富岳からポスト富岳へ
ー HPC システムの進化と計算基礎科学

牧野淳一郎
神戸大学理学研究科 惑星学専攻/CPS

「富岳で加速する素粒子・原子核・宇宙・惑星」シンポジウム 2022/12/13

まとめ

- 「次世代計算基盤に係る調査研究」なるものが 7/26 に採択チームが公表、発足した。 これは基本的にはポスト富岳の概念設計である。「システム研究調査チーム」に理研 R-CCS 近藤さんが代表のチームと神戸大学 牧野が代表のチームの 2 つが採択された。
- 神戸大学のほうの内容は、PFN の深層学習向けプロセッサをベースに高性能な汎用 HPC 向けシステムを検討。理研は富士通、インテル、AMD、NVIDIA 等と検討。「オールジャパン」と書類に書いてある。
- というような話はともかく、ポスト富岳をどうするべきかは大変難しい。ここでは、まず富岳までのスパコンの歴史と半導体技術の関係、富岳までの日本の国策プロジェクトが何を達成したかを振り返り、現在見える問題点とその解決方向、さらにそこで「計算基礎科学」の果たすべき役割について述べる。

話の構成

- 「次世代計算基盤に係る調査研究」の紹介
- 富岳にいたるスパコンの発展の歴史と半導体技術
- 日本の国策プロジェクトの達成したもの
- ポスト富岳はどうあるべきなのか?

「次世代計算基盤に係る調査研究」の紹介

- 調査研究の目的等
- 採択されたチームの体制等

調査研究の目的等

公募要領から引用:
ポスト「富岳」時代の次世代計算基盤を、国として戦略的に整備することは、以下の観点から必要不可欠である。

1. 科学技術の各研究分野からの利用ニーズに応え、世界最高水準の性能を有し、それが無ければ実現し得ない世界をリードする卓越した研究成果を創出する。
2. 計算科学・計算機科学の技術と人材を維持・育成し、科学技術・学術情報基盤としてのみならず社会情報基盤としても重要な次世代計算基盤を我が国で開発・運用・活用できる能力を維持し、優位性、独自性を経済安全保障の観点からも確保する。
3. これらの結果として、新たな科学技術の創出、産業競争力の強化、Society 5.0の実現、国民の安心・安全の確保等の我が国が直面する社会的課題の解決に貢献する。

色々書いてあるけど要するになに?

3 項目の中で、これができていれば他はある意味自動的に達成できるもの:

> 次世代計算基盤を我が国で開発・運用・活用できる能力を維持し、優位性、独自性を確保する。

つまり

> **独自性があり世界と比べて優位性があるスパコンを開発する**

もちろん、ベンチマークだけ速いとかでは駄目でちゃんと産業界・アカデミア両方が使いたいと思うようなものでないといけない。(= 「科学技術・学術情報基盤としてのみならず社会情報基盤としても重要」)

公募要領もっと色々書いてない？

書いてある。
【果たすべき役割】６項目
【求められる性能・機能】４項目
【調査研究の実施に当たって、留意すべき事項】４項目
以下最後の項目の話を少しだけ

【調査研究の実施に当たって、留意すべき事項】

- 「富岳」の成果及び課題
- 評価指標として考慮すべき項目

  演算性能、電力性能比、I/O 性能、コスト、運用可能性、生産性（アプリ開発のしやすさ）、商用展開・技術展開、カーボンニュートラルへの対応等
- 技術的課題や制約要因の抽出

  想定されるものを列挙し、半導体等技術等の原理的限界によるものと、条件により技術的に突破できるものとを精査する。
- 日本が独自に保有すべき技術と国際協調する技術の特定

  国内の開発対象とすべき範囲を提示する

採択されたチームの体制等

- 全体構成
- 理研チーム
- 神戸大学チーム

(公表資料+公表用資料から)

「次世代計算基盤に係る調査研究」実施体制

文部科学省（「次世代計算基盤に係る調査研究」評価委員会）

- **PD会議**
- **運営委員会**
- アドバイザ・ユーザWG

令和４年７月時点

システム調査研究チーム（代表機関：理化学研究所）

アーキテクチャ調査研究	理化学研究所、富士通株式会社、日本AMD株式会社、インテル株式会社
システムソフトウェア・ライブラリ調査研究	理化学研究所、東北大学、筑波大学、大阪大学、九州大学
アプリケーション調査研究	北海道大学、横浜市立大学、物質・材料研究機構、海洋研究開発機構、東京大学、理化学研究所、筑波大学、東京工業大学

その他協力機関：株式会社データダイレクト・ネットワークス・ジャパン、国立情報学研究所、名古屋大学、NVIDIA Corporation、Hewlett Packard Enterprise、京都大学、国立天文台、日本原子力研究開発機構

システム調査研究チーム（代表機関：神戸大学）

アーキテクチャ調査研究	株式会社Preferred Networks、東京大学、国立情報学研究所、神戸大学
システムソフトウェア・ライブラリ調査研究	会津大学、松江高専、株式会社Preferred Networks、神戸大学
アプリケーション調査研究	順天堂大学、株式会社Preferred Networks、海洋研究開発機構、国立環境研究所、東洋大学、名古屋大学、広島大学、東京大学、神戸大学

新計算原理調査研究チーム（代表機関：慶應義塾大学）

慶應義塾大学、理化学研究所、九州大学、東北大学、日本電気株式会社

その他協力機関：富士通株式会社

運用技術調査研究チーム（代表機関：東京大学）

東京大学、理化学研究所、東京工業大学、国立情報学研究所

その他協力機関：名古屋大学、大阪大学、九州大学、産業技術総合研究所

調査研究の全体概要（理化学研究所：システム調査研究）

取組概要

次世代計算基盤には、SDGs・Society5.0の実現に向けた課題解決のためのプラットフォームとしての役割が求められる。そこで、今後の科学に「研究DX」をもたらす**高度なデジタルツイン実現の基盤**として、広範な計算手法・シミュレーション技法や大規模データを駆使しつつ、それらが密に連携しながら全体のワークフロー実行が可能な汎用性の高い計算基盤の実現を目指し、あるべきアーキテクチャやシステムソフトウェア・ライブラリ技術について、アプリケーションとのコデザインを通じた調査研究を行う。

