次世代スーパーコンピューティング・シンポジウム2008
2008年9月16-17日
MY PLAZAホール

次世代スーパーコンピュータと人材育成

平尾　公彦

東京大学
THE UNIVERSITY OF TOKYO

Scienceのしっかりした基盤が重要

21世紀は予測の科学の時代、計算科学はその基盤

数値実験による新しい概念形成こそイノベーション

異分野融合と人材育成が重要

若い人のフレッシュなアイデアをいかに生かすか

内容

- ◆21世紀の科学技術と人材育成
- ◆スーパーコンピュータでブレークスルーを
- ◆計算科学の人材育成への東京大学の試み
- ◆まとめ

20世紀の科学技術の発展

20世紀の科学技術は飛躍的な進歩をとげた。量子論と相対論という、より基本的な法則が誕生したためである。

新しい概念による自然科学の意識革命がなされ、それを実証してきた世紀である。素粒子論、物質論、生命論、宇宙論によって私たちの自然観は根底から覆された。

科学技術は「自由」を獲得する人間の知的活動、偶然や宿命などに支配されない自由の枠組みを拡大する活動である。

20世紀はどんな時代だったのか

20世紀の技術革新

1. 電力
2. 自動車
3. 航空機
4. 安全な水
5. 電子技術
6. ラジオ･テレビ
7. 農業機械化
8. コンピュータ
9. 電話 ･通信
10. 空調･冷凍
11. 高速道路
12. 宇宙船
13. インターネット
14. 画像技術
15. 家庭用電化製品
16. 医療技術
17. 石油･石油化学
18. レーザー･光ファイバー
19. 原子力技術
20. 高性能材料

アメリカ工学アカデミー2005

いずれもその源は基礎科学にある

今、私たちはどのような時代に生きているのか？

この**100**年、世界人口は**15**億から**66**億に、先進国の寿命は**40**歳から**80**歳へ。**CO2**濃度は**280ppm**から**379ppm (2005)** まで上昇、環境劣化と気候変動は待ったなしで地球温暖化へと向かう。南北格差は拡大し、人口の**20%**は極貧で、毎年**1,600**万人が餓死している。これが現実であり、この解決が**21**世紀の課題である。

社会のための科学と技術

真理の探求のための科学研究は今後も尽きることがないが、21世紀の科学は、知識のためだけでなく、社会、平和、開発のためにあらねばならない

人類社会の目標は、肥大化した人間圏を地球と共生しうる持続的なシステムとして再構築すること、未来と富を再配分すること、健康で快適な生活と安全で安心な社会を保証することにある

他方、地球規模で解決すべき環境や資源エネルギーなどの深刻化な問題を共有しつつ、文化的生活、福祉、倫理、個人の尊厳などをいかに担保するかが重要

わが国に期待されること

わが国独自の文化と感性に根ざした科学技術、省エネルギー技術などの技術移転で世界の環境にも経済にも貢献できる

今後、先進国各国で予想される人口減少、高齢化、環境、エネルギーの課題をわが国が率先して克服し、豊かな国家社会を造る模範を示すいい機会である

技術革新と女性、外国人を含めた優れた人材の育成が鍵
個人を基軸とする活力ある社会を創ることが目標

21世紀の人類社会の課題

環境問題
地球温暖化
南北問題
人口増加
エネルギー問題
水・食糧問題
高齢化社会
…

いずれも極めて複雑、多岐の分野にまたがる複合的な課題
相関関係を重視した俯瞰的学術の創成が必要

これまでの方法論で解決できるのだろうか？

計算科学

計算科学は実験、理論と並ぶ第三の科学であり、**21**世紀の予測の科学の基盤をつくる

今、求められているのは個々の複雑な現象を定量的に解明し、相互作用を理解しつつ、部分を組み合わせて持続可能なシステムとして全体を創り上げること

計算科学は理論、実験とは異なるコンピュータによる数値実験という新しい研究手法を実現し、新しい概念を提出し、科学にブレークスルーをもたらすものとして期待されている

計算科学で何ができるのか？

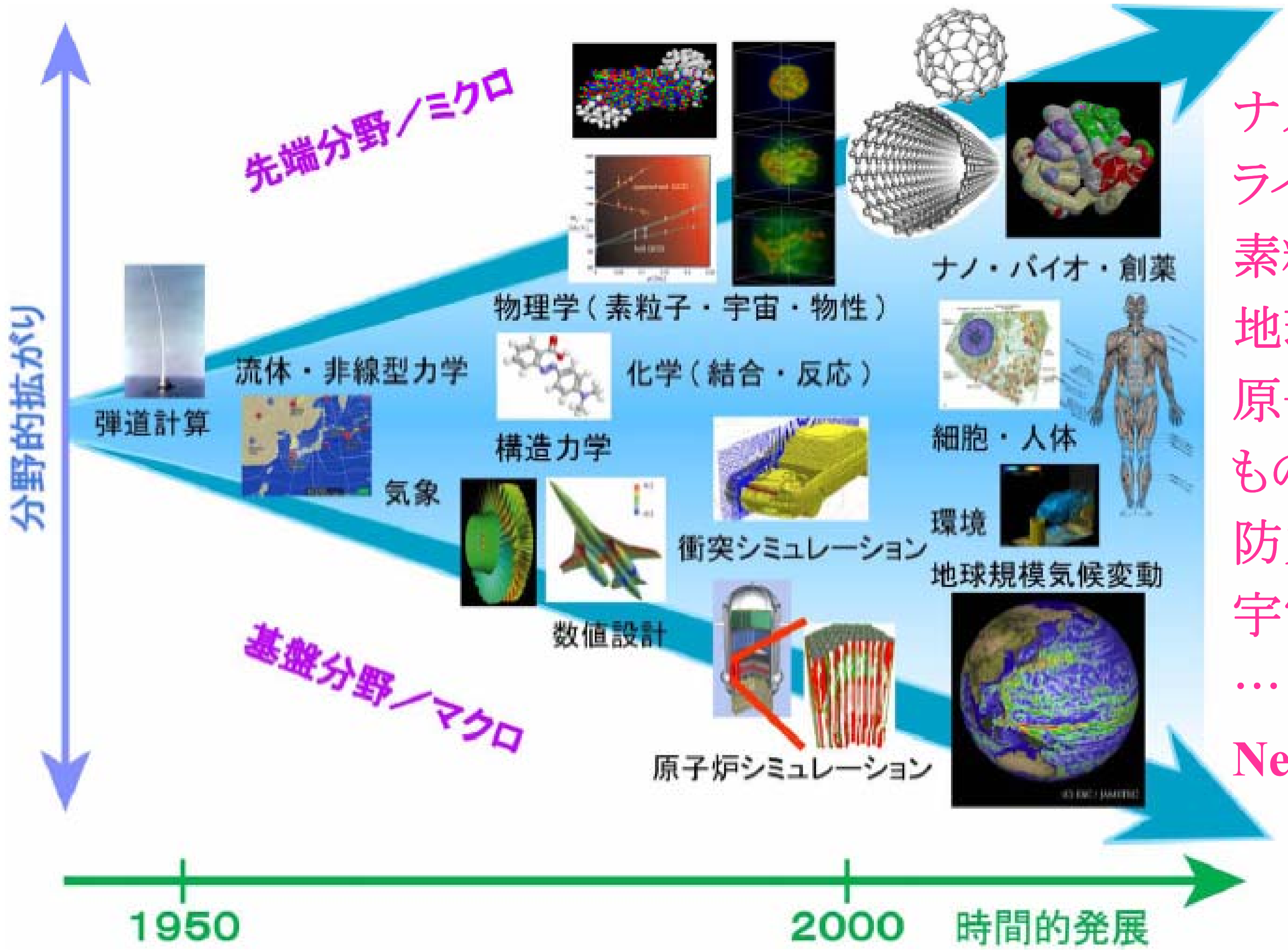
再現：　すでにわかっている事象を再構築して理解
（地震、津波などの自然災害）

最適化：　すでにわかっているシナリオを最適化
（工学的プロセスや産業プロセス）

予測：　未来または未知の状況を予測
（地球環境、物質設計、新現象の発見）

未来にわれわれが強い関心を示すのは、人間に予測能力がもともと欠けているところにその理由があるのではないか

計算科学の拡大



ナノテクノロジー
ライフサイエンス
素粒子・宇宙物理
地球・環境
原子力
ものづくり
防災・気象
宇宙・航空
…
New Science

近代科学の発展は自然現象を数学で表すことから始まった

デカルト、パスカル、フェルマー、ニュートン、ライプニッツ

微分積分学

偏微分方程式

拡散方程式、波動方程式、非線型現象

基本的な法則の認識に果たした数学の役割は大きい

複雑さを科学し、法則を見出す計算科学

複雑さを計算する多様性科学は、非線形工学をベースとした科学であり、科学技術の領域を広げる上で重要

計算科学はランダムな振る舞いやカオス的な振る舞いの裏にも数値計算で表せる法則があり、解析不可能と思われていた領域にも科学的にアプローチできる可能性を示唆

心理学や社会科学、経済学など実験や論証のしにくい領域にも数値実験による分析、切り口を提供

コンピュータはわれわれに欠けている能力を補ってくれる

コンピュータは数学の脆弱性を補う（4色問題）

コンピュータは測定不能、あるいは実現不能な境界領域の場の解析を行う（宇宙や深海、人体といった領域）

コンピュータは未来を予測する（気象予報、地球温暖化）

コンピュータはランダム、カオス的振る舞いの裏にある法則を見出す（複雑さを計算する多様性科学を確立）

理論、実験からでなく数値実験から新概念形成

将棋ソフト **Bonanza**

チェスでは**1996**年に**IBM**の**Deep Blue**が**Garry Kasparov**に勝利して以来、次第に人間がコンピュータに勝つのは難しくなっている

