

HPC人材育成のための教育のあり方について  
(中間報告)

平成22年11月

HPC 人材育成のための教育のあり方検討会

## 目 次

はじめに	2
1. 計算科学の位置付け	2
2. 育成すべき人材	3
3. 必要な教育体系	4
(1) レベルに応じた教育内容	5
(2) 大学における教育パターン	5
(3) 教育カリキュラム	8
(4) 企業人を対象とした教育プログラム	13
(5) 教育・人材育成を行っていくための体制 (計算科学教育センター(仮称)の設置)	15
4. キャリアパスと業績評価	17
5. 結論	18
参考資料1 HPC 人材育成のための教育のあり方検討会について	20
HPC 人材育成のための教育のあり方検討会 名簿	
参考資料2 審議経過	22

## はじめに

計算科学は、大規模計算という新しい方法論に基づく知の体系であり、コンピュータの発展とともに大きく成長を遂げている。ペタフロップス級のスーパーコンピュータを駆使した計算科学の時代を迎えるにあたり、さまざまな分野でこれまで以上に大規模な計算処理を主要な研究手段とする計算科学の発展が期待される。今やシミュレーションをはじめとする計算科学の手法は、先端的な科学研究のみならず、工学の諸分野はもちろん、ものづくりや金融など産業界の現場においても、また人文・社会科学の研究においてもなくてはならないものとなり、計算科学技術とも呼ばれている。

他方、計算科学に必要な大規模な計算処理のための研究分野は HPC (High Performance Computing、高性能計算) と呼ばれており、コンピュータ科学の重要な分野である。これは、アルゴリズム、プログラミング言語、コードチューニング、並列処理、データ配置、マッピング、可視化、デバッグツール、性能評価などの広い分野を含み、高性能計算のためのシステム化技術を対象としている。HPC は計算科学の諸分野とも密接と関連している。

このような状況において、計算科学を支える人材 (HPC 人材) を育成することは我が国の発展の上で大きな課題となっているが、残念ながら十分な取り組みは行われていない。その大きな原因は、系統だった人材育成・教育システムが確立していないということであり、この確立は、早急に取り組むべき重要課題である。

一方、我が国においては、平成 24 年の次世代スーパーコンピュータの完成を見据え、革新的ハイパフォーマンス・コンピューティング・インフラの整備とこれを主導するコンソーシアムの構築、次世代スパコンを中核拠点とする計算科学技術推進体制の構築を目指した取組を進めている。このようなインフラや体制整備は、人材育成のための体制を整備するという観点からも大きな機会であり、また、我が国の計算科学技術推進体制構築という観点からも、人材育成・教育は重要な課題である。

このため、本検討会においては、我が国における系統的な計算科学技術の人材育成・教育システムを検討するとともに、具体的な取組を行っていくための体制構築について検討を進めてきた。検討に当たっては、21 世紀 COE プログラムや大学院 GP における教育・人材育成の取組、海外の状況等を参考としながら、必要な議論を行った。その結果、計算科学の位置付け、育成すべき人材、必要な教育体系、計算科学人材のキャリアパスといったことについて一定の考え方をとりまとめるに至り、ここに、本検討会の中間報告として報告をする。

### 1. 計算科学の位置付け

計算科学においては、まず対象とする科学上の問題(Science)を何らかの数学的なモデルで記述することが必要である。これをモデリング(Modeling)という。数学的なモデルをコンピ

ュータで扱うためにはそれをコンピュータで解くための手順を記述する必要がある。この手順をアルゴリズム(Algorithm)という。アルゴリズムはコンピュータのプログラムとして記述される。プログラムは、コンパイラによって解釈され実行可能なコードに変換されるが、効率よくプログラムを開発し、改良し、高い計算性能を達成するためには、デバッグやチューニングのツールが必要である。コンパイラやツールなどのシステム・ソフトウェア(Software)も計算科学にとって重要な役割を果たす。プログラムを実際に行うのはコンピュータのハードウェア(Hardware)であり、有効に計算を行うにはハードウェアやこれを抽象化したアーキテクチャの知識も不可欠である。これら 5 つの段階を、それぞれの頭文字をとって、仮に SMASH と呼ぶことにする。

これまでの計算科学は、それぞれの分野の一部としての計算科学であり、分野の中に閉じていた。従って、分野の中での交流はあったが、異なる分野間での計算科学の交流は必ずしも盛んではなかった。しかし、違った分野であっても、対象をモデル化し、数式の形に定式化した段階ではかなり類似性が高く、モデル化の方法論やアルゴリズムなどで互いに学ぶところは少なくない。これにより、科学技術の諸分野を、横断的、包括的にとらえることができる。

さらに、並列コンピュータの利用技術にも共通性があり、分野間で交流することは有意義である。また、複雑化したアーキテクチャを使いこなすには、コンピュータ科学(「計算機科学」とも呼ばれる)との連携も不可欠である。計算科学を発展させるには、他分野と交流でき、コンピュータ科学とも連携できる計算科学者、また計算科学の分かるコンピュータ科学者がますます増えることが求められている。

## 2. 育成すべき人材

科学技術の諸分野における計算機シミュレーションの浸透、コンピュータハードウェアの急速な進歩、発達を背景に、大規模並列シミュレーションによる科学的発見、技術革新への期待は、産学において一層高まっている。最新のハードウェアを駆使して、大規模並列シミュレーションプログラムを開発するための体系的な教育プログラムは、幾つかの事例を除くと世界的に見てもまだ少ない。また、ハードウェア技術の急速な高度化、複雑化、多様化により、その計算能力を最大限引き出すアプリケーションの開発は、より一層複雑なものとなっている。

