

究極の「並列プログラミング教育」を目指して

地球惑星科学専攻での4年間と「学際計算科学・工学 人材育成プログラム」

中島研吾

東京大学情報基盤センター

(前 大学院理学系研究科地球惑星科学専攻)

1. はじめに

「計算科学」が「理論」、「実験」に続く「第三の科学」と呼ばれるようになって久しい。「計算科学」は昨今の並列計算機ハードウェア、ソフトウェア分野の研究開発の進歩と相俟って、発展を続けて来た。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。

筆者は2004年4月から2008年3月まで、本学理学系研究科地球惑星科学専攻に所属し、並列プログラミング教育に関連した講義・演習である「並列計算プログラミング」、「先端計算機演習I・II」¹を担当してきた。これらの講義・演習は、地球惑星科学専攻で2003年度～2007年度に実施された、東京大学21世紀COEプログラム「多圏地球システムの進化と変動の予測可能性(観測地球科学と計算地球科学の融合拠点の形成)」(以下「多圏地球COE」²)の一環として実施された。

筆者が担当した上記の講義・演習のうち、特に情報基盤センターのHitachi SR8000/11000を使用して実施した「先端計算機演習II」については、「スーパーコンピューティングニュース」でも何回か採り上げていただいた[1～6]。

本稿では、地球惑星科学専攻における4年間の並列プログラミング教育の成果、反省点について述べる。この経験を踏まえて、本学における全学的な教育プログラムとして2008年初頭から検討されている「学際計算科学・工学 人材育成プログラム」のアクティビティについても触れ、並列プログラミング教育のあるべき姿について述べる。

2. 地球惑星科学専攻における並列プログラミング教育

(1) 背景

2004年4月に地球惑星科学専攻に赴任して、講義を開講するにあたって(そもそもの仕掛人であるところの)松浦充宏教授に言われたことは、『地球シミュレータ』³を使いこなせるような人材を育ててよ。」というただ一言であった。個人的な話で恐縮だが、当時筆者は大学を卒業して20年近く会社員生活を続けていた。「地球シミュレータ」をターゲットとした並列有限要素法プラットフォーム「GeoFEM」⁴というプロジェクトをやっていたので、並列プログラミングの講習会の経験はあったが、学生の教育は未経験であった。また、当時、計算機科学を専門

¹ <http://nkl.cc.u-tokyo.ac.jp/class/>

² <http://www.eps.s.u-tokyo.ac.jp/jp/COE21/>

³ <http://www.es.jamstec.go.jp/>

⁴ <http://geofem.tokyo.rist.or.jp/>

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・演習は日本はもちろん、世界的にもまだ例が無く（現在も同じ状況であるが）、参考になるような文献もほとんど無かった。こうした世界的状況の中、地球惑星科学専攻において「『世界初』の並列プログラミング教育」が始められた背景には、2002年から稼動を開始していた「地球シミュレータ」の存在が大きいことは上記の松浦教授の一言からもうかがえる。

ほとんど手探り状態で準備を始めたのだが、当時の記録によると4月21日には講義を開始している。夏学期2コマ（並列計算プログラミング、先端計算機演習I）、冬学期1コマ（先端計算機演習II）という時間割りは決まっていたこともあり：

- 夏学期：MPI（Message Passing Interface）、冬学期：OpenMPを中心に教える
- 並列計算ライブラリ等の文法・利用法よりはむしろ、SPMD（Single Program Multiple Data）の考え方、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ための様々な工夫をすることの重要性、を「伝える」ことに主眼を置く

という基本方針のもとにスタートした。後者については、自分自身が「並列プログラミング」を習得した経験から最も重要と考えていることであり、「並列プログラミング教育」の根幹をなす考え方である。元のプログラムの中身、アルゴリズムの特性をよく理解していれば、それを並列化することはさほど難しいことではないのである。

最初の講義で特に強調したのは以下のような点である。4年間、多少言い方は変化したものの、精神としては変わらない：

- ① 並列計算プログラミングは決して難しくない
- ② 重要なのは背景にある物理、基本となるアルゴリズム、その上でのプログラミングである（後で述べるようにプログラミングはやっぱり大切なのであるが）
- ③ 計算機に使われてはいけない
- ④ 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

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 Computer Science 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⁵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。本当にこの順番に重要であるという考え方もできるし、これだけ幅広い分野を扱わなければならないということも言える、中々含蓄のある表現である。この言葉に出会ったのは2005年2月で最初の年度（2004年度）の終わる頃であった。2005年度以降は「並列計算プログラミング」、「先端計算機演習I・II」で主としてカバーするのは「SMASH」のうちの「ASH」、すなわち Algorithm の一部と Software 全般、Hardware の一部と位置づけて講義・演習を実施した。結果的には最初の「S」を除いて、「MA」くらいまでは教えることになってしまったが。

⁵ <http://www.math.umd.edu/~lvrmr/>

(2) 地球惑星物理学科，地球惑星科学専攻における計算機・プログラミング教育

筆者が所属していた，理学部地球惑星物理学科，理学系研究科地球惑星科学専攻は，計算機・プログラミング教育という点では，以下に述べるような非常に充実したプログラムを有している（これら以外にも「地球物理データ解析」，「時系列データ解析」等の講義もあり，これらに先立って教養課程では「情報」で，基礎的な計算機リテラシーが教えられている）：

- 地球惑星物理学演習（3年夏）：計算機リテラシー，FORTRAN 文法，基本的な数値解析
- 地球物理数値解析（4年夏）：偏微分方程式の数値解法
- 並列計算プログラミング，先端計算機演習 I・II（大学院）

「並列計算プログラミング」，「先端計算機演習 I・II」については次節で説明する。

「地球惑星物理学演習」は3年夏学期に週3回，各2コマ（3限，4限）を使って行われる「計算機漬け」の演習である（合計84コマ）。学習する項目は以下の通りである：

- 計算機リテラシー 16 コマ（8回）
- FORTRAN 文法 20 コマ（10回）
- 時間発展微分方程式 16 コマ（8回）
- 行列・連立一次方程式 16 コマ（8回）
- データ解析 16 コマ（8回）

