Arm HPCプラットフォーム・ワークショップ
2021年8月26日（木）午後
15:55-16:20「Ampere Altraと各種HPC向けCPUとの性能比較」

謝辞
- Oracle Corporation提供のOracle Cloud Infrastructureを利用させていただきました
- 今野雅様にOpenFOAMについて多くのアドバイスをいただきました

大島 聡史（名古屋大学 情報基盤センター 准教授）
問い合わせ先 ohshima@cc.nagoya-u.ac.jp

Ampere Altraと各種HPC向けCPUとの性能比較

背景と趣旨

- サーバ・スパコン向けCPU としては従来よりx86系CPU、特にIntel社のXeonが非常に多く使われているが、近年は富士通社のA64FXやAMD社のEPYCの採用例も増えており、最新CPUとその性能や最適化技術への注目はかつてなく高まっている
- 我々が名古屋大学情報基盤センターにて昨年7月から運用しているスーパーコンピュータ「不老」もA64FXとXeonを搭載しており、多くのユーザに利用していただいている
- スーパーコンピュータを調達しサービス提供する立場としても、自ら研究に利用する立場としても、最新のCPUの構成・性能・活用方法には注目している

- 新しいARM系CPUであるAmpere Altraを利用する機会を得たため、その性能をA64FX等のCPUと比較し報告する
- あわせて、現在の国内のスパコンでは採用例は多くないが注目は高い（と思う）EPYC (Rome, Milan)も利用する機会を得たため、あわせて比較して報告する

※ 性能値は5月に開催されたHPC研究会で発表した内容と同一です

対象環境

- Oracle Corporationに提供いただいたクラウド環境と名大の「不老」を用いた

CPU	搭載システム	1CPUあたり コア数（スレッド数）	動作周波数	1CPUあたり理論演算性能	搭載メモリ	ノード内 ソケット数	TDP
Ampere Altra	Oracle Cloud Infrastructure	80 (80)	3.00 GHz	1.920 TFLOPS (80c*3.0GHz*8Flop/cycle)	DDR4 3200 MHz 500 GB/socket (最大 4TB) 8channels 204.8 GB/s/socket	2	210W
EPYC 7742 (EPYC Gen 2, Rome)	Oracle Cloud Infrastructure	64 (128)	2.25 – 3.40 GHz	2.304 TFLOPS (64c*2.25GHz*16Flop/cycle)	DDR4 3200 MHz 1 TB/socket (最大 4TB) 8channels 204.8 GB/s/socket	2	225W
EPYC 7J13 (EPYC Gen 3, Milan)	Oracle Cloud Infrastructure	64 (128)	2.55 – 3.50 GHz	2.6112 TFLOPS (64c*2.55GHz*16Flop/cycle)	DDR4 3200 MHz 1 TB/socket (最大 4TB) 8channels 204.8 GB/s/socket	2	280W? ※2
A64FX	スーパーコンピュータ 「不老」 Type I サブシステム	48 (48) + アシスタントコア	2.20 GHz	3.3792 TFLOPS (48c*2.2GHz*32Flop/cycle)	HBM2 32 GB/CPU (8 GB/CMG) 1,024 GB/s/CPU	1	200W
Xeon Gold 6230 (Cascade Lake, CLX)	スーパーコンピュータ 「不老」 Type II サブシステム	20 (40) ※1	2.10 – 3.90 GHz	1.344 TFLOPS (20c*2.10GHz*32Flop/cycle)	DDR4 2933 MHz 192 GB/socket (最大 1TB) 6channels 140.784 GB/s/socket	2	125W

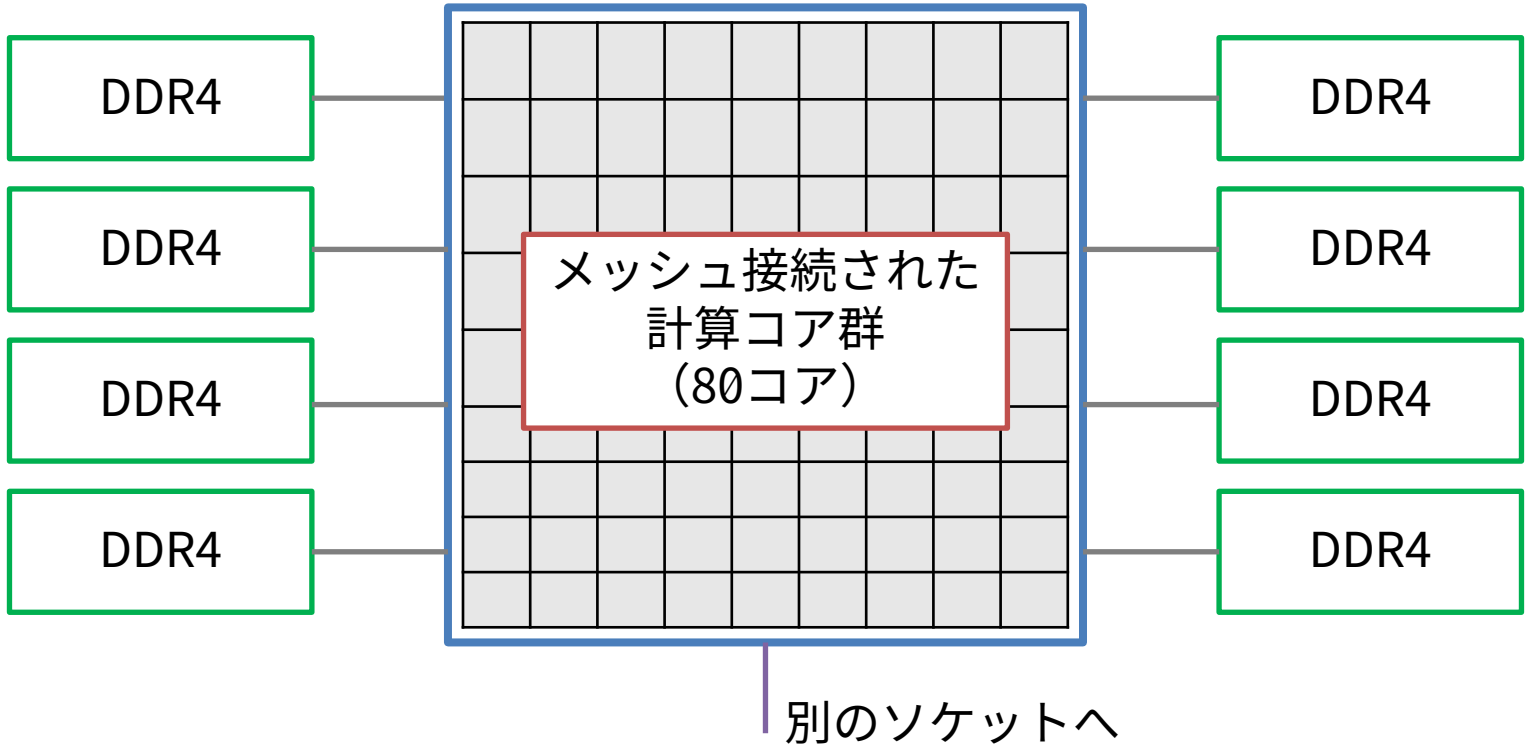
※1 ただしHTTは無効化設定してある

※2 一般向けモデルではないため情報がなく、近いスペックのモデルのTDPを参照した

Ampere Altra

- 概要と基本構成
  - Ampere Computing社の開発したCPU、いわゆるサーバ・クラウド・データセンタ向けのCPU
  - Arm v8.2+、64bit CPU、3.00GHz、80コア、TSMC 7nm FinFET、2*128bit SIMD、210W
    - 3.30GHzや64コアなどのバリエーションもあり
  - 対応メモリ：DDR4メモリ（DDR4-3200）、8チャネル
  - 1ノードあたり1ソケットまたは2ソケットを想定
    - 1ソケット Mt. snowと2ソケット Mt. Jadeの2プラットフォーム（ラックマウントサーバ）を展開
- 理論性能
  - 3.0GHz * 80コア * 8 命令同時実行可能 = 1.920TFLOPS
  - DDR4-3200、1ソケットあたり8チャネル
    - DDR4-3200=25,600MB/s、×8チャネルなので25.600GB/s*8=204.8GB/s/socket
    - メインストリームなメモリ、チャネル数が多い分だけ高性能
    - HBM系と比べれば低速だが最大搭載可能容量が多い（最大4TB/socket）

