シミュレーション天文学、専用計算機、汎用スパコン、深層学習

牧野淳一郎

研究会「シミュレーション天文学のこれまでとこれから-ハードウェア・アプリケーション・サイエンス-」 2023/9/4

という感じで歴史を振り返る的な話をしようかと先週くらいは思っていたのですが

もうちょっと、まとまりのある話にしたい気がしてきたのでそっちの話を多めに。

話の構成

- 牧野の研究史っぽいもの
  - GRAPE 以前
  - GRAPE
  - GRAPE 以後
- スーパーコンピュータアーキテクチャの歴史学と科学 (あるいは観測と理論)

牧野の研究史っぽいもの

- GRAPE 以前
- GRAPE
- GRAPE 以後

GRAPE 以前

- 1981/4 東大理 I 入学

  どの学科にいくかは入学してから色々考えるつもりで、、、

  2 年の夏休み (進学振り分けの最終希望をだすのは夏休みの後の試験の後だったはず) に、親に教養学科第一 (科学史・科学哲学) か基礎科学科第二 (「システム基礎科学」と書いてある。新学科) にいくといったら、母親の短大の時の友人 (家族ぐるみの付き合いが続いてた) の旦那が京都の物理の先生だから一度話をきいてこい、というのでいってきた。京都でバスの向き間違えて遅刻したはず。

  これが牧二郎先生。で、東大のことはよくわからないから駒場の河原林先生の話を聞け、ということになって、そのころは基礎科学科第二を考えてて、そういう話をしたらあそこは杉本さんがいるからいいんじゃない？となったような。

  これが多分杉本さんの名前を意識した最初。教養では地学とってなかった (図学とった) ので杉本さんの講義きいてない。

GRAPE 以前(2)

- 1983/4 東大教養学部基礎科学科第二進学

  環境問題の解決に貢献したいとか高尚なことを考えてたような気もする

  卒研のテーマにしたのは「地球系の温度変動モデル」。平沢冷 (さんずい) さんのだしたテーマ。平沢さんは「システム科学」やるというので学科設立に貢献して教育にも貢献した人だけど元々のバックグラウンドは電気化学で、ESR でなんか (私よくわかってない) が専門だったはず。東大総長やった向坊さんの弟子。この後は科学技術政策研究のほうに重点。温室効果研究はもちろん地球環境問題研究だけど、原子力推進な面も。

  というわけで平沢さん自身学生になにをやらせるのかわかってなかった感じがあり、色々教科書とか論文 (Manabe and Stouffer 1980 あたりとか) 読んで、3 次元計算は自分では無理なので 1 次元鉛直モデル (Manabe and Wetherald 1967) の再現的なことを。CO2 2 倍にしたら平均気温これくらい上がるとか一応でた。

GRAPE 以前(3)

でたけどこれ自分でつづけるのはしんどいなというのと、まあだからこれもう答わかってるじゃんというのがあって、もうちょっと単純な物理だけど非線型で計算機シミュレーションで色々なわかる、というのをやろうと。
大学院入試 (これが「大学院が正式にできてから」ということで 4 月にあったはず) の前に杉本さんの話は聞きにいったんだけど、面接の時に「君、僕のとこにはこないと思ってたよ」といわれた。

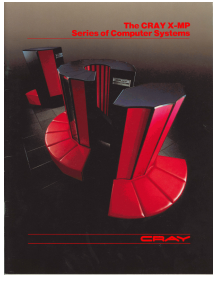
全く余談:
ちなみに温暖化研究自体は基礎科第二では阿部 (寛治) 研にいった江守さんが住さんとかと共同研究の形で発展させたような。その流れで阿部研で流体やってたのが気候研究にもらわれていったのが NICAM を作った富田浩文さんで、阿部研でその 2 年後輩が吉田直紀さん。

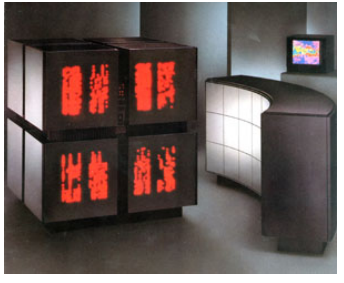
GRAPE 以前(4)

- というわけで、大学院では N 体で重力熱力学振動をだそうという今考えてみると無謀なテーマを。
- その時 D3 だった種艸さんと Aarseth の NBODY5 を読む。最初は三鷹の M-360R でちょこちょこ計算: 100 体はできた。 250 体とかやったかな？M1 の秋に野辺山に 1 週間いって、 M-380 で 400 体やった。 1 月に NBODY5 のベクトル化をやって、S-810 で 1000 体。
- 3月 (?) に、稲垣さんとこと須藤さんとこにきた Aarseth と議論、東大センターで NBODY2 ベクトル化を一緒にやってみた。
- 杉本さんのところに、83 年に IAS の教授になった Piet Hut から、宇宙物理でスパコンどう使ってくかの研究会するからお前もこいというのがきて、これおまえいってこいと、、、

GRAPE 以前(5)

- 多分誰にもわからないような××な英語で話をした(OHP は手書きじゃなくて、杉本ワープロを私が Pascal で書き直した上で XY プロッタで OHP に書けるようにしたので作ったので読めたはず) が、Piet とこの話を理論的につめて論文にしようということに。ApJ にだしたらこんなのは封筒の裏でする計算で論文じゃないというレフェリーコメントがきたけどなんとか通った。
- この辺で Barnes-Hut treecode とかもでてきて、D1, D2 辺りはそれのベクトル化とか CM-2 への実装とか。ボストンの冬は寒かったです。あとリクルートの Cray X-MP で球状星団の 3000 体の計算をやったり。





修士の学位授与式の写真があったので。

前列左端が牧野
中列右端が杉本さん
前列中央は藤垣さん
(現在東大副学長)

GRAPE

- 近田提案
- GRAPE-1
- GRAPE-2
- GRAPE-3
- GRAPE-4
- GRAPE-6

近田提案

1988年、天文・天体物理夏の学校

88/11/2

K. K.

「計算機を使う，計算機の技術を使う」

近田義広（国立天文台・野辺山）

観山さんは理論の立場から見ると，現在の日本ではスーパコンがどんどん使えるからイニシアチブをとる良い機会だと話されました．私は，その素晴らしい計算機を作り出す技術をうまく使いこなし，天文に応用すれば，計算機の利用者に止まる場合よりもっと大きな成果を期待できるぞ！ということをお話しします．

- 計算機買うんでなくて作れ!