受講者は35名程度で，1人1台，専用のUNIX端末が割り当てられる。各項目を担当する教員の他，必ず2～3人のTAがつく。TAは修士課程の大学院生が中心で，自身が3年夏学期にこの演習を体験している。週6コマが計算機・プログラミング関連の講義・演習に費やされる。中にはこれで計算機嫌いになる学生もいるらしいが，TAをやるような学生は「受講して良かった！」と感じているのであろう。1980年代にはこの演習のスタイルは確立していたそうである。この「地球惑星物理学演習」の凄まじさは是非一回演習のWEBページ⁶をご覧ください。教材も理解しやすいようによく工夫されており，非常に充実している。

「地球物理数値解析」は偏微分方程式の数値解法のアルゴリズムとプログラミング関する，週1コマの講義である。2005年度以降は，3人の教員によって以下の3項目を実施している：

- 差分法（I）：一次元，双曲型方程式
- 差分法（II）：多次元，楕円型方程式
- 有限要素法

各項目で4～5回の講義（うち1回は実習）が実施される。筆者は2005年度からこの「地球物理数値解析」の「有限要素法」の部分も担当した⁷。「地球物理数値解析」は大学院との共通講義であり，40名程度の受講者のうち20名は4年生，残り20名は大学院生（他大学から進学した修士課程の学生が多い）であった。

⁶ <http://www-geoph.eps.s.u-tokyo.ac.jp/ta/resume2008/index.php>

⁷ <http://nkl.cc.u-tokyo.ac.jp/07u/>

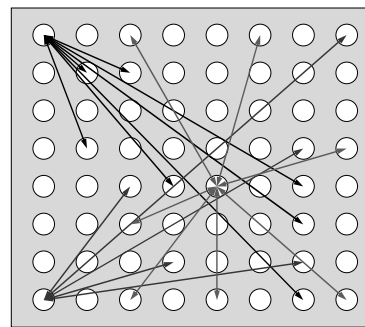
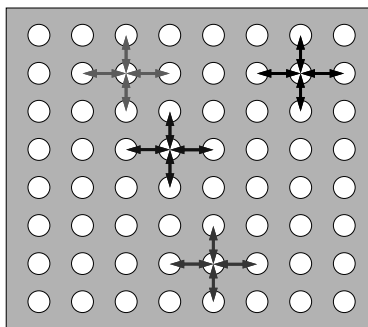
(3) 並列計算プログラミング, 先端計算機演習 I・II

本節では、2004 年度から開始した「並列計算プログラミング」, 「先端計算機演習 I・II」について紹介する。夏学期は水曜の 3 限が「並列計算プログラミング」, 4 限が「先端計算機演習 I」と時間が続いていたので、2 コマ分の講義を続けて実施することが多かった。原則として両者を併せて履修することとした。2. (1) でも述べたように、MPI (Message Passing Interface) を使って、差分法、有限要素法、境界要素法等によるアプリケーションを並列化する技術を身につける、ことが目標であった。表 1 は 2007 年度夏学期の講義内容である。

表1 2007年度夏学期 講義内容 (並列計算プログラミング, 先端計算機演習I)

	日付	内容
A01	4月11日(水)	授業および演習の概要, High-Performance Computing (HPC) の概要
A02	4月18日(水)	Grid Computing について 数値解析手法の基礎 (Gauss-Seidel, SOR 法)
A03	4月25日(水)	MPI によるプログラミング概要 (1)
A04	5月02日(水)	MPI によるプログラミング概要 (2), 課題 S1 出題
A05	5月09日(水)	MPI によるプログラミング概要 (3), 課題 S2 出題
A06	5月16日(水)	線形ソルバー, 課題 S3 出題
A07	6月06日(水)	課題 S1, S2 解説
A08	6月13日(水)	可視化, チューニング
A09	6月20日(水)	並列アプリケーション開発法入門 (I) 有限体積法, 課題 S3 解説
A10	7月04日(水)	並列アプリケーション開発法入門 (II) 有限体積法: 並列データ構造, 領域分割
A11	7月11日(水)	並列アプリケーション開発法入門 (III) 有限体積法: 並列可視化, 課題 P1 出題
A12	7月18日(水)	並列アプリケーション開発法入門 (IV) 粒子間熱伝導解析コード並列化, 有限要素法の高速度

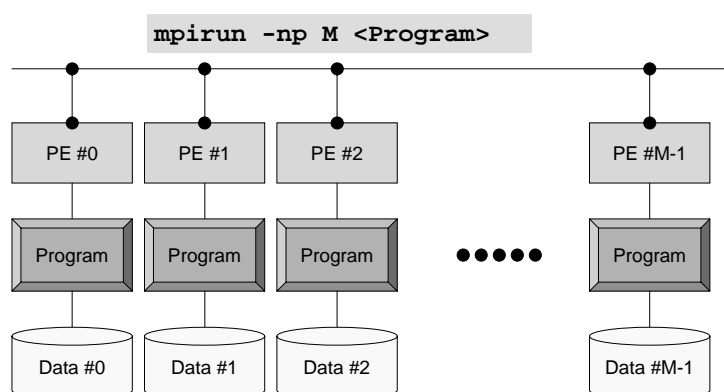
MPI には 400 以上の関数があるが、実際、科学技術シミュレーションで必要になるのは 10 程度である。本講義では、まず、科学技術シミュレーションの手法を図 1 に示す 2 種類 (局所的な手法、大域的な手法) に分類した。これらの手法は連立一次方程式の解法、並列化の手法が図 1 に示すように大きく異なっている。局所的な手法は空間的に近接した領域とのみ相互作用があるため、並列計算時の通信も 1 対 1 通信 (point-to-point communication) が中心である。大域的な手法はグローバルな相互作用があるため、グループ通信 (collective communication) を多用する。



- 局所的な手法 (差分法, 有限要素法等)
 - 疎行列, 1 対 1 通信 (point-to-point)
- 大域的な手法 (境界要素法, スペクトル法等)
 - 密行列, グループ通信 (collective)