将棋は？
Bonanza(2005年理論物理学者保木邦仁さんが開発、保木さんは将棋についてはほとんど知らないという)

2007年**3**月**21**日渡辺竜王と対戦、結果は**112**手で渡辺竜王の勝ちとなったが、渡辺竜王は、「人間では発想できない良手を指した意外な強さ」を指摘

人類は大型コンピュータを駆使する数値実験によって複雑な自然現象や社会現象を認識・予測する新しい科学的アプローチを手に入れたといえる

今こそ、計算科学の振興と人材育成が重要

国家プロジェクトとしての次世代スーパーコンピュータ開発の目的

科学技術のブレークスルーの実現

計算科学の革新と人材育成

グランドチャレンジ

高い目標を掲げ、妥協のないプロジェクト実施すべき

- 明確な科学技術上の目標設定
- 原点に立ち戻ったモデル化とアルゴリズムを
- ソフトウエアもスクラッチから創ることを覚悟

計算科学の革新は、妥協のない努力の積み重ねと分野融合の促進ではじめて実現しうる

次世代スーパーコンピュータへの期待

次世代スパコンが、顕微鏡の世界ではつまらない

- 見えないものが大切、次世代スパコンでしか到達できない世界が重要

コンピュータが発達したおかげで可能になる科学を**encourage**すべき

- 数学、非線形科学、カオス的振る舞いの裏にある法則

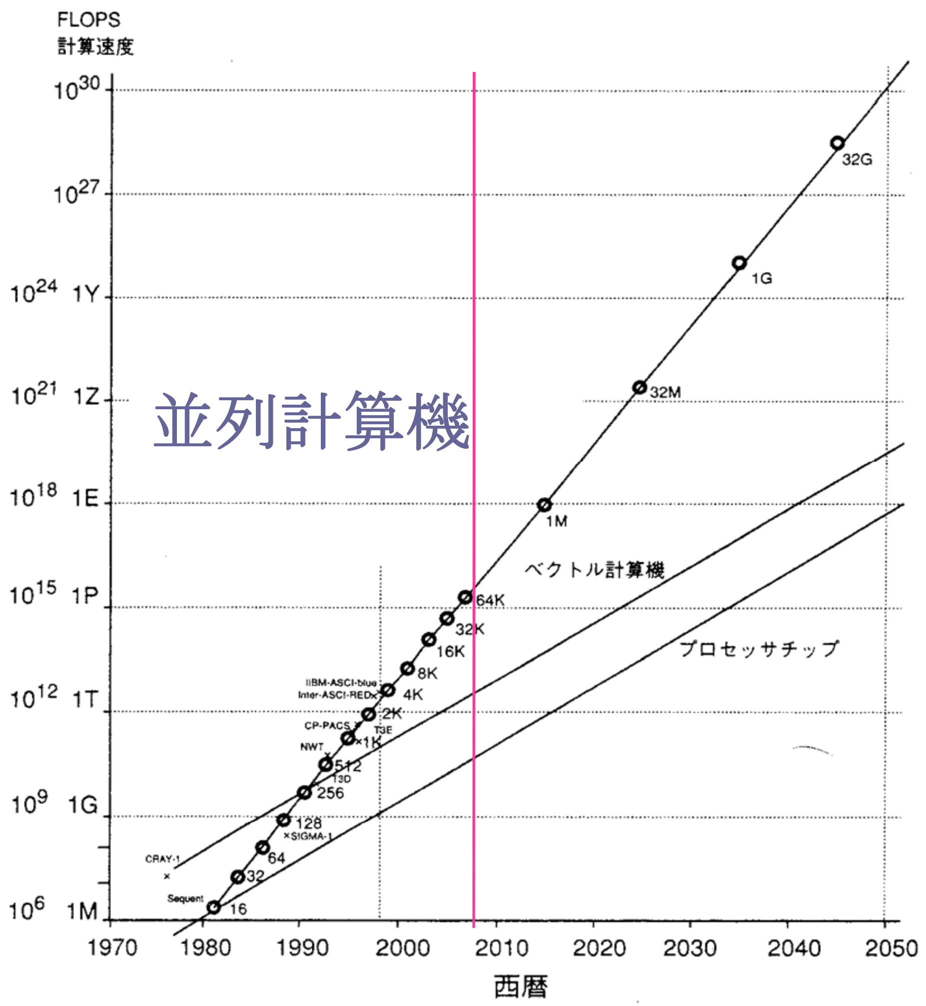
科学の発展は不連続、ブレークスルーは現在の延長にはない

- 質的に新しい可能性を誘起する研究システム、人材育成システムと環境が必要

次世代スーパーコンピュータは超並列計算機であること

計算科学は極めて学際的であること

コンピュータの発展



Cray-1 0.16 GFLOPS
地球シミュレータ 45TFLOPS
5,120CPUs
Blue Gene/L 478TFLOPS
212,992CPUs
T2Kオープンスパコン東大
140TFLOPS
15,232CPUs
IBM Roadrunner 1PFLOPS
12,960 IBM PowerXCell8i CPUs
and 6,480 AMD Opteron dual-core
processors
Blue Waters 2PFLOPS

次世代スーパーコンピュータ
10PFLOPS
Massively Parallel

超並列計算機

コンピュータ利用環境の健全な配備

重層的配備の重要性

10Petaスーパーコンピュータ

T2K

…

研究室の**PC**クラスター

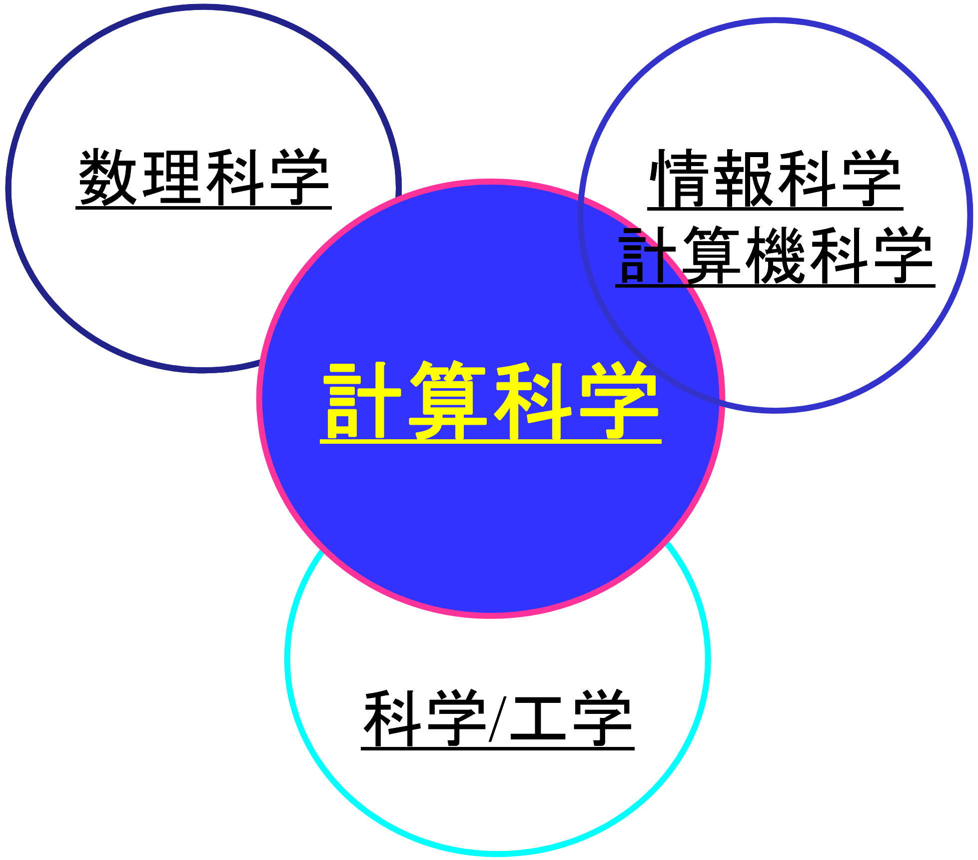
…

- 最高速システムとともに複数の中規模システムを配備
- 単一分野集中ではなく、計算科学の多様な分野を支える
- 重点分野は科学技術の発展とともに変遷
- 今日の発展途上分野から明日の重点分野の発展を促す