近田提案(2)

〔昨日，今日，明日〕

先ず，Ｍ１，Ｍ２の若い人達に話す時の定石の昔話から始めます．約２０年前の１９６０年代の末，三鷹の天文台に口径６ｍのミリ波望遠鏡が作られ，後の野辺山の望遠鏡の雛形が生まれた時，その制御とデータ収集はOKITAC4300と言うミニコンが受け持ちました．そして今，６ｍ鏡は水沢に運ばれてＶＬＢＩに使うために化粧直し中．今度の制御用計算機は日電のパソコンPC9801VMです．表１に見るように，この２０年間に記憶容量も演算速度も約４桁良くなっています．つまり，同じ予算でも１０年たてば１００倍の仕事をさせられるようになったと言うことです．これは天文台と言うローカルな世界だけのことではありません．事実，図１〔１〕の並列計算機の並列度の推移にみるように１０年で１００倍というのはかなりよいestimateです．但し，これは重要なことなのですが，同一の構造，同一の使われ方，つまり同一の製品系列の中での進展はこんなに早くありません．例えば野辺山の汎用計算機についてこの４年間の価格／性能比を見るとせいぜい２倍しか良くなっていません．つまり計算機（の技術）の使い方を常に見直して，自分の問題に適した構造の機械を使う（作る）ようにし，かつその上で解き易いように問題を立て直すことがないと「１０年で１００倍」を生かした人に比べ１０倍分は立ち遅れることになります．

計算機は10年で100倍速くなる。 同じ製品系列だと10倍しか速くならない。

近田提案(3)

全体の値段　∝　性能　∝　装置の大きさ

でもあります．表２に見るように上記３者の比例関係は大体成り立っているように見えます．しかし本当に価格／性能比は一定なのだろうか？　もしそうなら利用者は利用者に徹し，計算機システムの構成に気を使ったり，自分で情報処理機を作ったりしようとなどせずに，与えられたお金でＣＰＵ時間を買っていれば良いことになります．誰もが平等にお金の額に相当する「性能」を手にするのだから，残された仕事は金を取ってくることだけです．

そうではありません．汎用の大きな計算機は，出来るだけ広い範囲の利用者の平均的な要求を同時に満足させる「性能」を考えて作ってあります．ですから汎用「性能」を考えた場合は上の比例式は成り立っていますが，特定の範疇の性能を取り出すと比例関係は成り立ちません．だから，問題は自分の課題に対する「性能」とは何か？それを一番安く（容易に）手に入れるにはどうしたらいいか？です．例えば野辺山クラスの計算機１台とＥＷＳ(engineering work station)４００台とどちらがいいか？　もし演算速度だけを問題にするなら，ＥＷＳを４００台の方が良いでしょう．またＦＦＴの速さだけなら野辺山のＦＸはスーパコンピュータに比べて２桁以上速くて（表３）値段は１桁安いのだから，その方がいいに決まっています．汎用か専用かでこんなに性能／価格にひらきがでるのは一体どうしてなのでしょうか．

専用化すれば２ー３桁価格性能比をあげられる

近田提案(4)

具体的イメージをつかんでもらうために，例として図２にパソコンにつないでN体問題の重力を計算する回路の概念を示します．１０ＭＨｚクロックで間口がパラレルのパイプライン式で３２ｂｉｔ固定小数点で計算させるとすると，市販のＩＣで２００個　0．１立方米位の嵩で，２５０ＭＯＰＳ位のスーパコンピュータ並みの性能が得られるでしょう．大体，１万体問題を１万ステップ１日に計算できる勘定です．値段としては自作で４００万円位でしょう．もし億円オーダーの金をかけて，大型計算機につなぐようなパンとしたものを作る場合は，高性能をねらって専用ＩＣを作り，ＩＣ５０００個　４ｂｙｔｅ浮動小数点演算，２０ＭＨｚクロック，２０立方米，１００ＧＦＬＯＰＳ，１０万体問題を４万ステップ１日に計算できる．（但し，近田電子製作所の見積もりは甘いと言う声もあることを付け加えます）．

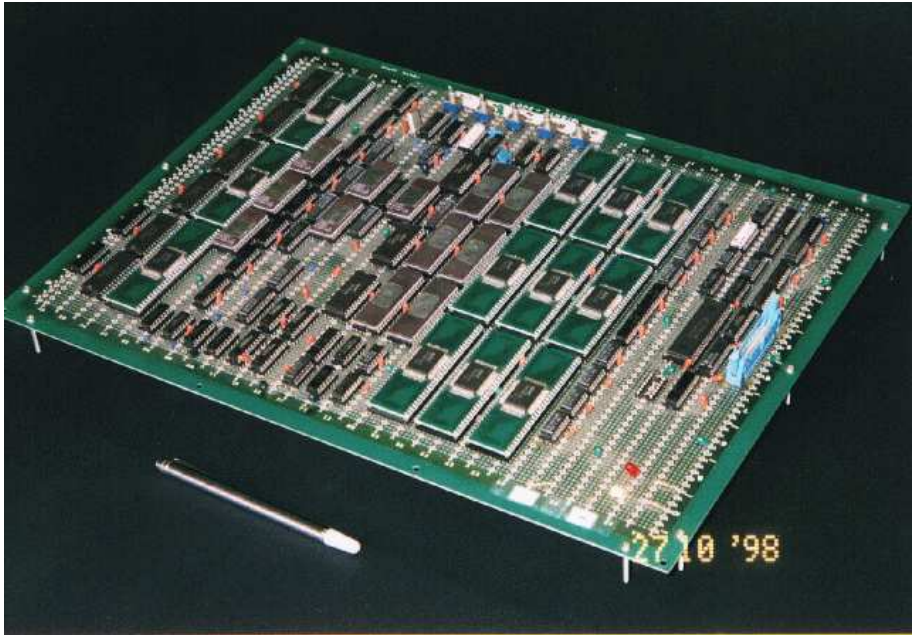
- 多体問題なら 400 万円 250 Mflops
- 数億円なら 100 Gflops
- 「但し、近田電子製作所の見積もりは甘いという声もあることを付け加えます」

