

SCALE USERS GUIDE

Version 5.3.3

Team SCALE
UGC working group

2019 年 2 月 6 日



目 次

第 1 部	概要	5
1.1	はじめに	6
1.1.1	SCALE とは?	6
1.1.2	SCALE-RM の構成	8
1.2	表記上の注意	10
第 2 部	インストール	11
2.1	準備	12
2.1.1	システム環境	12
2.2	SCALE のコンパイル	15
2.2.1	ダウンロードと環境設定	15
2.2.2	コンパイル	16
2.3	後処理ツール (net2g) のコンパイル	19
2.3.1	後処理ツール (net2g) のコンパイル	19
第 3 部	SCALE-RM のチュートリアル	20
3.1	動作確認と基本操作について	21
3.1.1	概要	21
3.1.2	モデルの実行方法	21
3.1.3	後処理と描画	23
3.1.4	応用に向けたガイドライン	24
3.2	現実大気実験	27
3.2.1	概要	27
3.2.2	入力データ (境界値データ) の準備	27
3.2.3	実験セットの準備	30
3.2.4	地形データの作成 : pp	31
3.2.5	初期値/境界値データの作成 : init	33
3.2.6	シミュレーションの実行 : run	35
3.2.7	結果のクイック描画 : net2g	39

第 4 部	各種設定	43
4.1	前処理	44
4.1.1	初期値/境界値データの作成方法	44
4.2	SCALE-RM のフレームワーク	50
4.2.1	設定ファイルを用意するための補助ツール	50
4.2.2	旧バージョンの設定ファイルの変換プログラム	51
4.2.3	対象計算領域の設定	52
4.2.3.1	MPI プロセス数	52
4.2.3.2	水平・鉛直格子数	53
4.2.3.3	水平・鉛直格子間隔	53
4.2.3.4	スポンジ層	54
4.2.3.5	緩和領域と境界ナッジング	55
4.2.4	地図投影法と計算領域の位置の設定	61
4.2.5	地形の設定	62
4.2.6	ユーザー定義の地形の準備	64
4.2.7	積分時間と時間刻み幅の設定	65
4.2.7.1	力学過程に対する時間刻み幅	65
4.2.7.2	CFL 条件の確認	66
4.2.7.3	物理過程に対する時間刻み幅	66
4.2.7.4	経過時間タイマーによるモデルの終了	67
4.2.8	ヒストリファイルと出力変数の設定	67
4.2.9	リスタート計算の方法	70
4.2.10	領域ネスティング実験の方法	72
4.2.10.1	子領域における地形の取り扱い	72
4.2.10.2	オフライン・ネスティング実験	75
4.2.10.3	オンライン・ネスティング実験	79
4.3	力学コアの設定	83
4.3.1	デカルト座標系 C-grid による力学コア	83
4.3.1.1	時間積分の数値解法の設定	83
4.3.1.2	時間・空間差分スキーム	83
4.3.1.3	数値拡散	84
4.3.1.4	正定値性	85
4.3.1.5	八口	85
4.3.1.6	コリオリ力	85
4.3.1.7	コリオリ力に伴う側面境界条件の注意点	86
4.4	物理過程の設定	87
4.4.1	雲微物理スキーム	87
4.4.2	積雲パラメタリゼーション	88
4.4.3	乱流スキーム	90
4.4.4	惑星境界層スキーム	90
4.4.5	放射スキーム	91

4.4.6 大気下端境界における表面フラックス	96
4.4.7 海洋モデル	98
4.4.7.1 海洋表層スキーム	100
4.4.7.2 海面過程	101
4.4.8 陸モデル	103
4.4.8.1 陸面スキームと表面スキーム	104
4.4.9 都市モデル	105
4.4.9.1 大気-都市面間フラックス	105
4.5 共通の設定	107
4.5.1 ログファイル	107
4.5.1.1 ログファイルの出力	107
4.5.1.2 ログファイル内の時間に関する情報	108
4.5.2 モニターファイル	108
4.5.3 物理定数	109
4.5.4 暦	110
4.5.5 乱数生成	110
4.5.6 パフォーマンスの測定	111
4.5.7 統計量のモニター	111
4.6 後処理	113
4.6.1 netcdf2grads (net2g)	113
4.6.2 SCALE NetCDF Operator (SNO)	117
4.6.2.1 設定例: 基本的な使い方	118
4.6.2.2 設定例: プラグインされている機能	121
第5部 詳細説明 (応用編)	123
5.1 SCALE におけるファイル入出力	124
5.1.1 SCALE-netCDF ファイルとは?	124
5.1.1.1 グローバル属性	124
5.1.1.2 ハ口領域データ	124
5.1.1.3 軸変数	124
5.1.1.4 データ変数	128
5.1.1.5 単一ファイルの入出力	128
5.1.1.6 NetCDF 3 に伴う制限	129
5.2 ユーザーが作成したプログラムの組み込み方	132
5.2.1 ユーザが設定を行うためのモジュール	132
5.2.1.1 mod_user モジュールとは?	132
5.2.1.2 コンパイル	133

5.3	SCALE ライブラリの使い方	134
5.3.1	SCALE を使用するユーザープログラム	134
5.3.1.1	コンパイル	135
5.3.2	リファレンスマニュアル	136
5.3.2.1	サブルーチン	136
5.3.2.2	ネームリストのパラメータ	137
5.3.2.3	ヒストリ変数	137
5.4	バルクジョブの実行方法	138
5.4.1	バルクジョブとは?	138
5.4.2	バルクジョブの設定	138
5.4.3	失敗プロセスの管理	139
5.4.4	バルクジョブのための準備	139
5.4.5	バルクジョブの実行	140
付 録 A	よくある質問とその回答 : FAQ	145

第1部

概要

1.1 はじめに

本書は初めて領域気象気候モデル SCALE-RM を利用する人に向けた解説書である。気象気候ライブラリー SCALE version 5.3.3 に対応した説明を記載する。SCALE の現バージョンには、領域モデル SCALE-RM と全球モデル SCALE-GM が含まれる。本版では、SCALE-RM の使い方についてのみ詳しく述べる。SCALE-GM については、次版で詳しく記載される予定である。

本書の構成は次の通りです。第 1 部では SCALE の概要、第 2 部では必要な環境とインストール方法について説明する。続いて、第 3.1 章では理想実験、第 3.2 章では現実大気実験を例にして、SCALE-RM の基本的な操作方法を説明する。これらの章はひと繋ぎのチュートリアルとなっており、SCALE-RM を初めて使うユーザは一通り通読することを推奨する。第 4 部と第 5 部では、モデルの設定の変更方法を記載し、また利用できるデータ形式やツールを説明する。これらの各章は基本的にその中で閉じているので、辞書として用いることができる。

本書中の不明点やお気づきの点がございましたら、SCALE user's メーリングリスト scale-users@ml.riken.jp までご連絡ください。

1.1.1 SCALE とは？

SCALE (Scalable Computing for Advanced Library and Environment) は、気候研究や天気予報を容易に様々な計算機上で行うためのソフトウェア・ライブラリである。本ライブラリは、前処理からシミュレーション、後処理、解析に至るまで全ての過程を網羅し、下記に挙げる長所を持つ。

- SCALE は、「BSD-2 ライセンス」のもとオープンソースソフトウェアとして提供している。商用、非商用に関わらず自由な利用、改変、再配布が可能である。
- SCALE には、SCALE-RM (SCALE-Regional Model) という領域モデルが含まれる。
- SCALE には、次節で説明するように様々なスキームが用意されている。ユーザが行いたい実験に合わせて適宜選択できる。
- SCALE では、SCALE-RM だけでなく他の数値モデルでも呼び出せる物理過程のフレームワークを提供している。

ライセンスの詳細は、トップディレクトリ直下の `scale-5.3.3/LICENSE` のファイルに記述されている。SCALE の使用前に一読されたい。また、SCALE の Web ページ (<http://scale.aics.riken.jp/>) にもソフトウェアの説明が記載されているので必要に応じて参照されたい。

本節では、SCALE の思想や実際のモデルとの関係について説明する。SCALE-RM の実行とは直接的には関係しないため、必要なければ読み飛ばしても構わない。

SCALE のライブラリとモデルの関係について

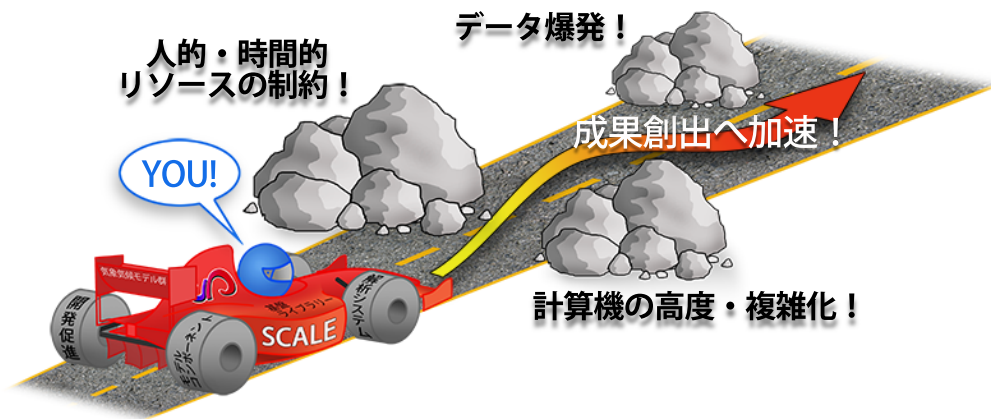


図 1.1.1: SCALE のねらい

SCALE は幾つかの外部共同研究者と共に理化学研究所 (RIKEN) で開発され、その改良と拡張が継続的に行われている。図 1.1.1 に SCALE の思想の概念図を示す。この図に示されるように、SCALE は様々な問題に対応することを目指している。SCALE は、小型 PC クラスタから次世代のスーパーコンピュータに至るまで様々な計算環境で広く用いられる事を念頭に開発されている。この目的のため、気候・気象科学を専門とする科学者と計算機科学を専門とする科学者が共同で開発している。

SCALE-RM は SCALE ライブラリを大いに利用した数値モデルの一つであり、図 1.1.2 に示すように SCALE のパッケージに含まれる。SCALE は、並列プロセスの管理、ファイルの入出力、プロセス間の通信、格子情報の設定を行う。また、SCALE は、大気の流れのソルバ(力学コア)や雲微物理や大気放射等の物理過程も提供する。その一方で、SCALE-RM は SCALE が提供する機能を組み合わせることで構築されている。SCALE-RM 自体は、大気の状態の入力データを予報変数として読み込んで保持し、SCALE の各コンポーネントを適宜呼び出すことで時間発展を計算する。ユーザは行いたい実験に応じて、各コンポーネントのスキームを選択できる。

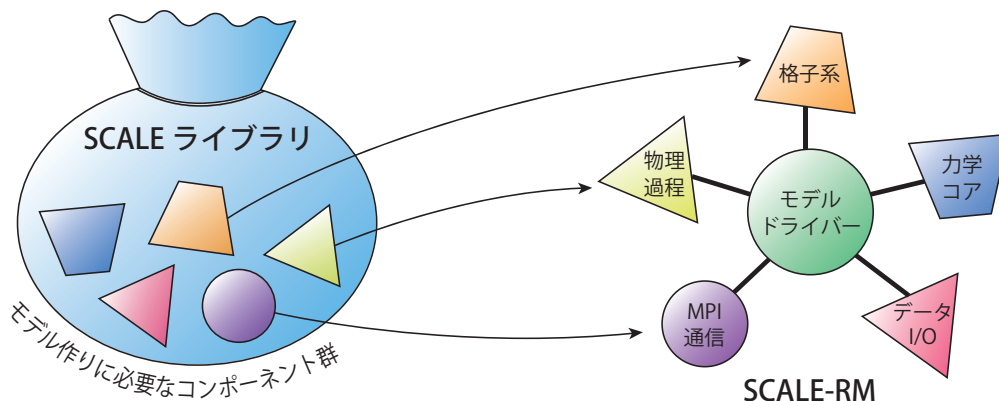


図 1.1.2: SCALE ライブラリと SCALE-RM(モデル) の関係

1.1.2 SCALE-RM の構成

SCALE に含まれる全てのコンポーネント中の全てのスキームを、SCALE-RM において利用できる。コンポーネントは 3 つの部分 (フレームワーク、力学コア、物理過程) に分類される。以下に、SCALE-RM の現版に実装済みである、様々なスキームを含むコンポーネントを列挙する^{*1)}。

フレームワーク

- 実距離に基づいた 3 次元カーテシアン格子系
- MPI 通信を用いた 2 次元領域分割
- 各種地図投影法
- 領域ネスティングシステム (1 way: 親領域 → 子領域へデータ転送)
 - オンライン・ネスティング: 複数ドメインの計算を同時に実行
 - オフライン・ネスティング: 外側ドメインの計算終了後に、その結果を用いて内側ドメインの計算を実行
- 複数事例一括実行システム (バルクジョブシステム)
- CF 規約^{*2)}に基づく netCDF ファイル I/O
 - netCDF3 または netCDF4 形式を選択
- 理想実験のための初期値データ生成
- 外部データから標高・土地利用区分データを作成
- 外部データから初期値・境界値データを作成
 - WRF-ARW^{*3)}、GrADS^{*4)}形式での入力に対応

力学コア

- 支配方程式系: 3 次元完全圧縮非静力学方程式系
- 空間離散化: 有限体積法
 - 2 次, 4 次, 6 次中央差分
 - 3 次, 5 次風上差分
- 時間離散化: 「完全陽解法」または「水平陽解法-鉛直陰解法」から選択
 - Heun 型 3 次ルンゲ・クッタスキーム
 - Wicker and Skamarock (2002) の 3 段ルンゲ・クッタスキーム
 - 4 次ルンゲ・クッタスキーム

^{*1)} モデルの構成や離散化法の詳細は、Team SCALE (2015)、Sato et al. (2015)、Nishizawa et al. (2015) を参照されたい。

^{*2)} <http://cfconventions.org/>

^{*3)} <http://www.wrf-model.org/>

^{*4)} <http://cola.gmu.edu/grads/>

- 非負保証:
 - フラックス修正法 (Flux Corrected Transport, FCT; [Zalesak, 1979](#))
 - [Koren \(1993\)](#) フィルター (3 次風上差分スキーム使用時のみ)
- 数値フィルタ: 4 次超粘性・拡散
- 地形: 地形に沿った座標系を用いて表現

物理過程

- 乱流過程: 以下から選択可能
 - [Smagorinsky \(1963\)](#) & [Lilly \(1962\)](#) 型のサブグリッドスケール乱流モデル ([Brown et al. \(1994\)](#) と [Scotti et al. \(1993\)](#) による補正)
 - [Deardorff \(1980\)](#) サブグリッドスケール乱流モデル
 - [Mellor and Yamada \(1982\)](#); [Nakanishi and Niino \(2004\)](#) による level 2.5 境界層乱流パラメタリゼーション
- 雲微物理: 以下から選択可能
 - [Kessler \(1969\)](#) による 3-class 1 モーメントバルクモデル
 - [Tomita \(2008\)](#) による 6-class 1 モーメントバルクモデル
 - [Seiki and Nakajima \(2014\)](#) による 6-class 2 モーメントバルクモデル
 - [Suzuki et al. \(2010\)](#) によるビン法モデル
- 放射過程: [Sekiguchi and Nakajima \(2008\)](#) による相関 k 分布法ブロードバンド大気放射伝達モデル
- 地表面モデル
 - 陸面モデル: 熱拡散・バケツモデル
 - 海洋モデル: 以下から選択可能
 - * 初期値固定
 - * 外部データ入力
 - * スラブモデル
 - 都市モデル: [Kusaka et al. \(2001\)](#) による単層キャノピーモデル
 - バルク交換係数 (陸面および海面): 以下から選択可能
 - * [Beljaars and Holtslag \(1991\)](#); [Wilson \(2001\)](#) による普遍関数によるバルク法
 - * [Uno et al. \(1995\)](#) による Louis 型バルク法

1.2 表記上の注意

本書中では、Unix システム上のシェルである「`bash`」での実行を想定している。異なる環境下では、適宜コマンドを読み替えて対応されたい。また、本書内では特に断りがない限り、下記の表記法に従うものとする。

コマンドラインのシンボル（`$`，`#`）は、コマンドの実行を示す。以下のように、2つの表記の違いはプログラムの実行権限の違いを表す。

`#` \leftarrow `root` 権限で実行するコマンド
`$` \leftarrow ユーザ権限で実行するコマンド

下記に示すように、四角い囲みで区切られた記述は、コマンドラインのメッセージ部分を表す。

----- コマンドラインのメッセージ
----- コマンドラインのメッセージ
----- コマンドラインのメッセージ

一方、下記のように丸い囲みで区切られた記述は、エディタでファイルを編集する記述内容、もしくはファイル内の記述の引用を表す。

----- ファイル中の記述
----- ファイル中の記述
----- ファイル中の記述

本書では、FORTRAN のネームリストを `[namelist]`、その項目を `(item_of_namelist)` のように表記する。

第2部

インストール

2.1 準備

本章では、SCALE や SCALE-RM のコンパイル方法、実行に必要とされる最小の計算環境を説明する。

2.1.1 システム環境

推奨のハードウェア構成

必要なハードウェアは実験設定に依存するが、ここでは第 3.1 章と第 3.2 章のチュートリアルを実行するために必要なスペックを示す。

- CPU : チュートリアルの理想実験でに物理コアが 2 コア以上、現実大気実験には 4 コア以上が望ましい。
- Memory : 理想実験には 512MB 以上、現実大気実験には 2GB 以上のメモリ容量が必要とされる。ただし、この要件は倍精度浮動小数点を使用した場合である。
- HDD : 現実大気実験には約 3GB のディスク空き容量が必要とされる。

必要なソフトウェア

- OS : Linux OS、MacOS
- コンパイラ : C、Fortran

SCALE のソースコードは Fortran 2003 規格に基づく機能を一部利用しているため、それに対応するコンパイラが必要である。例えば、GNU GFortran のバージョン 4.3 以前では Fortran 2003 規格の対応が不十分であるので、SCALE のコンパイルには利用できない。対応確認済みのコンパイラは、表 2.1.1 を参照されたい。

表 2.1.1: 対応確認済みのコンパイラ

コンパイラ名	
GNU (gcc/gfortran)	4.3 以前は非対応。4.4.x ではコンパイル時に Warning が出ることがある。
Intel (icc/ifort)	2013 年以降のバージョンを推奨。
PGI (gcc/pgfortran)	17.1 は確認済み

必要なライブラリ

- netCDF ライブラリ (<http://www.unidata.ucar.edu/software/netcdf/>)
- MPI ライブラリ (e.g., openMPI <http://www.open-mpi.org/>)
- LAPACK (<http://www.netlib.org/lapack/>) (SCALE-GM のみが要求)

netCDF4 が推奨されるが、netCDF3 も利用できる。netCDF3 を用いる場合は、いくつか制限が発生することに注意が必要である (第 5.1.1 節を参照)。Linux や Mac 用に配布された NetCDF のバイナリパッケージも使用することができる。また、「京」には、既にこれらの netCDF ライブラリがインストールされている。詳細は「京」のポータルサイトにある「R-CCS ソフトウェア」の記述やリンク先 (<https://www-sys-aics.riken.jp/releasedsoftware/ksoftware/pnetcdf/>) を参照されたい。

MPI 1.0/2.0 プロトコルに対応した MPI ライブラリを必要とする。対応確認済みの MPI ライブラリについては、表 2.1.2 を参照のこと。

表 2.1.2: 対応確認済みの MPI ライブラリ

MPI ライブラリ名	
openMPI	1.7.2 以降に対応。
Intel MPI	2013 年以降のバージョンに対応。
SGI MPT	2.09 以降に対応。

第 3.1 章や第 3.2 章のチュートリアルでは、上記のライブラリの準備が完了しているとする。

描画ツール

- GPhys / Ruby-DCL by 地球流体電脳倶楽部
 - URL: <http://ruby.gfd-dennou.org/products/gphys/>
 - 特徴: SCALE は、MPI プロセスによる領域分割に従った netCDF 形式の分割ファイルを出力する。GPhys に含まれる「gpview」や「gpvect」コマンドは、後処理を行わずに分割ファイルを直接可視化できる。
 - インストール方法: 主要な OS に対するインストール方法は、地球流体電脳倶楽部の Web ページ (<http://ruby.gfd-dennou.org/tutorial/install/>) に書かれている。
- Grid Analysis and Display System (GrADS) by COLA
 - URL: <http://cola.gmu.edu/grads/>
 - 特徴: 非常によく使われている描画ツールであるが、SCALE によって生成された netCDF 形式の分割ファイルを直接読み込めない。SCALE が出力したファイルを GrADS から読み込める単一のファイルに結合するためには、後処理ツール netcdf2grads (net2g) が必要である。net2g のインストール方法は第 2.3.1 章、使用方法の詳細は第 3、4 部および第 4.6.1 章を参照されたい。
- Ncview: netCDF の可視化ブラウザ (David W. Pierce により開発された)
 - URL: http://meteora.ucsd.edu/~pierce/ncview_home_page.html

- 特徴: Ncview は netCDF 形式のファイルのためのクイックビューアである。SCALE によって生成された分割ファイルの結合はできないが、各ファイルごとに結果を描画するときに便利である。
- インストール方法: http://meteora.ucsd.edu/~pierce/ncview_home_page.html を参照。

便利なツール (必須ではない)

- データ変換ツール: wgrib、wgrib2、NCL
SCALE-RM で読込可能な入力データを作成できる。チュートリアルの実験では wgrib を使用する。
- 演算性能の評価ツール: PAPI ライブラリ^{*1)}が使用可能。

^{*1)}<http://icl.utk.edu/papi/>

2.2 SCALE のコンパイル

2.2.1 ダウンロードと環境設定

以下の説明で使用する環境は次のとおりである。

- CPU: Intel Core i5 2410M 2 コア/4 スレッド
- Memory: DDR3-1333 4GB
- OS: CentOS 6.6 x86-64、CentOS 7.1 x86-64、openSUSE 13.2 x86-64
- GNU C/C++、Fortran コンパイラ

ソースコードの入手

最新のリリース版ソースコードは、<http://scale.aics.riken.jp/ja/download/index.html> から取得できる。ソースコードの tarball ファイルを展開すると、scale-5.3.3/ というディレクトリができる。

```
$ tar -zxvf scale-5.3.3.tar.gz
$ ls ./
scale-5.3.3/
```

Makedef ファイルと環境変数の設定

環境変数「SCALE_SYS」に設定した Makedef ファイルを使用して、SCALE はコンパイルされる。scale-5.3.3/sysdep/ には、いくつかの計算機環境に対応する Makedef ファイルが用意されており、これらの中から自分の環境に合ったものを選択する。表 2.2.1 に、Makedef ファイルが用意されている環境を示す。自分の環境に合致するファイルがなければ、既存のファイルを基に各自で作成する必要がある。例えば Linux x86-64 OS、GNU コンパイラ、openMPI を使用する場合には、該当

表 2.2.1: 環境例およびそれに対応する Makedef ファイル

OS/計算機	コンパイラ	MPI	Makedef ファイル
Linux OS x86-64	gcc/gfortran	openMPI	Makedef.Linux64-gnu-ompi
	icc/ifort	intelMPI	Makedef.Linux64-intel-impi
	icc/ifort	SGI-MPT	Makedef.Linux64-intel-mpt
macOS	gcc/gfortran	openMPI	Makedef.MacOSX-gnu-ompi
スーパーコンピュータ「京」	fccpx/frtpx	mpiccp/mpiirtpx	Makedef.K
Fujitsu PRIME-HPC FX10	fccpx/frtpx	mpiccp/mpiirtpx	Makedef.FX10

する Makedef ファイルは「Makedef.Linux64-gnu-mpi」である。このとき、環境変数は下記のように設定する必要がある。

```
$ export SCALE_SYS="Linux64-gnu-mpi"
```

実行環境が常に同じならば、.bashrc 等の環境設定ファイルに環境変数の設定を陽に記述しておく
と便利である。

SCALE は netCDF を必要とする。多くの場合、netCDF のパスは「nc-config」を用いることで自
動的に見つけられる。しかし、自動的に見つけれない場合には、例えば以下のように netCDF に
関する PATH を設定しなければならない。

```
$ export SCALE_NETCDF_INCLUDE="-I/opt/netcdf/include"
$ export SCALE_NETCDF_LIBS= \
    "-L/opt/hdf5/lib64 -L/opt/netcdf/lib64 -lnetcdff -lnetcdf -hdf5_hl -lhdf5 -lm -lz"
```

2.2.2 コンパイル

SCALE-RM のコンパイル

SCALE-RM のソースディレクトリに移動し、下記のコマンドによってコンパイルする。

```
$ cd scale-5.3.3/scale-rm/src
$ make -j 4
```

「-j 4」はコンパイル時の並列数を示している (例では 4 並列)。コンパイルにかかる時間を短縮する
ために、並列コンパイルのオプションを指定することが望ましい。並列数は実行環境に応じて設定し、
推奨の並列数は 2~8 である。コンパイルが成功すると、下記の 3 つの実行ファイルが scale-5.3.3/bin
以下に生成される。

```
scale-rm scale-rm_init scale-rm_pp
```

SCALE-GM のコンパイル

SCALE-GM のソースディレクトリに移動し、下記のコマンドによってコンパイルする。

```
$ cd scale-5.3.3/scale-gm/src
$ make -j 4
```

コンパイルが成功すると、下記の実行ファイルが scale-5.3.3/bin 以下に生成される。「fio」は、ヘッ
ダー情報を伴う、バイナリに基づく独自形式である。

scale-gm	(\scalegm の実行バイナリ)
gm_fio_cat	(fio 形式のための cat コマンド)
gm_fio_dump	(fio 形式のための dump ツール)
gm_fio_ico2ll	(fio 形式の正二十面体格子データから緯度経度格子データへの変換ツール)
gm_fio_sel	(fio 形式のための sel コマンド)
gm_mkhgrid	(バネ格子を用いた正二十面体 (水平) 格子の生成ツール)
gm_mkllmap	(緯度経度 (水平) 格子の生成ツール)
gm_mkmnginfo	(MPI プロセスの割り当てを管理するファイルの生成ツール)
gm_mkrawgrid	(正二十面体 (水平) 格子の生成ツール)
gm_mkvlayer	(鉛直格子の生成ツール)

注意点

コンパイルをやり直したい場合は、下記のコマンドで作成された実行バイナリを消去する。

```
$ make clean
```

ただし、コンパイルされたライブラリは消去されないことに注意が必要である。全てのコンパイル済みファイルを消去したい場合は、以下のコマンドを使用する。

```
$ make allclean
```

コンパイル環境、コンパイルオプションを変更して再コンパイルする場合は、「allclean」を実行されたい。

SCALE においてコンパイルやアーカイブは、ディレクトリ `scale-5.3.3/scalelib/` の中で行われる。作成されたオブジェクトファイルは、コンパイルを実行したディレクトリ下の「`.lib`」という名前の隠しディレクトリの中に置かれる。

デバッグモードでコンパイルしたい場合は、「`make -j 4 SCALE_DEBUG=T`」を実行してコンパイルする（コンパイル時に適用される全ての環境変数リストは表 [2.2.2](#) を参照）。細かくコンパイルオプションを変更したい場合は、`Makedef` ファイルを編集されたい。

表 2.2.2: コンパイル時の環境変数のリスト

環境変数	説明
SCALE_SYS	システム選択 (必須)
SCALE_DISABLE_MPI	MPI を使わない (utils のみ)
SCALE_DEBUG	デバッグ用コンパイルオプションでコンパイル
SCALE_QUICKDEBUG	クイックデバッグ用コンパイルオプション利用 (高速化そのまま + 浮動小数点エラー検出)
SCALE_USE_SINGLEFP	単精度浮動小数点を使用 (全ソース)
SCALE_USE_FIXEDINDEX	格子サイズをコンパイル時に固定して最適化促進
SCALE_ENABLE_OPENMP	OpenMP 機能を有効にする
SCALE_ENABLE_OPENACC	OpenACC 機能を有効にする
SCALE_USE_AGRESSIVEOPT	副作用が出る可能のある強い最適化まで行う (京・FX のみ)
SCALE_DISABLE_INTELVEC	ベクトル化オプションの抑制 (インテルコンパイラのみ)
SCALE_NETCDF_INCLUDE	NetCDF ライブラリの include ディレクトリパス
SCALE_NETCDF_LIBS	NetCDF ライブラリのディレクトリパスとライブラリ指定
SCALE_ENABLE_PNETCDF	parallel NetCDF を利用する
SCALE_COMPAT_NETCDF3	NetCDF3 互換の機能に限定する
SCALE_ENABLE_MATHLIB	数値計算ライブラリを利用する
SCALE_MATHLIB_LIBS	数値計算ライブラリのディレクトリパスとライブラリ指定
SCALE_ENABLE_PAPI	PAPI を利用する
SCALE_PAPI_INCLUDE	PAPI ライブラリの include ディレクトリパス
SCALE_PAPI_LIBS	PAPI ライブラリのディレクトリパスとライブラリ指定
SCALE_DISABLE_LOCALBIN	テストケースディレクトリに特別版のバイナリが作られないようにする
SCALE_IGNORE_SRCDEP	コンパイル時にソースコードの依存関係確認を行わない
SCALE_ENABLE_SDM	超水滴モデルを利用する

2.3 後処理ツール(net2g)のコンパイル

2.3.1 後処理ツール(net2g)のコンパイル

「net2g」は SCALE-RM 用の後処理ツールである。SCALE-RM の出力ファイルはノードごとに分割されている。これらの出力ファイル(history.*****.nc)を結合し、GrADSで直接読み込めるデータ形式へ変換するための後処理ツール「net2g」を SCALE は提供している。第 3.1 章、第 3.2 章のチュートリアルで使用するため、ここでは「net2g」のコンパイル方法を説明する。

まず、SCALE 本体のコンパイル時と同様に、使用環境に合った Makedef ファイルを指定するために環境変数を設定する。次に、「net2g」のディレクトリに移動し、make コマンドを実行する。MPI ライブラリを使用した並列実行バイナリは、下記のコマンドによって生成される。

```
$ cd scale-5.3.3/scale-rm/util/netcdf2grads_h
$ make -j 2
```

MPI ライブラリが無い場合に、逐次実行バイナリを生成するには、

```
$ make -j 2 SCALE_DISABLE_MPI=T
```

のコマンドによってコンパイルする。「net2g」という名前の実行ファイルが生成されていれば、コンパイルは成功である。作成した実行バイナリを消去する場合は、下記のコマンドを実行する。

```
$ make clean
```


第3部

SCALE-RMのチュートリアル

3.1 動作確認と基本操作について

3.1.1 概要

本章では、数値実験における SCALE-RM の基本的な操作を理想実験を題材にして説明する。第 2 部で行った SCALE のコンパイルが正常に完了しているかの確認も兼ねているので、本チュートリアルを実施することを強く推奨する。本章では下記のファイルが生成されているとして説明を進める。

```
scale-5.3.3/bin/scale-rm
scale-5.3.3/bin/scale-rm_init
scale-5.3.3/scale-rm/util/netcdf2grads_h/net2g
```

さらに、描画ツールとして GrADS を使用する。結果の確認には「gpview」も利用できる。GrADS や「gpview」(GPhys) のインストール方法は第 2.1.1 節を参照されたい。

以下のチュートリアルは、前準備、初期値作成、シミュレーション実行、後処理、描画の順で記述している。

3.1.2 モデルの実行方法

実験設定

理想実験のチュートリアルとして、積雲対流の理想実験を実施する。この実験では、典型的な大気鉛直分布および対流圏下層に初期擾乱を与え、積乱雲が発生し発達する様子を準 2 次元モデルにおいて表現する。表 3.1.1 に、実験設定を示す。

準備

理想実験は、ディレクトリ `scale-rm/test/tutorial/ideal` の中で実行する。このディレクトリに移動し、`scale-5.3.3/bin` にある実行バイナリへの静的リンクを張る。

```
$ cd scale-rm/test/tutorial/ideal
$ ln -s ../../../../bin/scale-rm      ./
$ ln -s ../../../../bin/scale-rm_init ./
```

ここで、「scale-rm」はモデル本体、「scale-rm_init」は初期値/境界値作成ツールである。

初期値の作成

初期値を作成するには、`scale-rm_init` に与える設定ファイルが必要である。設定ファイル `init_R20kmDX500m.conf` には、表 3.1.1 に対応する実験設定が書かれている。この設定ファイルを読み込めると、大気の成層構造と初期擾乱が計算される。

SCALE-RM の実行コマンドの一般的な形式は、

表 3.1.1: 理想実験の実験設定

項目	設定内容	備考
MPI プロセス数	東西：2、南北：1	計 2 プロセスでの並列計算を行う
水平格子間隔	東西：500 m、南北：500 m	東西-鉛直の面を切り取った準 2 次元実験である
水平格子点数	東西：40、南北：2 ^a	
鉛直層数	97 層（モデル上端 20 km）	下層ほど細かい層厚を持ったストレッチ格子の設定である
側面境界条件	周期境界	東西、南北境界とも
積分時間間隔	5 sec	雲微物理スキームは 10 sec 毎
積分期間	3,600 sec	720 steps
データ出力間隔	300 sec	
物理スキーム	雲微物理モデルのみ使用	6-class single moment bulk model (Tomita, 2008)
初期の鉛直分布	GCSS Case1 squall-line (Redelsperger et al., 2000)	風の分布は、Ooyama (2001) に基づいた鉛直シアを与える
初期擾乱	暖気塊 (warm bubble)	水平 4 km、鉛直 3 km の半径を持ち、極大値が 3 K の暖気塊を置く

^a現在は 2 次元実験を行うための枠組みは用意されていないが、Y 方向に同じ値をもつ初期値を与える事で 2 次元実験に相当する実験を行うことが可能である。この場合、八口の格子数と同じ数の格子数を Y 方向に設定する必要がある。八口に必要格子数については 4.3.1.2 参照。

```
$ mpirun -n [プロセス数] [実行バイナリ名] [設定ファイル]
```

である。[プロセス数] には MPI 並列で使用したいプロセス数、[実行バイナリ] には scale-rm や scale-rm_init といった実行バイナリ名を指定する。そして、[設定ファイル] には実験設定を記述した設定ファイルを指定する。2 プロセスの MPI 並列計算を行うように、init_R20kmDX500m.conf を設定している場合は、scale-rm_init を実行するコマンドは

```
$ mpirun -n 2 ./scale-rm_init init_R20kmDX500m.conf
```

と記述する。実行が成功すれば、コマンドラインのメッセージは下記のように表示される。

```
*** Start Launch System for SCALE-RM
*** Execute preprocess? : T
*** Execute model? : F
*** a single communicator
*** a single communicator
*** End Launch System for SCALE-RM
```

この実行によって、下記の 3 つのファイルが、現在のディレクトリ下に作成される。

```
init_LOG.pe000000
init_00000101-000000.000.pe000000.nc
init_00000101-000000.000.pe000001.nc
```

計算領域の全体は、MPI プロセス数だけ水平分割される。ファイル名において pe に続く番号は、MPI のプロセス番号を示している。ログファイル (init_LOG.pe000000) には、コマンドラインには表示

されない詳細情報が記録されている。この例では 2 つの MPI プロセスを使用しているが、0 番目のプロセス（マスターランク）に対するログファイルだけがデフォルト設定では出力される。実行が正常に終了すれば、LOG ファイルの最後に

```
++++++ finalize MPI...
++++++ MPI is peacefully finalized
```

が出力される。

init_00000101-000000.000.pe000000.nc と init_00000101-000000.000.pe000001.nc の 2 つのファイルは初期値ファイルであり、それぞれ約 600 KB のファイルサイズになる。ファイル名の末尾が「.nc」で終わるファイルは netCDF 形式のファイルであり、GPhys/Ruby-DCL や ncview によって直接読み込める。

シミュレーションの実行

プロセス並列数は、初期値を作成した時と同じにする必要がある。シミュレーション実行用の設定ファイルは run_R20kmDX500m.conf である。

```
$ mpirun -n 2 ./scale-rm run_R20kmDX500m.conf
```

本書の必要要件にあった計算機であれば、2 分程度で計算が終わる。この実行によって、3 つのファイル

```
LOG.pe000000
history.pe000000.nc
history.pe000001.nc
```

が、現在のディレクトリ下に作成される。実行が正常に終了すれば、LOG ファイルの最後に

```
++++++ finalize MPI...
++++++ MPI is peacefully finalized
```

と出力される。history.pe000000.nc と history.pe000001.nc の 2 つのファイルは、計算結果を含むヒストリファイルである。これらのファイル形式は netCDF であり、各ファイルのサイズは約 5.8 MB である。

3.1.3 後処理と描画

ここでは後処理と計算結果の描画方法を説明する。このチュートリアルでは、netCDF 形式の分散ファイルを単一のファイルに結合し、ダイレクトアクセスが可能な単純なバイナリ形式（GrADS 形式）に変換する。このバイナリ形式は、ユーザーによる結果の解析を容易にする。まず、2.3.1 節でコンパイルした後処理ツール net2g へのリンクを張る。

```
$ ln -s ../../../../util/netcdf2grads_h/net2g ./
```

net2g の実行方法は、基本的に SCALE-RM と同じであり、

```
$ mpirun -n [プロセス数] ./net2g [設定ファイル]
```

の形式で実行する。net2g_R20kmDX500m.conf は net2g 専用の設定ファイルである。この設定ファイルを net2g に与えて、次のように実行する。

```
$ mpirun -n 2 ./net2g net2g_R20kmDX500m.conf
```

エラーメッセージがなく、下記のメッセージだけが標準出力へ表示されていれば、変換は正常に完了している。

```
+++ MPI COMM: Corrective Finalize
```

net2g を実行する際には、SCALE-RM の実行時に使用した MPI プロセス数と同じか、その約数のプロセス数を使用しなければならない。この実行によって、実行ディレクトリ下に下記の 6 つのファイルが作成される。

```
QHYD_d01z-3dctl
QHYD_d01z-3dgrd
U_d01z-3dctl
U_d01z-3dgrd
W_d01z-3dctl
W_d01z-3dgrd
```

「grd」ファイルは、分割ファイルを結合することによって得られる、ダイレクトアクセス可能な単純バイナリ形式 (GrADS 形式) である。一方で、「ctl」ファイルは GrADS によって「grd」ファイルを読む込むときに必要な情報を含む。

計算が問題なく完了しているかを確認するため、GrADS スクリプト checkfig_ideal.gs を使って作図する。なお、GrADS のバージョンによって文法が異なるため、警告が出る場合には GrADS スクリプトを適宜変更されたい。作図は以下のコマンドで行う。

```
$ grads -blc checkfig_ideal.gs
```

コマンドが成功すれば、下記のファイルが生成される。

```
ideal_QHYD.png
ideal_W.png
```

シミュレーションと後処理が問題なく行われていれば、図 3.1.1 と同じ図が得られる。

他の変数もバイナリデータに変換したい場合は、net2g_R20kmDX500m.conf の [VARI] の (VNAME) に必要な変数を追加すれば良い。

```
&VARI
VNAME = "U", "W", "QHYD"
/
```

ヒストリファイルに出力された変数は、netCDF の ncdump 等を用いて簡単に確認できる。net2g の詳しい使い方は第 4.6.1 節を参照されたい。

3.1.4 応用に向けたガイドライン

本章では簡単な理想実験を例にして SCALE-RM の実行方法を説明した。次の段階として、モデルの解像度、計算領域、MPI プロセス数を変更する方法を把握することを勧める。本章の理想実験

に関しては、この実験で使用したディレクトリ下にある「sample」ディレクトリの中に、解像度設定、領域設定、物理スキーム等を変更した設定ファイルを数種類用意してある。これらは、設定を変更する際に参考となるだろう。また、ディレクトリ「scale-rm/test/case」の下には、様々な理想実験に対する設定を用意している。幾つかの理想実験については、それらの実験設定に特化したソースコードを必要とするため、設定ファイルの存在するディレクトリで `make` コマンドを再度実行する必要がある。初期値作成と実行の手順は、基本的に本章のチュートリアルと同じである。

雲微物理スキーム、放射スキーム、乱流スキーム等の物理過程の設定方法を確認することも重要である。これらの変更方法は第 4 章に記載されている。

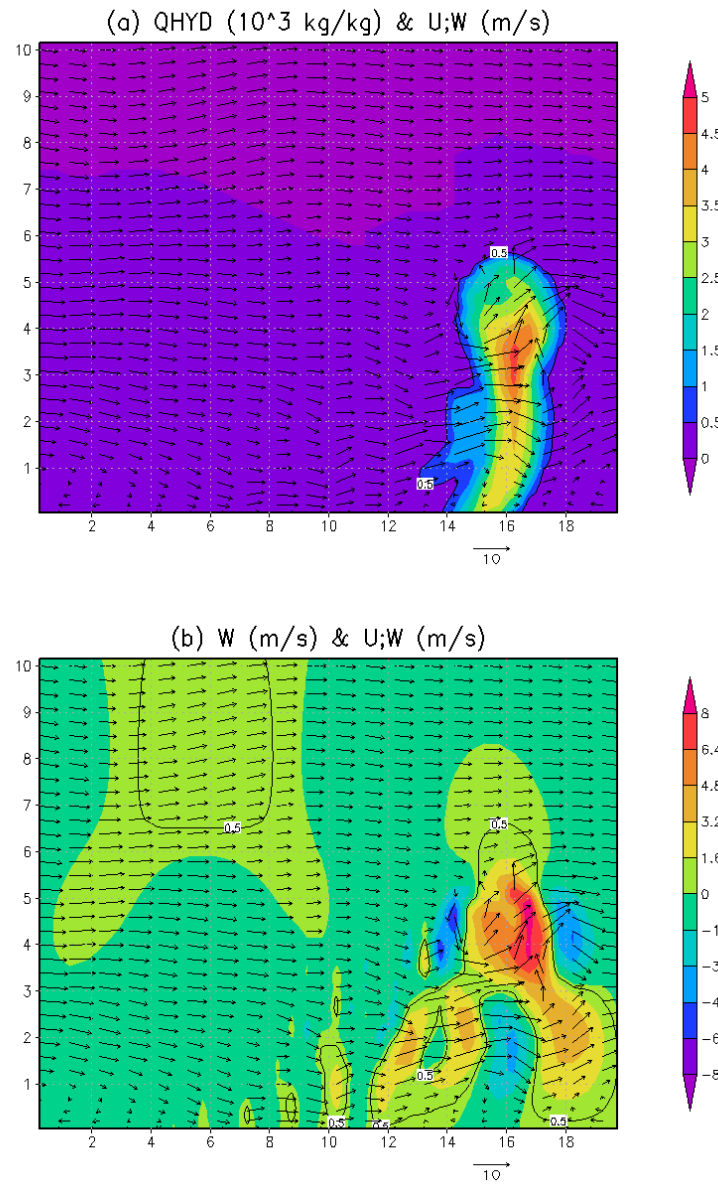


図 3.1.1: 積分開始から 1200 秒 (20 分) 後の $Y=750 \text{ m}$ における東西-鉛直断面図；カラーシェードは、(a) において全質量に対する凝結物の質量比、(b) において鉛直速度を表す。ベクトルは東西-鉛直断面内の風の流れを表す。

3.2 現実大気実験

3.2.1 概要

本章では、以下の流れ (図 3.2.1 も参照) に従う簡単な場合を例にして、現実大気実験の基本的な実行手順を示す。

1. 入力データの準備 (入力データは各自で準備しなければならない)
2. pp : 地形データの作成
3. init : 初期値・境界値データの作成
4. run : シミュレーションの実行
5. net2g : 出力データを netCDF から GrADS 形式に変換 (オプション)

これ以降の説明では、scale-5.3.3/scale-rm/test/tutorial/の絶対パスを`${Tutorial_DIR}`と書くことにする。

本章のチュートリアルで用いる計算領域の設定を表 3.2.1 に示す。また、対象とする計算領域を図 3.2.2 に示す。本チュートリアルの目的は、SCALE-RM を用いて現実大気実験を実施する方法を短時間で学ぶことである。そのために、計算が短時間で終わるように実験を設定している。したがって、この設定が物理的に妥当な実験として適切とは限らないことに注意が必要である。実際の研究を行うときには、必要に応じて実験設定を検討すべきである。

3.2.2 入力データ (境界値データ) の準備

現実大気実験を行う場合には、SCALE-RM に境界値データを与える必要がある。表 3.2.2 に、境界値データの作成に必要な外部入力データの項目を示す。この表において、青字で書かれた変数は必

