

並列計算プログラミング・ 先端計算機演習の概要

2011年夏季集中講義

中島 研吾

並列計算プログラミング(616-2057)・先端計算機演習(616-4009)

自己紹介

- 略歴

- 工学部航空学科出身，博士（工学）
- 株式会社三菱総合研究所等
- 2004年～：地球惑星科学専攻（多圏COE特任教員）
- 2008年～：情報基盤センター

- 専門

- 数値流体力学
- 並列プログラミングモデル，大規模数値解法

- 地球惑星科学とのかかわり

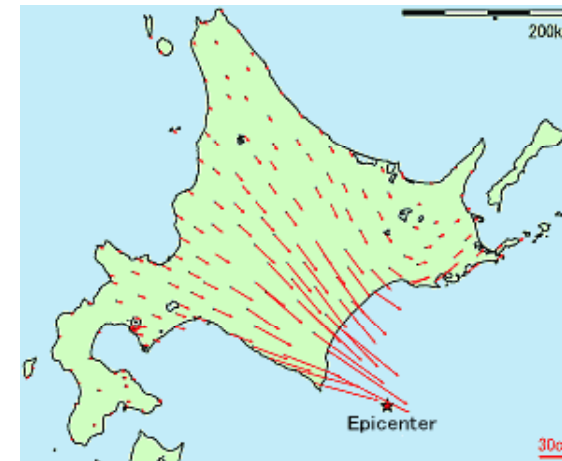
- 平成10～14年度：文部科学省科学技術振興調整費「高精度の地球変動予測のための並列ソフトウェア開発に関する研究」
- 平成17～22年度：科学技術振興機構「観測・計算を融合した階層連結地震・津波災害予測システム」（研究代表：松浦充宏教授（地球惑星科学専攻））

並列計算の意義

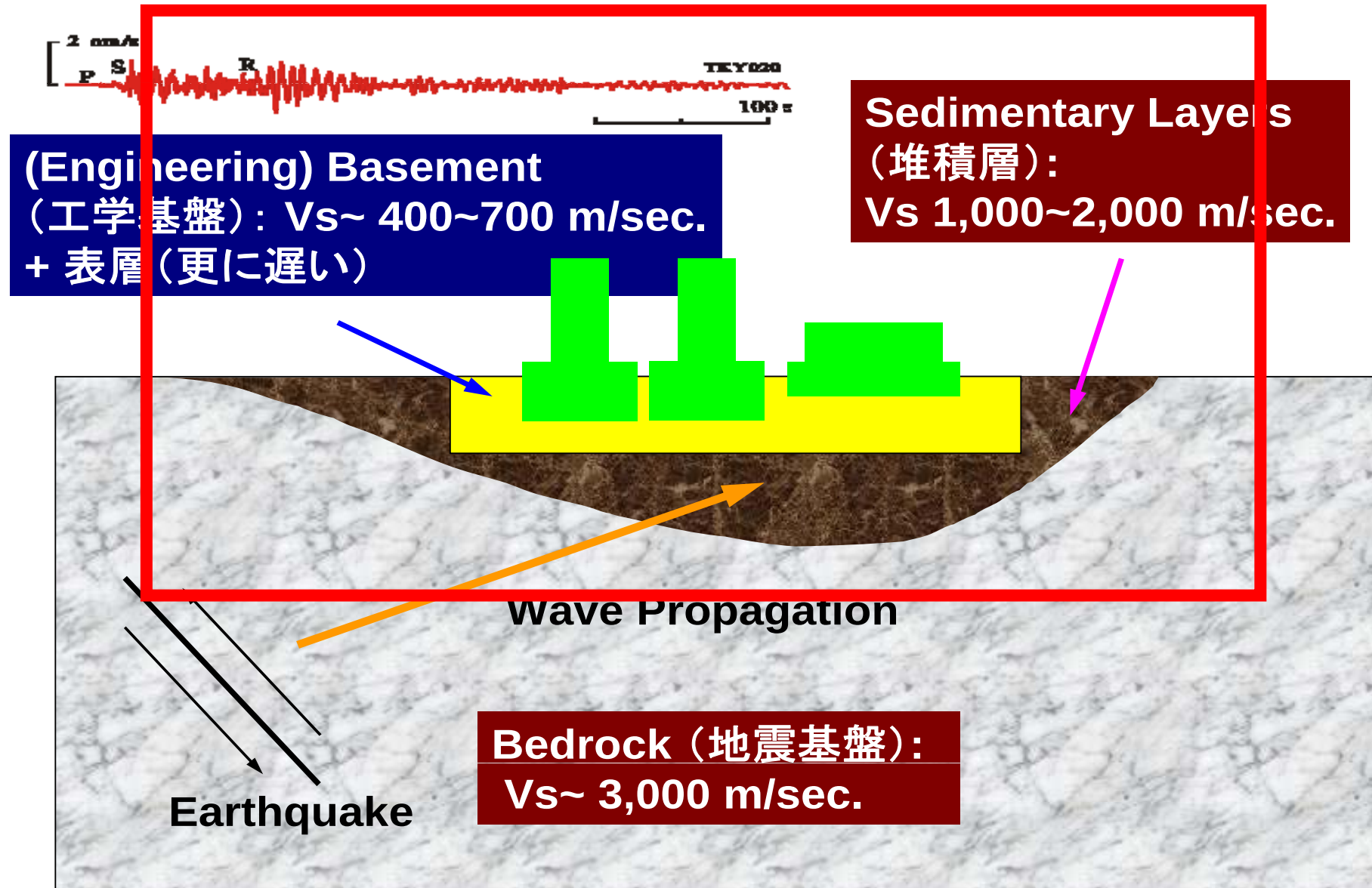
- たくさんの計算機を使って
- より速く
- より大規模に
- より複雑に
 - 連成, 連結シミュレーション

2003年 十勝沖地震

長周期地震波動(表面波)のために苫小牧の
石油タンクがスロッシングを起こし火災発生

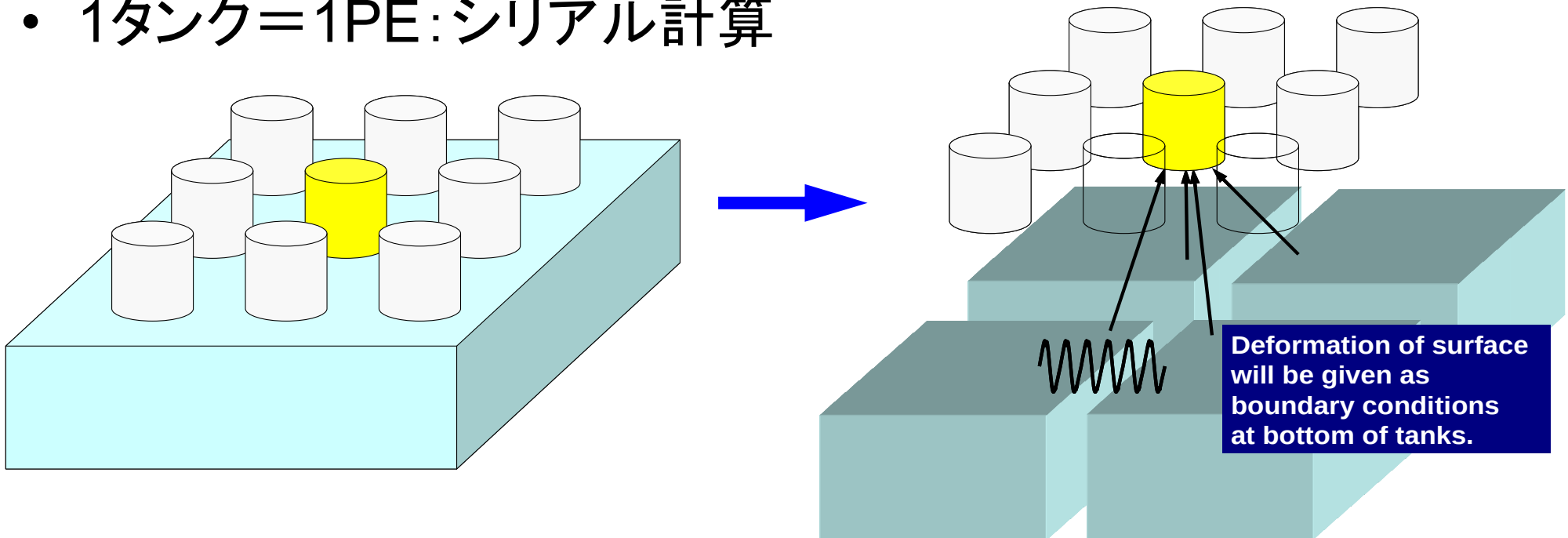


地盤・石油タンク振動連成シミュレーション



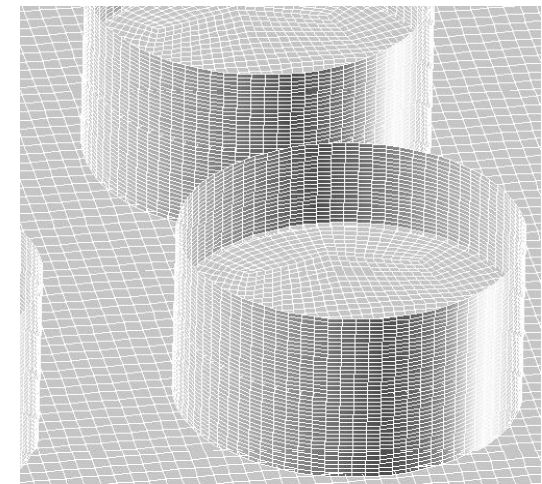
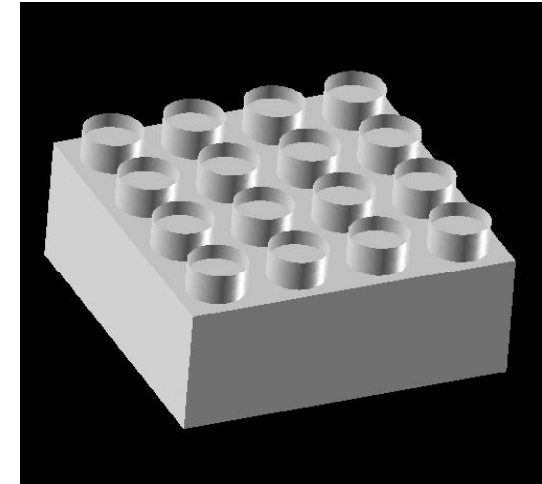
対象とするアプリケーション

- 地盤・石油タンク振動
 - 地盤⇒タンクへの「一方向」連成
 - 地盤表層の変位 ⇒ タンク底面の強制変位として与える
- このアプリケーションに対して、連成シミュレーションのためのフレームワークを開発，実装
- 1タンク=1PE:シリアル計算

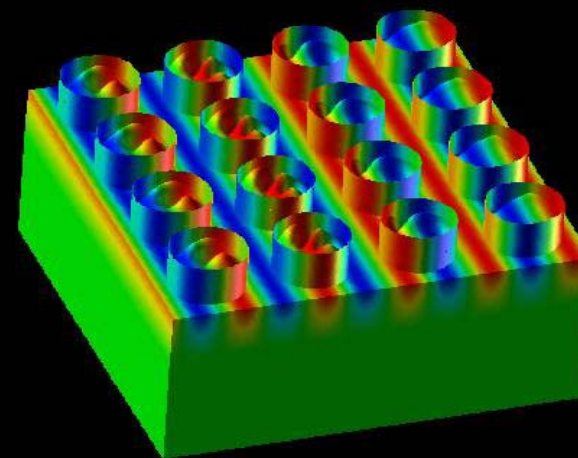
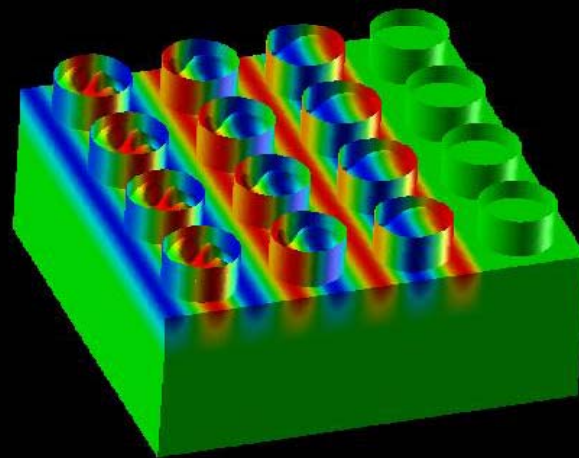
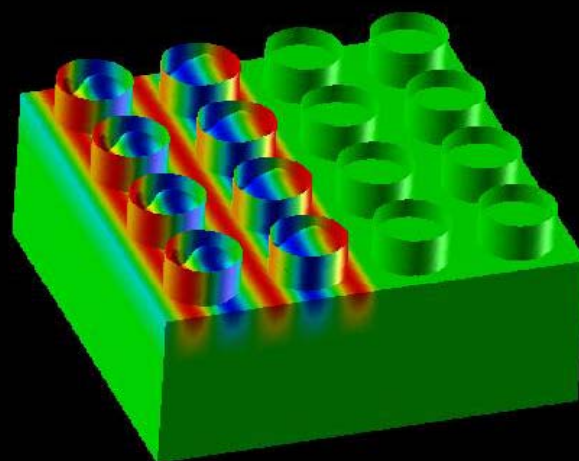
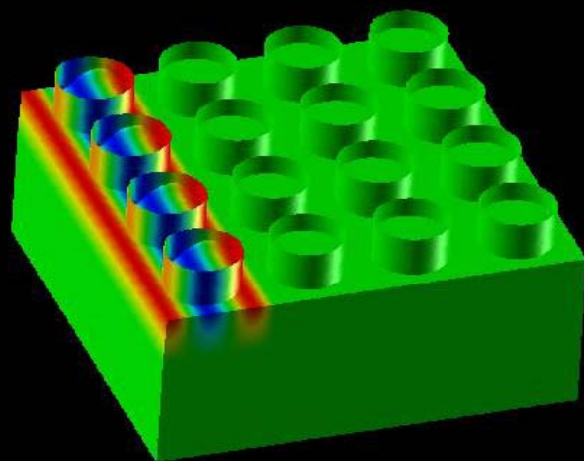