Ampere Altra：おおよその構成

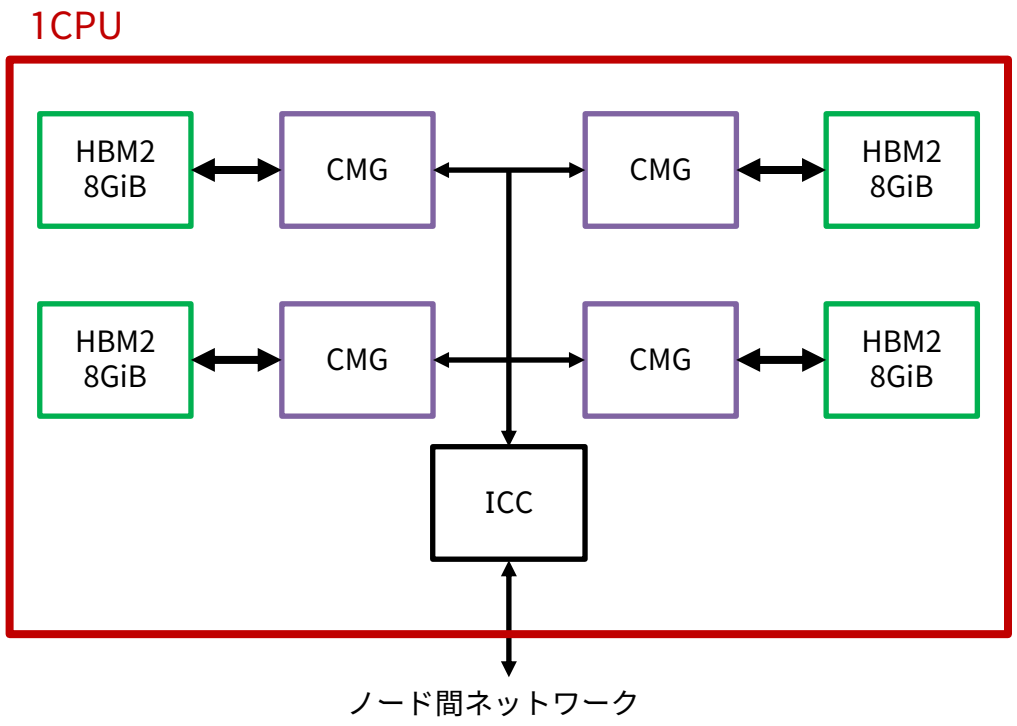


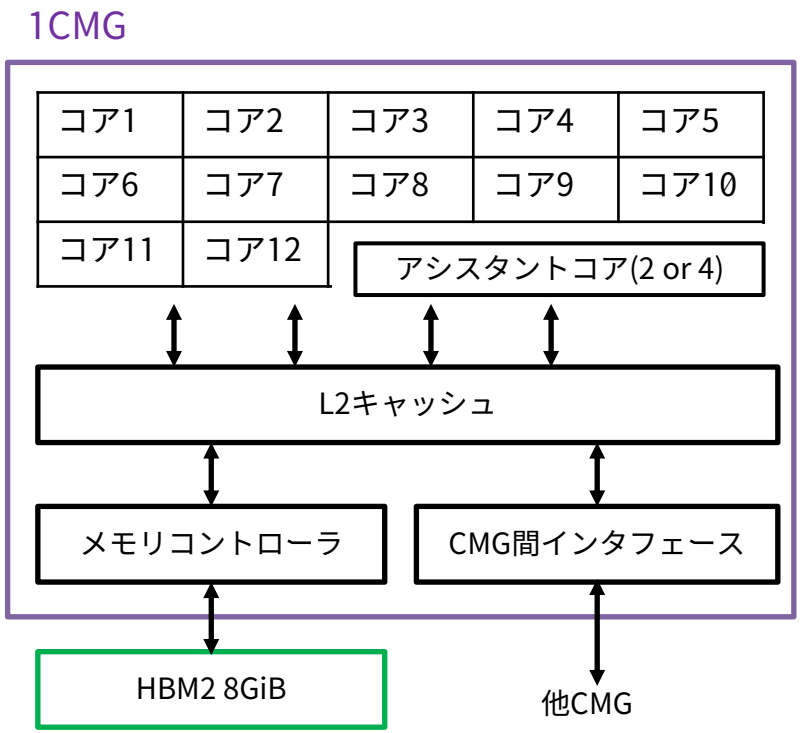
- 64 KB L1 I/D-cache per core
- 1MB L2 cache per core
- 32 MB shared cache per socker

- 計算コアはメッシュ接続（どのようなメッシュ形状かという情報は未発見）
- A64FXのように明確に不均質な構成ではない、NUMA構成も1ソケット=1NUMAノードのみ
- ただしホームノードが32に分かれているという情報は公開されている
- 資料

https://amperecomputing.com/altra/
https://www.servethehome.com/ampere-altra-80-arm-cores-for-cloud/

A64FX



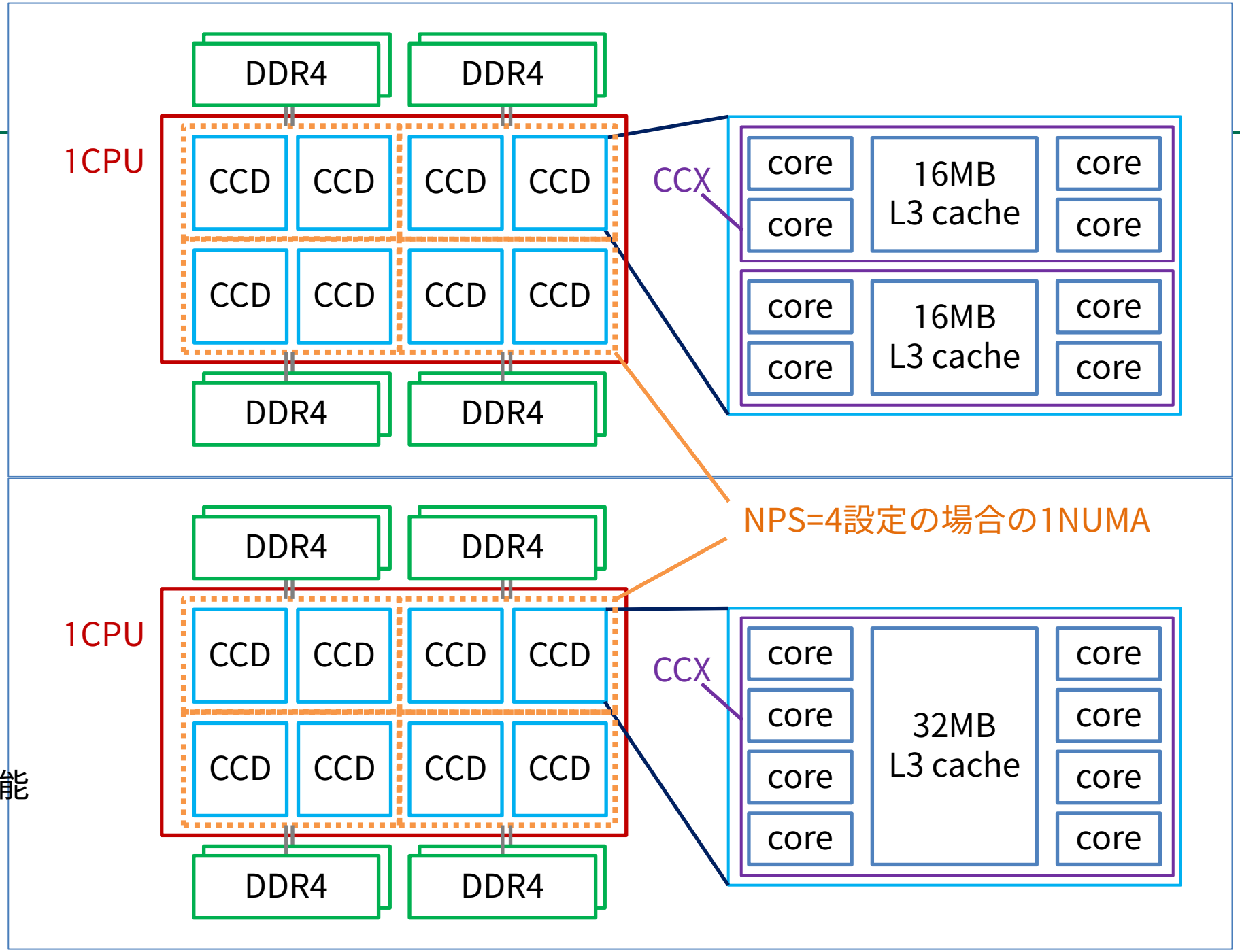


- Ampere Altra同様にARM系のCPUだが、HPC向けに拡張されている（Armv8.2-A+SVE）
- Ampere Altraと異なり、ノード内のCPU（とHBM）が明確に4つに分かれている
- 「1CMG+8GB HBM」が1NUMAノード単位