特に、システム設計の基本理念として演算精度も考慮しながら必要な計算性能を確保し、**電力制約の下でデータ移動を高度化・効率化する"FLOPS to Byte"指向のシステム構築**を、アーキテクチャ開発からアルゴリズム設計、アプリケーション技術に至るまで実践する。

ALL Japan体制のもと、実効的な性能を向上させる次世代計算基盤のシステム構成や要素技術の調査検討、要素技術の開発を、アーキテクチャ・システムソフトウェアとアプリケーションとのコデザインを通じて実施する。

調査内容

アーキテクチャ調査研究

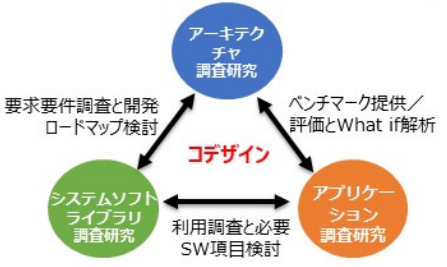
- 半導体技術・パッケージング技術動向を踏まえつつ、システム全体やその構成要素について考え得る技術的可能性や総合性能を調査
  （新規要素技術として三次元積層メモリ技術、強スケーリング・計算インテンシブ向けアクセラレータ、チップ間直接光通信技術などを意識）
- アプリケーション調査研究G提供のベンチマークセットの性能解析に基づき将来システムの性能を予測、また次世代アプリ開発へとフィードバック

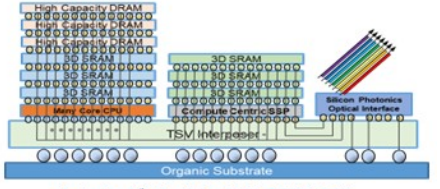
システムソフトウェア・ライブラリ調査研究

- 従来のソフトウェアに加え、データ利活用促進、機械学習技術と第一原理シミュレーションや大規模リアルタイムデータ処理との高度な融合、高セキュリティの担保などを主要検討項目と据え、国内で開発すべきソフトを優先度も含めて明らかにしつつ今後の開発ロードマップを策定

アプリケーション調査研究

- 複数アーキテクチャを統一的に評価するための広範なベンチマークセットを構築し、それを利用したアーキテクチャ評価結果を踏まえてアルゴリズムやパラメータの改善を検討しつつベンチマークセットを更新しながら、性能モデルも利用した探索的な"What if"の視点で解析を行う
- 次世代計算基盤に向けてどのようなアルゴリズムのクラスが大幅な進化が見込まれるか、の指標も抽出して今後のアプリ発展につなげる