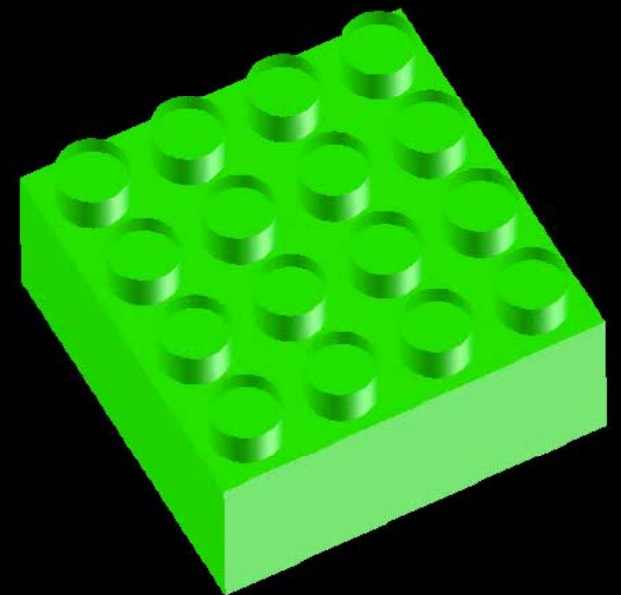
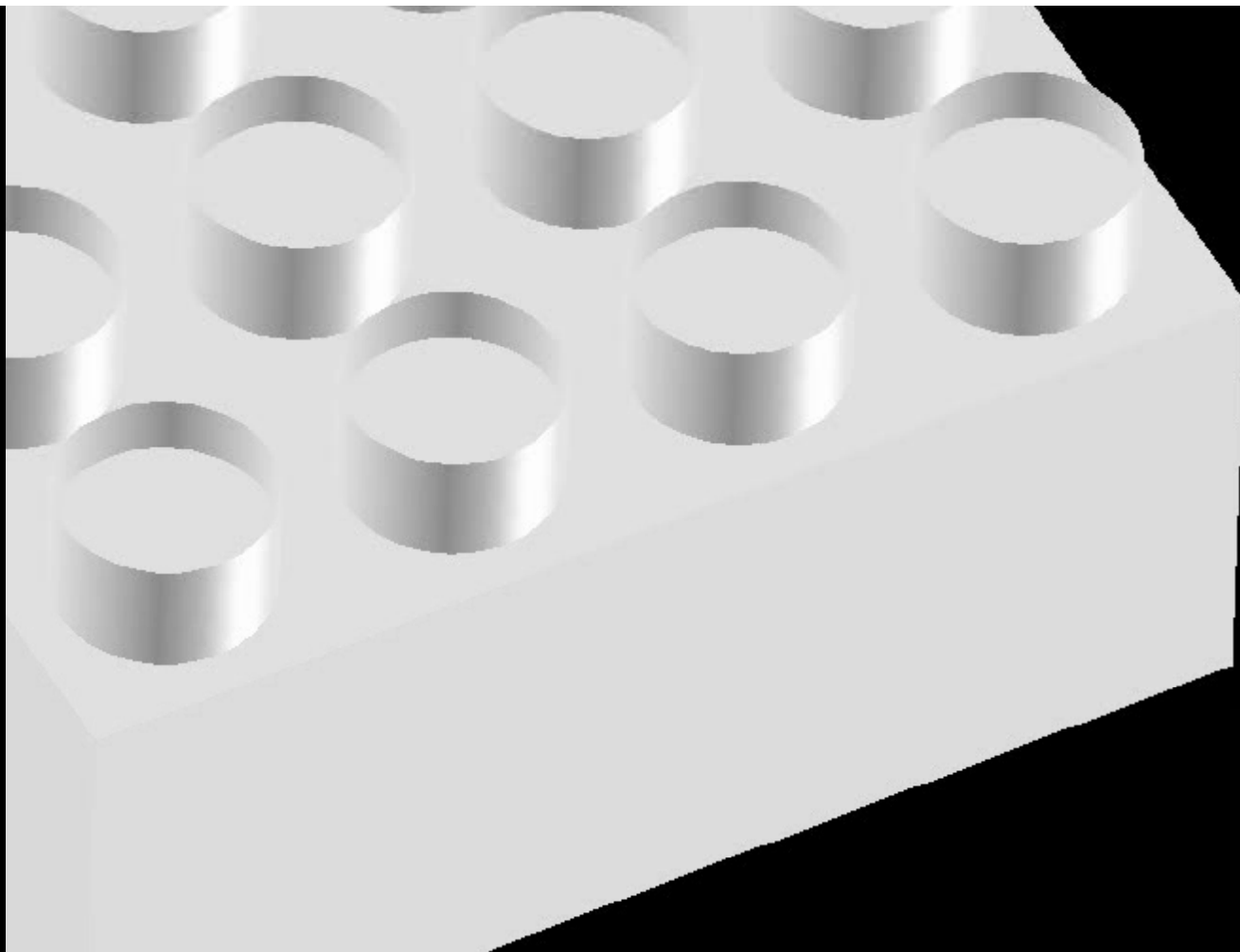
地盤，タンクモデル

- 地盤モデル：FORTRAN
 - 並列FEM, 三次元弾性動解析
 - 前進オイラー陽解法, EBE
 - 各要素は一辺2mの立方体
 - $240\text{m} \times 240\text{m} \times 100\text{m}$
- タンクモデル：C
 - シリアルFEM(EP), 三次元弾性動解析
 - 後退オイラー陰解法, スカイライン法
 - シェル要素+ポテンシャル流(非粘性)
 - 直径:42.7m, 高さ:24.9m, 厚さ:20mm, 液面:12.45m, スロッシング周期:7.6sec.
 - 周方向80分割, 高さ方向:0.6m幅
 - 60m間隔で4×4に配置
- 合計自由度数:2,918,169



計算結果





並列計算の意義

- たくさんの計算機を使って
- より速く
- より大規模に
- より複雑に
 - 連成，連結シミュレーション

(地震＋津波)連成
古村，齊藤

計算科学

概 要

- 本授業・演習の概要
 - 目的, 方針
 - 概要
 - スケジュール
- HPCの概要
 - 計算機ハードウェアの発達
 - 並列プログラミング言語

本授業の理念（HPより）（1/4）

- 近年，マイクロプロセッサの処理速度の上昇，並列計算機の発達によって，計算機システムの処理能力は飛躍的に増加している。世界の並列計算機の動向をまとめた「[TOP500リスト](#)」によると，世界最高速の計算機の速度は1993年6月から2011年6月の間に約10万倍近くにもなっている。2011年6月現在で世界最高速（約8.162 PFLOPS，PFLOPSとは「PETA Floating Point Operations Per Second」の略，1秒間に1000兆回の浮動小数点演算）の「[京コンピュータ](#)」は548,352台のコアから構成されている。
- 並列計算機の使用によって，より大規模で詳細なシミュレーションを高速に実施することが可能になり，新しい科学の開拓が期待される・・・しかしながら，いざ，自作のプログラムを並列計算機で動かそうとすると中々容易ではない。

本授業の理念（HPより）（2/4）

- 参考になる文献も少なく，英語のものが多い。これまで，計算機を専門としない学生に対して科学技術シミュレーションのための並列プログラミング技術を体系的に教える授業は，日本では皆無であった。多圏地球COEの一環として平成16年度から開講された「並列計算プログラミング」，「先端計算機演習」は，そうした試みの日本における最初のものの一つである。
- FORTRAN, C言語などで記述されたプログラムを並列計算機上で並列化するための手段の代表的なものとして [MPI\(Message Passing Interface\)](#) というプロセッサ間通信のための共通規格がある。MPIには400以上の関数があるが，科学技術シミュレーションにおいて必要になるのは10程度である。本授業では，科学技術シミュレーション手法を，局所的手法（差分法，有限要素法等）と大域的手法（境界要素法，スペクトル法等）に分類し，それぞれを並列化するために必要な最小限のMPI関数（1対1通信，グループ通信）について教え，あとはできるだけ実習によって経験を積んでもらうこととする。

本授業の理念（HPより）（3/4）

- アメリカのメリーランド大学 (University of Maryland) で Applied Mathematics and Scientific Computation Program というコンピュータサイエンスと科学技術シミュレーションのジョイントプログラムを主宰している [David Levermore教授](#) によると，科学技術シミュレーションの真髄は「**SMASH**」，すなわち，
 - Science
 - Modeling
 - Algorithm
 - Software
 - Hardware

であるという。本授業でカバーするのは，Algorithmの一部とSoftware全般，Hardwareの一部である（とは言え，Science，Modelingについても結構教えている）。

Scientific Computing = SMASH

David Levermore (Univ.Maryland)

<http://www.math.umd.edu/~lvrmr/>

Science

Modeling

Algorithm

Software

Hardware

- これだけ幅広い分野を扱わなければならない
- 分野間の協力の重要性を示しているとも言える。
- 一人で全てをカバーすることは難しいが、「並列」シミュレーションコードを全て自力で開発するためには、これだけの範囲の知識と経験はそれぞれある程度必要
 - 例：チューニング（地球シミュレータ）

本授業の理念（HPより）（4/4）

- 本授業の中で強調したいことは以下の4点である：
 - 並列計算プログラミングは決して難しくない。
 - 計算機に使われてはいけない。
 - 良い並列プログラムは、良いシリアルプログラム（serial program, 単独CPUのためのプログラム）から生まれる。
- 授業、実習で学んだことが受講者がシミュレーションによる研究を本格的に実施する際に少しでも助けになればと願っている。もちろん授業内容が直接役に立つにこしたことはないが、それよりも「並列計算は難しいものではない」という意識を、本授業、実習を通して持ってくれることが最も重要である。

計算科学＝第三の科学

- ・ ・ ・と言われて久しいが，現実には実験，観測の後追いに過ぎない。それでも実験より手軽に実施可能である。
- 第一原理的手法を使用すれば，実験不可能なことでも，シミュレーションによって解明される，であろうことが明らかになりつつある。
 - － バイオ，ナノテクノロジー