大規模並列シミュレーションに基づく科学技術の諸分野の研究開発の持続的な発展を促進するためには、科学技術とコンピュータ科学、応用数理学の密接な協力とともにそれらを融合する学際的な研究分野の創成が必要である。そのような融合分野(計算科学技術(Computational Science and Engineering, CS&E))の確立を最終的な目的として、人材育成、開発支援体制の整備を実施する必要がある。

まず重要なことは、以下のような 2 種類の人材を科学技術の各専門分野において育成す

ることが必要である。

- A 型（作れる人材）：科学技術の各専門分野において、大規模並列シミュレーションプログラムを自ら開発し、研究開発を推進していく人材
- B 型（使える人材）：科学技術の各専門分野において、既存の大規模並列シミュレーションプログラム（商用コード、オープンソースコード）を利用して、研究開発を推進していく人材

大学や大学院における教育プログラムを整備し、持続的に毎年新たな人材を育成していくことは重要である。これまで B 型の人材はある程度育成されてきたが、今後は A 型の人材も育成する必要がある。

A 型の人材の場合、各分野の研究者、技術者となってから更なる計算機ハードウェアの変化にキャッチアップしていくことは必要であるが、容易ではない。個々の研究者、技術者が継続的に最先端の研究開発を持続していくためには、「HPC ミドルウェア」に代表されるような並列シミュレーションコード開発支援環境の整備が重要である。開発支援環境の開発、整備こそが科学技術とコンピュータ科学、応用数理科学の融合分野である「計算科学技術」の重要なミッションであり、主たる研究テーマである。

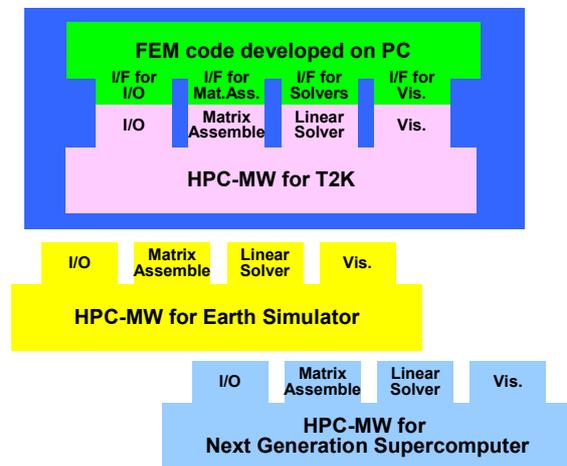


図1 HPC ミドルウェア（FEM の例）

融合分野としての計算科学技術を担うのが S 型（いじれる人材）であり、並列シミュレーションコード開発支援環境の研究開発を担う。S 型人材はコンピュータ科学、応用数理科学をバックグラウンドとし、数値アルゴリズム等を専門とする学生の中から育成することが期待される。

### 3. 必要な教育体系

今後ペタフロップス級のスーパーコンピュータを駆使した計算科学の時代を迎えるにあたり、様々な分野でこれまで以上に大規模な計算処理を主要な研究手段とする計算科学の進展が期待される。計算科学に関する教育体系を検討する上では、計算科学・コンピュータ科学の学際性を考慮することが重要である。例えば、異なる応用計算分野であっても、解くべき問題に対し、モデル化および数式として定式化した段階ではかなり類似性が高いことが多い。従って、それを解くアルゴリズムや離散化の手法にも共通性がある。また解

くべき問題の高負荷部が、大規模な連立一次方程式、固有値問題等の数値計算に帰着する場合には、数値解析の手法そのものにも共通性がある。また、超並列計算機の性能を最大限に引き出すための並列コンピュータの利用技術に関しても共通性があり、通信ミドルウェア、I/O、耐障害対応等、計算機システムそのものに関する基礎知識もこれまで以上に要求され、計算結果の可視化技術も含めたこれらの知識は分野を問わず重要である。更に、全体の系をまるごと解析する手法としてマルチスケール・マルチフィジックス・シミュレーションはますます重要となることが予想される。これらの連成解析においては、他の複数の応用分野との密接な連携が必要であるほか、時空の離散化手法が異なるモデル間を結ぶための学際的な知見が必要である場合が多い。

### (1) レベルに応じた教育内容

教育について検討を行う際、多様なレベルに応じた取組を考えることが必要である。ここでは、中高生レベルから大学院博士課程・社会人レベルまでを視野に入れ、レベル毎に施すべき教育内容を示す。

#### [中高生レベル]

- 出張講義・実習、教員へのトレーニング、プログラミングコンテスト等

#### [学部レベル]

- 一般教養としての数値解析、コンピュータ科学
- 情報リテラシ（情報倫理、セキュリティ、知財ツールの利用方法等）
- 情報基礎（計算論、言語、アルゴリズム、プログラミング、並列処理・並列プログラミングの基礎等）

#### [大学院修士課程レベル]

- 並列処理・並列プログラミング
- モデリング
- 計算結果の評価・解析
- コンピュータアーキテクチャ
- 可視化

#### [大学院博士課程・社会人レベル]

- 研究の進展や必要に応じた教育を提供できる体制を構築
- 品質の評価

### (2) 大学における教育パターン

大学での学部教育、大学院教育は、計算科学人材を社会に送り出すための最終的な教育の場であり、ここでの教育の在り方は、非常に重要である。

計算科学のような境界領域での人材育成を考えるにあたっては、教育すべき科目はコンピュータ科学と応用領域の双方にわたるので、通常の教育の2倍なることを考慮しなけ

ればならない。この時、双方の領域を広く教育する方法およびどちらかの領域に軸足を置く教育方法が考えられる。

具体的な教育のパターンを考える。まず、双方の領域を教育する場合のパターンは図 2 に示す 3 通りとなる。学部時代に応用系の教育を受けたのち大学院で情報系の教育を受けるパターン 1、学部時代に情報系の教育を受けて大学院で応用系の学問に進むパターン 2、学部時代に情報系の教育を受けた学生、および学部時代に応用系の教育を受けた学生を一緒にして大学院で計算科学を教育するパターン 3 である。パターン 1 と 2 の場合は、ダブルメジャーやデュアルディグリーといった教育システムの導入も検討すべきである。パターン 3 では大学院教育において専門知識が異なる学生の教育となるので、講義形式ではなく、演習やグループ別教育の導入などの配慮が望まれる。