図 1 科学技術シミュレーション手法の分類

講義では、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，できるだけ実習によって経験を積んでもらうことにした。MPI の基本的な考え方である SPMD（Single-Program Multiple-Data）（図 2）を実際のアプリケーションの並列化を通して学ぶことに主眼を置いた。SPMD の考え方は計算機科学専攻の学生でも中々理解するのが難しく，教育にあたって苦労するところなのだそうであるが，アプリケーションという具体的な対象があると逆に理解が進むようである。講義の中では，アプリケーションの中身をよく理解し，並列化するための局所データ構造をどのように考えるか，というような話題が中心であった。表 1 によると，MPI に関する講義は 3 回実施されているが，ほとんどの時間は局所データ構造の考え方の説明に費やした。アプリケーションの中身，アルゴリズムの特性，をよく理解していれば，「並列化」というのはそれほど難しいことでは無い，ということ伝えることにはある程度成功したと考えている。

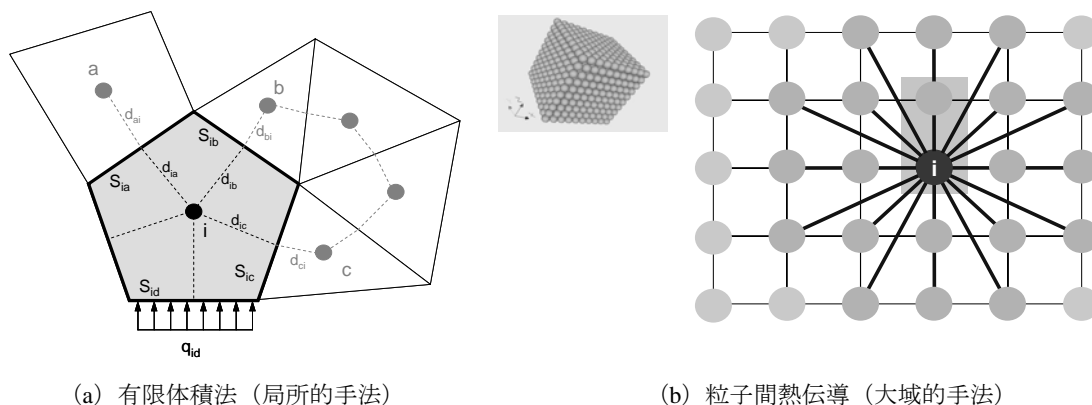


各プロセスは「同じことをやる」が「データが違う」
大規模なデータを分割し，各部分について各プロセス（プロセッサ）が計算する

図 2 SPMD (Single-Program Multiple-Data) の考え方

いわゆる「並列化」に関係した講義のほか，「Grid Computing」のような先端的なトピック，「線形ソルバー」，「可視化手法」，「チューニング」等についても講義，実習を行った。

「チューニング」については，キャッシュブロッキング，ループアンローリング等スカラープロセッサ向けのテクニックの他，「地球シミュレータ」を念頭において，ベクトルプロセッサ向けのチューニングについても講義した。結果的に講義・演習全体で「SMASH」の「MASH」はほぼカバーするような内容になっている。並列化のためのターゲットアプリケーションとしては，図 3 に示すような有限体積法（局所的手法），粒子間熱伝導（大域的手法）を扱った。



(a) 有限体積法（局所的手法）

(b) 粒子間熱伝導（大域的手法）

図 3 並列計算プログラミング，先端計算機演習 I で扱ったアプリケーション

プログラミングの実習のために、AMD Opteron プロセッサ (1.8GHz) 32 台 (16 ノード) から構成される PC クラスタを導入した。また、可視化ツールとして「MicroAVS (KGT 社製)」⁸を導入した。WEB 環境を有効に活用し、連絡事項の伝達、教材配布のほか、並列計算用ユーティリティプログラムのオンラインマニュアル等にも利用した (図 4)。

プログラミング教育にあたっては実習が重要である。特に講義時間中に説明するプログラム類についてその場で動作を確認することは更なる学習意欲を喚起するという意味でも重要である。本専攻には大学院生向けの計算機室、端末室がないため (学部生向けには 2. (2) でも触れたように専用の端末室がある)、初年度 (2004 年度) は講義時間中に教員がデモンストレーションを実施するのみであったが、2005 年度からは情報基盤センター (本郷) 大演習室を使用した講義・演習を実施した。大演習室では各自 1 台の端末を利用できるため、特に MPI プログラミング実習の初期においては非常に有効であった (図 5)。



図 4 講義・演習用ウェブサイト



図 5 情報基盤センター大演習室での実習風景

地球惑星科学専攻の学生に FORTRAN ユーザーが多い (2. (2) で紹介した「地球惑星物理学演習」でも FORTRAN を学習している) こともあり、プログラミング言語は FORTRAN 中心であったが、他専攻、他研究科 (物理、天文、化学、工学系、新領域等) の受講者が次第に増えてきたこともあり、C 言語で書かれたサンプルコード、模範解答も用意する等配慮した。

評価は、以下のような課題実習レポート：

- 課題 S1 : MPI の基本的な機能 (台形法則等, グループ通信)
- 課題 S2 : Gauss-Seidel 法等による一次元熱伝導方程式ソルバーの並列化 (1 対 1 通信)
- 課題 S3 : CG (Conjugate Gradient) 法による一次元熱伝導方程式ソルバーの並列化
- 課題 P1 : 有限体積法による三次元熱伝導方程式ソルバーの並列化

の他、毎週「ログ」を提出してもらって 10% 程度配点した。これは、毎週本講義・演習のために各学生がどのくらいの時間を割いているかを把握するためのもので、講義の進み具合の調整の参考にすることを意図していた。

冬学期は「先端計算機演習 II」を担当した。夏学期講義・演習に引き続き、並列化、ベクトル化において必須のプログラミング技術であるオーダリング、共有メモリ型並列計算機における並列化に有効な OpenMP、多重格子 (Multigrid) 法に関する講義・実習を、夏学期に開発し