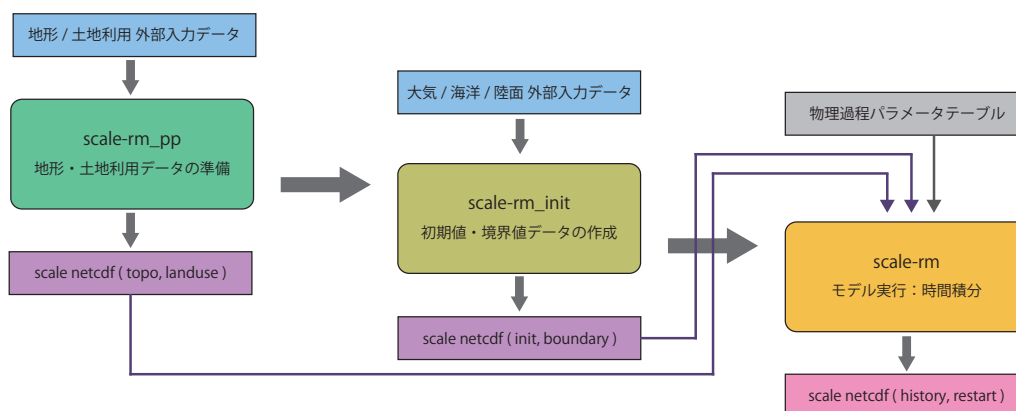


図 3.2.1: SCALE-RM におけるモデルの実行手順

表 3.2.1: 実験設定の概略

項目	設定
MPI プロセス分割 (東西 x 南北)	2 x 2 (合計 4 プロセス)
水平格子数 (東西 x 南北)	90 x 90
鉛直層数	36 層
水平格子間隔	$\Delta x = \Delta y = 20$ km
積分期間	2007 年 7 月 14 日 18UTC ~ 15 日 00UTC (6 時間積分)
時間ステップ間隔	90 sec (全 240 steps)

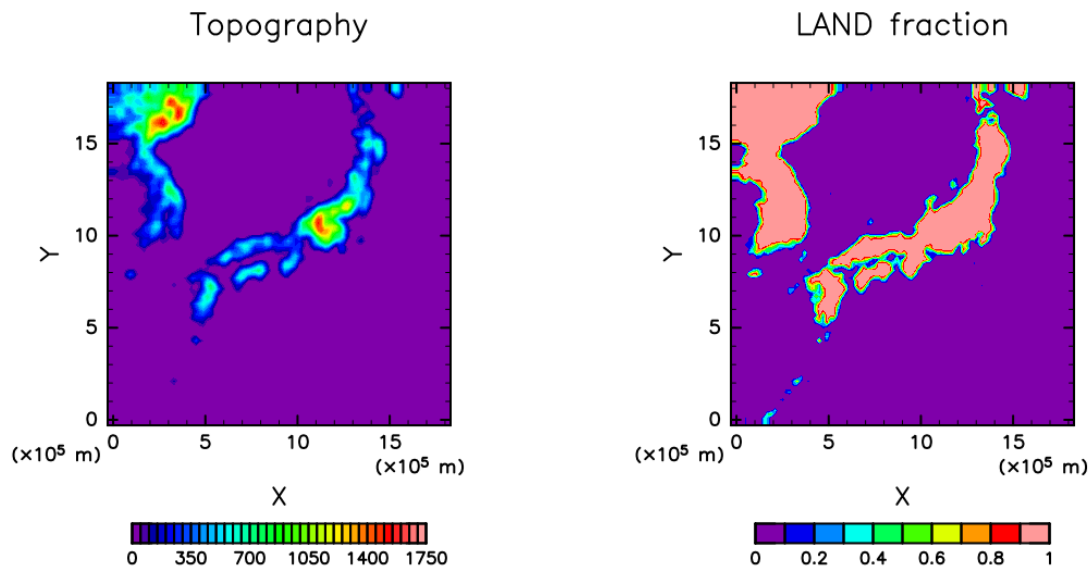


図 3.2.2: 計算領域の地形と海陸分布。

須であり、その他 (黒字) は任意である。

標高データと土地利用区分データ

標高データと土地利用区分データは、各格子点における標高、海陸比率、湖比率、都市被覆率、植生比率、土地 (植生) 利用区分を与えるために必要である。全球中の任意の地域を対象とした計算が行えるように、USGS(U.S. Geological Survey) の標高データ GTOPO30 と、GLCCv2 の土地利用区分データを SCALE-RM では用意している。これらのファイルの形式は、SCALE-RM 用に変換済みである。

1. データベースのダウンロード

SCALE-RM 用に形式を変換した標高・土地利用区分のデータを http://scale.aics.riken.jp/download/scale_database.tar.gz から取得し、任意のディレクトリに展開する。

```
$ tar -zxvf scale_database.tar.gz
```

表 3.2.2: 現実大気実験に必要な外部入力データの項目

SCALE-RM の地形と土地利用を作成するための元データ
<p>標高データ</p> <p>土地利用区分データ</p>
SCALE-RM の初期値境界値を作成するための外部入力データ (一般的には GCM データ)
<p>親モデルの緯度・経度情報</p> <p>— 3 次元大気データ —</p> <p>東西風速、南北風速、気温、比湿 (相対湿度)、気圧、ジオポテンシャル高度</p> <p>— 2 次元大気データ —</p> <p>海面更正気圧、地上気圧、10 m 東西風速、10 m 南北風速、2 m 気温、2 m 比湿 (相対湿度)</p> <p>— 2 次元陸面データ —</p> <p>親モデルの海陸マップ</p> <p>地表面温度 (Skin temp)</p> <p>親モデル土壌データの深さ情報、土壌温度、土壌水分 (体積含水率 or 飽和度)</p> <p>— 2 次元海面データ —</p> <p>海面水温 (Skin temp を SST にも用いる場合には省略可)</p>

```
$ ls
scale_database/topo/    <- 標高データ
scale_database/landuse/ <- 土地利用区分データ
```

2. パスの設定

現実大気実験に必要なファイル一式を準備するために、「実験用セッソー式作成ツール」を用いる。このツールを利用するためには、上記のデータベースが存在するディレクトリの名前 (以後 `SCALE_DB` と表記) を環境変数 `SCALE_DB` に設定しておかなければならない。

```
$ export SCALE_DB="${path_to_directory_of_scale_database}/scale_database"
```

ここで、`path_to_directory_of_scale_database` は、標高・土地利用区分データを含む tar ファイルの展開先ディレクトリの名前である。例えば、`scale_database.tar.gz` を展開したディレクトリの絶対パスが `/home/user/scale` であった場合は、以下のように設定する。

```
$ export SCALE_DB="/home/user/scale/scale_database"
```

大気・陸面・海面水温データ

初期値/境界値データは、4 byte バイナリ (GrADS 形式。以降「binary」形式と表記) に変換すれば読み込める。上述したように「バイナリ」データはユーザーが用意する。ただし、本チュートリアルでは、この「バイナリ」データを準備するためのプログラムをディレクトリ `SCALE_DIR/tutorial/tools/` に用意してある。手順を以下で説明する。ただし、grib1 形式の NCEP FNL (Final) Operational Global Analysis data を使用するために、`wgrib`^{*1)} はインストール済みであるとする。

*1) <http://www.cpc.ncep.noaa.gov/products/wesley/wgrib.html>

1. データの取得

NCAR のサイト <http://rda.ucar.edu/datasets/ds083.2/> から、2007 年 7 月 14 日 18 時から 12 時間分のデータをダウンロードし、`${Tutorial_DIR}/real/tools/FNL_input/grib1/2007` に配置する。以下が、grib1 形式のデータのリストである。

```
fnl_20070714_18_00.grib1
fnl_20070715_00_00.grib1
```

2. データ形式を grib 形式からバイナリ形式に変換

`${Tutorial_DIR}/real/tools/` にある `convert_FNL-grib2grads.sh` を実行する。

```
$ cd ${Tutorial_DIR}/real/tools/
$ sh convert_FNL-grib2grads.sh 2007071418 2007071500 FNL_input FNL_output
```

成功すれば、下記のファイルが作成される。

```
$ ls FNL_output/*/*
FNL_output/200707/FNL_ATM_2007071418.grd
FNL_output/200707/FNL_ATM_2007071500.grd
FNL_output/200707/FNL_LND_2007071418.grd
FNL_output/200707/FNL_LND_2007071500.grd
FNL_output/200707/FNL_SFC_2007071418.grd
FNL_output/200707/FNL_SFC_2007071500.grd
```

NCEP-FNL のデータ構造や変数名が変更された場合には、意図したファイルが作成されない。その場合は、使用する NCEP-FNL データに応じて `convert_FNL-grib2grads.sh` を修正する必要がある。

3.2.3 実験セットの準備

現実大気実験では、理想実験と比べて多くの実行手順やファイルが必要である。加えて、前処理 (pp)、初期値作成 (init)、シミュレーション実行 (run) で使用する設定ファイル (***.conf) 内の実験設定は整合的でなければならない。準備段階におけるファイルの不足や設定の不一致は、モデルが正常に動かない原因となる。このような状況を回避するために、必要なファイルの一式を生成するためのツール「実験用セット一式作成ツール」が用意されている。まず始めに以下のディレクトリに移動し、次の手順によって現実大気実験のチュートリアルに必要なファイルの一式を用意する。

```
$ cd $Tutorial_DIR/real/
$ ls
Makefile : 実験セット一式作成のための Makefile
README   : スクリプトの使用に関する README
USER.sh  : 実験設定の記述
config/  : 一連のファイルの作成に対する各々の設定
          (基本的に、ユーザは書き換える必要はない)
sample/  : USER.sh のサンプルスクリプト
```

```
data/      : チュートリアルのためのツール類
tools/     : チュートリアル用の初期条件のためのツール
            (チュートリアルの場合を除いて、基本的に各自で準備する)
```

```
$ make
$ ls experiment/      : このディレクトリは make により追加される
    init/
    net2g/
    pp/
    run/
```

make を実行すると、USER.sh に記述された設定に従って experiment ディレクトリの下に実験セットが作成される。実験用セッソー式作成ツールに関する詳しい説明については、第 4.2.1 節を参照されたい。

3.2.4 地形データの作成：pp

pp ディレクトリへ移動し、実験のための地形データを以下のように作成する。

```
$ cd ${Tutorial_DIR}/real/experiment/pp/
$ ls
    Makefile
    pp.d01.conf
    scale-rm_pp
```

pp ディレクトリの中には、pp.d01.conf という名前の設定ファイルが準備されている。計算領域の位置や格子点数等の実験設定に応じて、pp.d01.conf を適宜編集する必要がある。本チュートリアルでは pp.d01.conf は編集済みであるので、そのまま利用すれば良い。表 3.2.1 に実験設定を示す。

pp.d01.conf のネームリストの中で、計算領域に関係する設定は `[PARAM_PRC_CARTESC]`、`[PARAM_GRID_CARTESC_INDEX]`、`[PARAM_GRID_CARTESC]` で行っている。X 方向、Y 方向ともに領域は 2 分割されており、総計として 4 つの MPI プロセスを使用する。1 つの MPI プロセスあたりの格子点数は、`(IMAX = 45)`、`(JMAX = 45)` と指定されており、は X 方向、Y 方向それぞれの総格子点数は $90 (= 2 \times 45)$ である。各方向の格子幅は `[PARAM_GRID_CARTESC]` の `(DX, DY)` において 20,000 m (20 km) と指定されている。したがって、計算領域の一辺の長さは 90×20 km であるので、計算領域は $1800 \text{ km} \times 1800 \text{ km}$ の正方領域である。


```
&PARAM_PRC_CARTESC
  PRC_NUM_X = 2,
  PRC_NUM_Y = 2,
  PRC_PERIODIC_X = .false.,
  PRC_PERIODIC_Y = .false.,
/

&PARAM_INDEX_GRID_CARTESC_INDEX
  KMAX = 36,
  IMAX = 45,
  JMAX = 45,
/

&PARAM_GRID_CARTESC
  DX = 20000.0,
  DY = 20000.0,
  FZ(:) = 80.841, 248.821, 429.882, 625.045, 835.409, 1062.158,
          1306.565, 1570.008, 1853.969, 2160.047, 2489.963, 2845.575,
          3228.883, 3642.044, 4087.384, 4567.409, 5084.820, 5642.530,
          6243.676, 6891.642, 7590.074, 8342.904, 9154.367, 10029.028,
          10971.815, 11988.030, 13083.390, 14264.060, 15536.685, 16908.430,
          18387.010, 19980.750, 21698.615, 23550.275, 25546.155, 28113.205,
  BUFFER_DZ = 5000.0,
  BUFFER_DX = 400000.0,
  BUFFER_DY = 400000.0,
/
```

scale-rm_pp 専用のネームリストとして [PARAM_CONVERT] がある。(CONVERT_TOPO) を .true. にすると標高データが処理され、(CONVERT_LANDUSE) を .true. にすると土地利用区分データが処理がされる。

```
&PARAM_CONVERT
  CONVERT_TOPO = .true.,
  CONVERT_LANDUSE = .true.,
/
```

また、[PARAM_CNVTPOPO_GTOP030] の中の (GTPO030_IN_DIR) と [PARAM_CNVLANDUSE_GLCCv2] の中の (GLCCv2_IN_DIR) はそれぞれ、標高データと土地利用区分データの場所を指定している。

```
&PARAM_CNVTPOPO_GTOP030
  GTPO030_IN_DIR = "./topo/GTOP030/Products",
  GTPO030_IN_CATALOGUE = "GTPO030_catalogue.txt",
/

&PARAM_CNVLANDUSE_GLCCv2
  GLCCv2_IN_DIR = "./landuse/GLCCv2/Products",
  GLCCv2_IN_CATALOGUE = "GLCCv2_catalogue.txt",
  limit_urban_fraction = 0.3D0,
```

上記の設定ファイルの準備後に、以下のコマンドによって scale-rm_pp を実行し、地形データを

作成する。

```
$ mpirun -n 4 ./scale-rm_pp pp.d01.conf
```

本チュートリアルでは、表 3.2.1 に示すように 4 つの MPI プロセスを使用する。ジョブが正常に終了すれば、ログファイル (pp_LOG_d01.pe000000) の最後に

```
++++++ finalize MPI...
++++++ MPI is peacefully finalized
```

と出力される。また、topo_d01.pe#####.nc (約 310KB のファイルサイズ) と landuse_d01.pe#####.nc (約 380KB のファイルサイズ) というファイルが MPI プロセス数だけ生成される (今の場合 4 つずつ)。ここで、#####には MPI プロセスの番号が入る。これらのファイルには、各格子点における標高、海陸比率、湖比率、都市被覆率、植生比率、土地 (植生) 利用区分の情報が格納されている。

OPTION

「gpview」がインストールされている場合は、次のコマンドによって地形データが正しく作成されているかを確認できる。

```
$ gpview topo_d01.pe00000*@TOP0 --aspect=1 --nocont
$ gpview landuse_d01.pe00000*@FRAC_LAND --aspect=1 --nocont
```

結果が正常であれば、図 3.2.2 と同様の図が表示される。

3.2.5 初期値/境界値データの作成 : init

init ディレクトリに移動し、SCALE-RM によるシミュレーションに必要な初期値/境界値データを作成する。

```
$ cd ${Tutorial_DIR}/real/experiment/init
$ ls
  Makefile
  init.d01.conf
  init.launch.conf
  param.bucket.conf
  scale-rm_init
```

ディレクトリの中には、設定ファイル init.d01.conf が準備されている。他に init.launch.conf というファイルも存在するが、ここでは使用しない。init.d01.conf ファイルには表 3.2.1 に示すチュートリアル用の設定が既になされているが、pp.d01.conf と同様に実験設定に応じて変更されたい。初期値/境界値データの生成には、前節で作成した地形データが使用される。地形データは、init.d01.conf において相対パスで設定する。

```
&PARAM_TOPO
  TOPO_IN_BASENAME = "../pp/topo_d01",
/

&PARAM_LANDUSE
  LANDUSE_IN_BASENAME = "../pp/landuse_d01",
/
```

その他に、init.d01.conf の設定の中で特に確認して欲しい項目は、[\[PARAM_MKINIT_REAL_ATMOS\]](#)、[\[PARAM_MKINIT_REAL_OCEAN\]](#)、[\[PARAM_MKINIT_REAL_LAND\]](#) の項目である。

<pre>&PARAM_MKINIT_REAL_ATMOS NUMBER_OF_FILES = 2, FILETYPE_ORG = "GrADS", BASENAME_ORG = "namelist.grads_boundary.FNL.grib1", BASENAME_BOUNDARY = "boundary_d01", BOUNDARY_UPDATE_DT = 21600.0, PARENT_MP_TYPE = 3, USE_FILE_DENSITY = .false., / &PARAM_MKINIT_REAL_OCEAN 略 INTRP_OCEAN_SFC_TEMP = "mask", INTRP_OCEAN_TEMP = "mask", / &PARAM_MKINIT_REAL_LAND 略 USE_FILE_LANDWATER = .true., INTRP_LAND_TEMP = "mask", INTRP_LAND_WATER = "fill", INTRP_LAND_SFC_TEMP = "fill", /</pre>	<p>読み込むファイルの数 表 4.1.1 から選択する</p> <p>境界値データの出力名 入力データの時間間隔</p> <p>親モデルの大気密度データを使うか</p> <p>SST の欠測値処理方法 SST の欠測値処理方法</p> <p>親モデルの土壌水分データを使うか 土壌温度の欠測値処理方法 土壌水分の欠測値処理方法 地表面温度の欠測値処理方法</p>
---	--

気象場データのファイル形式は ([FILETYPE_ORG](#)) で指定する。ここでは、GrADS 形式のデータを読み込むために “grads” を与えている。入力ファイルの詳細は第 4.1.1 節を参照されたい。

第 3.2.2 節でバイナリ形式に変換した入力データ (FNL) へのリンクを、作業ディレクトリに張る。リンクを適切に張るために、`${Tutorial_DIR}/real/data` の中に「gradsinput-link_FNL.sh」というスクリプトを用意している。

```
$ cp ../../data/gradsinput-link_FNL.sh ./
$ sh gradsinput-link_FNL.sh
```

上記のコマンドを実行し、下記のリンクが作成されていれば成功である。

```
ATM_00000.grd -> ../tools/FNL_output/200707/FNL_ATM_2007071418.grd
ATM_00001.grd -> ../tools/FNL_output/200707/FNL_ATM_2007071500.grd
LND_00000.grd -> ../tools/FNL_output/200707/FNL_LND_2007071418.grd
LND_00001.grd -> ../tools/FNL_output/200707/FNL_LND_2007071500.grd
SFC_00000.grd -> ../tools/FNL_output/200707/FNL_SFC_2007071418.grd
SFC_00001.grd -> ../tools/FNL_output/200707/FNL_SFC_2007071500.grd
```

次に、GrADS 形式のバイナリデータを SCALE で読み込むためのネームリストファイルを、ディレクトリ `init` へリンクする。

```
$ ln -s ../../data/namelist.grads_boundary.FNL.2005053112-2016051106 ./
```

上記の準備が完了したら、4 つの MPI プロセスを使用して `scale-rm_init` を実行する。

```
$ mpirun -n 4 ./scale-rm_init init.d01.conf
```

正常にジョブが終了すれば、以下のファイルが生成される。

```
$ ls
boundary_d01.pe000000.nc
boundary_d01.pe000001.nc
boundary_d01.pe000002.nc
boundary_d01.pe000003.nc
init_d01_20070714-180000.000.pe000000.nc
init_d01_20070714-180000.000.pe000001.nc
init_d01_20070714-180000.000.pe000002.nc
init_d01_20070714-180000.000.pe000003.nc
init_LOG_d01.pe000000
```

`init_LOG_d01.pe000000` はログファイルであり、処理が正常に完了していれば、ファイルの最後に

```
+++++ finalize MPI...
+++++ MPI is peacefully finalized
```

が出力される。`boundary_d01.pe#####.nc` は境界値データ、`init_d01_20070714-180000.000.pe#####.nc` は初期値データであり、各ファイルのサイズは約 18.9 MB、約 12.6 MB である。ここで、##### は MPI プロセス番号を表す。

OPTION

「`gpview`」がインストールされている場合は、以下のコマンドによって初期値と境界値が正しく作成されているかを確認できる。

```
$ gpvect --scalar --slice z=1500 --nocont --aspect=1 --range=0.002:0.016 --int 0.001 \
--xintv=10 --yintv=10 --unit_vect init_d01_20070714-180000.000.pe00*@QV \
init_d01_20070714-180000.000.pe00*@MOMX init_d01_20070714-180000.000.pe00*@MOMY \
--title "QV, MOMX, MOMY"
```

処理が正常に終了していれば、図 3.2.3 と同様の図が得られる。

3.2.6 シミュレーションの実行：run

`run.conf` の準備

`run` ディレクトリへ移動する。

```
$ cd ${Tutorial_DIR}/real/experiment/run
```

QV, MOMX, MOMY

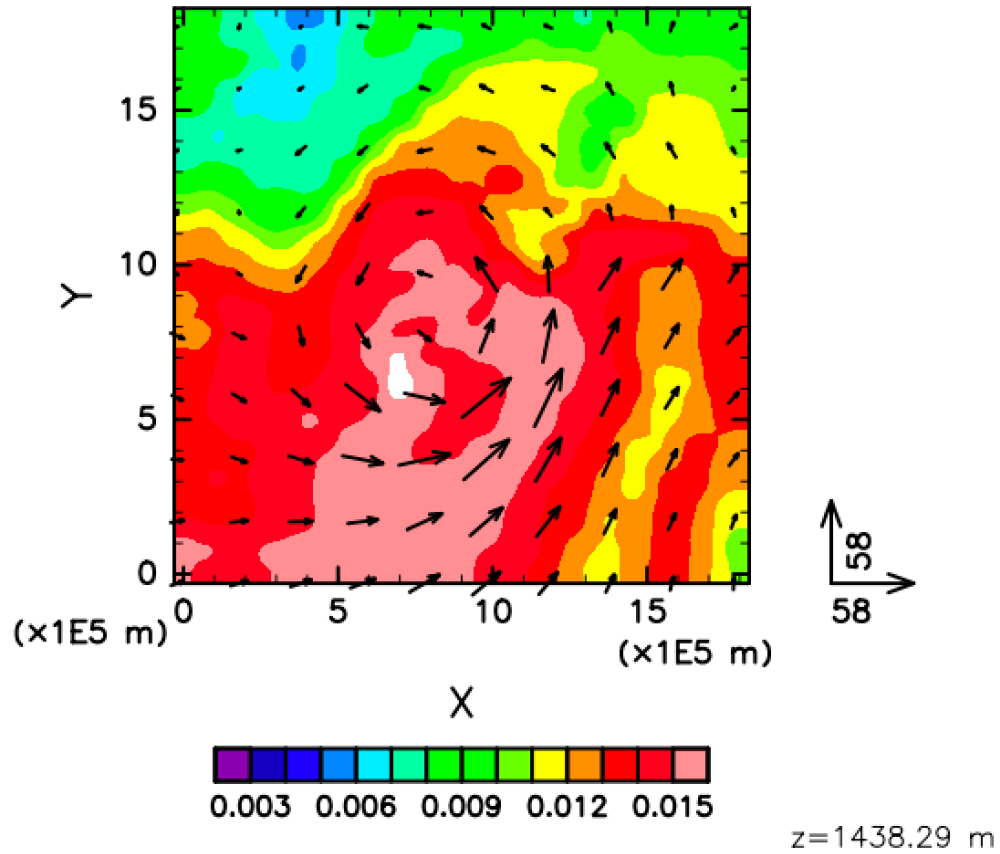


図 3.2.3: チュートリアル実験における初期場の様子 ($z = 1500$ m)。色は比湿、ベクトルは水平運動量フラックスを表す。

このディレクトリの中には、表 3.2.1 に示すチュートリアル用の設定を施した設定ファイルが準備されている。他に `run.launch.conf` というファイルも存在するが、ここでは使用しない。

モデル本体の実行には、事前に作成した地形データや初期値/境界値データを使用する。これらのファイルの指定は、`run.d01.conf` における下記の部分で設定している。

```
&PARAM_TOPO
  TOPO_IN_BASENAME = "../pp/topo_d01",
/

&PARAM_LANDUSE
  LANDUSE_IN_BASENAME = "../pp/landuse_d01",
/

&PARAM_RESTART
  RESTART_OUTPUT = .true.,
  RESTART_OUT_BASENAME = "restart_d01",
  RESTART_IN_BASENAME = "../init/init_d01_20070714-180000.000",
/

&PARAM_ATMOS_BOUNDARY
  ATMOS_BOUNDARY_TYPE = "REAL",
  ATMOS_BOUNDARY_IN_BASENAME = "../init/boundary_d01",
  ATMOS_BOUNDARY_START_DATE = 2007, 7, 14, 18, 0, 0,
  ATMOS_BOUNDARY_UPDATE_DT = 21600.0,
  ATMOS_BOUNDARY_USE_DENS = .true.,
  ATMOS_BOUNDARY_USE_QHYD = .false.,
  ATMOS_BOUNDARY_ALPHAFACDENS = 1.0,
  ATMOS_BOUNDARY_LINEAR_H = .false.,
  ATMOS_BOUNDARY_EXP_H = 2.0,
/
```

run.d01.conf の中で、時間積分に関する設定は [PARAM_TIME] で行う。初期時刻は (TIME_STARTDATE) に UTC で指定し、チュートリアルでは 2007 年 7 月 14 日 18 時 UTC に設定している。積分時間は (TIME_DURATION) で与える。物理過程に対する時間ステップは、各物理スキームごとに設定できる。

```
&PARAM_TIME
  TIME_STARTDATE = 2007, 7, 14, 18, 0, 0,      時間積分を開始する時刻
  TIME_STARTMS = 0.D0,
  TIME_DURATION = 6.0D0,                        : 積分期間
  TIME_DURATION_UNIT = "HOUR",                  : TIME_DURATION の単位
  TIME_DT = 90.0D0,                             : トレーサー移流計算の時間ステップ
  TIME_DT_UNIT = "SEC",                         : TIME_DT の単位
  TIME_DT_ATMOS_DYN = 45.0D0,                   : トレーサー移流計算以外の力学過程の時間ステップ
  TIME_DT_ATMOS_DYN_UNIT = "SEC",              : TIME_DT_ATMOS_DYN の単位

  ..... 略 .....

/
```

計算結果の出力に関する設定は、(PARAM_HISTORY) で行う。

&PARAM_FILE_HISTORY

```
FILE_HISTORY_DEFAULT_BASENAME = "history_d01", : 出力するファイル名
FILE_HISTORY_DEFAULT_TINTERVAL = 3600.D0,      : 出力時間間隔
FILE_HISTORY_DEFAULT_TUNIT = "SEC",             : 出力時間間隔の単位
FILE_HISTORY_DEFAULT_TAVERAGE = .false.,
FILE_HISTORY_DEFAULT_DATATYPE = "REAL4",
FILE_HISTORY_DEFAULT_ZCOORD = "model",          : 鉛直内挿は適用しない
FILE_HISTORY_OUTPUT_STEP0 = .true.,             : 初期時刻 (t=0) の値を出力するかどうか
/
```

上記の設定に従って、下記の (HISTITEM) に列挙した変数を出力する。必要があれば、(HISTITEM) においてオプション変数を加えることで、変数毎に出力間隔を変更できる。また、瞬間値の代わりに平均値を出力することも可能である。これらの詳細は、4.2.8 を参照されたい。

&HISTOTRY_ITEM item="MSLP" /	海面更正気圧
&HISTOTRY_ITEM item="PREC" /	降水強度 (2 次元)
&HISTOTRY_ITEM item="OLR" /	外向き赤外放射 (2 次元)
&HISTOTRY_ITEM item="U10" /	地表 10m での X 方向水平速度成分 (2 次元)
&HISTOTRY_ITEM item="V10" /	地表 10m での Y 方向水平速度成分 (2 次元)
&HISTOTRY_ITEM item="T2" /	地表 2m での温度 (2 次元)
&HISTOTRY_ITEM item="Q2" /	地表 2m での水蒸気比湿 (2 次元)
&HISTOTRY_ITEM item="SFC_PRES" /	地表気圧 (2 次元)
&HISTOTRY_ITEM item="SFC_TEMP" /	バルクの地表面温度 (2 次元)
&HISTOTRY_ITEM item="DENS" /	密度 (3 次元)
&HISTOTRY_ITEM item="QV" /	水蒸気比湿 (3 次元)
&HISTOTRY_ITEM item="QHYD" /	全凝結物の全質量に対する比 (3 次元)
&HISTOTRY_ITEM item="PRES" /	圧力 (3 次元)
&HISTOTRY_ITEM item="U" /	X 方向水平速度成分 (3 次元)
&HISTOTRY_ITEM item="V" /	Y 方向水平速度成分 (3 次元)
&HISTOTRY_ITEM item="T" /	温度 (3 次元)
&HISTOTRY_ITEM item="W" /	鉛直方向速度成分 (3 次元)
&HISTOTRY_ITEM item="Uabs" /	風速 (3 次元)
&HISTOTRY_ITEM item="PT" /	温位 (3 次元)
&HISTOTRY_ITEM item="RH" /	相対湿度 (3 次元)

力学過程や物理過程に対するスキームとして他のスキームを用いたい場合は、力学過程に関しては [&PARAM_ATMOS_DYN]、物理過程に関しては [PARAM_ATMOS ,PARAM_OCEAN ,PARAM_LAND ,PARAM_URBAN] で設定できる。詳細は、第 4.3.1 節、4.4 節を参照されたい。

シミュレーションの実行

実行に必要なファイルは下記であり、あらかじめ用意されている。

```
$ ls
MIPAS  PARAG.29  PARAPC.29  VARDATA.RM29  cira.nc
                                : 放射スキーム用のパラメータファイル
run.d01.conf      : 設定ファイル
param.bucket.conf : 陸面スキーム用のパラメータファイル
scale-rm          : SCALE-RM の実行バイナリ
run.launch.conf   : ネスティング計算用の launch ファイル
```

(チュートリアルでは使用しない)

準備が整ったら、4-MPI 並列により SCALE-RM を実行する。

```
$ mpirun -n 4 ./scale-rm run.d01.conf >& log &
```

実行が完了するまでには、ある程度時間を要する (推奨環境において 10 ~ 20 分程度かかる)。そのため、上記のように標準出力をファイルに書き出すようにして、バックグラウンドで実行すると便利である。計算は進みながら、途中経過のログは "LOG_d01.pe000000" に出力される。ジョブが正常に終了すると、"LOG_d01.pe000000" の最後に

```
+++++ finalize MPI...
+++++ MPI is peacefully finalized
```

と出力され、下記のファイルが作成される。

```
$ ls
history_d01.pe000000.nc
history_d01.pe000001.nc
history_d01.pe000002.nc
history_d01.pe000003.nc
```

各ファイルのサイズは約 34 MB である。出力ファイル (history_d01.pe#####.nc) は、MPI プロセス数に応じて分割されている。ここで、##### は MPI プロセス番号を表す。これらのファイルには、(HISTORY_ITEM) で指定した変数が出力されている。出力ファイルの形式は、気候・予報 (CF) メタデータ規約に準拠した NetCDF である。

3.2.7 結果のクイック描画：net2g

本節では、netcdf2grads の使用方法を説明する。プログラム netcdf2grads (略して net2g) は、プロセス毎に分割された netCDF ファイル (history.**.nc ^{*2)}) を GrADS 形式の単一のバイナリファイルに結合する。この変換した GrADS バイナリデータを使って、シミュレーションの結果を確認する。

GrADS バイナリに変換

プロセスごとに分割された netCDF 形式のヒストリファイルから GrADS バイナリに変換するために、net2g を使用する。ここでは最低限の手順のみを説明することにする。詳細な使用方法は [4.6.1 節](#) を参照されたい。

まず、net2g ディレクトリに移動する。

```
$ cd ${Tutorial_DIR}/real/experiment/net2g
$ ls
Makefile
net2g -> ../../../../util/netcdf2grads_h/net2g
net2g.2D.d01.conf
net2g.3D.d01.conf
```

^{*2)} 「gpview」がインストールされている場合には、「gpview」でも作図することができる。このツールは history データを変換することなく直接作図することができるため、素早く結果を確認した場合には適している。

このディレクトリの中には設定ファイルとバイナリファイルがある。バイナリファイルは、[2.3.1 節](#)でコンパイルした実行ファイルにリンクされている。ここでは例として、2次元変数の MSLP、PREC を GrADS 形式に変換する手順を示す。また、3次元変数の U、V を 850 hPa 面、500hPa 面、200 hPa 面で抽出して、GrADS 形式に変換する手順も説明する。2次元変数と3次元変数のための設定ファイルはそれぞれ、net2g.2D.d01.conf と net2g.3D.d01.conf である。

net2g を実行する時のプロセス数は、シミュレーションの実行時に用いたプロセス数の約数である必要がある。ここでは、4 プロセスを使用する。net2g は 2次元変数と3次元変数を同時に変換できないために、以下のように別々に実行する。

```
$ mpirun -n 4 ./net2g net2g.2D.d01.conf
$ mpirun -n 4 ./net2g net2g.3D.d01.conf
```

エラーメッセージがなく、下記のメッセージだけが標準出力へ表示されていれば、変換は正常に完了している。

```
+++ MPI COMM: Corrective Finalize
```

成功すれば、下記のファイルが作成される。**ctl は SCALE-RM の XY 格子系に対する ctl ファイル、**lccr.ctl は緯度経度座標系で結果を作図するための ctl ファイルである。

```
MSLP_d01z-2d.ctl
MSLP_d01z-2d.grd
MSLP_d01z-2d_lccr.ctl
PREC_d01z-2d.ctl
PREC_d01z-2d.grd
PREC_d01z-2d_lccr.ctl
PRES_d01z-3d.ctl
PRES_d01z-3d.grd
PRES_d01z-3d_lccr.ctl
U_d01z-3d.ctl
U_d01z-3d.grd
U_d01z-3d_lccr.ctl
V_d01z-3d.ctl
V_d01z-3d.grd
V_d01z-3d_lccr.ctl
```

計算結果の確認

GrADS スクリプト checkfig_real.gs を用いて、計算結果を確認する。

```
$ cp ../../data/checkfig_real.gs ./
$ grads -blc checkfig_real.gs
```

変換が正常に終了すれば、下記のファイルが作成される。なお、GrADS のバージョンによって文法が異なるので、警告が出る場合はスクリプトを適宜変更されたい。

real_mslp.png

real_prec.png

real_wind.png

計算が成功していれば、図 [3.2.4](#), [3.2.5](#), [3.2.6](#) と同じ図が得られる。

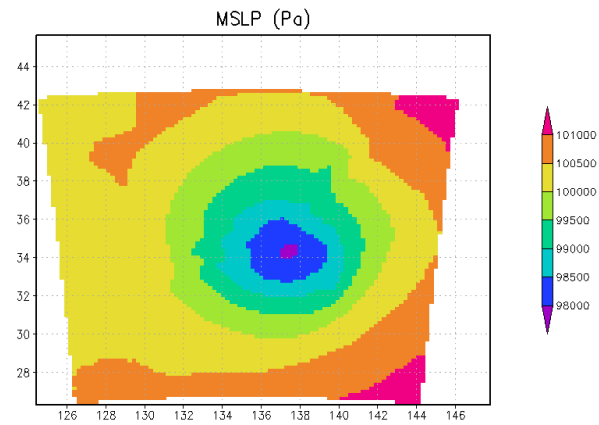


図 3.2.4: 計算開始から 6 時間後の海面更正気圧

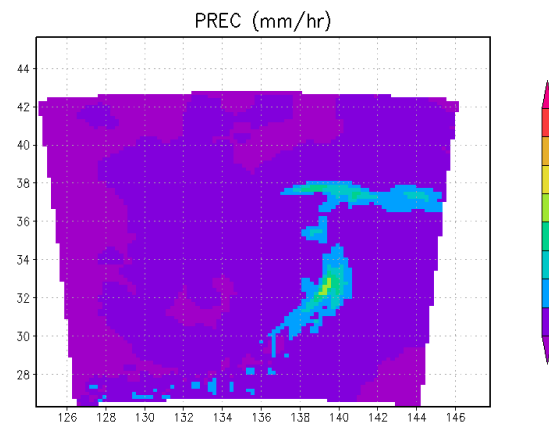


図 3.2.5: 計算開始から 6 時間後の降水フラックス

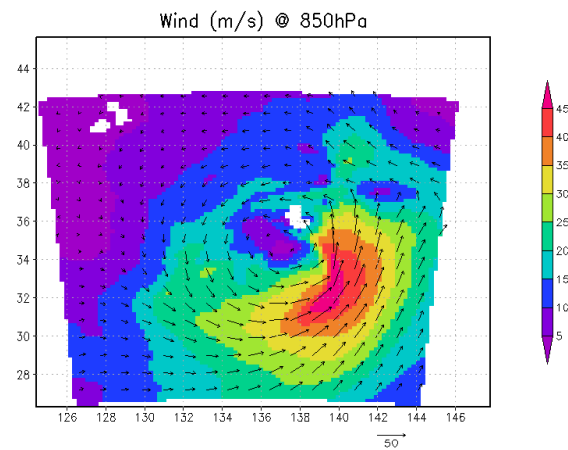


図 3.2.6: 計算開始から 6 時間後の 850hPa の風速 (色は絶対値、ベクトルは向き)

第4部

各種設定

4.1 前処理

4.1.1 初期値/境界値データの作成方法

表 4.1.1: SCALE で対応している外部入力データ

データ形式	FILETYPE_ORG	備考
SCALE データ形式	SCALE-RM	ヒストリファイルのみ対応。latlon カタログを必要とする。
バイナリ形式	GrADS	データ読み込み用のネームリストを別途必要とする。
WRF データ形式	WRF-ARW	「wrfout_」, 「wrfst」の両方に対応する。

SCALE-RM では、表 4.1.1 に示される様々な種類の外部データを読み込むことによって、初期値データや境界値データを作成できる。プログラム `scale-rm_init` は、ファイル `init.conf` の設定に従って外部データを初期値・境界値データに変換する。入力データの形式は、`[PARAM_MKINIT_REAL_***]` の `(FILETYPE_ORG)` で指定する。

SCALE データ形式は主にオフライン・ネスティング実験で使用される。詳細については、第 4.2.10.2 節を参照されたい。

WRF データ形式も使用でき、WRF によるモデル出力データを直接使用できる。ただし、ファイルは SCALE-RM の境界値データの作成に必要な全てのデータを含まなければならない。

本書における「バイナリデータ形式」は、Fortran が直接アクセスできる単精度浮動小数点のバイナリデータとして定義される。GRIB/GRIB2 データは、バイナリデータ形式に変換することで SCALE-RM に読み込ませることができる。この方法は、第 3.2.2 節で説明される。その他の任意のデータについても、バイナリデータ形式に変換すれば使用できる。

SCALE の最新版の出力ファイル形式は、バージョン 5.2 以前の形式とは異なる。そのため、バージョン 5.2 以前で作成された初期値/境界値ファイルは SCALE 5.3.3 では使用できない。

バイナリ形式データの入力

入力データの形式は、設定ファイル `init.conf` の `[PARAM_MKINIT_REAL_***]` で以下のように指定する。

```
&PARAM_RESTART
  RESTART_OUTPUT = .true.,
  RESTART_OUT_BASENAME = "init_d01",
/

&PARAM_MKINIT_REAL_ATMOS
  NUMBER_OF_FILES = 2,
  FILETYPE_ORG = "GrADS",
  BASENAME_ORG = "namelist.grads_boundary.FNL.grib1",
  BASENAME_BOUNDARY = "boundary_d01",
  BOUNDARY_UPDATE_DT = 21600.0,
  PARENT_MP_TYPE = 3,
  USE_FILE_DENSITY = .false.,
/

&PARAM_MKINIT_REAL_OCEAN
  NUMBER_OF_FILES = 2,
  FILETYPE_ORG = "GrADS",
  BASENAME_ORG = "namelist.grads_boundary.FNL.grib1",
  INTRP_OCEAN_SFC_TEMP = "mask",
  INTRP_OCEAN_TEMP = "mask",
/

&PARAM_MKINIT_REAL_LAND
  NUMBER_OF_FILES = 2,
  FILETYPE_ORG = "GrADS",
  BASENAME_ORG = "namelist.grads_boundary.FNL.grib1",
  USE_FILE_LANDWATER = .true.,
  INTRP_LAND_TEMP = "mask",
  INTRP_LAND_WATER = "fill",
  INTRP_LAND_SFC_TEMP = "fill",
/
```

バイナリデータを読み込むときは、(FILETYPE_ORG) に "GrADS" を設定する。SCALE-RM では ファイル名やバイナリデータの構造を含むネームリストファイル `namelist.grads_boundary**` を「ctl」ファイルの代わりに準備する。

(NUMBER_OF_FILES) は入力ファイルの数である。単一の入力ファイルの場合は、ファイル「ファイル名.grd」のみを準備する。複数の入力ファイルの場合には、時間の進む方向に「ファイル名.XXXXX.grd」と名前をつけたファイルを準備する。プログラム `scale-rm_init` は、00000 から (NUMBER_OF_FILES)-1 までの数字を付けたファイルを読み込む。入力ファイルのヘッダー名（つまり「ファイル名」）はネームリストファイルで指定されるが、これについては後で説明する。

(BOUNDARY_UPDATE_DT) は入力データの時間間隔である。変換された初期値ファイルのヘッダー名は、[PARAM_RESTART] の (RESTART_OUT_BASENAME) で設定する。(BASENAME_BOUNDARY) は、変換された境界値ファイルのヘッダー名である。(BASENAME_BOUNDARY) を指定しなければ、境界値ファイルは出力されない。

以上の設定は、[PARAM_MKINIT_REAL_ATMOS]、[PARAM_MKINIT_REAL_OCEAN]、[PARAM_MKINIT_REAL_LAND] の間で共通である。[PARAM_MKINIT_REAL_OCEAN] や [PARAM_MKINIT_REAL_LAND] を別途指定しない限り、これらの情報は引き継がれる。(USE_FILE_DENSITY) は FILETYPE_ORG="SCALE-RM" とした場合のオプションであり、バイナリデータを選択した場合は .false. を与えなければならない。(PARENT_MP_TYPE) は親モデルの水物質カテゴリーの種類であるが、バイナリデータを読み込む場合は 3 を設定しなければならない。

土壌水分の設定は、親モデルからデータを与える方法と、領域全体で一定値を与える方法の 2

種類ある。前者の場合は、3次元の土壌水分データを必要とする。後者の場合は、`init.conf` の `[PARAM_MKINIT_REAL_LAND]` に `USE_FILE_LANDWATER = .false.` を設定する。また、土壌水分の条件は、土壌空隙率に対する水が占める割合 (飽和度) として `INIT_LANDWATER_RATIO` で指定する。デフォルト値は 0.5 である。また、単位体積あたりの土壌の隙間の大きさ (空隙率) は土地利用に応じて変わる。

```
&PARAM_MKINIT_REAL_LAND
  USE_FILE_LANDWATER = .false.   土壌水分をファイルから読むかどうか。デフォルトは.true.
  INIT_LANDWATER_RATIO = 0.5     USE_FILE_LANDWATER=.false. の場合、
                                  空隙率に対する水が占める割合 (飽和度)。
..... 略.....
/
```

バイナリデータ (GrADS 形式) を入力ファイルに用いる場合は、ユーザが用意する。その形式については、GrADS の Web ページ (<http://cola.gmu.edu/grads/gadoc/aboutgriddeddata.html#structure>) を参照されたい。「ctl」ファイルの代わりに、データのファイル名やデータ構造を SCALE-RM に与えるためのネームリストファイル (`namelist.grads_boundary**`) の一例を下記に示す。

```
#
# Dimension
#
&nml_grads_grid
  outer_nx = 360, ; 大気データの x 方向の格子数
  outer_ny = 181, ; 大気データの y 方向の格子数
  outer_nz = 26, ; 大気データの z 方向の層数
  outer_nl = 4, ; 土壌データの層数
/

#
# Variables
#
&grdvar item='lon', dtype='linear', swpoint=0.0d0, dd=1.0d0 /
&grdvar item='lat', dtype='linear', swpoint=90.0d0, dd=-1.0d0 /
&grdvar item='plev', dtype='levels', lnum=26,
  lvars=100000,97500,...(省略)...,2000,1000, /
&grdvar item='MSLP', dtype='map', fname='FNLsfc', startrec=1, totalrec=6 /
&grdvar item='PSFC', dtype='map', fname='FNLsfc', startrec=2, totalrec=6 /
&grdvar item='U10', dtype='map', fname='FNLsfc', startrec=3, totalrec=6 /
&grdvar item='V10', dtype='map', fname='FNLsfc', startrec=4, totalrec=6 /
&grdvar item='T2', dtype='map', fname='FNLsfc', startrec=5, totalrec=6 /
&grdvar item='RH2', dtype='map', fname='FNLsfc', startrec=6, totalrec=6 /
&grdvar item='HGT', dtype='map', fname='FNLatm', startrec=1, totalrec=125 /
&grdvar item='U', dtype='map', fname='FNLatm', startrec=27, totalrec=125 /
&grdvar item='V', dtype='map', fname='FNLatm', startrec=53, totalrec=125 /
&grdvar item='T', dtype='map', fname='FNLatm', startrec=79, totalrec=125 /
&grdvar item='RH', dtype='map', fname='FNLatm', startrec=105, totalrec=125, knum=21 /
&grdvar item='llev', dtype='levels', lnum=4, lvars=0.05,0.25,0.70,1.50, /
&grdvar item='lsmask', dtype='map', fname='FNLland', startrec=1, totalrec=10 /
&grdvar item='SKINT', dtype='map', fname='FNLland', startrec=2, totalrec=10 /
&grdvar item='STEMP', dtype='map', fname='FNLland', startrec=3, totalrec=10,
  missval=9.999e+20 /
&grdvar item='SMOISVC', dtype='map', fname='FNLland', startrec=7, totalrec=10,
  missval=9.999e+20 /
```