どちらかの領域に軸足を置いて専門家を育て、専門でない他の領域の専門家と共同研究が行えるように教育するのがパターン 4 である。大学院修士の時に他の領域の学生、専門家と議論をする教育的な場を積極的に提供する枠組みである。博士になると他の領域の学生、専門家と研究に関する議論を展開できるような場を提供する。今後は、異なる専門領域の研究者の協働作業が重要になると考えると、自分の専門分野を確立した上で、他分野の理解を実践的に進めるという教育方法も基本パターンとして成り立つと考える。

これらのパターンのどれが有効に働くかは 2. で述べた育成すべき人材の種類に関連があると思われるが更なる検討が必要であろう。

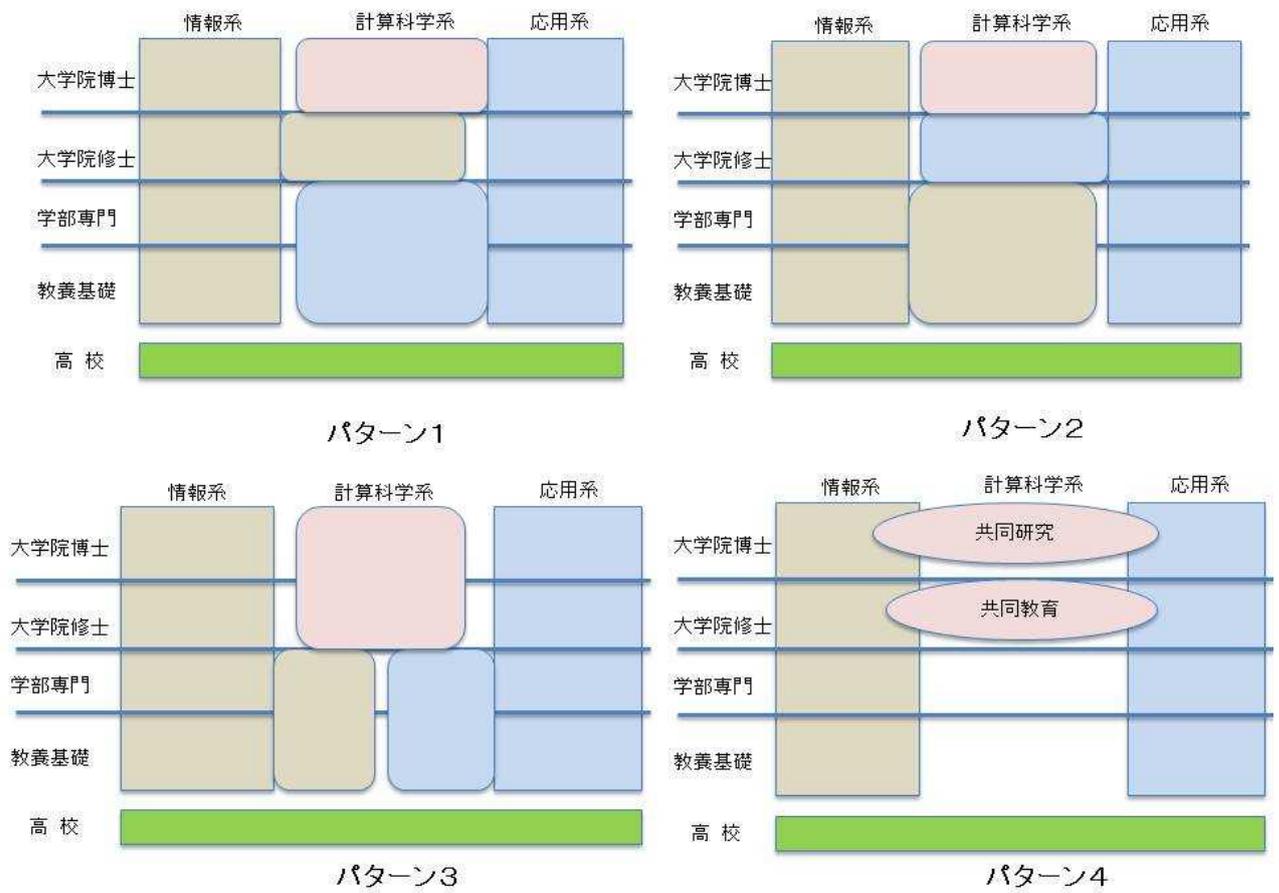


図2 大学における教育パターン

### (3) 教育カリキュラム

#### 基本的な考え方：4S 型人材育成戦略

並列計算機を駆使した大規模シミュレーションを実施し、新しい科学的発見、技術革新に結びつけるためには、まず、プログラミング、特に並列計算機におけるプログラミング（以下「並列プログラミング」と呼ぶ）の技術を習得していることが必須である。わが国においては、コンピュータ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講義・演習は皆無であったが、最近では新たな取り組みもみられる。その一例として以下に東京大学での事例を紹介する。

東京大学では平成 20 年 2 月より「学際計算科学・工学人材育成委員会」を発足させ、関連部局（理学系、工学系、情報理工学系、新領域創成科学の各研究科）と情報基盤センターによって、全学的な HPC 教育プログラムに向けて具体的なカリキュラムの策定を進めており、平成 21 年度から本格的に開始している。