杉本さんの反応

- 10月に近田さんの話を聞きに野辺山に
- 川合先生(慶応)、牧野、伊藤君が同行
- 近田さん、奥村さん他で話を。

「近田さんが色々教えてくれそうだから、やってみよう」

GRAPE-1(1989)



GRAPE-1 開発プロセス

- 88 年 12 月くらいまでは、伊藤がデジタル回路の勉強、平方根回路の検討とか
- 89 年 2 月前後に、回路規模検討、ROM テーブル、対数フォーマット等検討
- 5 月に川合先生のところで (というか朴さんに) ご指導いただく。特に大きな問題なく、伊藤君すごいな的。
- 8 月終わりにはハードウェアができていた

開発はどんなふうだったか

詳しくは伊藤君の本をどうぞ。

ハードウェアは私が知らないうちにできていた(伊藤君が作った)ので良く知らない。

GRAPE-1 と近田原提案の違い

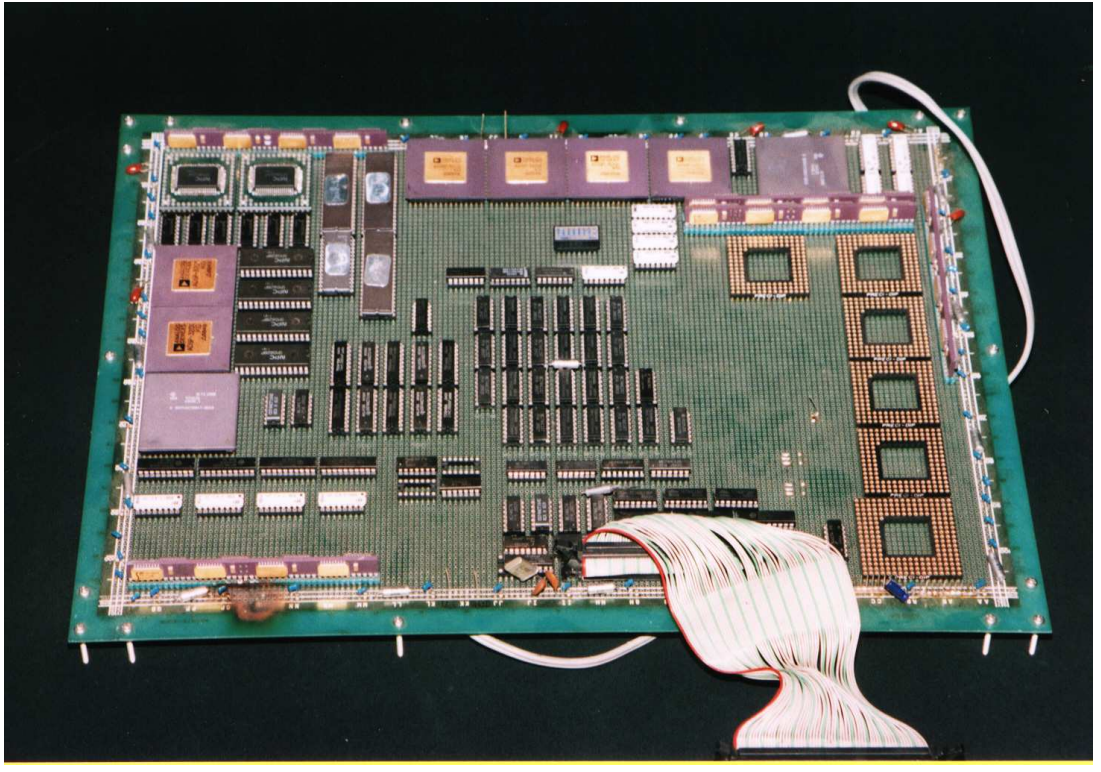
- パイプライン構成は基本的に同じ
- 大きな違い: 数値表現の形式
  - 近田さんの想定(多分): 固定小数点32ビットで基本的にはそのままいく。2乗すると64ビットになって、 -1.5 乗の結果は96ビットになるけど、、、
  - 当時の素直な解決方法: 85年くらいからでてきていた、浮動小数点計算専用 LSI を使う。
  - GRAPE-1: 途中を、8ビット対数表現にする。

ハードウェアコストを近田電子製作所見積もりより1桁下げた!