計算科学全体の持続的かつ厚みのある発展

計算科学は学際的

Identity Crisis



関係は双方向、計算科学の発展がそれぞれの分野の変革を促す

Scientific Computing = SMASH

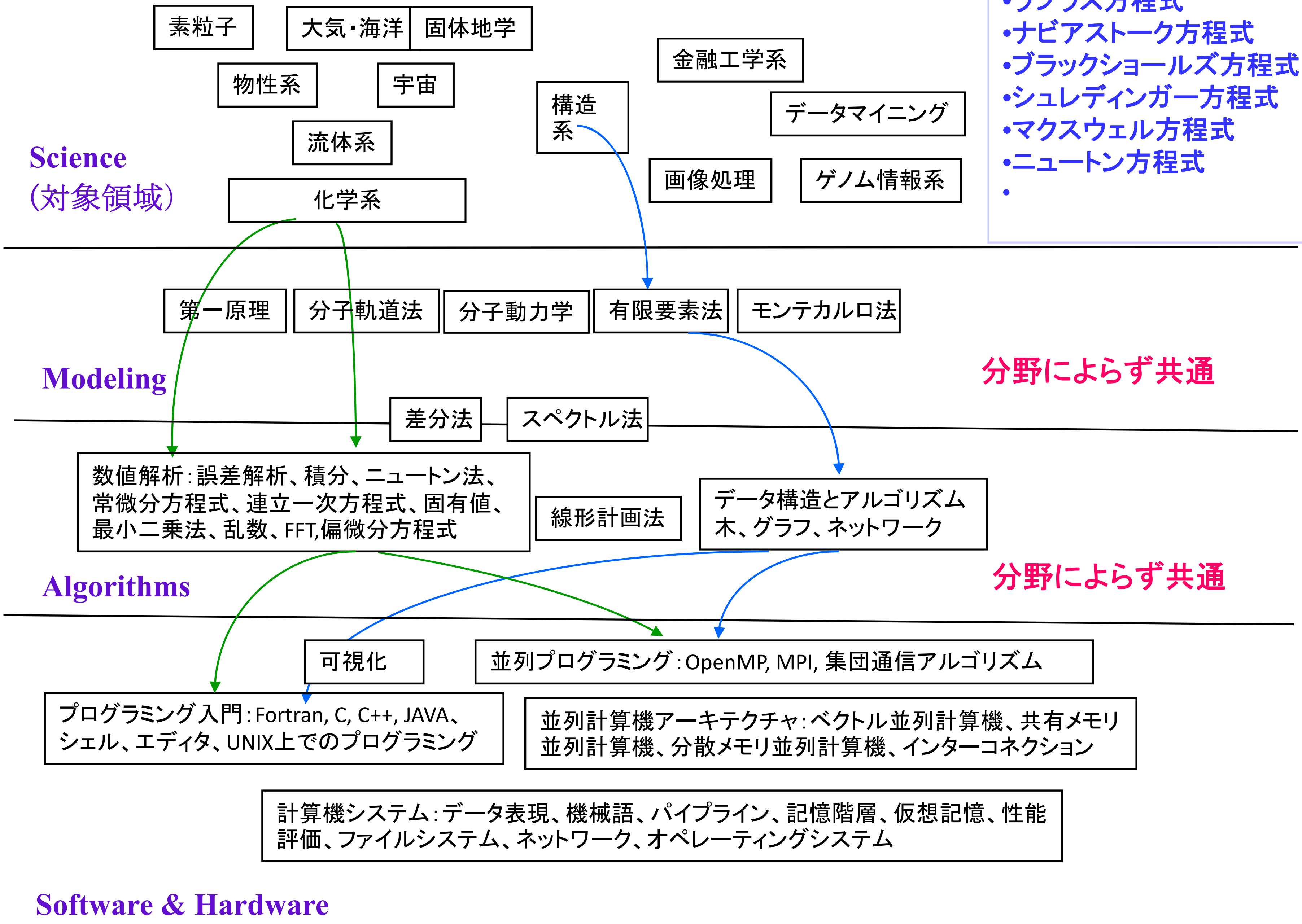
David Levermore (Univ. of Maryland)

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
- 超並列計算機を使いこなさねばならない

計算科学全体像

- ポアソン方程式
- ラプラス方程式
- ナビアストーク方程式
- ブラックショールズ方程式
- シュレディンガー方程式
- マクスウェル方程式
- ニュートン方程式
-



わたしの専門分野
量子化学ー分子の理論

Science
Modeling
Algorithm
Software
Hardware

物質科学:Multi-Scale Hierarchical Approach

Electrons　Atoms　Molecules　Nanoparticles　Devices

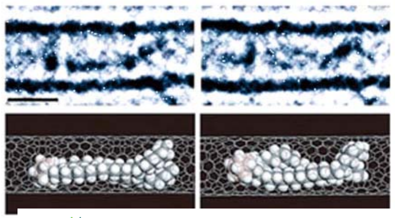
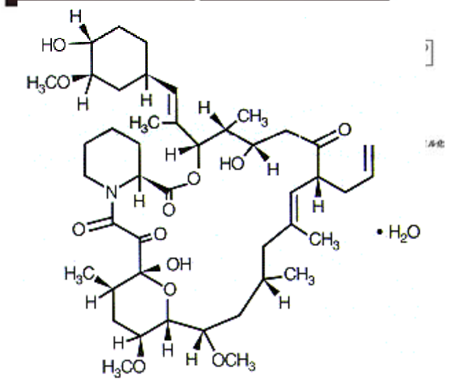
原子分子

ナノ・材料・生命

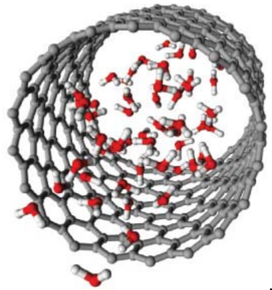
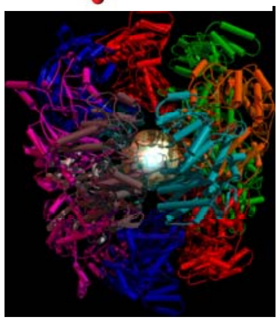
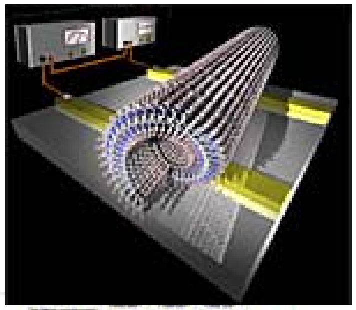
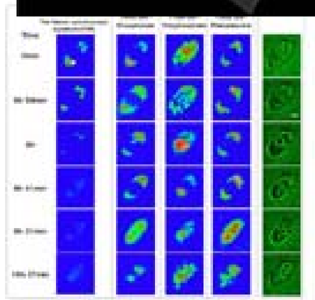
環境・エネルギー
健康・安全

1Å　1nm　1μm　1m

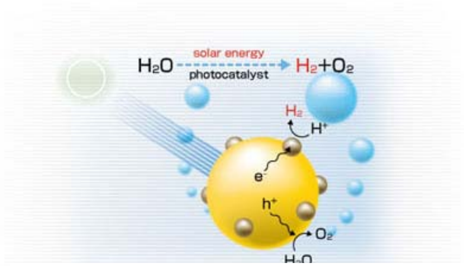
Femto-Picoseconds　Pico-Nano-Microseconds　Seconds-Minutes

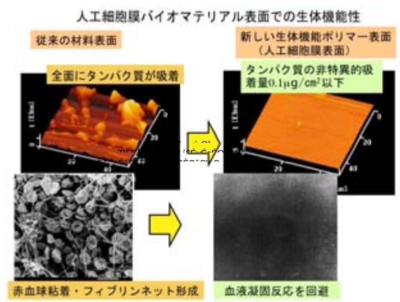



分子設計・反応設計
量子化学

超分子設計・生物化学
分子動力学

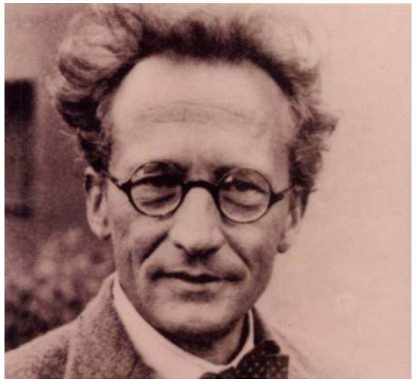




元素戦略・グリーン
エネルギー・医療
連続体モデル

原子分子の基礎方程式

N個の原子核と***n***個の電子から成る系の**Schrödinger**方程式



$$H\Psi = E\Psi$$

$$H = -\sum_{\alpha}^{N} \frac{\nabla_\alpha^2}{2M} - \sum_{i}^{n} \frac{\nabla_i^2}{2} - \sum_{\alpha,i} \frac{Z_\alpha}{r_{\alpha i}} + \sum_{i>j} \frac{1}{r_{ij}} + \sum_{\alpha,\beta} \frac{Z_\alpha Z_\beta}{R_{\alpha\beta}}$$