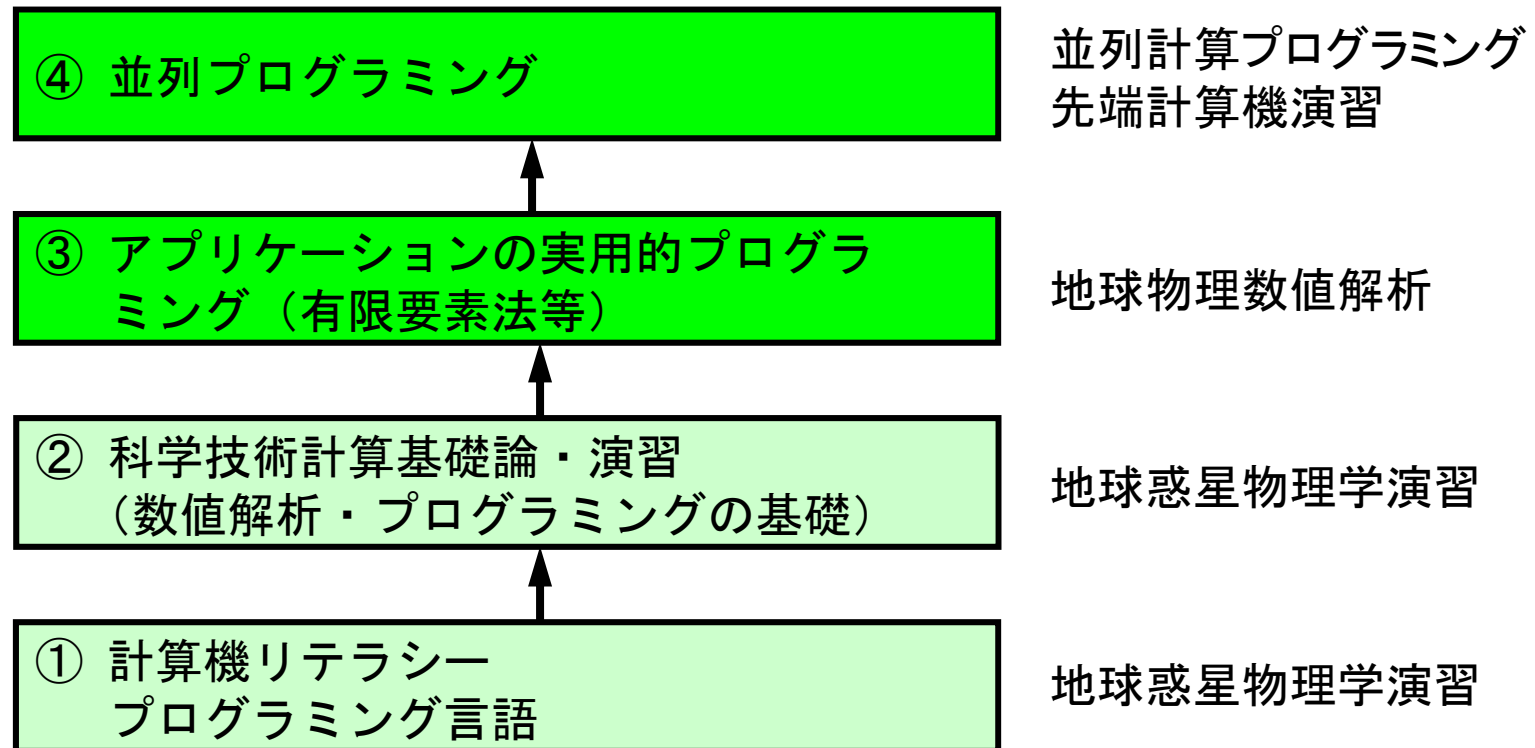
地球惑星科学とシミュレーション

- 地球惑星科学
 - 観測，実験とそれに基づく理論の重要性
 - シミュレーション技術が比較的発達しているのは，観測データが豊富な大気・海洋分野
 - 観測が難しい分野においてはシミュレーションはより一層重要なはずである．．．
 - 実際問題として検証ができないとシミュレーションも進歩しない
 - 大規模な解析空間 ⇒ 大規模計算の必要性
 - 全地球，連成現象：Multiscale, Multiphysics
- 地球シミュレータ
- 地球惑星科学 多圏COE
 - 大規模シミュレーション技術に関する体系的教育の必要性

本学科，専攻における計算機教育

- 地球惑星物理学演習：3年夏
 - 計算機リテラシー
 - FORTRAN
 - 基本的な数値解析
- 地球物理数値解析：4年夏，大学院（2005～）
 - 偏微分方程式の数値解法
 - 差分法，有限要素法
- 並列計算プログラミング，先端計算機演習I／II（2004～）
 - 有限体積法，境界要素法
 - SPMDの考え方，MPI，OepMP
 - <http://nkl.cc.u-tokyo.ac.jp/class/>
- この他「地球物理データ解析」，「時系列データ解析」など

並列プログラミングへの道：積み重ね重要



学際計算科学・工学 人材育成 プログラム(東京大学)

- 情報基盤センター, 理学系研究科, 情報理工学系研究科, 工学系研究科, 新領域創成科学研究科, 生産技術研究所によって, 全学的なHPC教育プログラムの整備が検討されている(2008.2.~)
- 地球惑星物理学科・地球惑星科学専攻における取り組みがモデルとなっている
 - 1990年代初頭から世界的にも他に類を見ない充実した計算機・プログラミング教育が行なわれている
 - FY.2004~: 並列プログラミング教育(21世紀COE)
- 平成21年度から段階的に実施

並列計算プログラミング教育



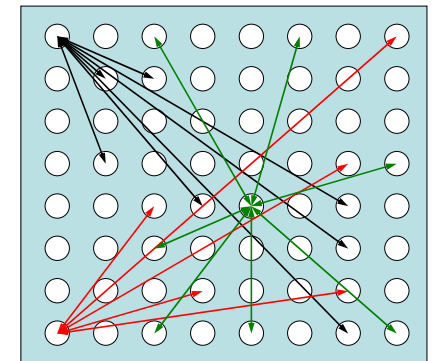
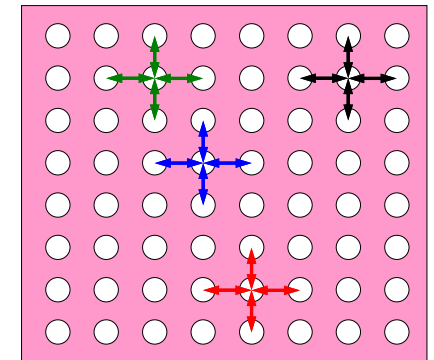
- 「並列計算プログラミング」, 「先端計算機演習I・II」は「地球シミュレータ」に代表されるような大型並列計算機を使いこなし, 未解決の問題にチャレンジしていくような研究者を育成することが重要な目標の一つである。
- これまで, 計算機科学を専門としない学生に対して科学技術シミュレーションのための並列プログラミング技術を体系的に教える授業は, 日本では皆無であった。
- 「並列計算プログラミング」, 「先端計算機演習」は, そうした試みの日本で最初の内の一つであると自負している。
 - 現在でもこれだけの内容をカバーするものは世界的に見ても他に例は無い！

本授業・演習の目的

- 科学技術計算における様々な手法・アルゴリズムを理解し、各アルゴリズムにおいて並列性を引き出す様々な考え方を学ぶ
 - まずは、元のアプリケーションの中身(アルゴリズム, 実装)をよく理解していること, そうすれば「並列化」は難しくない
- 並列計算プログラミング技法の習得
 - MPI, OpenMP
 - 情報の探し方の習得: これは重要
- SMASH
 - ASHが中心となるが, Science, Modelingを抜きにして, 科学技術計算プログラミングは語ることは不可能である
 - できるだけ誰でも理解できる「SM」を選択

題 材

- 並列化のアプローチが異なる2種類の離散化手法(M)
 - Scienceとしては理解しやすい「熱伝導」を中心
- 局所的手法(差分法, 有限要素法等)
 - 有限体積法(ガウス・グリーン型)
 - 疎行列
- 大域的手法(境界要素法, スペクトル法等)
 - 粒子間熱伝導
 - 密行列
- 幅広いバックグラウンドの受講生への配慮
 - 各大講座
 - 天文, 物理, 工学系, 情報理工学系
 - FORTRAN, C



並列プログラミング言語

- メッセージパッシングライブラリ
 - MPI, PVM
 - 複雑と言われているが、柔軟な処理可能、移植性も大。
- 並列化コンパイラ
 - HPF (High Performance FORTRAN)
 - プログラミングは簡単だが(配列の分割), 融通が効かない。複雑な問題に対しては効率も出にくい。移植性も悪い・・・多分実際に使っているのは一部の「地球シミュレータ」ユーザーのみ。
- SMP用ディレクティブ
 - OpenMPなど
 - 共有メモリユニット用の並列化用
 - SMPクラスタではMPIと組み合わせた「ハイブリッド」プログラミング

FY.2004~2007で感じた問題点

- ③の部分がそもそもちゃんとできていない
 - ①, ②を知っていても③へ行くのは難しい
 - ①計算機リテラシー, プログラミングの初歩
 - ②基本的な数値解析アルゴリズム
 - ③アプリケーションの実用的プログラミング(有限要素法等)
 - ④並列計算プログラミング
- 「並列計算プログラミング」の中でも結局③レベルのことを教える必要があった
- 地球惑星物理学科・地球惑星科学専攻の教育環境は理想に限りなく近い・・・とは言え
 - 修士課程入学者の半数以上は学外から進学
 - 「地球惑星物理学演習」に相当するような訓練を受けているものは稀

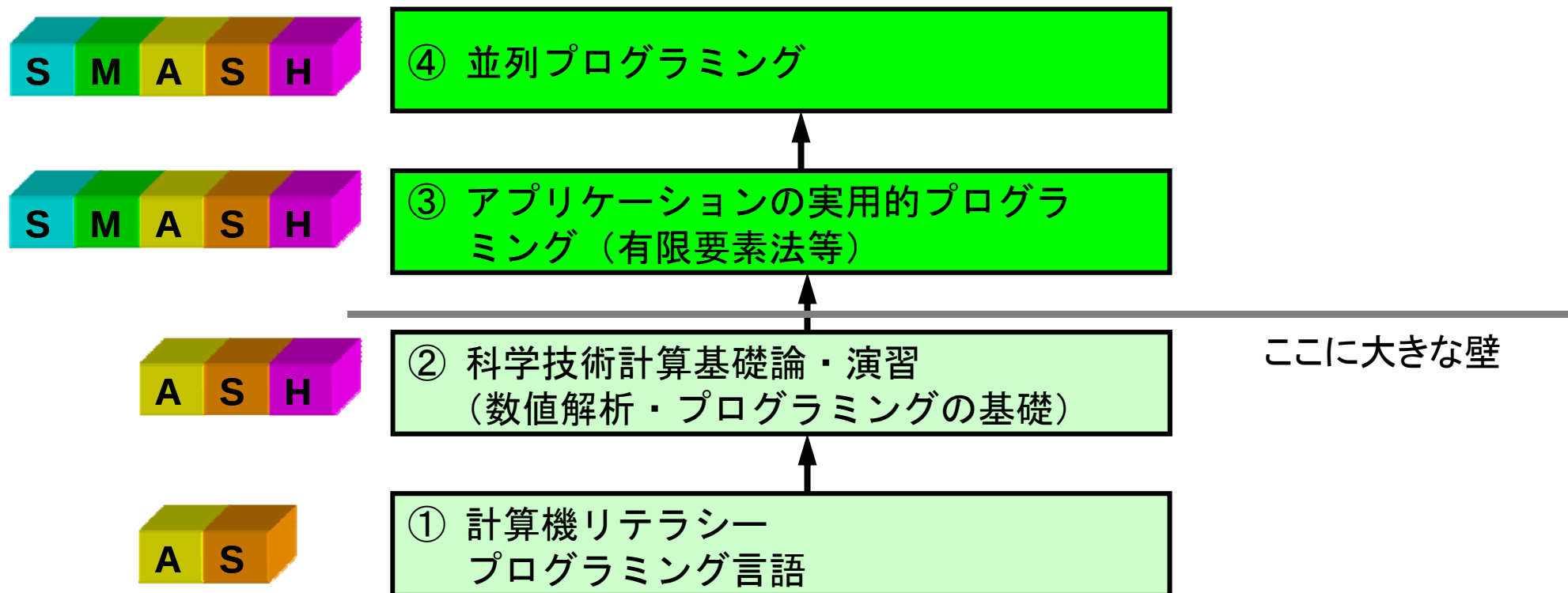
並列プログラミングへの道



実際の進め方

- SMASHをバランスよく教えます
 - 幅広い視野
- ③, ④をカバー, 場合によっては②も含む
 - ①は既習として進めます
- 大学院からの進学者にできるだけ配慮します
- ③の勉強の仕方
 - プログラミング能力(SMASH)をつけるために, 徹底して実アプリケーションコードのソースを「読む」能力をつけるところに重点を置きます
 - 英語, 漢文の音読のごとく
 - 並列計算で大事なのはMPI等の文法ではなく, 並列データ構造等の設計です。対象アプリケーション, アルゴリズムに対する深い理解が必要となります⇒③をしっかり勉強しておくことが結局④にもつながります

並列プログラミングへの道



(再び) 本授業・演習の目的

- 科学技術計算における様々な手法・アルゴリズムを理解し、各アルゴリズムにおいて並列性を引き出す様々な考え方を学ぶ
 - まずは、元のアプリケーションの中身(アルゴリズム, 実装)をよく理解していること, そうすれば「並列化」は難しくない
- 並列計算プログラミング技法の習得
 - MPI, OpenMP
 - 情報の探し方の習得: これは重要
- SMASH
 - ASHが中心となるが, Science, Modelingを抜きにして, 科学技術計算プログラミングは語ることは不可能である
 - できるだけ誰でも理解できる「SM」を選択

本授業・演習の目的（続き）

- 並列計算技術のあり方を考える端緒としたい
 - 自分の研究，分野にとって必要な計算機，計算技術とは？
 - 計算機に使われるのではない，使う立場で考える
- 最終的には，「第三の科学」を開拓するための術（すべ）となれば幸いである。