EPYC (Rome, Milan)

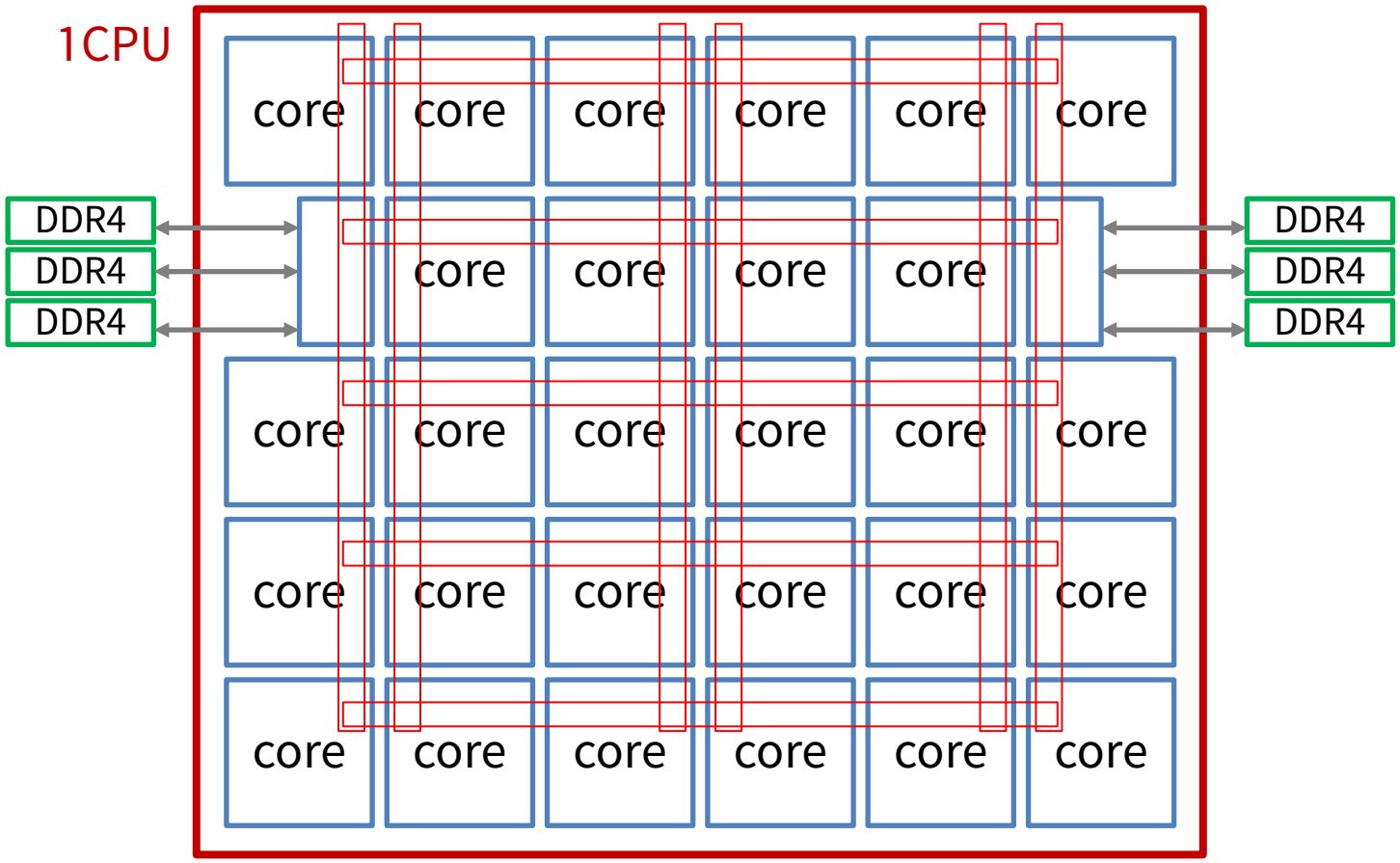
EPYC Gen.2
"Rome"

EPYC Gen.3
"Milan"



- チップレット技術による
  階層的なコア構成
- いくつかのNUMA構成が可能
  - 1CPU=1NUMA
  - 2CCD=1NUMA　など
- 間引いた構成で製品展開

Xeon Cascade Lake



- メッシュ接続のフラットなコア構成（赤い細線がメッシュ接続の構成に対応）
- NUMA構成は1CPU=1NUMAノード
  - 厳密には均等距離ではないのは明らかだが、全コア全メモリが均等という想定で使う

対象環境（再掲）

- Oracle Corporationに提供 た ド環境 「不老」

多い　速い　低め　速さはそれなり、容量は多め

CPU	搭載システム	1CPUあたり コア数（スレッ 数）	動作周波数	1CPUあたり理論演算性	搭載メモリ	ノード内 ソケット数	TDP
Ampere Altra	Oracle Cloud Infrastructure	80 (80)	3.00 GHz	1.920 TFLOPS (80c*3.0GHz*8Flop/cycle)	DDR4 3200 MHz 500 GB/socket (最大 4TB) 8channels 204.8 GB/s/socket	2	210W
EPYC 7742 (EPYC Gen 2, Rome)	Oracle Cloud Infrastructure	64 (128)	2.25 – 3.40 GHz	2.304 TFLOPS (64c*2.25GHz*16Flop/cycle)	DDR4 3200 MHz 1 TB/socket (最大 4TB) 8channels 204.8 GB/s/socket	2	225W
EPYC 7J13 (EPYC Gen 3, Milan)	Oracle Cloud Infrastructure	64 (128)	2.55 – 3.50 GHz	2.6112 TFLOPS (64c*2.55GHz*16Flop/cycle)	DDR4 3200 MHz 1 TB/socket (最大 4TB) 8channels 204.8 GB/s/socket	2	280W? ※2
A64FX	スーパーコンピュータ「不老」 Type I サブシステム	48 (48) + アシスタントコア	2.20 GHz	3.3792 TFLOPS (48c*2.2GHz*32Flop/cycle)	HBM2 32 GB/CPU (8 GB/CMG) 1,024 GB/s/CPU	1	200W
Xeon Gold 6230 (Cascade Lake, CLX)	スーパーコンピュータ「不老」 Type II サブシステム	20 (40) ※1	2.10 – 3.90 GHz	1.344 TFLOPS (20c*2.10GHz*32Flop/cycle)	DDR4 2933 MHz 192 GB/socket (最大 1TB) 6channels 140.784 GB/s/socket	2	125W