何故対数表現？

- 8 ビットに収めたかった: 512kbits の ROM で 2 項演算が実現できる。($x^2 + y^2$ とかが石 1 つですむ)
- 最初と最後だけ固定小数点形式にすれば、途中は全部対数形式でも問題によっては十分な精度がでる。
- 牧野は学部 3 年生の時に学科の特別講義で石黒さん (多分) が野辺山 FX の話をして、数ビットで計算しても大丈夫というようなことをいったのをおぼえていたような気がする。
- 無衝突系の場合、2 粒子間の力の必要な精度は低い。多数の粒子があることで統計的に高い精度が出れば十分 (ということを理論的に証明できた)。

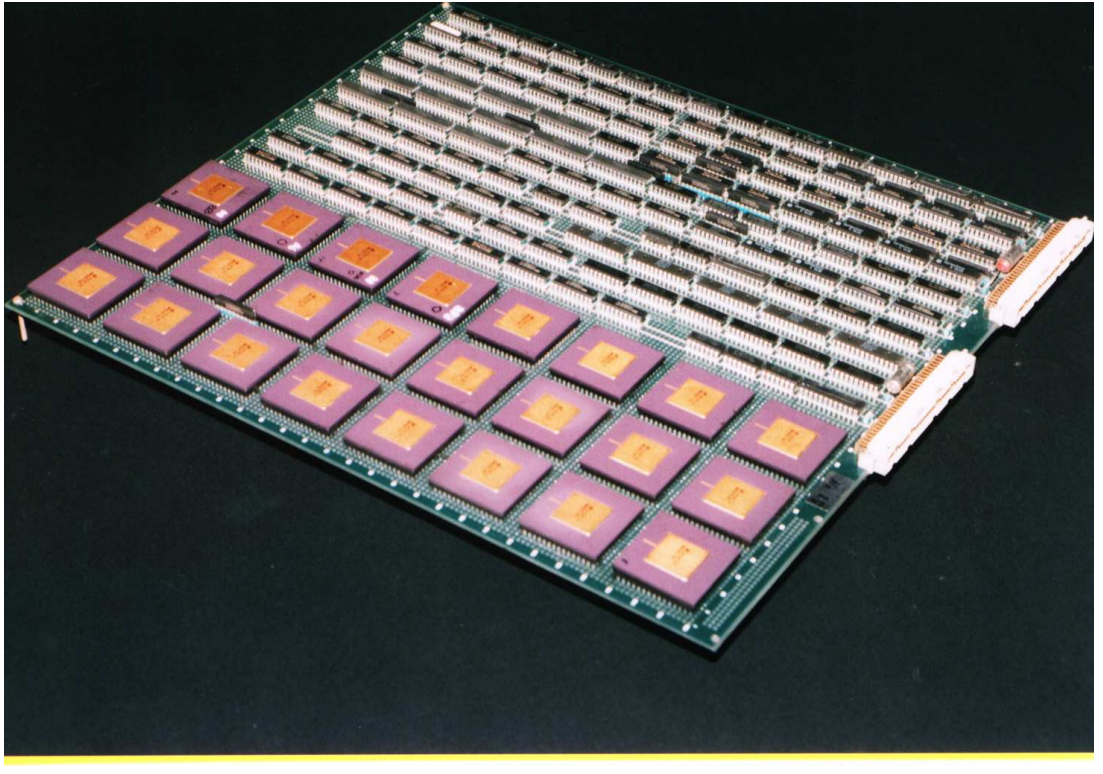
GRAPE-2(1990)



GRAPE-2 の中身

- 8ビット演算とかは止めて普通に浮動小数点演算 (倍精度は最初と最後だけ)
- 高精度が必要な問題も色々あるため
- 40Mflops
- 戎崎、伊藤、牧野
- 詳しくは伊藤君の本を参照

GRAPE-3(1991)



GRAPE-3

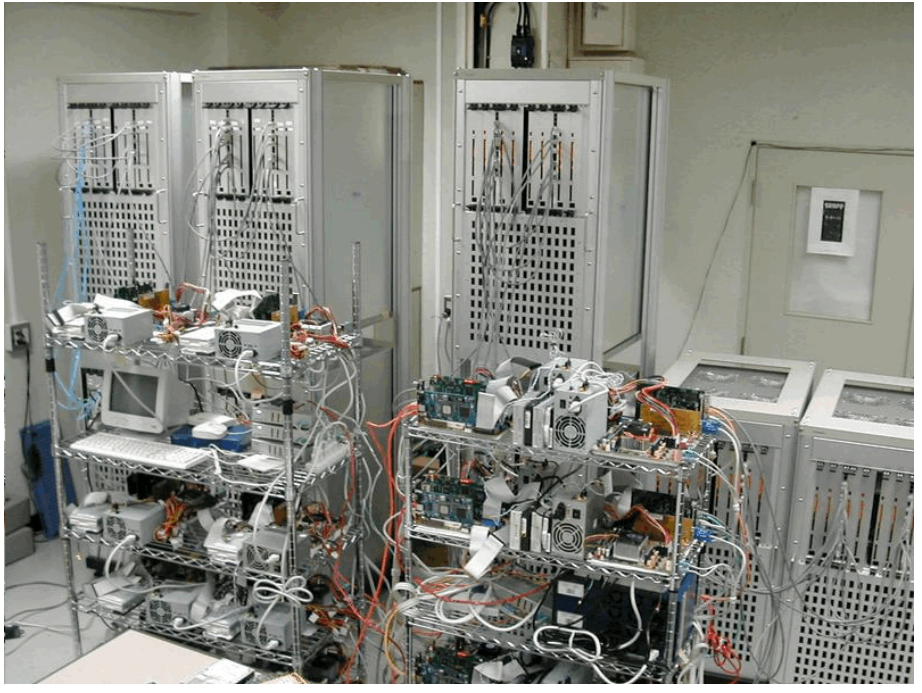
- 大雑把には GRAPE-1 パイプライン(但し、大きな ROM は使えないので対数での加算の実装は全く違う)
- 仕様決定、シミュレータ (C で記述) は牧野
- 論理設計以降は富士ゼロックス (橋本、富田)
- SCS Genesil で設計
- ファブは NS. 1μm
- ボードは奥村
- 1 千万ちょっとの予算で 20Gflops を実現。

GRAPE-4(1995)