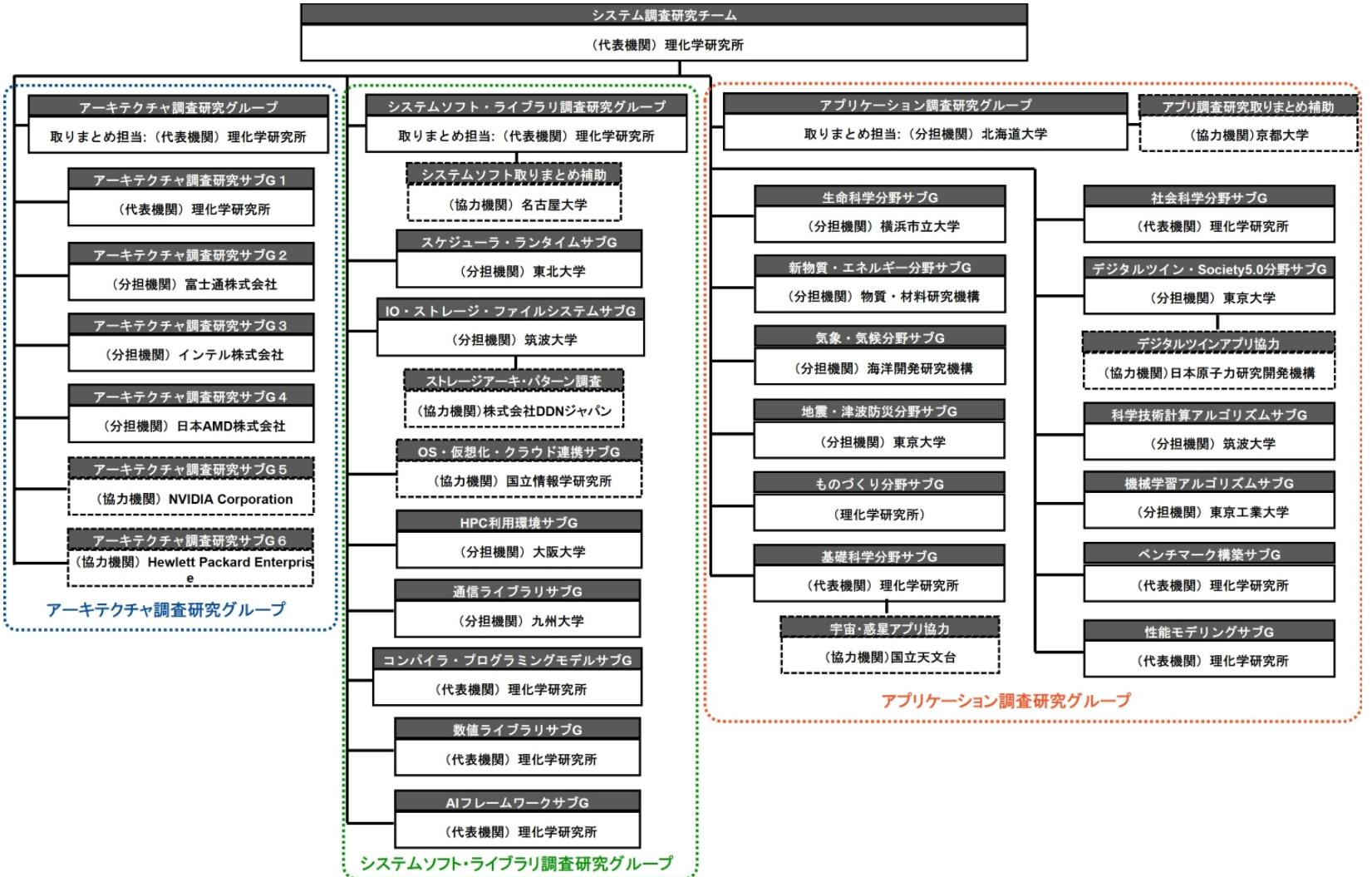




プロセッシングエレメントの検討要素技術例

スケジュール

	2022 Q3	2022 Q4	2023 Q1	2023 Q2	2023 Q3	2023 Q4	2024 Q1
アーキテクチャ	テクノロジ・アーキ技術の調査・検討			ベンチマーキングによる性能解析/予測		アーキ再検討	
システムソフトウェア	既存ツールや利用動向調査			新規開発ソフト項目検討・定量的評価		将来の要件整理	
アプリケーション	アプリ調査とベンチマーク設計			ベンチマーク評価に基づく性能分析		アルゴリズム最適化検討	


システム調査研究チーム
（代表機関）理化学研究所
アーキテクチャ調査研究グループ
取りまとめ担当:（代表機関）理化学研究所
アーキテクチャ調査研究サブG 1
（代表機関）理化学研究所
アーキテクチャ調査研究サブG 2
（分担機関）富士通株式会社
アーキテクチャ調査研究サブG 3
（分担機関）インテル株式会社
アーキテクチャ調査研究サブG 4
（分担機関）日本AMD株式会社
アーキテクチャ調査研究サブG 5
（協力機関）NVIDIA Corporation
アーキテクチャ調査研究サブG 6
（協力機関）Hewlett Packard Enterprise
アーキテクチャ調査研究グループ
システムソフト・ライブラリ調査研究グループ
取りまとめ担当:（代表機関）理化学研究所
システムソフト取りまとめ補助
（協力機関）名古屋大学
スケジューラ・ランタイムサブG
（分担機関）東北大学
IO・ストレージ・ファイルシステムサブG
（分担機関）筑波大学
ストレージアーキ・パターン調査
（協力機関）株式会社DDNジャパン
OS・仮想化・クラウド連携サブG
（協力機関）国立情報学研究所
HPC利用環境サブG
（分担機関）大阪大学
通信ライブラリサブG
（分担機関）九州大学
コンパイラ・プログラミングモデルサブG
（代表機関）理化学研究所
数値ライブラリサブG
（代表機関）理化学研究所
AIフレームワークサブG
（代表機関）理化学研究所
システムソフト・ライブラリ調査研究グループ
アプリケーション調査研究グループ
取りまとめ担当:（分担機関）北海道大学
アプリ調査研究取りまとめ補助
（協力機関）京都大学
生命科学分野サブG
（分担機関）横浜市立大学
新物質・エネルギー分野サブG
（分担機関）物質・材料研究機構
気象・気候分野サブG
（分担機関）海洋開発研究機構
地震・津波防災分野サブG
（分担機関）東京大学
ものづくり分野サブG
（理化学研究所）
基礎科学分野サブG
（代表機関）理化学研究所
宇宙・惑星アプリ協力
（協力機関）国立天文台
社会科学分野サブG
（代表機関）理化学研究所
デジタルツイン・Society5.0分野サブG
（分担機関）東京大学
デジタルツインアプリ協力
（協力機関）日本原子力研究開発機構
科学技術計算アルゴリズムサブG
（分担機関）筑波大学
機械学習アルゴリズムサブG
（分担機関）東京工業大学
ベンチマーク構築サブG
（代表機関）理化学研究所
性能モデリングサブG
（代表機関）理化学研究所
アプリケーション調査研究グループ

理研チームの要点

- 電力制約の下でデータ移動を高度化・効率化する”FLOPS to Byte” 指向のシステム構築
- 新規要素技術として三次元積層メモリ技術、強スケーリング・計算インテンシブ指向アクセラレータ、チップ間直接光通信技術
- アーキテクチャ調査研究に富士通、インテル、AMD、NVIDIA、HPE が参加。海外のプロセッサ・アクセラレータ企業を含む体制

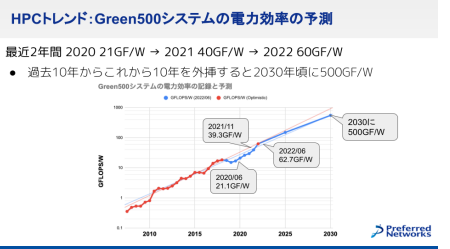
調査研究の全体概要（神戸大学：システム調査研究チーム）

取組概要

本調査研究ではPFN・神戸大学の持つ

- 世界最高性能のアクセラレータ実現技術
- AI応用フレームワーク/アプリケーションソフトウェア技術

を最大限に活用しつつ、従来から重要な HPC アプリケーションとAI利用等新しい応用の双方について高い実行効率を実現できるシステムの構成・評価をアプリケーショングループとの密接な協力によって実現する