※1 ただしHTTは無効化設定してある

※2 一般向けモデルではないため情報がなく、近いスペックのモデルのTDPを参照した

対象ベンチマーク

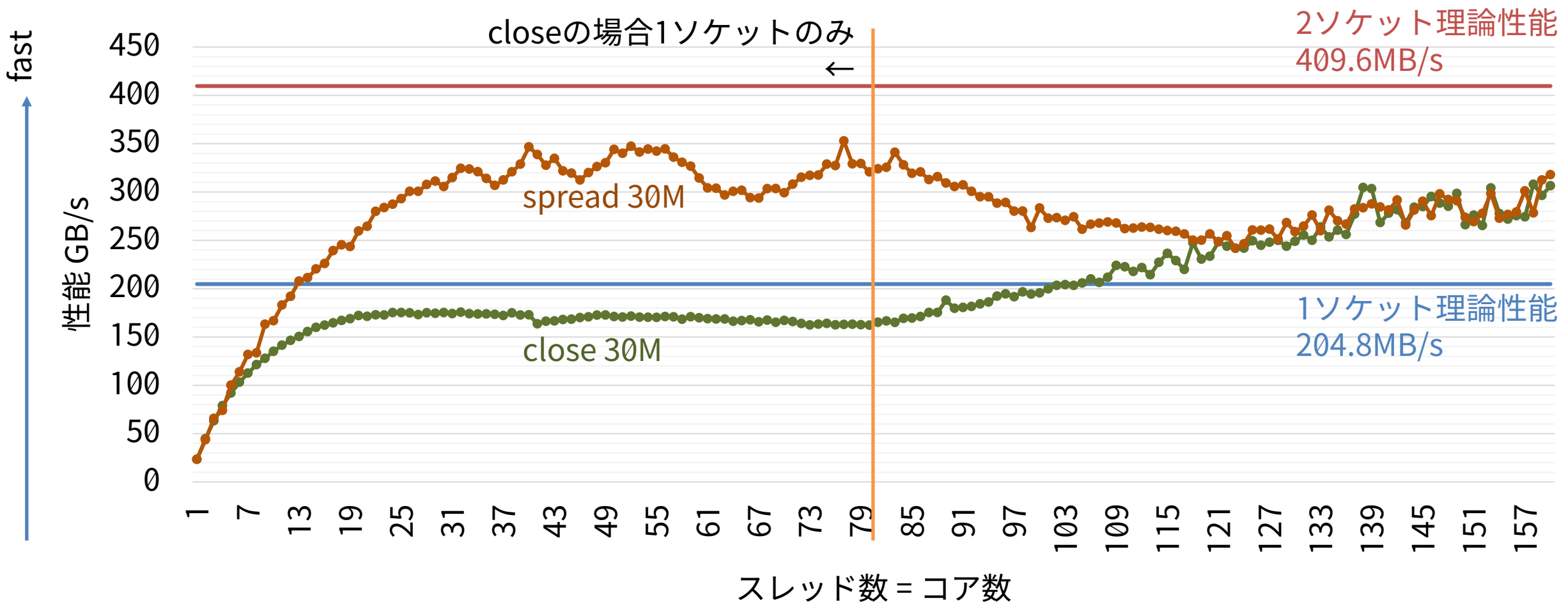
- HPC分野における基本的なものと我々がよく用いているものを利用

- メモリ転送性能
  - STREAMベンチマーク（STREAM Triad）
    https://www.cs.virginia.edu/stream/
- 密行列演算性能
  - 単純な行列積和演算（BLAS DGEMM）
  - HPLベンチマーク
    https://www.netlib.org/benchmark/hpl/
- 疎行列演算性能
  - 単純な行列ベクトル積演算（SpMV）
  - HPCGベンチマーク
    https://www.hpcg-benchmark.org/
- アプリケーション性能
  - GKVベンチマーク
  - OpenFOAM

コンパイラ・ライブラリなど

- Ampere Altra
  - GCC 9.3.1　（yumで導入したdevtoolsetのGCC）
  - Free Arm Performance Libraries (ARMPL) 20.3　BLASやLAPACKなどのライブラリ一式
  - OpenMPI 3.1.3　yumで導入したもの
- A64FX
  - 「不老」でユーザに提供している富士通TCS環境（Fujitsu Technical Suite tcsds-1.2.31）
- CLX
  - 「不老」でユーザに提供しているIntel環境（Intel Compiler + Intel MPI）
    - `intel/2020.4.304 module`で提供しているもの
- EPYC
  - GCC 10.2.1　（yumで導入したdevtoolsetのGCC）
  - Intelコンパイラ 2001.1.1, 2021.1.2（oneAPI Toolkits）
  - AMD Optimizing C/C++ Compiler (AOCC)　STREAMベンチマーク向け
- 標準的な推奨最適化オプションでコンパイル（-O3 -march=nativeなど）