$$\Psi \equiv \Psi(r_1, r_2, \cdots, r_n; R_1, R_2, \cdots, R_N)$$

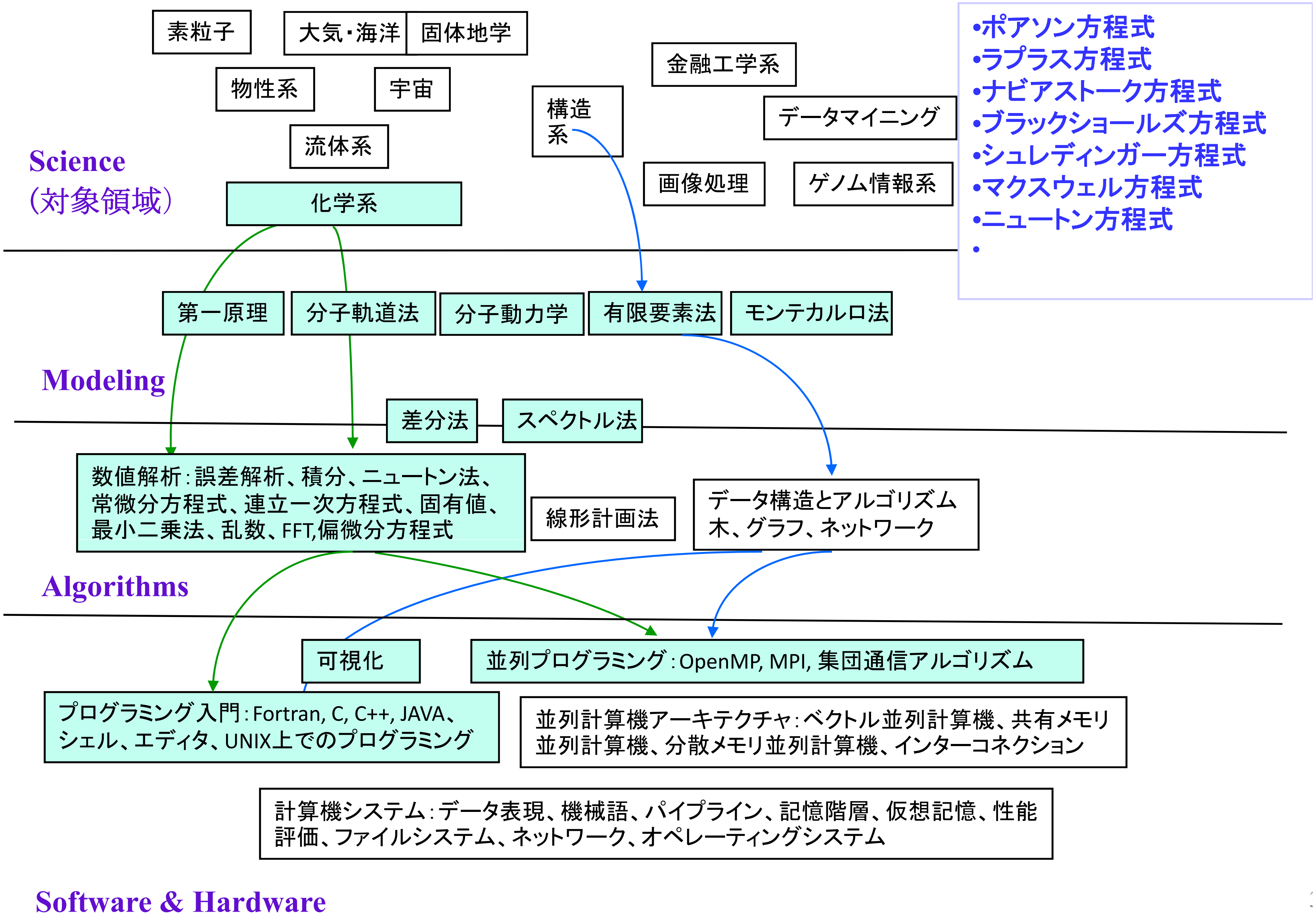
2階の微分方程式を解析的展開法を利用して、行列式の固有値問題という代数の問題に帰着させて近似的に解く。波動関数、エネルギーは電子の軌道関数の汎関数

$$\Psi \equiv \Psi(\{\varphi_i\}), \qquad E \equiv E(\{\varphi_i\})$$

軌道関数も原子軌道で展開

$$\varphi_i = \sum_{p}^{m} \chi_p C_{pi}$$

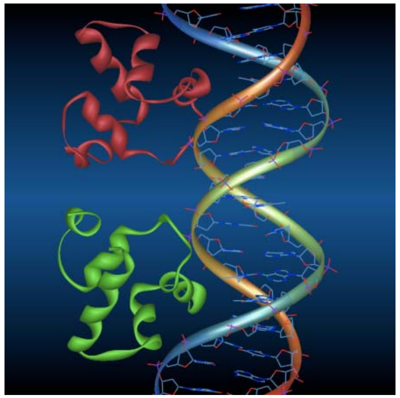
分子・分子集合体の計算科学

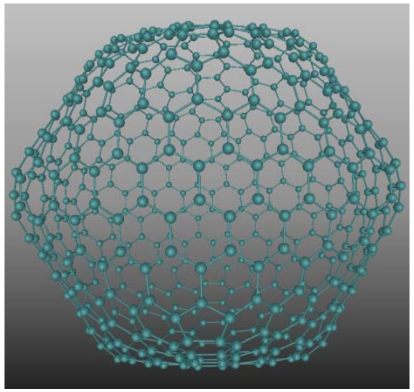


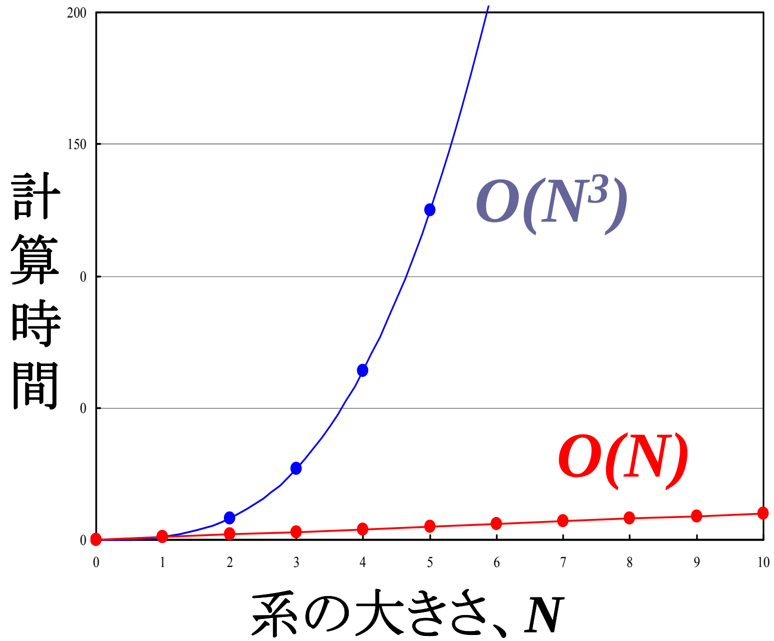
分子計算の悩み

系の大きさ(N)依存性が極めて大きい

計算時間は$N^{3.0}$で増加







N依存性を下げる理論(Linear Scaling)が求められている！

ネックはCoulomb積分の計算

Coulomb積分

$$\left(ab\middle|cd\right)=\iint \chi_a(r_1)\chi_b(r_1)\frac{1}{r_{12}}\chi_c(r_2)\chi_d(r_2)dr_1dr_2$$

通常は電子密度をガウス型基底関数で展開し、解析的に求積
(高速多重極展開法(**FMM**)や**RI**法、**Cholesky**分解法との併用)

基底関数の数をNとすると**Coulomb**積分の数はN^4で増加
Nが大きくなると、あっという間に積分の数は莫大になる

大規模系には不向きなアルゴリズム

Coulombポテンシャルを数値的に求める

Coulombポテンシャル***V*****(*r*)**は

$$V(r)=\int_{-\infty}^{\infty}\frac{1}{r_{12}}\rho(r_2)dr_2=\frac{2}{\sqrt{\pi}}\int_0^{\infty}\int_{-\infty}^{\infty}e^{-t^2(r_1-r_2)^2}\rho(r_2)dr_2dt$$