特徴は、以下に示す「4S 型人材育成戦略：System、Stage、Status、Style」である。

#### System :

科学技術計算プログラミングを習得するためには「SMASH」すなわち、Science、Modeling、Algorithm、Software、Hardware の幅広い分野をカバーすることが必要である。カリキュラム全体としてはもちろん個々の講義、演習においてもできるだけ「SMASH」をバランス良く習得できるよう配慮することが体系的教育の最も重要なポイントである。

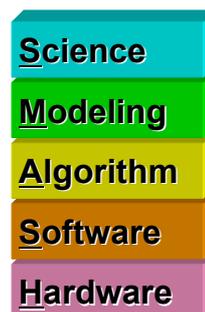


図 3 SMASH

#### Stage :

並列プログラミングを習得するためには、

- ① 計算機リテラシ、プログラミング言語
- ② 基本的な数値解析
- ③ 実アプリケーションのプログラミング
- ④ その並列化

という 4 つの段階 (stage) が必要である。最も重要かつ、最も体系的教育が欠如しているのが③の段階である。本人材育成プログラムでは④までを修士課程修了時まで習得することを目指している。

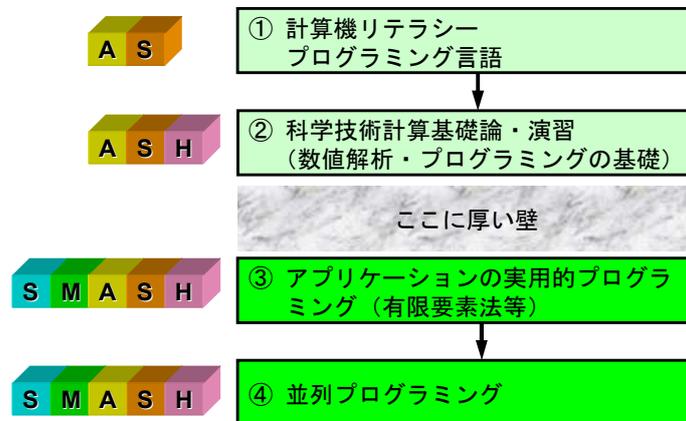


図4 並列プログラミングへの道と SMASH

Status :

2. で述べた A 型（作れる人材）、B 型（使える人材）、S 型（いじれる人材）など、様々な人材の育成を視野に入れていく必要がある。

Style :

「Stage」であげた、①、②については学部レベルでほとんどの学生が学んでいるが、③、④については新たな履修科目となる。集中講義、講習会、e-Learning など様々な形態 (style) の教育を提供することで、学生の負担増を極力抑え、効果的な学習を促進する。

1) 基本的な方針

基本的な考え方としては、以下が重要である：

- 修士課程までは、とにかく基礎をしっかりやる、ただ修士課程修了後に就職する学生も多数居ることを考慮して、特に図 4 の③、④段階は実用的な内容を含む必要がある
- 専門的な教育は博士課程以降、あるいは個別に実施する
- 理研（計算科学研究機構）、学際大規模情報基盤共同利用・共同研究拠点（8 大学）の協力のもとにインターンシップを推進する
- 社会人教育も念頭において、短期間に集中的に学べる教育プログラムが必要

カリキュラム、教材作りについては以下のことが重要である：

- それぞれの講義・演習の目的、受講者を明確にすること（「SMASH」が指標となる）

- 多くのものを一度に追うことは避けなければならない（例：物理、エンジニアリングジャッジとプログラミング）
- 研究内容を講義する、実プログラムを演習に使うことは避けなければならない（最先端の内容を教えるという観点ではある程度許容されるが、中心におくべきではない）
- 教材（例題プログラム）の作り込み
  - 最先端研究に使うものでも、Toyでもダメ
  - 少々の誰にでも理解できる Science を盛り込むことがキーとなる
  - 限られた時間内で基礎から応用まで幅広い内容を効率的に学べるような内容であること

ここでは主に A 型（作れる人材）の育成をターゲットとしたカリキュラムを紹介する。A 型（作れる人材）は主に HPC-MW 等を使って並列アプリケーションを自分で開発できる人である。B 型（使える人材）については、各分野における伝統的な講義、演習、実験に計算機シミュレーションを導入することである程度の育成は可能である。

S 型（いじれる人材）については、コンピュータ科学、応用数理科学を専門とする学生の中から育成する場合、アプリケーションの内容を知ることが第一である。したがってアプリケーション側のプログラミング教育（図 4 の③、④段階）と共通の講義、教材が適用可能である。

次節 2) ではこのうち共通的な内容について述べる。並列プログラミングについて修士課程修了までに習得しておくというのが基本的な方針である。B 型、S 型人材についてもある程度のプログラミング教育を修士課程修了までに習得しておくことは必須である。

そこで得たことを更に最先端シミュレーションにつなげるためには、より大規模な環境で、ある程度分野に特化した指導を個別に行う必要がある。そのために 3) で述べるような「若手育成枠」を考える。

## 2) 教育カリキュラム（共通：学部・修士）

東京大学の例では、図 4 に示した 4 つの Stage のうち、基礎的な段階である①、②については、既に多くの学部学科で実施されている。

図 4 に示した 4 つのレベルのうち、③、④については、以下のようなことが言える。

- もとのアプリケーションのアルゴリズム、実装を良く理解していれば、並列化はそれほど困難なことではない。従って③レベルが最も重要である。
- ②⇒③の間には大きな壁があり、②レベルの講義で教えられている内容を基にして、実用的なアプリケーションプログラムを開発することは困難である（例：有限要素法における疎行列解法）。

- ③に関連して開講されている講義は、理論に偏ったものが多く、プログラミングまでカバーしているものはほとんど無い。
- 実用的なアプリケーションプログラムに触れるのは、研究室配属後であるケースが多く、研究室の先輩等から部分的な知識を得る。
- 従って、少数の例外を除いて、自らアプリケーションプログラムを開発するための実装能力と必要な体系的知識、経験を持つものは居ない。