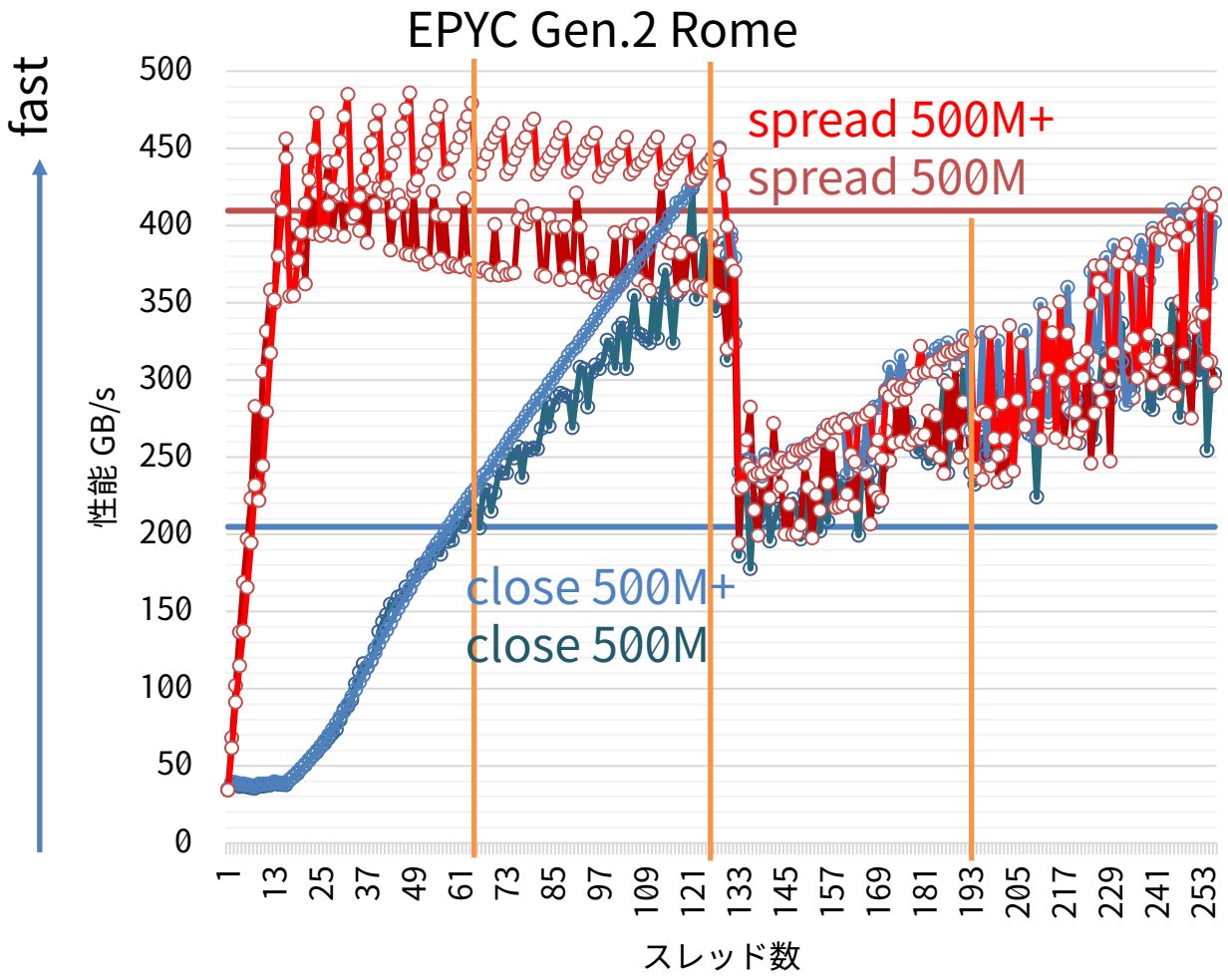
STREAM：Ampere Altra

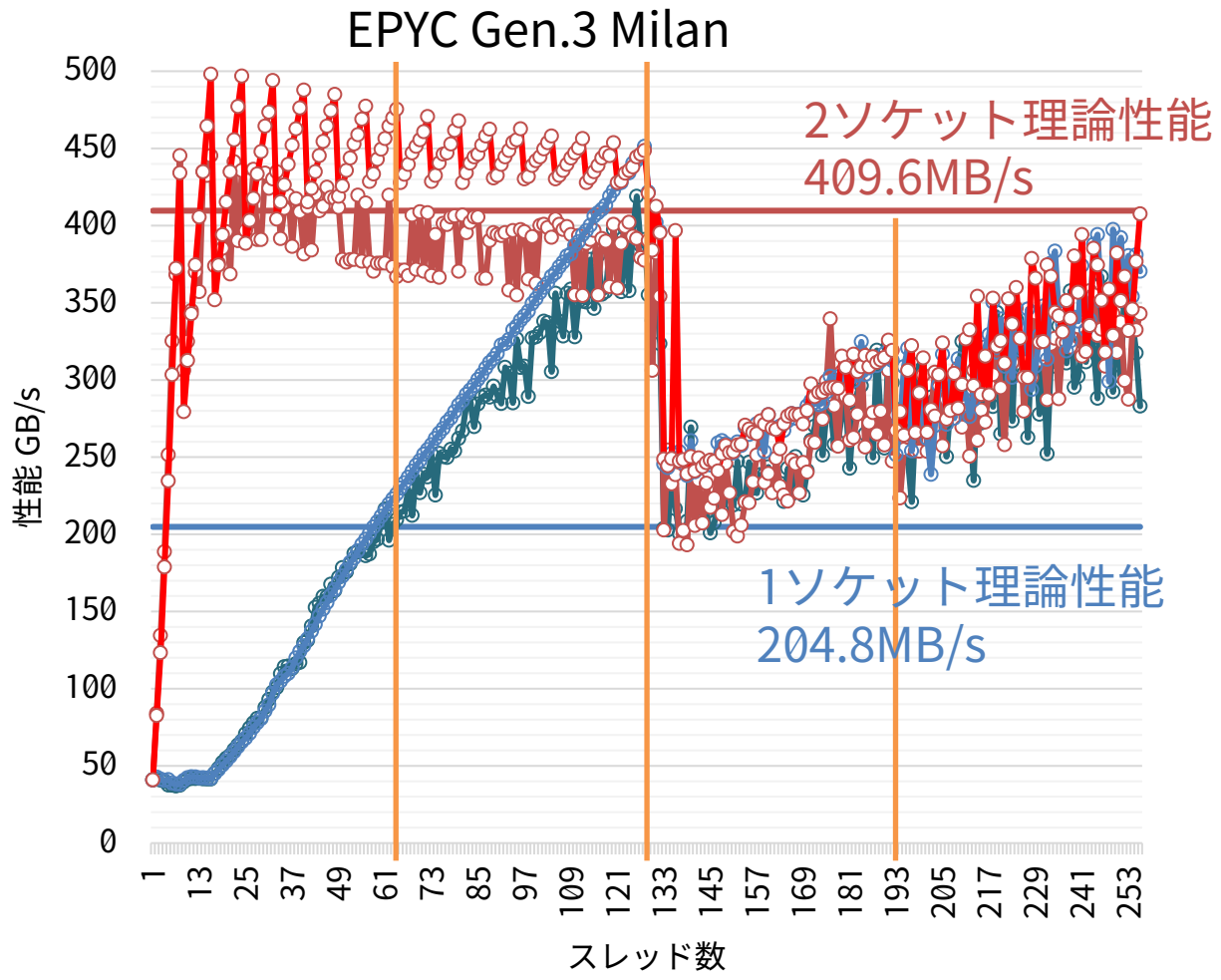


- 環境変数OMP_PROC_BINDでスレッド配置を指定
- closeは片方のソケットを埋めてからもう一方のソケットへスレッドを配置、spreadは均等にスレッドを配置
- 30Mは転送配列の長さ（30,000,000）

- 全体的にスレッド数が少なめの時点では良好だが多コア時にいまひとつ
- 多スレッドのハンドリング能力があまり良くない？
- なにかチューニングのコツがある？

STREAM：EPYC、NPS=4





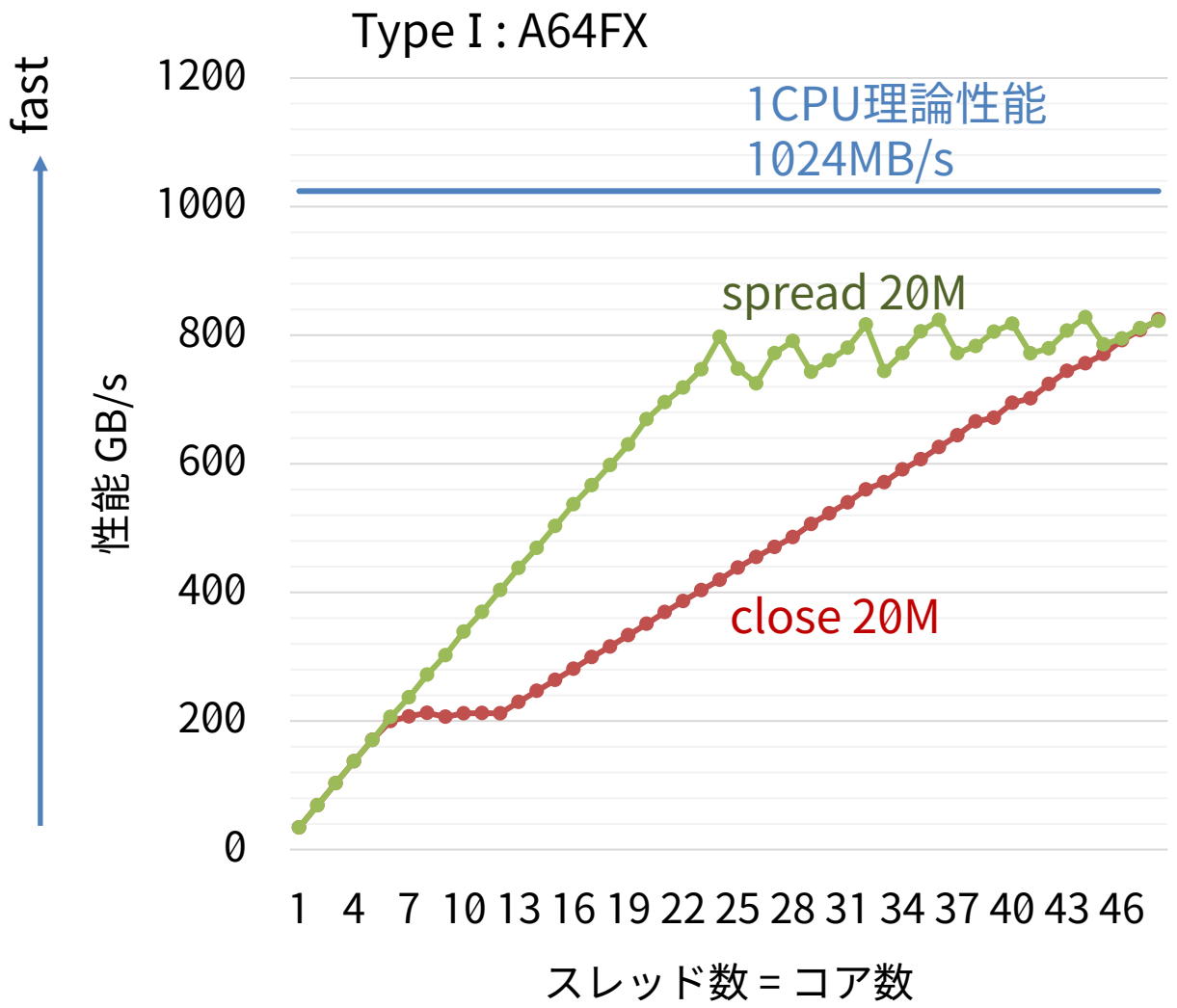
- 500M+は環境設定を最適化したもの、詳細は次ページ
- いくつかの問題サイズを試した中で妥当な性能の傾向のものを選択。性能の傾向は妥当なのだが、最大値が理論値を超えてしまっていて変。