調査内容

PFNと神戸大学が開発するアクセラレータを中心に複数のハードウェアアーキテクチャの組合せを、典型的かつ重要なアプリケーションタイプを対象に評価

アーキテクチャ調査研究

神戸大学・PFNが開発するMN-Core X とそれに適合したCPUによる省電力化・効率改善に重点

システムソフトウェア・ライブラリ調査研究

DSL・フレームワークによる高効率コードの自動生成に重点(深層学習分野のアプローチを他分野へも)

アプリケーション調査研究

これまでの伝統的HPCアプリの他、AI応用、商用ソフトウェアに重点

図表なし
（適宜

CPU・アクセラレータの最適な組み合わせを評価

スケジュール

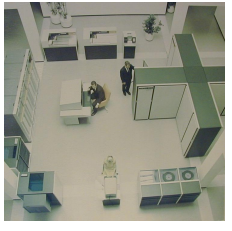
神戸大学チームの要点

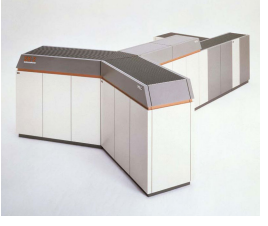
- PFN・神戸大学の持つ世界最高性能のアクセラレータ実現技術・AI 応用フレームワーク/アプリケーションソフトウェア技術を活用
- DSL・フレームワークによる高効率コードの自動生成に重点 (深層学習分野のアプローチを他分野へも)
- アーキテクチャ検討への企業としての参加は PFN のみ

富岳にいたるスパコンの発展の歴史と半導体技術

1940年代から現在までの80年間、ほぼ10年で100倍。(最後の10年を除いて、、、)
何故そのような指数関数的進歩を長期に続けたのか？
基本的な理由:

- 使うスイッチ素子が高速になった
- 使うスイッチ素子が小型、低消費電力になって、沢山使えるようになった
- 使うスイッチ素子が安くなって、沢山使えるようになった(スパコンの物理的大きさは70年代が最小。そこまで段々小さくなって、そこからまた大きくなった)





素子の性能向上、サイズ低下と
スパコンの性能向上

「スパコンの進歩」は５つの時期に分けられる

I スパコンが完全パイプライン化した乗算器をもたない時期 (1969 年まで)

II スパコンは１つ以上の演算器をもつが、１チップにはまだ演算器１つが入らない時代 (CMOS では 1989 年まで)

III1 チップに複数の演算器がはいるが、微細化すると動作クロックが上がり、電力が下がった時代 (2001 年頃まで)

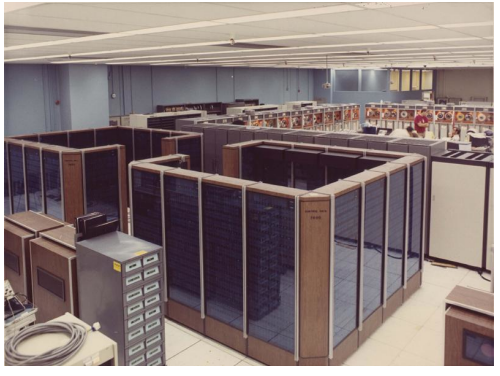
IV チップに多数の演算器がはいるが、微細化してもクロックが上がらず、電力もあまりへらない時代 (2001-2015)

V 微細化が限界に近付いた時代 (2015-)

このそれぞれで、素子の性能向上がどう使われた (る) かは違う。

I期: 1演算器未満の時代 ～ 1969

- この時期には、メモリアクセスのほうが演算より速い
- 演算器の性能向上が最重要課題
- CDC 6600 くらいまで。
- 計算機の構成方法もまだ手探り。
- CDC 7600 で 1 演算器はいるようになる



CDC7600 36.4MHz clock
1969
IC はまだ使ってない

II期: 1演算器以上の時代 ~ 1990

- 1つ以上の演算器を有効に使うことが課題
- パイプライン化(ベクトル命令)、複数の演算ユニットの SIMD and/or MIMD 並列実行が必要になる
- 演算器の数が増えると、メモリとどうつなぐかが課題になる。
- 演算器 16-32 個で破綻する (1992 年頃): メモリと演算器の間のデータ移動回路が大規模・複雑になり過ぎるため
- 末期の計算機: Cray T-90, 日立 S-3800
- 富士通 VPP500 では、分散メモリに: ある程度の成功
- 他の方向: 1チップマイクロプロセッサでの分散メモリ。メインメモリはキャッシュでバンド幅節約。プロセッサ間の通信は細い線でいいことにする。これが 90 年代以降の主流になった

III期: 1チップ高速化の時代 $\sim$ 2001

- 1チップマイクロプロセッサを分散メモリで多数つなぐのはII期末期から
- ありあまるトランジスタを演算器を増やすのにはつかわないで動作クロックの向上、キャッシュメモリの大型化に使った時代
- 1990年代。 牧野の意見としては「失われた10年」
- 計算機全体としてはII期に起こった問題がチップレベルでも起こるのを先送りにした
- メモリとの接続は速度が不足になった(memory wall): キャッシュでごまかす方針
- 迷走したプロセッサ開発も多い: 各社の複数チップ共有メモリプロセッサ。(失敗プロジェクト: ASCI Red 以外の ASCI マシン)
- 末期の計算機: Intel Pentium 4, DEC Alpha 21264

IV期: 1チップ多演算器化の時代 $\sim$ 2015

- 引き続き、1チップマイクロプロセッサ、分散メモリ。
- デザインルール 130 nm あたりから、動作電圧低下、速度向上に限界
- マルチコア化、コア内 SIMD 化を同時に進めてきた
- 80年代のベクトルスパコンの辿った道を追いかけたことになる
- ということは、16-32コアで破綻。
- 破綻がくることの予見: Intel Xeon Phi (60コア)
- A64fx, AMD EPYC は NUMA にすることで16コアの限界を回避 (言い換えると4プロセッサを1パッケージにいれてるだけ、、、)
- Top500 上位マシンのほとんどはアクセラレータに