(1) ポアソン方程式を解くことで**V(r)**を数値的に求める

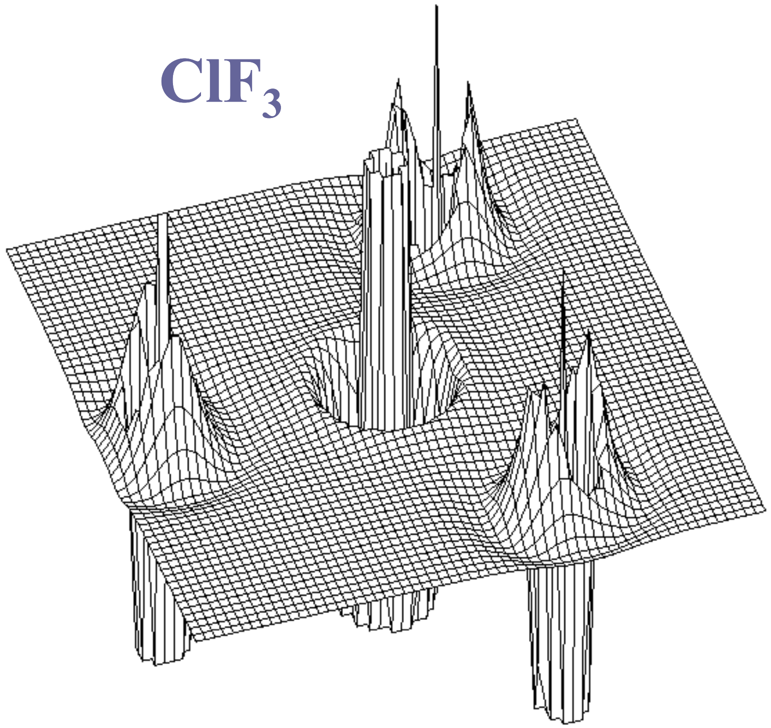
(2) 積分を離散問題に近似して**V(r)**を数値的に求める

Coulombポテンシャル***V*****(r)**が得られれば**Coulomb**積分は

$$E_J=\int\rho(r)V(r)dr=\iint\frac{\rho(r_1)\rho(r_2)}{r_{12}}dr_1dr_2$$

分子系のCoulombポテンシャルの特徴

- 原子核上では鋭くスパイク状に立っている
- **Coulomb**力は長距離力であり、原子核から遠く離れたところでも減衰せずに残っている



性格の異なる**2**つの領域をどうやってうまく表現するか？

計算科学というより、**Science**の問題

Gaussian & Finite-Element Coulomb (GFC)法

ポアソン方程式を解いてV(r)を数値的に求める

CoulombポテンシャルをGauss型と有限要素の混合基底を用いて展開する

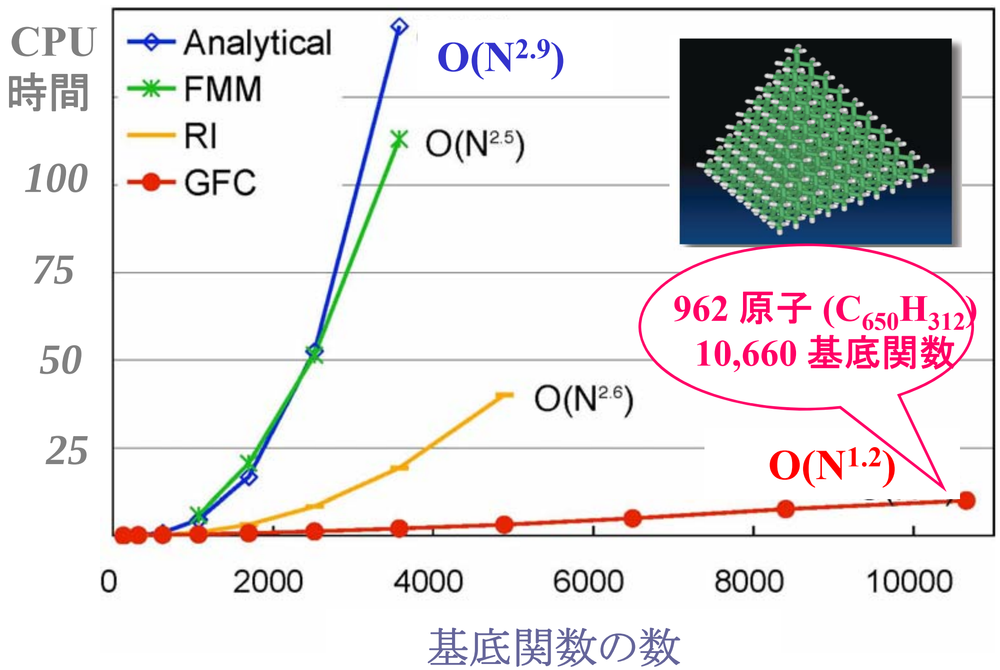
$$V(r) = \sum_i c_i^{FE} \xi_i^{FE}(r) + \sum_i c_i^{Gauss} \xi_i^{Gauss}(r)$$

Coulombポテンシャルはポアソン方程式を解いて得られる

$$\nabla^2 V(r) = -4\pi\rho(r)$$

Linear Scaling

3次元ダイヤモンド



スペクトル・エレメント法

積分を離散問題に近似してV(r)を数値的に求める

$$\rho(\mathbf{r}) = \sum_{lmn}^{N} d_{lmn} T_l(x) T_m(y) T_n(z)$$

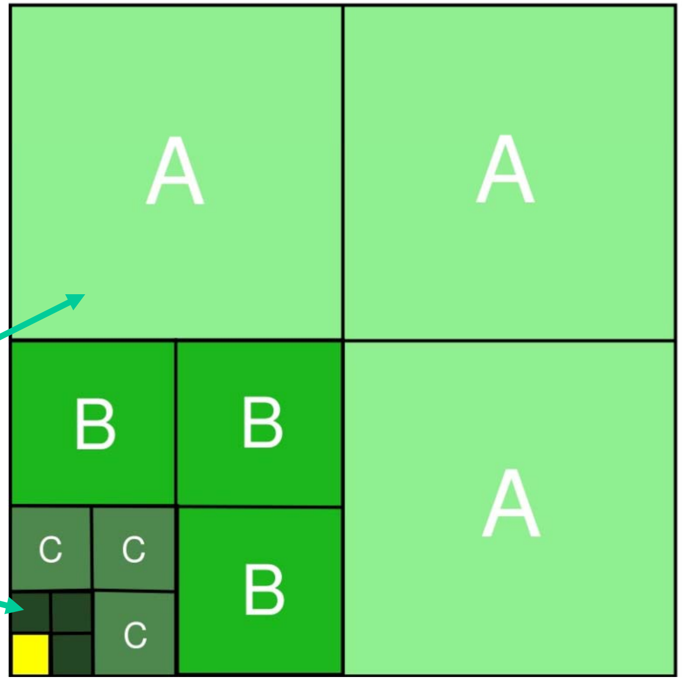
電子密度をChebyshev 多項式で展開

$$\frac{1}{\mathbf{r}_{12}} = \sum_{p}^{N_t} w_p e^{-t_p^2 (\mathbf{r}_1 - \mathbf{r}_2)^2}$$

Coulomb演算子の離散化

$$v_{uvw} = \sum_{p}^{N_t} w_p \sum_{l}^{N_x} F_{ul}^{xp} \sum_{m}^{N_y} F_{vm}^{yp} \sum_{n}^{N_z} F_{wn}^{zp} d_{lmn}$$

$$F_{wn}^{zp} = \int_{-1}^{1} T_n(z) e^{-t_p^2 (z_w - z)^2} \mathrm{dz}$$

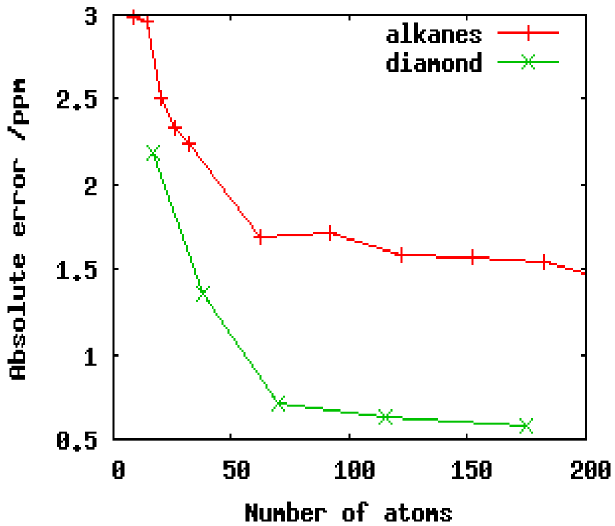


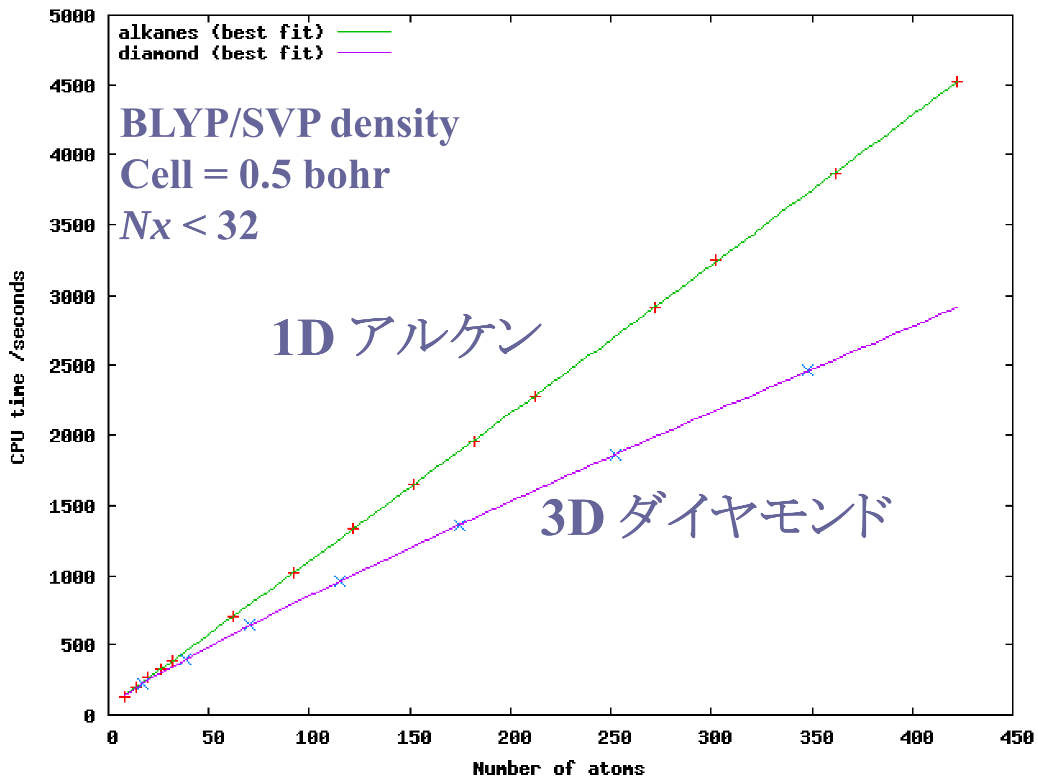
高速多重極展開法(FMM)

Chebyshev 多項式を利用した求積

境界値問題、大次元連立1次方程式が不用
並列計算機に適したアルゴリズム

スペクトル・エレメント法





Low scaling:
alkanes ~ $O(N^{1.01})$
diamond ~ $O(N^{0.89})$

Linear Scaling

10PFLOPS, 1万原子系のDynamicsが可能

コンピュータの使い方以前の問題として、
Scienceや数学をしっかり教育する必要がある、
これは大学の仕事

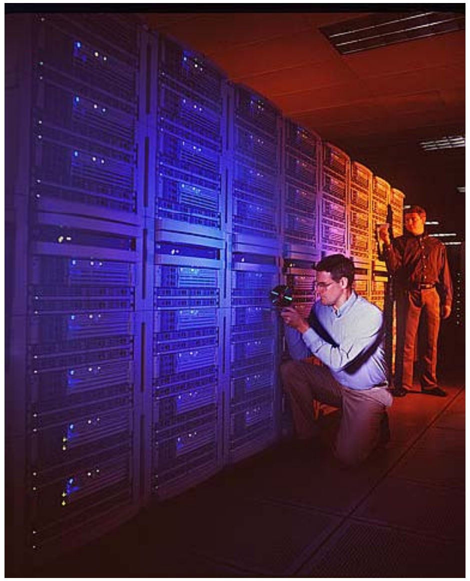
革新は「知」から

波動関数理論のための人工知能

若い人のフレッシュなアイデア

コンピュータに式を導出させ、並列プログラムを作成させよう

- 与えられた任意の波動関数法の式導出と高性能並列プログラム生成を完全自動化する
- ハードウェアの高速化・ソフトウェア技術の進歩
  - 算術演算
  - より高度な論理演算・人工知能