GRAPE-4 概要

- 48 個プロセッサチップがのったボード 36 枚
- 1 チップ 20 演算、32 MHz 動作で 640 Mflops
- LSI logic $1\mu\mathrm{m}$ スタンダードセル
- プロセッサチップは泰地、「予測子」チップ牧野

GRAPE-6 system



2002年の 64 Tflops
システム

4 ブロック
16 ホスト
64 プロセッサボード

他の GRAPE 型機械

- 1991 GRAPE-1A : GRAPE-1 の細かい改良。設計: 福重
- 1992 GRAPE-2A : 最初の MD 用 GRAPE。設計: 伊藤・福重
- 1992 HARP-1: 小久保君設計。 Hermite 公式用
- 1993 GRAPE-3A: 商業版
- 1996 MD-GRAPE: 最初の MD 用 GRAPE チップ チップ設計: 泰地
- 2001 MDM (MDG2): 理研・戎崎グループによる。 75T
- 2006 PE (MDG3): 理研・泰地グループによる。 1P
- 1992- MD-Engine: 富士ゼロックス、大正製薬
- 2010- MDG4, 4a: 理研・泰地グループによる。

計算法とか

「GRAPE 向け」(ベクトル機向けとかも含めて) 計算法色々

- ツリーの並列化: 87-88 に Barnes、Hernquist とかと。
- 独立時間刻みの並列化: Steve McMillan の Cyber 用のが元。
- Hermite 法: Piet との雑談から発生した気が。
- 疑似粒子多重極法
- Bridege (Fujii+2007): マルチスケール・マルチフィジクスなシステムのハミルトニアン分割による一般的な取り扱い。
- Particle-Particle Partcile-Tree 法 (Oshino+ 2011): Tree と独立時間刻みの融合。まあ Mercury の遠距離部分に tree 使ったともいえる。

- FDPS (Iwasawa+ 2016): 2 粒子間相互作用の多体計算ならなんでも並列化できる (自画自賛ですが) 素晴らしいフレームワーク。GPLUM、PETAR はこれ使って書かれている。
- SPH 関連色々

GRAPE 以降

- 開発費の問題
- 解決の方向
- 現状

開発費の問題

- 半導体の進歩に比例して開発費用も増大

年	計算機	チップ初期コスト	設計ルール
1992	GRAPE-4	200K$	1μm
1997	GRAPE-6	1M$	250nm
2004	GRAPE-DR	4M$	90nm
2016	MN-Core	$>$ 10M$?	N12

全体予算は初期コストの４倍程度必要。
15億の予算は天文専用ではとれない。

GRAPE-DR

- プログラム可能アーキテクチャ — GRAPE より広い応用範囲
- SIMD 並列計算機を 1 チップに。
- チップ性能: 単精度 512, 倍精度 256Gflops (500MHz で、、、)

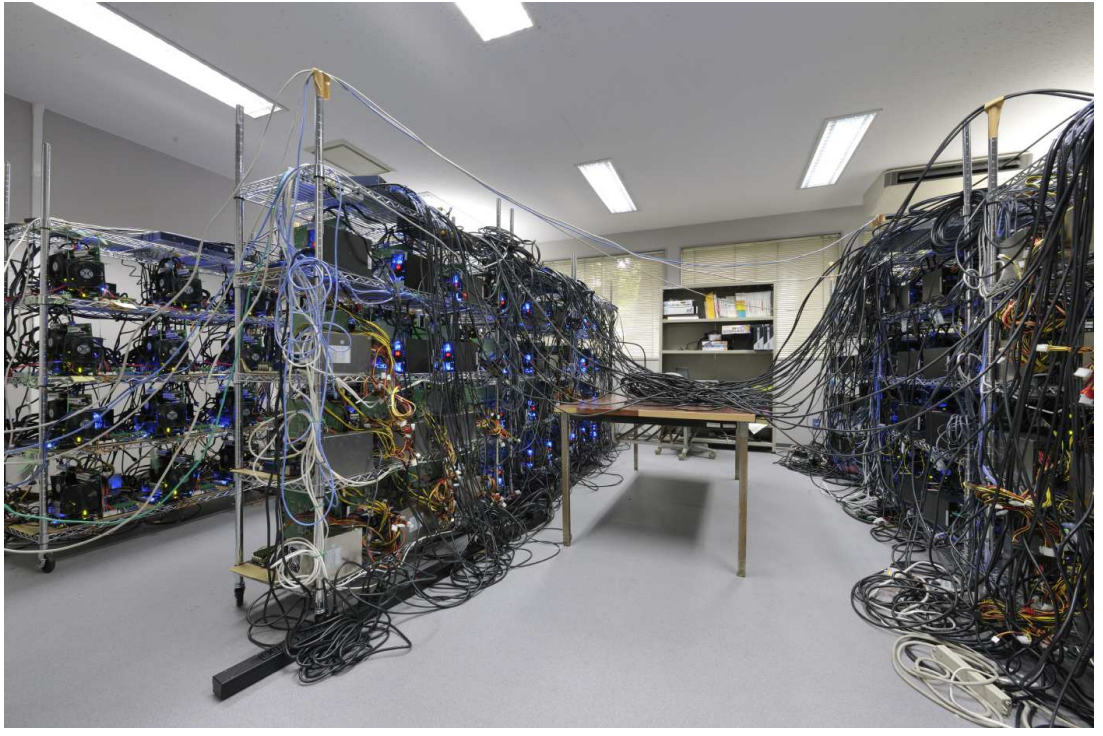
基本的な計算モデル

$$R_i = \sum_j f(x_i, y_j)$$

を並列に評価。

- ２重ループ、一方について積算。
- y_j がなければ単純な並列計算。
- 行列乗算もできるように作る: Top500 狙うため

GRAPE-DR クラスタシステム



GRAPE-DR クラスタシステム

- 128-ノード, 128-カード システム (105TF 理論ピーク @ 400MHz)
- Linpack 実測性能: 39 Tflops@400MHz (100 ノード、理論ピークの半分)
- ホスト計算機: Intel Core i7+X58, 24GB メモリ (100 ノードくらい。残りは 18GB)
- ネットワーク: x4 DDR Infiniband

Little Green500 (2010/6)

Listed below are the Little Green500's Top 10 most energy-efficient supercomputers in the world as of June 2010.