担当者，時間割，講義室

- 担当教員

- 中島研吾（情報基盤センター）

- 情報基盤センター（浅野）別館3F, ex : 22719

- e-mail : nakajima(at)cc.u-tokyo.ac.jp

- 講義

- 9月12～16日，9月20～22日

- 09:00～12:15, 13:30～15:00（9月14日は午前のみ）

- 演習：自由参加

- 9月12・13・15・16・21日（14・20・22日も部屋，端末は使えます）

- 15:15～16:45

- 講義室

- 情報基盤センター1F 大演習室2

スケジュール (1/2)

日付	時間	番号	内容
9月12日(月)	0900-1030	CE01	イントロダクション
	1045-1215	CE02	一次元熱伝導解析(1/2)
	1330-1500	CE03	一次元熱伝導解析(2/2), T2Kオープンスパコン(東大)
	1515-1645	演習	演習(自由参加)
9月13日(火)	0900-1030	CE04	MPIによるプログラミング概要(Ⅰ)(1/2)
	1045-1215	CE05	MPIによるプログラミング概要(Ⅰ)(2/2)
	1330-1500	CE06	MPIによるプログラミング概要(Ⅱ)(1/3)
	1515-1645	演習	演習(自由参加)
9月14日(水)	0900-1030	CE07	MPIによるプログラミング概要(Ⅱ)(2/3)
	1045-1215	CE08	MPIによるプログラミング概要(Ⅱ)(3/3)
9月15日(木)	0900-1030	CE09	並列アプリケーション開発法(Ⅰ)有限体積法(1/2)
	1045-1215	CE10	並列アプリケーション開発法(Ⅰ)有限体積法(2/2)
	1330-1500	CE11	チューニング入門
	1515-1645	演習	演習(自由参加)
9月16日(金)	0900-1030	CE12	チューニング入門(演習)
	1045-1215	CE13	課題S1解説
	1330-1500	CE14	並列アプリケーション開発法(Ⅱ) 有限体積法:局所分散データ構造, 領域分割(1/3)
	1515-1645	演習	演習(自由参加)

スケジュール (2/2)

日付	時間	番号	内容
9月20日(火)	0900-1030	CE15	並列アプリケーション開発法(Ⅱ) 有限体積法:局所分散データ構造, 領域分割(2/3)
	1045-1215	CE16	並列アプリケーション開発法(Ⅱ) 有限体積法:局所分散データ構造, 領域分割(3/3)
	1330-1500	CE17	課題S2解説
9月21日(水)	0900-1030	CE18	並列アプリケーション開発法(Ⅲ)有限体積法:並列化(1/2)
	1045-1215	CE19	並列アプリケーション開発法(Ⅲ)有限体積法:並列化(2/2)
	1330-1500	CE20	並列アプリケーション開発法(Ⅳ)粒子間熱伝導(1/2)
	1515-1645	演習	演習(自由参加)
9月22日(木)	0900-1030	CE21	並列アプリケーション開発法(Ⅳ)粒子間熱伝導(2/2)
	1045-1215	CE22	最近の話題(1/2)
	1330-1500	CE23	最近の話題(2/2)

評価（案）

- 「並列計算プログラミング」, 「先端計算機演習」の評価は一体として実施する: 実習課題レポート
- 課題
 - MPI例題(その1)(グローバル通信)(課題S1)
 - MPI例題(その2)(共役勾配法)(課題S2)
 - チューニング関連(課題S3)
 - 熱伝導解析コードによる並列計算実習(課題P1)
 - 解説を実施, 模範解答を公開する。
 - 提出期限: 10月21日(金)17:00

課題・評価についての考え方

- 色々なレベルがある
 - 全部自分でプログラムを作る
 - 解説を聞いてから自分でプログラムを作る
 - 解説に従ってやる
 - 敢えて解説に従わないでやる
 - 模範解答のプログラムを使って計算する
- 「模範解答のプログラムで計算するだけでは意味がない」と思うかも知れないがそんなことはない。何らかの形で全ての課題をこなすことを心がけてほしい。
 - 少なくともプログラムは「読んで」ください

課題・評価についての考え方（続き）

- プログラミングの習得のためには正しいやり方で実習によって経験を積むしかない。そのための課題である。
 - しっかりやれば、自分の研究に応用することも容易にできるに違いない。これがベスト。
 - 資料を見れば必要になったときにいくらでも習得できる、「並列計算プログラミング」はその程度のものである、という考え方も決して悪くは無い（そのくらいのマテリアルは用意するつもり）。積極的に勧めるものではないが。
- とにかく将来、何らかの形で受講者の今後の研究、仕事の役に立つような課題を設定していくつもりである。

ホームページ等

- <http://nkl.cc.u-tokyo.ac.jp/11e/>
 - クラスタ使用法等を含めて, 最新の情報を随時追加する。
 - 資料の印刷は各自でお願いします。
- 履修用アンケート(まだの人は至急中島宛送付)

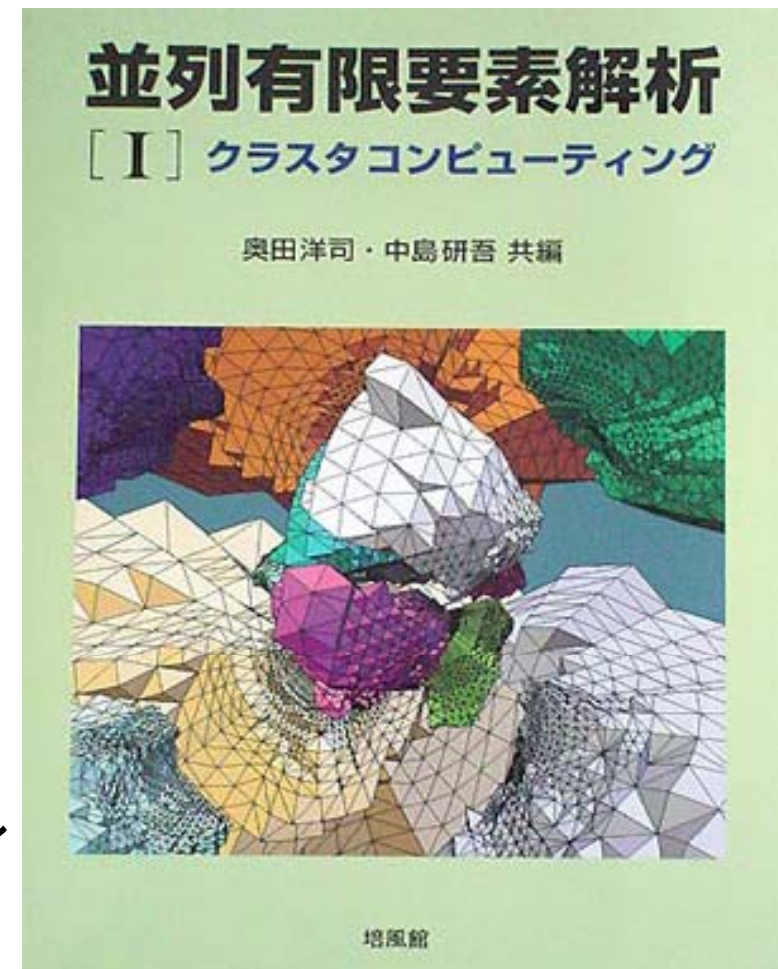
参考文献

- いろいろな書籍が出版されているが、これ、と言ったものはない。
 - 全てに目を通していてもない。
 - 特に和書には中々適切なものが無い
 - 計算機科学の専門家向けのものが多い:この分野の研究者が多いこともある
- とりあえず、特に推薦できる書籍をHPに示す。これらは全て中島のところにある。

奥田，中島編「並列有限要素解析〔I〕クラスタコンピューティング」

培風館，2004.

- 「GeoFEM」の成果のまとめ
 - <http://geofem.tokyo.rist.or.jp>
- 「地球シミュレータ」上での最適化，シミュレーション結果を紹介
- 初心者向けでは無い
- 高い・・・
 - 若干残部があるので希望者には貸し出します。



その他：演習，質問

- 情報基盤センター演習室の端末を使用する
 - アカウントを取得しておくこと
 - 講習会を受講する必要あり
 - <http://www.ecc.u-tokyo.ac.jp/>
- わからないことがあったら，遠慮なく質問してください
 - 「在室」の場合はいつでもOKです。
 - 「演習」のコマを有効に利用してください

その他：計算機環境

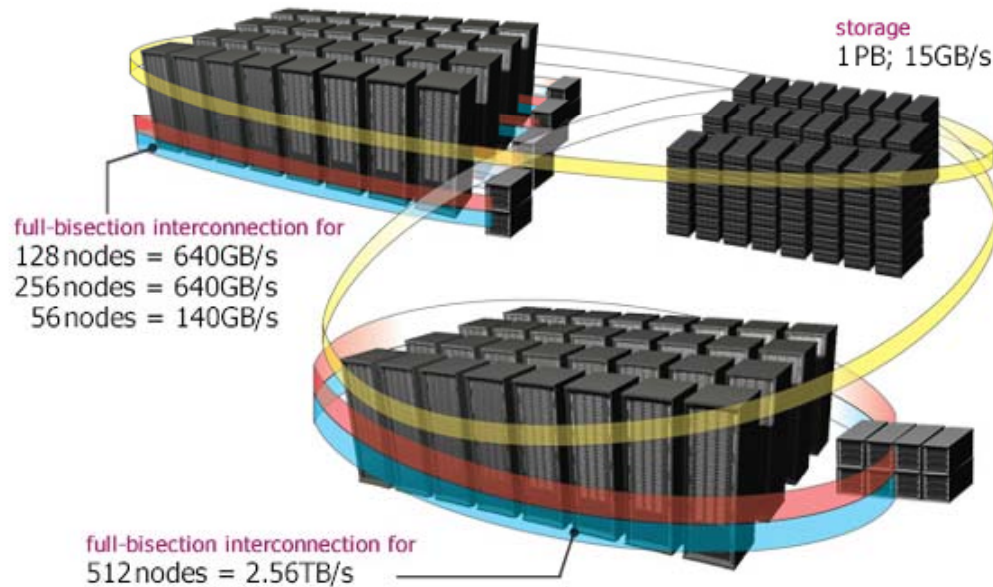
- 情報基盤センター
 - ECCS2008(教育用計算機システム)
 - Hitachi HA8000(T2Kオープンスパコン):10月末まで使用可能