結合クラスター法
Coupled Cluster (CC)法

量子化学計算でもっとも高精度で系統的な多体理論

CCD, CCSD, CCSDT, CCSDTQ, …

$$\Psi = \exp(T)\Phi_0$$

$$T = T_1 + T_2 + \cdots = \quad T_1 + T_2 + \frac{T_1^2}{2!} + T_3 + T_1 T_2 + T_4 + \frac{T_2^2}{2!} + \cdots$$

1950年代に核物理の分野で**F. Coester**と**H. Kümmel** によって提唱され、**60**年代に**J. Čížek**と**J. Paldus**によって量子化学分野に導入されて大きく発展。最近、核物理の分野に逆輸入

なぜ自動化するのか？
Coupled-Cluster Singles & Doubles で既にこれだけ複雑

$$E=\langle\Phi_0|[e^{-T_1-T_2}He^{T_1+T_2}]_C|\Phi_0\rangle;\quad 0=\langle\Phi_i^a|[e^{-T_1-T_2}He^{T_1+T_2}]_C|\Phi_0\rangle;\quad 0=\langle\Phi_{ij}^{ab}|[e^{-T_1-T_2}He^{T_1+T_2}]_C|\Phi_0\rangle$$

```
E = [ + 1.0 ] * Sum ( h1 p2 ) * f ( h1 p2 ) * t ( p2 h1 )+[ + 0.25 ] * Sum ( h3 h4 p1 p2 ) * t ( p1 p2 h3 h4 ) * v ( h3 h4 p1 p2 )+[ + 0.5 ] * Sum ( h2 h4 p1 p3 ) * t ( p1 h2 ) * t ( p3
        h4 ) * v ( h2 h4 p1 p3 )

0 = [ + 1.0 ] * f ( p2 h1 )+[ - 1.0 ] * Sum ( h3 ) * f ( h3 h1 ) * t ( p2 h3 )+[ + 1.0 ] * Sum ( p3 ) * f ( p2 p3 ) * t ( p3 h1 )
+[ - 1.0 ] * Sum ( h4 p3 ) * t ( p3 h4 ) * v ( h4 p2 h1 p3 )+[ + 1.0 ] * Sum ( h3 p4 ) * f ( h3 p4 ) * t ( p4 p2 h3 h1 )
+[ + 0.5 ] * Sum ( h4 h5 p3 ) * t ( p3 p2 h4 h5 ) * v ( h4 h5 h1 p3 )+[ + 0.5 ] * Sum ( h5 p3 p4 ) * t ( p3 p4 h5 h1 ) * v ( h5 p2 p3 p4 )
+[ - 1.0 ] * Sum ( h4 p3 ) * t ( p3 h1 ) * t ( p2 h4 ) * f ( h4 p3 )+[ - 1.0 ] * Sum ( h3 h5 p4 ) * t ( p2 h3 ) * t ( p4 h5 ) * v ( h3 h5 h1 p4 )
+[ - 1.0 ] * Sum ( h5 p3 p4 ) * t ( p3 h1 ) * t ( p4 h5 ) * v ( h5 p2 p3 p4 )+[ - 0.5 ] * Sum ( h4 h5 p3 p6 ) * t ( p3 p2 h4 h5 ) * t ( p6 h1 ) * v ( h4 h5 p3 p6 )
+[ - 0.5 ] * Sum ( h5 h6 p3 p4 ) * t ( p3 p4 h5 h1 ) * t ( p2 h6 ) * v ( h5 h6 p3 p4 )+[ + 1.0 ] * Sum ( h4 h6 p3 p5 ) * t ( p3 p2 h4 h1 ) * t ( p5 h6 ) * v ( h4 h6 p3 p5 )
+[ - 1.0 ] * Sum ( h4 h6 p3 p5 ) * t ( p3 h1 ) * t ( p2 h4 ) * t ( p5 h6 ) * v ( h4 h6 p3 p5 )

0 = [ + 1.0 ] * v ( p3 p4 h1 h2 ) + [ - 1.0 + 1.0 * P( p3 p4 h1 h2 => p4 p3 h1 h2 ) ] * Sum ( h5 ) * t ( p3 h5 ) * v ( h5 p4 h1 h2 )
+[ + 1.0 - 1.0 * P( p3 p4 h2 h1 => p3 p4 h1 h2 ) ] * Sum ( p5 ) * t ( p5 h2 ) * v ( p3 p4 h1 p5 )+[ - 1.0 + 1.0 * P( p3 p4 h1 h2 => p3 p4 h2 h1 ) ] * Sum ( h5 ) * f ( h5 h1 ) * t ( p3
        p4 h5 h2 )
+[ - 1.0 + 1.0 * P( p4 p3 h1 h2 => p3 p4 h1 h2 ) ] * Sum ( p5 ) * f ( p4 p5 ) * t ( p5 p3 h1 h2 ) +[ + 0.5 ] * Sum ( h5 h6 ) * t ( p3 p4 h5 h6 ) * v ( h5 h6 h1 h2 )
+[ + 1.0 - 1.0 * P( p3 p4 h2 h1 => p4 p3 h2 h1 ) - 1.0 * P( p3 p4 h2 h1 => p3 p4 h1 h2 ) + 1.0 * P( p3 p4 h2 h1 => p4 p3 h1 h2 ) ] * Sum ( h6 p5 ) * t ( p5 p3 h6 h2 ) * v ( h6 p4
        h1 p5 )
+[ + 0.5 ] * Sum ( p5 p6 ) * t ( p5 p6 h1 h2 ) * v ( p3 p4 p5 p6 ) +[ + 1.0 ] * Sum ( h5 h6 ) * t ( p3 h5 ) * t ( p4 h6 ) * v ( h5 h6 h1 h2 )
+[ - 1.0 + 1.0 * P( p3 p4 h2 h1 => p4 p3 h2 h1 ) + 1.0 * P( p3 p4 h2 h1 => p3 p4 h1 h2 ) - 1.0 * P( p3 p4 h2 h1 => p4 p3 h1 h2 ) ] * Sum ( h6 p5 ) * t ( p5 h2 ) * t ( p3 h6 ) * v (
        h6 p4 h1 p5 )
+[ + 1.0 ] * Sum ( p5 p6 ) * t ( p5 h1 ) * t ( p6 h2 ) * v ( p3 p4 p5 p6 ) +[ - 1.0 + 1.0 * P( p3 p4 h2 h1 => p3 p4 h1 h2 ) ] * Sum ( h5 p6 ) * f ( h5 p6 ) * t ( p3 p4 h5 h2 ) * t ( p6
        h1 )
+[ + 1.0 - 1.0 * P( p3 p4 h1 h2 => p4 p3 h1 h2 ) ] * Sum ( h5 p6 ) * f ( h5 p6 ) * t ( p6 p3 h1 h2 ) * t ( p4 h5 )
+[ + 0.5 - 0.5 * P( p3 p4 h2 h1 => p3 p4 h1 h2 ) ] * Sum ( h5 h6 p7 ) * t ( p3 p4 h5 h6 ) * t ( p7 h2 ) * v ( h5 h6 h1 p7 )
+[ - 1.0 + 1.0 * P( p3 p4 h2 h1 => p4 p3 h2 h1 ) + 1.0 * P( p3 p4 h2 h1 => p3 p4 h1 h2 ) - 1.0 * P( p3 p4 h2 h1 => p4 p3 h1 h2 ) ] * Sum ( h6 h7 p5 ) * t ( p5 p3 h6 h2 ) * t ( p4
        h7 ) * v ( h6 h7 h1 p5 )
+[ - 1.0 + 1.0 * P( p3 p4 h2 h1 => p3 p4 h1 h2 ) ] * Sum ( h5 h7 p6 ) * t ( p3 p4 h5 h2 ) * t ( p6 h7 ) * v ( h5 h7 h1 p6 )
+[ - 1.0 + 1.0 * P( p3 p4 h2 h1 => p4 p3 h2 h1 ) + 1.0 * P( p3 p4 h2 h1 => p3 p4 h1 h2 ) - 1.0 * P( p3 p4 h2 h1 => p4 p3 h1 h2 ) ] * Sum ( h6 p5 p7 ) * t ( p5 p3 h6 h2 ) * t ( p7
        h1 ) * v ( h6 p4 p5 p7 )
+[ - 0.5 + 0.5 * P( p3 p4 h1 h2 => p4 p3 h1 h2 ) ] * Sum ( h7 p5 p6 ) * t ( p5 p6 h1 h2 ) * t ( p3 h7 ) * v ( h7 p4 p5 p6 )
+[ + 1.0 - 1.0 * P( p3 p4 h1 h2 => p4 p3 h1 h2 ) ] * Sum ( h7 p5 p6 ) * t ( p5 p3 h1 h2 ) * t ( p6 h7 ) * v ( h7 p4 p5 p6 )
+[ + 0.5 - 0.5 * P( p4 p3 h1 h2 => p3 p4 h1 h2 ) ] * Sum ( h7 h8 p5 p6 ) * t ( p5 p4 h1 h2 ) * t ( p6 p3 h7 h8 ) * v ( h7 h8 p5 p6 )
+[ + 0.25 ] * Sum ( h7 h8 p5 p6 ) * t ( p5 p6 h1 h2 ) * t ( p3 p4 h7 h8 ) * v ( h7 h8 p5 p6 )
+[ - 0.5 + 0.5 * P( p3 p4 h1 h2 => p3 p4 h2 h1 ) ] * Sum ( h5 h8 p6 p7 ) * t ( p3 p4 h5 h1 ) * t ( p6 p7 h8 h2 ) * v ( h5 h8 p6 p7 )
+[ - 1.0 + 1.0 * P( p4 p3 h1 h2 => p3 p4 h1 h2 ) ] * Sum ( h6 h8 p5 p7 ) * t ( p5 p4 h6 h1 ) * t ( p7 p3 h8 h2 ) * v ( h6 h8 p5 p7 )
+[ + 1.0 - 1.0 * P( p3 p4 h2 h1 => p3 p4 h1 h2 ) ] * Sum ( h6 h7 p5 ) * t ( p5 h2 ) * t ( p3 h6 ) * t ( p4 h7 ) * v ( h6 h7 h1 p5 )
+[ - 1.0 + 1.0 * P( p3 p4 h1 h2 => p4 p3 h1 h2 ) ] * Sum ( h7 p5 p6 ) * t ( p5 h1 ) * t ( p6 h2 ) * t ( p3 h7 ) * v ( h7 p4 p5 p6 )
+[ + 0.5 ] * Sum ( h7 h8 p5 p6 ) * t ( p5 h1 ) * t ( p6 h2 ) * t ( p3 p4 h7 h8 ) * v ( h7 h8 p5 p6 )
+[ + 1.0 - 1.0 * P( p4 p3 h1 h2 => p3 p4 h1 h2 ) - 1.0 * P( p4 p3 h1 h2 => p4 p3 h2 h1 ) + 1.0 * P( p4 p3 h1 h2 => p3 p4 h2 h1 ) ]
        * Sum ( h6 h8 p5 p7 ) * t ( p5 h1 ) * t ( p4 h6 ) * t ( p7 p3 h8 h2 ) * v ( h6 h8 p5 p7 )
+[ + 1.0 - 1.0 * P( p3 p4 h1 h2 => p3 p4 h2 h1 ) ] * Sum ( h7 h8 p5 p6 ) * t ( p5 h1 ) * t ( p6 h7 ) * t ( p3 p4 h8 h2 ) * v ( h7 h8 p5 p6 )
+[ + 0.5 ] * Sum ( h5 h6 p7 p8 ) * t ( p3 h5 ) * t ( p4 h6 ) * t ( p7 p8 h1 h2 ) * v ( h5 h6 p7 p8 )
+[ - 1.0 + 1.0 * P( p4 p3 h1 h2 => p3 p4 h1 h2 ) ] * Sum ( h5 h7 p6 p8 ) * t ( p4 h5 ) * t ( p6 h7 ) * t ( p8 p3 h1 h2 ) * v ( h5 h7 p6 p8 )
+[ + 1.0 ] * Sum ( h7 h8 p5 p6 ) * t ( p5 h1 ) * t ( p6 h2 ) * t ( p3 h7 ) * t ( p4 h8 ) * v ( h7 h8 p5 p6 )
```