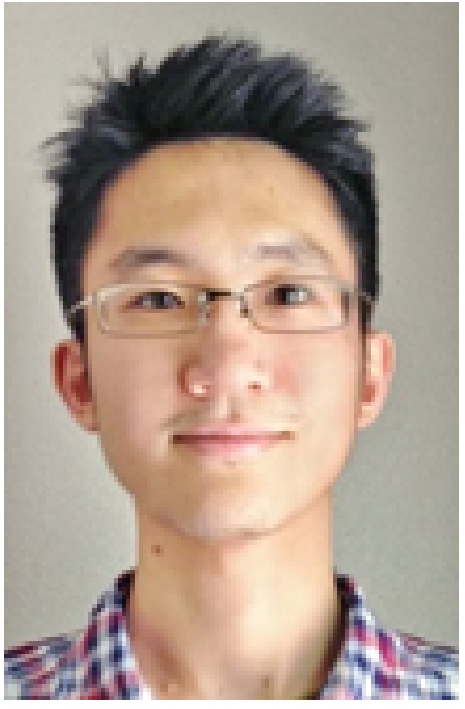
Green500 Rank	MFLOPS/W	Site*	Computer*	Total Power (kW)
1	815.43	National Astronomical Observatory of Japan	GRAPE-DR accelerator Cluster, Infiniband	28.67
2	773.38	Forschungszentrum Juelich (FZJ)	QPACE SFB TR Cluster, PowerXCell 8i, 3.2 GHz, 3D-Torus	57.54
2	773.38	Universitaet Regensburg	QPACE SFB TR Cluster, PowerXCell 8i, 3.2 GHz, 3D-Torus	57.54
2	773.38	Universitaet Wuppertal	QPACE SFB TR Cluster, PowerXCell 8i, 3.2 GHz, 3D-Torus	57.54
5	536.24	Interdisciplinary Centre for Mathematical and Computational Modelling, University of Warsaw	BladeCenter QS22 Cluster, PowerXCell 8i 4.0 Ghz, Infiniband	34.63
6	530.33	Repsol YPF	BladeCenter QS22 Cluster, PowerXCell 8i 3.2 Ghz, Infiniband	26.38

- ベンチマークは Top500 と同じ。 消費電力当りの性能でランキング
- ちょっと小さいめのシステムまであり、のリストで１位。 何故か NHK まで取材にきた。

GRAPE-DR の後

- 「京」の時にはシステム提案はしたけどさっくり無視された。
- 富岳の時には、FS は 筑波大代表で日立が作る前提の「演算加速部」の検討に参加。CPU より 5 倍くらい電力性能はよかったが、諸事情で採用はされず。
- この検討等をベースに、 2016 年から、当時理研のチームメンバーだった村主さんが PFI の設立時メンバーであった縁で、PFN との共同研究による深層が向けプロセッサ開発がスタート。「MN-core」という名前に
- 2020 年 6 月、MN-Core を使った MN-3 システムが Green500 1 位に。NVIDIA A100 を上回る。
- 2023 年、MN-Core 2 完成。

村主さん



- 2010 京大の第一期の白眉フェロー
- 2014 AICS (理研計算科学研究機構) 研究員。差分法の高速化の理論限界の研究、差分法向け DSL Formura の開発等に従事
- 2016 ゴードンベル賞ファイナリスト
- 2017/7/11 死去

スーパーコンピュータアーキテクチャの歴史学と科学(あるいは観測と 理論)

計算機アーキテクチャの世代交代とそれが起こる理由

あまり聞いたことがない人が多いかもしれないが「ベルの法則」(ゴードン・ベル賞のベル) というのがある:

Roughly every decade a new, lower priced computer class forms based on a new programming platform, network, and interface resulting in new usage and the establishment of a new industry.

大体 10 年くらいでアーキテクチャが変わる

スパコンだと

- 1976 年まで: スカラー計算機。最後は CDC 7600
- 1976 年から 1992 年まで: ベクトル (共有メモリ並列含む) 計算機。Cray-1 から C-90 まで。
- 1993 年から 2008 年まで: (マルチコア含む) マイクロプロセッサの分散メモリ並列計算 Cray T3D から、、、Red Storm あたりまで。
- 2008 年から: GPU アーキテクチャのアクセラレータとマイクロプロセッサの組合せ: IBM RoardRunner (Cell なので GPU じゃないけど)

大体 15 年毎。GPU の次はまだでてきてない。

アーキテクチャの変化が必要な理由

近田さんが 1988 年に書いた理由:

> 計算機 (の技術) の使い方を常に見直して、自分の問題に適した構造の機械を使う (作る) ようにし、かつおの上で解き易いように問題を立て直すことがないと「10 年で 100 倍」を生かした人に比べ 10 倍分は立ち遅れることになります。

要するにこれ。

基本的な理由: そのアーキテクチャでは、半導体の進歩を有効に利用できなくなる

- アーキテクチャのスケーラビリティの限界
- 半導体技術の変化

スカラー計算機→ベクトル計算機

- スカラー計算機の進歩: 増えるトランジスタを「1 つの演算器を速くする」に
- 主記憶は磁気コアで、半導体よりはるかに遅いのが想定
- CDC 7600 (S. Cray の設計) あたりが最後
- S. Cray はこの後、4 プロセッサ並列の CDC 8600 を開発していたが、完成前に CDC やめて Cray Research を設立。ベクトルの Cray-1 を開発する。