The T2K Open Supercomputer Alliance

<http://www.open-supercomputer.org/>

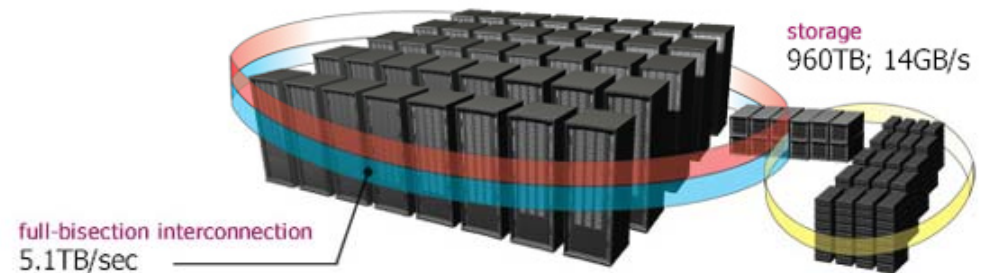
University of Tokyo

nodes = 952 Rpeak = 140.1TFlops Memory = 31TB



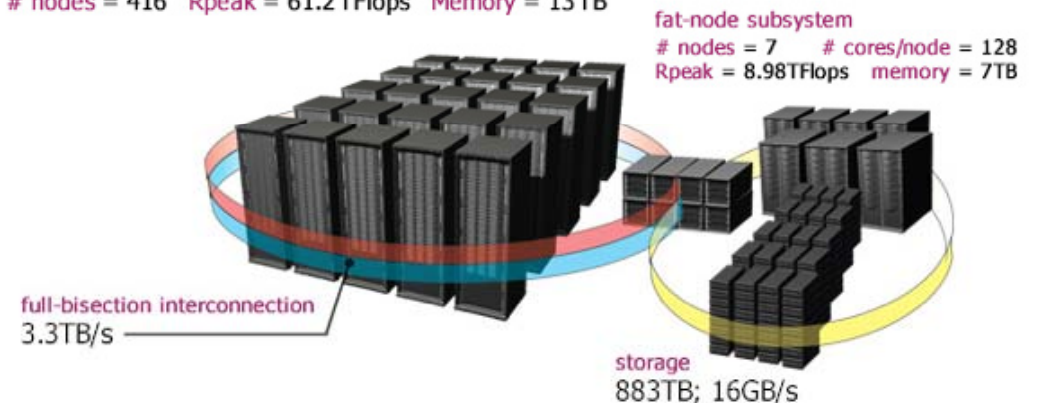
University of Tsukuba

nodes = 648 Rpeak = 95.4TFlops Memory = 20TB



Kyoto University

nodes = 416 Rpeak = 61.2TFlops Memory = 13TB

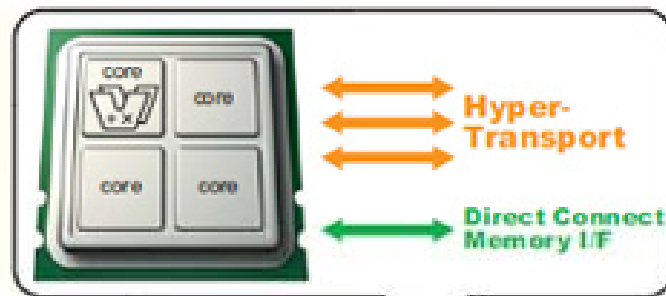


Tsukuba: Appro-Cray
Tokyo: Hitachi
Kyoyo: Fujitsu

The T2K Open Supercomputer Alliance

<http://www.open-supercomputer.org/>

Opteron 8350 Barcelona
R_{peak} = 36.8GFlops

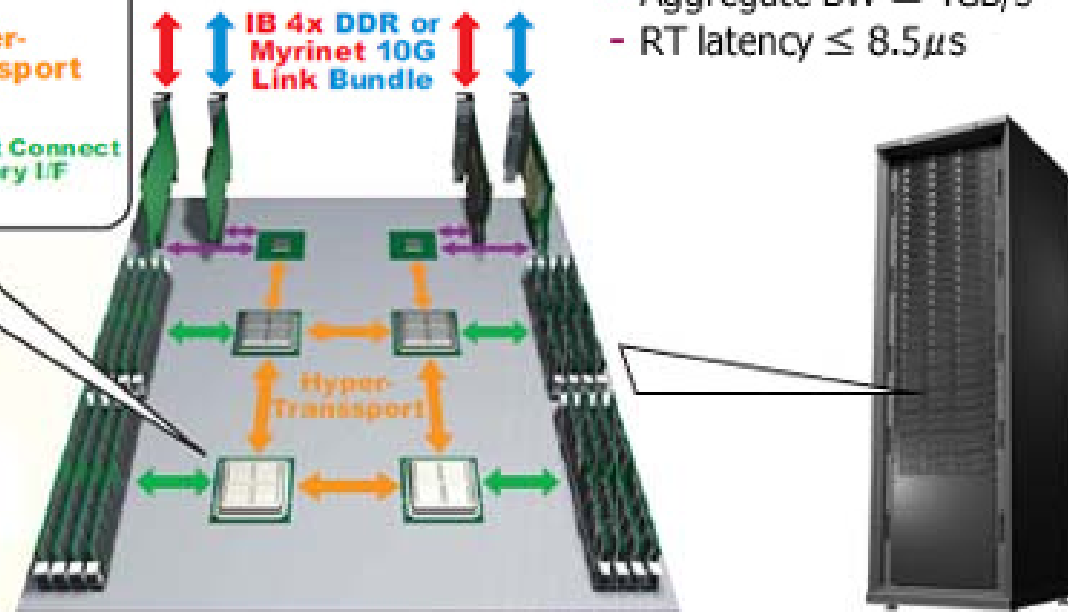


Per-Node Performance

- ▶ R_{peak} = 147GFlops
- ▶ Memory
 - Capacity = 32GB
 - Aggregate BW = 42.6GB/s

Inter-Node Link Performance

- ▶ Aggregate BW = 5 or 8GB/s
- ▶ MPI Performance
 - Aggregate BW $\geq$ 4GB/s
 - RT latency $\leq 8.5\mu s$



本授業・演習の実情，問題点

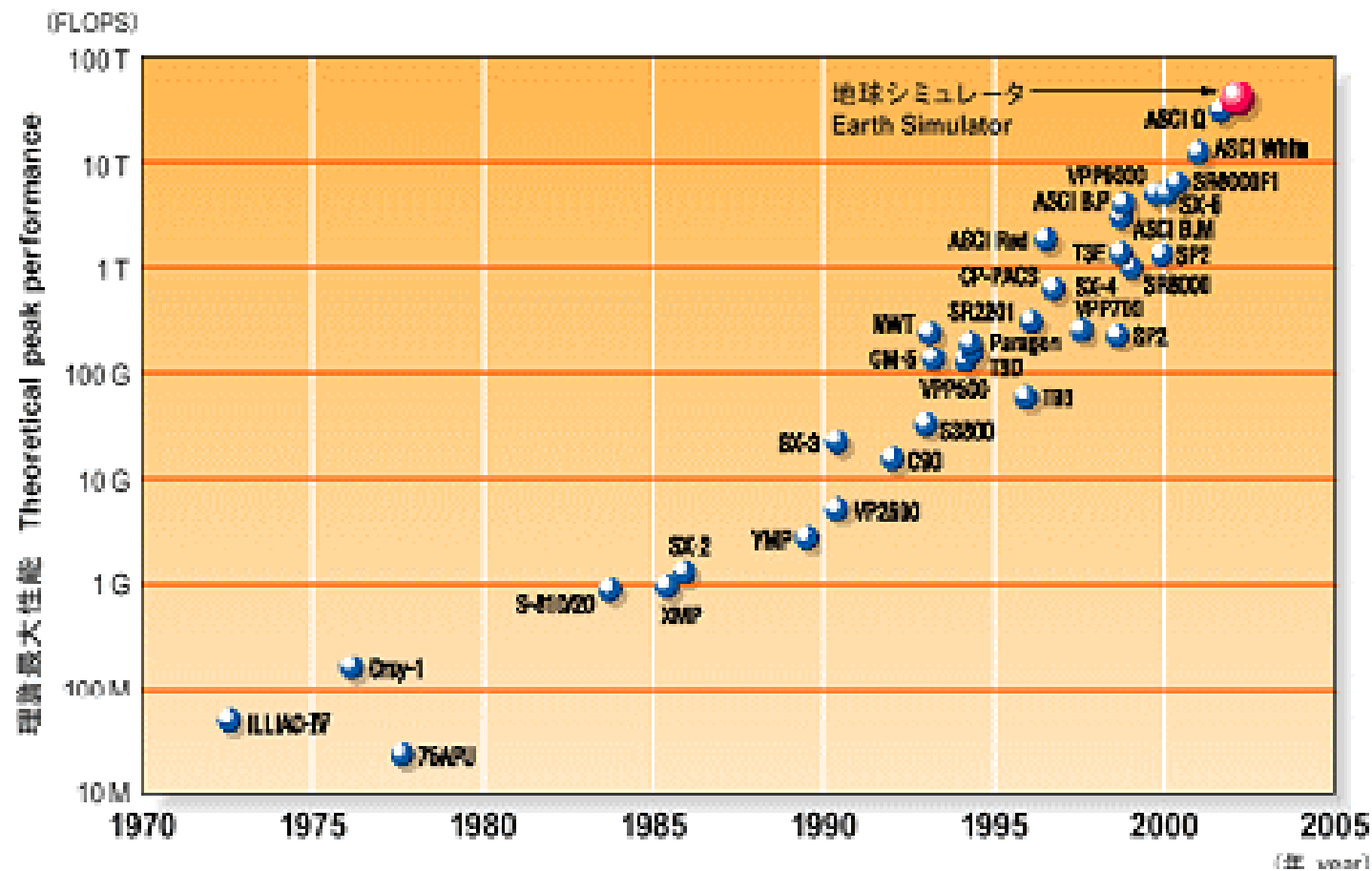
- バックグラウンド
 - 数値解析
 - シミュレーション
 - 専攻
 - 各自の興味
 - 感想文を書いてもらおうと色々な意見がある

- 本授業・演習の概要
 - 目的, 方針
 - 概要
 - スケジュール
- HPCの概要
 - 計算機ハードウェアの発達
 - 並列プログラミング言語

計算機ハードウェアの歴史

- プロセッサは1.5年に2倍の割合で処理速度が増加
 - Moore's Law: 集積度が18ヶ月から24ヶ月で2倍
- 並列計算機の発達
- 1983年: 1 GFLOPS, 1996年: 1 TFLOPS, 2002年: 36 TFLOPS, 2005年: 280TFLOPS, 2008年: 1,000TFLOPS (1PFLOPS)
 - MFLOPS: Millions of Floating Point Operations per Second. (1秒間に 10^6 回の浮動小数点処理)
 - GFLOPS: 10^9 回, TFLOPS: 10^{12} 回, PFLOPS: 10^{15} 回

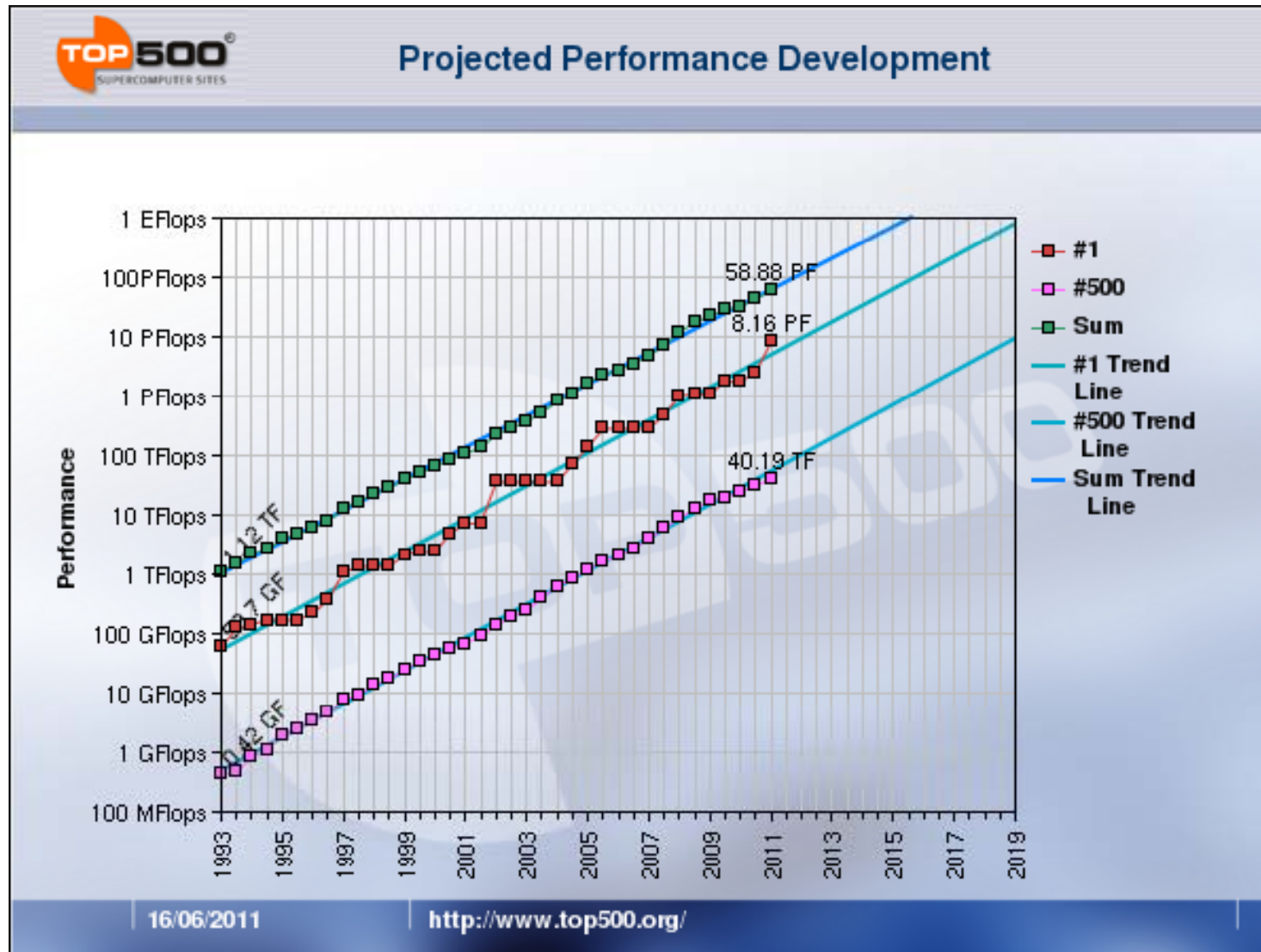
計算機ハードウェア発達の歴史



TOP 500 List

<http://www.top500.org/>

- 年2回更新
- LINPACKと言われるベンチマークテストを実施する。
 - 密行列を係数とする連立一次方程式を解く
 - ベクトル機でもスカラー機でも性能が出やすい
- 実際のアプリケーションではこれほどの性能は出ない
 - 差分法, スペクトル法系の手法: ピーク性能の60%程度
 - AFES on the Earth Simulator (Old): 26 TFLOPS (ピーク性能の65%)
 - 有限要素法
 - GeoFEM on the Earth Simulator (512ノード) (Old): 10 TFLOPS (30%)
 - スカラー機ではこれほど出ない: 5%~10%
- 消費電力に関するGreen500というランキングもある