V期: 「ポストムーア」 2015以降ずっと？

- コア内 SIMD の増強は限界に。 AMD、ARM は 256bit SIMD にとどまる。
- コア数増加は NUMA で
- CPU の電力性能はあまり上がらない。デスクトップ・サーバーともに最大電力は増える傾向。(Xeon MAX: 350W, AMD EPYC 9004: 360W)
- 最大クロックも微妙に上昇傾向
- x86 の終わりの始まり？低電力な命令セット？(ARM？ RISC/V?)
- GPU は深層学習というキラーアプリケーションを得て飛躍。CPU に比べて 3-5 倍程度の面積あたり性能と電力性能。さらに低精度行列乗算ユニット搭載で先行して高性能化 (NVIDIA V100)
- 深層学習向けプロセッサは 2016 年くらいには沢山プロジェクトやベンチャー企業があったが、、、V100 に蹴散らされた感が。

日本の国策プロジェクトの達成したもの

国家プロジェクトとしての計算機開発

- 通産省系
  - 科学技術用高速計算機システム (1981-89)
  - 第五世代 (1982-91)
  - RWCP (1992-2001)
- 文科省系
  - 数値風洞 (?-1993)
  - CP-PACS (1992-96、国家プロジェクトというほどではない？)
  - 地球シミュレータ (-2002)
  - 「京」 (2006-2012)
  - 富岳 (2014-2021)

数値風洞 (1993)

- II 期から III 期への移行の一方の象徴。CMOS 技術ではなく、まだ 1 チッププロセッサではないが、分散メモリに移行した
- 分散メモリに移行したため、従来のプログラムは基本的に全て走らなくなった。「VPP-Fortran」という特殊な方言で書くことが必須
- 普通はこんなの絶対メーカーが作らないが、、、
- ネットワークは完全クロスバーとした。高コストだが、共有メモリベクトルプロセッサのメモリインターフェースに比べるとたいしたことはなかった。
- 商業的にも成功だったといえる。Top10 の 3-4 台を占めた。同時期の共有メモリベクトルプロセッサの 10 倍のピーク性能を実現したことが大きい。
- 他の計算機の 10 倍速くなるならプログラムの書換えをいとわない人は結構いるということ

CP-PACS (1996)

- 1 チップマイクロプロセッサによる分散メモリ並列計算機
- III 期中期の標準的設計。革新的とはちょっといいがたい
- この時期の計算機としては高メモリバンド幅。1 チップの速度はそこそこ、という意味では若干時代に逆行。
- ネットワークは豪華。同軸ケーブルの山 (3 次元ハイパークロスバーというもの)
- 商業的にはそこそこ。

地球シミュレータ

- 1 チップベクトルプロセッサ、8 チップで共有メモリ、640 ノード構成。
- ノード数を抑えることで、数値風洞と同様のクロスバー構造を維持
- 共有メモリとしたことで、メモリ、メモリインターフェースは高コスト、大消費電力に
- 数値風洞からアーキテクチャが逆行した (部分的に複数プロセッサ共有メモリにもどった)
- つまり、設計としては IV 期になって現れた II 期の亡霊的存在
- 商業的には微妙

「京」

- 8 コアマイクロプロセッサ。2SIMD、2 スーパースカラー実行で 4 演算器
- IV 期中期のある意味標準的設計
- 細かくみると、8 コアで L2 キャッシュを共有するのは II 期末期的。標準的な分散 L2 から見ると時代に逆行
- メモリバンド幅が高いのもどちらかというと逆行的
- 多次元トーラスは時期的には微妙
- 商業的には成功とはいいがたい。「京」以外の最大のシステム: 東大基盤センターの 4800 ノード、60 億円。全部合わせても 100-150 億円？

富岳

- 48(12×4) コアマイクロプロセッサ。8SIMD、2 スーパースカラー実行で 16 演算器
- IV 期末期の標準的設計だが電力性能に大きくふって深いパイプライン、比較的弱いアウトオブオーダー資源、 L1, L2 バンド幅
- Top500 上位では BG/Q 以降唯一の CPU オンリーシステム。時代に逆行感は否定しがたい。
- 商業的には成功とはいいがたい。「富岳」以外の大規模システム: 東大、JAXA、名大、合わせても富岳の 1/10 程度?

革新と成功

国家プロジェクト(あるいはそれに近い巨大プロジェクト)でのスパコン開発を見ると

- 革新的でなければ商業的成功に結びつかない
- 革新的=他の計算機より圧倒的に高い性能
- 圧倒的な性能がちゃんと使えるものならプログラムとかなんとかする人々というのはいる

  (計算基礎科学の「貢献」=革新的なマシンの early adopter)

つまり、本来国プロは革新的=世界の誰もやってないことをするべき
といっても、ではどうすればいいのか？が問題。誰もやってないことをやればもちろん普通失敗する。

(じゃあどうすればよかったのか)

ある意味話は簡単: 商業的に成功しなかったのは目標が低かったから (価格性能比・電力性能比)→ 世界の水準以上に高い目標を設定すればよい

お金２つか３つに分けて競争させるとか (アメリカ・中国はそう)

まあいうのは簡単

ポスト富岳はどうあるべきなのか?