大気データの格子数を `outer_nx`, `outer_ny`, `outer_nz` で指定し、土壌データ (STEMP、SMOISVC) の層数を `outer_nl` で指定する。

QV や RH のデータは大気上層でしばしば提供されていない。そのような場合はデータが存在する層数を `knum` で指定する。上層での値の与え方として 2 種類の方法を用意している。デフォルトでは、以下のように `upper_qv_type = "ZERO"` である。

```
&PARAM_MKINIT_REAL_GrADS
  upper_qv_type = "ZERO"    "ZERO": QV=0
                           "COPY": 湿度の入力データが存在する最上層の RH を、データが存在し
                           ない上層にコピーする
/
```

[`grdvar`] の設定は、表 4.1.2 に示すようにデータによって異なる。表 4.1.3 において、体積含水率は土の体積 V の中に占める水の体積 V_w の割合 (V_w/V) である。また、飽和度は V の中に占める間隙の体積 V_v に対する水の体積 V_w の割合 (V_w/V_v) である。[`PARAM_MKINIT_REAL_LAND`] の (`USE_FILE_LANDWATER`) が `true` である場合は、SMOISVC か SMOISDS のどちらかを用意する必要

がある。

表 4.1.2: [grdvar] の変数

grdvar の項目	説明	備考
item	変数名	表 4.1.3 より選択
dtype	データ形式	"linear", "levels", "map"から選択
(dtype) が"linear"の場合のネームリスト ("lon", "lat"専用)		
swpoint	スタートポイントの値	
dd	増分	
(dtype) が"levels"の場合のネームリスト ("plev", "llev"専用)		
lnum	レベルの数 (層数)	
lvars	各層の値	
(dtype) が"map"の場合のネームリスト		
fname	ファイル名の頭	
startrec	変数 (item) のレコード番号	t=1 の時刻の値
totalrec	一時刻あたりの全変数のレコード長	
knum	3次元データの層数	(オプション) outer_nz と異なる場合。 RH と QV のみ使用可。
missval	欠陥値の値	(オプション)

表 4.1.3: [grdvar] の (item) の変数リスト。アスタリスクは「オプションであるが、可能な限り推奨される」ことを意味する。二重のアスタリスクは、「利用できるが、推奨されない」ことを意味する。高高度の場所を対象とする場合は、HGT を用いることを強く推奨する。

変数名	説明	単位	(dtype)
lon	経度データ	[deg.]	linear, map
lat	緯度データ	[deg.]	linear, map
plev	気圧データ	[Pa]	levels, map
* HGT	高度 (ジオポテンシャル) データ	[m]	map
** DENS	air density	[kg/m ³]	map
U	東西風速	[m/s]	map
V	南北風速	[m/s]	map
** W	鉛直風速	[m/s]	map
T	気温	[K]	map
RH	相対湿度 (QV がある場合は省略可)	[%]	map
QV	比湿 (RH がある場合は省略可)	[kg/kg]	map
** QC	雲水の質量比	[kg/kg]	map
** QR	雨水の質量比	[kg/kg]	map
** QI	雲氷の質量比	[kg/kg]	map
** QS	雪の質量比	[kg/kg]	map
** QG	霰の質量比	[kg/kg]	map
* MSLP	海面更正気圧	[Pa]	map
* PSFC	地上気圧	[Pa]	map
* U10	10m 東西風速	[m/s]	map
* V10	10m 南北風速	[m/s]	map
* T2	2m 気温	[K]	map
* RH2	2m 相対湿度 (Q2 がある場合は省略可)	[%]	map
* Q2	2m 比湿 (RH2 がある場合は省略可)	[kg/kg]	map
* TOPO	GCM の地形	[m]	map
* lsmask	GCM の海陸分布	0:海 1:陸	map
SKINT	地表面温度	[K]	map
llev	土壌の深さ	[m]	levels
STEMP	土壌温度	[K]	map
SMOISVC	土壌水分 (体積含水率)	[-]	map
	(SMOISDS がある場合は省略可)		
SMOISDS	土壌水分 (飽和度)	[-]	map
	(SMOISVC がある場合は省略可)		
SST	海面温度 (SKINT がある場合は省略可)	[K]	map

4.2 SCALE-RMのフレームワーク

4.2.1 設定ファイルを用意するための補助ツール

実験のための設定ファイル`***.conf`は、`pp`、`init`、`run`用にそれぞれ用意する必要がある。ネームリストのいくつかの項目はこれらの設定ファイル間で共通でなければならず、不整合がある場合にはモデルは適切に動かない。そのような失敗を避けるために、設定ファイルの準備を行うための便利な補助ツール (実験用セトー式作成ツール) を、下記のように用意している。

```
$ cd ${Tutorial_DIR}/real/
$ ls
  Makefile
    : 実験に必要なファイルを生成するための Makefile
  README
    : README ファイル
  USER.sh
    : 設定を指定するためのシェルスクリプト
  config/
    : 各々の設定に対する設定ファイル (ユーザは書き換える必要はない)
  sample
    : USER.sh のサンプルスクリプト
  data
    : 現実大気実験のチュートリアル用のファイル
  tools
    : 現実大気実験のチュートリアル用のファイル。FNL データを grib 形式からバイナリ形式に変換するときに使われる。
```

本ツールの初期設定は現実大気実験のチュートリアルに合わせているが、ユーザーは `USER.sh` で設定を変更できる。

`sample/` ディレクトリの下に、いくつかのサンプルスクリプトを典型的な設定例として用意している。必要に応じて、これらの内容を `USER.sh` にコピーして使用すると良いだろう。

```
$ ls sample/
  USER.default.sh
    : 現実大気実験のチュートリアル用の USER.sh と同じ (シングルドメイン用)。
  USER.offline-nesting-child.sh
    : オフライン・ネスティングによる実験の子領域用。
  USER.offline-nesting-parent.sh
    : オフライン・ネスティングによる実験の親領域用。
  USER.online-nesting.sh
    : オンライン・ネスティング用。
```

ツールの使い方

使い方は README に書かれているように、以下の通りである。

1. ユーザが希望する実験設定に従って、USER.sh を編集する。
2. make コマンドを実行する。

これにより、experiment ディレクトリ以下に実験に必要な設定ファイル一式が作成される。

USER.sh の設定は現実大気のチュートリアルに対する設定になっているので、以下のようにチュートリアル設定を別ファイルとして残しておくことを勧める。

```
$ mv experiment/ tutorial/
    : (既に experiment ディレクトリがある場合)
$ cp USER.sh USER_tutorial.sh
... USER.sh を編集 ...
$ make
$ cp -rL experiment 任意の場所/
    : 「任意の場所」は、任意の場所にあるディレクトリの名前を意味する。
```

USER.sh の編集

スクリプトファイルの最初の方に、ドメインの数を指定する NUM_DOMAIN がある。その下には、スクリプトが生成する設定ファイルに指定される項目が並んでいて、これらの項目を適切に変更すると良いだろう。項目の後に「# required parameters for each domain」というコメントが存在する場合には、項目の値をドメインの数だけスペースで区切って書く。項目の値の数と NUM_DOMAIN で設定したネスティングドメインの数が異なれば、実験用のファイル一式が作成されないことに注意が必要である。USER.sh にない項目については、experiment ディレクトリ以下に作成された設定ファイルを直接編集されたい。

4.2.2 旧バージョンの設定ファイルの変換プログラム

SCALE のバージョンが 5.2 から 5.3 になったときに、ネームリストのパラメータに対して数多くの変更がなされた。そのため、設定ファイルをバージョン 5.2 用から 5.3 用に変換するためのプログラムを用意している。プログラムの実行には、「ruby」(<https://www.ruby-lang.org/en/>)が必要である。

変換プログラムの使用方法是以下である。

```
$ ruby scale-5.3.3/utils/config-converter/config-converter_5.2-5.3.rb \\  
    old.conf > new.conf
```

4.2.3 対象計算領域の設定

本節では、格子数、計算領域とその MPI プロセスとの関係を説明する。計算領域は、水平格子間隔と格子点数および MPI プロセス数によって決定される。この関係を図 4.2.1 に示す。水平方向に 2 次元の領域分割を行うことで並列化がなされている。

格子点数および MPI プロセス数は、`[PARAM_ATMOS_GRID_CARTESC_INDEX]` 内の `(IMAX, JMAX)`、`[PARAM_PRC_CARTESC]` 内の `(PRC_NUM_X, PRC_NUM_Y)` で設定する。図 4.2.1 に示すように、計算領域全体は、X 方向に `(PRC_NUM_X)` 個、Y 方向に `(PRC_NUM_Y)` 個に分割される。プロセス数はゼロから始まり、左下から右上の順で番号付けされる (図 4.2.1 における矢印)。分割された各領域は 1 つの MPI プロセスによって担当され、各 MPI プロセスは `(IMAX, JMAX, KMAX)` 個の格子ブロックを受け持つ。ここで注意すべきことは、「指定する格子点数は各プロセスが受け持つ値」であり、計算領域全体の格子点数ではないことである。別の言い方をすれば、計算領域全体は、水平格子間隔、格子点数、MPI プロセス数に依存する。

したがって、各方向の格子点と総格子点数は、以下のように計算される。

$$\text{領域内 X 方向の格子数} = \text{IMAX} \times \text{PRC_NUM_X} \quad (4.2.1)$$

$$\text{領域内 Y 方向の格子数} = \text{JMAX} \times \text{PRC_NUM_Y} \quad (4.2.2)$$

$$\text{領域内の総格子数} = (\text{IMAX} \times \text{PRC_NUM_X}) \times (\text{JMAX} \times \text{PRC_NUM_Y}) \times (\text{KMAX})$$

ここで、`(KMAX)` は鉛直方向の格子点数であり、`[PARAM_ATMOS_GRID_CARTESC_INDEX]` 内で指定する。

また、領域全体の大きさは、式 (4.2.1、4.2.2) を使って、

$$\text{X 方向の領域の長さ} = \text{X 方向の格子点数} \times \text{DX}$$

$$\text{Y 方向の領域の長さ} = \text{Y 方向の格子点数} \times \text{DY}$$

と決定される。ここで、第 4.2.3.3 節で記述したように、`(DX, DY)` は `[PARAM_ATMOS_GRID_CARTESC]` 内で指定する。水平方向の解像度と領域の大きさを設定し、利用する MPI プロセス数を与えた場合は、1 つの MPI プロセスが受け持つ格子点数は上記の関係から決められる。

次の小節では、MPI プロセス数、格子数、格子間隔の設定をより詳しく説明する。これらの設定は `pp.conf`、`init.conf`、`run.conf` の設定ファイル間で一致させなければならないことに注意が必要である。

4.2.3.1 MPI プロセス数

MPI プロセス数は設定ファイルの `[PARAM_PRC_CARTESC]` 内で指定する。SCALE-RM の入出力ファイルは MPI プロセス毎に分割されているため、分割ファイルの総数は MPI プロセス数によって変化する。例えば、2-MPI 並列用に作成した初期値ファイルは、4-MPI 並列のモデル実行には使用できない。MPI プロセス数を変更する場合は、`pp.conf`、`init.conf`、`run.conf` 内の `[PARAM_PRC_CARTESC]` を編集し、`pp`、`init` の過程を再度行う必要がある。

```
&PARAM_PRC
  PRC_NUM_X = 2,    ; X 方向 (東西方向) の MPI 並列分割数
  PRC_NUM_Y = 1,    ; Y 方向 (南北方向) の MPI 並列分割数
/
```

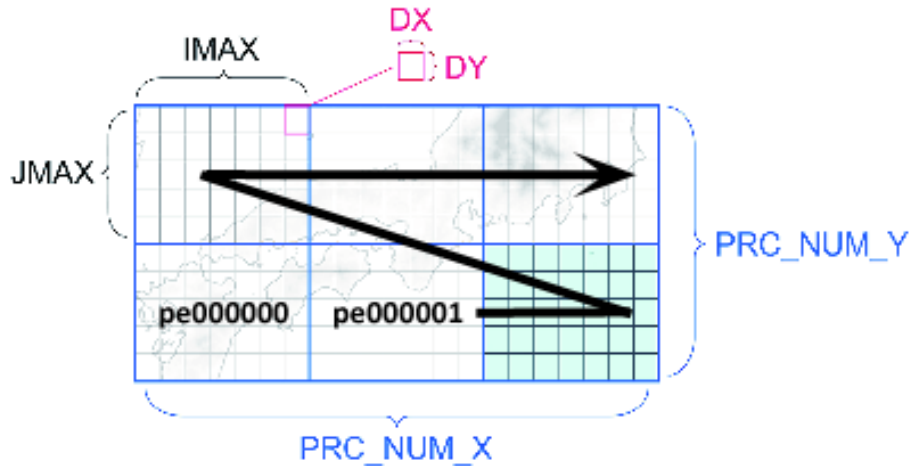


図 4.2.1: 計算領域全体に対する、水平格子間隔 (DX, DY)、MPI プロセスあたりの格子数 (IMAX, JMAX)、MPI プロセス数 (PRC_NUM_X, PRC_NUM_Y) の関係。水色の領域は、1 つの MPI プロセスが担当する領域に対応する。

MPI プロセスの総数は、 $PRC_NUM_X \times PRC_NUM_Y$ によって与えられる。上記の例では、X 方向に領域を 2 分割し、Y 方向には領域を分割しないので、2-MPI 並列ということになる。ジョブを投げる際の MPI コマンドに指定する MPI プロセス数として、この総プロセス数を与える必要がある。この条件を満たさない場合には、プログラムは計算を行わずに直ちに終了し、下記のメッセージが標準出力に出力される。

```
xxx total number of node does not match that requested. Check!
```

4.2.3.2 水平・鉛直格子数

格子数は、設定ファイル (***.conf) の [PARAM_ATMOS_GRID_CARTESC_INDEX] で指定する。指定する水平格子数は、1 つの MPI プロセス当たりの値であることに注意が必要である。

```
&PARAM_ATMOS_GRID_CARTESC_INDEX
  KMAX = 97,           ; 鉛直層数
  IMAX = 20,           ; プロセスあたりの X 方向の格子点数
  JMAX = 25,           ; プロセスあたりの Y 方向の格子点数
/
```

4.2.3.3 水平・鉛直格子間隔

第 4.2.3.5 節で述べる緩和領域を除いて水平格子間隔は等間隔でのみ設定できるが、鉛直格子間隔は任意に設定できる。全方向について格子間隔を等間隔で設定する場合には、[PARAM_ATMOS_GRID_CARTESC] 内の (DX, DY, DZ) にそれぞれ、東西、南北、鉛直方向の格子間隔を指定する (単位は [m])。

```
&PARAM_ATMOS_GRID_CARTESC
DX = 500.D0,           ; X 方向（東西方向）の格子間隔
DY = 500.D0,           ; Y 方向（南北方向）の格子間隔
DZ = 500.D0,           ; Z 方向（鉛直方向）の格子間隔
/
```

以下の囲みに、等間隔でない格子系を指定する方法を示す。モデルはローレンツ格子を採用しており、速度ベクトルと他のスカラー量に対する定義点は半格子分ズれている。ここでは、スカラー量の位置をセンターポイントと呼び、それから半格子ズれた位置をフェイスポイントと呼ぶ。鉛直格子点の位置を直接指定する場合は、`[PARAM_ATMOS_GRID_CARTESC]` 内の `(FZ(:))` に配列として与えればよい^{*1)}。詳細は図 4.2.2 を参照されたい。また、`(FZ(:))` で指定する要素の数は、鉛直層数 (`[PARAM_ATMOS_GRID_CARTESC_INDEX]` 内の `(KMAX)`) と一致している必要があることに注意が必要である。例として、理想実験の設定ファイルを下記に示す。

```
&PARAM_ATMOS_GRID_CARTESC
DX = 500.D0,           X 方向 の格子間隔 (等間隔) [m]
DY = 500.D0,           Y 方向の格子間隔 (等間隔) [m]
FZ(:) =                Z 方向のフェイスポイントの位置 [m]
  80.0000000000000000 ,
  168.00000190734863 ,
  264.80000610351567 ,
  ~ 中略 ~
  14910.428862936289 ,
  15517.262523292475 ,
  16215.121232702089 ,
  17017.658748523147 ,
  17940.576891717363 ,
  19001.932756390710 ,
  20222.492000765058 ,
  BUFFER_DZ = 5000.D0,   第 4.2.3.5 節参照
  BUFFFACT = 1.0D0,      第 4.2.3.5 節参照
/
```

上記の設定は、標高 0 m の場所で処理される。地形が存在する場所での鉛直格子点の位置は、地形に沿った座標系によって適切に取り扱われる。

鉛直格子点の位置は任意に設定できるが、変わった設定をすると計算不安定がしばしば生じる。これを避けるために、ディレクトリ `texttttscale-5.3.3/scale-rm/util/makevgrid/` の中に `make_vgrid.f90` という Fortran プログラムと幾つかのネームリストの例を用意している。必要があれば参考として用いることができる。このツールは `(FZ(:))` の値を直接出力するので、コピーして設定ファイルに貼り付けると良いだろう。

4.2.3.4 スポンジ層

SCALE-RM は高度座標を採用している。最上層の境界条件は剛体壁であり、モデル上端において音波や重力波がしばしば反射する。反射波の悪影響を軽減するために、モデル領域の上部に「スポンジ層」と呼ばれる減衰層を設ける。スポンジ層内では、レイリー摩擦によって鉛直速度を減衰させ

^{*1)}この場合には、シミュレーションで用いられたものと同じ浮動小数点の精度を用いることが望ましい。デフォルトでは、モデルは倍精度の浮動小数点を使用するとしてコンパイルされる。

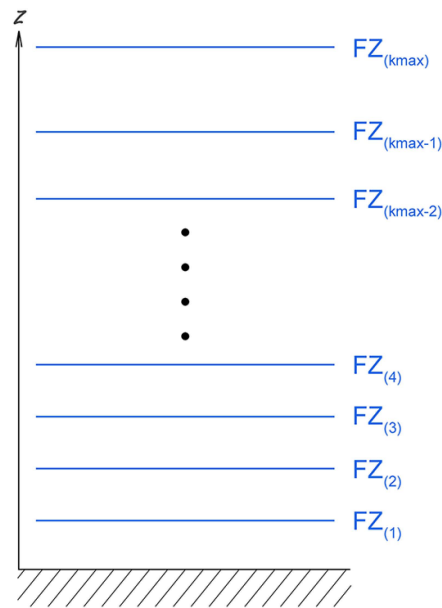


図 4.2.2: SCALE-RM におけるフェイスポイントの定義点。[PARAM_ATMOS_GRID_CARTESC] で (FZ) を指定する時は、 $k = 1$ での値として第 1 層目上端の高さを与える。 $k = 1$ での値は地表面の高さでないことに注意が必要である。

る。緩和の時定数 (e-folding time) はモデル上端で最も短く、高度が下がるにつれて長くする。スポンジ層の下端より下では、緩和の時定数は無限大に設定する。スポンジ層の厚さの指定方法は 2 種類あり、[PARAM_ATMOS_DYN] で設定する。

1. スポンジ層の層数を指定

(ATMOS_DYN_wdamp_layer) でスポンジ層の層数を指定する。この層数はモデル上端から数える。

2. スポンジ層の下端高度 [m] の指定

(ATMOS_DYN_wdamp_height) で指定した高度よりも上部にある層を、スポンジ層として設定する。

デフォルトでは上記のパラメータは両方とも設定されず、スポンジ層は適用されない、両方が指定された場合は、(ATMOS_DYN_wdamp_layer) が優先される。

スポンジ層上端での緩和の時定数は、(ATMOS_DYN_wdamp_tau) で設定する (単位は秒)。

これには (TIME_DT_ATMOS_DYN) よりも小さな値は設定できない。時定数を陽に指定しない場合は、(TIME_DT_ATMOS_DYN) の 10 倍の値が自動で設定される。(TIME_DT_ATMOS_DYN) については、第 4.2.7 節を参照されたい。また、具体的な設定例は、第 4.3.1.2 節を参照されたい。

4.2.3.5 緩和領域と境界ナッジング

一般に水平境界では、境界条件として与えられる入力データと実際の計算で得られる出力データの間に値の不一致が起こる。計算において、この不一致は非物理的なモード等の幾つかの問題を生じさせる。この問題を回避するために、領域内に「緩和領域」を設ける。

図 4.2.3 に示すように、SCALE-RM では計算領域のすぐ内側に緩和領域を設置する。緩和領域では、境界値データや親モデルのデータによって指定した値にある時定数で近づけるように、予報変数を更新する。この緩和を以下ではナッジングと呼ぶ。

緩和領域

緩和領域の幅は、設定ファイルの [PARAM_ATMOS_GRID_CARTESC] の中で設定する。この設定は全ての手順で共通していなければならないことを再度注意する。緩和領域の幅を設定する方法は、以下の 2 種類がある。

1. (BUFFER_NX, BUFFER_NY, BUFFER_NZ) によって緩和領域とする格子数を指定
2. (BUFFER_DX, BUFFER_DY, BUFFER_DZ) によって緩和領域の幅 (参考値)[m] を指定

デフォルトでは上記のパラメータは両方とも指定されず、緩和領域は設定されない。また、両方が指定された場合は、格子数による指定が優先される。水平方向には東西南北の四方境界に緩和領域が設定されるが、鉛直方向には領域上端にのみ緩和領域が設定され、下端には設定されない。緩和領域は計算領域の内側に設定されるため、ナッジングの影響を受けない実際の対象領域 (緩和領域を除いた範囲) は計算領域よりも狭くなることに注意が必要である。

以下に 2 種類の設定例を示す。

```
&PARAM_ATMOS_GRID_CARTESC
BUFFER_NX = 30,           ; X 方向 (東西方向) の緩和領域の格子数
BUFFER_NY = 30,           ; Y 方向 (南北方向) の緩和領域の格子数
BUFFFACT = 1.00,         ; 全方向の緩和領域内の格子間隔に対するストレッチ係数 (デフォルトは 1.0)
/
```

```
&PARAM_ATMOS_GRID_CARTESC
BUFFER_DZ = 5000.00,      ; Z 方向 (モデル上端から下向き方向) の緩和領域の幅 (参考値) [m]
BUFFER_DX = 300000.00,    ; X 方向 (東西方向) の緩和領域の幅 (参考値) [m]
BUFFER_DY = 300000.00,    ; Y 方向 (南北方向) の緩和領域の幅 (参考値) [m]
BUFFFACT_Z = 1.2000,      ; Z 方向の緩和領域内の格子間隔に対するストレッチ係数
BUFFFACT_X = 1.0500,      ; X 方向の緩和領域内の格子間隔に対するストレッチ係数
BUFFFACT_Y = 1.0500,      ; Y 方向の緩和領域内の格子間隔に対するストレッチ係数
/
```

X 方向の緩和領域の設定方法を以下で説明する。緩和領域の格子数 *ibuff* は、(BUFFER_NX) に等しい。(BUFFER_NX) を用いずに (BUFFER_DX) で指定した場合は、*ibuff* は

$$\sum_{n=1}^{ibuff} BDX(n) \geq BUFFER_DX$$

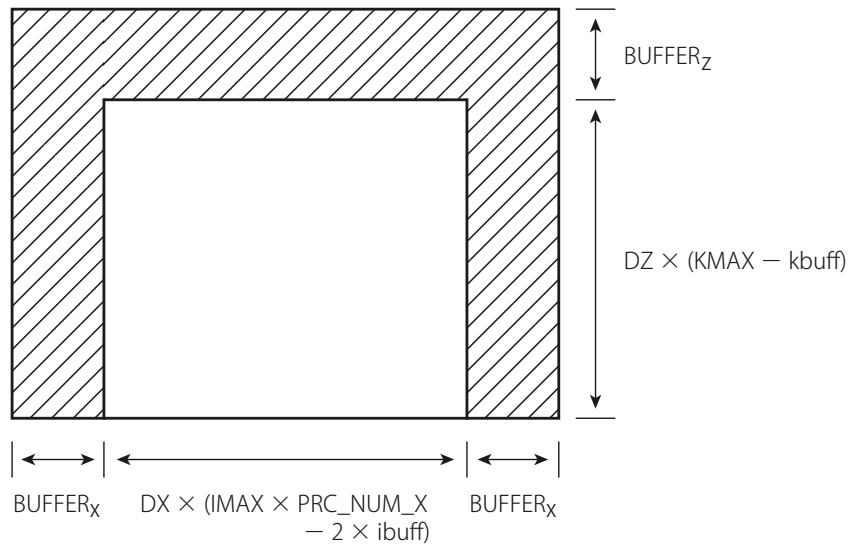


図 4.2.3: 全計算領域内での緩和領域の配置：斜線部分が緩和領域を意味する。図は XZ 断面を示しているが、YZ 断面についても同様である。

の関係を満たす最小の整数であるように自動的に計算される。したがって、緩和領域の幅 $BUFFER_X$ ($= \sum_{n=1}^{ibuff} BDX(n)$) は $(BUFFER_DX)$ と一致するとは限らないことに注意が必要である。最後に、緩和領域を除いた計算領域の大きさは、

$$DX \times (IMAX \times PRC_NUM_X - 2 \times ibuff)$$

と表現される。Y 方向、Z 方向についても同様に設定されるが、Z 方向 の実際の対象領域は、

$$DZ \times (KMAX - kbuff)$$

と表現される。ここで、 $kbuff$ はモデル上端の緩和領域の格子数である。

一般に、緩和領域の幅の設定や格子の置き方には明確な指標はない。これらの設定は解く問題に依存する。SCALE-RM では、モデル上部における鉛直方向の緩和領域の格子点数は 5 点以上、側面境界の緩和領域の格子点は 20 ~ 40 点程度を推奨している。実験によっては、さらに緩和領域の格子点を増やしたり、適切なストレッチ係数を用いて緩和領域自体を広げたり、緩和の時定数を調整したりする必要があるだろう。緩和の時定数については、以下で説明する。

緩和領域の格子間隔は、デフォルトでは $[PARAM_ATMOS_GRID_CARTESC]$ 中の (DX, DY, DZ) で指定した通りである。ただし、 $(BUFFFACT)$ に 1 以上に設定することによって、ストレッチさせることもできる。格子間隔を等間隔で指定した場合は、この $(BUFFFACT)$ の設定は全方向に対して適用される。各方向で別々に設定したい場合は、 $(BUFFFACT_X, BUFFFACT_Y, BUFFFACT_Z)$ を指定する。 $(FZ(:))$ を与えることで鉛直レベルを指定した場合 (第 4.2.3.3 節参照) は、上記のストレッチの設定は Z 方向には適用されない。

緩和領域内の格子間隔 (BDX) は次の通り決定される。

$$BDX(n) = DX \times BUFFFACT^n$$

ここで、 n は緩和領域内の格子点番号を表し、計算領域の内側から外側へ向う番号である。緩和領域の格子間隔は、 $(BUFFFACT=1.0)$ とした場合は内部領域と同じであり、 $(BUFFFACT=1.2)$ とした場合

は内部から外側の領域に向かって 1.2 倍の割合で広がっていく。(BUFFFACT) の値は任意に設定できるが、計算不安定を避けるために 1.0 から 1.2 までの値が推奨される。

最後に、緩和領域の大きさ $BUFFER_X$ は、

$$BUFFER_X = DX \times \frac{BUFFFACT^{i_{buff}} - 1}{BUFFFACT - 1}$$

となる。緩和領域の幅 (BUFFER_DX) を同じに設定した場合でも、(BUFFFACT) を大きくすると緩和領域の格子数は少なくなる。(BUFFER_NX) を与えた場合は、緩和領域の幅のみが変わる。

緩和領域におけるナッジングの方法

緩和領域におけるナッジングの設定は、[PARAM_ATMOS_BOUNDARY] 内のパラメータで行う。境界データの種類の種類は、[PARAM_ATMOS_BOUNDARY] 内の (ATMOS_BOUNDARY_TYPE) によって設定する (表 4.2.1)。

表 4.2.1: 境界値データの種類の選択

値	種類の説明
NONE	ナッジングしない
CONST	指定された定数値にナッジングする
INIT	初期値にナッジングする
OFFLINE	ファイルから読み込んだ値にナッジングする (時間変化なし)
REAL	親モデルまたは親領域の値にナッジングする (時間変化あり)

以下は、[PARAM_ATMOS_BOUNDARY] 内のパラメータである。

&PARAM_ATMOS_BOUNDARY

```

ATMOS_BOUNDARY_TYPE = 'NONE',
ATMOS_BOUNDARY_IN_BASENAME = '',

```

```

ATMOS_BOUNDARY_IN_CHECK_COORDINATES \
= .true.,

```

```

ATMOS_BOUNDARY_START_DATE \
= (/ -9999, 0, 0, 0, 0, 0 /),
ATMOS_BOUNDARY_UPDATE_DT = 0.0D0,
ATMOS_BOUNDARY_INTERP_TYPE \
= 'lerp_initpoint',

```

```

ATMOS_BOUNDARY_OUT_BASENAME = '',
ATMOS_BOUNDARY_OUT_TITLE \
= 'SCALE-RM BOUNDARY CONDITION',
ATMOS_BOUNDARY_OUT_DTYPE = 'DEFAULT',
ATMOS_BOUNDARY_USE_DENS = .false.,
ATMOS_BOUNDARY_USE_VELZ = .false.,
ATMOS_BOUNDARY_USE_VELX = .false.,
ATMOS_BOUNDARY_USE_VELY = .false.,
ATMOS_BOUNDARY_USE_POTT = .false.,
ATMOS_BOUNDARY_USE_QV = .false.,
ATMOS_BOUNDARY_USE_QHYD = .false.,
ATMOS_BOUNDARY_VALUE_VELZ = 0.0D0,
ATMOS_BOUNDARY_VALUE_VELX = 0.0D0,
ATMOS_BOUNDARY_VALUE_VELY = 0.0D0,
ATMOS_BOUNDARY_VALUE_POTT = 300.0D0,
ATMOS_BOUNDARY_VALUE_QTRC = 0.0D0,
ATMOS_BOUNDARY_ALPHAFACDENS = 1.0D0,
ATMOS_BOUNDARY_ALPHAFACVELZ = 1.0D0,
ATMOS_BOUNDARY_ALPHAFACVELX = 1.0D0,
ATMOS_BOUNDARY_ALPHAFACVELY = 1.0D0,
ATMOS_BOUNDARY_ALPHAFACPOTT = 1.0D0,
ATMOS_BOUNDARY_ALPHAFACQTRC = 1.0D0,
ATMOS_BOUNDARY_SMOOTHER_FACT = 0.2D0,
ATMOS_BOUNDARY_FRACZ = 1.0D0,
ATMOS_BOUNDARY_FRACX = 1.0D0,
ATMOS_BOUNDARY_FRACY = 1.0D0,
ATMOS_BOUNDARY_TAUZ = DT * 10.0D0,
ATMOS_BOUNDARY_TAUZ = DT * 10.0D0,
ATMOS_BOUNDARY_TAUZ = DT * 10.0D0,
ATMOS_BOUNDARY_LINEAR_V = .false.,

```

```

ATMOS_BOUNDARY_LINEAR_H = .false.,

```

```

ATMOS_BOUNDARY_EXP_H = 2.0D0,

```

; 境界値データの種類。表 4.2.1 を参照。
; 境界値データのファイル名 (OFFLINE または REAL type の場合)

; 境界データファイル内の座標変数を確認するかのフラグ

; 境界値データの開始時刻 (REAL type の場合のみ)
; 境界値データの時間間隔 (REAL type の場合のみ)

; 時間方向の補間の種類
same_parent: 最新の時間ステップでの値を使用 (補間なし),
nearest_neighbor: 最も近い時間ステップでの値を使用,
lerp_initpoint: 2 つの時間ステップでの値を用いて瞬間値として線形補間する,
lerp_midpoint: lerp_initpoint と同様であるが、境界データの時間ステップ間の時間平均値を使用
; 初期の境界値データを出力するファイル名

; 出力ファイルに対するタイトル
; 出力のデータ型 (REAL4 or REAL8)
; 密度に対するナッジングのスイッチ
; w に対するナッジングのスイッチ.
; u に対するナッジングのスイッチ.
; v に対するナッジングのスイッチ.
; θ に対するナッジングのスイッチ.
; 水蒸気に対するナッジングのスイッチ.
; 水物質に対するナッジングのスイッチ.
; w の値 (CONST type の場合のみ)
; u の値 (CONST type の場合のみ)
; v の値 (CONST type の場合のみ)
; θ の値 (CONST type の場合のみ)
; 水蒸気の値 (CONST type の場合のみ)
; 密度に対する $1/\tau$ の係数.
; w に対する係数.
; u に対する係数.
; v に対する係数.
; θ に対する係数.
; 水蒸気に対する係数.
; 点ごとの差に対する水平方向の平滑化の係数.
; z 方向の緩和領域に対するナッジング領域の割合.
; x 方向の割合.
; y 方向の割合.
; 上端境界でのナッジングの時定数 (秒).
; 東西境界でのナッジングの時定数 (秒).
; 南北境界でのナッジングの時定数 (秒).
; z 方向に関するナッジングの時定数分布の種類. .true. であれば線形分布、そうでなければサイン型の分布.
; x, y 方向に関するナッジングの時定数分布の種類. .true. であれば線形分布、そうでなければ指数関数分布.
; 指数関数分布の場合における指数の係数.

ナッジングによる時間変化率は、

$$\left. \frac{\partial \phi_{k,i,j}}{\partial t} \right|_{\text{nudging}} = -\alpha \Delta \phi_{k,i,j} + \alpha_s \left(\frac{\Delta \phi_{k,i-1,j} + \Delta \phi_{k,i+1,j} + \Delta \phi_{k,i,j-1} + \Delta \phi_{k,i,j+1}}{8} - \frac{\Delta \phi_{k,i,j}}{2} \right), \quad (4.2.3)$$

と書かれる。ここで、 $\Delta \phi$ は境界値データとの差であり、 $\alpha_s = \alpha \times \text{ATMOS_BOUNDARY_SMOOTHER_FACT}$ である。 α は 3 方向に対する係数 $\alpha_x, \alpha_y, \alpha_z$ の最大値である。これらの係数は、以下のような長さスケール e に依存する。

$$e = \max \left(1 - \frac{d}{\text{BUFFER} \times \text{ATMOS_BOUNDARY_FRAC}}, 0 \right), \quad (4.2.4)$$

ここで、 d は境界からの距離である。もし `(ATMOS_BOUNDARY_LINEAR_V)` が `.true.` であれば、

$$\alpha_z = e_z / \tau_z, \quad (4.2.5)$$

`.false.` であれば、

$$\alpha_z = \sin^2(\pi e_z / 2) / \tau_z, \quad (4.2.6)$$

である。ここで、 τ_z は `(ATMOS_BOUNDARY_TAUZ)` である。水平方向については、`(ATMOS_BOUNDARY_LINEAR_H)` が `.true.` であれば、

$$\alpha_x = e_x / \tau_x, \quad (4.2.7)$$

`.false.` であれば、

$$\alpha_x = e_x \exp\{-(1 - e_x) \times \text{ATMOS_BOUNDARY_EXP_H}\} / \tau_x. \quad (4.2.8)$$

である。 α_y は α_x と同様の方法によって導かれる。

τ は境界 ($d = 0$) での緩和時間であり、計算された値と境界値の差はこの時間スケールで $1/e$ 倍となる。他方、式 4.2.4 の右辺 2 項目によって、 $\Delta \phi$ の two-grid スケールの成分は $\tau / \text{ATMOS_BOUNDARY_SMOOTHER_FACT}$ の時間で $1/e$ 倍となる。 τ のデフォルトの値は、`(TIME_DT)` の 10 倍である。`(TIME_DT)` については第 4.2.7 節を参照されたい。

`[PARAM_ATMOS_BOUNDARY]` 内の `(ATMOS_BOUNDARY_TYPE)` が `REAL` であれば、水平速度・温位・水蒸気に対する側面境界でのナッジングが強制的に適用される。この場合は、`(ATMOS_BOUNDARY_USE_VELX)`、`(ATMOS_BOUNDARY_USE_VELY)`、`(ATMOS_BOUNDARY_USE_POTT)`、`(ATMOS_BOUNDARY_USE_QV)` といったスイッチは、上端境界に対してのみ使われる。オンライン・ネスティング (第 4.2.10.3 節を参照) の子領域で計算が行われるときには、境界の種類として `REAL` を用いて上記と同様の設定が適用される。

上端境界の周辺における同様な減衰のさせ方として、レイリー摩擦が存在する (第 4.2.3.4 節を参照)。

4.2.4 地図投影法と計算領域の位置の設定

SCALE-RM では、格子点は実距離に基づいて配置される。各格子点での緯度・経度の値は、基準位置の緯度経度を与えることによって、ある地図投影法から計算される。格子の緯度・経度に関する情報は、SCALE が生成する NetCDF 形式の全出力ファイルに含まれる。計算領域の位置と地図投影法は、(PARAM_MAPPROJECTION) で設定できる。この設定は、[pp.conf](#)、[init.conf](#)、[run.conf](#) の設定ファイル間で一致させなければならない。上記の設定例を以下に示す。

```
&PARAM_MAPPROJECTION
MAPPROJECTION_basepoint_lon = 138.727778D0,
MAPPROJECTION_basepoint_lat = 35.360556D0,
MAPPROJECTION_type = 'MER',           ; 表 4.2.2 から選択.
/
```

表 4.2.2: SCALE-RM で選択可能な地図投影法

MPRJ_type	地図投影法
NONE	地図投影なし (理想実験用)、デフォルト
LC	ランベルト正角円錐図法
PS	ポーラステレオ図法
MER	メルカトル図法
EC	正距円筒図法

(MPRJ_basepoint_lat, MPRJ_basepoint_lon) は、それぞれ基準点の緯度・経度である。デフォルトの設定では、基準点は計算領域の中心である。SCALE-RM では、北緯を正、南緯を負の値として表現し、東経を正、西経を負の値として表現する。経度は 180 度以上の値を用いて表現することができる。上記の設定では、計算領域の中心が北緯 35.360556 度、東経 138.727778 度に設定されている。計算領域の全体は、指定された大きさでこの場所を中心にして配置される。

(MAPPROJECTION_type) は地図投影法の種類を表しており、MER はメルカトル図法を意味する。表 4.2.2 は、SCALE-RM で現在選択できる地図投影法を示している。メルカトル図法の場合には、投射する円筒に接する基準緯線を (MAPPROJECTION_M_lat) で設定する (単位は度)。一般的に基準緯線は赤道にとることが多い。しかし、メルカトル図法は基準緯線に近いほど歪みが少なく正確であるので、SCALE-RM では、(MAPPROJECTION_M_lat) を陽に指定しなければ (MAPPROJECTION_basepoint_lat) を基準緯線として用いる。

次に、地図投影法の中でも利用頻度が高い、ランベルト正角円錐図法の設定を以下で説明する。以下の例は、現実大気実験チュートリアルで使った run.d01.conf ファイル内の記述と同じである。

```
&PARAM_MAPPROJECTION
MAPPROJECTION_basepoint_lon = 135.220404,
MAPPROJECTION_basepoint_lat = 34.653396,
MAPPROJECTION_type = 'LC',
MAPPROJECTION_LC_lat1 = 30.0,
MAPPROJECTION_LC_lat2 = 40.0,
/
```

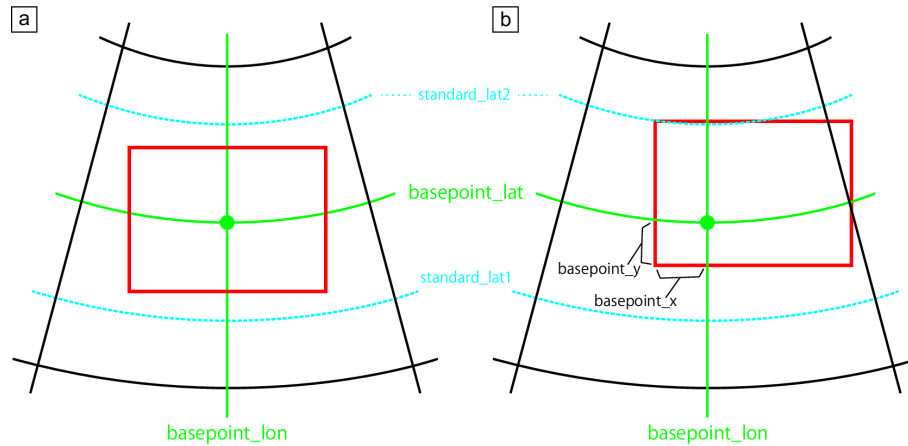


図 4.2.4: 投影中心と計算領域の関係：(a) はデフォルト設定の場合、(b) は投影中心の位置を計算領域中心からずらした場合。赤線は計算領域の境界を表す。

SCALE-RM では、2 標準緯線型の投影方法を採用している。南側、北側の標準緯線はそれぞれ (`MAPPROJECTION_LC_lat1`, `MAPPROJECTION_LC_lat2`) で指定する (単位は [度])。両標準緯線に挟まれた領域では、経線に対する緯線の長さの比が、地球の楕円体面上での長さの比と近くなるように調節される。

さらに下記のように設定すれば、基準点 (`MAPPROJECTION_basepoint_x`, `MAPPROJECTION_basepoint_y`) を、デフォルト設定である計算領域の中心からずらすことができる。

```
&PARAM_MAPPROJECTION
MAPPROJECTION_basepoint_lon = 135.220404,
MAPPROJECTION_basepoint_lat = 34.653396,
MAPPROJECTION_basepoint_x = 100.0,
MAPPROJECTION_basepoint_y = 100.0,
MAPPROJECTION_type = 'LC',
MAPPROJECTION_LC_lat1 = 30.0,
MAPPROJECTION_LC_lat2 = 40.0,
/
```

地図投影の中心位置は、計算領域の南西端 (左下角) からの距離によって指定する。つまり、(`MAPPROJECTION_basepoint_x`, `MAPPROJECTION_basepoint_y`) はそれぞれ、X 方向や Y 方向に対する左下角と基準位置の間の距離である (単位は [m])。これらを指定しない場合は、地図投影の中心は計算領域の中心に設定される。図 4.2.4 に、両方の場合における地図投影の中心と計算領域の関係を示す。

4.2.5 地形の設定

SCALE-RM では地形を表現するために、地形に沿った座標系を採用している。この座標系では、最下層の格子の底面が標高に対して沿うように与えられる。許容される最大の地形傾斜角度 θ_{\max}

[radian] は、次の式で計算する。

$$\theta_{\max} = \arctan(\text{RATIO} \times \text{DZ}/\text{DX})$$

ここで、DZ と DX はそれぞれ、鉛直方向と水平方向の格子間隔である。上記の計算式から分かるように、許容される最大傾斜角度は空間解像度に応じて変わる。RATIO が 1.0 よりも大きければ地形はより細かく表現され、1.0 よりも小さければ粗く表現される。RATIO を非常に大きく設定した場合には、計算が途中で破綻する危険性が高くなることに注意が必要である。SCALE-RM では RATIO のデフォルト値は 1.0 に設定している。

scale-rm_pp は、外部入力する標高データを SCALE 形式に変換するためのプログラムである。詳細な設定は、設定ファイル pp.conf の [PARAM_CNVTOP0] の中で行う。以下に例を示す。

```
&PARAM_CNVTOP0
CNVTOP0_UseGTOPO30 = .true.,      ; GTOPO30 データセットを用いるか?
CNVTOP0_UseDEM50M = .false.,     ; DEM50M データセットを用いるか?
CNVTOP0_UseUSERFILE = .false.,   ; ユーザ定義のデータセットを用いるか?
CNVTOP0_smooth_type = 'LAPLACIAN', ; 平滑化のためのフィルタの種類
                                   (OFF,LAPLACIAN,GAUSSIAN)
CNVTOP0_smooth_maxslope_ratio = 10.D0, ; 許容する傾斜の DZ/DX に対する倍率
CNVTOP0_smooth_maxslope = -1.D0,    ; 許容する傾斜角の最大値 [deg]
CNVTOP0_smooth_local = .true.,      ; 最大傾斜角度を超えた格子でのみ平滑化を続けるかどうか?
CNVTOP0_smooth_itelim = 10000,      ; 平滑化の繰り返し回数の制限値
CNVTOP0_smooth_hypdiff_niter = 20,  ; 超粘性による平滑化の繰り返し回数
CNVTOP0_interp_level = 5,           ; 補間に用いる近隣の格子点数
CNVTOP0_copy_parent = .false.,      ; 子ドメインの緩和領域に親ドメインの地形をコピーするか?
/
```