- PFLOPS: Peta ($=10^{15}$) Floating OPerations per Sec.
- Exa-FLOPS ($=10^{18}$) will be attained in 2019

	Site	Computer/Year Vendor	Cores	$R_{\max}$	R_{peak}	Power
1	RIKEN Advanced Institute for Computational Science (AICS) Japan	K computer, SPARC64 VIIIfx 2.0GHz, Tofu interconnect / 2011 Fujitsu	548352	8162.00	8773.63	9898.56
2	National Supercomputing Center in Tianjin China	Tianhe-1A - NUDT TH MPP, X5670 2.93Ghz 6C, NVIDIA GPU, FT-1000 8C / 2010 NUDT	186368	2566.00	4701.00	4040.00
3	DOE/SC/Oak Ridge National Laboratory United States	Jaguar - Cray XT5-HE Opteron 6-core 2.6 GHz / 2009 Cray Inc.	224162	1759.00	2331.00	6950.60
4	National Supercomputing Centre in Shenzhen (NSCS) China	Nebulae - Dawning TC3600 Blade, Intel X5650, NVidia Tesla C2050 GPU / 2010 Dawning	120640	1271.00	2984.30	2580.00
5	GSIC Center, Tokyo Institute of Technology Japan	TSUBAME 2.0 - HP ProLiant SL390s G7 Xeon 6C X5670, Nvidia GPU, Linux/Windows / 2010 NEC/HP	73278	1192.00	2287.63	1398.61
6	DOE/NNSA/LANL/SNL United States	Cielo - Cray XE6 8-core 2.4 GHz / 2011 Cray Inc.	142272	1110.00	1365.81	3980.00
7	NASA/Ames Research Center/NAS United States	Pleiades - SGI Altix ICE 8200EX/8400EX, Xeon HT QC 3.0/Xeon 5570/5670 2.93 Ghz, Infiniband / 2011 SGI	111104	1088.00	1315.33	4102.00
8	DOE/SC/LBNL/NERSC United States	Hopper - Cray XE6 12-core 2.1 GHz / 2010 Cray Inc.	153408	1054.00	1288.63	2910.00
9	Commissariat a l'Energie Atomique (CEA) France	Tera-100 - Bull bullx super-node S6010/S6030 / 2010 Bull SA	138368	1050.00	1254.55	4590.00
10	DOE/NNSA/LANL United States	Roadrunner - BladeCenter QS22/LS21 Cluster, PowerXCell 8i 3.2 Ghz / Opteron DC 1.8 GHz, Voltaire Infiniband / 2009 IBM	122400	1042.00	1375.78	2345.50

$R_{\max}$: 実効性能 (TFLOPS)

R_{peak} : ピーク性能 (TFLOPS)

<http://www.top500.org/>

次世代スーパーコンピュータ「京」

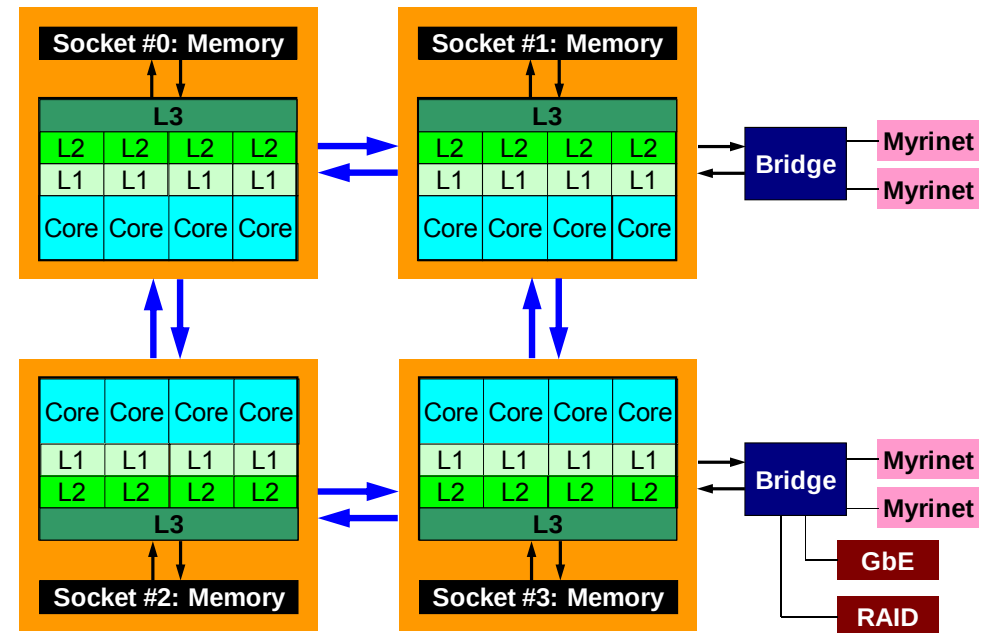
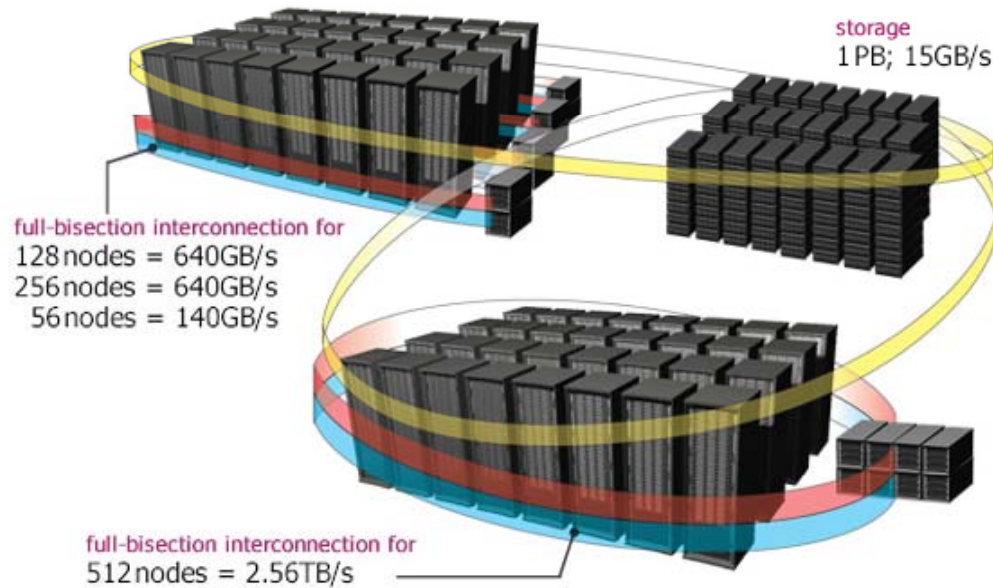
- 理化学研究所計算科学研究機構
(神戸)
 - <http://www.aics.riken.jp/>
- 京速計算機
 - 「京」=「兆」の10,000倍 = $10 \times 10^{15} = 10$ Peta FLOPS
- 2011年6月現在世界最速(未完成)



T2Kオープンスパコン(筑波, 東大, 京都)

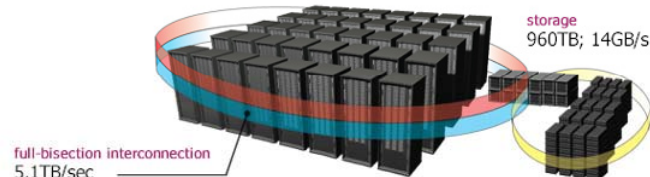
University of Tokyo

nodes = 952 Rpeak = 140.1TFlops Memory = 31TB



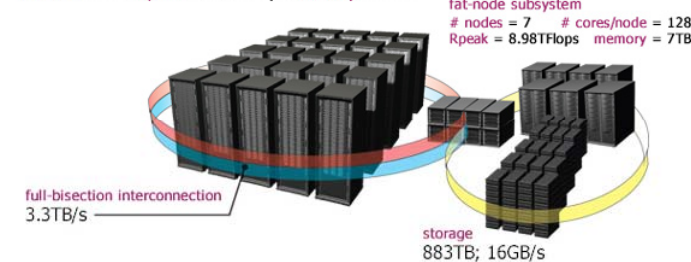
University of Tsukuba

nodes = 648 Rpeak = 95.4TFlops Memory = 20TB



Kyoto University

nodes = 416 Rpeak = 61.2TFlops Memory = 13TB



専用計算機

- GRAPE (GRAvity PipE)
 - 宇宙物理学におけるN対N問題用専用ハードウェア: MD, 境界要素法等
 - コストパフォーマンス
- MDGRAPE
 - MD (Molecular Dynamics) 専用のGRAPE
 - MDGRAPEを通常のクラスタのAcceleratorのように使用することもできる。

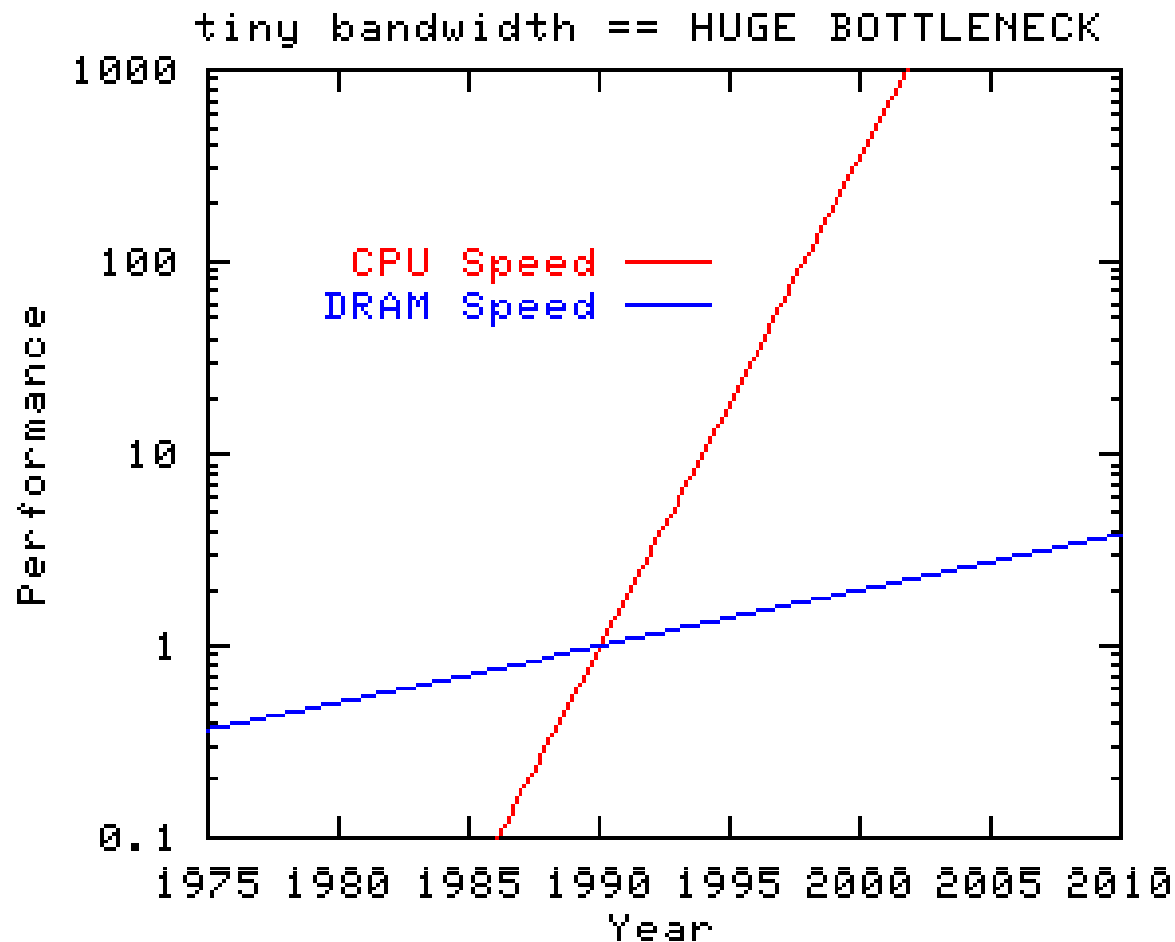
様々な技術的課題

- コアの性能→今後10年間でそれほど変化無し
- 電力消費量
 - Exascale(2019年頃)だと2GW(電気代2,000億円／年)
 - これを100MW未満に抑えるための技術革新必要
 - マルチコア化, メニーコア化
- メモリー性能の低下
 - GPGPU: 高いメモリバンド幅
 - 用途は限られる, プログラミングの困難さ
- 設置場所
- 故障
 - Exascale: 「億」規模のコア数