人工知能
（平田　聡博士）

波動関数法は系統的:

CCSD(1.3万行), CCSDT(3.4万 行), CCSDTQ(8万行), …

- 波動関数法の大多数は第**2**量子化によって定義される
- 導出される方程式は、「テンソル表現」をとる
- プログラムは、行列の足し算・掛け算の組合わせ
- 従って、波動関数法の式導出およびプログラム作成は抽象化・自動化できる

コンピュータの算術演算以外の側面、
より高度な論理演算や人工知能をもっと積極的に活用すべき

若い人のフレッシュなアイデアを生かすシステムを！

学際計算科学・工学人材育成プログラム

東京大学の試み

次世代スーパーコンピュータを駆使した
大規模シミュレーションを実施し、
新しい科学を開拓する人材の育成

学際計算科学・工学人材育成委員会

平尾公彦

東京大学情報基盤センター
米澤明憲センター長
石川　裕教授、中島研吾教授、片桐孝洋准教授

常行真司教授（理学系）、奥田洋司教授（工学系）
中谷明弘准教授（新領域）、須田礼仁准教授（情報理工）
加藤千幸教授（生研）

部局横断的な学際領域の教育をどう実現するのか？

一般論－どういう人材を育てるのか？

アプリプログラムを上手に使える人材

実験/理論と計算シミュレーションの両方を使いこなせる

アプリケーションソフトを研究開発できる人材

新しい計算モデルを創造し実現することができる人材モデル
アルゴリズム、並列プログラミングを一体で扱えなければブレークスルーはない

開発ツールを作れる人材

並列計算機システムを研究開発できる人材
並列計算機のアーキテクチャ、**OS**、ミドルウェア、プログラミング環境を研究する人材、このままではスーパーコンピュータシステムを開発できる人材が枯渇してしまう

東京大学はどういう人材を育てるのか？

グランドチャレンジを自ら作り上げる人

モデル、アルゴリズム、ソフトウエアを自ら開発できる人

グランドチャレンジに果敢に挑戦する人

システム・アーキテクチャを共同開発できる人

⇩

学際計算科学・工学人材育成プログラム

計算科学教育

より深い理解

専門とする科学/工学については、よりファンダメンタルなことを、より深く、きっちりと教える、深い理解には汎用性がある

ふくらみをもった幅広い視野

専門の科学/工学を深く理解するとともに、応用数学、計算機科学を、体系的に理解させる

全学体制で計算科学の人材育成プログラムを実施

学際計算科学・工学人材育成プログラム

- 情報科学は学際的であり、全学体制による環境整備が重要
  - 教育＋その後の研究支援
  - 単に数年の教育だけでなく, **HPC**技術を駆使した研究を生涯支援するための環境整備（＝開発基盤, 協力体制）が必要
- 計算科学と計算機科学、数理科学の融合, スパイラル効果による発展

これまでの東京大学の計算科学教育
理学研究科地球惑星科学専攻の例

- 一般教育（**1, 2**年）における情報科学教育
- 地球惑星物理学演習（**3**年）
  計算機リテラシー、**FORTRAN**、基本的な数値解析
  計算機漬け
- 地球物理数値解析（**4**年）
  偏微分方程式の数値解法
- 並列計算プログラミング、先端計算演習（大学院）
  MPIを使って差分法、有限要素法などのアプリの並列化技術、先駆的

これまでの経験からの教訓

- 並列計算プログラムは決して難しくない
- 重要なのは背景にある科学/工学、基本となるアルゴリズム **Science, Modeling, Algorithm**であるーその上でのプログラミング
- 計算機に使われてはならない
- 良い並列プログラムはよ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
- もっとも重要なのは「並列プログラミング」に先立つ「実用的なアプリケーションのプログラミング」の習得
- 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