- 半導体技術の方向
- 富岳にみるポストムーア制約
- 本当の問題は?
- 現実的な汎用性
- 必要な方向

半導体技術の方向 — ポストムーアの技術的制約はなにか？

(今更ですが、、、)
ムーアの法則の半導体技術側のベース: CMOS スケーリング則 (デナード則):
トランジスタのサイズ (1 次元的) を半分にした時に

- 電圧を半分にする。寄生容量は半分になる。
- 速度は 2 倍、スイッチングの消費電力は 1/8、
- (トランジスタの値段は 1/4)

つまり: 並列性を完全に利用できるとすれば、同じ面積のチップで、トランジスタサイズが半分になると消費電力が同じで性能が 8 倍になる。
1985-2015 の間、概ね 4.5 年 (3 年はちょっと嘘) でロジック LSI のプロセスルールは半分になったので、「並列性を有効に使えて」「デナード則が成り立つなら」計算の性能は同じ消費電力で 10 年で 100 倍。

実際には

CMOS スケーリングは 3 つの意味で終焉

- **低電圧化の限界—2006 年くらい、大体 1V (今は 0.4V という話もあるが低速)**
- **寄生容量低減の限界— 2014 年、 FinFET への移行と同時。**
- **コスト低減の限界—これも 2014 年、 FinFET への移行と同時。**

あともうひとつ: 配線遅延・配線消費電力の問題
微細化しても、「単位長さ辺りのキャパシタンス」はかわらない、ところが、「単位長さあたりの抵抗」は配線幅の二乗に反比例して増える:

- **配線長がスケールしても配線遅延は「デザインルールに反比例して」増える**
- **配線長一定だと遅延は二乗で増える。電力は減らない**

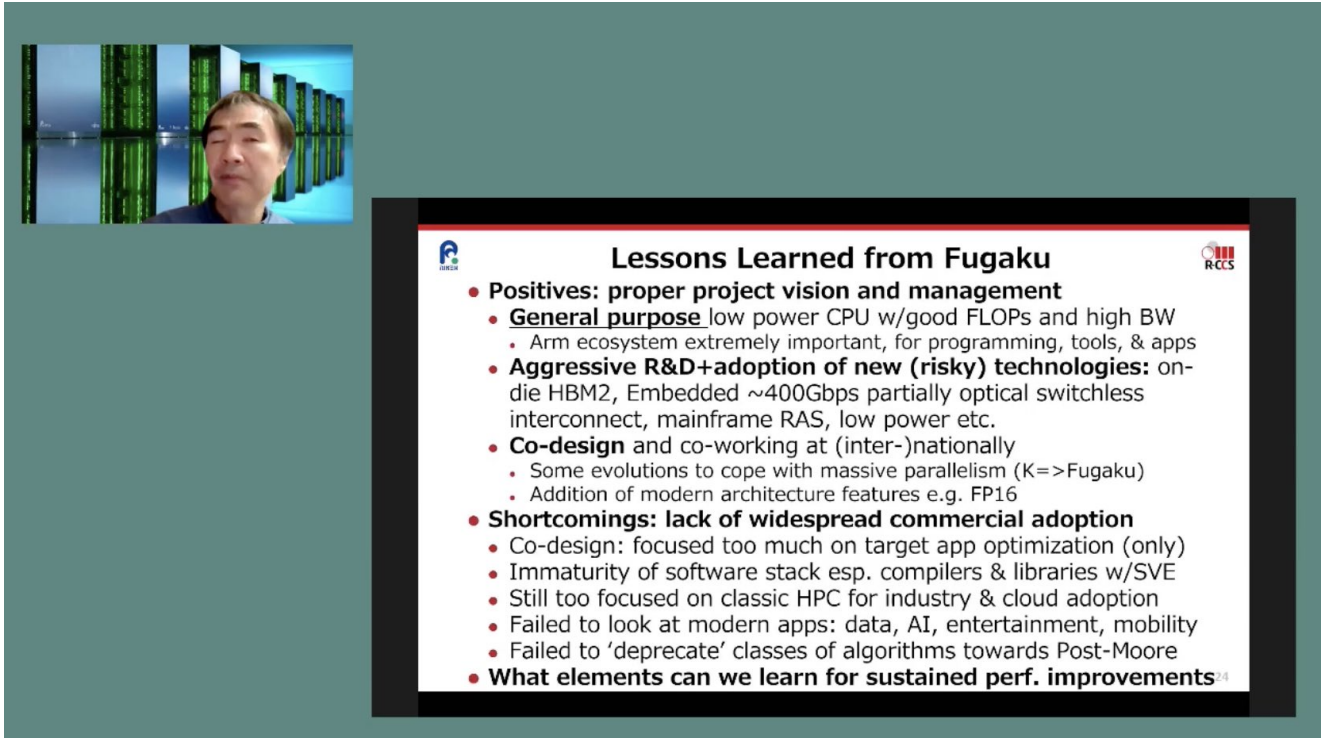
どういう問題が起こるか？

- プロセッサコア
  - 同じロジックでシュリンクしても性能があまり上がらない — SIMD 幅増やしてきた理由
  - SIMD 幅あまり増やすとコアの物理寸法が大きくなる – 遅延、消費電力が増える
  - 最近はトランジスタの価格が上がる – 高クロック化が必要になってきている
- チップ全体
  - 「ちょっと」長い配線 (共有 L2$ とか LLC までとか) の遅延が微細化が進むと大きくなり、電力が下がらない = LLC の遅延が巨大になり、バンド幅をあげられない –チップレット化の理由の１つ

「富岳」にみるポストムーア制約

- プロセッサコア
  - SIMD 幅は 512bit 2way。この辺が限界。
  - 演算、L1 レイテンシは 9 サイクルと大きい (x86 は 4 程度)
  - おそらく電力制約から OoO 資源は最小限 (x86 の半分程度)
- チップ全体
  - 共有 L2 アーキテクチャのため、L2 までの配線が長い (4 コアグループなのでチップサイズの半分にはなる): 非常に大きな遅延 (50 サイクルくらい)、
  - バンド幅も大きいとはいいがたい。コア当り 43/21 bytes/cycle。 B/F として 1 程度。メモリは 0.36。(load/store)