スカラー計算機では

- IC の開発で使えるゲート数が 1 演算器を超えて大きくなった
- 半導体メモリの開発でメモリが非常に高速になった

ことを生かせなかった。
ベクトル: 半導体メモリは有効利用できた。

ベクトル計算機→マイクロプロセッサ超並列

- ベクトル計算機の進歩: 1 プロセッサのパイプライン数や主記憶を共有するプロセッサの数を増やす
- メモリもプロセッサも多数の IC から構成されて、沢山内部配線があるので、その間の配線も沢山あっても大丈夫、が前提
- Cray-1 くらいのプロセッサが 1 チップにはいると話が破綻。
- 1 プロセッサチップと少数のメモリチップで 1 ノードを構成し、その間はネットワークでつなぐ構成が有利になる。
- 初期の代表: Cray T3D。

マイクロプロセッサ超並列→GPU

- 1チップの中に沢山プロセッサがはいるようになる。
- これは実はシステムレベルでのベクトル並列プロセッサの限界と同じ。
- マルチ(メニー)コアプロセッサでは、キャッシュコヒーレンシを維持した階層キャッシュとオフチップの共有メモリ
- キャッシュコヒーレンシの維持のための電力が支配的に
- GPU では(OS 動かしたりしないので)キャッシュコヒーレンシを緩和したり、捨てたりできる。

GPU→???

- コヒーレンシ諦めても、階層キャッシュで物理的に遠くにデータ移動させること自体が性能の限界になってきている
- と書いた以上、階層キャッシュを諦めて、なるべく物理的に遠くにデータ移動させないのが対応。
- PEZY-SC(キャッシュもあるけど)、Sunway SC26010、MN-Core等では演算器の物理的に近くに大容量メモリ配置

何故横方向のデータ移動が制約になるか？

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**

数字をいれてみると

- 10cm ($10^5 \mu\mathrm{m}$) の配線は 20pF のキャパシタンスがある。これは微細化してもかわらない (線と平面の間のキャパシタンス) 電圧 1V だと、この線は 10pJ/bit を消費する。
- DDR5 メモリは 20pJ/bit くらい。電圧 1.3V とか。。
- LPDDR5 と GDDR6x: 10 pJ/bit くらい。配線が DDR5 のモジュールより短くて電圧も低い。
- HBMx: 3-4 pJ/bit。概ね 25mm くらいまで配線短くなっている。

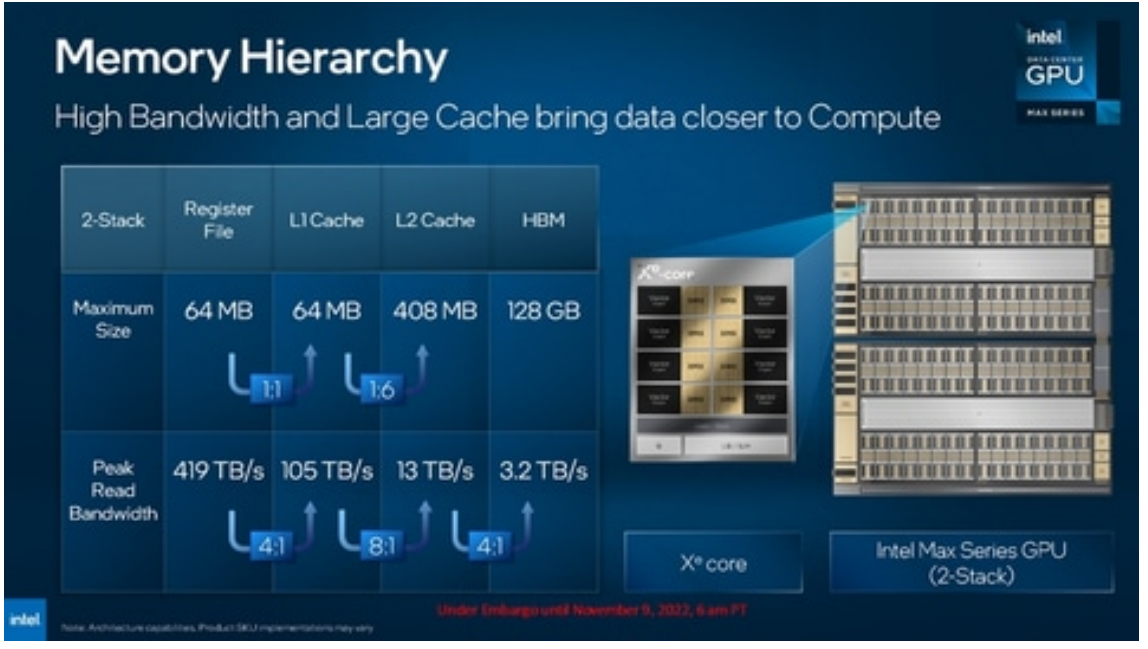
3pJ/bit は 64 ビットワード移動に 0.2nJ なので、HBM でも 5GW 移動すると 1J 消費する。 B/F=4 の計算機作ると 10GF/W しかでない。 0.1 にしても DRAM メモリだけで 400GF/W までしかいかないのでキャッシュの分とかいれると 100GF/W あたり。

つまり

GPU の次のアーキテクチャ に移行するためには、「データのチップ上・基板上の横方向の移動」を限りなく小さくする必要がある。

シリコンフォトニクスでとかいう話があるが、今のところ光変換のコストのほうが大きい

Intel GPU の例



B/F の値
レジスタ: 8? (普通 16 いる。行列乗算器だと 8 はありえる)
L1: 2
L2: 0.25
メモリ 0.06
A64fx に比べても数字が小さい
それでも電力消費が大きい原因になっている