SCALE-RM では地形データの入力として、国土地理院が提供する GTOPO30 と DEM50M に対応している。プログラム scale-rm_pp によってユーザが準備した地形データを変換できる (第 4.2.6 節を参照)。また、上記のデータセットを組み合わせることもできる。(CNVTOP0_UseGTOPO30) と (CNVTOP0_UseDEM50M) の両方を true に設定した場合は、プログラムは以下のようにデータを作成する。

- GTOPO30 のデータセットを計算領域の格子点に内挿する。
- DEM50M が対象とする領域は、DEM50M のデータセットを用いて内挿し、上書きする。
- 平滑化を適用する。

デフォルトでは、対象とする格子点の周辺にある、入力データの最寄りの 5 格子点が内挿に使われる。使用する格子点数は (CNVTOP0_interp_level) によって決定される。地形のリグリッドにおいて、急な傾斜を含む標高を平滑化するためのフィルタとして、ラプラシアンフィルタとガウシアンフィルタの 2 種類が存在する。これは (CNVTOP0_smooth_type) で選択することができ、デフォルトではラプラシアンフィルタが用いられる。平滑化の操作において、傾斜角が最大許容角度 θ_{\max} を下回るまで、選択されたフィルタが適用される。(CNVTOP0_smooth_maxslope_ratio) を指定することによって、上記の RATIO を直接設定できる。または、度数で最大傾斜角を決める

(CNVTOPO_smooth_maxslope) を用いることができる。平滑化の繰り返し回数の上限はデフォルトでは 10000 回であるが、(CNVTOPO_smooth_itelim) を設定することで繰り返し回数を増やせる。(CNVTOPO_smooth_local) を .true. に設定した場合は、繰り返されるフィルタ操作は平滑化が完了していない格子点でのみ続けられる。

小さな空間スケールのノイズを取り除くために、付加的な超粘性を地形に適用する。シミュレーションにおける数値的なノイズを減らすために、このフィルタリングを行うことを推奨する。(CNVTOPO_smooth_hypdiff_niter) に負の値を設定した場合は、このフィルタは適用されない。

(CNVTOPO_copy_parent) は、ネスティング計算のための設定項目である。一般的に、子ドメインは親ドメインよりも空間解像度が高いために、子ドメインの方が地形がより細かく表現される。このとき、子ドメインの緩和領域における大気データと親ドメインにおける大気データの間の不整合によって、問題がしばしば起きる。この問題を回避するために、(CNVTOPO_copy_parent) を .true. とすれば親ドメインの地形を子ドメインの緩和領域にコピーできる。親ドメインが存在しない場合は (CNVTOPO_copy_parent) を .false. に設定しなければならない。(CNVTOPO_copy_parent) を利用する場合の設定は、第 4.2.10.1 節で詳しく説明する。

4.2.6 ユーザー定義の地形の準備

(CNVTOPO_UseUSERFILE) を .true. に設定した場合は、プログラム scale-rm_pp は [PARAM_CNVTOPO_USERFILE] で指定したファイルの変換を試みる。以下はその設定例である。

```
&PARAM_CNVTOPO_USERFILE
USERFILE_IN_DIR = "./input_topo",           ; 入力ファイルがあるディレクトリのパス
USERFILE_IN_FILENAME = "GTOP030_e100n40.grd", ; 入力ファイルの名前
USERFILE_DLAT = 0.0083333333333333D0,       ; 格子間隔 (緯度,degree)
USERFILE_DLON = 0.0083333333333333D0,       ; 格子間隔 (経度,degree)
USERFILE_IN_DATATYPE = "INT2",               ; データの種類 (INT2,INT4,REAL4,REAL8)
USERFILE_LATORDER_N2S = .true.,              ; データは緯度方向に関して北から南へと格納
                                           ; されているか?
USERFILE_LAT_START = -10.D0,                 ; 格子点の開始位置 (緯度,degree)
USERFILE_LAT_END = 40.D0,                   ; 格子点の終了位置 (緯度,degree)
USERFILE_LON_START = 100.D0,                 ; 格子点の開始位置 (経度,degree)
USERFILE_LON_END = 140.D0,                   ; 格子点の終了位置 (経度,degree)
/
```

この例では、GTOP030_e100n40.grd という名前のデータファイルが、ディレクトリ ./input_topo に存在する。データは、北緯 40 度から南緯 10 度、東経 100 度から東経 140 度までを対象としている。格子間隔は緯度経度ともに 30 秒角である。よって、このデータは緯度方向に 6000 点、経度方向に 4800 点を含む。値は 2 バイトの整数で格納されている。ユーザー定義の地形データは、USERFILE_IN_DATATYPE を除いて GrADS(direct access) 形式と同様な単純なバイナリでなければならない。

4.2.7 積分時間と時間刻み幅の設定

積分時間や時間刻み幅は、実験の目的や設定に応じて適切に設定する必要がある。時間刻み幅は、モデルの空間解像度に依存する。数値不安定を回避するために、より短い時間刻み幅がしばしば要求される。積分時間と時間刻み幅は、シミュレーション実行用の設定ファイルの **[PARAM_TIME]** で設定できる。

&PARAM_TIME	
TIME_STARTDATE = 2014, 8, 10, 0, 0, 0,	積分を開始する日付:放射過程計算で必要
TIME_STARTMS = 0.D0,	開始時刻 [mili sec]
TIME_DURATION = 12.0D0,	積分時間 [単位はTIME_DURATION_UNIT で設定]
TIME_DURATION_UNIT = "HOUR",	TIME_DURATION の単位
TIME_DT = 60.0D0,	時間積分の時間刻み幅
TIME_DT_UNIT = "SEC",	TIME_DT の単位
TIME_DT_ATMOS_DYN = 30.0D0,	力学過程計算の時間刻み幅
TIME_DT_ATMOS_DYN_UNIT = "SEC",	TIME_DT_ATMOS_DYN の単位
TIME_DT_ATMOS_PHY_CP = 600.0D0,	積雲パラメタリゼーション計算の時間刻み幅
TIME_DT_ATMOS_PHY_CP_UNIT = "SEC",	TIME_DT_ATMOS_PHY_CP の単位
TIME_DT_ATMOS_PHY_MP = 60.0D0,	雲物理過程計算の時間刻み幅
TIME_DT_ATMOS_PHY_MP_UNIT = "SEC",	TIME_DT_ATMOS_PHY_MP の単位
TIME_DT_ATMOS_PHY_RD = 600.0D0,	放射過程計算の時間刻み幅
TIME_DT_ATMOS_PHY_RD_UNIT = "SEC",	TIME_DT_ATMOS_PHY_RD の単位
TIME_DT_ATMOS_PHY_SF = 60.0D0,	大気下端境界 (フラックス) 過程計算の時間刻み幅
TIME_DT_ATMOS_PHY_SF_UNIT = "SEC",	TIME_DT_ATMOS_PHY_SF の単位
TIME_DT_ATMOS_PHY_TB = 60.0D0,	乱流過程計算の時間刻み幅
TIME_DT_ATMOS_PHY_TB_UNIT = "SEC",	TIME_DT_ATMOS_PHY_TB の単位
TIME_DT_ATMOS_PHY_BL = 60.0D0,	混合層過程計算の時間刻み幅
TIME_DT_ATMOS_PHY_BL_UNIT = "SEC",	TIME_DT_ATMOS_PHY_BL の単位
TIME_DT_OCEAN = 300.0D0,	海面過程計算の時間刻み幅
TIME_DT_OCEAN_UNIT = "SEC",	TIME_DT_OCEAN の単位
TIME_DT_LAND = 300.0D0,	陸面過程計算の時間刻み幅
TIME_DT_LAND_UNIT = "SEC",	TIME_DT_LAND の単位
TIME_DT_URBAN = 300.0D0,	都市過程計算の時間刻み幅
TIME_DT_URBAN_UNIT = "SEC",	TIME_DT_URBAN の単位
TIME_DT_ATMOS_RESTART = 21600.D0,	リスタートファイル (大気) の出力間隔
TIME_DT_ATMOS_RESTART_UNIT = "SEC",	TIME_DT_ATMOS_RESTART の単位
TIME_DT_OCEAN_RESTART = 21600.D0,	リスタートファイル (海面) の出力間隔
TIME_DT_OCEAN_RESTART_UNIT = "SEC",	TIME_DT_OCEAN_RESTART の単位
TIME_DT_LAND_RESTART = 21600.D0,	リスタートファイル (陸面) の出力間隔
TIME_DT_LAND_RESTART_UNIT = "SEC",	TIME_DT_LAND_RESTART の単位
TIME_DT_URBAN_RESTART = 21600.D0,	リスタートファイル (都市) の出力間隔
TIME_DT_URBAN_RESTART_UNIT = "SEC",	TIME_DT_URBAN_RESTART の単位
TIME_DT_WALLCLOCK_CHECK = 21600.D0,	実経過時間を確認する時間間隔
TIME_DT_WALLCLOCK_CHECK_UNIT = "SEC",	TIME_DT_WALLCLOCK_CHECK の単位
TIME_WALLCLOCK_LIMIT = 86400.D0,	経過時間の制限 [sec]
TIME_WALLCLOCK_SAFE = 0.95D0,	経過時間制限に対する安全率
/	

4.2.7.1 力学過程に対する時間刻み幅

(**TIME_DT**) は時間積分に対する時間刻み幅であり、 Δt と大抵書かれる。 Δt はトレーサ移流に対する時間刻み幅であり、また全ての物理過程の基本単位でもある。計算不安定を回避するために、

(**TIME_DT**) は、格子間隔を移流速度の最大値で割った値よりも小さくしなければならない。力学変数の時間積分は移流速度ではなく音速で制約されるため、力学過程の時間刻み幅 (**TIME_DT_ATMOS_DYN**) は Δt よりも小さく与えるべきである。(**TIME_DT_ATMOS_DYN**) の値は、計算安定性に関連して時間積分スキームに依存する。(**TIME_DT_ATMOS_DYN**) の標準的な値として、(**ATMOS_DYN_TINTEG_SHORT_TYPE**) が RK4 の場合は最小格子間隔を 420 m/s で割った値、RK3 の場合には最小格子間隔を 840 m/s で割った値が目安である。ただし、(**TIME_DT_ATMOS_DYN**) は、(**TIME_DT**) の約数でなければならないことに注意されたい。また、(**TIME_DT**) の (**TIME_DT_ATMOS_DYN**) に対する比が大きすぎる場合は計算不安定がしばしば起きる。(**TIME_DT**)/(**TIME_DT_ATMOS_DYN**) が、2 または 3 となるように設定することを推奨する。これらの条件については、第 4.2.7.2 節も参照されたい。(**TIME_DT_ATMOS_DYN**) と (**TIME_DT_ATMOS_DYN_UNIT**) を設定する代わりに、この比 ((**TIME_DT**)/(**TIME_DT_ATMOS_DYN**)) を (**TIME_NSTEP_ATMOS_DYN**) で指定することができる。(**TIME_NSTEP_ATMOS_DYN**) は整数でなければならない。

4.2.7.2 CFL 条件の確認

移流に対する時間刻み幅 (**TIME_DT**) は、格子幅を速度で割った値よりも小さくしなければならない (すなわち、Courant-Friedrichs-Lewy (CFL) 条件)。無次元数 $U\Delta t/\Delta x$ はクーラン数と呼ばれる。ここで、 U は速度、 Δx は格子幅、 Δt は時間刻み幅である。CFL 条件とは、クーラン数が 1 よりも小さくなければならないことである。

SCALE-RM には、クーラン数が制限値を超えているかを確認する機能がある。この機能を有効にするには、[**PARAM_ATMOS_VARS**] の (**ATMOS_VARS_CHECKCFL_SOFT**) や (**ATMOS_VARS_CHECKCFL_HARD**) を設定する。これらのデフォルト値はそれぞれ、1.0 と 2.0 である。シミュレーション中にクーラン数が (**ATMOS_VARS_CHECKCFL_SOFT**) を超えれば、以下のメッセージをログファイルに出力される。

```
INFO [ATMOS_vars_monitor] Courant number = xxx exceeded the soft limit = yyy
```

もし (**ATMOS_VARS_CHECKCFL_HARD**) を超えれば、以下のメッセージが標準出力に出力され、シミュレーションは強制終了される。

```
ERROR [ATMOS_vars_monitor] Courant number = xxx exceeded the hard limit = yyy
```

4.2.7.3 物理過程に対する時間刻み幅

物理過程に対する時間刻み幅は、各物理過程が与える時間変化率を更新するタイミングを表す。モデルが開始するとすぐに、初期の時間変化率を得るためにモデルの初期化時に各物理過程が呼ばれる。その後、各物理過程ごとに指定した時間間隔で各時間変化率が更新される。物理過程に対する時間間隔は全て、(**TIME_DT**) の倍数でなければならない。

表面フラックスは大気に対する表面過程で計算される。対照的に、あるモデル格子がいくつかの種類の利用区分 (海面・都市・陸面) を含む場合は海面・陸面・都市モデルが用いられ、これらのモデルによって表面フラックスが計算される。フラックスの格子平均値は、各利用区分に対するフラックスの利用区分の割合に応じた重み付き平均値として得られる。

上述したように、各過程の初期の時間変化率はモデルの初期化中に更新される。したがって、リスタートファイルの出力間隔は、全過程の時間刻み幅の倍数であることが要求される。そうしなければ、リスタート計算は、通しで時間積分を行った計算と一致しない。`(TIME_DT_ATMOS_RESTART)`、`(TIME_DT_OCEAN_RESTART)`、`(TIME_DT_LAND_RESTART)`、`(TIME_DT_URBAN_RESTART)` を指定していない場合は、リスタートファイルはシミュレーションの最後 (すなわち `(TIME_DURATION)`) に生成される。リスタート計算の詳細は第 4.2.9 節を参照されたい。

4.2.7.4 経過時間タイマーによるモデルの終了

幾つかのバッチジョブシステムでは、実行時間の制限が大抵設けられている。しかし、長時間積分のシミュレーションの所要時間を推定することは難しく、しばしばジョブが時間制限を超えることがある。この問題を解決するために、SCALE-RM ではセルフタイマーを用いた終了オプションを提供している。

経過時間が `(TIME_WALLCLOCK_LIMIT)` (秒) で指定した時間に達したときに、積分時間を終えていない場合でもリスタートファイルを出力し、時間ループを終了させる。`(TIME_WALLCLOCK_LIMIT)` に対する安全率が存在する。このデフォルトの値は 0.9 であり、`(TIME_WALLCLOCK_SAFE)` で指定する。

上述したように、リスタート出力の間隔は全ての物理過程や表面サブモデルに対する時間刻み幅の倍数とするべきである。しかしながら、セルフタイマーは唐突にシミュレーションを停止する。予期されるタイミングと異なるタイミングでリスタート出力が行われることを避けるために、経過時間を確認するタイミングを指定することができる。経過時間は、`(TIME_DT_WALLCLOCK_CHECK)` と `(TIME_DT_WALLCLOCK_CHECK_UNIT)` で指定した時間間隔で確認される。これらのパラメータを指定しない場合は、物理過程と表面サブモデルの最大時間間隔が設定される。確認の時間間隔を非常に長く設定した場合は、終了のタイミングが遅れる可能性があることに注意が必要である。

上記の例では、`(TIME_WALLCLOCK_LIMIT)` を 24 時間、`(TIME_WALLCLOCK_SAFE)` を 0.95 に設定している。経過時間は、シミュレーション時間の 6 時間ごとに確認される。経過時間が 22.8 時間を超過するとリスタートファイルが生成されて、シミュレーションは停止するだろう。

4.2.8 ヒストリファイルと出力変数の設定

ヒストリファイルと出力変数は、`run.conf` 内の `[PARAM_FILE_HISTORY_CARTESC]`、`[PARAM_FILE_HISTORY]`、`[HISTORY_ITEM]` で設定する。ヒストリファイルのデフォルトの形式は、`[PARAM_FILE_HISTORY]` で指定する。

```
&PARAM_FILE_HISTORY_CARTESC
FILE_HISTORY_CARTESC_PRES_nlayer = -1,      ; 圧力レベル数
                                              (圧力レベルへの補間に関するオプション)
FILE_HISTORY_CARTESC_PRES = 0.DO             ; 補間を行う圧力レベル (下層から上層の順) [hPa]
                                              (圧力レベルへの補間に関するオプション)
FILE_HISTORY_CARTESC_BOUNDARY = .false.,     ; 八口のデータを出力するか?
                                              .true.: 出力する, .false.: 出力しない.
/
```

```

&PARAM_FILE_HISTORY
FILE_HISTORY_TITLE = "",
FILE_HISTORY_SOURCE = "",

FILE_HISTORY_INSTITUTION = "",
FILE_HISTORY_TIME_UNITS = "seconds",
FILE_HISTORY_DEFAULT_BASENAME = "history_d01",
FILE_HISTORY_DEFAULT_POSTFIX_TIMELABEL = .false.,
FILE_HISTORY_DEFAULT_ZCOORD = "model",
FILE_HISTORY_DEFAULT_TINTERVAL = 3600.DO,
FILE_HISTORY_DEFAULT_TUNIT = "SEC",
FILE_HISTORY_DEFAULT_TAVERAGE = .false.,
FILE_HISTORY_DEFAULT_DATATYPE = "REAL4",
FILE_HISTORY_OUTPUT_STEPO = .true.,
FILE_HISTORY_OUTPUT_WAIT = 0.DO,
FILE_HISTORY_OUTPUT_WAIT_TUNIT = "SEC",
FILE_HISTORY_OUTPUT_SWITCH_TINTERVAL = -1.DO,
FILE_HISTORY_OUTPUT_SWITCH_TUNIT = "SEC",
FILE_HISTORY_ERROR_PUTMISS = .true.,

FILE_HISTORYAggregate = .false.,
/

```

; データに関する簡単な説明 (5.1.1 節参照)
 ; データを作成したソフトウェアの名前 (5.1.1 節参照)
 ; データの作成者 (5.1.1 節参照)
 ; netCDF 中の時間軸の単位
 ; 出力ファイルのベース名
 ; ファイル名に時間のラベルを加えるか?
 ; 鉛直座標の種類
 ; ヒストリ出力の時間間隔
 ; DEFAULT_TINTERVAL の単位
 ; 出力の時間間隔中の値を平均するか?
 ; 出力データの種類: REAL4 or REAL8
 ; 初期時刻 (t=0) のデータを出力するか?
 ; 出力を抑制する時間
 ; OUTPUT_WAIT の単位
 ; ファイルを切り替える時間間隔
 ; OUTPUT_SWITCH_TINTERVAL の単位
 ; データの準備状況の整合性を確認するか?
 ; PnetCDF を用いて単一のファイルにまとめるか?

デフォルトでは、各プロセスがヒストリファイルを出力する。`(FILE_HISTORYAggregate)` を `.true.` に設定した場合は、parallel NetCDF を用いることによって分散した出力ファイルが単一のファイルへとまとめられる。`(FILE_HISTORYAggregate)` のデフォルト設定は、`[PARAM_FILE]` 内の `(FILEAggregate)` によって決定される (第 5.1.1 節を参照)。

`(FILE_HISTORYDefault_TINTERVAL)` はヒストリ出力の時間間隔であり、その単位は `(FILE_HISTORYDefault_TUNIT)` によって定義される。単位は、`"MSEC"`, `"msec"`, `"SEC"`, `"sec"`, `"s"`, `"MIN"`, `"min"`, `"HOUR"`, `"hour"`, `"h"`, `"DAY"`, `"day"` から選択できる。`(FILE_HISTORYDefault_TAVERAGE)` を `.true.` とし平均値の出力を設定した場合は、`(FILE_HISTORYDefault_TINTERVAL)` に指定した直近の期間に渡って平均されたヒストリデータを出力する。

ヒストリ出力の時間間隔は、それと関係したスキームの時間間隔と等しいか倍数でなければならない。この整合性の確認を無効にしたい場合には、`(FILE_HISTORYError_PUTMISS)` を `.false.` に設定すれば良い。

`(FILE_HISTORYDefault_POSTFIX_TIMELABEL)` を `.true.` に設定した場合は、時間に関するラベルが出力ファイル名に付加される。時間のラベルはシミュレーションの現時刻に基づいて生成され、その形式は `YYYYMMDD-HHMMSS.msec` によって定義される。

`(FILE_HISTORYOutput_STEPO)` を `.true.` に設定した場合は、時間積分前の時刻における変数 (初期値) をヒストリファイルに出力する。`(FILE_HISTORYOutput_WAIT)` と `(FILE_HISTORYOutput_WAIT_TUNIT)` で定義したシミュレーションの期間、ヒストリ出力を抑制することができる。値が負であれば、出力の抑制は起こらない。`(FILE_HISTORYOutput_SWITCH_TINTERVAL)` は出力ファイルの切り替えの時間間隔であり、その単位は `(FILE_HISTORYOutput_SWITCH_TUNIT)` で定義する。値が負であれば、ヒストリ出力のために各プロセス毎に単一ファイルだけが使われる。このオプションを有効にした場合は、時間に関するラベルがファイル名に付加される。

大気の 3 次元変数を出力するために、3 種類の鉛直座標が利用できる。デフォルトでは

(`FILE_HISTORY_DEFAULT_ZCOORD`) = "model"が選択される。この場合は、変数はモデルのもとの座標系 (SCALE-RM において地形に沿った、 z^* 座標系) を用いて出力される。(`FILE_HISTORY_DEFAULT_ZCOORD`) を "z" に設定した場合は、変数は絶対高度へと補間される。出力データのレベル数は、モデルのレベル数と同じである。各レベルにおける高度は、地形を伴わない格子セルにおけるモデル高度と同じである。(`FILE_HISTORY_DEFAULT_ZCOORD`) を "pressure" に設定した場合は、変数は圧力レベルへと補間される。この場合は、`[PARAM_FILE_HISTORY_CARTESC]` 内の (`FILE_HISTORY_CARTESC_PRES_nlayer`) と (`FILE_HISTORY_CARTESC_PRES`) を設定する必要がある。

`[PARAM_FILE_HISTORY_CARTESC]` 内の (`FILE_HISTORY_CARTESC_BOUNDARY`) を `.true.` にした場合は、周期境界条件の場合を除いて、対象領域の外側に位置する八口のデータも出力される。

(`FILE_HISTORY_CARTESC_BOUNDARY`) の設定は、全ての出力変数に適用される。

出力変数は、`[HISTORY_ITEM]` を加えることで設定される。出力の形式は、`[PARAM_FILE_HISTORY]` で指定されたデフォルト設定に従う。下記のように、「(オプション)」と書かれたネームリストの項目を追加することで、特定の変数に対する形式をデフォルト設定から変更できる。

<code>&HISTORY_ITEM</code>	
<code>NAME = "RAIN",</code>	変数名. 変数のリストはリファレンスマニュアル内にある (5.3.2 節を参照)
<code>OUTNAME = "",</code>	(オプション) NAME と同じ
<code>BASENAME = "rain_d01",</code>	(オプション) <code>FILE_HISTORY_DEFAULT_BASENAME</code> と同じ
<code>POSTFIX_TIMELABEL = .false.,</code>	(オプション) <code>FILE_HISTORY_DEFAULT_POSTFIX_TIMELABEL</code> と同じ
<code>ZCOORD = "model",</code>	(オプション) <code>FILE_HISTORY_DEFAULT_ZCOORD</code> と同じ
<code>TINTERVAL = 600.DO,</code>	(オプション) <code>FILE_HISTORY_DEFAULT_TINTERVAL</code> と同じ
<code>TUNIT = "SEC",</code>	(オプション) <code>FILE_HISTORY_DEFAULT_TINTERVAL</code> と同じ
<code>TAVERAGE = .true.,</code>	(オプション) <code>FILE_HISTORY_DEFAULT_TAVERAGE</code> と同じ
<code>DATATYPE = "REAL4",</code>	(オプション) <code>FILE_HISTORY_DEFAULT_DATATYPE</code> と同じ
<code>/</code>	

`[HISTORY_ITEM]` で要求した変数がシミュレーションの時間ステップ中に準備されていない場合は、実行が停止し、エラーログがログファイルに書かれる。この状況は、(`NAME`) にスペルミスがある場合や、要求した変数が選択したスキーム内で使用されていない場合に発生し得る。

「(オプション)」と書かれたネームリストの項目は、変数 (`NAME`) に対してのみ適用される。変数に対してデフォルト設定を用いる場合は、「(オプション)」の項目は省略できる。例えば、`[PARAM_FILE_HISTORY]` の上記の設定を維持しつつ、`[HISTORY_ITEM]` に対して以下の設定を付け加えよう。ファイル `history_d01.xxxxxx.nc` に、 U と V の瞬間値を 3600 秒間隔で 4 バイトの実数値として格納する。一方で、RAIN については 600 秒間隔でその期間に渡った平均値をファイルに格納する。 T の値は、 U や V と同じ規則で T として出力し、圧力座標系に補間した値を T_{pres} として出力する。

```
&HISTORY_ITEM NAME="T" /
&HISTORY_ITEM NAME="U" /
&HISTORY_ITEM NAME="V" /
&HISTORY_ITEM NAME="RAIN", TINTERVAL=600.DO, TAVERAGE=.true. /
&HISTORY_ITEM NAME="T", OUTNAME="T_pres", ZCOORD="pressure" /
```

4.2.9 リスタート計算の方法

リスタート機能は、計算システムで決められたジョブ実行の時間制限のために、シミュレーションが途切れてしまう場合などに役立つ。リスタート機能を用いることで、長期間の一続きのシミュレーションを複数のランに分割できる。リスタートファイルは、初期のランで生成されたデータと同じ形式を持つ。各シミュレーションの最後にリスタートファイルを出力する以外にも、特定の時間間隔で複数のリスタートファイルを出力する機能もある。リスタートファイルに対する設定は、シミュレーション実行用の設定ファイル中の `[PARAM_RESTART]` と `[PARAM_TIME]` で行う。以下の例では、ファイル `restart1_***` によってシミュレーションをリスタートし、6 時間ごとにリスタートファイル `restart2_***` を生成する。

```
&PARAM_RESTART
RESTART_IN_BASENAME = "restart1_d01_20070715-000000.000",
RESTART_IN_POSTFIX_TIMELABEL = .false.,
RESTART_OUTPUT = .true.,
RESTART_OUT_BASENAME = "restart2_d01",
RESTART_OUT_POSTFIX_TIMELABEL = .true.,
RESTART_OUT_TITLE = "",
RESTART_OUT_DTYPE = "DEFAULT",
/

&PARAM_TIME
TIME_STARTDATE = 2007, 7, 15, 00, 0, 0,
TIME_STARTMS = 0.DO,
TIME_DURATION = 12.0DO,
TIME_DURATION_UNIT = "HOUR",
..... 略 .....
TIME_DT_ATMOS_RESTART = 21600.DO,
TIME_DT_ATMOS_RESTART_UNIT = "SEC",
TIME_DT_OCEAN_RESTART = 21600.DO,
TIME_DT_OCEAN_RESTART_UNIT = "SEC",
TIME_DT_LAND_RESTART = 21600.DO,
TIME_DT_LAND_RESTART_UNIT = "SEC",
TIME_DT_URBAN_RESTART = 21600.DO,
TIME_DT_URBAN_RESTART_UNIT = "SEC",
/
```

入力する初期値ファイルまたはリスタートファイルのベース名。

RESTART_IN_BASENAME の後に入力時の年月日時刻を追加するか?

リスタートファイルを出力するか?
.true.: 出力する、.false.: 出力しない。

リスタートファイルのファイルのベース名。

RESTART_OUT_BASENAME の後に出力時の年月日時刻が追加を追加するか?

リスタートファイルに書かれる題目
REAL4 or REAL8 or DEFAULT

リスタート計算の開始時刻
計算開始時刻 [mili sec]
積分時間 [単位はTIME_DURATION_UNIT で設定]
TIME_DURATION の単位

リスタートファイル (大気) の出力間隔
TIME_DT_ATMOS_RESTART の単位

リスタートファイル (海洋) の出力間隔
TIME_DT_OCEAN_RESTART の単位

リスタートファイル (陸面) の出力間隔
TIME_DT_LAND_RESTART の単位

リスタートファイル (都市) の出力間隔
TIME_DT_URBAN_RESTART の単位

リスタートファイルの出力間隔は、`(TIME_DT_ATMOS_RESTART)`、`(TIME_DT_OCEAN_RESTART)`、`(TIME_DT_LAND_RESTART)`、`(TIME_DT_URBAN_RESTART)` で指定する。これらが指定されていない場合は、積分時刻の最終時刻 `(TIME_DURATION)` にリスタートファイルが作成される。出力されるリスタートファイルの名前は、`(RESTART_IN_BASENAME)` で指定する。`(RESTART_OUT_POSTFIX_TIMELABEL)` は、`(RESTART_OUT_BASENAME)` の後のファイル名に出力時の日時を自動的に追加するかを指定する。デフォルト設定は、`(RESTART_OUT_POSTFIX_TIMELABEL=.true.)` である。

リスタートファイルは、全てのシミュレーションに対する互換性はない。リスタートファイルに含まれる変数は、設定ファイルで選択したスキームによって異なる。整合性を担保したリスタートファ

イルを用意するための簡単な方法は、一連のシミュレーションにおいてスキームに対して同じ設定を使用することである。

他の設定は、通常のランと基本的に同じである。(RESTART_IN_BASENAME) は、大気や表面サブモデルの初期状態を含む入力ファイルの名前である。通常のランでは scale-rm_init で準備した init_*** を用いるが、リスタートランでは前のランで出力されたリスタートファイルを用いる。

(RESTART_IN_POSTFIX_TIMELABEL) は (RESTART_OUT_POSTFIX_TIMELABEL) と同様であるが、(RESTART_IN_BASENAME) に対する日時の付加を指定する。デフォルト設定では、(RESTART_IN_POSTFIX_TIMELABEL = .false.) である。

上記の例において、(RESTART_IN_BASENAME) = "restart1_d01_20070715-000000.000" と設定することは、(RESTART_IN_POSTFIX_TIMELABEL = .true.) として (RESTART_IN_BASENAME) = "restart1_d01" と設定することと等価である。リスタート計算の開始日時や積分時間はそれぞれ、(TIME_STARTDATE) と (TIME_DURATION) で指定する。

現実大気実験の場合は、初期値データに加えて scale-rm_init で作成した境界値データが必要である。以下に例を示す。

<pre>&PARAM_ATMOS_BOUNDARY ATMOS_BOUNDARY_TYPE = "REAL", ATMOS_BOUNDARY_IN_BASENAME = "../init/output/boundary_d01", ATMOS_BOUNDARY_START_DATE = 2010, 7, 14, 18, 0, 0, ATMOS_BOUNDARY_UPDATE_DT = 21600.D0, /</pre>	<p>現実実験の場合は "REAL"。 境界値データのファイル名の頭。 境界値データの初期時刻。 境界値データの時間間隔。</p>
--	---

リスタート計算において、境界値データの適切な日時は、境界値ファイル boundary_***.nc から読み込まれる。境界値データの最初の日時は、[PARAM_ATMOS_BOUNDARY] の (ATMOS_BOUNDARY_START_DATE) で指定しなければならない。(ATMOS_BOUNDARY_START_DATE) を指定しなかった場合は、実際の日時とは異なっても、境界値ファイルの最初のデータがリスタート計算の最初の日時に設定される。

4.2.10 領域ネスティング実験の方法

ネスティングとは、領域が重複するように複数の計算領域を入れ子（ネスト）構造に設定する方法である。図 4.2.5 は、3 つの領域を用いたネスティングの例を示している。外側の領域は、大きな空間スケールの現象を表現するため、低い水平解像度で広い領域を設定する。一方、内側の領域は、小さな空間スケールの現象を解像するために、狭い範囲であるが高い水平解像度で設定する。外側の領域の計算結果は、内側の領域に対する境界値データとして用いられる。ここでは、データを渡す外側の領域を「親領域」、データを受ける内側の領域を「子領域」と呼ぶことにする。

ネスティングの方法は下記のように分類される。

- 実行方法

- オンライン・ネスティング

- 計算途中で親領域と子領域の情報を交換しながら、親領域と子領域の計算を同時に実行する方法。

- オフライン・ネスティング

- 最初に親領域の計算を行って子領域用の初期値・境界値を作成し、その後に子領域の計算を行う方法。

- データの受け渡し方法

- 一方向ネスティング

- 親領域は子領域にデータを送るが、子領域は親領域にデータを送らない。親領域の結果は、子領域の結果の影響を受けない。

- 双方向ネスティング

- 親領域は子領域にデータを送り、子領域からのデータも受け取る。したがて、二つの領域の計算は互いに影響し合う。この方法はオンライン・ネスティング時に適用できるが、SCALE-RM v5.3.3 ではまだ実装されていない。

オンラインとオフラインの違いは、親領域から子領域にデータを与える更新頻度にある。オンライン・ネスティング実験では、子領域の境界条件は親領域の時間刻み幅 (Δt) 毎に更新される。オフライン・ネスティング実験では、更新頻度は親領域の計算におけるヒストリファイルの出力間隔に依存する。

ネスティングがオフラインかオンラインかに関わらず、親領域と子領域の格子間隔比 (DX_{d01}/DX_{d02}) に関してコードの実装上の制限はない。ただし、この比率が大きすぎると計算結果の物理性能が下がる可能性がある。SCALE-RM では、5 倍以下で使用することを推奨する。

本節では、親領域の設定ファイルを***.d01.conf、子領域の設定ファイルを***.d02.conf と表記する。

4.2.10.1 子領域における地形の取り扱い

ネスティング実験では、一般的に、親領域と子領域の間で空間解像度が異なるために地形の解像度も異なる。子領域の緩和領域 (第 4.2.3.5 節を参照) では、大気の変数は親領域の変数へとナッジングされる。2 つの領域間で地形の表現が異なると、親領域で計算されるナッジングのための参照データが存在しないことがある。その場合は外挿により大気データを見積もることになるが、外挿による見積もりの精度が悪いと不整合が生じる。地形の違いによる不整合を回避するために、SCALE-RM

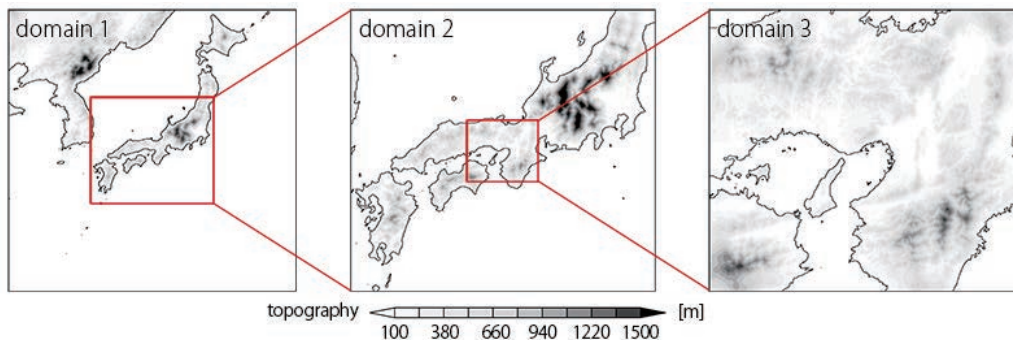


図 4.2.5: 西日本を対象とした領域ネスティングの例。domain 1 が最外領域で domain 3 が最内領域である。赤い矩形と線は、領域の位置や他の領域との関係を示している。水平格子間隔は domain 1 では 7.5 km、domain 2 では 2.5 km、domain 3 では 0.5 km である。

では「地形コピー」機能を使用することを推奨している。この機能は、子領域の緩和領域における地形として親領域の地形をコピーする。この機能を使えば、図 4.2.6 に示すように、子領域の緩和領域の地形と親領域の地形を完全に一致させられる。さらに、地形の解像度を外側から内側に行くに従って徐々に高めるために、緩和領域の内側に地形の遷移領域を置く。地形の遷移領域では、地形は親領域と子領域の地形を重み付けすることで生成される。地形遷移領域の幅は、デフォルト設定では緩和領域と同じ幅である。これよりも内側の計算領域では、地形は子領域の地形を与える。「実験用セッソー式作成ツール」(第 4.2.1 節)を利用する場合は、地形コピー機能が自動的に適用される。

本節で示す pp.d0*.conf ファイルは、サンプル設定ファイル

`${Tutorial_dir}/real/sample/USER.online-nesting.sh` を `USER.sh` に名前を変更して、「実験用セッソー式作成ツール」を実行することで作成される。説明を読み進める上で参考にしてもらいたい。以降は、具体的な設定方法と実行手順を説明する。

地形コピー機能の使い方

親領域の地形データを `scale-rm_pp` で作成する際に、親領域の大きさを子領域に与えるためのカタログファイルを出力させるには、下記の設定が `pp.d01.conf` に必要である。

```
&PARAM_DOMAIN_CATALOGUE
  DOMAIN_CATALOGUE_FNAME = "latlon_domain_catalogue.d01.txt",  カタログファイルのファイル名
  DOMAIN_CATALOGUE_OUTPUT = .true.,                            カタログファイルを出力するか?
/
```

その他の設定項目は通常通りで良い。

次に、地形コピー機能で親の地形を用いるために、子領域に対する `pp.d02.conf` ファイルを以下のように編集する。ここでは、親領域の地形の出力データが `topo_d01.pe***.nc` として保存されていると想定する。

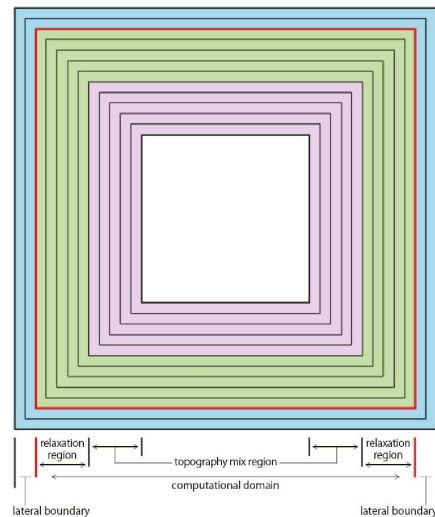


図 4.2.6: 地形コピー機能を適用したときの地形の水平分布。水色で塗られた最外にある格子は HALO 領域であり、その格子数は水平移流スキームに依存する。これらの格子は側面境界である。赤色の線で囲われた部分は、計算領域である。緑色や桃色の領域はそれぞれ、緩和領域と地形遷移領域である。最内の白色の領域では、地形は子領域のもとの地形と同じである。地形遷移領域では、外側から内側にかけて徐々に親領域の地形データから子領域の地形データへ遷移する。

<code>&PARAM_NEST</code>	
<code>OFFLINE_PARENT_BASENAME = "topo_d01",</code>	親領域のファイルのベース名
<code>OFFLINE_PARENT_PRC_NUM_X = 2,</code>	親領域の PRC_NUM_X
<code>OFFLINE_PARENT_PRC_NUM_Y = 2,</code>	親領域の PRC_NUM_Y
<code>LATLON_CATALOGUE_FNAME = "latlon_domain_catalogue.d01.txt",</code>	親領域のカatalogueファイル
/	
<code>&PARAM_CNVTPOPO</code>	
~ 中略 ~	
<code>CNVTPOPO_copy_parent = .true.,</code>	地形コピー機能を適用するかどうか
/	
<code>&PARAM_COPYTOPO</code>	
<code>COPYTOPO_IN_BASENAME = "topo_d01",</code>	親領域の地形データファイルのベース名
<code>COPYTOPO_TRANSITION_DX = -1,</code>	x 方向の遷移域の幅
<code>COPYTOPO_TRANSITION_DY = -1,</code>	y 方向の遷移域の幅
<code>COPYTOPO_ENTIRE_REGION = .false.,</code>	子領域の全域に親領域の地形をコピーするかどうか
<code>COPYTOPO_LINEAR_H = .true.,</code>	
/	

[PARAM_CNVTPOPO] の (CNVTPOPO_copy_parent) を .true. とすれば、地形のコピー機能が適用される。(COPYTOPO_ENTIRE_REGION) は、子領域の全域に渡って親領域の地形をコピーするかを決めるオプションである。これが .true. の場合は、子領域の地形は親領域から完全にコピーされる。(COPYTOPO_LINEAR_H) は地形の遷移方法を指定するパラメータである。(COPYTOPO_LINEAR_H) が .true. であれば子領域と親領域の地形の混合割合が線形的に変化し、そうでなければ指数関数的に変化する。

る。遷移領域の幅は (COPYTOPO_TRANSITION_DX) や (COPYTOPO_TRANSITION_DY) で指定する。これらの値が負であればデフォルトの設定が適用され、地形の遷移領域の幅は緩和領域の幅と同じに取られる。

地形の作成

地形コピー機能を使用する場合は、子領域は親領域のカタログファイルを必要とするため、親領域から順番に地形を作成しなければならない。領域が3つ以上ある場合は、実行の順番は以下になる。

```
$ mpirun -n [プロセス数] ./scale-rm_pp pp.d01.conf
$ mpirun -n [プロセス数] ./scale-rm_pp pp.d02.conf
$ mpirun -n [プロセス数] ./scale-rm_pp pp.d03.conf
```

4.2.10.2 オフライン・ネスティング実験

以下の2点は、オフライン・ネスティング実験に対する制限事項である。

- 子領域は親領域に完全に含まれる。
- 子領域の積分期間は、親領域の積分期間と同じかそれより短い。

また、オフライン・ネスティング実験は次の順番でなされる。

1. 親領域の時間積分を行う。
2. 親領域のヒストリ出力あるいは初期値/リスタート出力を用いて、子領域の初期値/境界値を作成する。
3. 作成した初期値/境界値を用いて、子領域の時間積分を行う。

以下では、上記の流れに沿って説明を進める。

親領域の時間積分

子領域の境界値データとして用いる親領域のデータを準備するために、いくつか必要な設定がある。親領域の計算に対する設定ファイルは、「実験用セッター式作成ツール」(第4.2.1節を参照)で作成する。サンプルファイル\${Tutorial_dir}/real/sample/USER.offline-nesting-parent.shをUSER.shに名前を変更し、makeを実行する。

親領域の時間積分はシングルドメインの場合と同じ方法で実行するが、設定に関して次の5点に注意する必要がある。

- 子領域の計算に必要な変数全てが、親領域の計算によってヒストリ/リスタート出力として作成されている。
- ヒストリ/リスタート出力の間隔が十分に短い。
- 親領域の計算領域の情報を子領域に与える「カタログファイル」を出力する。
- ヒストリファイルを用いる場合は、親領域のヒストリデータはモデル面で出力させる。

- 子領域の計算開始時刻が親領域と同じ場合は、親領域における $t=0$ のヒストリ出力データが必要である。

「カタログファイル」を出力するためには、前処理・初期化・計算実行のいずれかの設定ファイルが、以下のように設定されていなければならない。

```
&PARAM_DOMAIN_CATALOGUE
DOMAIN_CATALOGUE_FNAME = "latlon_domain_catalogue_d01.txt",   カタログファイルのファイル名
DOMAIN_CATALOGUE_OUTPUT = .true.,                             カタログファイルを出力。
/
```

カタログファイルの出力が `.true.` であれば、(この例の場合には) `latlon_domain_catalogue_d01.txt` が出力される。「実験用セット一式作成ツール」を用いる場合は、これと同じ名前のファイルがディレクトリ `pp` の中に生成される。このファイルの中には、親領域の計算で各 MPI プロセスが担当する計算領域の四隅の緯度経度が記述されている。

ヒストリファイルを用いたければ、次のような設定が必要である。

```
&PARAM_FILE_HISTORY
FILE_HISTORY_DEFAULT_BASENAME = "history",
FILE_HISTORY_DEFAULT_TINTERVAL = 900.DO,   ヒストリデータの出力時間間隔。
FILE_HISTORY_DEFAULT_TUNIT = "SEC",        FILE_HISTORY_DEFAULT_TINTERVAL の単位。
FILE_HISTORY_DEFAULT_TAVERAGE = .false.,
FILE_HISTORY_DEFAULT_DATATYPE = "REAL4",
FILE_HISTORY_DEFAULT_ZCOORD = "model",     モデル面データを出力。
FILE_HISTORY_OUTPUT_STEP0 = .true.,        t=0 の値を出力に含める。
/
```

(`FILE_HISTORY_DEFAULT_TINTERVAL`) はヒストリデータの出力間隔であり、子領域の計算で用いる更新時間間隔を設定する。相対的に短い時間間隔でデータを出力する場合には、ディスクの空き容量にも注意が必要である。その他、`[PARAM_FILE_HISTORY]` の各項目の詳細は、第 4.2.8 節を参照されたい。