というのが現状である。

基本的に①、②レベルのものについては、③、④につながるガイドラインを作成して、できるだけ既存の講義・演習、内容のマイナーチェンジにとどめることが現実的である。③、④についてはプログラミングも含む基礎的な講義・演習を様々なアプリケーションの手法をカバーできるような講義・演習を開講することが重要である。実際には、下記のような手法別にそれぞれ講義を開講することが効果的であると考えられる。

- 差分法、有限体積法
- 有限要素法
- 境界要素法
- スペクトル法
- 分子動力学法
- その他の非数値的手法（探索等）

理解しやすい現象を対象として、ごく基本的な事項（定式化、行列解法、プログラミング、可視化）等について並列化まで含めて講義、演習を実施する。科学技術計算である以上 SMASH の頭の S (Science) を抜きにして考えることは不可能であり、誰にでも理解できる現象を題材として扱う必要がある。プログラムのステートメントレベルでの解説、スパコンを使った実習の導入は効果的である。

受講者の負担を増やさないためには：

- ①、②段階：既存の講義を活用
- ③：週 1 コマ (90 分)、一学期 (15 回)
- ④：週 1 コマ (90 分)、一学期 (15 回)

に抑えることが重要である。

様々な理由により①、②レベルの習熟度が充分でない学生に対しては、短期集中講義、講習会、e-Learning 教材の充実が重要である。これまでの経験を考慮すると、5 日間×4～5 コマの集中講義は効果的である。一般にこのような集中講義は夏季・秋季、冬季、春季の

休業期間中に実施するのが通例であるが、学期中にも、早朝、夜間等を利用して開講することも考慮すべきである。また、①、②レベルまでであれば、e-Learning 教材を整備すれば、独習することも可能である。効果的な教材の検討が必要である。

この他、以下のような内容に関する講義・演習についても検討する必要がある：

- 計算機システムそのものに関する講義
- ソフトウェア工学

### 3) 若手育成枠

「若手育成枠」は、企業の研究者・技術者を含む科学・工学分野の若手研究者・技術者に対して、大規模コード開発を支援するとともに、ある程度の規模の計算機資源を重点的に配分し世界レベルの研究者・技術者の育成に資するものである。概要は以下である：

- ① 公募による
- ② 10,000 コア程度の計算機資源を 1 ヶ月占有できる他、1,000 コア程度の計算機資源は常時使用できる。計算資源は、理研（計算科学研究機構）、学際大規模情報基盤共同利用・共同研究拠点（8 大学）から供出する。
- ③ 理研（計算科学研究機構）、学際大規模情報基盤共同利用・共同研究拠点のコンピュータ科学・応用数理科学分野の研究者の個別の指導、支援を受けられる。
- ④ 1 回 1 週間～2 週間、年 2～3 回、各機関に滞在するインターンシップを含み、集中的に指導、支援を受けられる。
- ⑤ 本育成枠の活動によって学位を取得した場合、Designed Emphasis in CSE (DE-CSE) のような「副専攻」相当の称号を与える（これは拠点の活動を通じて各大学にも展開する）。

応募資格等は以下のように考えている：

- ① 独自開発したコードを持っており、それを大規模化してシミュレーションを実施することによって新しい科学、技術を開拓したいと考えている若手研究者・技術者（A 型）
- ② 原則として博士課程以上、年齢は 40 歳未満
- ③ 企業の研究者、技術者も応募可能
- ④ 期間は 2 年（ないし 3 年）間、延長無し
- ⑤ 1) で紹介したような教育プログラムもしくは相当以上のコースを修了しており、基本的な並列計算プログラミングに関する知識と経験を有すること

支援を行うコンピュータ科学、応用数理科学の研究者もできるだけ同世代の若手（S 型人

材)を中心とし、次世代を担う分野間協力体制の形成に資する。ここで得られた成果も「HPC-MW」の整備に貢献する。

研究成果をまとめた論文がプログラム終了後1年以内にSC(ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis)、IPDPS(IEEE International Parallel & Distributed Processing Symposium)、ICS(ACM/SIGARCH International Conference on Supercomputing)級の国際会議で採択された場合には費用を支援するような援助の方法も考えられる。

- ① 外部資金獲得、研究プロジェクトへの展開
- ② 学際計算科学・工学人材育成プログラムを受講していない者への対応

#### 4) 組織を越えた連携

以上に記した教育カリキュラムの実現や若手教育枠の創設においては、全国的な組織を越えた連携が求められる。この達成のため、後述の計算科学教育センター(仮称)の設置の検討が行われることを期待する。

#### (4) 企業人を対象とした教育プログラム

産業界ではものづくり分野を中心としてシミュレーションをはじめとする大規模計算に基づく手法(計算科学技術)の活用がイノベーション創出に直結すると認識されており、その成否を握るのは人材である。イノベーション創出にとっては、まずなによりも計算科学技術を自在に使いこなす人材(B型)がもっとも必要であり、さらにそのような計算科学技術そのものを生み出す人材(A型)が必要である。これはいわば車の両輪であって、さらにこの両輪をつなぐ車軸のような存在、すなわち最先端HPCシステムに精通し、プログラム移植やチューニングなどを行う人材(S型)が必要である。我が国のこれまでの大学教育においては主に計算科学技術を生み出す人材の教育プログラムに注力してきたが、それ以外の人材育成に関しては手薄の感が否めない。

現在の課題に対して適切にシミュレーション技術を適用し、結果を正しく解析できる人材の育成、すなわち計算科学分野における課題解決型人材育成は産業界がもっとも期待するところである。したがって企業人に対する教育プログラムはこの部分が中心となる。この教育プログラムは分野ごとに設定されるべきで、演習・実習中心の短期集中型として、個々の企業では経験できない規模の計算を短時間でもよいから体験することで、自らの課題解決へのきっかけを与えることが期待される。また、単に解析を行うだけでなく、モデルの妥当性や計算精度、解析結果の吟味といった“解析品質の考え方”も体得すべき重要な視点である。このような教育プログラムの実施・受講がその後の産学共同研究に発展することも期待される効果のひとつである。