並列プログラミングへの道のり

① 計算機リテラシー
プログラミング言語

↓

② 科学技術計算基礎論・演習
（数値解析・プログラミングの基礎）

ここに厚い壁

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 並列プログラミング

Science
Modeling
Algorithm
Software
Hardware

①～④の各レベル（縦糸）において「SMASH」（横糸）
③が最も重要，かつ教育困難
現状はアルゴリズム中心

e-LearningとCertification

e-Learning

- 様々なバックグラウンドの受講者の多様なニーズに柔軟に対応
- 受講者の負担を極力増やさない

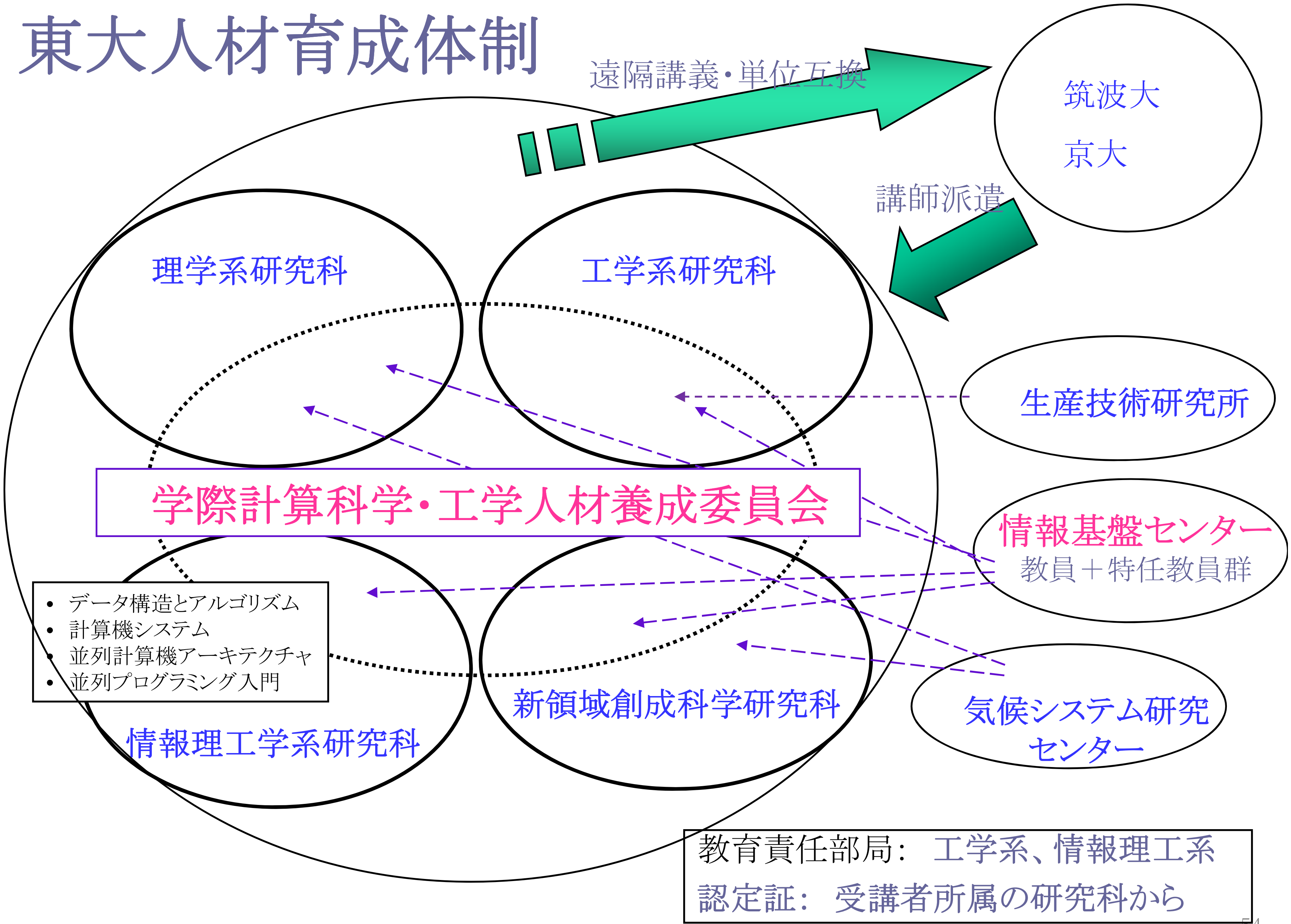
  コマ数をなるべく増やさない

Certification

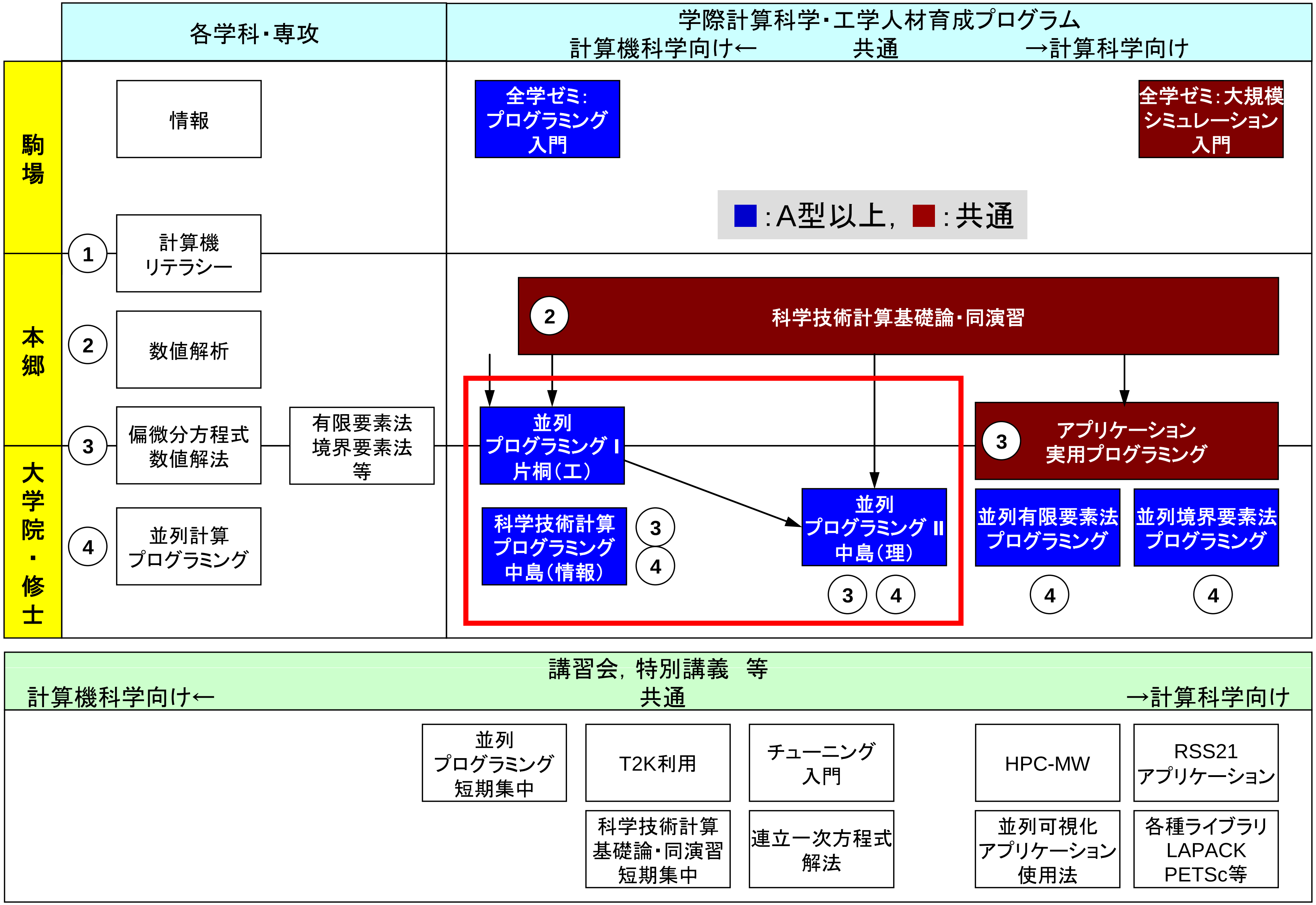
- 単位、認定証（certification）、Double Major

  第三者による評価, 検定

東大人材育成体制

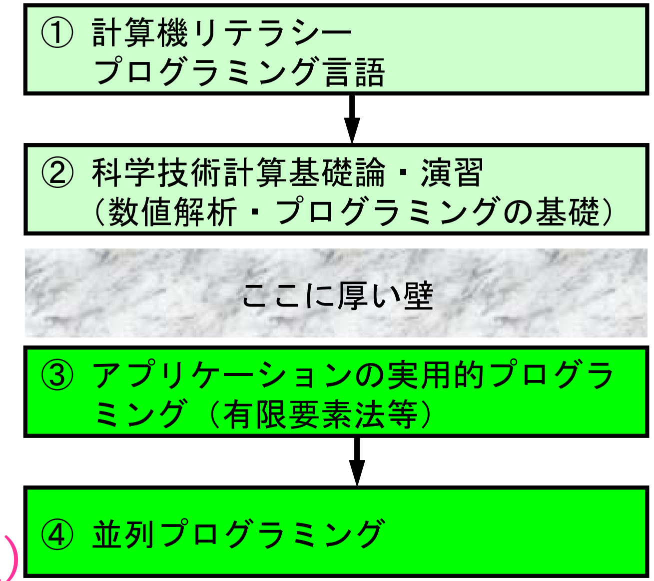

遠隔講義・単位互換
筑波大
京大
講師派遣
理学系研究科
工学系研究科
生産技術研究所
学際計算科学・工学人材養成委員会
情報基盤センター
教員＋特任教員群
• データ構造とアルゴリズム
• 計算機システム
• 並列計算機アーキテクチャ
• 並列プログラミング入門
新領域創成科学研究科
情報理工学系研究科
気候システム研究センター
教育責任部局： 工学系、情報理工系
認定証： 受講者所属の研究科から

カリキュラム案(連続体力学)



長期的スケジュール：モデル講義

- H20年度冬学期以降開講（赤字は既存，開講決定）
  - ②
    - 科学技術計算基礎論（情報理工）
  - ③，④
    - 有限要素法プログラミング（工）
    - 分子動力学プログラミング（理）
    - 差分法プログラミング（工?）
    - 境界要素法プログラミング（新領域?）
    - 基礎的共通講義
      - 並列計算プログラミング（連続体）（理）
      - 科学技術計算プログラミング（有限要素法）（情報理工）
      - 非数値分野プログラミング（新領域）
  - ④
    - 並列計算アーキテクチュア（情報理工）
    - スパコンプログラミング（工）



計算科学領域の確立

計算科学は学際領域であり、発展の激しい分野

異分野融合、学際領域の開拓とともに、新しいディシプリンの構築、体系化が必要

計算科学のディシプリンの確立、マッピングあるいは学術体系化は、この分野の研究や教育に参加する若者にとっては道案内役として重大な役割を負う

人材養成面でAll Japan体制作りに協力

- 全国の教育機関と連携し、人材養成ネットワークを作り
- 遠隔講義・演習を活用し
- 単位・資格認定の相互乗り入れを可能にし
- 演習用マシンの相互乗り入れを可能にし
- 教育人材の交流を図り

⇩

大学と連携し、次世代スパコン施設にも人材養成の機能を！

啓発活動

- 優れた潜在的ユーザー、分野の掘り起こし
- コンピュータには見向きもしなかった人々への教育・啓発システムを
- 次世代スーパーコンピュータで得られる研究成果を小中学生を含む子供たちに発信できる仕組みを
- 国際共同研究の推進と人材養成を

Scienceのしっかりした基盤が重要

21世紀は予測の科学の時代、計算科学はその基盤

数値実験による新しい概念形成こそイノベーション

異分野融合と人材育成が重要

若い人のフレッシュなアイデアをいかに生かすか