リスタートファイルを用いたければ、設定は次のようになる。

```
&PARAM_RESTART
RESTART_OUTPUT = .true.
RESTART_OUT_BASENAME = 'restart_d01',
/
&PARAM_TIME
TIME_DT_ATMOS_RESTART = 900.DO,   リスタートデータの出力時間間隔
TIME_DT_ATMOS_RESTART_UNIT = "SEC",
TIME_DT_OCEAN_RESTART = 900.DO,   リスタートデータの出力時間間隔
TIME_DT_OCEAN_RESTART_UNIT = "SEC",
TIME_DT_LAND_RESTART = 900.DO,    リスタートデータの出力時間間隔
TIME_DT_LAND_RESTART_UNIT = "SEC",
TIME_DT_URBAN_RESTART = 900.DO,   リスタートデータの出力時間間隔
TIME_DT_URBAN_RESTART_UNIT = "SEC",
/
```

これらのパラメータの詳細は、第 4.2.9 節を参照されたい。

計算実行用の設定ファイル中の `[FILE_HISTORY_ITEM]` には、子領域の初期値/境界値データの作成に必要な変数を全て記述しなければならない。オフライン・ネスティングに必要な変数は子領域の計算設定に依存し、標準的な現実大気の計算における変数は以下である。

```
T2, MSLP, DENS, MOMZ, MOMX, MOMY, RHOT, QV
LAND_SFC_TEMP, URBAN_SFC_TEMP, OCEAN_SFC_TEMP
OCEAN_SFC_ALB_IR_dir OCEAN_SFC_ALB_IR_dif,
OCEAN_SFC_ALB_NIR_dir OCEAN_SFC_ALB_NIR_dif,
OCEAN_SFC_ALB_VIS_dir OCEAN_SFC_ALB_VIS_dif,
LAND_SFC_ALB_IR_dir, LAND_SFC_ALB_IR_dif,
LAND_SFC_ALB_NIR_dir, LAND_SFC_ALB_NIR_dif,
LAND_SFC_ALB_VIS_dir, LAND_SFC_ALB_VIS_dif,
OCEAN_TEMP, OCEAN_SFC_ZOM, LAND_TEMP, LAND_WATER
```

(親モデルが用いる雲微物理モデルに応じて出力する)

```
QC, QR, QI, QS, QG
NC, NR, NI, NS, NG
```

設定が完了したら、`scale-rm` を実行して親領域の時間積分を行う。

一般的に、親計算の最下層よりも低い層にある変数は外挿によって計算される。外挿は非現実的な値を生じさせる可能性があり、特に親計算の最下層が子計算の最下層よりも遥かに高い場合に問題となるだろう。この問題を避けるために、ヒストリファイルを用いる場合は、これらの変数を計算するために平均海面圧力と 2-m 温度を用いる。一方で、リスタートファイルを用いる場合にはそういった量がファイルに含まれない。したがって、親の最下層よりも低い層の変数は、最下層の変数の単なるコピーである。

子領域に対する初期値/境界値データの作成

子領域の計算用の設定ファイルは、「実験用セット一式作成ツール」(第 4.2.1 節を参照)を用いることで作成できる。サンプルスクリプト `${Tutorial_dir}/real/sample/USER.offline-nesting-child.sh` を `USER.sh` に名前を変更し、`make` を実行する、

親領域の計算で得られたヒストリデータを用いて初期値/境界値データを作成する場合は、初期値生成のための設定ファイルを以下のように設定する。


```

&PARAM_NEST
OFFLINE_PARENT_BASENAME = "history_d01",  親領域のファイル名
OFFLINE_PARENT_PRC_NUM_X = 2,             run.d01.conf のPRC_NUM_X
OFFLINE_PARENT_PRC_NUM_Y = 2,             run.d01.conf のPRC_NUM_Y
LATLON_CATALOGUE_FNAME = \
    "latlon_domain_catalogue_d01.txt",     親領域を実行した時に作成したカタログファイル

&PARAM_MKINIT_REAL_ATMOS
NUMBER_OF_TSTEPS = 25,                    history ファイル内の時間ステップ数
NUMBER_OF_FILES = 1,
FILETYPE_ORG = "SCALE-RM",
BASENAME_ORG = "history_d01",              run.d01.conf のHISTORY_DEFAULT_BASENAME
BOUNDARY_UPDATE_DT = 900.DO,              history ファイルの出力時間間隔 (単位は"SEC")
/

&PARAM_MKINIT_REAL_OCEAN
NUMBER_OF_TSTEPS = 25,                    history ファイル内の時間ステップ数
BASENAME_ORG = "history_d01",              run.d01.conf のHISTORY_DEFAULT_BASENAME
NUMBER_OF_FILES = 1,
FILETYPE_ORG = "SCALE-RM",
BOUNDARY_UPDATE_DT = 900.DO,              history ファイルの出力時間間隔 (単位は"SEC")
/

&PARAM_MKINIT_REAL_LAND
NUMBER_OF_TSTEPS = 25,                    history ファイル内の時間ステップ数
NUMBER_OF_FILES = 1,
BASENAME_ORG = "history_d01",              run.d01.conf のHISTORY_DEFAULT_BASENAME
FILETYPE_ORG = "SCALE-RM",
BOUNDARY_UPDATE_DT = 900.DO,              history ファイルの出力時間間隔 (単位は"SEC")
/

```

SCALE-RM 形式の出力データから初期値/境界値データを作成する場合は、(FILETYPE_ORG) に "SCALE-RM" を指定する。基本的に、(BOUNDARY_UPDATE_DT) は親領域の設定ファイル (run.d01.conf) の (FILE_HISTORY_DEFAULT_TINTERVAL) と同じ値を設定する。[PARAM_NEST] の項目は、ネスティング実験のための設定項目である。オフライン・ネスティングでは、(OFFLINE_PARENT_BASENAME) に親領域データのファイル名を指定する。また、(OFFLINE_PARENT_PRC_NUM_*) に親領域のプロセス数を設定する。これらは、親領域の設定ファイル (run.d01.conf) を参照して適切に設定されたい。

設定ファイルを編集し終わったら、scale-rm_init を実行し、子領域の初期値/境界値を作成する。実行時に下記のようなメッセージが表示されて計算が止まる場合は、子領域が親領域に完全には含まれていないことを意味する。

```

xxx ERROR: REQUESTED DOMAIN IS TOO MUCH BROAD
xxx -- LONGITUDINAL direction over the limit

```

子領域の時間積分

初期値/境界値データの作成が終わったら、scale-rm を実行して子領域の時間積分を行う。これは、通常の現実大気実験と同じである。1 点だけ注意すべき点として、run.d02.conf の [PARAM_ATMOS_BOUNDARY] の (ATMOS_BOUNDARY_UPDATE_DT) が親領域のヒストリデータの出力間隔と同じであることを確認すること。現在のバージョンでは、これらの時間間隔に不整合がある場合でも警告やエラーメッセージが

発せられないまま、計算が進むことに注意が必要である。

```
&PARAM_ATMOS_BOUNDARY
~ 中略 ~
ATMOS_BOUNDARY_UPDATE_DT = 900.DO,
/
```

多段のオフライン・ネスティング実験を行う場合は、以上の方法を繰り返せばよい。つまり、子領域における上記の時間積分の結果を親領域の結果とみなして、さらに内側にある孫領域の計算のための初期値/境界値を作成する。

4.2.10.3 オンライン・ネスティング実験

オンライン・ネスティング実験を実行する際には、以下の2つの制約が存在する。

- 子領域の積分時間は、親領域の積分時間と一致していなければならない。
- 親領域の時間ステップは、子領域の時間ステップの倍数でなければならない。

一方で、親領域と子領域で鉛直層数、鉛直レベル、地図投影法、物理スキームが一致している必要はない。オンライン・ネスティング実験では、全ての領域の計算は同時に実行される。現在のバージョンでは、SCALE-RM は一方向ネスティングのみ対応する。ネスティングの段数は最大で10段まで可能である。

SCALE-RM のオンライン・ネスティング実験では、複数領域の時間積分を逐次的でなく並列的に行う。図 4.2.7 に示すように、MPI プロセスはいくつかのグループに分割される。各々のグループは一つの領域を担当し、独立したモデルのように計算を進める。複数の領域を立ち上げるために、`launch.conf` という設定ファイルが実行時に必要である。

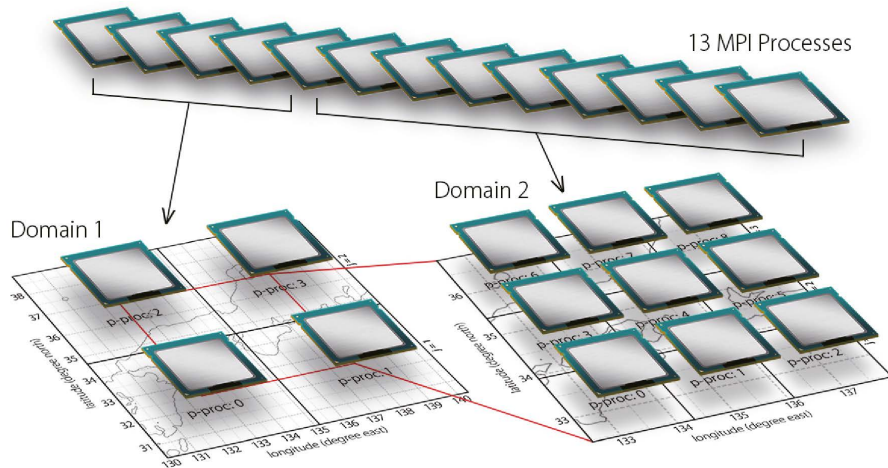


図 4.2.7: オンライン・ネスティング実験における MPI プロセスの配分。この例では 13 のプロセスが最初に立ち上げられ、これらを適切に分配する。つまり、Domain 1 では 2×2 の 4-MPI 並列計算、Domain 2 では 3×3 の 9-MPI 並列計算が行われる。MPI 通信によって Domain 1 から Domain 2 へデータを受け渡ししながら時間積分が進められる。

以下では、最も単純なオンライン・ネスティングの例として2段ネスティングを説明する。ここで記述される実験セト一式は、サンプルファイル`{Tutorial_dir}/real/sample/USER.online-nesting.sh`を`USER.sh`に名前を変更して、「実験用セト一式作成ツール」を実行することで作成される(第4.2.1節を参照)。以下の説明では、各領域に対する地形/土地利用データと初期値/境界値データの作成を終えているとする。地形データの作成手順は、第4.2.10.1節に示した通りである。

オンライン・ネスティングの設定

親領域と子領域のそれぞれの設定ファイル(`run.***.conf`)において、オンライン・ネスティングのための設定を以下のように`[PARAM_NEST]`に追加する。

run.d01.conf の設定

```
&PARAM_NEST
  ONLINE_DOMAIN_NUM = 1,           領域の番号。外側から1番。
  ONLINE_IAM_PARENT = .true.,
  ONLINE_IAM_DAUGHTER = .false.,
  ONLINE_BOUNDARY_USE_QHYD = .true.,
  ONLINE_AGGRESSIVE_COMM = .true.,
/
```

run.d02.conf の設定

```
&PARAM_NEST
  ONLINE_DOMAIN_NUM = 2,           領域の番号。外側から1番。
  ONLINE_IAM_PARENT = .false.,
  ONLINE_IAM_DAUGHTER = .true.,
  ONLINE_BOUNDARY_USE_QHYD = .true.,
  ONLINE_AGGRESSIVE_COMM = .true.,
/
```

(`ONLINE_DOMAIN_NUM`) は、領域の ID 番号であり、外側領域から内側領域へ順番に番号を振っていく。上の例において、親領域と子領域の ID 番号はそれぞれ1番と2番である。

(`ONLINE_IAM_PARENT`) と (`ONLINE_IAM_DAUGHTER`) は、各領域がその親領域や子領域を持っているかを指定する項目である。 N 番目の領域において (`ONLINE_IAM_PARENT`) が `.true.` であれば、 N 番目の領域の計算データは $N + 1$ の領域番号をもった子領域に送られる。(`ONLINE_IAM_DAUGHTER`) が `.true.` であれば、 N 番目の境界データは $N - 1$ 番の領域番号をもった親から受け取る。最も外側の領域は親領域としてのみ働き、最も内側の領域は子領域としてのみ働く。一方で、中間的な領域は親領域と子領域の両方を担うので、(`ONLINE_IAM_PARENT`) と (`ONLINE_IAM_DAUGHTER`) は共に `.true.` である。表4.2.3は N 段ネスティング実験の設定を示している。

(`ONLINE_BOUNDARY_USE_QHYD`) は、境界条件として凝結物の値を使うかを指定する。外部入力データから側面境界条件を作成するときには、凝結物の値は通常使われない。しかし、ネスティング実験では、二領域間で物理スキームが違わないために、親領域で計算された凝結物を子領域の境界条件として与えることができる。計算領域の内部に影響を与えるような、流入境界付近での雲や降水の発生の遅れが、この改善策によって抑制されると考えられる。

表 4.2.3: N 段ネスティングの設定例

領域	ONLINE_DOMAIN_NUM	ONLINE_IAM_PARENT	ONLINE_IAM_CHILD
最外領域	1	.true.	.false.
中間領域	2 ~ (N-1)	.true.	.true.
最内領域	N	.false.	.true.

ランチャーの設定

オンライン・ネスティング実験には、run.***.conf の他に、起動用設定ファイル launch.conf が必要である。

```
&PARAM_LAUNCHER
NUM_DOMAIN = 2,
PRC_DOMAINS = 4, 16,
CONF_FILES = run.d01.conf, run.d02.conf,
/
```

領域の数
それぞれの領域で使用する MPI プロセス数 (領域の数だけ必要)
それぞれの領域の設定ファイル (領域の数だけ必要)

(PRC_DOMAINS) と (CONF_FILES) の記載順は対応している必要がある。上記の例の場合は、親領域は 4-MPI 並列、子領域は 16-MPI 並列で実行するように指定されている。launch.conf で指定する MPI プロセス数は、各々の領域の設定ファイル (run.***.conf) で指定された総 MPI プロセス数 (PRC_NUM_X×PRC_NUM_Y) と一致させなければならない。

実行時には、シングル領域計算の場合とは異なり、計算全体で使用する MPI プロセス数を指定する。例えば、上記の場合だと 20 プロセスを指定する。

```
$ mpirun -n [プロセス数] ./scale-rm launch.conf
```

複数領域の計算が同時に実行されるときに、混同を避けるために、異なるファイル名を入出力ファイルに対して使用しなければならない。例えば、「実験用セトー式作成ツール」によって用意した設定ファイルでは、ヒストリ出力のファイル名を history_d01.pe***.nc, history_d02.pe***.nc としている。

実行時に次のようなメッセージが出力されて、計算が異常終了することがある。これは、子領域の計算領域が親領域の計算領域よりも大きいことを意味するエラーメッセージである。このようなメッセージが出た場合は、地形・土地利用データ、および初期値/境界値の作成からやり直し、設定が適切であるか再度確認されたい。

```
ERROR [COMM_CARTESC_NEST_domain_relate] region of daughter domain is larger than that
of parent
```

MPI プロセスの分配に関するガイドライン

オンライン・ネスティング実験では、図 4.2.7 に示したように、複数の領域間で MPI プロセスを共有しない。つまり、それぞれの MPI プロセスは特定の領域の一部分を担当することになる。このため、ユーザはいくつの MPI プロセスを各領域に割り当てるかを決める必要がある。この割り当て配分が適切でない場合は、長い待ち時間が発生する。これを避けるためには、各プロセスにおける

時間積分の計算量がプロセス間で可能な限り揃うように、MPI プロセスを配置するのが合理的である^{*2)}。ここで、時間積分の計算量は、格子数とタイムステップ数の積として定義する。

ここでは、N 段ネスティングを考える。n 番目の領域の X 方向、Y 方向、Z 方向の格子数をそれぞれ IMAX_n、JMAX_n、KMAX_n と表す。また、n 番目の領域の時間ステップ (TIME_DT) を DT_n と表す。この時、一番外側の領域 (n=1) の時間積分のタイムステップ DT_1 を基準とし、この時間を積分するのに必要な n 番目の領域の計算ステップ数は、

$$TSTEP_n = DT_1 / DT_n$$

と表される。領域全体での計算量は、領域が持つ格子数を掛けて

$$OPR_n = IMAX_n \times JMAX_n \times KMAX_n \times TSTEP_n$$

と見積もられる。したがって、n 番目の領域に配分する MPI プロセス数の目安は、

$$MPI_total \times \frac{OPR_n}{\sum_{m=1}^N OPR_m} \quad (4.2.9)$$

と見積もられる。ここで、MPI_total は全 MPI プロセス数である。

X 方向 と Y 方向に分配するプロセス数 (PRC_NUM_X, PRC_NUM_Y) には任意性が残る。IMAX と JMAX の違いができるだけ小さくなるように、これらを設定することが推奨される。その理由は、このような設定によってハ口領域を減らせるためである。結果的に、計算機の演算性能を引き出しやすいと考えられる^{*3)}。

上記の説明では、格子点数と積分時間の時間刻み幅のみを考慮した。しかし、現実大気のネスティング実験のような実計算では、各物理過程の時間間隔の違いや、領域内や領域間の MPI 通信にかかる時間の違いも計算時間に影響を及ぼす。オンライン・ネスティングの設定では、通常は最内領域で最も計算負荷が高く、そこでの MPI 通信の待ち時間が最小となるようにプロセスを分配するのが効率的であることが多い。大規模計算、長期積分、アンサンブル実験を行う場合は、上記の方法で効率的な配分を大まかに見積もり、微調整することを勧める。

^{*2)} 正確を期すなら演算量を見積もる必要がある。

^{*3)} ただし、「京」のようにスレッド並列も併用するハイブリッド並列の場合には、スレッド間の演算量の不釣り合いを小さくするために X 方向よりも Y 方向の格子点数を大きく取る必要がある。

4.3 力学コアの設定

4.3.1 デカルト座標系 C-grid による力学コア

本節では、デカルト座標系 C-grid による力学コアについて記述する。SCALE-RM では、デカルト座標系 C-grid が採用されている。C-grid において、密度・熱力学変数・水蒸気といったスカラー量はセル中心で定義され、運動量やフラックスといったベクトル量の成分はセル中心から半格子ずれた位置 (staggered point) で定義される。詳細は SCALE-RM の記述文書を参照されたい。

4.3.1.1 時間積分の数値解法の設定

力学コアの時間積分の数値解法は、設定ファイル内の [PARAM_ATMOS] の (ATMOS_DYN_TYPE) で行う。

```
&PARAM_ATMOS
  ATMOS_DYN_TYPE = "HEVI", ; 表 4.3.1 より選択。
/
```

陽解法を用いる場合は時間刻み幅は音速に依存するが、陰解法を用いる場合は依存しない。多くの現実大気実験では、鉛直格子間隔は水平格子間隔よりも非常に小さい。そのため、完全陽解法 (「HEVE」) を用いると、鉛直格子間隔や音速に応じて、かなり小さな時間刻み幅を設定する必要がある。そのため、現実大気実験では「HEVI」がしばしば用いられる。

表 4.3.1: 力学過程における時間積分法の選択肢

設定名	スキームの説明	備考
HEVE	完全陽解法 (水平陽解法-鉛直陽解法)	
HEVI	水平陽解法-鉛直陰解法	現実大気実験ではこちらを推奨

4.3.1.2 時間・空間差分スキーム

時間・空間差分スキームの設定は、[PARAM_ATMOS_DYN] で設定する。現実大気実験で推奨される設定の例を以下に示す。

```

&PARAM_ATMOS_DYN
ATMOS_DYN_TINTEG_SHORT_TYPE = RK4,           ; 表 4.3.2 の時間スキームより選択
ATMOS_DYN_TINTEG_TRACER_TYPE = RK3WS2002,    ; 時間積分スキームより選択
ATMOS_DYN_FVM_FLUX_TYPE = UD3,               ; 表 4.3.2 の空間差分スキームより選択
ATMOS_DYN_FVM_FLUX_TRACER_TYPE = UD3KOREN1993, ; 空間差分スキームより選択
ATMOS_DYN_FLAG_FCT_TRACER = .false.,         ; FCT スキームを利用するかどうか
ATMOS_DYN_NUMERICAL_DIFF_COEF = 0.D0,
ATMOS_DYN_NUMERICAL_DIFF_COEF_TRACER = 0.D0,
ATMOS_DYN_enable_coriolis = .true.,
ATMOS_DYN_wdamp_height = 15.D3,              ; スポンジ層の下端高度
                                              (レイリー摩擦用)
/

```

表 4.3.2 に、時間積分・空間差分スキームの他のオプションを示す。時間刻み幅は、選択するスキームに応じて数値安定性を考慮して設定すべきである。時間ステップを決定する基準は、第 4.2.7 節に記述する。

表 4.3.2: 時間積分・空間差分スキームの設定

設定名	スキーム名
時間積分スキーム	
RK3	3 次 3 段ルンゲ・クッタスキーム (Heun)
RK3WS2002	Wicker and Skamarock (2002) 3 段ルンゲ・クッタスキーム
RK4	4 次 4 段ルンゲ・クッタスキーム
空間差分スキーム	
	最小のハ口格子数
CD2	2 次中央差分 1
CD4	4 次中央差分 2
CD6	6 次中央差分 3
UD3	3 次風上差分 2
UD5	5 次風上差分 3
UD3KOREN1993	3 次風上差分 + Koren(1993) フィルター 2

SCALE-RM において、力学の予報変数に対する移流スキーム (`ATMOS_DYN_FVM_FLUX_TYPE`) で指定) のデフォルト設定は 4 次中央差分 (CD4) である。地形の起伏が大きい計算で CD4 を用いると、格子スケールの偽の鉛直流が山頂周辺でしばしば確認される。この格子スケールの流れは、UD3 を使用することで緩和される。そのため、地形の起伏が大きい実験では UD3 を使用することを推奨する。

4.3.1.3 数値拡散

数値安定性は、計算で用いるスキームに依存する。数値拡散は安定性を良くするだろう。SCALE-RM では、数値拡散として超粘性と発散減衰 (divergence damping) を使用できる。これらの設定例を以下に示す。

```
&PARAM_ATMOS_DYN
  ATMOS_DYN_NUMERICAL_DIFF_ORDER = 1,
  ATMOS_DYN_NUMERICAL_DIFF_COEF = 1.D-4,
  ATMOS_DYN_NUMERICAL_DIFF_COEF_TRACER = 0.DO,
  ATMOS_DYN_DIVDMP_COEF = 0.DO,
/
```

超粘性の微分階数は (ATMOS_DYN_NUMERICAL_DIFF_ORDER) の4倍である。(ATMOS_DYN_NUMERICAL_DIFF_COEF) と (ATMOS_DYN_NUMERICAL_DIFF_COEF_TRACER) は、超粘性に対する無次元の係数である。この係数の値が大きいくほど減衰は強く、もしこの係数が 1 であれば、2-grid scale のノイズは 1 タイムステップで $1/e$ 倍まで減衰する。係数が 1 よりも大きい場合には、超粘性自体が数値不安定を引き起こす可能性がある。(ATMOS_DYN_NUMERICAL_DIFF_COEF) は密度・運動量・温位といった力学の予報変数に対する係数であり、(ATMOS_DYN_NUMERICAL_DIFF_COEF_TRACER) は比湿・水物質・乱流運動エネルギーといったトレーサ変数に対する係数である。UD3, UD5 等の風上スキームを用いる場合は既に数値拡散が含まれているので、(ATMOS_DYN_NUMERICAL_DIFF_COEF) をゼロに設定できる。

発散減衰もまた、数値安定性を向上させるために利用できる。その係数は (ATMOS_DYN_DIVDMP_COEF) で設定する。

4.3.1.4 正定値性

多くの場合、トレーサ移流では非負値が保証されることが要求される。UD3KOREN1993 スキームでは非負値が保証されるが、その他のスキームではそうでない。UD3KOREN1993 以外のスキームを選択した場合は、非負保証のために FCT フィルタを用いることができる。移流スキームは (ATMOS_DYN_FVM_FLUX_TRACER_TYPE) で指定し、FCT フィルタは (ATMOS_DYN_FLAG_FCT_TRACER) を .true. とすれば利用できる。

4.3.1.5 八口

必要な八口の格子点数は、表 4.3.2 に示すように空間差分スキームに依存する。x 方向や y 方向に対する八口の格子点数はそれぞれ、[PARAM_ATMOS_GRID_CARTESC_INDEX] における (IHALO) と (JHALO) で設定する。デフォルトでは八口の格子点数は 2 であり、「UD3」、「UD3UD3KOREN1993」、「CD4」に対して適切な設定である。例えば、5 次風上差分スキームに対する八口は以下のように設定する。

```
&PARAM_ATMOS_GRID_CARTESC_INDEX
  IHALO = 3,
  JHALO = 3,
/
```

4.3.1.6 コリオリ力

この小節では、SCALE-RM におけるコリオリ力の取り扱いを説明する。デフォルトではコリオリパラメータはゼロであるので、実験においてコリオリ力を導入するには (いくつかの) パラメータを

設定する必要がある。コリオリパラメータの設定には2種類あり、 f -面/ β -面および球面である。この種類は [PARAM_ATMOS_DYN] の (ATMOS_DYN_coriolis_type) で指定できる。

f -面/ β -面

(ATMOS_DYN_coriolis_type) を「PLANE」に設定した場合は、コリオリパラメータ f は $f = f_0 + \beta(y - y_0)$ と計算される。デフォルトでは $f_0 = 0$ および $\beta = 0$ あり、コリオリ力は考慮されない。 $\beta = 0$ とした平面は f -面と呼ばれ、そうでない場合は β -面と呼ばれる。 f_0, β, y_0 は、[PARAM_ATMOS_DYN] のパラメータによって次のように設定する。

```
&PARAM_ATMOS_DYN
  ATMOS_DYN_coriolis_type = 'PLANE',
  ATMOS_DYN_coriolis_f0 = 1.0D-5, ! f0
  ATMOS_DYN_coriolis_beta = 0.0D0, ! beta
  ATMOS_DYN_coriolis_y0 = 0.0D0, ! y0
  :
/
```

(ATMOS_DYN_coriolis_f0), (ATMOS_DYN_coriolis_beta) のデフォルト値はともにゼロであり、(ATMOS_DYN_coriolis_y0) のデフォルト値は領域中心の y である。

地衡風に伴うコリオリ力と地衡風バランスにある圧力勾配力を加えたい場合は、ユーザー定義ファイル `mod_user.f90` を修正する必要がある (第 5.2.1 節を参照)。`scale-rm/test/case/inertial_oscillation/20km` のテストケースは、地衡風の圧力勾配力を入れた f -面での実験例である。

球面

球面において、コリオリパラメータは $f = 2\Omega \sin(\phi)$ のように緯度に依存する。ここで、 Ω は球の角速度、 ϕ は緯度である。この場合、(ATMOS_DYN_coriolis_type) は「SPHERE」に設定する必要がある。球の角速度は [PARAM_CONST] のパラメータ (CONST_OHM) で設定する (第 4.5.3 節を参照)。各格子点の緯度は、第 4.2.4 節で説明した地図投影法に応じて決定される。

4.3.1.7 コリオリ力に伴う側面境界条件の注意点

全ての設定 (f -面、 β -面、球面) において x 方向の側面境界は周期境界にすることができる。また、 f -面では y 方向の側面境界も周期境界にすることができる。一方で、 β -面や球面においてコリオリパラメータは南北の境界で異なる値を持つために、 y 方向の側面境界に対して周期境界条件を用いることはできない。

南北境界におけるナッジング型の側面境界条件は、 f -面や β -面による実験で用いられるだろう。`scale-rm/test/case/rossby_wave/beta-plane` のテストケースは、ナッジングを行う南北境界を適用した β -面上での実験例である。ナッジングを行う境界の詳細は、第 4.2.3.5 節を参照されたい。

4.4 物理過程の設定

4.4.1 雲微物理スキーム

雲微物理スキームの選択は、`init.conf` と `run.conf` 中の `[PARAM_ATMOS]` の `(ATMOS_PHY_MP_TYPE)` で設定する。このとき、`(ATMOS_PHY_MP_TYPE)` は初期値作成用とシミュレーション実行用の設定ファイル両方で同じスキームを指定する必要がある。雲微物理スキームに対する更新間隔は `[PARAM_TIME]` で設定する。呼び出しの時間間隔の詳細な設定は、第 4.2.7 節を参照されたい。以下の例は、氷雲を含む 6-class 1 モーメントバルク法を用いるときの設定を示している。

```
&PARAM_ATMOS
  ATMOS_PHY_MP_TYPE = "TOMITA08", ; 表 4.4.1 より選択。
/
```

表 4.4.1: 雲微物理スキームの選択肢

スキーム名	スキームの説明	参考文献
OFF	雲微物理による水の相変化を計算しない	
KESSLER	3-class 1 モーメントバルク法	Kessler (1969)
TOMITA08	6-class 1 モーメントバルク法	Tomita (2008)
SN14	6-class 2 モーメントバルク法	Seiki and Nakajima (2014)
SUZUKI10	スペクトルピン法 (氷雲を含めるかはオプションで選択)	Suzuki et al. (2010)

典型的な雲微物理スキームとして、以下の 4 種類のスキームが用意されている。

1. 氷を含まない 1 モーメントバルク法 [Kessler \(1969\)](#)
このスキームでは、粒径分布関数は質量濃度のみで表現されると仮定する。カテゴリは雲粒と雨粒の 2 種類であり、空気的全密度に対する雲水や雨水の密度比が予報される。
2. 氷を含む 1 モーメントバルク法 [Tomita \(2008\)](#)
このスキームでは、粒径分布関数に関して [Kessler \(1969\)](#) と同様の仮定を置くが、水のカテゴリは雲粒、雨粒、氷粒、雪片、あられの 5 種類である。
3. 氷を含む 2 モーメントバルク法 [Seiki and Nakajima \(2014\)](#)
このスキームでは、粒径分布関数は雲粒の数密度とそれらの質量濃度によって表現される。
4. 1 モーメントピン法 [Suzuki et al. \(2010\)](#)
このスキームでは、粒径分布関数を各カテゴリに対して適切なピン数を用いて離散化すること

で表現する。水のカテゴリは、雲粒、雨粒、氷粒、雪片、霰、雹の5種類である。粒径分布を表現する精度はビン数の取り方に依存する。

上記の雲微物理スキームの精度は1から4の順に高くなるが、その分計算コストも高くなる。

SUZUKI10を選択する場合には、(ATMOS_PHY_MP_TYPE)の指定に加えて、初期値生成やシミュレーション実行用の設定ファイルの両方に以下のような設定が必要である。

```
&PARAM_ATMOS_PHY_MP_SUZUKI10_bin
nbin = 33,           ; ビンの数
ICEFLG = 1,         ; 氷雲を考慮するか?(0:なし, 1:あり)
kphase = 0,         ; 衝突併合過程における衝突カーネル関数の種類。0: hydro-
                    ; dynamic kernel, 1: Golovin type kernel (Golovin (1963)), 2:
                    ; Long type kernel (Long (1974))。詳細は SCALE-RM の記述
                    ; 文書を参照されたい。
/
```

この場合も、初期値生成の設定ファイル内の PARAM_ATMOS_PHY_MP_SUZUKI10_bin は、シミュレーション実行用の設定ファイルと同じにしなければならない。micpara.dat という雲微物理の計算に必要なファイルは、自動生成される。micpara.dat が既存の場合は、それを計算に使う。最初の行に書かれている nbin を変更した場合は、このファイルは作り直される。もし run.conf に記載されている nbin が micpara.dat 内の nbin と異なる場合は、以下のようなエラーメッセージを出力し、計算が行われずにプログラムは終了する。

```
ERROR [ATMOS_PHY_MP_suzuki10_setup] nbin in inc_tracer and nbin in micpara.dat is
different check!
```

このエラーを避けるために、前もって古い micpara.dat を消去して、再生成する必要がある。新しいデータファイルは、SUZUKI10を設定して SCALE-RM を実行すれば自動で作り直される。

4.4.2 積雲パラメタリゼーション

積雲パラメタリゼーションは、設定ファイル init.conf と run.conf 中の [PARAM_ATMOS] の (ATMOS_PHY_CP_TYPE) で指定する。積雲パラメタリゼーションを呼び出す時間間隔は、[PARAM_TIME] で設定する (詳細は第 4.2.7 節を参照)。

```
&PARAM_ATMOS
ATMOS_PHY_CP_TYPE = "KF", ; 表 4.4.2 に示すスキームから選択
/
```

表 4.4.2: 積雲パラメタリゼーションの選択肢

スキーム名	スキームの説明	参考文献
OFF	積雲パラメタリゼーションを使用しない	
KF	Kain-Fritsch 対流パラメタリゼーション	Kain and Fritsch (1990); Kain (2004)

SCALE-RM の現版では、積雲パラメタリゼーションとして KF のみ対応している。KF は質量フラックス保存型の積雲パラメタリゼーションスキームであり、サブグリッドスケールの一つの積雲を表現する。格子間隔が 5 km 以下の場合に、非自然的な強力な深い対流が計算されることを避けるために、この積雲パラメタリゼーションを使用することを推奨する。積雲パラメタリゼーションと雲微物理のスキームは、RAIN_CPRAIN_MP という名前で別々に降水量を出力する。RAIN と PREC は、両者のスキームによる合計の降水量である。つまり、 $RAIN = RAIN_CP + RAIN_MP$ 、 $PREC = PREC_CP + PREC_MP$ である。KF は大気中の水蒸気と水物質（雲水・雲氷等）の変化を計算することに注意が必要である。水物質の変化は、雲微物理の過程でさらに計算される。KF では、雲水や雲氷等の数密度は考慮されない。したがって、KF における水物質の変化と関係した数密度の変化は、指定した関数によって見積もられ、2 モーメントの雲微物理スキームへと渡される。

Kain-Fritsch スキームに対する設定

KF では、以下のチューニングパラメータを設定できる。

```
&PARAM_ATMOS_PHY_CP_KF
PARAM_ATMOS_PHY_CP_kf_trigger = 1,      ; トリガー関数の種類: 1=Kain, 3=Narita-Ohmori
PARAM_ATMOS_PHY_CP_kf_dlccape = 0.1,    ; CAPE の減率
PARAM_ATMOS_PHY_CP_kf_dlifetime = 1800, ; 深い対流の生存時間のスケール [sec]
PARAM_ATMOS_PHY_CP_kf_slifetime = 2400, ; 浅い対流の生存時間のスケール [sec]
PARAM_ATMOS_PHY_CP_kf_DEPTH_USL = 300,  ; 上昇流の発生源となる層 (updraft source layer) の
                                           探索開始時の深さ [hPa]
PARAM_ATMOS_PHY_CP_kf_prec = 1,          ; 降水の種類: 1=Ogura-Cho, 2=Kessler
PARAM_ATMOS_PHY_CP_kf_rate = 0.03,       ; Ogura-Cho の降水関数における雲水と降水の比
PARAM_ATMOS_PHY_CP_kf_thres = 1.E-3,    ; Kessler の降水関数における Autoconversion の比
PARAM_ATMOS_PHY_CP_kf_LOG = false,      ; 警告メッセージを出力するか?
/
```

ユーザーはトリガー関数として以下の 2 つから選択できる。

1. Kain タイプ [Kain \(2004\)](#)
SCALE-RM におけるデフォルト。
2. Narita and Ohmori タイプ [Narita and Ohmori \(2007\)](#)
日本域における KF 積雲パラメタリゼーションスキームに対するトリガー関数。

また、降水関数は以下の 2 つから選択できる。

1. Ogura-Cho タイプ [Ogura and Cho \(1973\)](#)
SCALE-RM におけるデフォルト。この場合、[\(PARAM_ATMOS_PHY_CP_kf_rate\)](#) というチューニングパラメータをさらに設定できる。
2. Kessler タイプ [Kessler \(1969\)](#)
Kessler type の簡単な降水関数。この場合、[\(PARAM_ATMOS_PHY_CP_kf_thres\)](#) というチューニングパラメータをさらに設定できる。

[\[PARAM_TIME\]](#) 内の [\(TIME_DT_ATMOS_PHY_CP\)](#) で指定する、KF を呼び出す時間間隔もまたチューニングパラメータであり、降水量に影響を及ぼす。[\(TIME_DT_ATMOS_PHY_CP\)](#) の最初の設定として 300 秒を推奨する。[\(PARAM_ATMOS_PHY_CP_kf_LOG\)](#) を true にした場合は、上昇流の発生源となる層がモデルの上下端を超えた際に警告メッセージを出力する。上昇流の発生源となる層はしきい値（デフォルトでは 50 hPa）よりも厚い必要があるが、この条件を満たさなくても計算は止まらない。

4.4.3 乱流スキーム

Large-eddy シミュレーション (LES) において、サブグリッドスケール乱流モデルは、移流項によるサブグリッドスケールへのエネルギーカスケードを表現するために存在する。

乱流スキームの選択は、`init.conf` と `run.conf` 中の `[PARAM_ATMOS]` の `(ATMOS_PHY_TB_TYPE)` で設定する。乱流スキームを呼び出す時間間隔は、`[PARAM_TIME]` で設定する (詳細は第 4.2.7 節を参照)。

```
&PARAM_ATMOS
  ATMOS_PHY_TB_TYPE = "MYNN", ; 表 4.4.3 より選択。
/
```

表 4.4.3: 乱流スキームの選択肢

スキーム名	スキームの説明	参考文献
OFF	乱流過程を計算しない	
SMAGORINSKY	Smagorinsky Lilly 型のサブグリッドスケール乱流モデル	Smagorinsky (1963); Lilly (1962); Brown et al. (1994); Scotti et al. (1993)
D1980	Deardorff 型のサブグリッドスケール乱流モデル	Deardorff (1980)

SMAGORINSKY スキームは、RANS シミュレーションにおいて水平渦粘性としても使用できる。惑星境界層パラメタリゼーション (第 4.4.4 節) は、鉛直混合のみを取り扱うスキームである。RANS シミュレーションにおいて水平粘性を考慮したければ、水平混合のためにサブグリッドスケール乱流モデルを使用されたい。その場合は、以下のように `[PARAM_ATMOS_PHY_TB_SMG]` 内の `(ATMOS_PHY_TB_SMG_horizontal)` を `.true.` にしなければならない。

```
&PARAM_ATMOS_PHY_TB_SMG
  ATMOS_PHY_TB_SMG_horizontal = .true.,
/
```

4.4.4 惑星境界層スキーム

惑星境界層 (PBL) パラメタリゼーションは、惑星境界層における乱流による鉛直混合を表現するためのスキームである。このスキームは、レイノルズ平均ナビエ-ストークス方程式の計算 (RANS) で用いられる。

惑星境界層パラメタリゼーションスキームは、設定ファイル `init.conf`、`run.conf` 中の `[PARAM_ATMOS]` の `(ATMOS_PHY_BL_TYPE)` で指定する。このスキームを呼び出す時間間隔は、`[PARAM_TIME]` で設定する (詳細は第 4.2.7 節を参照)。

```
&PARAM_ATMOS
  ATMOS_PHY_BL_TYPE = "MYNN", ; 表 4.4.4 に示したスキームから選択
/
```

表 4.4.4: 惑星境界層スキームの選択

スキーム名	スキームの説明	参考文献
OFF	惑星境界層の過程を計算しない	
MYNN	MYNN level 2.5 boundary scheme	Mellor and Yamada (1982) ; Nakanishi and Niino (2004)

惑星境界層スキームは鉛直混合のみを計算する。RANS において水平渦粘性を考慮したい場合には、サブグリッドスケール乱流モデルを水平混合を表現するために用いることができる (第 4.4.3 節を参照)。

4.4.5 放射スキーム

放射スキームの選択は、init.conf と run.conf 中の [PARAM_ATMOS] の (ATMOS_PHY_RD_TYPE) で設定する。放射スキームが呼び出される時間間隔は、[PARAM_TIME] で設定する (第 4.2.7 節を参照)。

```
&PARAM_ATMOS
  ATMOS_PHY_RD_TYPE = "MSTRNX", ; 表 4.4.5 に示した放射スキームから選択。
/
```

表 4.4.5: 放射スキームの選択肢

スキーム名	スキームの説明	参考文献
OFF または NONE	放射過程を計算しない	
OFFLINE	外部ファイルから与えた放射データを指定	
MSTRNX	mstrnX (相関 k 分布法ブロードバンド大気放射伝達モデル)	Sekiguchi and Nakajima (2008)

OFFLINE の場合の設定

[PARAM_ATMOS] の (ATMOS_PHY_RD_TYPE) を OFFLINE とした場合は、データのファイル名やその情報は [PARAM_ATMOS_PHY_RD_OFFLINE] で設定する。

```

&PARAM_ATMOS_PHY_RD_OFFLINE
  ATMOS_PHY_RD_OFFLINE_BASENAME = "",
  ATMOS_PHY_RD_OFFLINE_AXISTYPE = "XYZ",

  ATMOS_PHY_RD_OFFLINE_ENABLE_PERIODIC_YEAR = .false.,
  ATMOS_PHY_RD_OFFLINE_ENABLE_PERIODIC_MONTH = .false.,
  ATMOS_PHY_RD_OFFLINE_ENABLE_PERIODIC_DAY = .false.,
  ATMOS_PHY_RD_OFFLINE_STEP_FIXED = 0,

  ATMOS_PHY_RD_OFFLINE_CHECK_COORDINATES = .true.,

  ATMOS_PHY_RD_OFFLINE_STEP_LIMIT = 0,

  ATMOS_PHY_RD_OFFLINE_DIFFUSE_RATE = 0.5D0,

/

```

; 外部データのファイルのベース名
 ; ファイルにおける 3 次元データの
 空間次元順。'XYZ' or 'ZXY'
 ; 年ごとの繰り返しデータかどうか
 ; 月ごとの繰り返しデータかどうか
 ; 日ごとの繰り返しデータかどうか
 ; とある決まった時間ステップのデー
 タのみを使う場合に、そのステップ
 番号を指定する。時間変化するデー
 タを与える場合は 0 以下を指定する。
 ; ファイル内の座標変数とモデル実
 行時の値の整合性を確認するかどうか
 ; 読み出すデータの時間ステップ数
 の上限。この数を越える時間ステッ
 プのデータは読まれない。0 を与え
 ると上限無し。
 ; 地表面下向き直達短波放射のデー
 タを与えない場合に使われる散乱率
 (散乱日射/全天日射)

外部データのファイル形式は、初期値/境界値データのファイルと同じ軸情報を持つ netCDF 形式のファイルである。外部データとして、表 4.4.6 にある変数を与える。

表 4.4.6: 外部ファイルとして与える放射データ

変数名	変数の説明	次元数
RFLX_LW_up	上向き長波放射フラックス	空間 3 次元+時間 1 次元
RFLX_LW_dn	下向き長波放射フラックス	空間 3 次元+時間 1 次元
RFLX_SW_up	上向き短波放射フラックス	空間 3 次元+時間 1 次元
RFLX_SW_dn	下向き短波放射フラックス	空間 3 次元+時間 1 次元
SFLX_LW_up	地表面上向き長波放射	空間 2 次元+時間 1 次元
SFLX_LW_dn	地表面下向き長波放射	空間 2 次元+時間 1 次元
SFLX_SW_up	地表面上向き短波放射	空間 2 次元+時間 1 次元
SFLX_SW_dn	地表面下向き短波放射	空間 2 次元+時間 1 次元
SFLX_SW_dn_dir	地表面下向き直達短波放射	空間 2 次元+時間 1 次元 オプション

MSTRNX に対する設定

放射計算のための太陽放射量は、モデルの計算設定に従って日付、時刻、緯度経度を用いて計算される。理想実験のために、計算領域全域で緯度経度や時刻を固定した値を任意に与えることもできる。また、太陽定数の変更が可能である。これらは、以下のように [PARAM_ATMOS_SOLARINS] で設定する。

<code>&PARAM_ATMOS_SOLARINS</code>	
<code>ATMOS_SOLARINS_constant = 1360.250117</code>	太陽定数 [W/m ²]
<code>ATMOS_SOLARINS_set_ve = .false.</code>	春分点条件に設定するかどうか
<code>ATMOS_SOLARINS_set_ideal = .false.</code>	黄道傾斜角と離心率を固定するかどうか
<code>ATMOS_SOLARINS_obliquity = 0.0</code>	ATMOS_SOLARINS_set_ideal=.true. の場合の黄道傾斜角 [deg.]
<code>ATMOS_SOLARINS_eccentricity = 0.0</code>	ATMOS_SOLARINS_set_ideal=.true. の場合の離心率
<code>ATMOS_SOLARINS_fixedlatlon = .false.</code>	放射計算の緯度経度を固定するかどうか
<code>ATMOS_SOLARINS_lon = 135.221</code>	ATMOS_SOLARINS_fixedlatlon=.true. の場合の経度 [deg.]
<code>ATMOS_SOLARINS_lat = 34.653</code>	ATMOS_SOLARINS_fixedlatlon=.true. の場合の緯度 [deg.]
<code>ATMOS_SOLARINS_fixeddate = .false.</code>	放射計算の日時を固定するかどうか
<code>ATMOS_SOLARINS_date = -1,-1,-1,-1,-1,-1,</code>	ATMOS_SOLARINS_fixeddate=.true. の場合の年月日時刻 [Y,M,D,H,M,S]
<code>/</code>	