これを使うほうから見ると、、、



SIAMPP22

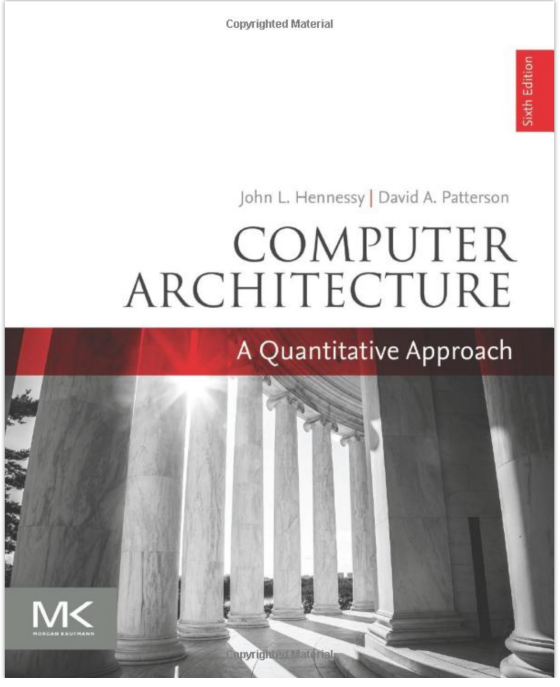
「昔のアプリケーションに最適化されすぎてモダンなアプリケーションに対応できてない」

とはいえ要するに:「実用アプリケーションで性能でてないじゃないか」

ポスト富岳の方向性

- (例えば) A64fx のコアの延長はありそうにないようにみえる
- では代替は？ GPU なのか？
- そもそも「これが駄目だからあっち」という考え方でいいのか？

本当の問題はなにか？



この定量的アプローチは絶対的に正しい」

右側の本の序文から

「ヘネパタは、コンピュータの設計の指針として「定量的アプローチ」を掲げている。これによるとコンピュータの設計は、実用的なプログラムを実行した評価に基づき、コスト、性能、電力を定量的に測定することを通して行わなければならない。

この主張自体はもちろん間違ってはいない

が、実際の HPC 応用を考えると非常に大きな問題がある。(HPC 応用以外も同じ)

現存の「実用プログラム」は現存の (というのは嘘で 10-20 年前に存在した) プロセッサに最適化されている (まあ全然最適化されてないこともあるが)

↓

設計で可能な探索空間 (「最適解」が見つかるところを含めて) が、現存 (あるいは 10-20 年前の) プロセッサの近傍になる。

つまり: 本当の最適解 (がどんなものかはさておき) 全然明後日の方向にプログラムとアーキテクチャが共進化してきた「かもしれない」

= 「本当の問題」

つまり問題は

- 実は我々は「このプロセッサはどれくらい良いか」を定量的に判断する規準をもっていない。
- 「実用的なプログラム」での「コスト、性能、電力」は局所的。大域的に有効ではない。
- 実際、商業的に大成功であった数値風洞では既存のプログラムは (1 ノード実行以外) 一切動かなかった。

つまり

- アーキテクチャはアプリケーションの書換え、場合によってはアルゴリズムの変更もいれないと意味がある評価にならない
- アプリケーションの書換えが現実的にできる仕掛けは用意する必要がある
- その前提で「コスト、性能、電力」を評価するべき。エネルギー最小とか。

HPC における「エネルギー最小」の定義の問題

- エンジン: 「効率 100%」が定義可能。入力熱エネルギーから第二法則の限界まで機械的エネルギーに。飛行機も同様: 圧力抗力、誘導抗力ゼロ。
- 計算機: 半導体技術、動作電圧、ターゲット周波数とかで同じ論理回路でも電力は全く変わる。そもそも、アプリケーションだって色々ある。基準にできるものはあるか？
- でも実はもう半導体技術はこの先大して進歩しないので、半導体技術は given でいい。それなら、「演算に必要な電力と総電力の割合」でどうか?
- そうすると、結局、効率を、「全消費電力と、その半導体技術で構成した理想的な演算回路の組合せ論理部分の消費電力の比」と定義できたことになる。(これは実は半導体技術に依存しない定義になっている)

アプリケーションの「多様性」

アプリケーションにはどれくらい色々あるか？日、米、欧のエクサスケールプロジェクトでの対象アプリケーションを見ると:

分類	数
構造格子	14
非構造格子	13
粒子	2
ランダムグラフ	2
密行列	3
その他	3

「ランダムグラフ」は神経回路シミュレータ。NEST とかは実装は構造格子に近い。
「その他」の 1 つはゲノミクス。　日本の GENOMON(中身は BWA-MEM、、、)
今はもちろん深層学習が重要。これは結局 (低精度) 密行列。

つまり

- 構造格子・非構造格子が多い
- 残りの多くが深層学習と量子化学・物性計算 (密行列)
- あと粒子法とゲノム (動的計画法)
- これくらいできれば実際上「汎用スパコン」
- 深層学習は専用計算機でいいかも (AI プロセッサ、、、) あるいは AI プロセッサがおまけで他のこともできるとか。(「現実的な HPC のエコシステム」)

現実的汎用性からみた過去と現在のプロセッサ

- 現実のプロセッサについて、「アプリケーションでの演算部論理の消費電力の割合」をだすのはそんなに簡単ではない。
- 雑だが相関はありそうな指標: コアロジックの全トランジスタ数に対する、演算ロジックに使われていそうなトランジスタ数の割合。
- 過大評価: 演算器以外はフルに動いてるとは限らない。
- 過小評価: 効率 100%で演算器が動いてるわけではない
- ちょっと別の話: 特に 7nm 以降、「トランジスタの価格は上昇」
- それなりに意味があるかも？