マイクロプロセッサの動向

CPU性能, メモリバンド幅のギャップ



<http://www.streambench.org/>

疎行列ソルバーの性能: 三次元弾性問題

ICCG法, T2K・SR11000 1ノード: メモリバンド幅が効く

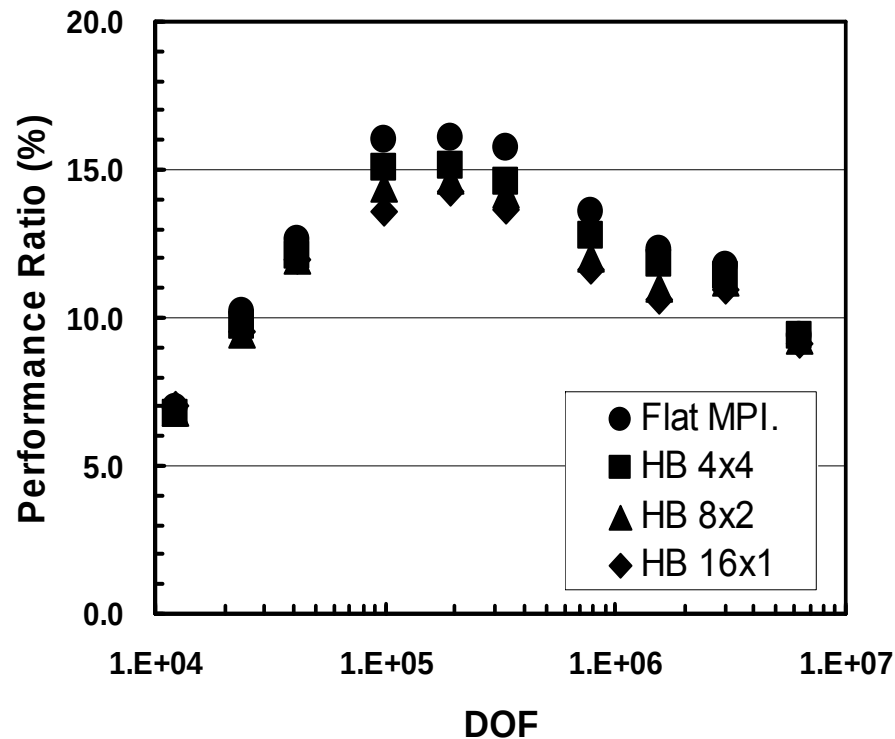
Hitachi SR11000/J2

Power 5+ 2.3GHz x 16

147.2 GFLOPS/node

100 GB/s for STREAM/Triad

L3 cache: 18MB/core



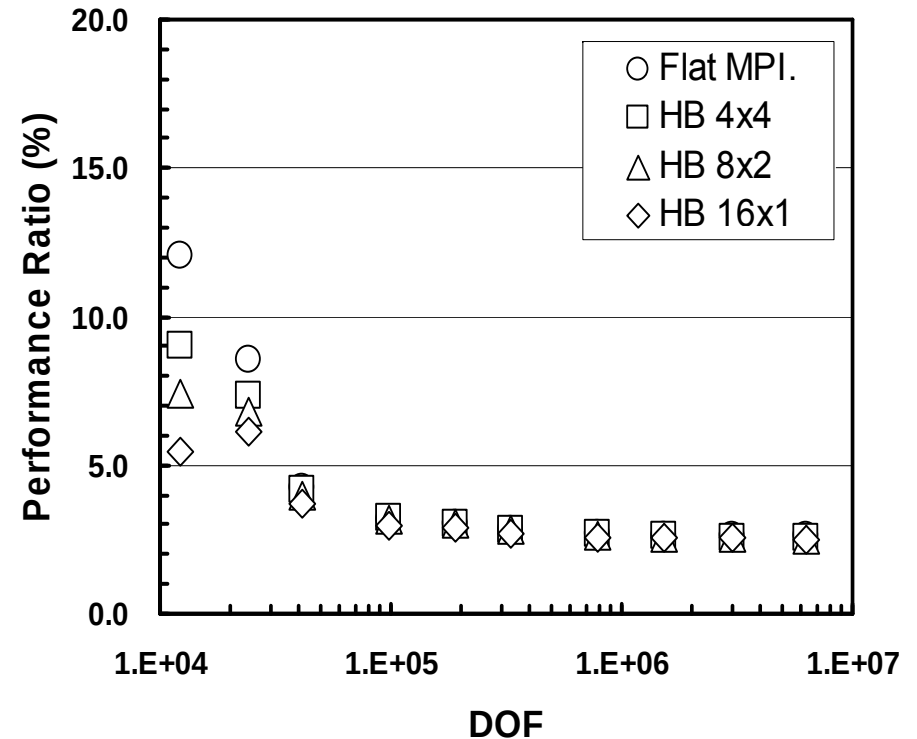
T2K/Tokyo

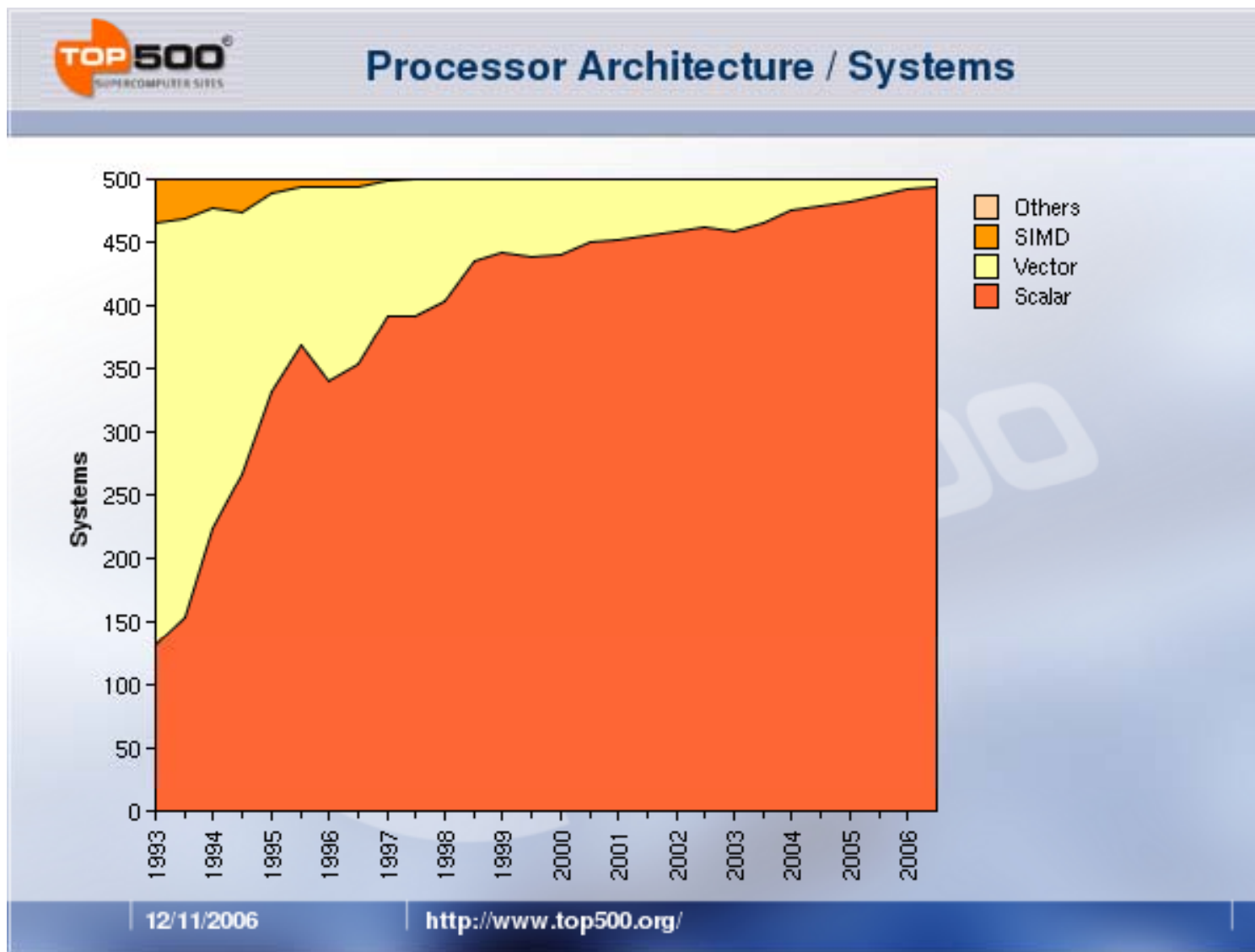
Opteron 2.3GHz x 16

147.2 GFLOPS/node

20 GB/s for STREAM/Triad

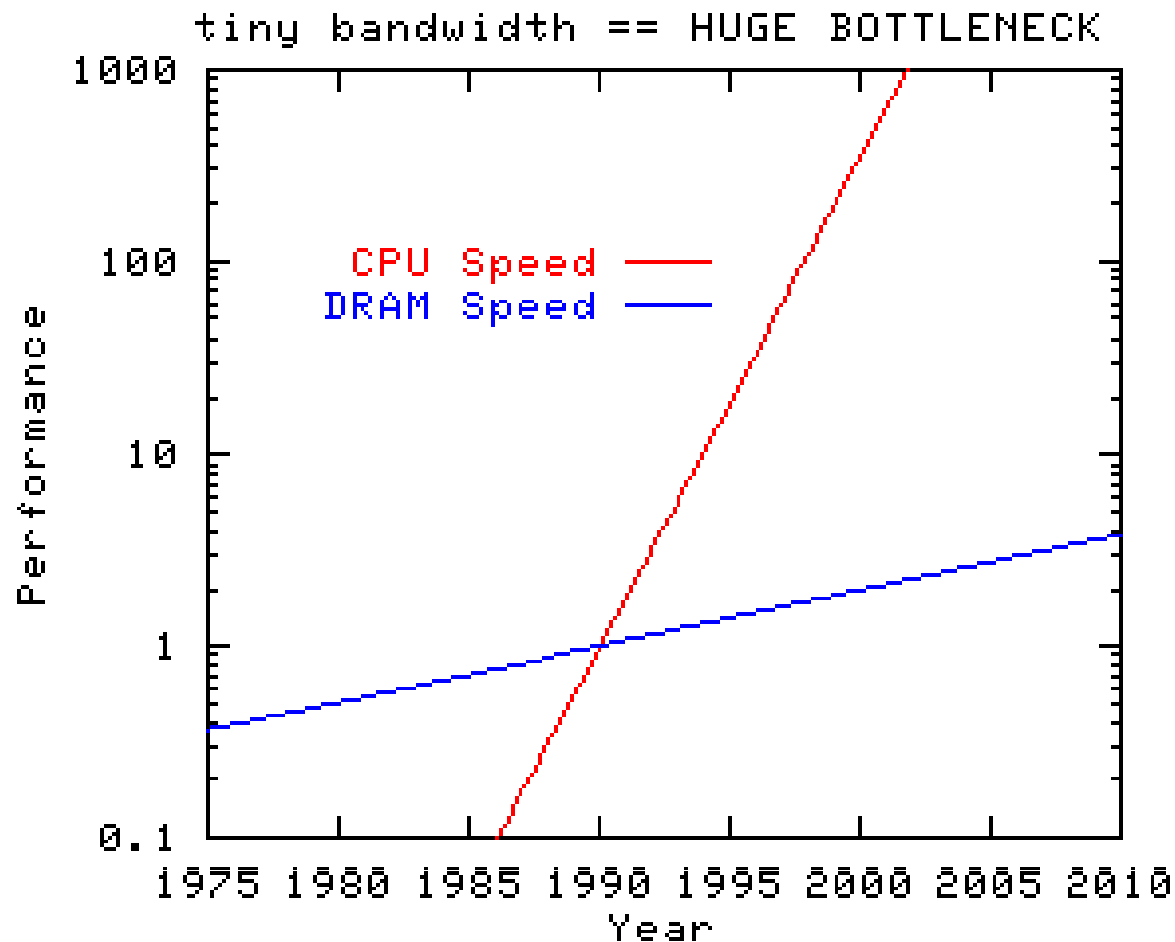
L3 cache: 0.5MB/core





マイクロプロセッサの動向

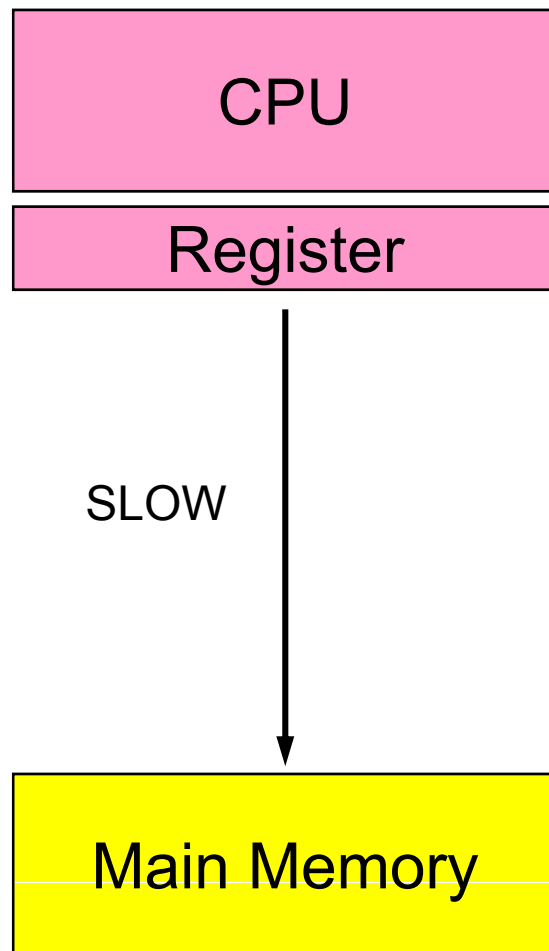
CPU性能, メモリバンド幅のギャップ