STREAM：EPYC　チューニング

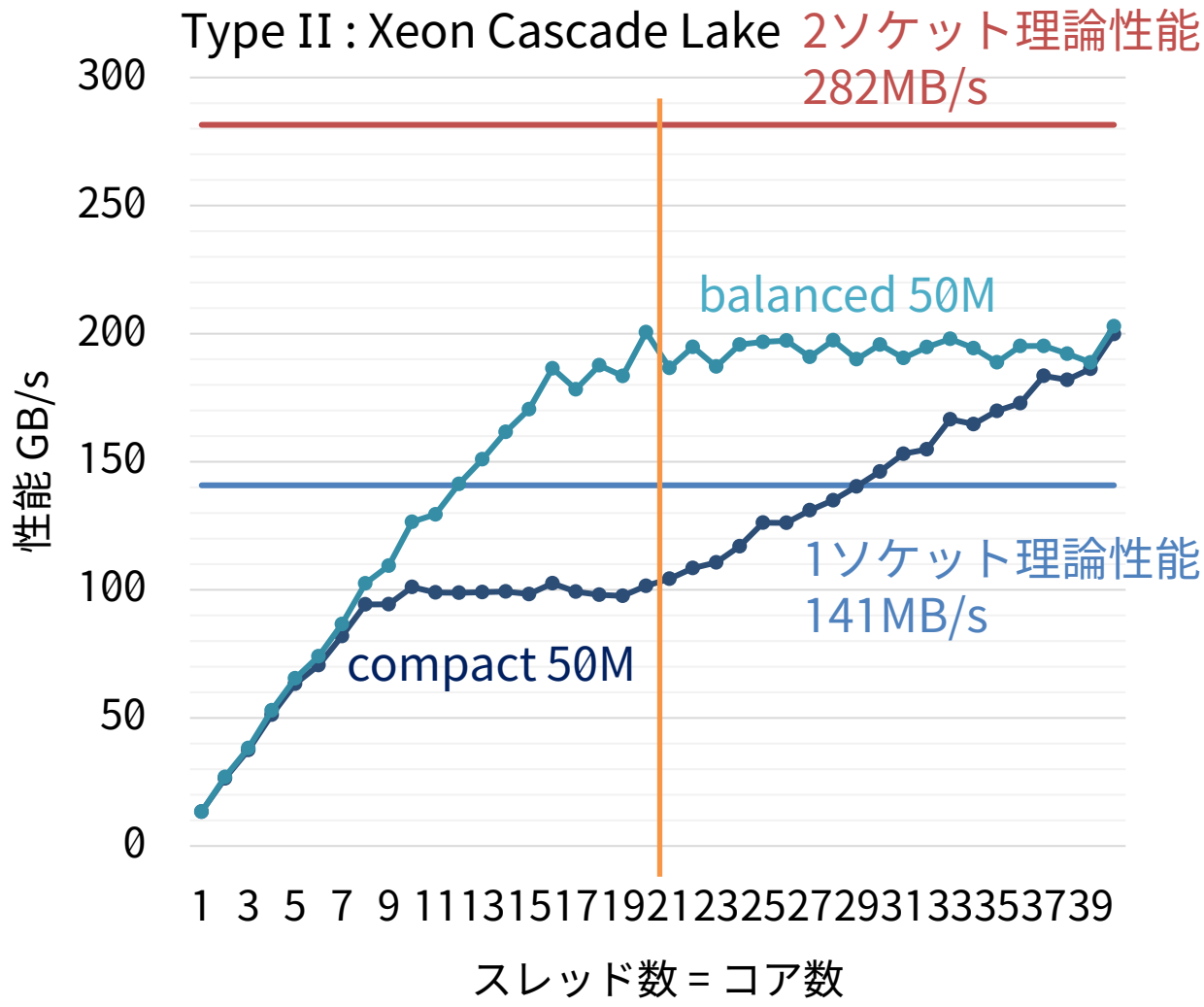
- NUMA構成は2CCD=1NUMA (NPS=4設定)
  - 全体で1NUMAの場合とNUMA設定だけでは大きな差はなし
- AMDの公開している資料を参考に調整
  - High Performance Computing (HPC) Tuning Guide for AMD EPYC 7002 Series Processors
  - High Performance Computing (HPC) Tuning Guide for AMD EPYC 7003 Series Processors
    - 実はSTREAMやHPLの性能がしっかり公開されている（が、その性能が再現できるかは……）
- Streaming Storeのオプションが効果大
  - AMD Optimizing C/C++ Compiler (AOCC)の-fnt-storeオプション
  - GCCには対応するオプションがない
  - Intelコンパイラでは-qopt-streaming-stores always相当のはずだが設定しても向上しなかった
- さらにシステム設定を調整することで性能向上（500M+はこの設定を適用）
  - 他のプログラムでも有効な可能性はあるが、今回ではSTREAMにのみ適用

```
echo 0 > /proc/sys/kernel/randomize_va_space
echo 0 > /proc/sys/vm/nr_hugepages
echo 0 > /proc/sys/kernel/numa_balancing
echo 'never' > /sys/kernel/mm/transparent_hugepage/enabled
echo 'never' > /sys/kernel/mm/transparent_hugepage/defrag
```

STREAM：スーパーコンピュータ「不老」



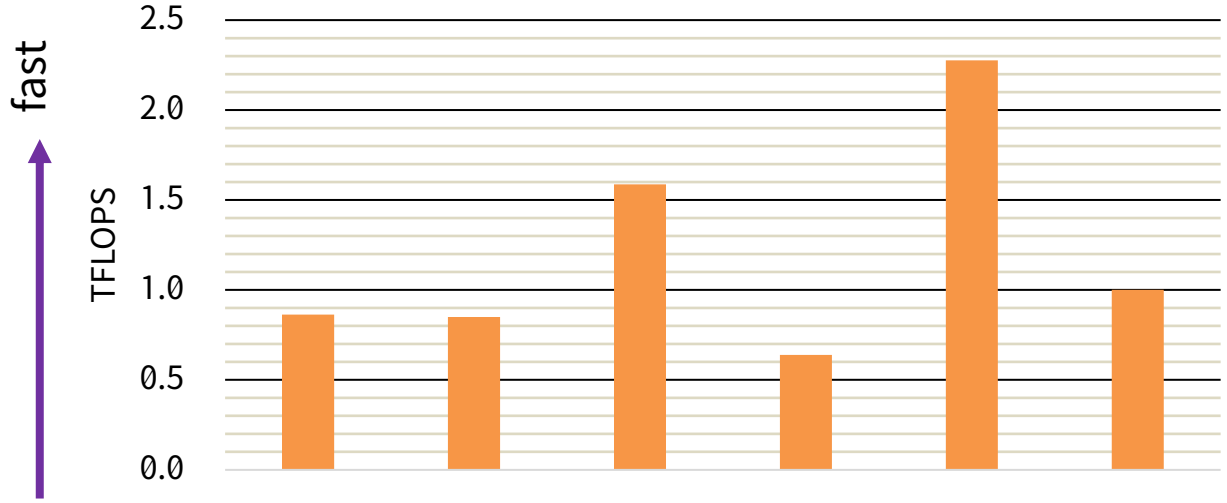
- 素直な傾向、最大800MB/s超の高い性能
- zfillやdemandページングの設定が重要



- 傾向は妥当、乱高下はしておらず安定
- 理論値に対する性能比はややもの足りないか

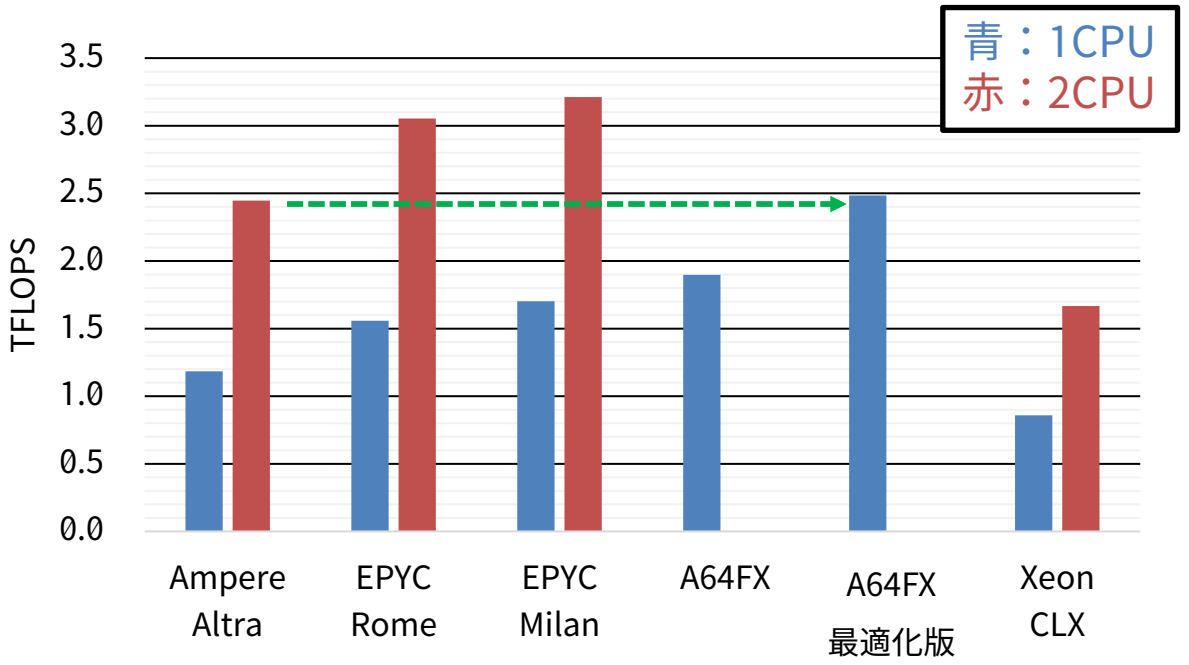
密行列演算性能：密行列積和演算（DGEMM）とHPLベンチマーク

- DGEMM：各環境向けのBLASライブラリのcblas_dgemm、2048×2048
  - Ampere AltraはFree Arm Performance Libraries
  - EPYCはAOCL（MKLもほぼ同性能だった）