⁸ <http://www.kgt.co.jp/feature/microavs/>

た有限体積法プログラムを使用して実施した。特に ICCG 法によるポアソン方程式ソルバーを OpenMP によって並列化する際に問題となる、「データ依存性」を回避するための「オーダリング」手法に関する講義・演習に主眼を置いた [4,5,6]。表 2 に講義内容を示す。

2004 年度は情報基盤センターの Hitachi SR8000/128 1 ノード (8CPU) を受講者に利用させて頂いたが、2005 年度からは新しく導入された Hitachi SR11000 1 ノード (8CPU) を利用した。夏学期に引き続き、情報基盤センター (本郷) 大演習室を使用した。

評価は夏学期と同様課題実習レポートとログに基づいて実施した。レポート課題は以下の 3 課題である：

- 課題 W1：有限体積法によるポアソン方程式ソルバー：オーダリングの適用
- 課題 W2：有限体積法によるポアソン方程式ソルバー：OpenMP による並列化
- 課題 W3：有限体積法によるポアソン方程式ソルバー：不均質場における多重格子法

表2 2007年度冬学期 授業内容 (先端計算機演習II)

	日付	内 容
B01	10月03日(水)	授業および演習の概要, スケジュール
B02	10月10日(水)	1.1:有限体積法によるアプリケーション
B03	10月17日(水)	1.2:ICCG法について
B04	10月24日(水)	1.3:OpenMPについて
B05	10月31日(水)	1.4:オーダリングについて(1)
B06	11月07日(水)	1.4:オーダリングについて(2), 課題W1, 課題W2出題
B07	11月14日(水)	2.1:マルチグリッド理論的背景
B08	11月21日(水)	2.2:3次元ポアソン方程式ソルバー
B09	11月28日(水)	2.2:3次元ポアソン方程式ソルバー(続き)
B10	12月12日(水)	課題W-1解説
B11	12月19日(水)	2.3:Semi-Coarsening, 課題W3出題
B12	01月16日(水)	課題W-2解説, 実習
B13	01月23日(水)	演習
B14	01月30日(水)	2.4:代数的マルチグリッド, 課題W-3解説

(4) 成果, 問題点等

2. (3) で述べたような教育内容は最初から確立していたわけではなく、4年間で様々な変遷があった。最初は2. (2) で述べたような学科・専攻の講義体系に関する予備知識も無く、また「並列計算プログラミング」, 「先端計算機演習I・II」の受講予定者は博士課程の学生が中心という風に聞いていたので、「SMASH」で言うと後半の「SH」に重点を置き、「A」は少々という感じで、比較的テクニック習得を重視したスタイルをとっていた。

初年度(2004年度)夏学期はM1からD3まで40名上の受講者があった(最終的に単位を取得したのは20名程度)。今から思うと、かなり高度で盛りだくさんな内容であったが、講義時間中の質疑応答等も活発で「手応え」もあった。冬学期は10名程度に受講者は減少したものの、夏学期と同様に手応えを感じることができた。

2005年度以降は、受講者は夏学期15~20名程度、冬学期5~10名程度という人数が定着した。2004年度と変わったのは：

- 2. (3) でも述べたように他専攻，研究科からの受講者が増加した（25%以上）
- 受講者のほとんどが M1 となった

ということである。

地球惑星科学専攻では，修士課程入学者の半分以上は学外からの進学者であり，学部時代に 2. (2) で述べた「地球惑星物理学演習」のような充実した計算機，プログラミング教育を受けている者は稀である。2005 年度から「地球物理数値解析（4 年生，大学院共通）」を担当するようになったが，この講義と「並列計算プログラミング」，「先端計算機演習 I・II」を両方同時に受講している学生も少なからずあり，例えば「有限差分法の精度」，「Gauss-Seidel 法」というような 2004 年度は既習事項として扱っていたことについても知らない学生の割合が多くなっていることがわかった。2004 年度は博士課程の学生が多かったこともあって，このような問題は表面化しなかったのであるが，M1 の受講者が増えたことを考慮して，2005 年度途中からモデリングや基礎的なアルゴリズムもその都度教えるというスタイルに徐々に変化していった。すなわち「SMASH」のうち「ASH」から「MASH」に本格的に重点を置くようになった。

地球惑星科学専攻の場合，大学院に入学すると教育はかなり専門化する。「並列計算プログラミング」，「先端計算機演習 I・II」には様々な分野の学生が受講するため，題材とするアプリケーションの選択には非常に気を遣った。受講者は年度によって変遷はあったが，大気海洋，宇宙惑星，地球惑星システム，固体地球，地球生命圏の 5 つの大講座のうち，半分以上が大気海洋，次いで固体地球が多かった。アプリケーションとしては，熱伝導，ポアソン方程式等すべての分野で共通に扱うものを題材とした。

筆者自身は「有限要素法」を「局所的手法」の例として扱いたかったのであるが，それには「有限要素法」そのものを教えなければならないということもあり，比較的予備知識が少なく済む「有限体積法」を扱った。また「大域的手法」は当初は「境界要素法」であったが，2006 年度以降はアルゴリズムを理解しやすい「粒子間熱伝導」に変更した。

課題レポートについても，当初は受講者が全部自分でプログラムを作ることを期待していたのであるが，受講者に様々なレベルがあることも考慮して，2005 年度以降は課題の模範解答の説明を丁寧に行うことにした。評価に差はつけるが，必ずしもすべて自力でやる必要はなく，解説を聞いてからプログラムを作成してもよいというように条件を緩和した。筆者の解説に従って受講者が模範解答を「読解」するスタイルが中心となったが，M1 の受講生が多かったことを考慮すると，これは結果的には実用的プログラミング技術の習得という点では効果的であったと思う。もちろん，全て自力で課題をこなす強者も毎年少なからず居たのであるが。