我が国のH P C産業のうち、ハードウェア産業は世界的にみてもトップグループに位置するが、ソフトウェア産業は脆弱な状況にある。これを強化するうえでH P Cソフトウェア技術を生み出せる人材の育成は重要である。ここではアルゴリズムに立ち戻った教育も必要であるため、講義と演習を組み合わせた半期程度の期間が必要である。企業人対象であることを考慮すればサテライト教育も取り入れるべきであろう。この教育プログラムでは受講者の素養に応じて教育プログラムを用意する必要がある。すなわちハードウェアの素養をもつ技術者にはソフトウェア工学の新しい知識を、ソフトウェアの素養を持つ技術者にはハードウェア工学の新しい知識を教育することが必要で、その意味で、H P Cのキャリアアップ教育であり、大学における再教育プログラムと位置付けることもできる。このような教育は従来の大学教育で行われてきたことに近く、教育内容も充実している従来例が多いが、とくに理学系の教育実施機関においては品質工学的な視点を取り入れるべきである。

計算科学技術を生み出し、それを十分に使いこなしていく上で、プログラムをチューニングし、大規模システムを運用していく支援要員の育成も企業にとっては重要な視点である。このような人材の育成はこれまでほとんど行われてはならず、企業においてもO J Tを通して各社独自に育成してきた。しかし昨今のシステムの大規模化や技術進展の激しさから企業内育成は困難になっている。大規模なシステムを運用する場（たとえば理化学研究所計算科学研究機構や各戦略拠点など）での実践的な育成システムも検討されるべきである。

学生が在学中に産業界での計算科学技術の活用実態について知ることはその後のキャリア形成にとって有益なことである。そのよい機会を与えるのがインターンシップ制度である。計算科学技術のような先端的な技術に関しては従来のような短期インターンシップ制度ではなく、ある程度手を動かして産業界での計算科学技術の利活用の在り方を具体的に知るという長期インターンシップ制度の導入が望ましい。一方で産業界の技術者が大学で学ぶ逆インターンシップ制度の導入も望まれる。このような長期インターンシップ制度は大学側からすれば学生の教育・育成をある一定期間、大学とは関係のない企業にゆだねることになるので、大学教育のあり方としては疑問であるという意見もある。長期インターンシップを導入するに当たってはその実施内容や評価方法について、大学・企業の両者間で合意を形成しておく必要がある。また、インターンシップ制度／逆インターンシップ制度は大学と企業にそれぞれ負担を強いるものであるので若手研究者派遣助成制度なども検討するべきである。

大学と産業界がそれぞれの課題を共有し、お互いの戦略を相互理解することがこの分野の人材育成にとっては必要なことであり、そのようなことを定期的に議論できる場の設定と、その間をつなぐ教育コーディネータの育成・配置にも取り

組む必要がある。

また、以上のような取組を組織横断的に支援する社会人教育の拠点の形成が望まれる。

#### (5) 教育・人材育成を行っていくための体制（計算科学教育センター（仮称）の設置）

計算科学（計算機シミュレーション）は、あらゆる研究分野で有力な研究手法であり、その有用性は計り知れない。この計算機シミュレーション技術を広く普及させるためには、全国的な教育体制が必要と考える。それぞれの大学では、いろいろな教育分野で、異なる教育目標、異なるカリキュラムで計算科学の教育を目指している。しかしながら、一部の大学を除いては、計算科学としての十分な教員・カリキュラムが用意できていないのが現状である。また、演習用の計算資源も十分とは言えない。そのために、それぞれの大学や教育機関で計算科学教育の充実を図るとともに、全国の大学の計算科学教育を補完する教育センター的な全国的教育機構が必要である。個々の大学でのカリキュラム改革は、本報告（3）教育カリキュラムで議論されているので、ここでは全国的な計算科学教育センターを提案する。

ここで提案する計算科学教育センター（仮称）に期待される機能は、以下の8点である。

1. 最先端の研究状況を広める。
2. 不足しているカリキュラムを補完する。
3. 演習環境を提供する。
4. シミュレーション相談室を整備する。
5. 単位認定や単位互換の制度を確立する。
6. 社会人教育の拠点となる。
7. 中学生、高校生への計算科学に対する理解増進を図る。
8. 民間企業、研究機関及び海外の大学との交流拠点となる。

#### 1. 最先端の研究状況を広める。

最先端の研究内容を1、2時間のセミナーという形式で、広く全国の学生に聴講の機会を与える。大学院の学生に対して、論文を読むだけではなく、最先端研究者によるセミナーによる講演と質疑応答により、研究成果のより深い理解と関心を与えるものである。全国の大学から容易に聴講するためには、TV 会議システム、放送の利用やインターネットによる多点での聴講システムが必要である。

#### 2. 不足しているカリキュラムを補完する。

それぞれの大学や専攻では、異なる教育目標を持ち、独自のカリキュラムで教育を行っている。そのカリキュラムには、不十分な教員数のため必要な講義が不足する。この不足講義を相互に補完するのが目的である。それぞれの大学から講師を派遣し、開講する講義

を全国の大学で受講することにより、不足する講義を補うことができる。この講義には、受講生を一カ所に集めて、1週間単位のシミュレーション・スクール形式の集中講義か、通常の授業形式でTV会議システムか、放送大学的な放送利用か、インターネットによる聴講ソフトを用いて遠隔地で受講可能にしなければならない。当然、これらの授業はe-Learning用コンテンツに編集し、必要あればどこからでも受講可能にすることが必要である。