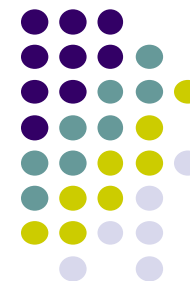


<http://www.streambench.org/>

スカラープロセッサ

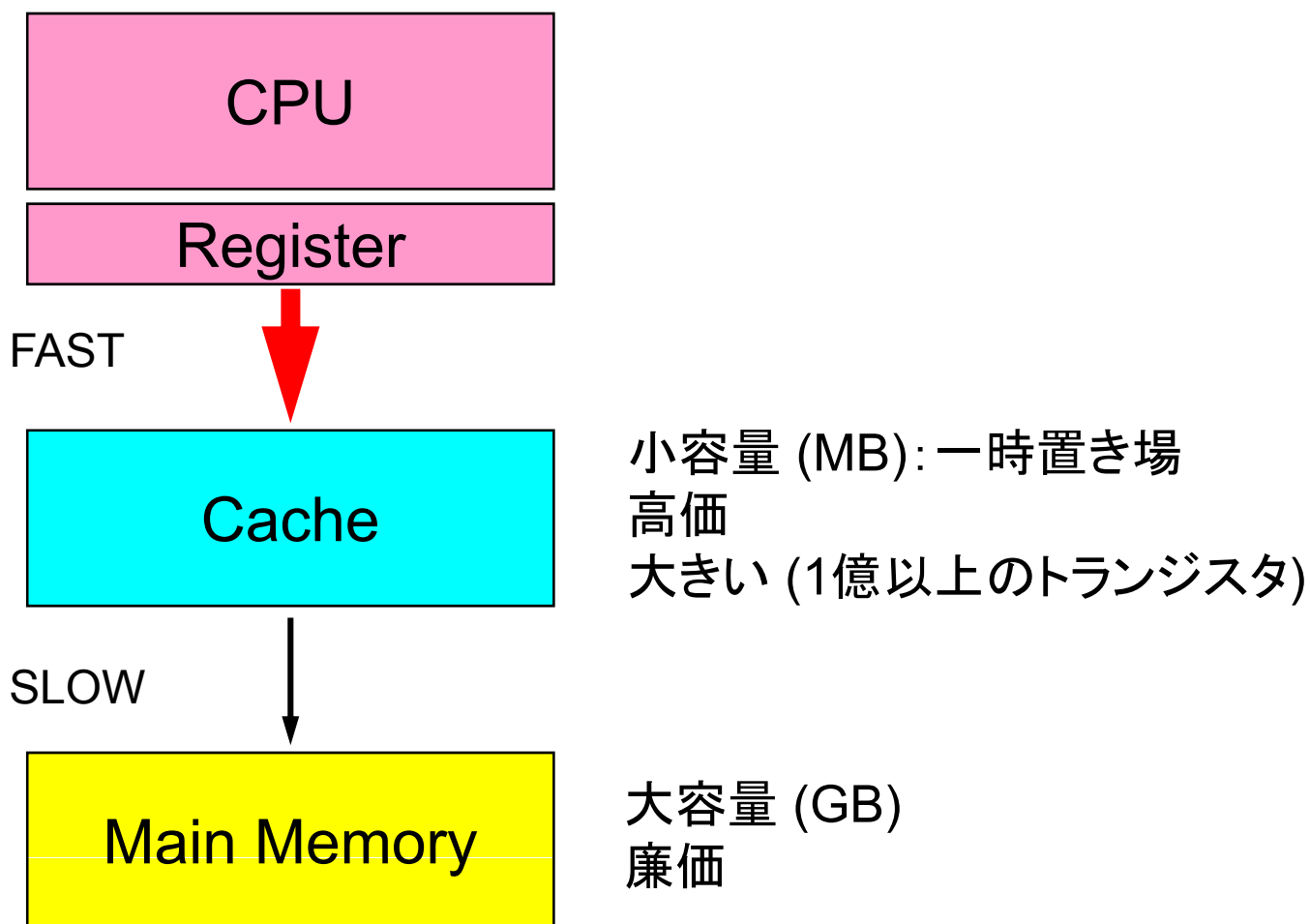
メモリへ直接アクセスするのは实际的でない





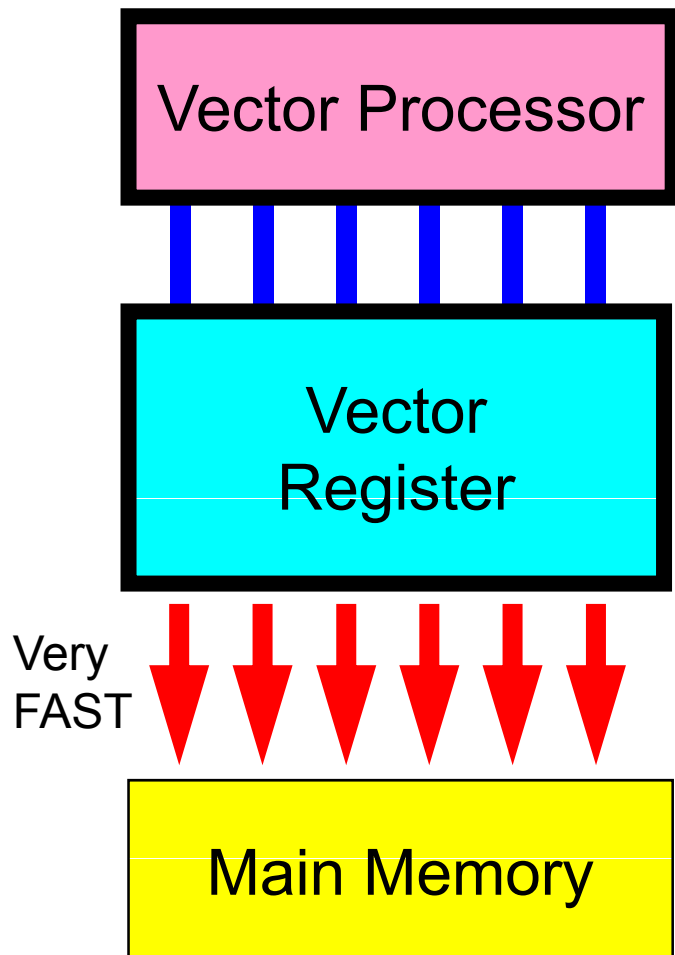
スカラープロセッサ

CPU-キャッシュ-メモリの階層構造



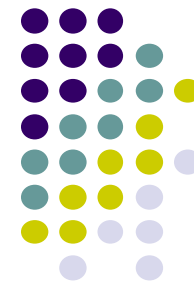
ベクトルプロセッサ

ベクトルレジスタと高速メモリ

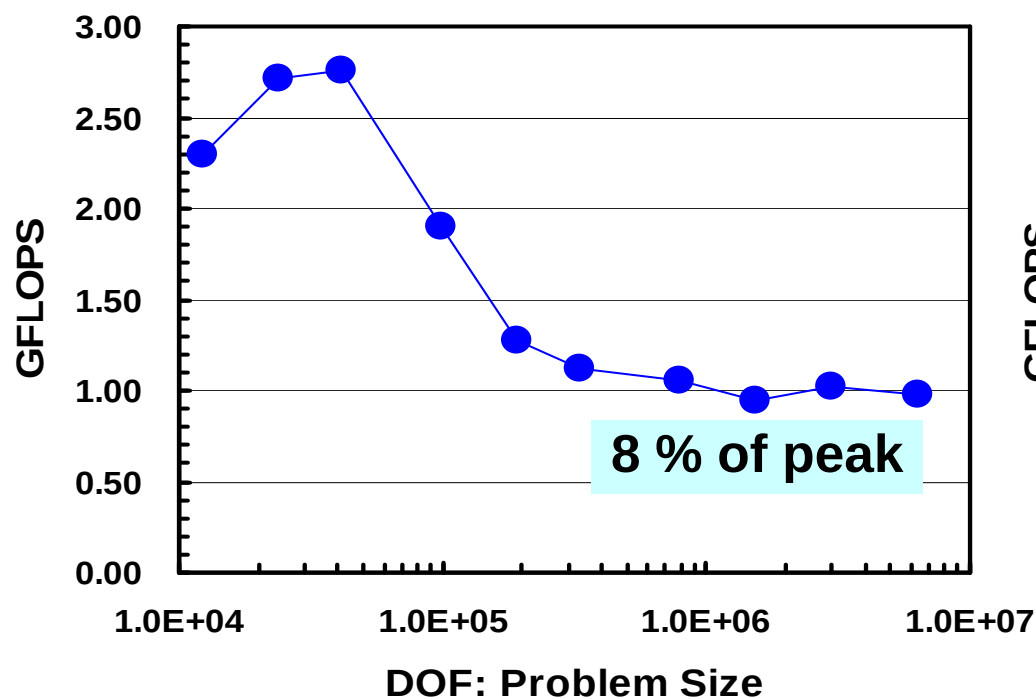


- 単純構造のDOループの並列処理
- 単純, 大規模な演算に適している

```
do i= 1, N  
  A(i)= B(i) + C(i)  
enddo
```

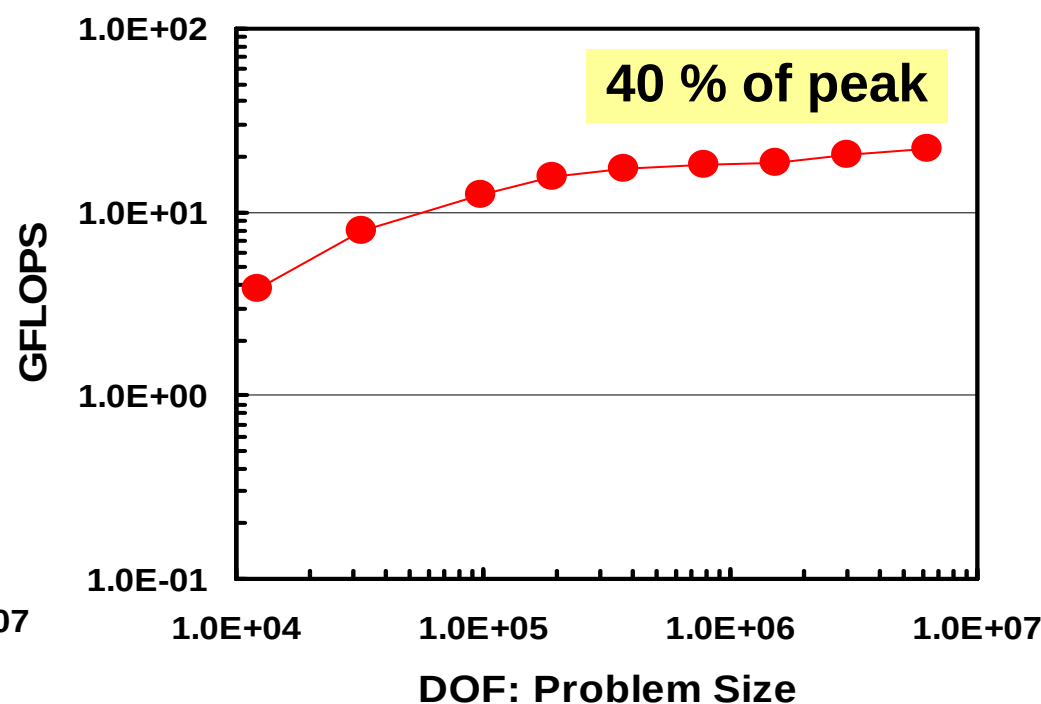


典型的な挙動:ICCG法



IBM-SP3:

問題サイズが小さい場合はキャッシュの影響のため性能が良い



Earth Simulator:

大規模な問題ほどベクトル長が長くなり、性能が高い