4 年間の「並列プログラミング教育」の効果について知ることは難しい。「アプリケーションの中身，アルゴリズムの特性をよく理解していれば，並列化はそれほど難しいことでは無い」ことが図 2 で示した SPMD の概略とともに受講者の心に残っていれば最低限の目標を果たしたと言えるが，「地球シミュレータを使いこなす人材を育てる」という点では，あまり期待に応えられなかったかも知れない。特に M1 の受講者が多かったこともあり，実際に研究に適用する等して効果が出るには時間がかかる。受講者の感想によると「将来の役に立つと思う」という暖かい言葉が多数であったが・・・。

2008 年 2 月 28 日に，多圏地球 COE における計算機・プログラミング教育の総括のために，

「HPC 教育に関するワークショップ」⁹を開催した。ワークショップでは、これまでの受講者を代表して、大石裕介氏（現 富士通）、松村義正氏（地球惑星科学専攻博士課程、気候システム研究センター）の2名に講演をお願いしたが、2名とも2004年度（初年度）の受講者である。開始から4年経過して成果が現れ始めたということもできる。今後の経過を見守る必要がある。教育プログラムの成果をあげるためには色々な意味での「継続」は重要である。講義・演習を続けるだけでなく、M1のうちに講義・演習で「並列プログラミング」の基礎を学び、博士課程に進んでそれを自分の研究に適用する、という理想的なサイクルを確立するための「メンテナンス」、「アフターケア」も忘れてはならない。プログラミングの知識は実際の問題に応用してこそ意味があるものなのである。

4年間の任期を終えて、筆者は地球惑星科学専攻を離れることになったが、幸いなことにごく近所への異動であったし、また「並列計算プログラミング」、「先端計算機演習」は夏季集中講義として継続することになった¹⁰。教育内容の一層の充実を図るとともに、これまでの受講者が受講内容を研究に役立てられるようサポートして行きたいと考えている。

地球惑星物理学科、地球惑星科学専攻の提供する計算機・プログラミング関連の講義・演習は非常に充実しており、他に類を見ない高いレベルである。一つだけ残念なのは、各講義・演習間の連携が不十分なことである。これについては次章で検討する。

3. 「学際計算科学・工学 人材育成プログラム」構想

(1) 背景

理化学研究所「次世代スーパーコンピュータ」¹¹の開発等を背景に、大規模並列シミュレーションへの期待は、産学において更に高まっている。また筑波大、東大、京大による「T2K オープンスパコン」¹²が2008年6月より運用を開始する。

本学では2008年初頭から、「T2K オープンスパコン、次世代スーパーコンピュータ等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」を育成するための全学的な教育プログラムとして「学際計算科学・工学 人材育成プログラム」構想が検討されている。平尾公彦副学長をヘッドとして、理学系研究科、工学系研究科、情報理工学系研究科、新領域創成科学研究科、生産技術研究所、情報基盤センターの協力によって作業が進められており、筆者も情報基盤センターの一員として参画している。

本構想の詳細については、片桐孝洋准教授（情報基盤センター）の講演資料〔7〕を参考にさせていただきたい。本章では主に、有限要素法、差分法等の連続体シミュレーションを念頭において、「学際計算科学・工学 人材育成プログラム」の概要について述べる。

(2) 検討の現状

現在は、各学科、専攻における講義、演習の現状を調査しつつ、2. で述べた、地球惑星物理学科、地球惑星科学専攻で実施されている講義・演習をモデルとして、人材育成プログラムのガイドラインを策定中である。2008年度冬学期から試験的な講義・演習を開始し、2009年度からの実施を目指している。

⁹ <http://nkl.cc.u-tokyo.ac.jp/seminars/0802-HPcedu/>

¹⁰ <http://nkl.cc.u-tokyo.ac.jp/08s/>

¹¹ <http://www.nsc.riken.jp/>

¹² <http://www.open-supercomputer.org/>

「学際計算科学・工学 人材育成プログラム」では科学・工学分野の学生が並列プログラミングを習得し、大規模シミュレーションができるようにしよう、というのが主たる目標なのであるが、それを支援する並列シミュレーションコード開発基盤、ライブラリ等を考慮すると、アプリケーションに関する知識を持った計算機科学分野の研究者の育成も重要である。「学際計算科学・工学 人材育成プログラム」では、そのような教育プログラムも検討中であるが、以下では主に「科学・工学分野向け」のプログラムについて述べる。

「並列計算機を駆使した大規模シミュレーションを実施」するためには、「SMASH」という言葉からもわかるように、様々な分野にわたる知識と経験が必要となるため、学部初年度よりの連携した教育の積み重ねが重要である。「SMASH」の様々な項目を（必要に応じて）体系的に教育することが「学際計算科学・工学 人材育成プログラム」の基本的な考え方である。本人材育成プログラムでは受講者を以下の4つのタイプに分類している [7]：

- A 型 大規模シミュレーションコードを線形ソルバーなどの基礎的なライブラリを除いて全て自分で開発する、「SMASH」の全てをカバー
- B 型 大規模シミュレーションコードを HPC-MW¹³ [8] や HEC-MW¹⁴ [9] 等の並列シミュレーションコード開発基盤を使用して開発する、「SMASH」の「SMA」
- C 型 商用コードを含む大規模並列アプリケーションのユーザー、「SMASH」の「S」または「SM」
- S 型 A 型に加えて、更に HPC-MW, HEC-MW 等並列シミュレーションコード開発基盤の整備に貢献できる人材

C 型はやや特殊であるが、A 型・S 型、B 型、C 型の各タイプのニーズに応じて必要な教育プログラムを提供することを予定している。A 型・S 型、B 型の受講者が博士課程で「並列計算機を駆使した大規模シミュレーション」により研究を推進するためには、大学院修士課程までに「並列プログラミング」を習得しておくことが望ましい。従って、学部レベルでの教育の充実は非常に重要である。