データ

	Single Core	MIMD	In core SIMD	Global SIMD	Coherent Cache	Non-Coherent Cache	Local memory	Transistor efficiency
CM-2	-	-	-	✓	-	-	✓	30%
Illiac IV	-	-	-	✓	-	-	✓	7%
Cray-1	✓	-	-	-	-	-	-	6.5%
Intel i860	✓	-	-	-	-	✓	-	6.5%
PEZY-SC	-	✓	-	-	-	✓	✓	2.4%
NVIDIA A100	-	✓	-	✓	-	✓	✓	1.7%
Fujitsu A64fx	-	✓	✓	-	✓	-	-	1.3%
Intel Skylake	-	✓	✓	-	✓	-	-	0.8%
Intel P4	✓	-	✓	-	-	✓	-	0.3%
Intel Core2	-	✓	✓	-	✓	-	-	0.08%
SW26010	-	✓	✓	-	-	-	✓	-

表見ると:

- MIMD+In core SIMD+コヒーレントキャッシュでは 1% を超えるのは難しそう。
- A64fx は 1% 超えていて数字は確かに良い。
- A100 と PEZY-SC は 2%前後で、さらに良い。
- 昔のベクトルプロセッサや SIMD 超並列マシンは非常によい。

要するに: マルチコア+コヒーレント階層キャッシュで高い効率のプロセッサを設計するのは「無理」
ある意味当たり前: キャッシュが大量のトランジスタ (と電力) を制御しにくいやり方で消費する。

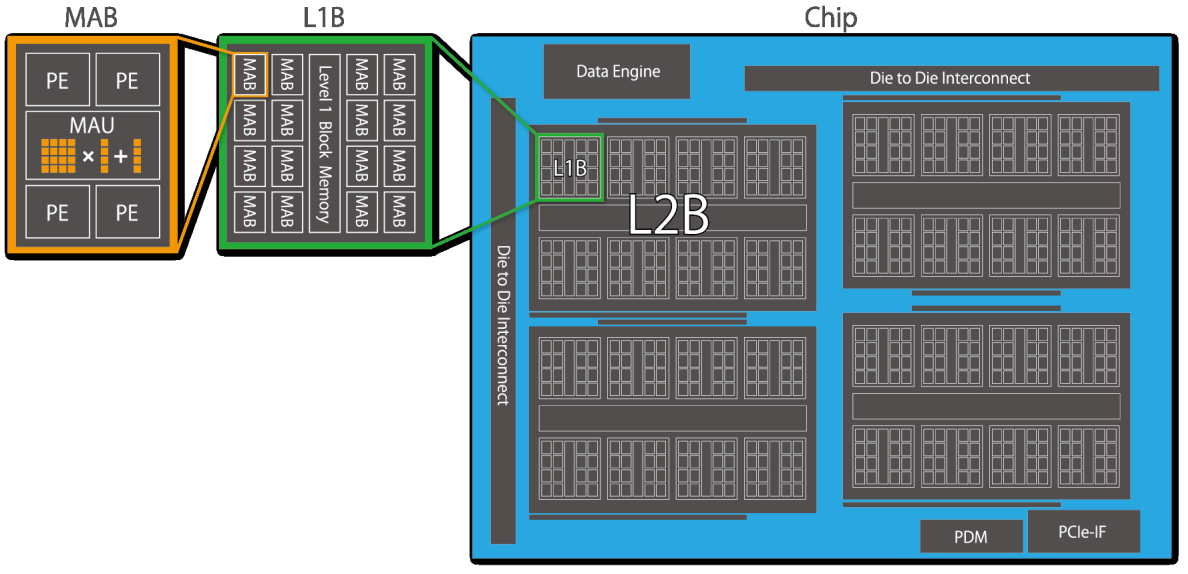
アプリケーション実行効率は、、、SC22 ゴードンベル賞の AMR PIC だと A100 で 1.2TF/device。19.5TF がピークなので、、、

必要な方向

ここは神戸大学 FS の方向。全然違う方向もありえる。

- near-memory computing. DRAM と演算器を (物理的に) 近くに、配線も短く。(3D 実装。TSV は高価なのでできれば 2 枚貼り合わせ)
- 階層キャッシュではなくて演算器に近いメモリを一番大容量にしてデータ移動を減らす。DRAM もそこからつなげる
- キャッシュではなく on-chip SRAM と DRAM の間のデータ移動を明示的に制御するアーキテクチャ
- MN-Core は既に大体そういう方向
- そんなののプログラミングはやってられないのでそこをなんとかする DSL やコンパイラ

MN-Core アーキテクチャ



PE のローカルメモリが一番大きい。L1B, L2B にもメモリはあるがそれらは転送のバッファリングのための最小限。

ソフトウェアの方向

- 深層学習なら TensorFlow, PyTorch, JAX とか (MN-Core で実用)。このレベルだとわりと本当にアーキテクチャに依存しない記述が可能 (OpenACC の pragma みたいなのはマシン依存、、、)
- あとはアプリケーションのタイプ毎に。構造格子なら JAX で、とか JAX を生成するものとか
- 例えば気象コードならそれ+OpenACC 的な、格子点での物理過程を Fortran90 とか C++とかで書けるものは必要
- 粒子法でプロセッサ間の粒子移動とかを効率的に実装するには低レベルの OpenCL 的なものが必要そう。
- QCD はこれから御相談。

まとめ

- 「次世代計算基盤に係る調査研究」なるものが 7/26 に採択チームが公表、発足した。 これは基本的にはポスト富岳の概念設計である。「システム研究調査チーム」に理研 R-CCS 近藤さんが代表のチームと神戸大学 牧野が代表のチームの 2 つが採択された。
- 神戸大学のほうの内容は、PFN の深層学習向けプロセッサをベースに高性能な汎用 HPC 向けシステムを検討。理研は富士通、インテル、AMD、NVIDIA 等と検討。「オールジャパン」と書類に書いてある。
- というような話はともかく、ポスト富岳をどうするべきかは大変難しい。ここでは、まず富岳までのスパコンの歴史と半導体技術の関係、富岳までの日本の国策プロジェクトが何を達成したかを振り返り、現在見える問題点とその解決方向、さらにそこで「計算基礎科学」の果たすべき役割について述べる。