	Ampere Altra 1CPU	EPYC Rome 1CPU	EPYC Milan 1CPU	A64FX 1CMG	A64FX 1CPU	Xeon CLX 1CPU
理論演算性能比 %	44.9	36.9	60.8	75.6	67.4	74.5

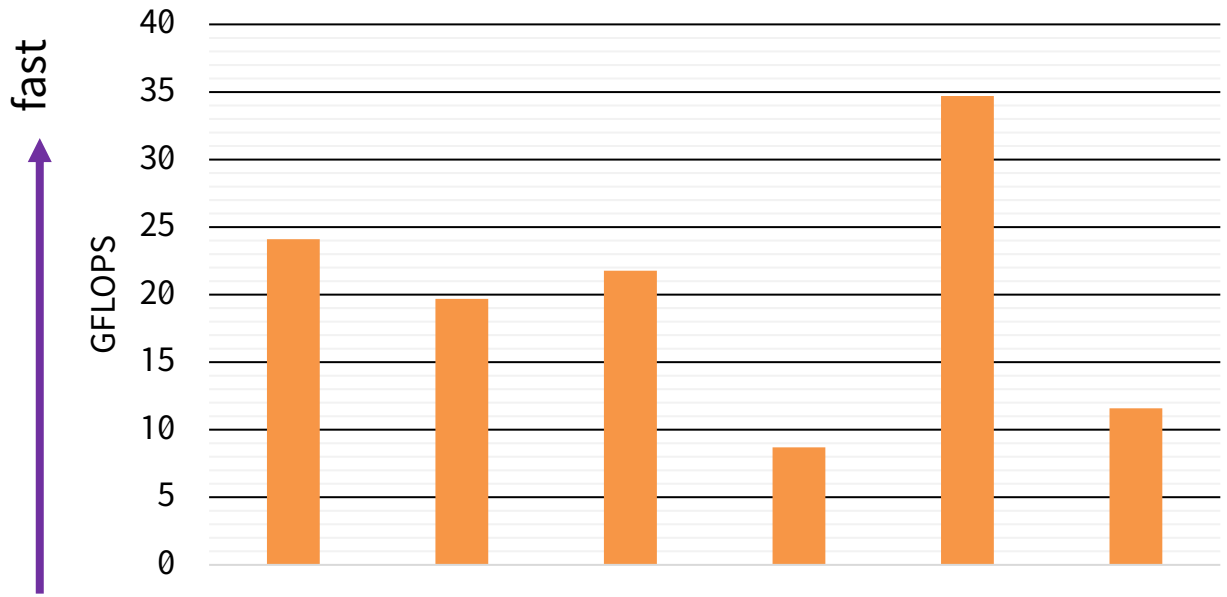
- HPL：HPL-2.3、A64FX最適化版以外はコードチューニングなし、いくつか実行時パラメタを試したなかで速かったもの
  - AMDの資料によるとEPYCは2ソケットで3.8TF以上でるらしい



	Ampere Altra	EPYC Rome	EPYC Milan	A64FX	A64FX 最適化版	Xeon CLX
1CPU	61.7	67.6	65.2	56.1	73.5	63.9
2CPU	63.7	66.3	61.5			62.0

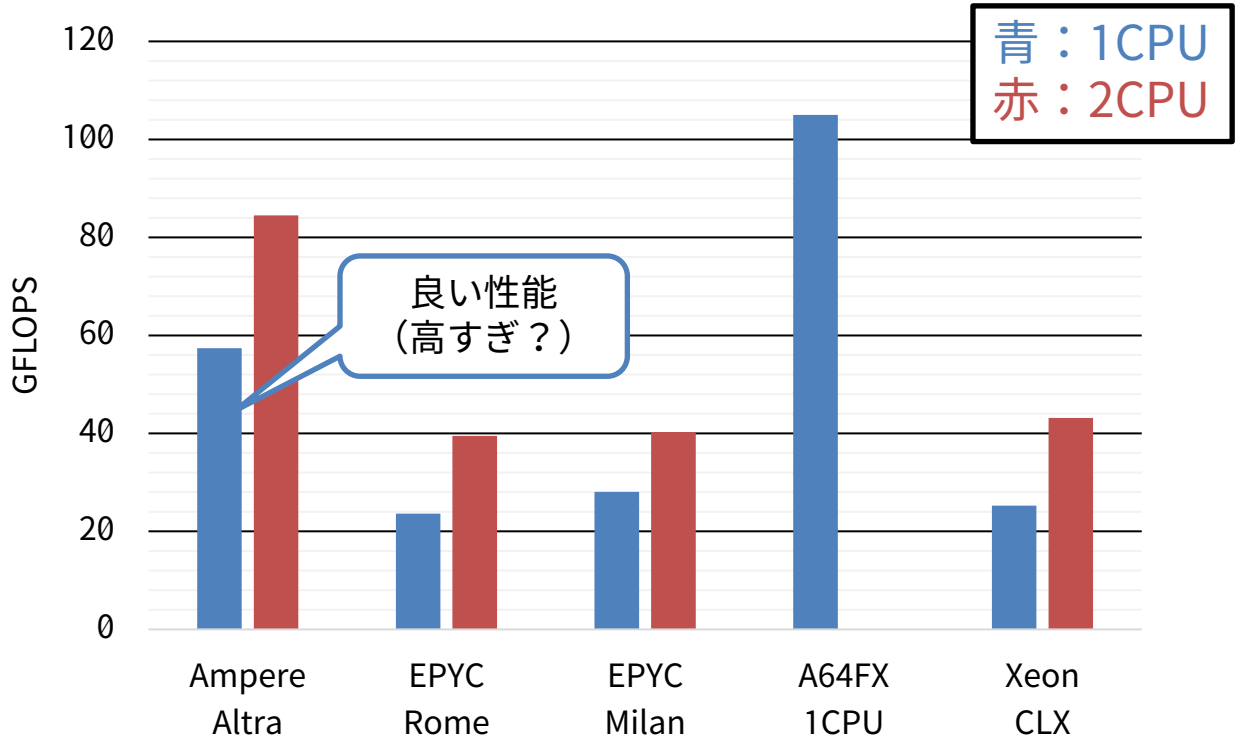
疎行列演算性能：疎行列ベクトル積（SpMV）とHPCG

- SpMV：単純なCRS形式SpMV、約1.4GB、300x300x300の3次元7点ステンシル相当
  - （演算性能比%を記したが）基本的にメモリ性能依存



	Ampere Altra 1CPU	EPYC Rome 1CPU	EPYC Milan 1CPU	A64FX 1CMG	A64FX 1CPU	Xeon CLX 1CPU
理論演算性能比 %	1.26	0.85	0.83	1.03	1.03	0.86

- HPCG 3.1： A64FXは富士通による最適化版、他は配布コードそのまま（OpenMP指示文の問題だけ修正）、プロセスやスレッドの数を幾つか試した最速値



	Ampere Altra	EPYC Rome	EPYC Milan	A64FX 1CPU	Xeon CLX
1CPU	3.0	1.0	1.1	3.1	1.9
2CPU	2.2	0.9	0.8		1.6