### 3. 演習環境を整える。

計算機シミュレーションの手法を理解するのに、実際に計算機シミュレーション・コードを操作する演習が不可欠である。並列コンピュータの特性を理解し、自らシミュレーション・コードを作成し、計算結果を解析評価することは、計算機シミュレーションの本質を理解するために必要である。日本の大学が備える計算機は、往々にして研究用ジョブが優先され、教育用への利用に適さない構成となっている。教育用には、短時間ジョブが同時に多数走ることとなり、演習時間中には、研究用の長時間ジョブを止めざるえない。また教育に適したソフトウェアのインストールも必要となる。そのために、教育用のための並列計算機として、短時間ジョブを多数受け入れ、商用ソフトを含めて教育用に多くのソフトが走る環境を整え、全国の大学から簡易にアクセスできるシステムを提供することが重要である。さらに、計算機シミュレーションの結果を解析・評価するために必要な可視化技術の演習も重要と考える。

### 4. シミュレーション相談室を整備する。

演習のための計算機環境を整えるのと同時に、演習中に生じる問題・質問に的確に回答できるシミュレーション相談室の設置が求められる。この相談室では、演習のみならず計算機シミュレーションに関する一般の質問等に応ずる体制が必要である。初心者の導入質問から最先端研究の質問まで対応可能とし、計算機シミュレーションに関するあらゆる情報が提供できる相談室が求められる。この相談室の室員として、受講生自ら努め、他の受講生の質問に答えることにより、自らの研鑽を積むことができる。このシミュレーション相談室は、離れた場所から離れた場所へのコミュニケーション手段が必要であり、インターネットを通してウィンドー画面やデータの共有化、会話を行うシステムが必要となる。

### 5. 単位認定や単位互換の制度を確立する。

計算科学教育センターで開講するシミュレーション・スクール、講義や演習は、受講生の大学における単位としての認定や、また他大学で開講される講義を相互に単位互換できる制度を確立することが重要である。単位認定により、受講生の意欲が増進されることは明らかであり、計算科学教育センターで開講される講義が他大学から認められることを意味し、たいへん重要である。当然、開講する授業の内容、評価方法など単位認定に値するものに整備しなければならない。

### 6. 社会人教育の拠点となる。

対象を学部学生と大学院生に絞るのではなく、社会人にも講義や演習を公開し、企業に

計算機シミュレーションを普及させることは、たいへん重要と考える。計算機シミュレーションの手法を企業研究に取り入れることにより、成長産業の育成と基幹産業の競争力強化を目指すばかりではなく、計算機シミュレーションの技術を身につけた学生の活躍の場を広げることができる。社会人教育には、大学教育と異なり、受講生の持つ資質・知見に差があり、達成目標も受講生により異なる。受講生それぞれに適したテーラーメイド的なカリキュラム指導が大切である。

#### 7. 中学生、高校生への計算科学に対する理解増進を図る。

大学教育をより効果的なものにするために、中学生や高校生に計算機シミュレーションの面白さ、有用性を示し、理解増進を図る必要がある。柔軟な考え方をもち中高生に3次元可視化で表示された計算機シミュレーションの結果を示し、物理現象の考え方、もの作りへの活用や、社会現象などの予測などを理解させる教育方法は、これからの中高校での教育を改革するひとつの方法を提示するものであり、たいへん重要なものとする。(参考2)

#### 8. 民間企業、研究機関及び海外の大学との交流拠点となる。

計算科学に関する情報提供として、民間企業、国内研究機関及び海外の大学・研究機関との交流拠点の役割を担うことは重要である。特に、計算科学は多種多様な研究分野にまたがり、異分野の交流が新しい研究分野の育成に貢献するものとする。異分野の研究者が集い、議論する場が非常に重要である。この交流の場を供給することも計算科学教育センターの大きな役割とする。民間企業との交流として、学生のインターシップと共同研究の全国的な窓口があげられる。企業からのインターシップの申し出と、全国的な学生募集を担うことにより、大学生の活躍の場の拡大に繋がり、企業の計算機シミュレーションの有用性の理解を増すことができる。研究機関に関しても、企業と同様のインターシップの受け入れにより、研究機関の教育への貢献と同時に、最先端研究者の育成のために人材発掘が可能になると考える。これらの窓口を全国的に一本化することにより、個々の大学で担うよりはるかに効率的かつ効果的である。

以上の機能を持つ教育センターを構築し拠点化することにより、計算科学の普及、発展に大いに貢献できるものとする。そのためには、早急な計算科学教育センターの設置が望まれる。

## 4. キャリアパスと業績評価

計算科学分野の人材育成について検討するにあたっては、育成した人材のキャリアパスについて考慮することが必要である。一方、計算科学人材のキャリアパスは、研究者(大学、公的研究機関、民間企業)、技術者といった職種により多様であり、一概に述べることは出来ない。今我々が検討しなければならないのは、計算科学技術分野の人材が適切なキ

キャリアパスを歩むための環境整備を行うことであり、特に、当該分野の業績評価システムの確立が求められる。

業績の評価システムについては、研究者と技術者双方が正当に評価されるシステムが必要。特に、チューニング、ソフトウェア開発、データ処理等を行う技術者を評価する指標がなく、このような人材が評価されていない状況を変えることが必要である。具体的な取組としては、以下のようなものが考えられる。

- －HPC 技術者認定試験
- －研究支援人材に対する待遇や評価意識の改善
- －チューニング等に対する学問的評価の検討（論文における扱い等）

## 5. 結論

計算科学を支える HPC 人材を系統的に育成することは、早急に取り組むべき重要課題である。育成すべき人材としては、既存のプログラムを利用して計算科学の研究開発を行える人材のみならず、プログラムを開発できる人材も重要である。また、プログラムの開発・利用のための支援環境をつくる人材も今後ますます必要になる。

