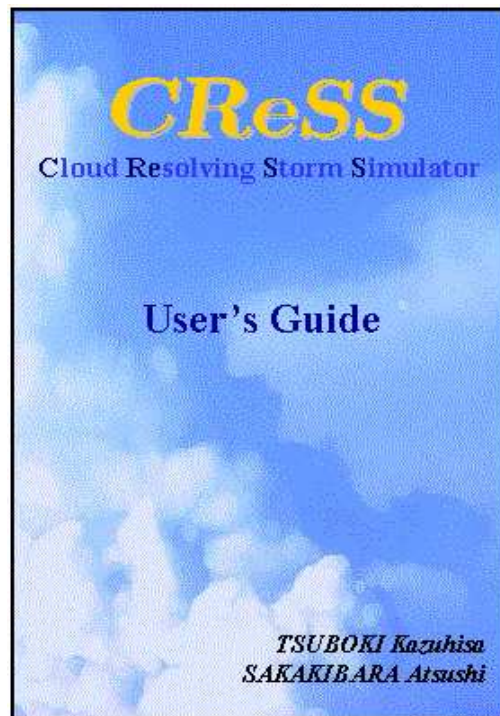


# *CReSS*

## User's Guide Ver.2



Kazuhisa Tsuboki Atsushi Sakakibara



本モデルの「公開」は、「完全なる自由な利用」を意味するものではありません。  
次項に示されている事柄を、必ず遵守してください。

本モデルの開発者ならびに該当する所属機関の権利は、堅く守られています。

名古屋大学地球水循環研究センター  
坪木 和久

財団法人高度情報科学技術研究機構  
榊原 篤志

ここで開発された気象モデル **CReSS** (Cloud Resolving Storm Simulator) は、開発途上の特別な理由がある場合を除いて、その利用目的が営利、非営利にかかわらず、紳士の姿勢であることを前提として、かつ、下記の点を了承する限りにおいて、原則的に完全公開とします。

- 一、完全公開とは、その利用において、個人、団体、営利目的、非営利目的、日本国の内外を問わないということです。ただし、非平和的活動に関する目的の場合はこの限りではありません。
- 一、公開されるモデルの利用は、全て利用者の責任においてなされるものとします。本モデルを利用したことにより生じた如何なる損害についても、たとえそれが公開されたプログラムソースに含まれるバグに原因があったとしても、その開発者、提供者、および、管理者等に、その責任を問うことはできません。
- 一、本モデルの変更は利用者の自由ですが、その結果生じた如何なる損害についても、前項と同様に責任を問うことはできません。
- 一、本モデルを利用する場合は、その開発者の名誉を尊重して下さい。論文、報告書等、本モデルを利用したことにより生じた如何なる結果についても、その本文、謝辞、または、それに該当するものにおいて、本モデルを利用したことを必ず明記して下さい。これは、ユーザーズマニュアル等、本モデルに関連する全てのものに適用されます。
- 一、本モデル、あるいは、その改造版を第三者に提供することは自由ですが、その場合も前項の全ての条件が適用されます。
- 一、本モデルは絶えずそのバージョンの更新がなされます。そのため、古いバージョンで得られた結果やモデルに用いるデータが、更新されたバージョンにおいても同じように得られる、あるいは、用いられることは保証しません。
- 一、本モデルの利用にあたり、人的資源の問題から、基本的にサポート体制はとりません。バグレポート、バージョンアップ等は、利用者自身で確認する必要があります。
- 一、本モデルの利用において、発見したバグ、あるいは、変更したほうがよい点、その他の意見等は、可能な限り開発者に連絡をして下さい。また、利用者が開発したものの本モデルへの提供は、何らの障害もありません。
- 一、本モデルの利用者は、その利用にあたり如何なる場合も紳士的であることを前提としています。

上記事項に同意する方に限り、**CReSS** を [http://www.tokyo.rist.or.jp/CReSS\\_Fujin](http://www.tokyo.rist.or.jp/CReSS_Fujin) からダウンロードすることができます。

# Introduction

In recent years, a massive computer has developed. In the last 20th century, it became that its theoretical performance has tera FLOPS, its main memory has giga byte and its storage has tera byte. It is evidence that in the early 21st century we will be able to do the super massive calculation which we have never considered to do. What should we do using its capability of calculation? We can say that one of aspects of meteorology and atmospheric science is the pursuit for faster and more massive numerical models. A cloud model is a model, which expresses complicate interactions between the atmosphere as continuum and water vapor and the particles of cloud and precipitation in the atmosphere. It is one of the most massive numerical model in meteorology and atmospheric physics.