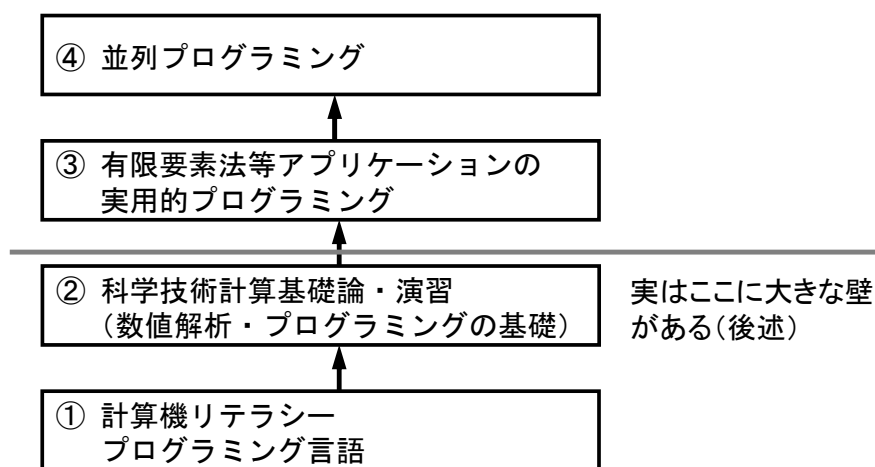


図 6 並列プログラミングへの道のり（連続体シミュレーションの場合）

¹³ <http://www.ciss.iis.u-tokyo.ac.jp/rss21/theme/multi/heckernel/>

¹⁴ <http://www.ciss.iis.u-tokyo.ac.jp/fsis/theme/hpc/index.html>

図6は、2. で述べた、地球惑星物理学科、地球惑星科学専攻で実施されている講義・演習をモデルとして、有限要素法、差分法等の連続体シミュレーションにおいて、「並列プログラミングへ」を到達点とした場合の「道のり」である。

①から④まで「密接に連携した」4つのレベルがあり、2. で述べた講義・演習とほぼ以下のように対応している（「C型」以外は①～④を一通り学習することが望ましいと考えている）：

- ①：地球惑星物理学演習（3年夏）前半：学部2年～3年の履修が望ましい
- ②：地球惑星物理学演習（3年夏）後半：学部3年
- ③：地球物理数値解析（4年夏）：学部4年，遅くとも修士1年
- ④：並列計算プログラミング，先端計算機演習（大学院）：修士1年～2年

教養課程でも「情報」という講義で一般的な計算機リテラシーを学ぶが、①はよりプログラミングに近いものを想定している。①については既に各学科で行われているので、基本的にはガイドラインを作成するのみにとどめる予定である。

また①の前の段階として、教養課程における全学ゼミとして「大規模シミュレーション入門」のような題目で実施し、オムニバス形式で各分野における先端的シミュレーションについて学内外の研究者に紹介してもらうことも検討している（後掲図7参照）。

②「科学技術計算基礎論・演習」は一般的な数値解析アルゴリズム，プログラミングの他，「SMASH」の「H」を低学年のうちから意識させるためにチューニングに関する項目も取り入れた以下のような内容を想定している（須田礼仁准教授（情報理工学系研究科）作成）：

- C言語（またはFORTRAN）プログラミング入門
- 浮動小数点数と丸め誤差，乱数とモンテカルロ法，非線形方程式，多項式，常微分方程式，積分，行列計算，FFT
- 高性能計算入門（アーキテクチャと計算性能，MPI超入門）

夏学期または冬学期中に15週実施し，各回1コマの講義，引き続いて1コマの演習というスタイルで実施する予定である。

③は実用的なアプリケーションのプログラミング（非並列），④は，③の並列化に関する講義・演習である。現状は，①，②に（ある程度）対応した講義，演習は各学部で行われているが，③，④は非常に少なく，特に④については，3. で紹介した地球惑星科学専攻の他，片桐准教授（情報基盤センター）が工学部・工学系研究科で担当している計算機科学分野向けの「スパコンプログラミング」¹⁵があるのみである。

③に関連した既存の講義・演習は，有限要素法等について，理論やアルゴリズムを講義することが中心で（「SMA」），実際にプログラミングまで立ち入って講義や演習で体系的に扱う（「SMAS」）ということはほとんどなされていない。実用的なアプリケーションプログラムに触れるのは，研究室へ配属されてからというのが現状である。これでは，新しいアプリケーションプログラムを自力で開発したり，並列化したりということは一般的に非常に困難である。

②のような講義を受講してLU分解等の基本的な数値解析のプログラムを作ることができ，

¹⁵ <http://www.cc.u-tokyo.ac.jp/publication/news/VOL10/No2/200803katagiri.pdf>

有限要素法のアルゴリズムを知っていても、実用的なアプリケーションプログラムが作れるわけではなく、図 6 にも示したように、②から③の間には実は大きな壁がある（「SMA」と「SH」の間の壁とも言える）。例えば、有限要素法では、問題を線形化して得られる連立一次方程式の係数行列が 0 の多い「疎行列」であり、CRS（Compressed Row Storage）[10] 等非ゼロ成分のみ格納する方式が使用される。②のレベルでは行列の全成分を格納する方式が通例であるため、③相当の講義で理論・アルゴリズムのみしか教えないと、CRS のような「一見些細なことであるがとても重要な」項目は抜け落ちてしまう。

これまで繰り返し述べて来たように、「アプリケーションのアルゴリズムの特性、中身（＝プログラミング）をよく理解していれば、並列化はそれほど難しいことでは無い」ということを考慮すると、③のプロセスが最も重要でありながら、体系的な教育が不足している部分である。

(3) 実用的プログラミングの教育

「プログラミングは経験が重要」と言われている。この言葉は正しいが、最低限の知識は必要である。最良では無いかもしれないが、効果的で着実な方法は、実際のアプリケーションプログラムを「読む」ことである。教員には非常に負担がかかるが、高等学校の古文、漢文の授業のようなスタイルが実は効果的なのである。

実用的なプログラミングの教育は本当に難しい。本学のある工学系の教員によると：

- 限られた講義時間の中では、理論、アルゴリズムを教えるので手一杯
- 専門書の巻末のプログラム例をもとに自作のアプリケーションプログラムを作るような学生もいるが、ごく稀である