(`ATMOS_SOLARINS_set_ideal`) を `.true.` とした場合は、(`ATMOS_SOLARINS_obliquity`) や (`ATMOS_SOLARINS_eccentricity`) でそれぞれ指定した黄道傾斜角 = 地軸 (deg.) と離心率を用いて太陽放射を計算する。これらの設定は理想実験や地球以外の惑星を想定した実験を行うときに有用である。

(`ATMOS_SOLARINS_fixedlatlon`) を `.true.` とした場合は、計算領域全域において (`ATMOS_SOLARINS_lon`, `ATMOS_SOLARINS_lat`) で指定した緯度経度をもとに太陽放射を計算する。これらのデフォルト値は、`[PARAM_MAPPROJECTION]` で設定した (`MAPPROJECTION_basepoint_lon`, `MAPPROJECTION_basepoint_lat`) である。`[PARAM_MAPPROJECTION]` の説明は第 4.2.4 節を参照されたい。

(`ATMOS_SOLARINS_fixeddate`) を `.true.` とした場合は、(`ATMOS_SOLARINS_date`) で指定した日時(年、月、日、時、分、秒)を用いて太陽放射を計算する。このとき、負の値を指定したものは固定されない。例えば、(`ATMOS_SOLARINS_date`) を 1950,3,21,-1,-1,-1 と指定したときには、日付は 1950 年 3 月 21 日(春分)に固定されるが、太陽放射の日変化が計算で考慮される。

(`ATMOS_SOLARINS_set_ve`) を `.true.` とした場合は、理想実験で用いる春分点条件のための設定がまとめて行われる。このオプションは、黄道傾斜角と離心率をゼロ、計算領域全域を緯度 0 度、経度 0 度、日時を 1950 年 3 月 21 日 12 時 00 分 00 秒に設定する。上述した (`ATMOS_SOLARINS_set_ideal`, `ATMOS_SOLARINS_fixedlatlon`, `ATMOS_SOLARINS_fixeddate`) を用いてこれらの値を設定した場合は、それらの設定が優先される。

実験設定によっては、モデル上端の高度が大気の高さと比べて随分と低く、10-20 km ということがしばしばある。この状況に対処するために、放射計算に対する上端高度を、モデル上端の高度とは別に設定できる。放射計算用の上端高度のとり方は、放射スキームのパラメータファイルに依存する。例えば、MSTRNX を使う場合は、そのデフォルトのパラメータテーブルは最上層が 100 km であることを想定している。モデル上端よりも高高度で放射計算を行うために、いくつかの層が用意される。デフォルトでは追加される層は 10 層である。もしモデル上端が 22 km であれば、放射計算のために、格子間隔 7.8 km の層を 10 層追加する。これらは `[PARAM_ATMOS_PHY_RD_MSTRN]` で設定する。

MSTRNX では、放射計算のためのパラメータテーブルが必要である。デフォルトでは、太陽放射から赤外放射までの波長帯を 29 バンド 111 チャンネルに分割し、雲・エアロゾル粒子は 9 種類、粒径を 8 ピンで表した時のテーブルを用いる。ディレクトリ `scale-rm/test/data/rad/` に、3 種類のパラメータファイルを用意している。


```
scale-rm/test/data/rad/PARAG.29      ; 気体吸収パラメータファイル
scale-rm/test/data/rad/PARAPC.29     ; 雲・エアロゾル吸収・散乱パラメータファイル
scale-rm/test/data/rad/VARDATA.RM29  ; 雲・エアロゾル粒径パラメータファイル
```

これらのファイルは、`[PARAM_ATMOS_PHY_RD_MSTRN]` で以下のように指定する。

<code>&PARAM_ATMOS_PHY_RD_MSTRN</code>	
<code>ATMOS_PHY_RD_MSTRN_KADD = 10</code>	モデルトップから TOA までの間の大気層数
<code>ATMOS_PHY_RD_MSTRN_TOA = 100.0</code>	放射計算で考慮する大気最上層 (TOA) の高さ [km] (パラメータファイルに依存)
<code>ATMOS_PHY_RD_MSTRN_nband = 29</code>	波長帯ビンの数 (パラメータファイルに依存)
<code>ATMOS_PHY_RD_MSTRN_nptype = 9</code>	エアロゾルの種類の数 (パラメータファイルに依存)
<code>ATMOS_PHY_RD_MSTRN_nradius = 8</code>	エアロゾル粒径ビンの数 (パラメータファイルに依存)
<code>ATMOS_PHY_RD_MSTRN_GASPARA_IN_FILENAME = "PARAG.29"</code>	気体吸収パラメータの入力ファイル
<code>ATMOS_PHY_RD_MSTRN_AEROPARA_IN_FILENAME = "PARAPC.29"</code>	雲・エアロゾル吸収・散乱パラメータの入力ファイル
<code>ATMOS_PHY_RD_MSTRN_HYGROPARA_IN_FILENAME = "VARDATA.RM29"</code>	雲・エアロゾル粒径パラメータの入力ファイル
<code>ATMOS_PHY_RD_MSTRN_ONLY_QCI = .false.</code>	放射計算で雲水・雲氷のみを考慮する (雨・雪・あられを無視する) かどうか
<code>/</code>	

上記の MSTRNX のパラメータファイルは、version 5.2 リリース時に更新された。そのため、SCALE-RM の最新版では新しいパラメータファイルを用いることを推奨する。version 5.1 以前で提供していた MSTRNX のパラメータファイルは、ディレクトリ `scale-rm/test/data/rad/OpenCLASTR` 以下に置いている。粒子の種類の数や粒径のピン数が、新しいパラメータファイルとは異なる。これらの古いパラメータファイルを利用したい場合は、`[PARAM_ATMOS_PHY_RD_MSTRN]` において下記のように (`ATMOS_PHY_RD_MSTRN_nptype`, `ATMOS_PHY_RD_MSTRN_nradius`) を変更する必要がある。

```
ATMOS_PHY_RD_MSTRN_nptype = 11
ATMOS_PHY_RD_MSTRN_nradius = 6
```

放射計算のために追加した層では、気温、気圧、二酸化炭素やオゾン等の気体濃度の鉛直分布を与える必要がある。この鉛直分布の与え方は 2 種類あり、気候値またはユーザが準備した ASCII 形式の入力データを用いることができる。

気候値を与える場合は、SCALE-RM では気温・気圧については CIRA86 ^{*1)}(Committee on Space Research; NASA National Space Science Data Center, 2006)、気体種については MIPAS2001(Remedios et al., 2007) をデータベースとして用意している。気候値の分布は、これらのデータベースを日付・時刻・緯度経度について内挿することで得られる。**[PARAM_ATMOS_SOLARINS]** において日付と位置を固定した場合には、これらの設定に従って分布の計算は行われる。気候値の入力ファイルもまた、ディレクトリ `scale-rm/test/data/rad/` に用意している。

```
scale-rm/test/data/rad/cira.nc          ; CIRA86 データ (NetCDF format)
scale-rm/test/data/rad/MIPAS/day.atm    ; MIPAS2011 データ (中緯度) (ASCII format)
scale-rm/test/data/rad/MIPAS/equ.atm    ; MIPAS2011 データ (熱帯) (ASCII format)
scale-rm/test/data/rad/MIPAS/sum.atm    ; MIPAS2011 データ (夏半球高緯度) (ASCII format)
scale-rm/test/data/rad/MIPAS/win.atm    ; MIPAS2011 データ (冬半球高緯度) (ASCII format)
```

これらのファイル名やディレクトリ名は、**[PARAM_ATMOS_PHY_RD_PROFILE]** で指定する。例えば、上記の 5 ファイルを実行ディレクトリに配置した場合は以下のように設定する。

&PARAM_ATMOS_PHY_RD_PROFILE	
<code>ATMOS_PHY_RD_PROFILE_use_climatology = .true.</code>	CIRA86 と MIPAS2001 の気候値を利用するかどうか
<code>ATMOS_PHY_RD_PROFILE_CIRA86_IN_FILENAME = "cira.nc"</code>	CIRA86 ファイル名
<code>ATMOS_PHY_RD_PROFILE_MIPAS2001_IN_BASENAME = "."</code>	MIPAS2001 ファイルがあるディレクトリ名

放射計算で考慮される気体は、水蒸気 (H_2O)、二酸化炭素 (CO_2)、オゾン (O_3)、一酸化二窒素 (N_2O)、一酸化炭素 (CO)、メタン (CH_4)、酸素 (O_2)、クロロフルオロカーボン類 (CFCs) である。これらの濃度は以下のように、**[PARAM_ATMOS_PHY_RD_PROFILE]** でゼロに設定することができる。

&PARAM_ATMOS_PHY_RD_PROFILE	
<code>ATMOS_PHY_RD_PROFILE_USE_CO2 = .true.</code>	false の場合、 CO_2 濃度を常に 0 に設定する
<code>ATMOS_PHY_RD_PROFILE_USE_O3 = .true.</code>	false の場合、 O_3 濃度を常に 0 に設定する
<code>ATMOS_PHY_RD_PROFILE_USE_N2O = .true.</code>	false の場合、 N_2O 濃度を常に 0 に設定する
<code>ATMOS_PHY_RD_PROFILE_USE_CO = .true.</code>	false の場合、 CO 濃度を常に 0 に設定する
<code>ATMOS_PHY_RD_PROFILE_USE_CH4 = .true.</code>	false の場合、 CH_4 濃度を常に 0 に設定する
<code>ATMOS_PHY_RD_PROFILE_USE_O2 = .true.</code>	false の場合、 O_2 濃度を常に 0 に設定する
<code>ATMOS_PHY_RD_PROFILE_USE_CFC = .true.</code>	false の場合、すべての CFC 濃度を常に 0 に設定する

ユーザが指定した分布を用いる場合は、高度 [m]、気圧 [Pa]、気温 [K]、水蒸気量 [kg/kg]、オゾン濃度 [kg/kg] を ASCII 形式で準備しなければならない。水蒸気とオゾン以外の気体濃度はゼロと設定され、時間変化は取り扱われない。ユーザが用意したファイルの例は、以下に用意されている。

```
scale-rm/test/data/rad/rad_o3_profs.txt
```

^{*1)}<http://catalogue.ceda.ac.uk/uuid/4996e5b2f53ce0b1f2072adadaeda262>

これを用いるには、[PARAM_ATMOS_PHY_RD_PROFILE] 内の (ATMOS_PHY_RD_PROFILE_use_climatology) を .false. に設定し、(ATMOS_PHY_RD_PROFILE_USER_IN_FILENAME) にファイル名とディレクトリ名を指定する必要がある。

```
&PARAM_ATMOS_PHY_RD_PROFILE
  ATMOS_PHY_RD_PROFILE_use_climatology = .false.
  ATMOS_PHY_RD_PROFILE_USER_IN_FILENAME = "rad_o3_profs.txt"
/
```

CIRA86 と MIPAS2001 の気候値を利用するかどうか
気候値を利用しない場合に用いるユーザ指定ファイルの名前 (ASCII 形式)

ユーザが準備するファイルにおいて、デフォルトのモデル設定とは独立に層数や層の高度を与えることができる。実行時に、モデルの層での値は与えた分布から内挿される。ただし、放射計算で想定される最上層の高度が、入力した分布の高度よりも高い場合は、外挿が行われることに注意が必要である。

4.4.6 大気下端境界における表面フラックス

大気下端境界における表面フラックススキームは、[PARAM_ATMOS] の (ATMOS_PHY_SF_TYPE) で以下のように設定する。

```
&PARAM_ATMOS
  ATMOS_PHY_SF_TYPE = "COUPLE", ; 表 4.4.7 から表面フラックススキームを選択。
/
```

海面・陸面・都市モデルを使用しない場合は、大気下端境界は理想実験で用いられる仮想的な地表面であることを想定している。この表面フラックススキームが呼び出される時間間隔は、[PARAM_TIME] で設定する (詳細は第 4.2.7 節を参照)。海面・陸面・都市モデルを用いる場合は、(ATMOS_PHY_SF_TYPE) を “COUPLE” に設定する。

一定値を用いる場合の設定

(ATMOS_PHY_SF_TYPE) を "CONST" とした場合は、表面フラックスは run.conf で指定した値に固定できる。下記の値はデフォルト設定である。

表 4.4.7: 大気下端境界の選択肢

スキーム名	スキームの説明
NONE	地表面フラックスを計算しない (海面・陸面・都市モデルの実行設定に応じて COUPLE に変更される)
OFF	地表面フラックスを計算しない
CONST	地表面フラックスを一定値に固定
BULK	地表面フラックスをバルクモデルで計算
COUPLE	海面・陸面・都市モデルから表面フラックスを受け取る

```

&PARAM_ATMOS_PHY_SF_CONST
  ATMOS_PHY_SF_FLG_MOM_FLUX = 0          0: バルク係数を一定にする
                                           1: 摩擦速度を一定にする
  ATMOS_PHY_SF_U_minM = 0.0E0            絶対速度の下限値 [m/s]
  ATMOS_PHY_SF_Const_Cm = 0.0011E0       運動量に対する一定バルク係数値
                                           (ATMOS_PHY_SF_FLG_MOM_FLUX = 0 のとき有効)
  ATMOS_PHY_SF_CM_min = 1.0E-5           運動量に対するバルク係数の下限値
                                           (ATMOS_PHY_SF_FLG_MOM_FLUX = 1 のとき有効)
  ATMOS_PHY_SF_Const_Ustar = 0.25E0      一定摩擦係数値 [m/s]
                                           (ATMOS_PHY_SF_FLG_MOM_FLUX = 1 のとき有効)
  ATMOS_PHY_SF_Const_SH = 15.E0          一定地表面顕熱フラックス値 [W/m2]
  ATMOS_PHY_SF_FLG_SH_DIURNAL = .false.  顕熱フラックスに日変化をつけるか否か [logical]
  ATMOS_PHY_SF_Const_FREQ = 24.E0        顕熱フラックスに日変化を付けるときのサイクル [hour]
  ATMOS_PHY_SF_Const_LH = 115.E0         一定地表面潜熱フラックス値 [W/m2]
/

```

バルクモデルを用いる場合の設定

(**ATMOS_PHY_SF_TYPE**) を BULK とした場合は、指定した表面温度と粗度長を用いてバルクモデルによって表面フラックスが計算される。蒸発効率、0 から 1 の範囲で任意に与えることができる。この柔軟性によって、海面だけでなく陸面を想定した理想実験を行える。蒸発効率は run.conf 中の (**ATMOS_PHY_SF_BULK_beta**) の [**PARAM_ATMOS_PHY_SF_BULK**] で以下のように指定する。

```

&PARAM_ATMOS_PHY_SF_BULK
  ATMOS_PHY_SF_BULK_beta = 1.0, ; 蒸発効率 (0 から 1 までの値でなければならない)。値を 0 に設
                                定した場合、表面は完全に乾燥しているとする。1 に設定した場
                                合は (海面のように) 表面は完全に湿っているとする。
/

```

バルク交換係数のスキームは、run.conf 中の [**PARAM_BULKFLUX**] の (**BULKFLUX_TYPE**) で以下のよう

```

&PARAM_BULKFLUX
  BULKFLUX_TYPE = "B91W01", ; 表 4.4.8 に示すバルク交換係数スキームから選択。
/

```

表 4.4.8: バルク交換係数スキームの選択肢

スキーム名	スキームの説明	参考文献
B91W01	普遍関数によるバルク法 (デフォルト)	Beljaars and Holtslag (1991); Wilson (2001)
U95	Louis 型のバルク法、(Louis (1979) の改良版)	Uno et al. (1995)

4.4.7 海洋モデル

海面過程は、海面の状態の更新と大気-海面間のフラックス計算の2つに大別される。これらの過程を計算する時間間隔は、[PARAM_TIME] で設定する (詳細は第 4.2.7 節を参照)。

海洋モデルのスキームは、設定ファイル中の [PARAM_OCEAN] の (OCEAN_DYN_TYPE), (OCEAN_SFC_TYPE), (OCEAN_ICE_TYPE), (OCEAN_ALB_TYPE), (OCEAN_RGN_TYPE) で設定する。

```
&PARAM_OCEAN
  OCEAN_DYN_TYPE = "SLAB",           ; 表 4.4.9 に示す海洋表層の取り扱いから選択
  OCEAN_SFC_TYPE = "FIXED-TEMP",    ; Select the ocean surface type shown in 表 4.4.10 に示す海面
                                     ; スキームから選択
  OCEAN_ICE_TYPE = "SIMPLE",         ; 表 4.4.11 に示す海氷スキームから選択
  OCEAN_ALB_TYPE = "NAKAJIMA00",    ; 表 4.4.12 に示す海面アルベドスキームから選択
  OCEAN_RGN_TYPE = "MOON07",        ; 表 4.4.13 に示す海面粗度の計算方法から選択
/
```

表 4.4.9: 海洋表層スキームの選択肢

スキーム名	スキームの説明
NONE または OFF	海面モデルを利用しない
INIT	初期値のまま固定する
OFFLINE	外部データによって更新する
SLAB	板海モデル

表 4.4.10: 海洋表面スキームの選択肢 (OCEAN_SFC_TYPE)。現版では 1 種類のみ。

スキーム名	スキームの説明
FIXED-TEMP	海面温度を診断せずにフラックスを計算する。

表 4.4.11: 海氷スキームの選択肢 (OCEAN_ICE_TYPE).

スキーム名	スキームの説明
NONE	海洋モデルを無効化
INIT	初期条件に固定
SIMPLE	簡単な海氷モデル

表 4.4.12: 表面アルベドスキームの選択肢 (OCEAN_ALB_TYPE).

スキーム名	スキームの説明	参考文献
INIT	初期条件に固定	
CONST	一定値を使用	
NAKAJIMA00	太陽天頂角から短波放射に対するアルベドを計算	Nakajima et al. (2000)

表 4.4.13: 海面粗度の計算方法の選択 (OCEAN_RGN_TYPE).

スキーム名	スキームの説明	参考文献
MOON07	経験式に基づく (時間変化あり)	Moon et al. (2007)
INIT	初期条件に固定	
CONST	一定値を使用	
MILLER92	経験式に基づく (時間変化なし)	Miller et al. (1992)

4.4.7.1 海洋表層スキーム

海洋比率がゼロでない (すなわち陸面比率が 1.0 以下である) 表面の格子点では、物理量は海洋サブモデルによって計算されなければならない。陸-海洋比率の設定は、`[PARAM_LANDUSE]` によって制御できる。海洋比率を持つ格子点が存在するにも関わらず、`(OCEAN_DYN_TYPE)` を "NONE" や "OFF" に設定した場合は、以下のエラーメッセージがログファイルに出力される。

```
ERROR [CPL_vars_setup] Ocean fraction exists, but ocean component has not been called.
Please check this inconsistency. STOP.
```

`(OCEAN_DYN_TYPE)` を "SLAB" と設定した場合は、海洋表層は板海 (slab ocean) として取り扱われる。板海の温度は、表面からの熱フラックスによって時間とともに変化する。熱容量を支配する板海の深さは、`[PARAM_OCEAN_DYN_SLAB]` 内の `(OCEAN_DYN_SLAB_DEPTH)` で指定できる。

```
&PARAM_OCEAN_DYN_SLAB
  OCEAN_DYN_SLAB_DEPTH = 10.0_RP, ; 板海の深さ [m]
/
```

板海モデルでは、外部データを用いることで海面温度を緩和 (すなわちナッジング) させることができる。ナッジングのパラメータは `[PARAM_OCEAN_DYN_SLAB]` で指定する。

```
&PARAM_OCEAN_DYN_SLAB
  OCEAN_DYN_SLAB_nudging = .false., ; 海洋の変数に対してナッジング
                                      ; を行うか?
  OCEAN_DYN_SLAB_nudging_tau = 0.0_DP, ; ナッジングのための緩和の時定
                                      ; 数
  OCEAN_DYN_SLAB_nudging_tau_unit = "SEC", ; 緩和の時定数の単位
  OCEAN_DYN_SLAB_nudging_basename = "", ; 入力ファイルのベース名
  OCEAN_DYN_SLAB_nudging_enable_periodic_year = .false., ; 年周期データか?
  OCEAN_DYN_SLAB_nudging_enable_periodic_month = .false., ; 月周期データか?
  OCEAN_DYN_SLAB_nudging_enable_periodic_day = .false., ; 日周期データか?
  OCEAN_DYN_SLAB_nudging_step_fixed = 0, ; データの特定のステップ数を用
                                      ; いるか?
  OCEAN_DYN_SLAB_nudging_offset = 0.0_RP, ; 変数のオフセット値
  OCEAN_DYN_SLAB_nudging_defval = UNDEF, ; 変数のデフォルト値
  OCEAN_DYN_SLAB_nudging_check_coordinates = .true., ; 変数の座標を確認するか?
  OCEAN_DYN_SLAB_nudging_step_limit = 0, ; データを読み込む時間ステップ
                                      ; 数の最大値
/
```

`(OCEAN_DYN_SLAB_nudging_tau)` が 0 であるときは、海面温度の値は外部ファイルによって完全に置き換わる。`(OCEAN_DYN_SLAB_nudging_step_fixed)` が 1 以下であれば、現時刻における値は外部データを時間内挿することで計算される。`(OCEAN_DYN_SLAB_nudging_step_fixed)` に特定のステップを指定した場合は、そのステップのデータが時間内挿することなく常に用いられる。`(OCEAN_DYN_SLAB_nudging_step_limit)` に 0 よりも大きい値を設定した場合は、その制限を超える時間ステップのデータを読み込まず、最後に読み込んだデータをナッジングに用いる。この制限は、`(OCEAN_DYN_SLAB_nudging_step_limit)` が 0 の場合には設定されない。

(OCEAN_DYN_TYPE) を"OFFLINE"に設定した場合は、海洋表層の力学過程や物理過程は計算されない。海面温度の時間変化は外部ファイルによって与えられる。この設定は、SCALE の旧バージョンで (OCEAN_TYPE) = "FILE"とした場合と同じである。また、板海スキームで (OCEAN_DYN_SLAB_nudging_tau) を 0 に設定した場合とも同じである。

```
&PARAM_OCEAN_DYN_OFFLINE
OCEAN_DYN_OFFLINE_basename = "",           ; 入力ファイルのベース名
OCEAN_DYN_OFFLINE_enable_periodic_year = .false., ; 年周期データか?
OCEAN_DYN_OFFLINE_enable_periodic_month = .false., ; 月周期データか?
OCEAN_DYN_OFFLINE_enable_periodic_day = .false., ; 日周期データか?
OCEAN_DYN_OFFLINE_step_fixed = 0,           ; データの特定のステップ数を用いるか?
OCEAN_DYN_OFFLINE_offset = 0.0_RP,         ; 変数のオフセット値
OCEAN_DYN_OFFLINE_defval = UNDEF,          ; 変数のデフォルト値
OCEAN_DYN_OFFLINE_check_coordinates = .true., ; 変数の座標を確認するか?
OCEAN_DYN_OFFLINE_step_limit = 0,          ; データを読み込む時間ステップ数の最大値
/
```

オフラインモードにおける外部ファイルに対する各パラメータは、板海スキームにおけるナッジングの設定に対するパラメータと同様である。

4.4.7.2 海面過程

海面過程は以下のサブプロセスを含む。

- 海氷のない海洋 (open ocean) 面の計算
 - 海面アルベドの計算
 - 海面粗度長の計算
 - 大気海洋間の熱/蒸発/放出フラックスの計算
- 海氷面の計算
 - 海氷面アルベドの計算
 - 海氷面粗度長の計算
 - 海氷と海洋表層間の熱伝導計算
 - 大気と海氷間の熱/蒸発/放出フラックス
 - 海氷と海洋表層間の熱フラックスと水フラックスの計算

open ocean での海面アルベドは、(OCEAN_ALB_TYPE) で選択したスキームによって設定される。(OCEAN_ALB_TYPE) を"CONST"とした場合は、open ocean での海面アルベドは (PARAM_OCEAN_PHY_ALBEDO_const) で指定した定数値となる。(OCEAN_ALB_TYPE) を"NAKAJIMA00"とした場合は、短波放射に対するアルベドは太陽天頂角に依存するように計算されるが、長波放射 (IR) に対しては (PARAM_OCEAN_PHY_ALBEDO_const) で設定したアルベドの値が用いられる。

```
&PARAM_OCEAN_PHY_ALBEDO_const
OCEAN_PHY_ALBEDO_IR_dir = 0.05D0, ; 長波 (赤外) 直達光に対する海面アルベド
OCEAN_PHY_ALBEDO_IR_dif = 0.05D0, ; 長波 (赤外) 散乱光に対する海面アルベド
OCEAN_PHY_ALBEDO_NIR_dir = 0.07D0, ; 長波 (近赤外) 直達光に対する海面アルベド
OCEAN_PHY_ALBEDO_NIR_dif = 0.06D0, ; 短波 (近赤外) 散乱光に対する海面アルベド
OCEAN_PHY_ALBEDO_VIS_dir = 0.07D0, ; 短波 (可視) 直達光に対する海面アルベド
OCEAN_PHY_ALBEDO_VIS_dif = 0.06D0, ; 短波 (可視) 散乱光に対する海面アルベド
/
```

海氷面のアルベドは (OCEAN_ALB_TYPE) に関わらず一定値であり、その値は (PARAM_OCEAN_PHY_ALBEDO_seaice) で設定する。

```
&PARAM_OCEAN_PHY_ALBEDO_seaice
OCEAN_PHY_ALBEDO_seaice_IR_dir = 0.05D0, ; 長波 (赤外) 直達光に対する海氷面アルベド
OCEAN_PHY_ALBEDO_seaice_IR_dif = 0.05D0, ; 長波 (赤外) 散乱光に対する海氷面アルベド
OCEAN_PHY_ALBEDO_seaice_NIR_dir = 0.60D0, ; 長波 (近赤外) 直達光に対する海氷面アルベド
OCEAN_PHY_ALBEDO_seaice_NIR_dif = 0.60D0, ; 短波 (近赤外) 散乱光に対する海氷面アルベド
OCEAN_PHY_ALBEDO_seaice_VIS_dir = 0.80D0, ; 短波 (可視) 直達光に対する海氷面アルベド
OCEAN_PHY_ALBEDO_seaice_VIS_dif = 0.80D0, ; 短波 (可視) 散乱光に対する海氷面アルベド
/
```

海面粗度長は (OCEAN_RGN_TYPE) で選択したスキームを用いて計算される。(OCEAN_RGN_TYPE) を"CONST"とした場合は、(PARAM_OCEAN_PHY_ROUGHNESS_const) で設定したパラメータを用いる。

```
&PARAM_OCEAN_PHY_ROUGHNESS_const
OCEAN_PHY_ROUGHNESS_ZOM = 1.0D-5, ; 運動量に対する海面粗度長 [m]
OCEAN_PHY_ROUGHNESS_ZOH = 1.0D-5, ; 熱に対する海面粗度長 [m]
OCEAN_PHY_ROUGHNESS_ZOE = 1.0D-5, ; 水蒸気に対する海面粗度長 [m]
/
```

(OCEAN_RGN_TYPE) を"MOON07"あるいは"MILLER92"に設定した場合は、選択したスキーム内で運動量・熱・水蒸気に対する粗度長が計算される。(PARAM_OCEAN_PHY_ROUGHNESS)を設定することで、これらの値の計算における幾つかの制限値を指定できる。

```
&PARAM_OCEAN_PHY_ROUGHNESS
OCEAN_PHY_ROUGHNESS_visck = 1.5D-5, ; 動粘性 [m2/s]
OCEAN_PHY_ROUGHNESS_Ustar_min = 1.0D-3, ; 摩擦速度の最小制限値 [m/s]
OCEAN_PHY_ROUGHNESS_ZOM_min = 1.0D-5, ; 運動量に対する粗度長の最小制限値 [m]
OCEAN_PHY_ROUGHNESS_ZOH_min = 1.0D-5, ; 熱に対する粗度長の最小制限値 [m]
OCEAN_PHY_ROUGHNESS_ZOE_min = 1.0D-5, ; 水蒸気に対する粗度長の最小制限値 [m]
/
```

海氷面の粗度長は (OCEAN_RGN_TYPE) に依らず一定値であり、その値は (PARAM_OCEAN_PHY_ROUGHNESS_seaice) で設定する。(PARAM_OCEAN_PHY_ROUGHNESS) で指定した粗度長の最小制限値が海氷面に対しても適用される。

```
&PARAM_OCEAN_PHY_ROUGHNESS_seaice
OCEAN_PHY_ROUGHNESS_seaice_ZOM = 2.0D-2, ; 運動量に対する海水面粗度長 [m]
OCEAN_PHY_ROUGHNESS_seaice_ZOH = 2.0D-3, ; 熱に対する海水面粗度長 [m]
OCEAN_PHY_ROUGHNESS_seaice_ZOE = 2.0D-3, ; 水蒸気に対する海水面粗度長 [m]
/
```

上記の表面アルベドと粗度長を用いて、(OCEAN_SFC_TYPE) で選択したスキームによって大気-海洋間や大気-海氷間の表面フラックスが計算される。この計算では、[PARAM_BULKFLUX] の (BULKFLUX_TYPE) で指定したバルクスキームが用いられる (詳細は第 4.4.6 節を参照)。

海氷過程

(OCEAN_ICE_TYPE) を "SIMPLE" とした場合は、海氷過程が考慮されるようになる。海氷と海洋表層間の熱伝導は、[PARAM_OCEAN_PHY_TC_seaice] で指定したパラメータを用いて計算される。

```
&PARAM_OCEAN_PHY_TC_seaice
OCEAN_PHY_thermalcond_max = 10.D0, ; 深さあたりの熱伝導率の最大値 [J/m2/s/K]
OCEAN_PHY_thermalcond_seaice = 2.D0, ; 海氷の熱伝導率 [J/m/s/K]
/
```

海氷過程のパラメータは [PARAM_OCEAN_PHY_ICE] で設定する。

```
&PARAM_OCEAN_PHY_ICE
OCEAN_PHY_ICE_density = 1000.D0, ; 海氷の密度 [kg/m3]
OCEAN_PHY_ICE_mass_critical = 1600.D0, ; 被覆率が 1 の場合の海氷の質量 [kg/m2]
OCEAN_PHY_ICE_mass_limit = 50000.D0, ; 海氷の質量の最大制限値 [kg/m2]
OCEAN_PHY_ICE_fraction_limit = 1.D0, ; 海氷の被覆率の最大制限値 [1]
/
```

SCALE において海氷の質量は予報変数である。海氷の被覆率は以下の式から診断する。

$$\text{海氷の被覆率} = \sqrt{\frac{\text{海氷の質量}}{\text{OCEAN_PHY_ICE_mass_critical の値}}}$$

4.4.8 陸モデル

海面過程と同様に陸面過程も、陸面の状態量の更新と大気-陸面間のフラックス計算の 2 つに大別される。これらの過程が呼び出される時間間隔はどちらも [PARAM_TIME] で設定する (詳細については第 4.2.7 節を参照)。

4.4.8.1 陸面スキームと表面スキーム

陸モデルでは、陸面温度と土壌温度、土壌水分量といった陸面の状態量の更新を行う。このスキームは、init.conf と run.conf 中の [PARAM_LAND] の (LAND_SFC_TYPE) や (LAND_DYN_TYPE) で設定する。

```
&PARAM_LAND
  LAND_DYN_TYPE = "BUCKET", ; 表 4.4.14 に示す陸の力学から選択
  LAND_SFC_TYPE = "SKIN",   ; 表 4.4.15 に示す陸面の種類から選択
/
```

表 4.4.14: 陸面スキームの選択肢

スキーム名	スキームの説明
NONE or OFF	陸面モデルを使用
BUCKET	熱拡散/バケツモデル
INIT	初期条件に固定

表 4.4.15: 陸モデルの表面スキームの選択肢

スキーム名	スキームの説明
SKIN	スキームにより表面温度を決定する。
FIXED-TEMP	表面温度が最上層の土壌温度と同じであると仮定する。

[PARAM LANDUSE] で入力・設定された土地利用分布に陸面が含まれる場合は、(LAND_TYPE) に NONE または OFF を選択できない。この条件を満たさない場合は、下記のメッセージを LOG ファイルに出力して、プログラムは計算を行わずに直ちに終了する。

```
ERROR [CPL_vars_setup] Land fraction exists, but land component has not been called.
Please check this inconsistency. STOP.
```

(LAND_DYN_TYPE) を "BUCKET" とした場合は、外部データを用いて陸の変数を緩和させることができる (ナッジング)。ナッジングのパラメータは run.conf で指定できる。

```

&PARAM_LAND_DYN_BUCKET
LAND_DYN_BUCKET_nudging = .false.,           ; 陸の変数に対してナudgingを行うか?
LAND_DYN_BUCKET_nudging_tau = 0.0_DP,         ; ナudgingにおける緩和の時定数
LAND_DYN_BUCKET_nudging_tau_unit = "SEC",     ; 緩和の時定数の単位
LAND_DYN_BUCKET_nudging_basename = "",       ; 入力データのベース名
LAND_DYN_BUCKET_nudging_enable_periodic_year = .false., ; 年周期データか?
LAND_DYN_BUCKET_nudging_enable_periodic_month = .false., ; 月周期データか?
LAND_DYN_BUCKET_nudging_enable_periodic_day = .false., ; 日年周期データか?
LAND_DYN_BUCKET_nudging_step_fixed = 0,       ; データの特定のステップ数を用いるか?
LAND_DYN_BUCKET_nudging_offset = 0.0_RP,     ; 変数のオフセット値
LAND_DYN_BUCKET_nudging_defval = UNDEF,      ; 変数のデフォルト値
LAND_DYN_BUCKET_nudging_check_coordinates = .true., ; 変数の座標を確認するか?
LAND_DYN_BUCKET_nudging_step_limit = 0,      ; データを読み込む時間ステップ数の最大値
/

```

(`LAND_DYN_BUCKET_nudging_tau`) が 0 である場合は、海面温度の値は外部ファイルによって完全に置き換わる。(`LAND_DYN_BUCKET_nudging_step_fixed`) が 1 以下であれば、現時刻における値は外部データを時間内挿することで計算される。(`LAND_DYN_BUCKET_nudging_step_fixed`) に特定のステップを指定した場合は、そのステップのデータが時間内挿することなく常に用いられる。(`LAND_DYN_BUCKET_nudging_step_limit`) に 0 よりも大きい値を設定した場合は、その制限を超える時間ステップのデータを読み込まず、最後に読み込んだデータをナudgingに用いる。この制限は、(`LAND_DYN_BUCKET_nudging_step_limit`) が 0 の場合には設定されない。

(`LAND_DYN_TYPE`) に対して "NONE", "OFF" 以外を指定した場合は、粗度長に対するパラメータテーブルや土地利用分布の入力データを必要とする。パラメータテーブルは、ファイル `scale-rm/test/data/land/param.bucket.conf` に用意してある。

大気-陸面間のフラックス

陸面のアルベドや粗度長はパラメータテーブルによって与える。大気-陸間の表面フラックスは、(`LAND_SFC_TYPE`) で選択したスキームによって計算される。この計算では、(`PARAM_BULKFLUX`) の (`BULKFLUX_TYPE`) で指定したバルクスキーム (詳細は第 4.4.6 節を参照) が用いられる。

4.4.9 都市モデル

都市過程は都市の状態量の更新と大気-都市面間のフラックス計算の 2 つに大別される。これらの過程を計算する時間間隔はどちらも (`PARAM_TIME`) で設定する (詳細は第 4.2.7 節を参照)。

4.4.9.1 大気-都市面間フラックス

都市の表面温度や水分量等の都市面の状態量の更新、大気-都市面間のフラックス計算を担う都市スキームは、`init.conf` と `run.conf` 中の (`PARAM_URBAN`) の (`URBAN_DYN_TYPE`) で以下のように設定する。

```
&PARAM_URBAN
  URBAN_DYN_TYPE = "KUSAKA01", ; 表 4.4.16 から都市スキームを選択。
/
```

表 4.4.16: 都市スキームの選択肢

スキーム名	スキームの説明	参考文献
NONE または OFF	都市モデルを利用しない	
LAND	都市域は陸モデルによって計算される	
KUSAKA01	単層キャノピーモデル	Kusaka et al. (2001)

[PARAM LANDUSE] で入力・設定された土地利用分布に都市が含まれる場合は、(URBAN_TYPE) に NONE または OFF を選択できない。この条件を満たさない場合は、下記のメッセージを LOG ファイルに出力して、プログラムは計算を行わずに直ちに終了する。

```
ERROR [CPL_vars_setup] Urban fraction exists, but urban component has not been called.
Please check this inconsistency. STOP.
```

(URBAN_DYN_TYPE) として LAND を選択した場合は、表面フラックスや都市域の土壌変数を計算するために陸モデルが用いられる。この場合、第 4.4.8 節に記述した陸面に対するパラメータテーブルに、都市域に対するパラメータを与える必要がある。

KUSAKA01 スキームには、都市の形態を指定するパラメータが数多く存在する。都市モデルのパラメータは、第 5.3.2 節のネームリストの記述を参照されたい。

4.5 共通の設定

4.5.1 ログファイル

4.5.1.1 ログファイルの出力

scale-rm、scale-rm_init、scale-rm_ppを実行するときに、SCALE-RMはログファイルを出力できる。デフォルト設定において、scale-rmでは”LOG.pe000000”、scale-rm_initでは”init_LOG.pe000000”、scale-rm_ppでは”pp_LOG.pe000000”に、番号ゼロのプロセスからのログメッセージが書き込まれる。ユーザーは以下のように設定ファイルを編集することで、ログファイルの出力設定を変更できる。

```
&PARAM_IO
IO_LOG_BASENAME = 'LOG',           ; ログファイルのベース名
IO_LOG_ALLNODE = .false.,          ; 全てのプロセスに対するログファイルを出力するか?
IO_LOG_SUPPRESS = .false.,         ; .true. であれば、ログの出力を抑制する
IO_LOG_NML_SUPPRESS = .false.,     ; .true. であれば、ネームリストのパラメータの出力を抑制する
IO_NML_FILENAME = '',             ; 指定された場合には、ネームリストのパラメータを指定したファイルに出力する。指定がなければ、ログファイルに出力する。
IO_STEP_TO_STDOUT = -1,            ; 正であれば、時間ステップの情報を標準出力に出力する
/
```

ログファイルの名前は [PARAM_IO] の (IO_LOG_BASENAME) で設定する。上記のデフォルト設定の場合は、マスタープロセスに対するログファイル名は”LOG.pe000000”である。全てのプロセスに対するログファイルを出力するかは、(IO_LOG_ALLNODE) で制御する。(IO_LOG_ALLNODE) を.true.とした場合は、全プロセスに対するログファイルが生成され、そうでない場合はログファイルはマスタープロセス (すなわちランク 0) からのみ出力される。

(IO_LOG_SUPPRESS) を.true. に設定した場合はログファイルは生成されず、ほぼ全てのログメッセージが出力されない。この場合でも、経過時間に関する情報だけは標準出力 (STDOUT) に送られる。

ネームリストのパラメータは、(IO_LOG_NML_SUPPRESS) を.true. に設定した場合を除いて出力される。デフォルトでは、パラメータはログファイルに出力される。(IO_NML_FILENAME) を設定することによって、異なるファイルに出力することもできる。(IO_NML_FILENAME) で指定したファイルは、その後の実行の入力設定ファイルとして用いることができる。

時間ステップの情報はログファイルに出力される。その情報の詳細は次節で説明する。(IO_STEP_TO_STDOUT)> 0と設定されている場合は、時間ステップの情報は標準出力にも出力される。全ての時間ステップの情報はログファイルに出力される。標準出力については、出力するステップ間隔を (IO_STEP_TO_STDOUT) に数値で指定する。

4.5.1.2 ログファイル内の時間に関する情報

`scale-rm` を実行すると、ログファイル中に以下の形式の行を見つけることができる。

```
+++++ TIME: 0000/01/01 00:06:36 + 0.600 STEP: 1984/ 432000 WCLOCK: 2000.2
```

この行は、以下のような計算状況に関するメッセージを意味する。

- 現在、初期時刻「0000/01/01 00:00:00 + 0.000」から 6m36.6s の時間積分が行われたこと。
- 今回の時間ステップは、全時間ステップ数である 432000 回中の 1984 回目であること。
- 経過時間 (cpu 時間) は 2000.2s であること。

さらに、これらの情報から本計算に必要な時間を推測できる。この場合には、推定される所要時間は、121 時間 ($= 2000.2 \times 432000 \div 1984$) である。

ログファイル内のメッセージは以下の形式で出力される。

```
type [subroutine name] message
messages
...
```

type : メッセージの種類 (以下の中の 1 つをとる)

- INFO: ジョブ実行に関する一般的な情報
- WARN: ジョブ実行に関する重要な出来事
- ERROR: 実行停止を伴う致命的なエラー

subroutine name : メッセージを書き込んだサブルーチンの名前

message : メッセージの本文

以下は、エラーメッセージの例である。

```
ERROR [ATMOS_PHY_MP_negative_fixer] large negative is found. rank = 1
k,i,j,value(QHYD,QV) = 17 8 1 1.7347234759768071E-018 0.0000000000000000
k,i,j,value(QHYD,QV) = 19 8 1 -5.4717591620764856E-003 0.0000000000000000
...
```

4.5.2 モニターファイル

モニターファイルと出力変数は、`run.conf` 中の `[PARAM_MONITOR]` と `[MONITOR_ITEM]` で設定する。モニターのデフォルトの形式は、`[PARAM_MONITOR]` で設定する。

```
&PARAM_MONITOR
MONITOR_OUT_BASENAME = "monitor", ; 出力ファイルのベース名
MONITOR_USEDEVATION = .true., ; 最初のステップからの偏差を使うか?
MONITOR_STEP_INTERVAL = 1, ; モニター出力ステップ間隔
/
```

モニターコンポーネントは、乾燥空気の質量・水蒸気・全エネルギー・表面での降水フラックス等の物理量の領域全体での合計値を出力する。これらの出力は、質量収支やエネルギー収支の確認に役立つ。モニターファイルは ASCII 形式であり、ファイル名は (MONITOR_OUT_BASENAME) .pe000000 と設定される。モニター出力の時間間隔は、時間刻み幅 (Δt) の倍数として (MONITOR_STEP_INTERVAL) に指定する。

```
&MONITOR_ITEM
NAME = "ENGT", 変数名。変数のリストは表 4.5.1 に示される。
/
```

例えば、(MONITOR_STEP_INTERVAL) = 10 および (MONITOR_USEDEVATION) = .false. と指定して、[MONITOR_ITEM] に以下の設定を付け加えたとする。

```
&MONITOR_ITEM NAME="ENGK" /
&MONITOR_ITEM NAME="ENGP" /
&MONITOR_ITEM NAME="ENGI" /
&MONITOR_ITEM NAME="ENGT" /
```

このとき、モニターファイルは以下のように出力される。

```
ENGT ENGP ENGK ENGI
STEP= 1 (MAIN) 1.18127707E+17 2.92701438E+16 2.40231436E+13 8.88335403E+16
STEP= 11 (MAIN) 1.18127712E+17 2.92701415E+16 2.40249223E+13 8.88335453E+16
STEP= 21 (MAIN) 1.18127711E+17 2.92701439E+16 2.40223566E+13 8.88335443E+16
STEP= 31 (MAIN) 1.18127710E+17 2.92701454E+16 2.40213480E+13 8.88335435E+16
STEP= 41 (MAIN) 1.18127710E+17 2.92701495E+16 2.40210662E+13 8.88335392E+16
STEP= 51 (MAIN) 1.18127710E+17 2.92701439E+16 2.40205575E+13 8.88335456E+16
STEP= 61 (MAIN) 1.18127711E+17 2.92701565E+16 2.40200252E+13 8.88335340E+16
STEP= 71 (MAIN) 1.18127711E+17 2.92701457E+16 2.40195927E+13 8.88335455E+16
STEP= 81 (MAIN) 1.18127710E+17 2.92701486E+16 2.40193679E+13 8.88335425E+16
STEP= 91 (MAIN) 1.18127710E+17 2.92701573E+16 2.40188095E+13 8.88335342E+16
STEP= 101 (MAIN) 1.18127710E+17 2.92701404E+16 2.40180752E+13 8.88335517E+16
```

4.5.3 物理定数

いくつかの物理定数の値は、設定ファイルの [PARAM_CONST] で変更できる。

```
&PARAM_CONST
CONST_RADIUS = 6.37122D+6,      ; 惑星半径 [m]
CONST_OHM = 7.2920D-5,          ; 惑星の自転角速度 [1/s]
CONST_GRAV = 9.80665D0,         ; 標準重力加速度 [m/s2]
CONST_Rdry = 287.04D0,          ; 気体定数 (乾燥空気) [J/kg/K]
CONST_CPdry = 1004.64D0,        ; 定圧比熱 (乾燥空気) [J/kg/K]
CONST_LAPS = 6.5D-3,            ; 国際標準大気の温度減率 [K/m]
CONST_Pstd = 101325.D0,         ; 標準圧力 [Pa]
CONST_PRE00 = 100000.D0,        ; 圧力の参照値 [Pa]
CONST_Tstd = 288.15D0,          ; 標準温度 (15°C) [K]
CONST_THERMODYN_TYPE = 'EXACT', ; 内部エネルギーの定式化の種類
CONST_SmallPlanetFactor = 1.D0, ; 小さな惑星設定に対するファクタ [1]
/
```