必要な方向

- 基本的には、DRAM をプロセッサダイの上(下かも)につむ。HBM でやってることと変わりはない。
- 但し、 HBM では DRAM ダイの中心に集中しているのは配線を、もうちょっと細かい単位でだす必要がある(横方向の移動を減らす)
- プロセッサアーキテクチャ自体も、演算器の物理的に近くに分散したメモリがある形に移行する必要がある。
- 90 年代の SIMD 超並列計算機(CM-2、MasPar とか)に類似。
- ポスト富岳 FS の神戸大学チームはこういう方向を検討している

後付けの歴史評論ならなんでもいえるけど、もうちょっと理論的にならないか？

そもそも、定量的にアーキテクチャを評価する基準はどうあるべき？
よくわかんないので他の業界を考える。

- **エンジン: 燃料の熱エネルギーをなるべく沢山機械的運動エネルギーに変えられるのがよいエンジン。熱力学第二法則による限界がある。**
- **飛行機: 主翼に対する摩擦抵抗以外は原理的には減らせるはず。誘導抗力、圧力抗力。**
- **(アルゴリズム: 演算数で比較はできる。)**

「汎用」「高性能」は定義可能か？ —現実的な「汎用」性

- まず、何を最小にするものと定義するか？
  - エンジンや飛行機: まずは必要エネルギー。それがコストの全部ではないが大きな割合を占める。
  - 計算機でも、エネルギー消費=電気代は大きな割合になっている。
- エネルギー消費最小は実際的な意味はありそう
  - でも現実的に定義できるか？
- 最近だとチップ自体も高いので、トランジスタ数自体も重要

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？

半導体技術に依存しない定義は可能か？

- 一つの考え方: 半導体技術 (使うプロセスルール、動作電圧) を決める
- そうすると、あるアルゴリズムに従った計算 (計算性能や数値フォーマットも決めたとして) するのに「エネルギー最小」の回路は原理的には決まるはず。
- これは、各演算を必要な最小精度で実現し、演算の組合せ論理だけの消費電力を考えることに相当する。
- メモリアクセス、レジスタアクセス、パイプラインレジスタの電力消費、チップ内でのデータ移動に伴う電力消費はすべて無駄、つまり飛行機の場合の圧力抵抗や誘導抵抗みたいなものと考える。演算は摩擦抵抗。
- 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。(これは半導体技術に依存しない定義になっている)

この定義には意味があるか？

- どうしてもメモリアクセスが多い
- どうしても演算毎にデータが物理的に遠くまで動く必要がある

とかでなければそんなに悪くないはず。実際のアプリケーションではどうかということに。
(アプリケーションの話は今日は省略)

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
- 製造コストに対しては適切な指標
- それなりに意味があるかも？

Table 1: The architectural features of processors discussed

	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%
Sunway 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電力を制御しにくいやり方で消費する。

実行効率の観点

- コヒーレント階層キャッシュっておいしいの?
- コヒーレント階層キャッシュのままで頑張るとどうなるの?

この辺皆様思うところがあるのではないかみたいな。
ちょっと余談: A64fx の例

- メモリから (間接アクセスでも) 連続ロードしてちょっとだけ演算して/ 演算なしでメモリにストアするとかは確かにすごく速い。
- 演算パイプライン、L1 のレイテンシが大きいがアーキテクチャレジスタも OoO 資源も不足。L1, L2 のバンド幅も不足。L2 レイテンシが他のマシンに比べて数倍大きい
- 演算沢山あっても演算でパイプライン埋めるのはすごく難しい

「理想に近い」プロセッサの例

(現在の半導体技術で、、、)

- 専用計算機ならもちろんそうできる。GRAPE-3,4,5,6 とか。
  - GRAPE-4、6 は演算器 1 つ当たり 20k トランジスタくらい (FMA ユニット換算)。普通の FMA ユニットは 100k トランジスタくらい必要。
  - なんでも倍精度でやるのに比べて 5-10 倍効率あげられる。
- 汎用プロセッサでは、、、手前味噌だけど GRAPE-DR の例を。
  - チップレベル SIMD
  - ローカルメモリ+レジスタ+階層的チップ内ネットワーク
  - チップ内ネットワークは放送/縮約をサポート
  - 演算器の割合 14%くらい。確かに高い。電力性能はチップレベルで 3GF/W くらい (90nm)。
  - まあ「汎用」といえるか？という問題点はある。
  - トランジスタ効率では専用回路に比べると 10 倍以上悪い。

主要な問題点

- GRAPE-DR くらい極端な設計でもまだ専用回路に比べると数十倍損
- GRAPE-DR みたいな極端な設計で本当に色々使えるか？プログラムどうやって書くのか？

(前者は、粒子系以外ではそこまで差がないかも)
後者はちゃんと考える必要あり。ある意味、現在のポスト富岳 FS ではこの辺をやっている。
まあなんとかなりそうな気がする。

「京」、富岳を振り返ると

- 「京」は微妙だが、富岳では完全に半導体技術の変化とそのために必要になったアーキテクチャの変化に対応できていない。
- このために、非常に無理があるプロジェクトになり、商用展開ができていない。
- ポスト富岳でこれを繰り返してはいけない。
- 売れる(=他の人達がお金だしても使いたいと思う)ものを作る計画になっていたか？の振り返りが必要だと思う。

まとめ

- 球状星団シミュレーションから「汎用」スパコンまで、なんだかいきあたりばったりにやってきた気も。
- 計算機シミュレーションでできる範囲を広げる、半導体技術を有効に使う、そのために問題設定から、というと近田さんのいってたことをやってきただけかも。
- ポスト富岳では、GPU の「次」のアーキテクチャが必要。DRAM も演算器の物理的に近くに、が鍵。

皆様ありがとうございました。

皆様ありがとうございました。
まだしばらくよろしくお願いいたします。