブロック分割数は計算機に よって異なる
- データ全体を見ると、ブロック分割数30³では、多くの場合に3%程度の 高速化が得られた