ということであった。

筆者はこれまで「地球物理数値解析」、「並列計算プログラミング」、「先端計算機演習 I・II」の中で、アプリケーションプログラムを「読む」というスタイルでの教育を行って来ており、短時間でアルゴリズムとプログラミングを教育する場合には効果があると感じている。「地球物理数値解析」では、「有限要素法」のアルゴリズムとプログラミングの教育に 5 コマ（7.5 時間）が割り当てられていた。差分法との違いを理解してもらうために、どうしても二次元、双一次アイソパラメトリック四辺形要素まで勉強しても教えたい、という野望を抱いていた。わずか 5 コマという限られた時間の中では、アプリケーションプログラムを読みながらアルゴリズムを理解する方法（「SMASH」の「AS」）が効果的であった¹⁶。前処理付き共役勾配法、前述の疎行列格納法等についてもこの 5 コマの中でしっかり教えることができた。「もっと理論的なことを教えてほしかった」という受講者もいたが。

この③のプロセスが最も重要なところなので、他の方法も含めて重点的に検討して行きたい。教育経験のある方のご意見も伺いたいところである。

(4) 柔軟な対応：教材の充実

現在は主に、学期を通して実施する講義・演習を想定したガイドライン、カリキュラムを想定しているが、様々なケースに柔軟に対応できるように：

¹⁶ <http://nk1.cc.u-tokyo.ac.jp/07u/FEM-005.pdf>

- 集中講義，講習会
- E-learning
- 教科書

等，多様な形態のプログラムを充実させていく必要がある。特に図 6 の④に相当する講義・演習はむしろ集中講義，講習会形式で実施するのが適していると考えられる。

「学際計算科学・工学 人材育成プログラム」の策定にあたっては，なるべく受講者の負担が増えないように配慮する方針である。すなわち，既存の講義・演習で図 6 の①～④に類似したものがあれば，それで代用するかまたはガイドラインに合わせて講義・演習内容を調整する。

①レベルの講義・演習については既にほとんどの学科で実施されている。

調整が難しい場合は，集中講義，講習会等のプログラムを充実させ，短期集中的に学習したり，E-learning，教科書等の教材による自習を可能とすることも必要である。特に図 6 の①，②レベルの講義・演習はどの分野でも必要なものであり，需要も多いことから E-learning のような教材を充実させることは重要である。また，特に「B 型」，「C 型」の受講者向けには，アプリケーション，開発基盤，ライブラリの講習会をできるだけ頻繁に開催する必要がある。

2. (3) でも紹介したように，本学の場合，他大学から大学院に進学する学生も増加している。図 6 の①～③レベルの講義・演習は学部レベルで実施されることから，他大学からの進学者が無理なくキャッチアップするためにも，講習会，E-learning，教科書等，様々なオプションを提供することは重要である。

また「学際計算科学・工学 人材育成プログラム」修了者は，自ら大規模シミュレーションコードの開発に取り掛かるわけであるが，教育の成果を研究開発に生かすためには 2. (4) で述べたような「アフターケア」についても検討していく必要がある。「並列プログラミング」における「生涯教育」と呼ぶべきか。

図 7 は「連続体シミュレーション」について，「学際計算科学・工学 人材育成プログラム」が提供する様々な教育プログラムと各学科，専攻の既存の講義・演習の関係について示したものである。丸付き数字は図 6 の①～④に相当する。前述した「計算機科学分野向け」の講義・演習も含まれている。「学際計算科学・工学 人材育成プログラム」側に記載してある講義・演習は新たに開講する可能性もあるが，前述の方針もあり，既存の講義・演習を現在策定中のガイドラインに沿って調整することで「モデル講義・演習」としていくのが現実的である。

図 7 中の「並列プログラミング I・II」は：

- 「I」 「スパコンプログラミング」(片桐准教授，工学部・工学系研究科)
- 「II」 「並列計算プログラミング」，「先端計算機演習」(筆者，理学系研究科)

に対応しており，既に開講している科目である。

「並列プログラミング II」は，非並列のアプリケーションのプログラミングも含まれているので，図 6 のレベルとしては③，④の両方を含んでおり，計算機科学分野の受講者がアプリケーションについて学ぶのにも適している。平成 20 年度は夏季休暇中の集中講義であるが，情報基盤センターのメンバーの協力も得て，早速「T2K オープンスパコン」を使って講義・演習を実施する予定である。