教育体系を考える上で重要なことは、計算科学が高度な学際性（異なる分野間、あるいは計算科学とコンピュータ科学との間）を持つことであり、どこに軸足を置くかで様々な教育パターンが考えられる。また、学部、大学院（修士、博士）、中学高校などさまざまな段階での教育体系を考える必要がある。本報告では、いくつかの実例を報告した。今後は、各教育機関だけでなく、連携して計算科学教育センター（仮称）を設置する方向も考えられる。

企業人の教育も重要な課題であり、教育機関と連携するとともに、企業でのイノベーション創出に資する問題解決型の人材育成が望まれる。

このように育成された人材がその能力を十分発揮できる環境整備が必要である。学界においてはプログラムの構築やコードチューニングが相応しく評価されなければならない。産業界においても、そのような人材を相応しく処遇することが期待される。このため、HPC 技術を認定する試験を行うことも考えられる。

今後、系統だった人材育成・教育システムの確立が急務である。



## HPC人材育成のための教育のあり方検討会について

## 1 設置の趣旨

現代の科学技術と産業において、大規模なシミュレーションはなくてはならないものである。一方、これを支える人材（HPC（High Performance Computing）人材）は、科学技術、産業のどちらの現場においても圧倒的に不足しており、その育成は急務である。しかしながら、我が国においてはHPC人材育成・教育のシステムが確立しておらず、有効な人材育成を行い得る体制が出来ていない。

一方、国は、次世代スーパーコンピュータの開発・整備を機に、我が国の計算科学技術推進体制を構築していくこととしており、その具体的な方策について文部科学省次世代スーパーコンピュータ戦略委員会において検討が進んでいる。このような中、我が国におけるHPC人材育成・教育システムを確立し、国において検討されている計算科学技術推進体制の中に組み込んでいくことは非常に重要なことである。このため、あるべきHPC人材育成・教育システムについて、その具体的内容にまで踏み込んだ検討を行うとともに、これをどのような体制で進めていくべきかについて検討を行うため、本検討会を設置する。

## 2 検討事項

我が国における系統的なHPC人材育成・教育システムの確立について

[ 具体的検討項目 ]

- ・ 大学における多様なレベルに応じた人材育成・教育プログラム
  - 対象学生と教育内容
  - 具体的カリキュラム
- ・ 人材育成・教育推進体制
  - オールジャパン体制の構築
  - 大学間、戦略機関との連携
  - 次世代スパコンの活用
- ・ 社会人・企業人教育

## 3 目標

上記検討事項について具体的方策を取りまとめ、高等教育局及び研究振興局の施策への反映も念頭におき、その実現を目指す。

## 4 構成及び運営

- ・ 構成員は別紙のとおりとする。（今後適宜増員あり）
- ・ 運営について、必要に応じ文部科学省研究振興局情報課が支援する。
- ・ オブザーバーとして高等教育局専門教育課が参画する。
- ・ 検討状況については、適宜次世代スーパーコンピュータ戦略委員会、情報科学技術委員会等に報告することとする。

## H P C 人材育成のための教育のあり方検討会 名簿

- |         |                                 |
|---------|---------------------------------|
| 青柳 睦    | 九州大学情報基盤研究開発センター長               |
| 伊藤 聡    | 東芝研究開発センター機能材料ラボラトリ<br>一研究主幹    |
| 宇川 彰    | 筑波大学副学長                         |
| 薄井 洋基   | 神戸大学学長補佐（第1回）                   |
| ○ 小柳 義夫 | 工学院大学情報学部長                      |
| 金田 行雄   | 名古屋大学工学研究科教授                    |
| 賀谷 信幸   | 神戸大学システム情報学研究科計算科学<br>専攻長（第2回～） |
| 佐藤 哲也   | 兵庫県立大学教授                        |
| 西尾 章治郎  | 大阪大学理事・副学長                      |
| 平尾 公彦   | 理化学研究所特任顧問                      |
| 松宮 徹    | 新日本製鐵株式会社顧問                     |
| 美濃 導彦   | 京都大学学術情報メディアセンター教授              |
| 米澤 明憲   | 東京大学情報理工学系研究科教授                 |

○：座長

（五十音順 敬称略）

## 審議経過

### 第1回 平成22年3月10日（水）

- ・ HPC人材育成のための教育のあり方検討会設置について
- ・ 学際計算科学・工学人材育成について（米澤明憲、中島研吾）
- ・ 神戸大学システム情報学研究科（薄井洋基）
- ・ 今後の検討事項

### 第2回 平成22年4月16日（金）

- ・ 今後の検討事項
- ・ 計算科学の学際性と求められる人材像（青柳睦）
- ・ 計算理工学専攻、21世紀COEプログラム「計算科学フロンティア」、計算科学連携教育研究センター（金田行雄）
- ・ 兵庫県立大学の新研究科（佐藤哲也）
- ・ 育成すべき人材像について

### 第3回 平成22年5月26日（水）

- ・ 今後の検討事項
- ・ 革新的ハイパフォーマンス・コンピューティング・インフラ（HPCI）とこの構築を主導するコンソーシアムのグランドデザイン
- ・ 産業界での人材育成について（伊藤聡）
- ・ 計算科学領域での人材育成（美濃導彦）
- ・ 「大学連合による計算科学の最先端人材育成」の活動報告（賀谷信幸）
- ・ 育成すべき人材像について

### 第4回 平成22年6月21日（月）

- ・ これまでの議論のまとめと提言に向けた検討

### 第5回 平成22年7月6日（火）

- ・ これまでの議論のまとめ
  - 応用系のカリキュラム（青柳睦、賀谷信幸）
  - 情報系のカリキュラム（中島研吾）
  - 企業のカリキュラム（伊藤聡）
  - 欧米の事例と名古屋大学の事例（金田行雄）

### 第6回 平成22年8月11日（水）

- ・ これまでの議論のまとめ