(CONST_THERMODYN_TYPE) が 'EXACT' の場合は、潜熱の温度依存性を考慮する。(CONST_THERMODYN_TYPE) が 'SIMPLE' の場合は、水物質の各カテゴリーの比熱を乾燥空気の定積比熱に設定し、潜熱の温度依存性は無視する。(CONST_RADIUS) には (CONST_SmallPlanetFactor) が掛けられる。同時に、(CONST_OHM) には (CONST_SmallPlanetFactor) の逆数が掛けられる。

4.5.4 暦

暦の種類は、設定ファイルの [PARAM_CALENDAR] で指定できる。デフォルトでは、グレゴリオ暦を用いる。

```
&PARAM_CALENDAR
CALENDAR_360DAYS = .false.      ; 12x30 days の暦を用いるか?
CALENDAR_365DAYS = .false.      ; うるう年を考慮するか?
/
```

暦の設定は太陽天頂角の計算に影響を及ぼし、一年の長さや黄道の一週が一致するように計算される。異なる暦を使用している外部データは、読み込むべきでないことに注意されたい。

(CALENDAR_360DAYS) を .true. とした場合は、1 年が 12 ヶ月、1 ヶ月が 30 日ある暦を設定する。(CALENDAR_365DAYS) を .true. とした場合は、うるう年の無いグレゴリオ暦を用いる。

4.5.5 乱数生成

乱数生成のパラメータは設定ファイルの [PARAM_RANDOM] で設定する。

```
&PARAM_RANDOM
RANDOM_FIX = .false.            ; 乱数のシードを固定するか?
/
```

SCALE では、乱数生成のための組み込み関数を用いる。生成される乱数は擬似乱数であることに注意が必要である。乱数のシードは、現在の日時、cpu 時間、プロセス ID によって決定される。

(RANDOM_FIX) を .true. とした場合は、シードを特定の数字に固定する。このオプションは、初期場としてランダムな擾乱を用いるシミュレーションの結果を再現するときに便利である。

4.5.6 パフォーマンスの測定

パフォーマンス測定のためのパラメータは、設定ファイルの [PARAM_PROF] で与える。

```
&PARAM_PROF
  PROF_rap_level = 2          ; ラップを測定するレベル
  PROF_mpi_barrier = .false.  ; ラップ毎に MPI のバリア命令を追加するか?
/
```

経過時間を測定するために、ソースコードにはユーティリティ関数 (PROF_rapstart, PROF_rapend) が埋め込まれている。これらの測定区間は、詳細なパフォーマンス測定のためにも用いられる。ラップ時間の結果は、ログファイルの終わりに表示される。[PARAM_IO] の (IO_LOG_ALLNODE) を .true. とした場合は、各プロセスの結果が個々のログファイルに報告される。[PARAM_IO] の (IO_LOG_SUPPRESS) が .true. とした場合は、結果は標準出力に送られる。それぞれの測定区間は、出力に関するレベルを持つ。(PROF_rap_level) よりも大きな出力のレベルを持つ区間は、経過時間が測定されない。

(PROF_mpi_barrier) を .true. とした場合は、現在時刻を取得する前後で MPI のバリア命令が呼ばれる。このオプションは計算時間と通信時間を分離するために役立つ。計算時間はしばしばプロセス間の大きな不均衡を明らかにする。

4.5.7 統計量のモニター

統計量モニターに関するパラメータは、設定ファイルの [PARAM_STATISTICS] で与える。

```
&PARAM_STATISTICS
  STATISTICS_checktotal = .false.      ; 変数の合計量を計算し、ログファイルへ出力するか?
  STATISTICS_use_globalcomm = .false.  ; 全通信を用いて全量を計算するか?
/
```

(STATISTICS_checktotal) を .true. とした場合は、デバッグのために、いくつかの変数の領域内の合計量を計算して、ログファイルに出力する。(STATISTICS_use_globalcomm) を .true. とした場合は、全通信を用いて領域全体の合計量が計算される。ただし、これはシミュレーション時間を長くする可能性がある。このオプションを .false. とした場合は、各プロセスに割り当てられた空間領域内での合計量が計算される。

表 4.5.1: モニターに出力可能な変数

Values	Description	Unit
DENS	大気の種類	[kg]
MOMZ	z 方向の運動量	[kg m/s]
MOMX	x 方向の運動量	[kg m/s]
MOMY	y 方向の運動量	[kg m/s]
RHOT	温度	[kg K]
TRACER*	予報変数のトレーサー	[each unit × kg]
QDRY	乾燥空気の質量	[kg]
QTOT	水物質の質量	[kg]
EVAP	表面での蒸発	[kg]
PRCP	降水量	[kg]
ENGT	全エネルギー (ENGP + ENGK + ENGI)	[J]
ENGP	ポテンシャルエネルギー ($\rho * g * z$)	[J]
ENGK	運動エネルギー ($\rho * (W^2 + U^2 + V^2)/2$)	[J]
ENGI	内部エネルギー ($\rho * C_v * T$)	[J]
ENGFLXT	全エネルギーのフラックスの収束 (SH + LH + SFC_RD - TOA_RD)	[J]
ENG SFC_SH	表面での顕熱フラックス	[J]
ENG SFC_LH	表面での潜熱フラックス	[J]
ENG SFC_RD	表面での正味の放射フラックス (SFC_LW_up + SFC_SW_up - SFC_LW_dn - SFC_SW_dn)	[J]
ENG TOA_RD	大気上端での正味の放射フラックス (TOA_LW_up + TOA_SW_up - TOA_LW_dn - TOA_SW_dn)	[J]
ENG SFC_LW_up	表面での上向き長波放射フラックス	[J]
ENG SFC_LW_dn	表面での下向き長波放射フラックス	[J]
ENG SFC_SW_up	表面での上向き短波放射フラックス	[J]
ENG SFC_SW_dn	表面での下向き短波放射フラックス	[J]
ENG TOA_LW_up	大気上端での上向き長波放射フラックス	[J]
ENG TOA_LW_dn	大気上端での下向き長波放射フラックス	[J]
ENG TOA_SW_up	大気上端での上向き短波放射フラックス	[J]
ENG TOA_SW_dn	大気上端での下向き短波放射フラックス	[J]

4.6 後処理

4.6.1 netcdf2grads (net2g)

net2g には 2 つの役割があり、[i] ノード毎に分割されたヒストリファイル (history.***.nc) を結合し、[ii] GrADS で読み込めるバイナリ形式に変換することを行う。net2g は MPI 並列プログラムとしても実行できる。その他、net2g の機能として以下が利用できる。

- モデル面から任意の高度面や圧力面へのデータ内挿
- 3 次元変数に対する鉛直カラム中の平均値・最大値・最小値、鉛直積算値の出力
- 3 次元変数を鉛直層ごとに分割したファイルとして出力
- 時間ステップごとに分割したファイルとして出力

net2g のインストール方法は、第 2.3.1 節を参照されたい。なお、現行の net2g には下記の制約が存在することに注意が必要である。

- net2g に使用する MPI プロセス数は、SCALE-RM 実行時の MPI プロセス数の約数でなければならない。
- `[PARAM_FILE_HISTORY_CARTESC]` の (`FILE_HISTORY_CARTESC_BOUNDARY`) を `.false.` に設定して、SCALE-RM のヒストリファイルを出力する必要がある。
- 2 次元データと 3 次元データは同時に変換できない。
- 変換できるデータは history データのみである。

また、MPI プロセス数を多く取りすぎると、実行速度が落ちることに注意されたい。

MPI 並列を用いる場合には、net2g は以下のように実行する。

```
$ mpirun -n [プロセス数] ./net2g net2g.conf
```

最後の引数の net2g.conf は、net2g に対する設定ファイルである。一方、シングルプロセス版として net2g をコンパイルした場合は、

```
$ ./net2g net2g.conf
```

と実行する。

エラーなく次のメッセージだけが表示されていれば、実行は正常に終了している。

```
+++ MPI COMM: Corrective Finalize
```

次に、2 次元変数や 3 次元変数を変換する場合の設定ファイルの記述方法を説明する。ここでは、`scale-5.3.3/scale-rm/util/netcdf2grads_h/` にあるサンプル設定ファイル `net2g.3d.conf` と

net2g.2d.conf に基づいて説明する。本節では主要な設定項目だけを取り上げることにして、他のオプションについては、scale-5.3.3/scale-rm/util/netcdf2grads.h/にあるファイルREADME.net2g.conf に記載されている説明を参照されたい。

設定ファイル例：3次元変数の変換

```
&LOGOUT
  LOG_BASENAME = "LOG_d01_3d",
  LOG_ALL_OUTPUT = .false.,
/

&INFO
  TIME_STARTDATE = 2000, 1, 1, 0, 0, 0,
  START_TSTEP = 1,
  END_TSTEP = 25,
  DOMAIN_NUM = 1,
  CONFFILE = "../run/run.d01.conf",
  IDIR = "../run",
  Z_LEV_TYPE = "plev",
  MAPPROJ_ctl = .true.
/

&VARI
  VNAME = "PT","U","V","W","QHYD",
  TARGET_ZLEV = 850,500,200,
/
```

上記の例では、ある領域の3次元変数を気圧高度面へ内挿して出力する場合の設定を示している。各設定項目は次のとおりである。

- **[LOGOUT]**（このネームリストは必須ではない）
 - － **(LOG_BASENAME)**：デフォルトのLOG ファイル名「LOG」を変更したいときに指定する。
 - － **(LOG_ALL_OUTPUT)**：0 番以外のプロセスも LOG ファイルに出力させたい場合に、“true”にする。デフォルト値は“false”である。
- **[INFO]**
 - － **(TIME_STARTDATE)**：変換する NetCDF データの最初の日時を指定。
 - － **(START_TSTEP)**：変換する NetCDF データの最初の時間ステップを指定する。最初のいくつかのステップを飛ばしたい場合に適切な値を指定する。デフォルト値は1である。
 - － **(END_TSTEP)**：変換する NetCDF データの最後の時間ステップを指定する。必ず指定すること。
 - － **(DOMAIN_NUM)**：ドメイン番号を指定する。デフォルト値は1である。
 - － **(CONFFILE)**：SCALE-RM 実行時の run.***.conf のパスを指定する（ファイル名を含む）。
 - － **(IDIR)**：SCALE-RM のヒストリファイルのパスを指定する。
 - － **(Z_LEV_TYPE)**：鉛直方向のデータ変換の種類を指定する。“original”はモデル面を表す。“plev”は気圧面、“zlev”は高度面に内挿して出力する。“anal”を指定すると簡易解析を行なった結果を出力する（詳細は後ほど説明）。デフォルト値は“plev”である。

- (MAPPROJ_ctl) : pdef を使った投影図法に対応した「ctl」ファイルを出力するかどうか。現在は、ランベルト図法にのみ対応。

- [VARI]

- (VNAME) : 変換したい変数の名前を指定。デフォルトでは、"PT", "PRES", "U", "V", "W", "QHYD" が指定される。
- (TARGET_ZLEV) : (Z_LEV_TYPE) に応じた変換高度を指定。"plev" の場合の単位は [hPa]、"zlev" の場合の単位は [m] である。また、"original" の場合には格子点番号で指定する。デフォルトでは、14 層 (1000hPa、975hPa、950hPa、925hPa、900hPa、850hPa、800hPa、700hPa、600hPa、500hPa、400hPa、300hPa、250hPa、200 hPa) が指定される。

設定ファイルの変更例：3 次元変数の鉛直積算値を出力

以下では、簡易解析を利用する場合の設定ファイルの例を記述する。他の項目の設定は、前の設定と同じである。

```
&INFO
~ . . . ~
Z_LEV_TYPE = "anal",
ZCOUNT = 1,
/

&ANAL
ANALYSIS = "sum",
/

&VARI
VNAME = "QC", "QI", "QG",
/
```

(Z_LEV_TYPE) を "anal" に指定すると、3 次元変数に対して簡易解析が行われる。この設定によって、[ANAL] の項目を指定できるようになる。出力データは水平 2 次元データであるので、[VARI] の (TARGET_ZLEV) は指定できず、[INFO] の (ZCOUNT) は必ず「1」と指定する。

- [ANAL]

- (ANALYSIS) : 鉛直次元の簡易解析の種類を指定する。"max" と "min" はそれぞれ鉛直カラム中の最大値と最小値を表す。また、"max" は鉛直カラム積算値、"ave" を指定すると鉛直カラム平均値を算出する。デフォルト値は "ave" である。

設定ファイル例：2 次元変数の変換

以下の例は、ある領域の 2 次元変数を変換する場合の設定を示している。[INFO] の (ZCOUNT) に必ず「1」を指定する。

```
&LOGOUT
LOG_BASENAME = "LOG_d01_2d",
/

&INFO
TIME_STARTDATE = 2000, 1, 1, 0, 0, 0,
START_TSTEP = 1,
END_TSTEP = 25,
DOMAIN_NUM = 1,
CONFFILE = "../run/run.d01.conf",
IDIR = "../run",
ZCOUNT = 1,
MAPPROJ_ctl = .true.
/

&VARI
VNAME = "T2","MSLP","PREC"
/
```

設定ファイルの変更例：特殊な時間軸を持つデータの変換

第 4.2.8 節で説明したように、変数の出力間隔は基本的に (FILE_HISTORY_DEFAULT_TINTERVAL) で定義する。しかし、[HISTORY_ITEM] の (TINTERVAL) に (HISTORY_DEFAULT_TINTERVAL) とは異なる値を与えることで、特定の変数の出力間隔を変更することができる。net2g では、異なる出力間隔を持った変数のデータ変換に対応しており、[EXTRA] を設定すればこれを行える。下記の例では、第 4.2.8 節の最後で説明した例（二次元データの "RAIN" のみ 600 秒の間隔で出力）において、追加や修正が必要なネームリストを示している。

ヒストリファイルでは、複数の異なる時間間隔を持ったデータを取り扱うことができる。しかし、net2g では異なる出力時間間隔をもつ変数を同時に変換することはできないため、その場合はこれらの変数に対して別々に net2g を実行する必要がある。

```
&EXTRA
EXTRA_TINTERVAL = 600.0,
EXTRA_TUNIT = "SEC",
/

&VARI
VNAME = "RAIN",
/
```

緯度経度に関する注意点

SCALE-RM において、経度と緯度に対する変数名はそれぞれ「lon」と「lat」である。これらの名前は GrADS で予約語として使われているため、net2g による出力では経度の変数名を「long」、緯度の変数名を「lat」に置き換えている。

大型並列計算機における実行の注意点

スーパーコンピュータ「京」などの大型計算機で計算を行った場合は、出力ファイルの数が多く、各ファイルのサイズも大きい。そのような場合に、手元のディスクの容量がデータを保管するには不十分であったり、後処理に膨大な時間がかかることがある。そのため、SCALE-RM 本体の計算を行ったスーパーコンピュータ上で後処理も行うことを推奨する。「京」において、第 2.2.1 節で説明した環境変数を適切に設定していれば、「make」コマンドだけで net2g をコンパイルできる。

4.6.2 SCALE NetCDF Operator (SNO)

デフォルトでは、SCALE-RM の出力ファイル (SCALE-netCDF ファイルと呼ぶ) は、複数のプロセスにより分割された水平領域に応じて分けられる。この方法は、モデル実行中におけるファイル I/O のスループットの効率が良い。しかし、膨大なファイルを扱うことは困難でもある。ファイル数はプロセス数と共に増加する。また、MPI プロセス数を変えただけでも、次の実験のためにこれらのファイルを用いることはできない。さらに、多くの解析ツールや可視化ツールはこのような分散ファイルには対応していない。これらの問題に対する 1 つの解は出力ファイルの集約であり、Parallel netCDF (PnetCDF) を利用できる (第 5.1.1.5 節を参照)。別の解は、SNO という後処理ツールを使用することである。SNO は以下の特徴を持つ。

- 分割された複数のファイルを結合して、1 個のファイルあるいは複数のファイルにする。
- 1 個のファイルあるいは複数のファイルを、複数のファイルに分割する。
- 単一の NetCDF ファイルを GrADS で読み込むためのコントロールファイル (*.ctl) を作成する。
- GrADS 形式のファイルに変換する。
- 出力データを複数ステップに渡って平均する。
- 測地線 (緯度経度) 格子系へとリグリッドする。

SNO を用いることで、ファイルがハ口の格子を持つかに関わらず、ヒストリデータ、地形データ、境界値データ、初期値/リスタートデータを操作できる。

制限

SNO は、SCALE version 5.3 以降によって作成された SCALE-netCDF ファイルに対して利用できる。それより古い SCALE-netCDF ファイルでは、グローバル属性や軸データの属性に関する情報が不足しているために利用できない。SNO は複数のプロセスを用いて実行できる。しかし、利用できるプロセスの最大数は SNO から出力されるファイルの数に制限される。例えば、数百個の SCALE-netCDF ファイルを 1 個のファイルに変換したい場合には、その処理に使用できる MPI プロセス数は 1 つだけである。GrADS 形式では、異なる鉛直層を持つ複数の変数を含ませることは難しい。そのため、各変数は個別の集約したファイルに出力される。

複数ファイルの再編成は、依然として制限される。各ファイルは (ハ口の格子を除いて) 同じサイズの水平格子を持つ必要がある。設定の例を以下に記述する。

使い方

SNO の実行バイナリは SCALE-RM のメインプログラムと一緒にコンパイルされない。SNO は、ディレクトリ `scale-5.3.3/lib` に置かれる SCALE ライブラリ (`libscale.a`) を用いる。このライブラリは、SCALE-RM のコンパイル時に生成される。そのため、SCALE-RM のコンパイル後に以下のコマンドを実行することを推奨する。

```
$ cd scale-5.3.3/scale-rm/util/sno
$ make
```

コンパイルが成功すれば、実行バイナリファイルがディレクトリ `scale-5.3.3/bin` の下に作成される。SNO の実行例は以下である。

```
$ mpirun -n 2 ./sno sno.conf
```

この例では、「`mpirun`」コマンドを用いて 2 つの MPI プロセスで SNO を実行している。最後の引数は設定ファイルであり、その名前は任意である。

4.6.2.1 設定例: 基本的な使い方

共通設定

SNO は SCALE-RM のいくつかのコンポーネントと一緒に使用しており、以下のネームリストのパラメータを設定できる。

- `[PARAM_IO]`: ログファイル (第 4.5.1 節を参照)
- `[PARAM_PROF]`: パフォーマンス測定 (第 4.5.6 を参照)
- `[PARAM_CONST]`: 物理定数 (第 4.5.3 を参照)
- `[PARAM_CALENDAR]`: カレンダー (第 4.5.4 を参照)

`[PARAM_IO]` に対するオプションを指定しなければ、進捗状況を示すログはマスタープロセスの標準出力に出力される。

複数の SCALE-netCDF ファイルを単一の NetCDF ファイルに変換する場合

```
&PARAM_SNO
  basename_in = 'input/history_d02',
  dirpath_out = 'output',
  basename_out = 'history_d02_new',
  output_gradsctl = .true.,
/
```

この例では、ディレクトリ `./input` にある `history_d02.pe#####.nc` という名前のヒストリファイルを変換する。ここで、##### は MPI のプロセス番号を表す。ファイルの総数や 2 次元トポロジー等の分割ファイルの情報は、1 番目のファイル (この例では `history_d02.pe000000.nc`) から読

み込まれる。変換されたファイルは、history_d02_new.pe#####.nc という名前で ./output ディレクトリの中に出力される。出力ファイル数や変数に関するオプションは、この例では指定されていない。この例では、入力ファイルは単一ファイルに結合され、全ての変数は維持される。

(output_gradsctl) を .true. とした場合は、SNO は GrADS のためのコントロールファイルを出力する。このファイルは、出力ファイルが 1 個の場合にのみ生成される。以下は、コントロールファイルに関する詳細の例である。

```
SET ^history_d02.pe000000.nc
TITLE SCALE-RM data output
DTYPE netcdf
UNDEF -0.99999E+31
XDEF 88 LINEAR 134.12 0.027
YDEF 80 LINEAR 33.76 0.027
ZDEF 35 LEVELS
80.841 248.821 429.882 625.045 835.409 1062.158 1306.565 1570.008 1853.969
2160.047 2489.963 2845.574 3228.882 3642.044 4087.384 4567.409 5084.820 5642.530
6243.676 6891.642 7590.075 8342.904 9154.367 10029.028 10971.815 11988.030 13083.390
14264.060 15536.685 16908.430 18387.010 19980.750 21698.615 23550.275 25546.155
TDEF 25 LINEAR 00:00Z01MAY2010 1HR
PDEF 80 80 LCC 34.65 135.22 40 40 30.00 40.00 135.22 2500.00 2500.00
VARS 3
U=>U 35 t,z,y,x velocity u
PREC=>PREC 0 t,y,x surface precipitation flux
OCEAN_SFC_TEMP=>OCEAN_SFC_TEMP 0 t,y,x ocean surface skin temperature
ENDVARS
```

一般的に、単一の netCDF ファイルは外部メタデータファイルが無くても GrADS で読み込める。しかし、GrADS のインターフェイスは制限的であり、関連する座標や地図投影を含む SCALE-netCDF 形式を解釈できない。そのため、上記のコントロールファイルが必要である。

複数の SCALE-netCDF ファイルを GrADS ファイルに変換する場合

```
&PARAM_SNO
  basename_in = 'input/history_d02',
  dirpath_out = 'output',
  output_grads = .true.,
  vars = "U", "PRCP", "LAND_TEMP",
/
```

(output_grads) を .true. とした場合は、SNO は SCALE-netCDF 形式の代わりに GrADS 形式のファイルを出力する。空間方向に関して全ての分割データは結合される。各変数は個々のファイルに出力される。各ファイルの名前は変数名と同じに設定される。変換されたファイルは、ディレクトリ ./output に出力される。ただし、(basename_out) を指定した場合は、上記で指定した出力先のパスは設定されないことに注意が必要である。コントロールファイルもまた出力される。この例では、(vars) で指定した変数のみが変換される。

複数の SCALE-netCDF ファイルを複数の NetCDF ファイルに変換する場合

```
&PARAM_SNO
  basename_in = 'input/history_d02',
  basename_out = 'output/history_d02',
  nprocs_x_out = 4,
  nprocs_y_out = 6,
/
```

この例では、入力ファイル数は 4 ($[x,y]=[2,2]$) であり、各ファイルには x 方向と y 方向にそれぞれ (ハ口を除いて) 30 個の格子点が含まれる。出力ファイル数は 24 ($[x,y]=[4,6]$) である。上述したように、再分配後の各ファイルは同じ格子点数を持たなければならない。この場合、出力ファイルには x 方向に 15 個、 y 方向に 10 個の格子点を含む。30 \times 2=60 は 7 では割り切れないので、`nprocs_y_out` に 7 を設定することはできない。

SCALE-netCDF ファイルのグローバル属性を確認することで、再分配に必要な情報が得られる。例えば、以下のように「`ncdump`」コマンドを用いればヘッダー情報を確認できる。

```
$ ncdump -h history_d02.pe000000.nc
```

ダンプされた情報の最後に、グローバル属性を見つけられるだろう。

```
.....
// global attributes:
.....
:scale_cartesC_prc_rank_x = 0 ;
:scale_cartesC_prc_rank_y = 0 ;
:scale_cartesC_prc_num_x = 2 ;
:scale_cartesC_prc_num_y = 2 ;
.....
:scale_atmos_grid_cartesC_index_imaxg = 60 ;
:scale_atmos_grid_cartesC_index_jmaxg = 60 ;
.....
```

`scale_cartesC_prc_num_x` と `scale_cartesC_prc_num_y` はそれぞれ、二次元のファイルトポロジーにおける x 方向と y 方向のサイズである。また、`scale_cartesC_prc_rank_x` と `scale_cartesC_prc_rank_y` はそれぞれ、2 次元マップにおける x 方向と y 方向の位置である。このランク番号は 0 から始まる。`scale_atmos_grid_cartesC_index_imaxg` と `scale_atmos_grid_cartesC_index_jmaxg` はそれぞれ、領域全体における x 方向と y 方向の格子数である。これらの格子点数にはハ口格子は考慮しない。したがって、 x 方向や y 方向の分割数にはこれらの格子点数の約数を用いる。

まとめ

ここでは、`[PARAM_SNO]` 中のオプションの詳細を説明する。

```
&PARAM_SNO
  basename_in = "",           ; 入力ファイルのパスやベース名
  dirpath_out = "",          ; 出力先のパス
  basename_out = "",         ; 出力ファイルのベース名
  nprocs_x_out = 1,          ; x 方向の分割数
  nprocs_y_out = 1,          ; y 方向の分割数
  vars = "",                 ; 処理を行う変数の名前
  output_grads = .false.,    ; grads 形式で出力するか?
  output_gradsctl = .false., ; 単一の netCDF ファイルに対する grads のコントロールファイルを
                             ; 出力するか?
  debug = .false.,          ; デバッグのための詳細なログを出力するか?
/
```

(**basename_in**) は必須である。(bdirpath_out) が空であれば、出力先のパスはカレントディレクトリに設定される。(bbasename_out) は SCALE-netCDF ファイルに対して用いられる。(boutput_grads) を .true. にした場合は、出力ファイル名は各変数の名前と同じであり、(bbasename_out) は無視される。

(bnpocs_x_out) や (bnpocs_y_out) のデフォルト値は 1 である。これは複数のファイルが単一のファイルに結合されることを意味する。SNO に与える MPI プロセス数は、出力ファイルの総数 (= (bnpocs_x_out) x (bnpocs_y_out)) と同じか、それ以下でなければならないことに注意が必要である。

(bvars) を指定しなければ、入力ファイル中の全ての変数が処理される。

4.6.2.2 設定例: プラグインされている機能

SNO のいくつかの特徴は、プラグインとして与えられる。ファイルの出力や結合/分割を行う前に、時間平均や水平方向のリマッピング等の演算を適用できる。

月平均する場合

```
&PARAM_SNO
  basename_in = 'input/history_d02',
  basename_out = 'output/history_d02',
  nprocs_x_out = 2,
  nprocs_y_out = 2,
/

&PARAM_SNOPLGIN_TIMEAVE
  SNOPLGIN_timeave_type = 'NUMBER',
  SNOPLGIN_timeave_interval = 4,
/
```

この例では、入力ファイル数も 4 である。したがって、ファイル数は変換によって変わらない。**[PARAM_SNOPLGIN_TIMEAVE]** の (SNOPLGIN_timeave_type) を 'NUMBER' に設定した場合は、データを時間軸方向に平均する。平均の間隔は (SNOPLGIN_timeave_interval) で指定する。この場合は、4 出力ステップごとに変数が平均される。

他の例を以下に示す。

```
&PARAM_SNO
  basename_in = 'input/history_d02',
  basename_out = 'output/history_d02',
/

&PARAM_SNOPLGIN_TIMEAVE
  SNOPLGIN_timeave_type = 'MONTHLY',
/
```

この例では、ファイルの集約と時間平均の両方を行う。`(SNOPLGIN_timeave_type)` として、`'DAILY'`、`'MONTHLY'`、`'ANNUAL'` のいずれかを設定した場合は、対応して変数の日平均、月平均、年平均を SNO は試みる。データの日付や時刻はファイルから読み込まれる。シミュレーションで簡単な暦を用いた場合は、SNO の設定ファイルにも同様の `[PARAM_CALENDAR]` の設定を加える必要がある。

0.5 度間隔の格子にリグリッドする場合

```
&PARAM_SNO
  basename_in = 'input/history_d02',
  basename_out = 'output/history_d02',
/

&PARAM_SNOPLGIN_HGRIDOPE
  SNOPLGIN_hgridope_type = 'LATLON',
  SNOPLGIN_hgridope_lat_start = 30.0,
  SNOPLGIN_hgridope_lat_end = 40.0,
  SNOPLGIN_hgridope_dlat = 0.5,
  SNOPLGIN_hgridope_lon_start = 130.0,
  SNOPLGIN_hgridope_lon_end = 140.0,
  SNOPLGIN_hgridope_dlon = 0.5,
/
```

`[PARAM_SNOPLGIN_HGRIDOPE]` の `(SNOPLGIN_hgridope_type)` を `'LATLON'` に設定した場合は、緯度経度格子系への水平方向のリマッピングが行われる。このプラグインの演算は、出力ファイルが単一である場合にのみ利用できる。`[PARAM_SNOPLGIN_HGRIDOPE]` の他のオプションでは、出力領域の境界や格子点数を設定する。経度方向の格子点数 `nlon` は、以下のように計算される。

$$nlon = \frac{SNOPLGIN_hgridope_lon_end - SNOPLGIN_hgridope_lon_start}{SNOPLGIN_hgridope_dlon}.$$

この計算結果は整数に丸められる。そのため、最も東にある格子点の経度は `(SNOPLGIN_hgridope_lon_end)` よりも小さい可能性がある。緯度方向の格子点数も経度方向と同じ方法で計算する。

緯度-経度の領域は、シミュレーションで用いた領域よりも大きく取ることができる。リマッピングの過程で外挿は許されず、内挿値を持たない格子には欠損値が埋められる。

第5部

詳細説明（応用編）

5.1 SCALE におけるファイル入出力

5.1.1 SCALE-netCDF ファイルとは？

本節では、SCALE が直接読み書きする SCALE-netCDF ファイルについて説明する。SCALE では、データファイルの形式として netCDF(network Common Data Format) を採用している。NetCDF は Unidata (<http://www.unidata.ucar.edu>) が開発を行っているソフトウェアであり、自己記述的で計算機に依存しないデータ形式のファイルを生成することを可能にする。例えば、前者についてはファイル中に軸変数と一緒に変数を記述できる長所がある。また、後者については、どのエンディアンが使用されるかを気にせずにデータを扱える長所がある。SCALE では、上記の利点に基づいてある規約 (SCALE-netCDF convention) を定めている。この規約は、CF convection (<http://cfconventions.org>) におおよそ従っている。

5.1.1.1 グローバル属性

SCALE-netCDF ファイルには、ファイルに含まれるデータの情報 (空間分割に関する情報等) がグローバル属性 (global attribute) として格納されている (表 5.1.1)。
(`PRC_NUM_X`, `PRC_NUM_Y`) については、第 4.2.3 節を参照されたい。

5.1.1.2 ハロ領域データ

ファイルにハロ領域データが含まれるかは、ファイルの種類や設定によって依存する。ただし、ここで定義するハロ領域とは計算領域全体に対するハロであり、各々の局所領域に対するハロを意味しないことに注意されたい。

初期値 (またはリスタート) データおよび境界値データについては、側面境界条件が周期境界でない場合 (`[PARAM_PRC_CARTESC]` において (`PRC_PERIODIC_X`) と (`PRC_PERIODIC_Y`) を `.false.` とした場合)、もしくは単一ファイルの入出力の場合 (`[PARAM_IO]` の (`IO_AGGREGATE`) を `.true.` とした場合) にはハロ領域データが含まれる。それ以外の場合、ハロ領域データは含まれない。

一方で、ヒストリデータについては、側面境界条件が周期的でなく、かつ `[PARAM_HIST]` の (`HIST_BND`) を `.true.` とした場合にはハロ領域データが含まれる。それ以外の場合、ハロ領域データは含まれない。詳細は第 4.2.8 節を参照されたい。

5.1.1.3 軸変数

SCALE-netCDF ファイルには、軸に関するデータが格納されている。全ての軸変数は「long_name」と「units」の属性を持ち、これらはそれぞれ変数の説明や単位を記述する。また、`x`, `y`, `xh`, `yh` には、全領域データ格子数 (`size_global`)、ファイルに含まれるデータの全格子中での開始格子点位置 (`start_global`)、全領域データにおける先頭および末尾におけるハロ領域の格子数 (`halo_global`)、ファイルに含まれるデータのハロ領域の格子数 (`halo_local`) に関する属性が付加されている。

表 5.1.1: SCALE-netCDF ファイルのグローバル属性

名前	説明	備考
title	データの簡潔な説明	[PARAM_FILE_HISTORY] の (FILE_History_TITLE) の値
source	ソフトウェア名	ヒストリファイルに対しては [PARAM_FILE_HISTORY] の (FILE_History_SOURCE) の値。他のファイルに対しては [PARAM_IO] の (H_SOURCE) の値
institution	データ作成者	ヒストリファイルに対しては [PARAM_FILE_HISTORY] の (FILE_History_INSTITUTION) の値。他のファイルに対しては [PARAM_IO] の (H_INSTITUTE) の値
rankid	MPI プロセスのランク番号	モデルのPRC_myrank
Conventions	CF 規約のバージョン	“CF-1.6” for version 5.3
grid_name	格子の種類	SCALE-RM では “cartesC”
scale.cartesC_prc_rank_[xy]	二次元分割のマッピング番号	モデルにおいて変数 PRC_2Drank (PRC_myrank, i) に等しい (x: i=1, y: i=2)
scale.cartesC_prc_num_[xy]	二次元分割数	モデルでは、(PRC_NUM_X), (PRC_NUM_Y)
scale.cartesC_prc_periodic_[zxy]	境界条件が周期的か?	.false.or.true. モデルでは (PRC_PERIODIC_X), (PRC_PERIODIC_Y) に対応
scale.atmos_grid_cartesC_index_[ij]maxg	領域全体の格子点数	モデルにおいて、(IMAX)×(PRC_NUM_X), (JMAX)×(PRC_NUM_Y)
scale.atmos_grid_cartesC_index_kmax	大気モデルの鉛直層数	モデルでは (KMAX)
scale.ocean_grid_cartesC_index_kmax	海洋モデルの鉛直層数	モデルでは (OKMAX)
scale.land_grid_cartesC_index_kmax	陸モデルの鉛直層数	モデルでは (LKMAX)
scale.urban_grid_cartesC_index_kmax	都市モデルの鉛直層数	モデルでは (UKMAX)
scale.atmos_grid_cartesC_index_[kij]halo	ハロの格子数	モデルでは (KHALO), (IHALO), (JHALO)
Calendar	暦の種類	モデルでは (PARAM_CALENDAR)
time_units	時間の単位	
time_start	開始時刻 125	

(History_TITLE, History_SOURCE, History_INSTITUTION) は第 4.2.8 節を参照。

(PRC_NUM_X, PRC_NUM_Y, PRC_PERIODIC_X, PRC_PERIODIC_Y)、(KMAX, IMAX, JMAX) は第 4.2.3 節を参照。

(PARAM_CALENDAR) は第 4.5.4 節を参照。

表 5.1.2 に、軸データのリストを示す。座標変数は自身の次元を持ち、その変数名は次元名と同じである。小文字の名前の変数はファイルの中で主に用いられ、一方で大文字の名前の変数は計算で用いられる軸を表す。図 5.1.1 と図 5.1.2 はそれぞれ、座標変数の水平位置や鉛直位置を示している。同時に、表 5.1.2 も参照されたい。

格子の面積データや体積データもそれぞれ、「cell_area**」と「cell_volume**」としてファイルに格納されている。各変数の「cell_measures」属性には、対応する面積や体積データを指定する。

地図投影データは無次元変数として格納されており、これらの名前は変数の属性「grid_mapping」で指定する。

スタガード格子の位置関係に関する情報は、SGRID 規約 (<https://github.com/sgrid/sgrid>) に基づいて、属性や無次元変数として格納している。その無次元変数の名前は、各変数の「grid」属性で指定される。

ファイルには、地表面高度データ「topo」や陸に対するマスクのデータ「lsmask」も含まれる。

表 5.1.2: SCALE-netCDF に含まれる軸データ。

座標変数	
名前	説明
水平軸 & 時間軸: 共通	
x	ファイルに含まれるデータの x 方向の full level の位置
x_bnds	ファイルに含まれるデータの x 方向の full level のセル境界
xh	ファイルに含まれるデータの x 方向の half level の位置
xh_bnds	ファイルに含まれるデータの x 方向の half level のセル境界
y	ファイルに含まれるデータの y 方向の full level の位置
y_bnds	ファイルに含まれるデータの y 方向の full level のセル境界
yh	ファイルに含まれるデータの y 方向の half level の位置
yh_bnds	ファイルに含まれるデータの y 方向の half level のセル境界
time	時刻の情報
time_bnds	時刻の境界情報
CX	局所領域に対する x 方向の full level の格子位置 (ハ口格子を含む)
FX	局所領域に対する x 方向の half level の格子位置 (ハ口格子を含む)
CDX	x 方向の full level の格子間隔 (ハ口格子を含む)
FDX	x 方向の half level の格子間隔 (ハ口格子を含む)
CY	局所領域に対する y 方向の full level の格子位置 (ハ口格子を含む)
FY	局所領域に対する y 方向の half level の格子位置 (ハ口格子を含む)
CDY	y 方向の full level の格子間隔 (ハ口格子を含む)
FDY	y 方向の half level の格子間隔 (ハ口格子を含む)
CXG	全領域に対する x 方向の full level の格子位置 (ハ口格子を含む)
FXG	全領域に対する x 方向の half level の格子位置 (ハ口格子を含む)
CYG	全領域に対する y 方向の full level の格子位置 (ハ口格子を含む)
FYG	全領域に対する y 方向の half level の格子位置 (ハ口格子を含む)
鉛直軸: 大気	
z	ファイルに含まれる大気データの z 方向の full level の位置
z_bnds	ファイルに含まれる大気データの z 方向の full level のセル境界
zh	ファイルに含まれる大気データの z 方向の half level の位置
zh_bnds	ファイルに含まれる大気データの z 方向の half level のセル境界

前ページからの続き..

CZ	大気モデルの z 方向の full level の格子位置 (ハ口格子を含む)
FZ	大気モデルの z 方向の half level の格子位置 (ハ口格子を含む)
CDZ	大気モデルの z 方向の full level の格子間隔 (ハ口格子を含む)
FDZ	大気モデルの z 方向の half level の格子間隔 (ハ口格子を含む)

鉛直軸：海洋

oz	ファイルに含まれる海洋データの z 方向の full level の位置
oz_bnds	ファイルに含まれる海洋データの z 方向の full level のセル境界
ozh	ファイルに含まれる海洋データの z 方向の half level の位置
ozh_bnds	ファイルに含まれる海洋データの z 方向の half level のセル境界
OCZ	海洋モデルの z 方向の full level の格子位置
OFZ	海洋モデルの z 方向の half level の格子位置
OCDZ	海洋モデルの z 方向の full level の格子間隔

鉛直軸：陸面

lz	ファイルに含まれる陸面データの z 方向の full level の位置
lz_bnds	ファイルに含まれる陸面データの z 方向の full level のセル境界
lzh	ファイルに含まれる陸面データの z 方向の half level の位置
lzh_bnds	ファイルに含まれる陸面データの z 方向の half level のセル境界
LCZ	陸面モデルの z 方向の full level の格子位置
LFZ	陸面モデルの z 方向の half level の格子位置
LCDZ	陸面モデルの z 方向の full level の格子間隔

鉛直軸：都市キャノピー

uz	ファイルに含まれる都市データの z 方向の full level の位置
uz_bnds	ファイルに含まれる都市データの z 方向の full level のセル境界
uzh	ファイルに含まれる都市データの z 方向の half level の位置
uzh_bnds	ファイルに含まれる陸面データの z 方向の half level のセル境界
UCZ	都市モデルの z 方向の full level の格子位置
UFZ	都市モデルの z 方向の half level の格子位置
UCDZ	都市モデルの z 方向の full level の格子間隔

他の軸変数 (1D)

名前	説明
CBFZ	CZ でのバッファ係数
FBFZ	FZ でのバッファ係数
CBFX	局所領域に対する CX でのバッファ係数
FBFX	局所領域に対する FX でのバッファ係数
CBFY	局所領域に対する CY でのバッファ係数
FBFY	局所領域に対する FY でのバッファ係数
CBFXG	全領域に対する CXG でのバッファ係数
FBFXG	全領域に対する FXG でのバッファ係数
CBFYG	全領域に対する CYG でのバッファ係数
FBFYG	全領域に対する FYG でのバッファ係数

他の軸変数 (2D)

名前	説明
----	----

前ページからの続き..

lon	(y, x) での経度
lon_uy	(y, xh) での経度
lon_xv	(yh, x) での経度
lon_uv	(yh, xh) での経度
lat	(y, x) での緯度
lat_uy	(y, xh) での緯度
lat_xv	(yh, x) での緯度
lat_uv	(yh, xh) での緯度
他の軸変数 (3D)	
名前	説明
height	ヒストリファイル中の (z,y,x) or リスタート/初期値ファイル中の (y,x,z) での高度
height_xyw	ヒストリファイル中の (zh,y,x) or リスタート/初期値ファイル中の (y,x,zh) での高度
height_xvz	ヒストリファイル中の (z,yh,x) or リスタート/初期値ファイル中の (yh,x,z) での高度
height_uyz	ヒストリファイル中の (z,y,xh) or リスタート/初期値ファイル中の (y,xh,z) での高度
height_xvw	ヒストリファイル中の (z,yh,x) or リスタート/初期値ファイル中の (yh,x,z) での高度
height_uyw	ヒストリファイル中の (zh,y,xh) or リスタート/初期値ファイル中の (y,xh,zh) での高度
height_uvz	ヒストリファイル中の (z,yh,xh) or リスタート/初期値ファイル中の (yh,xh,z) での高度
height_uvw	ヒストリファイル中の (zh,yh,xh) or リスタート/初期値ファイル中の (yh,xh,zh) での高度

5.1.1.4 データ変数

データ変数は、long_name, units に加えて、未定義値を表す _FillValue や欠損値を表す missing_value を属性として持っている。

初期値 (リスタート) データファイル、境界値データのデータ構造はモデル内の配列構造と同じで、z, x, y の順番である。一方、ヒストリデータファイルは、x, y, z の順番である。

5.1.1.5 単一ファイルの入出力

デフォルトでは、全てのデータファイルは各プロセスごとに出力される。つまり、ファイル I/O は各プロセスで独立である。pnetCDF を使用するように SCALE-RM をコンパイルした場合 (環境変数を SCALE_ENABLE_PNETCDF=T と設定してコンパイルした場合) は、全プロセスからのデータを単一ファイルにまとめることができる (第 2.2.1 節を参照)。これを行うには、`[PARAM_FILE]` の `(FILE_AGGREGATE)` を `.true.` に設定する必要がある。あるいは、ヒストリ、地形、土地利用区分ファイルといった個々のファイルの種類に対して、単一ファイルの入出力を切り替えることもできる。ヒストリファイルについては、`[PARAM_FILE_HISTORY]` の `(FILE_HISTORY_AGGREGATE)` を `.true.` にすれば良い。初期値/リスタートファイルについては、`[PARAM_RESTART]` の `(RESTART_(IN|OUT)_AGGREGATE)`、または `[PARAM_(MODELNAME)_VARS]` の `((MODELNAME)_RESTART_(IN|OUT)_AGGREGATE)` を `.true.` にすれば良い。ただし、(MODELNAME) には「ATMOS」、「OCEAN」、「LAND」、「URBAN」が入る。地形ファイルや土地利用区分ファイルについてはそれぞれ、`[PARAM_TOPO]` の `(TOPO_(IN|OUT)_AGGREGATE)` や `[PARAM_LANDUSE]` の `(LANDUSE_(IN|OUT)_AGGREGATE)` を `.true.` にすれば良い。

表 5.1.3: netCDF の各バージョンの機能性

	複数の 時間間隔	圧縮	netCDF3 ファイルの 読み込み	netCDF4 ファイルの 読み込み	単一ファイルの 読み込み
netCDF3	NG	NG	OK	NG	NG*
netCDF4	OK	OK	OK	OK	NG*
pnetCDF	NG	NG	NG*	NG	OK