Study on cloud and precipitation using computer science has just begun to develop. Here, we name the meso-scale meteorology using computer 'computing meso-scale meteorology', it is still done by very few researchers. There are so many unknown things which are the formulation of the physical processes, the methods of calculations which are efficient and fewer error, the problems on the calculations and so on. Atmospheric phenomena are various and complicated, and there are a lot of unknown and undiscovered things. Using numerical models, we should be able to understand the various phenomena. However, we could have spent insufficient time and efforts for the 'computing meso-scale meteorology.'

What is it necessary for development of the study of the 'computing meso-scale meteorology?' Various numerical models must be created from various points of view. A large number of numerical experiments and verifications with the models are necessary for the development of the 'computing meso-scale meteorology.' We hope that the models develop over the various findings based on enormous calculations.

In recent, models develop to be very massive and complicate. It is impossible and inefficient that an individual researcher makes a such model from the beginning. Researchers have respective purposes, which are to create the method of calculation of numerical model and physical processes, to do the simulation of realistic phenomena, to experiment in the forecasts of meteorology and so on. The basic model is necessary for the purposes. If we are easy to use the basic model, it is possible that we study various problems by using the basic model, respectively. We develop the numerical model for clouds and meso-scale meteorological phenomena in Special Coordination Funds for Promoting Science and Technology (SCF), 'Study on Advanced Simulations of Down Burst Phenomenon' (research representative: Prof. Nobuhiko Kamiya) . The purpose of this model is to be the platform of studies as mentioned above.

In recent, some models which can represent clouds with non-hydrostatic and compressible equations including cloud physical processes have been developed. We referred Advanced Regional Prediction System (ARPS), which was developed by the University of Oklahoma, and the non-hydrostatic meso-scale model developed by the Meteorological Research Institute in Japan for the development of this model. Our works for development of the model are based on knowledgements from previous studies. We would like to pay our respects to the previous efforts.

Computers are rapidly developing, especially for hard wears. A fast single vector processor was mainly used for massive calculations until a few years ago. In recent, parallel computers with multiple processors become often to be used. Our model is designed to get a high efficiency of calculation on a parallel computer. The cloud model includes many physical processes with a high resolution, while its calculation

domain is smaller than those of global and climate models. Then, the parallel computer is very useful for the huge calculations with the cloud model.

Our model: ***CReSS*** (Cloud Resolving Storm Simulator) can reproduce clouds and simulate meso-scale precipitation systems organized from the clouds. ***CReSS*** (Cloud Resolving Storm Simulator) has a high resolution in horizontal and vertical, and expresses physical processes of clouds and precipitation as sufficient as possible. Processes of clouds and precipitation still has been clarified insufficiently. Both observational studies and the development of the cloud model are needed to get the further understandings of clouds and precipitation systems in the future. The goal of the development of ***CReSS*** is to contribute for a prediction of heavy rainfalls based on those understandings.

***CReSS*** is full open for public as the open provisions. This guide concretely describes the program codes and the way of the use of ***CReSS***. We hope that this guide is useful for all of ***CReSS*** users. The guide, however, should be revised with the improvement of the numerical model. Since the explanations of the guide may be insufficient, we hope that you give us any advices about our model and this guide.

---

This work is conducted as one of 'Study for Advanced Simulation of Down Burst Phenomenon' with Special Coordination Funds for Promoting Science and Technology; 'development of parallel softwares for large-scale simulation to predict the global climate change with high resolution.' The leader of this research project is Prof. Nobuhiko Kamiya, school of community policy, Aichi Gakusen University. We got an opportunity to develop ***CReSS*** by courtesy of Prof. Nobuhiko Kamiya. We thank for the support by the Agency for Science and Technology, and Research Organization for Information Science and Technology. Especially, we have been supported by Prof. Akimasa Sumi, Center of Climate System Research, University of Tokyo, Prof. Takao Takeda, Nagoya University and Dr. Yonejiro Yamagishi and Mr. Nakamura at the Research Organization for Information Science and Technology from the beginning of this project. We exchanged many informations with Mr. Takashi Arakawa at Research Organization for Information Science and Technology.

***CReSS*** is developed in Institute of Hydro-spheric-Atmospheric Science. We use the HITACHI SR8000 at Information Technology Center, University of Tokyo, FUJITSU VPP5000 at Information Technology Center, Nagoya University, HITACHI SR2201 and NEC SX4 at Japan Atomic Energy Research Institute, numerical wind tunnel at National Aerospace Laboratory of Japan and Compaq Computer Corporation work stations. We are also supported for software by CTI Co., Ltd. The authors express their appreciation to all of them.