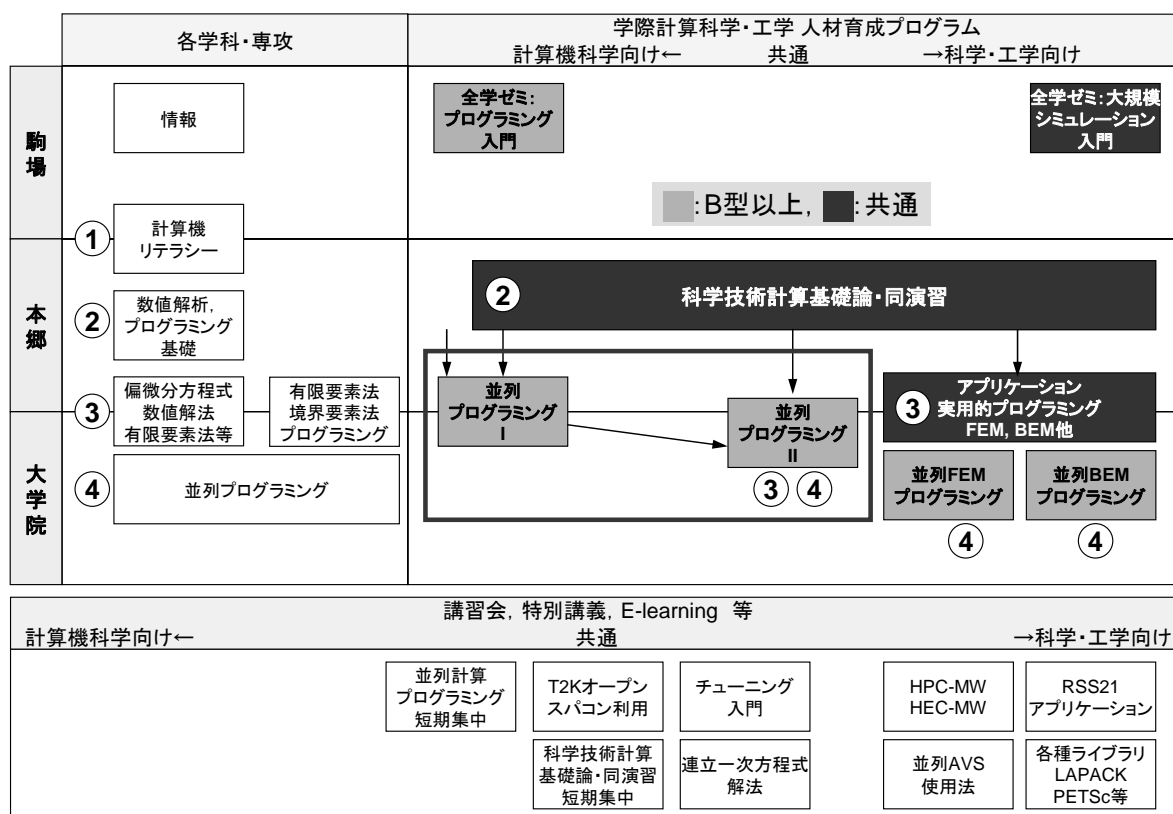


図7 学際計算科学・工学 人材育成プログラムと各学科・専攻の講義・演習
(連続体シミュレーション)

4. おわりに：T2K 連携，情報基盤センターの果たすべき役割

本稿では，筆者が2004年度から従事してきた，地球惑星科学専攻における「並列プログラミング教育」の概要と，それを全学的な教育プログラムとして拡張した「学際計算科学・工学 人材育成プログラム」の状況と将来構想について紹介した。

図6に①～④のレベルを紹介したが，現時点では詳細が決まっていない部分もあり，早急に検討する必要がある。本年度冬学期からは試験的に講義・演習を開始する予定であり，本広報誌を通じて経過については紹介していきたい。

本稿ではあまり詳しく述べなかったが，「学際計算科学・工学 人材育成プログラム」は「T2K オープンスパコン」のユーザー教育としての一面も持っており [7]，講習会，E-learning 教材，教科書等については，筑波大，東大，京大が連携して実施していく予定である。

「学際計算科学・工学 人材育成プログラム」によって「T2K オープンスパコン」を使って大規模シミュレーションを実施するユーザーが増加することは情報基盤センターとして嬉しいことではあるが，そのためには HPC-MW や HEC-MW のような大規模アプリケーション開発基盤の整備，受講者の「アフターケア」を積極的に実施していくことが重要である。

また，講習会，E-learning 教材，教科書等については，全国の大学の研究者，学生，産業界の研究者・技術者が参加，利用できるように公開していくことを基本方針としている。

より充実した教育プログラム，教材また「T2K オープンスパコン」利用環境を提供していくために，忌憚のないご意見，ご要望を頂くことができれば幸いである。

$$[\tilde{A}] = [M]^{-1} [A]$$

謝 辞

本稿で述べた、「並列計算プログラミング教育」の実施にあたっては、多くの皆様にご助力とご指導を頂いた。この場を借りて深く御礼を申し上げたい。

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、という意欲的な試みに挑戦できたことは貴重な体験であった。このような機会を与えてくださった山形俊男教授、松浦充宏教授を始め、東京大学大学院理学系研究科地球惑星科学専攻の教職員、学生の皆様に深く感謝したい。

また、端末室、並列計算機の利用に際しては、東京大学情報基盤センターの皆様に変にお世話になった。深甚なる謝意を表するものである。

参 考 文 献

- [1] 中島研吾 (2005) Hitachi SR8000を利用した並列プログラミング教育：東京大学21世紀COEプログラム 多圏地球システムの進化と変動の予測可能性, スーパーコンピューティングニュース (東京大学情報基盤センター) 7-5, 21-28
- [2] 中島研吾 (2006) Hitachi SR11000を利用した並列プログラミング教育：東京大学21世紀COEプログラム 多圏地球システムの進化と変動の予測可能性, スーパーコンピューティングニュース (東京大学情報基盤センター) 8-3, 21-28
- [3] 中島研吾 (2007) Hitachi SR11000を利用した並列プログラミング教育：東京大学21世紀COEプログラム 多圏地球システムの進化と変動の予測可能性, スーパーコンピューティングニュース (東京大学情報基盤センター) 9-3, 21-30
- [4] 中島研吾 (2007), OpenMPによるプログラミング入門 (I), スーパーコンピューティングニュース (東京大学情報基盤センター) 9-5, 7-20
- [5] 中島研吾 (2007), OpenMPによるプログラミング入門 (II), スーパーコンピューティングニュース (東京大学情報基盤センター) 9-6, 7-26
- [6] 中島研吾 (2008), OpenMPによるプログラミング入門 (III), スーパーコンピューティングニュース (東京大学情報基盤センター) 10-1, 19-37
- [7] 片桐孝洋 (2008) T2Kプロジェクトにおけるユーザ教育の連携, T2Kシンポジウムつくば2008, パネルディスカッション2「T2K連携とグリッド運営」:
<http://www.ccs.tsukuba.ac.jp/workshop/t2k-sympo2008/file/katagiri.pdf>
- [8] 奥田洋司, 中島研吾 (2003) HPCミドルウェア, 生産研究 55-3, 49-56
- [9] 奥田洋司 編著 (2008) 「並列有限要素解析 [II] 並列構造解析ソフトウェアFrontSTRを使いこなす」 培風館
- [10] Barrett R., Berry M., Chan, T.F., Demmel J.W., Donato J., Dongarra J.J., Eijkhout V., Pozo R., Romine C., van der Vorst H. (1994) Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, SIAM

(問い合わせ先)

中島 研吾 (nakajima@cc.u-tokyo.ac.jp)