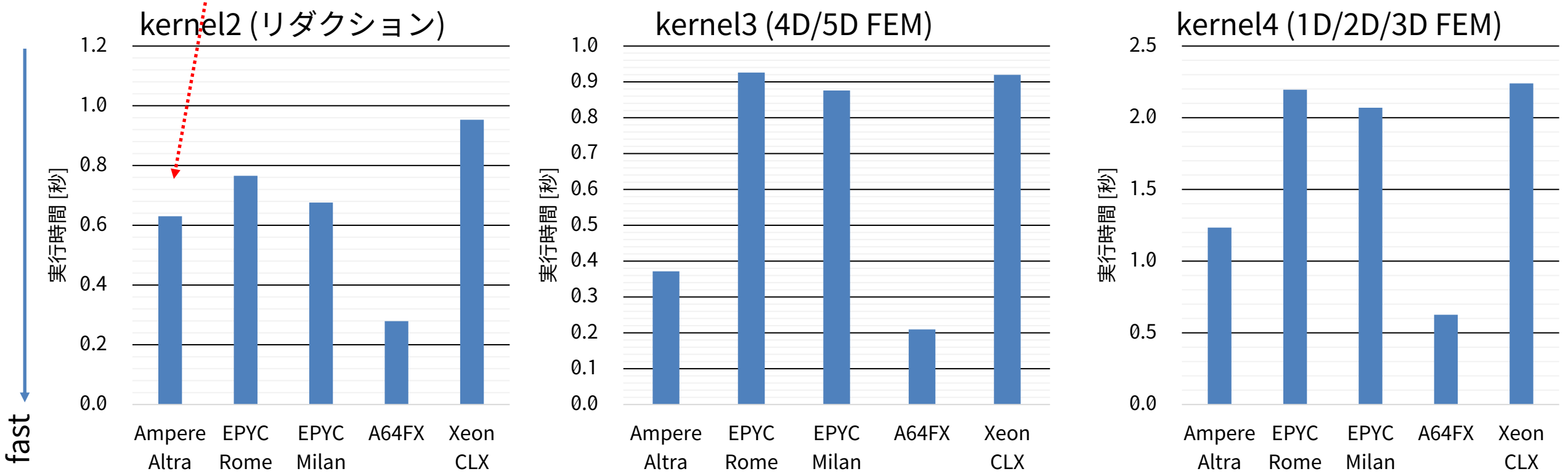
アプリケーション性能

- GKVベンチマーク
  - 磁場閉じ込め核融合に向けたプラズマ乱流現象の解析のために核融合科学研究所にて開発されたプラズマ乱流解析コード GKV (GyroKinetic Vlasov code) を元に、名古屋大学の片桐・渡邊らがそのカーネル部分を抜き出して作成したベンチマーク。名大ではスパコン調達等にも利用。
  - 4つのプログラムから構成される
    1. FFTとMPI_Alltotll、MPI+OpenMP並列化コード
    2. 配列のリダクション、OpenMP並列化コード
    3. 4次元および5次元の有限差分法カーネル、OpenMP並列化コード
    4. 1次元, 2次元, 3次元の有限差分法カーネル、OpenMP並列化コード

- OpenFOAM
  - 流体計算向けのOSS、MPI並列
  - 様々なシステムで測定実績のある「OpenCAE学会チャネル流ベンチマーク」を実行

GKVベンチマーク

- 全体的に問題が小さく2CPU使っても性能が向上しにくいため、1CPUの結果のみ示す
- Ampere Altra kernel2は80スレッドより64スレッドの方が高速。kernel3,4と比べてEPYCやCLXとの差が小さい。
  - 多スレッドのリダクションがやや不得意か？
  - べき乗数でないリダクションは苦手か？
- EPYC：NPS4の影響はあまりなさそうだが、numactlでメモリをインターリーブ使用しないと倍遅い点には注意が必要。
- A64FXは他を寄せ付けない高い性能を発揮



OpenFOAM、オープンCAE学会チャネル流ベンチマーク

- 様々なスパコン・サーバで測定した実績のあるベンチマーク
  - https://gitlab.com/OpenCAE/OpenFOAM-BenchmarkTest
  - 過去のHPC研究会などでも何度か用いられている
  - コンパイラオプションのバリエーションやプロセスの数と配置のバリエーションをいくつか試して測定した際の最速値
  - 問題設定→

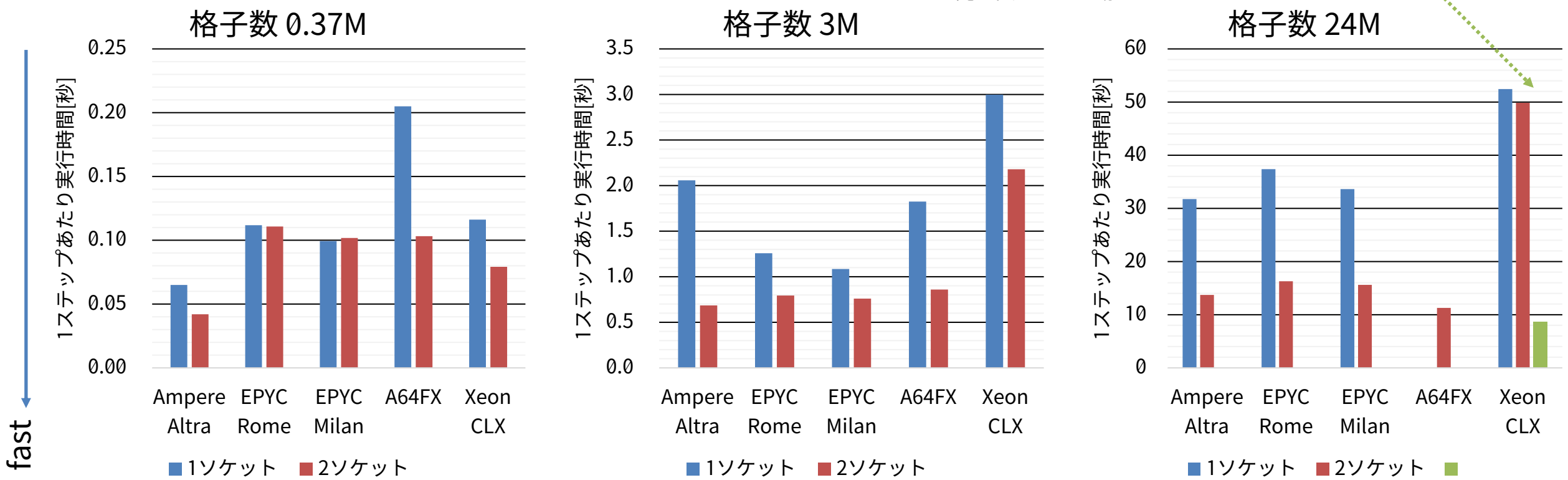
レイノルズ数 Re_τ	110
主流方向	一定の圧力勾配
主流・スパン方向	周期境界
ソルバ	pimpleFoam
乱流モデル	無し(laminar)
領域分割手法	scotch（周期境界面は同領域）
速度線形ソルバ	BiCG（前処理DILU）
圧力線形ソルバ	PCG（前処理DIC）
格子数	約0.37M, 約3M, 約24M

結果

- 問題サイズにより傾向に差あり。ある程度時間がかかる24Mに注目すると、Ampere Altraは良好な性能。主要計算カーネルがメモリ性能律速なので妥当性のある結果。
- A64FXがHWスペックの割に遅いのが気になる
  - A64FXは専用に最適化を行う余地があるのに対して、Ampere Altraはそうでもないのかもしれない

- 測定範囲だが、decompose（領域分割）にAmpere AltraはEPYCの1.5倍程度の時間がかかる。A64FXも遅く、ARMに向いていない実装となっている模様。

※CLXはコア数より多くのプロセス（2ソケット合計最大128プロセス）で実行したら性能向上してしまった。疑わしいのだが、ログを見る限り正しく動いている……。



まとめ

- Ampere Altraの性能を様々なCPUと比較した
  - 同じARM系のA64FXと比べると理論性能どおりの低い性能（不当に低いわけではない、ノードあたりのアプリ性能ではA64FXに匹敵・追い越すものもあり優れた性能と言えそう）
  - x86系CPUと比べて遜色ない性能だが、プログラムによって性能が出やすいもの・出にくいものはある
    - 例：kernelにより差があるGKVベンチ、A64FXもAmpere Altraも苦手なOpenFOAMの領域分割
  - 👍NUMA構成について考える必要がないため扱いやすい
    - A64FXやEPYCはソケット内の分割を考える必要があるため慣れていないユーザにはやや大変、実行しにくい・できない実行形態（プロセス数・スレッド数の組み合わせ）が生じやすい
  - 👎多スレッドの扱いが弱いかもしれない
    - 多スレッドのSTREAMやGKV kernel2（リダクション）の結果があまり良くない
    - 計算インテンシブでなければ全80コアを使わない選択も考慮する余地がありそう
  - プログラムの最適化についてはさらに調査・比較する余地がある
    - ベクトル化（SIMD化）の性能向上具合はどうだろうか？（SIMD偏重ではないのが逆に良い？）
    - 動作周波数が低くはないため、並列度があまり高くない部分が増えても耐えられそう？
    - CPUのせいかコンパイラのせいか、A64FXほど性能に敏感ではない？（最適化しないと全然性能が出ない、ということが少ないかもしれない？）
    - 最適化された高性能なライブラリ等が充実し高性能な実装のノウハウが共有できるだろうか