(*) 全プロセス数が 1 であれば読み込み可能

5.1.1.6 NetCDF 3 に伴う制限

netCDF version 3 と共に SCALE をコンパイルした場合は、次の制限が存在する。

- 異なる時間間隔の変数は、同じファイルに出力できない。
- データ圧縮が使用できない。
- netCDF4 形式のファイルは読み込めない。

異なる時間間隔で変数を出力したい場合は、それらを異なるファイルに出力するために `[HISTORY_ITEM]` の `(BASENAME)` を設定されたい。

pnetCDF は netCDF3 のファイル形式に基づくので、単一ファイルの入出力機能を用いる場合は複数の時間間隔や圧縮が制限されることに注意が必要である。

Horizontal grid positions for all the components

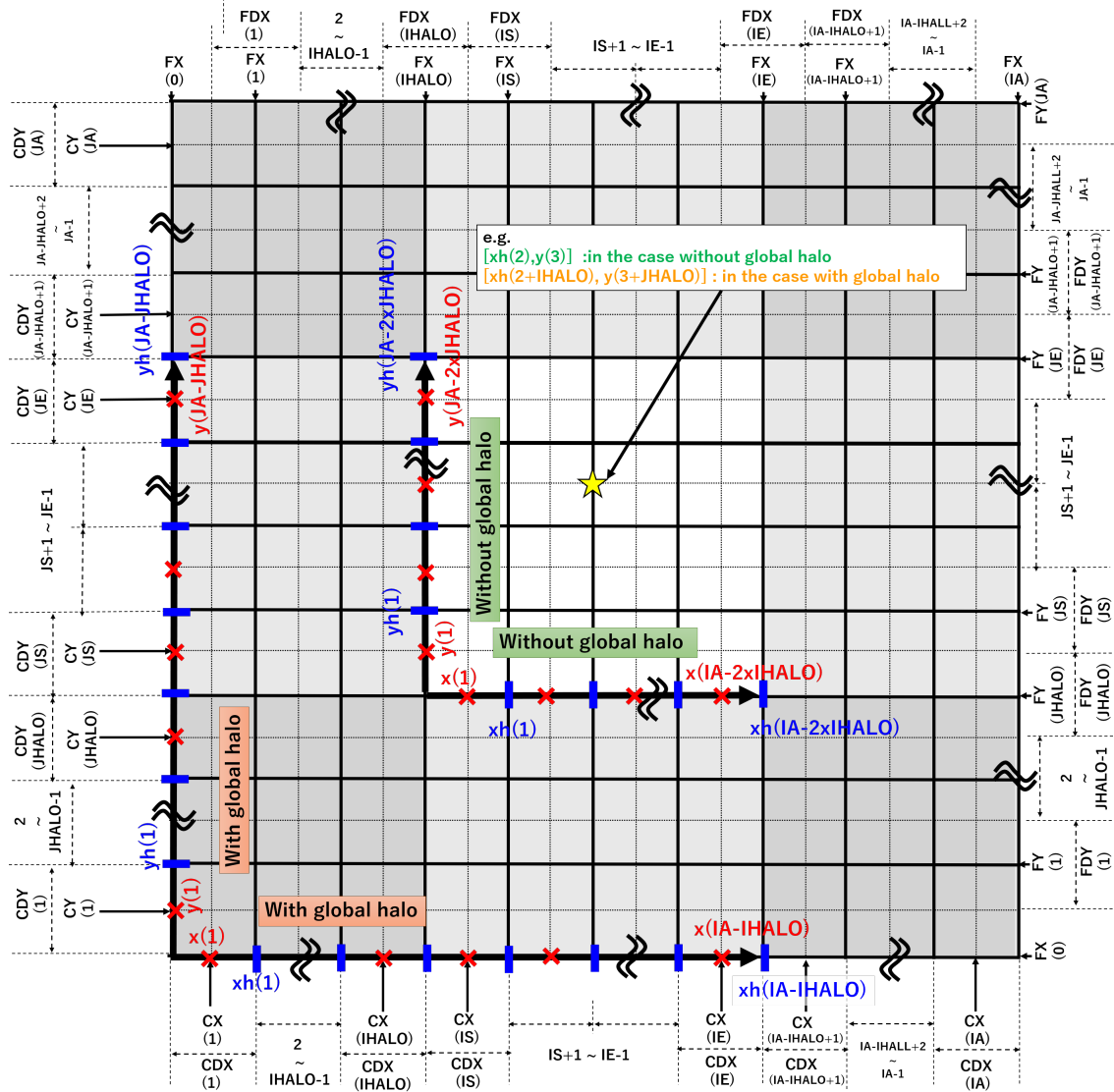


図 5.1.1: SCALE-netCDF ファイルにおける水平座標

Vertical grid positions for each component and physical meanings

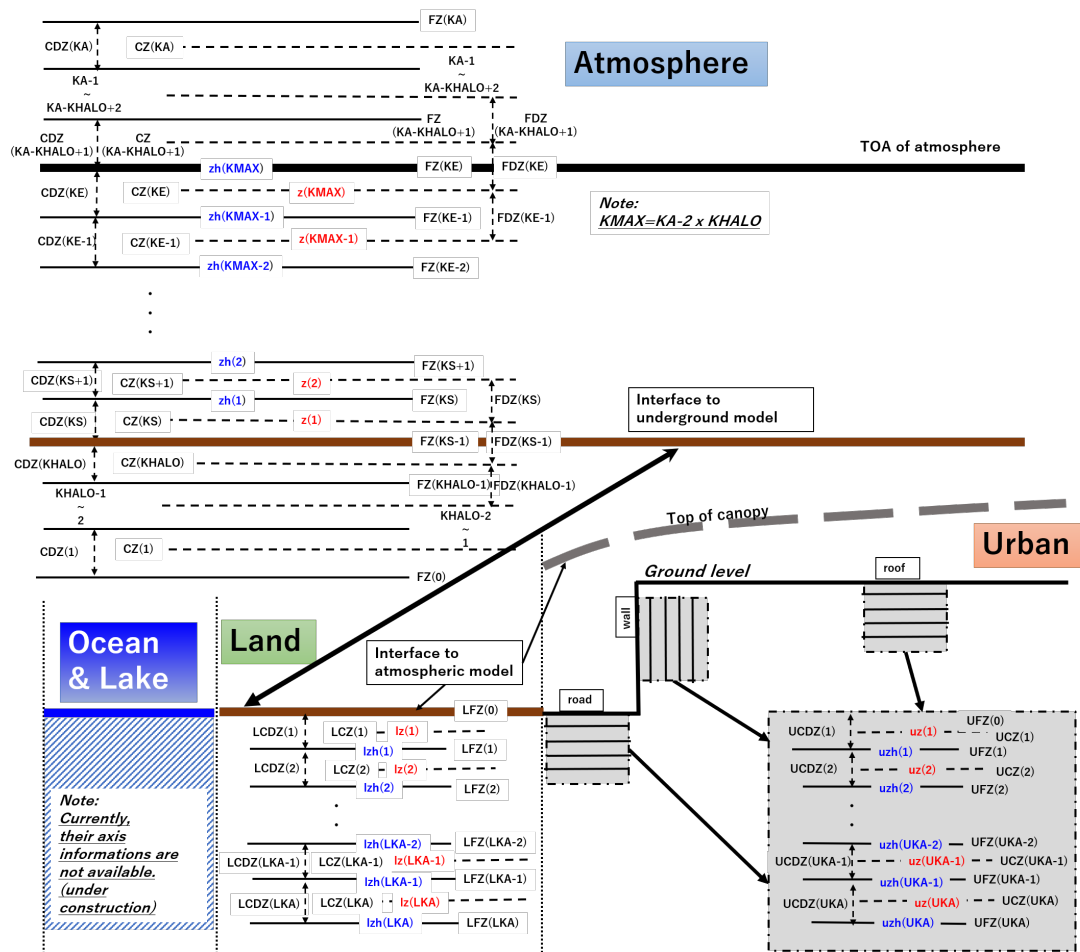


図 5.1.2: SCALE-netCDF ファイルにおける鉛直座標

5.2 ユーザーが作成したプログラムの組み込み方

5.2.1 ユーザーが設定を行うためのモジュール

計算に対するユーザーの要求を満たすために、SCALE-RM は数多くのオプションを用意している。これらはネームリストのパラメータで指定することができる。しかし、期待するオプションが存在しない場合は、ユーザー用のモジュール (mod_user) にプログラムを記述することで、ユーザーが望むようにモデル変数を直接書き換えられる。本節では、mod_user とは何であるかを説明し、その使い方を記述する。

5.2.1.1 mod_user モジュールとは？

デフォルトの mod_user モジュールは、scale-5.3.3/scale/scale-rm/src/user/mod_user.F90 に用意されている。適宜 mod_user.F90 を書き換え、このファイルをデフォルトのファイルの代わりにコンパイルする。

mod_user モジュールには、以下のサブルーチンを含めなければならない。

```
subroutine USER_tracer_setup
subroutine USER_setup
subroutine USER_mkinit
subroutine USER_update
subroutine USER_calc_tendency
```

以下は、SCALE-RM における各プロセスの実行順序である。

初期設定

IO の設定

MPI の設定

格子の設定

力学や物理スキームの管理モジュールの設定

トレーサーの設定

[USER_tracer_setup](#)

地形、陸面の設定

力学や物理スキームの変数やドライバーの設定

[USER_setup](#)

メインルーチン

時間進展

海洋/陸面/都市/大気モデルの更新

[User_update](#)

リスタートファイルの出力

海洋/陸面/都市/大気モデルにおける時間変化率の計算

`USER_calc_tendency`

ヒストリファイルの出力

`mod_user` の各サブルーチンが呼び出されるタイミングを、青色で示している。`USER_mkinit` は、初期値作成プログラム `scale-rm_init` で呼び出される。

`mod_user` のサブルーチンは基本的には各プロセスを処理した後に呼び出されるので、設定や変数を思うように置き換えることができる。また、`USER_tracer_setup` においてパッシブトレーサー等のトレーサーを追加できる。`mod_user.F90` の例として、各テストケース (`scale-5.3.3/scale-rm/test/case` 以下) に含まれる `mod_user.F90` が参考になるだろう。

5.2.1.2 コンパイル

テストケース用の Makefile を活用することで、ユーザが作成した `mod_user.F90` と共に SCALE-RM をコンパイルできる。この手順の例は以下である。

```
$ cd scale-5.3.3/scale-rm/test/case
$ mkdir -p your_dir/exp_name
$ cd your_dir/exp_name
$ cp ../../advection/500m/Makefile .
```

ユーザが作成した `mod_user.F90` を本ディレクトリにコピー

```
$ make
```

5.3 SCALE ライブラリの使い方

5.3.1 SCALE を使用するユーザープログラム

SCALE はサブルーチンの集合体である。これらのサブルーチンは、任意のプログラムで利用できる。ライブラリファイルは、SCALE をコンパイルした後に `scale-5.3.3/lib/` の下に「`scalelib.a`」として作成される。

以下は、ユーザがプログラム中で SCALE を使用するときのテンプレートである。

```
program template
  use scalelib
  implicit none

  call SCALE_init

  ! user instructions

  call SCALE_finalize

  stop
end program template
```

以下は、ファイルから読み込まれた大気の物理量から対流有効位置エネルギー (CAPE) を計算する擬プログラムである。メイン部分の前に、必要なモジュールを引用 (`use`) し、また必要な入力変数を用意しなければならない。以下の例は、プログラムの一部分であることに注意されたい。


```

use scale_const, only: &
  Rdry => CONST_Rdry, Rvap => CONST_Rvap, CPdry => CONST_CPdry
use scale_atmos_hydrometeor, only: &
  CPvap => CP_VAPOR, CL => CP_WATER
use scale_file, only: &
  FILE_open, FILE_read, FILE_close
use scale_atmos_adiabat, only: &
  ATMOS_ADIABAT_setup, ATMOS_ADIABAT_cape
:
real(8) :: z(kmax,imax,jmax), zh(0:kmax,imax,jmax)
real(8) :: temp(kmax,imax,jmax), pres(kmax,imax,jmax), dens(kmax,imax,jmax)
real(8) :: qv(kmax,imax,jmax), qc(kmax,imax,jmax), qdry(kmax,imax,jmax)
real(8) :: rtot(kmax,imax,jmax), cptot(kmax,imax,jmax)
real(8) :: cape(imax,jmax), cin(imax,jmax)
real(8) :: lcl(imax,jmax), lfc(imax,jmax), lnb(imax,jmax)
:
call FILE_open( basename, fid ) ! ファイルを開く
call FILE_read( fid, 'height', z(:,:,:) ) ! full-level での高度データを読み込む
call FILE_read( fid, 'height_xyw', zh(:,:,:) ) ! half-level での高度データを読み込む
call FILE_read( fid, 'T', temp(:,:,:) ) ! 温度データを読み込む
: ! PRES, DENS, QV, QC を読み込む
call FILE_close( fid )

! CAPE を計算するために必要ないくつかの変数を計算する
qdry(:,:,:) = 1.0D0 - qv(:,:,:) - qc(:,:,:) ! 乾燥空気の質量比
rtot(:,:,:) = qdry(:,:,:) * Rdry + qv(:,:,:) * Rvap ! 気体定数 \
cptot(:,:,:) = qdry(:,:,:) * CPdry + qv(:,:,:) * CPvap + ql(:,:,:) * CL ! 熱容量

call ATMOS_ADIABAT_setup
call ATMOS_ADIABAT_cape( kmax, 1, kmax, imax, 1, imax, jmax, 1, jmax, & ! 配列サイズ
  k0, & ! パーセルの持ち上げを開始する鉛直インデックス
  dens(:,:,:), temp(:,:,:), pres(:,:,:), & ! 入力
  qv(:,:,:), qc(:,:,:), qdry(:,:,:), & ! 入力
  rtot(:,:,:), cptot(:,:,:), & ! 入力
  z(:,:,:), zh(:,:,:), & ! 入力
  cape(:,:), cin(:,:), lcl(:,:), lfc(:,:), lnb(:,:) ) ! 出力

```

リファレンスマニュアル (第 5.3.2 節を参照) では、利用できるサブルーチンの一覧やそれらのサブルーチンの詳細を確認できる。ディレクトリ `scale-5.3.3/scalelib/test/analysis` に、SCALE-RM が出力したヒストリファイルを解析するサンプルプログラムを用意してあるので、必要に応じて参照されたい。

5.3.1.1 コンパイル

SCALE を用いたプログラムをコンパイルする前に、SCALE をコンパイルする必要がある。

```

$ cd scale-5.3.3/scalelib/src
$ make

```

ユーザーが作成したプログラムをコンパイルするときに、`scale-5.3.3/lib` に置かれている `libscale.a` をリンクする必要がある。また、モジュールファイルのパスをコンパイラに伝えなければならない。モジュールファイルのパスは `scale-5.3.3/include` であり、このパスを指定するオプションはコンパイラに依存する。オプションは、`sysdep` ディレクトリ下のファイル内で指定される変数 `MODDIROPT`

の値を見れば分かる (第 2.2.1 節を参照)。

```
$ ${FC} your-program ${MODDIROPT} scale-top-dir/include \\  
    'nc-config --cflags' -lscale-top-dir/lib -lscale 'nc-config --libs'
```

ユーザーが作成したプログラムをコンパイルするために、サンプルにある Makefile を以下のように利用することもできる。

```
$ cd scale-5.3.3/scalelib/test/analysis  
$ mkdir your_dir  
$ cd your_dir  
$ cp ../horizontal_mean/Makefile .
```

プログラムファイルを本ディレクトリにコピーする。

Makefile を編集する (BINNAME = your_program_name)。

```
$ make
```

5.3.2 リファレンスマニュアル

SCALE のサブルーチンに対するリファレンスマニュアルは、<https://scale.aics.riken.jp/doc/5.3.3/index.html> で公開している。このリファレンスマニュアルは、doxygen (<http://www.doxygen.org/>) によって生成されている。

リファレンスには、次の情報が含まれる。

- サブルーチン
- ネームリストのパラメータ
- ヒストリ変数

5.3.2.1 サブルーチン

サブルーチンの説明、引数、コールグラフがサブルーチンの情報として含まれる。サブルーチンのソースコードも見ることができる。ユーザーは、トップページやトップメニューにリンクされている「Module List」あるいは「File List」からサブルーチンを探し出すことができる。モジュールのリストには、各モジュールに対する簡単な説明が書かれている。

モジュール名の接頭子は、SCALE については「scale_」、SCALE-RM については「mod_」である。ファイル名は、「.F90」の接尾子を付けたモジュール名である。サブルーチン名は、接頭子を除いたモジュール名と関数を説明する名前からなる。例えば、ATMOS_ADIABAT_cape というサブルーチンは、ファイル scale_atmos_adiabat.F90 中のモジュール scale_atmos_adiabat 内に含まれる。

```
scale_atmos_adiabat.F90  
  
module scale_atmos_adiabat  
... ..  
contains  
!-----  
    subroutine atmos_adiabat_cape( &  
        Kstr, &  
        DENS, &  
        ...
```

5.3.2.2 ネームリストのパラメータ

ネームリストのパラメータのリストは、リファレンスマニュアルのトップページにあるリンク先、もしくは直接 <https://scale.aics.riken.jp/doc/5.3.3/namelist.html> に行くと見ることができる。リストには、パラメータ名、ネームリストのグループ名、変数が定義されているモジュールの名前が含まれる。パラメータは変数名で並び替えられている。パラメータの詳細は、ネームリストのグループ名またはモジュール名をクリックすれば確認できる。

5.3.2.3 ヒストリ変数

ヒストリ変数のリストは、リファレンスマニュアルのトップページにあるリンク先、または直接 <https://scale.aics.riken.jp/doc/5.3.3/history.html> に行くと見ることができる。リストには、変数名、簡単な説明、ヒストリデータのために変数が登録されているモジュールの名前が含まれる。ヒストリ変数はモジュール名で並び替えられている。変数の詳細情報は、モジュール名をクリックすれば確認できる。

5.4 バルクジョブの実行方法

5.4.1 バルクジョブとは？

SCALE-RM には、独立した実験を同時に複数実行できる「一括実行機能」、いわゆるバルクジョブ機能が備わっている。この機能は、パラメタスイープ実験、初期値アンサンブル実験、タイムスライス気候実験等を行うのに便利である。

バルクジョブ機能は、モデル本体 (scale-rm) の実行はもちろん、地形・土地利用データ、初期値/境界値データの作成にも適用できる。ただし、バルクジョブ機能による地形・土地利用データの作成は、地形コピー機能 (第 4.2.10.1 節を参照) を利用しない場合に限る。

以下の説明では、1 のバルクジョブに含まれる独立した実行命令を「サブジョブ」と呼ぶこととする。ここでは、3 つの 2 段オンライン・ネスティング実験を例に説明する。この 3 つの実験は、積分期間もしくは計算領域中心が異なる 3 つのサブジョブを想定している。ファイル launch.conf 中の [PARAM_LAUNCHER] の (NUM_DOMAIN, PRC_DOMAINS, CONF_FILES) (第 4.2.10.3 節参照) は、全ての設定で同じにする必要がある。その他の設定 (積分時間、使用するスキーム、1 つの MPI プロセスあたりの格子数等) は、サブジョブ間で異なっても構わない。

5.4.2 バルクジョブの設定

バルクジョブ機能は、オンライン・ネスティングで利用した MPI プロセスを分割・分配する機能を拡張したものである。従って、ジョブの起動のために launch.conf ファイルが必要になる。オンライン・ネスティングとバルクジョブ機能を併用する場合も、launch.conf ファイルは 1 個だけ用意すれば良い。そのような場合の例を、以下に示す。

&PARAM_LAUNCHER	
NUM_BULKJOB = 3,	サブジョブの数
NUM_DOMAIN = 2,	ネスティング領域の数
PRC_DOMAINS = 9, 36,	各領域の全プロセス数
CONF_FILES = run.d01.conf, run.d02.conf,	設定ファイル名
LOG_SPLIT = .false.,	MPI 分割に関するログを出力するか?
COLOR_REORDER = .true.,	MPI 分割におけるプロセス番号の再割り当てを行うか?
FAILURE_PRC_MANAGE = .false.,	失敗したプロセスの管理機能を使用するか?
NUM_FAIL_TOLERANCE = 1,	失敗プロセスの許容数
FREQ_FAIL_CHECK = 5,	DT ごとの FPM による調査頻度
/	

launch.conf に、(NUM_BULKJOB) の項目を付け加えれば十分である。他の設定は、第 4.2.10.3 節における設定と同様である。シングルドメイン実験 (ネスティングを使用しない) の場合は、(NUM_DOMAIN = 1) と指定し、(CONF_FILES) に設定ファイルを 1 つ指定すればよい。

(LOG_SPLIT) を .true. にした場合は、MPI のコミュニケータの分割のログが出力される。(LOG_SPLIT) のデフォルト値は .false. である。

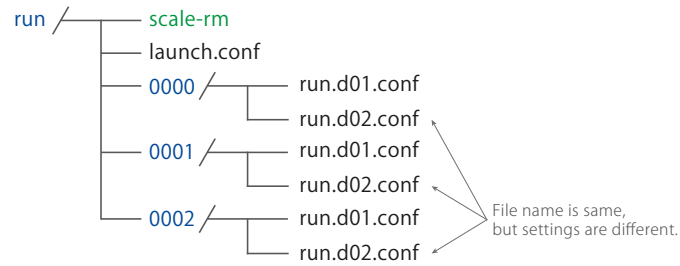


図 5.4.1: バルクジョブ機能を使って scale-rm を実行する場合のディレクトリ構造。「0000」や「0001」といった数字は、ジョブ番号に対応するディレクトリ (ジョブディレクトリと呼ぶ) の名前である。各ジョブディレクトリの中には、サブジョブの実行に必要な全設定ファイルが用意されていないなければならない。

(COLOR_REORDER) というオプションは、ジョブのプロセス数に応じて MPI のコミュニケータのグループ内のジョブの再配置を行うためのスイッチである。最も大きなプロセス数を持つジョブは最前列に配置される。効率的であるノード内通信が用いられる点で、この方法は合理的である。

5.4.3 失敗プロセスの管理

SCALE-RM のバルクジョブシステムでは、失敗プロセスの管理 (failure process management ; FPM) ツールが使用できる。FPM は、ある調査周期でジョブグループを監視する。いくつかのジョブが不幸にも異常終了した場合でも、失敗したジョブ数が制限値に達するまでは他のジョブを終了させない。(FAILURE_PRC_MANAGE) は FPM ツールを用いるためのスイッチである。(NUM_FAIL_TOLERANCE) と (FREQ_FAIL_CHECK) はそれぞれ、失敗したジョブの制限値と失敗したジョブを調べる間隔を設定するパラメータである。FPM ツールの現版では、単一領域に対してのみ使用でき、オンライン・ネスティング計算での完全なシステムには対応していない。オンライン・ネスティング計算においても FPM ツールを使用したい場合は、(NUM_FAIL_TOLERANCE) を全ジョブ数と同じにしなければならない。

5.4.4 バルクジョブのための準備

バルクジョブの実行にあたり、サブジョブの数だけディレクトリ (ジョブディレクトリと呼ぶ) を用意する必要がある。図 5.4.1 において、ジョブディレクトリは 0000/ 0001/ 0002/ に対応する。ディレクトリ名には、ゼロから始まる 4 桁の数字が付けられる。各ジョブディレクトリには、実験に必要な全てのファイル (設定ファイル、入力ファイル、出力用ディレクトリ等) を用意しなければならない。設定ファイルに指定されているディレクトリやファイルのパスが、以下で説明するように適切に設定されているか注意する必要がある。以下は、ジョブ 0000 の run.d01.conf の抜粋である。

```
&PARAM_IO
  IO_LOG_BASENAME = "0000/LOG_d01",
/

&PARAM_RESTART
  RESTART_OUTPUT = .true.,
  RESTART_OUT_BASENAME = "0000/restart_d01",
  RESTART_IN_BASENAME = "../init/0000/init_d01_00013046400.000",
/

&PARAM_TOPO
  TOPO_IN_BASENAME = "../pp/0000/topo_d01",
/

&PARAM_LANDUSE
  LANDUSE_IN_BASENAME = "../pp/0000/landuse_d01",
/

&PARAM_ATMOS_BOUNDARY
  ~ ... ~
  ATMOS_BOUNDARY_IN_BASENAME = "../init/0000/boundary_d01",
  ~ ... ~
/

&PARAM_FILE_HISTORY
  FILE_HISTORY_DEFAULT_BASENAME = "0000/history_d01",
  ~ ... ~
/
```

図 5.4.1 に示すように、ジョブディレクトリは実行バイナリと同じディレクトリの階層にある。つまり、設定ファイルは各ジョブディレクトリの下にあるが、入力ファイルや出力先のディレクトリは、実行バイナリの位置から見た相対パスを記述する必要がある。従って、ジョブ 0000 番の実験に対する出力用ディレクトリは 0000/であり、出力ファイル名は 0000/**となる。ジョブディレクトリ名を付け忘れてファイル名を全実験で同じにしてしまうと、同じファイルに出力を行うためデータが消失することに注意されたい。

5.4.5 バルクジョブの実行

バルクジョブの実行時には、以下のように MPI プロセスの総数を指定する。

```
$ mpirun -n 135 ./scale-rm launch.conf
```

この例では、1 サブジョブあたりが使用するプロセス数は 45 (= 9 + 36) であり、3 つのジョブで使用するプロセスの総数は 135 である。MPI のプロセス分割に関する情報を与えるメッセージは、LOG ファイルの中の SCALE のロゴの後に書き込まれる。下記は、ドメイン 1 のプロセス 0 からのログの出力例である。

```

+++++ start MPI
*** UNIVERSAL_COMM_WORLD : 0           ; 実行環境によって値が異なる
*** total process [UNIVERSAL] : 135
*** my process ID [UNIVERSAL] : 36
*** master rank? [UNIVERSAL] : F
*** GLOBAL_COMM_WORLD : 3             ; 実行環境によって値が異なる
*** total process [GLOBAL] : 45
*** my process ID [GLOBAL] : 36
*** master rank? [GLOBAL] : F
*** LOCAL_COMM_WORLD : 4              ; 実行環境によって値が異なる
*** total process [LOCAL] : 9
*** my process ID [LOCAL] : 0
*** master rank? [LOCAL] : T
*** ABORT_COMM_WORLD : 0
*** master rank ID [each world] : 0

```

[LOCAL] と表記されている項目は、ドメイン内のプロセスグループに関する情報である。また、[GLOBAL] と表記されている項目はネスティンググループ、[UNIVERSAL] と表記されている項目はジョブグループに関する情報である。LOCAL グループは GLOBAL グループに包含され、さらに GLOBAL グループは UNIVERSAL グループに包含される。total process は各グループ内の全プロセス数、my process ID はあるグループで見た時のプロセス番号を表す。

この例では、total process [UNIVERSAL] は 135 であるので、全体で 135 のプロセスが起動したことが確認できる。また、total process [GLOBAL] は 45 であるので、1 サブジョブあたり 45 プロセスを使用したことが分かる。この例ではドメイン 1 に対する LOG メッセージであるため、total process [LOCAL] が 9 と表記されていることは正しい。もしドメイン 2 の LOG メッセージを確認した場合、これは 36 である。LOG ファイルやヒストリファイルの番号に対応するプロセス番号は、my process ID [UNIVERSAL] である。異常終了時にも、この表記法に従ってメッセージが出力される。そのため、この表記法を理解していれば、大量のサブジョブを実行している時にどのプロセスでエラーが発生したか即座に判断できる。

関連図書

- A. C. M. Beljaars and A. A. M. Holtslag. Flux parameterization over land surfaces for atmospheric models. *J. Appl. Meteor.*, 30:327–341, 1991.
- A. R. Brown, S. H. Derbyshire, and P. J. Mason. Large-eddy simulation of stable atmospheric boundary layers with a revised stochastic subgrid model. *Quarterly Journal of the Royal Meteorological Society*, 120:1485–1512, 1994.
- Committee on Space Research; NASA National Space Science Data Center. Cospar international reference atmosphere (cira-86): Global climatology of atmospheric parameters, 2006.
- J. W. Deardorff. Stratocumulus-capped mixed layers derived from a three-dimensional model. *Boundary-Layer Meteorology*, 18:495–527, 1980.
- A. M. Golovin. The solution of the coagulation equation for cloud droplets in a rising air current. *Izv. Geophys. Ser.*, 5:482–487, 1963.
- J. S. Kain. The kain-fritsch convective parameterization: An update. *J. Appl. Meteor.*, 43:170–181, 2004.
- J. S. Kain and J. M. Fritsch. A one-dimensional entraining/detraining plume model and its application in convective parameterization. *Journal of the Atmospheric Sciences*, 47:2784–2802, 1990.
- E. Kessler. On the distribution and continuity of water substance in atmospheric circulation. *Meteorological Monograph*, 10:1–84, 1969.
- Barry Koren. *A robust upwind discretization method for advection, diffusion and source terms*. Centrum voor Wiskunde en Informatica Amsterdam, 1993.
- H. Kusaka, H. Kondo, Y. Kikegawa, and F. Kimura. A simple single-layer urban canopy model for atmospheric models: comparison with multi-layer and slab models. *Boundary-Layer Meteorol.*, 101:329–358, 2001.
- D. K. Lilly. On the numerical simulation of buoyant convection. *Tellus*, 14:148–171, 1962.
- A. Long. Solutions to the droplet collection equation for polynomial kernels. *Journal of the Atmospheric Sciences*, 31:1041–1052, 1974.
- J.-F. Louis. A parametric model of vertical eddy fluxes in the atmosphere. *Boundary-Layer Meteorol.*, 17:187–202, 1979. doi: 10.1007/BF00117978.
- G. L. Mellor and T. Yamada. Development of a turbulence closure model for geophysical fluid problems. *Rev. Geophys. Space Phys.*, 20:851–875, 1982.

- M. J. Miller, A. C. M. Beljaars, and T. N. Palmer. The sensitivity of the ecmwf model to the parameterization of evaporation from the tropical oceans. *Journal of Climate*, 5(5):418–434, 1992.
- I.-J. Moon, I. Ginis, T. Hara, and B. Thomas. A physics-based parameterization of air-sea momentum flux at high wind speeds and its impact on hurricane intensity predictions. *Mon. Weather Rev.*, 135:2869–2878, 2007. doi: 10.1175/MWR3432.1.
- T. Nakajima, M. Tsukamoto, Y. Tsushima, A. Numaguti, and T. Kimura. Modeling of the radiative process in an atmospheric general circulation model. *Applied Optics*, 39:4869–4878, 2000. doi: 10.1364/AO.39.004869.
- M. Nakanishi and H. Niino. An improved mellor-yamada level-3 model with condensation physics: Its design and verification. *Bound-Lay. Meteorol.*, 112:1–31, 2004.
- M. Narita and S. Ohmori. Improving precipitation forecasts by the operational nonhydrostatic mesoscale model with the kain-fritsch convective parameterization and cloud microphysics. *12th Conference on Mesoscale Processes*, 6-9 August 2007, Waterville Valley conference & event center:available from https://ams.confex.com/ams/12meso/techprogram/paper_126017.htm, 2007.
- S. Nishizawa, H. Yashiro, Y. Sato, Y. Miyamoto, and H. Tomita. Influence of grid aspect ratio on planetary boundary layer turbulence in large-eddy simulations. *Geosci. Model Dev.*, 8(10): 3393–3419, 2015. doi: 10.5194/gmd-8-3393-2015.
- Y. Ogura and H.-R. Cho. Diagnostic determination of cumulus cloud populations from observed large-scale variables. *Journal of the atmospheric sciences*, 30(15):1276–1286, 1973.
- Katsuyuki V Ooyama. A dynamic and thermodynamic foundation for modeling the moist atmosphere with parameterized microphysics. *Journal of the atmospheric sciences*, 58(15):2073–2102, 2001.
- J-L Redelsperger, PRA Brown, F Guichard, C How, M Kawasima, S Lang, Th Montmerle, K Nakamura, K Saito, C Seman, et al. A gcss model intercomparison for a tropical squall line observed during toga-coare. i: Cloud-resolving models. *Quarterly Journal of the Royal Meteorological Society*, 126(564):823–863, 2000.
- J. J. Remedios, R. J. Leigh, A. M. Waterfall, D. P. Moore, H. Sembhi, I. Parkes, J. Greenhough, M. P. Chipperfield, and D. Hauglustaine. Mipas reference atmospheres and comparisons to v4.61/v4.62 mipas level 2 geophysical data sets. *Atmos. Chem. Phys. Discuss.*, 7:9973–10017, 2007. doi: doi:10.5194/acpd-7-9973-2007.
- Y. Sato, S. Nishizawa, H. Yashiro, Y. Miyamoto, Y. Kajikawa, and H. Tomita. Impacts of cloud microphysics on trade wind cumulus: which cloud microphysics processes contribute to the diversity in a large eddy simulation? *Progress in Earth and Planetary Science*, 2(1):1–16, 2015. doi: 10.1186/s40645-015-0053-6.
- A. Scotti, C. Meneveau, and D. K. Lilly. Generalized smagorinsky model for anisotropic grids. *Physics of Fluids A*, 5:2306–2308, 1993.

- T. Seiki and T. Nakajima. Aerosol effects of the condensation process on a convective cloud simulation. *Journal of the Atmospheric Sciences*, 71:833–853, 2014.
- M. Sekiguchi and T. Nakajima. A k-distribution-based radiation code and its computational optimization for an atmospheric general circulation model. *J. of Quantitative Spectroscopy and Radiative Transfer*, 109:2779–2793, 2008.
- J. Smagorinsky. General circulation experiments with the primitive equations. *Mon. Weather Rev.*, 91:99–164, 1963.
- K. Suzuki, T. Nakajima, T. Y. Nakajima, and A. P. Khain. A study of microphysical mechanisms for correlation patterns between droplet radius and optical thickness of warm clouds with a spectral bin microphysics cloud model. *Journal of the Atmospheric Sciences*, 67(4):1126–1141, 2010.
- Team SCALE. *The detail formulation of SCALE-RM*. RIKEN/AICS, 2015.
- H. Tomita. New microphysical schemes with five and six categories by diagnostic generation of cloud ice (special issue; the international workshop on high-resolution and cloud modeling, 2006). *Journal of the Meteorological Society of Japan. Ser. II*, 86:121–142, 2008.
- I Uno, X-M Cai, DG Steyn, and S Emori. A simple extension of the louis method for rough surface layer modelling. *Boundary-Layer Meteorology*, 76(4):395–409, 1995.
- Louis J Wicker and William C Skamarock. Time-splitting methods for elastic models using forward time schemes. *Monthly Weather Review*, 130(8):2088–2097, 2002.
- D. K. Wilson. An alternative function for the wind and temperature gradients in unstable surface layers. *Bound-Lay. Meteorol.*, 99:151–158, 2001.
- S. T. Zalesak. Fully multidimensional flux-corrected transport algorithms for fluids. *J. Comput. Phys.*, 31:335–362, 1979. doi: 10.1016/0021-9991(79)90051-2.

付 録 A よくある質問とその回答：FAQ

ここでは、よくある質問を具体的な練習問題として列挙する。その後、それぞれの問題に対する回答を示す。

練習問題

1. 計算領域を変えず、MPI 並列数を変更したい
第 3.2 章の現実大気実験のチュートリアルにおいて、4-MPI 並列の設定を 6-MPI 並列に変更する。(参考：4.2.3.1 節、4.2.3.2 節)
2. MPI 並列数を変えず、計算領域を変更したい
第 3.2 章の現実大気実験のチュートリアルにおいて、MPI 並列数を変更せずに、計算領域を元のサイズに比べて、 x 方向に $4/3$ 倍に拡大、 y 方向は $2/3$ 倍に縮小する。(参考：4.2.3.2 節)
3. 計算領域を変えず、水平格子間隔を変更したい
第 3.2 章の現実大気実験のチュートリアルにおいて、計算領域を変えず、水平格子間隔をデフォルト値から 5 km に変更する。(参考：4.2.3.2 節、4.2.3.3 節、4.2.3.5 節、4.2.7 節)
4. 計算領域の位置を変更したい
第 3.2 章の現実大気実験のチュートリアルにおいて、計算領域の大きさは変えず、中心位置をデフォルト値から経度 139 度 45.4 分、緯度 35 度 41.3 分に変更する。(参考：4.2.4 節)
5. 積分時間を変更したい
第 3.2 章の現実大気実験のチュートリアルにおいて、積分時間を 6 時間から 12 時間に変更する。(参考：4.2.7 節)
6. 出力変数の追加と出力間隔の変更をしたい
第 3.2 章の現実大気実験のチュートリアルにおいて、出力の時間間隔をデフォルト値から 30 分に変更し、地表面での下向き短波放射と上向き短波放射の出力変数に追加する。(参考：4.2.8 節、5.3.2 節)
7. リスタート計算をしたい
第 3.2 章の現実大気実験のチュートリアルにおいて、まず 3 時間の積分を行う。続いて、最初の積分時に作成されたリスタートファイルを使用してさらに 3 時間の積分を行う。(参考：4.2.9 節、4.1.1 節)

回答

1. 計算領域は変えず、MPI 並列数を変更したい

[PARAM_ATMOS_GRID_CARTESC_INDEX] 内の (IMAX, JMAX)、[PARAM_PRC] 内の (PRC_NUM_X, PRC_NUM_Y) を変更する。下記 3 つの式を満たしていれば正解である。

$$MPI \text{ 並列数} = (PRC_NUM_X) \times (PRC_NUM_Y) = 6$$

$$X \text{ 方向の格子数} = (IMAX \times PRC_NUM_X) = 90$$

$$Y \text{ 方向の格子数} = (JMAX \times PRC_NUM_Y) = 90$$

2. MPI 並列数は変えず、計算領域を変更したい

MPI プロセスあたりの格子数を n 倍にすれば、領域全体の格子数も n 倍となる。したがって、[PARAM_ATMOS_GRID_CARTESC_INDEX] 内の (IMAX, JMAX) のみ変更すれば良い。赤字の部分がデフォルトからの変更点である。

```
&PARAM_ATMOS_GRID_CARTESC_INDEX
KMAX = 36,
IMAX = 60,           (オリジナル設定は IMAX = 45)
JMAX = 30,           (オリジナル設定は JMAX = 45)
/
```

3. 計算領域は変えず、水平格子間隔を変更したい

MPI 並列数を変えない場合は、[PARAM_ATMOS_GRID_CARTESC] の (DX, DY) と、[PARAM_ATMOS_GRID_CARTESC_INDEX] 内の (IMAX, JMAX) を変更する。

```
&PARAM_PRC_CARTESC
PRC_NUM_X = 2,
PRC_NUM_Y = 2,

&PARAM_ATMOS_GRID_CARTESC_INDEX
KMAX = 36,
IMAX = 180,          (オリジナル設定は IMAX = 45)
JMAX = 180,          (オリジナル設定は JMAX = 45)
/

&PARAM_ATMOS_GRID_CARTESC
DX = 5000.DO,        (オリジナル設定は DX = 20000.DO)
DY = 5000.DO,        (オリジナル設定は DY = 20000.DO)
/
```

MPI 並列数も変更している場合は、&PARAM_ATMOS_GRID_CARTESC の設定において、下記の関係を満たしていれば正解である。

$$X \text{ 方向の格子数} = (IMAX \times PRC_NUM_X) = 360 \quad (A.1)$$

$$Y \text{ 方向の格子数} = (JMAX \times PRC_NUM_Y) = 360$$

これに加えて、力学変数の時間積分に対する時間刻み幅 (TIME_DT_ATMOS_DYN) と (ATMOS_DYN_TINTEG_SHORT_TYPE) の調整も必要である (第 4.2.7 節を参照)。また、緩和領域も格子間隔の 20 倍から 40 倍となるように設定しなければならない。以下はその解答例であり、緩和領域を格子間隔の 20 倍に取った場合を示している。

```
&PARAM_PRC_CARTESC
  BUFFER_DX = 100000.DO, (オリジナル設定はBUFFER_DX = 400000.DO,)
  BUFFER_DY = 100000.DO, (オリジナル設定はBUFFER_DY = 400000.DO,)
/
```

4. 計算領域の位置を変更したい

計算領域の中心位置の座標を下記の通り、変更すれば良い。ここで、単位を度で設定する必要があることに注意されたい。例えば、139 度 45.4 分 = 139 + 45.5/60 度である。

```
&PARAM_MAPPROJECTION
  MPRPROJECTION_basepoint_lon = 139.7567D0, (オリジナル設定は135.220404D0)
  MPRPROJECTION_basepoint_lat = 35.6883D0, (オリジナル設定は34.653396D0)
  MPRPROJECTION_type = 'LC',
  MPRPROJECTION_LC_lat1 = 30.00D0,
  MPRPROJECTION_LC_lat2 = 40.00D0,
/
```

5. 積分時間を変更したい

```
&PARAM_TIME
  TIME_STARTDATE = 2007, 7, 14, 18, 0, 0,
  TIME_STARTTMS = 0.DO,
  TIME_DURATION = 12.0D0, (オリジナル設定は 6.0D0,)
  TIME_DURATION_UNIT = "HOURL",
/
```

さらに、scale-rm_init によって境界値を 12 時間以上用意しておく必要がある。第 4.1.1 節を参照して、(NUMBER_OF_FILES) の数を 3 以上に設定しなければならない。

6. 出力変数の追加と出力時間間隔の変更を行いたい

[PARAM_FILE_HISTORY] の中の (FILE_HISTORY_DEFAULT_TINTERVAL) を下記のように変更する。出力変数は、[HISTORY_ITEM] の中の (NAME) で指定する。ヒストリ変数の一覧は、リファレンスマニュアルで確認できる (第 5.3.2 節を参照)。

```
&PARAM_FILE_HISTORY
  FILE_HISTORY_DEFAULT_BASENAME = "history_d01",
  FILE_HISTORY_DEFAULT_TINTERVAL = 1800.DO, (オリジナル設定は3600.DO,)
  FILE_HISTORY_DEFAULT_TUNIT = "SEC",
/

&HISTORY_ITEM NAME="SFLX_SW_up" / 追加
&HISTORY_ITEM NAME="SFLX_SW_dn" / 追加
```

7. リスタート計算をしたい

最初の 3 時間分の積分については、run.conf を下記のように設定する。積分 3 時間目に、リスタートファイルが作成される。

```
&PARAM_TIME
TIME_STARTDATE = 2007, 7, 14, 18, 0, 0,
TIME_STARTMS = 0.DO,
TIME_DURATION = 3.0D0,          3 時間以上であれば良い。
TIME_DURATION_UNIT = "HOUR",
....(省略)....
TIME_DT_ATMOS_RESTART = 10800.DO,
TIME_DT_ATMOS_RESTART_UNIT = "SEC",
TIME_DT_OCEAN_RESTART = 10800.DO,
TIME_DT_OCEAN_RESTART_UNIT = "SEC",
TIME_DT_LAND_RESTART = 10800.DO,
TIME_DT_LAND_RESTART_UNIT = "SEC",
TIME_DT_URBAN_RESTART = 10800.DO,
TIME_DT_URBAN_RESTART_UNIT = "SEC",
/

&PARAM_RESTART
RESTART_OUTPUT = .true.,          (オリジナル設定は.false.,)
RESTART_IN_BASENAME = "../init/init_d01_20070714-180000.000",
RESTART_OUT_BASENAME = "restart_d01",          追加
/

&PARAM_ATMOS_BOUNDARY
ATMOS_BOUNDARY_TYPE = "REAL",
ATMOS_BOUNDARY_IN_BASENAME = "../init/output/boundary_d01",
ATMOS_BOUNDARY_START_DATE = 2010, 7, 14, 18, 0, 0,          追加。なくてもよい。
ATMOS_BOUNDARY_UPDATE_DT = 21600.DO,
/
```

(TIME_DURATION) を 3 時間に設定し、(RESTART_OUTPUT) を .true. とした場合は、リスタートファイルは積分終了時に作成される。したがって、(TIME_DT_ATMOS_RESTART)、(TIME_DT_OCEAN_RESTART)、(TIME_DT_LAND_RESTART)、(TIME_DT_URBAN_RESTART) を設定する必要はない。(TIME_DURATION) を 3 時間以上に設定した場合には、(TIME_DT_ATMOS_RESTART)、(TIME_DT_OCEAN_RESTART)、(TIME_DT_LAND_RESTART)、(TIME_DT_URBAN_RESTART) に、3 時間 (10800 秒) の約数かつ (TIME_DT) の倍数を指定する必要がある。

3 時間目から 6 時間目までのリスタート計算に対する設定は下記である。

```
&PARAM_TIME
TIME_STARTDATE = 2007, 7, 14, 21, 0, 0,
TIME_STARTMS = 0.DO,
TIME_DURATION = 3.0D0,          3 時間以上であれば良い。
TIME_DURATION_UNIT = "HOUR",
/

&PARAM_RESTART
RESTART_OUTPUT = .true.,          あってもなくてもよい。
RESTART_IN_BASENAME = "restart_d01_20070714-210000.000",          必須
RESTART_OUT_BASENAME = "restart2_d01",          あってもなくてもよい。
/

&PARAM_ATMOS_BOUNDARY
ATMOS_BOUNDARY_TYPE = "REAL",
ATMOS_BOUNDARY_IN_BASENAME = "../init/output/boundary_d01",
ATMOS_BOUNDARY_START_DATE = 2010, 7, 14, 18, 0, 0,          必須
ATMOS_BOUNDARY_UPDATE_DT = 21600.DO,
/
```


SCALE USERS GUIDE [日本語版]

執筆・編集

Team SCALE ユーザーズガイド制作委員会 (UGC Working Group)

本書中に不明点やお気づきの点、ご要望がございましたら、
SCALE ユーザー窓口 scale@ml.riken.jp
までご連絡ください。

Copyright ©Team SCALE, RIKEN R-CCS, 2016, 2017, 2018. All rights reserved.