September 2000

**Kazuhisa Tsuboki**  
**Atsushi Sakakibara**

By April 2001, 'Institute of Hydrospheric-Atmospheric Science' is reorganized by 'Hydrospheric Atmospheric Research Center'.

# For the second edition

The first edition of this *CReSS* User's Guide was published at September 22, 2000. That was a material for the first seminar of meteorological model for parallel computing sponsored by the Foundation of Research Organization for Information Science and Technology. We have since developed the *CReSS*, so the second seminar was to be taken place sponsored by the foundation. We added to write some new contents in the second edition of *CReSS* User's Guide: the method of numerical computation, the process between the boundary layer of atmosphere and surface and the result of the simulation of the squall line.

*CReSS* have be developed to adopt processes of cloud physics, turbulence, atmospheric boundary layer and land surface. *CReSS* become to develop a model which reproduces most meteorological phenomena excepting stratus clouds in which the radiation is important. We are doing the forecast experiment by using the three dimensional and time-various data as the initial and lateral boundary values. We have carried through the first stage of the development of the cloud resolving model although a number of problems to develop are remained. We will advance the revision of User's Guide with the development of the model. The second edition of *CReSS* User's Guide includes the latest informations for the present *CReSS*.

We would like to develop the User's Guide so that many researchers understand and use *CReSS*.

The development of *CReSS* is supported by many people and institutions mentioned the introduction of the first edition. We thank to whome it may concern and we hope to support us more.

September 2001

Kazakhs Tsuboki  
Atsushi Sankara





# Contents

Introduction	v
For the second edition	vii
<b>I Fundamental Theory and Formulations</b>	<b>1</b>
<b>1 Summary of <i>CReSS</i></b>	<b>3</b>
1.1 Progress of development . . . . .	4
1.2 Outline of model . . . . .	4
<b>2 Formulation of the system of basic equations</b>	<b>7</b>
2.1 The basic equations system (in terrain-excluding coordinates) . . . . .	8
2.2 The basic equations system (in terrain-following coordinates) . . . . .	9
2.2.1 General curvilinear coordinates . . . . .	9
2.2.2 Coordinate along terrain . . . . .	12
2.2.3 The basic dynamical equations in terrain-following coordinates . . . . .	14
2.3 基本方程式系 — 地図投影 . . . . .	17
2.3.1 直交曲線座標 . . . . .	17
2.3.2 地図投影座標系における基本方程式系 — 地形を含まない場合 . . . . .	19
2.3.3 地図投影座標系における基本方程式系 — 地形を含む場合 . . . . .	23
2.3.4 正角投影図法 . . . . .	26
<b>3 Subgrid Scale Diffusion</b>	<b>29</b>
3.1 Parameterization of Turbulence Transport . . . . .	30
3.2 Eddy Viscosity Model . . . . .	31
3.2.1 Formulation of Diffusion Term . . . . .	31
3.2.2 One order closure of Smagorinsky . . . . .	33
3.2.3 One and a half order closure with turbulence kinetic energy . . . . .	35
<b>4 Physical Processes of Cloud and Precipitation</b>	<b>39</b>
4.1 Parameterization of “Warm Rain” with a Bulk Method . . . . .	40
4.1.1 Equations for “Warm Rain” Process of Cloud and Precipitation . . . . .	40
4.1.2 Microphysical Processes . . . . .	41
4.2 Parameterization of Cloud and Precipitation with the Mixed-phase Processes . . . . .	42
4.2.1 Parameterization of the Mixed-phase Processes with a Bulk Method . . . . .	42
4.2.2 The Equation System for Cloud and Precipitation Processes . . . . .	43
4.2.3 Expression of Particles of Cloud and Precipitation . . . . .	46
4.2.4 Formulations of Physical Processes of Source terms . . . . .	52
4.2.5 Moist Saturation Adjustment . . . . .	74
4.2.6 Formulation of Variational Terms on Mixing Ratio and Number Concentration with Precipitation . . . . .	75
4.2.7 The other constants . . . . .	76

<b>5</b>	<b>Atmospheric Boundary Layer, Ground Surface Process and Ground Temperature</b>	<b>77</b>
5.1	Basic Theories of Atmospheric Boundary Layer . . . . .	78
5.1.1	Structure of Atmospheric Boundary Layer . . . . .	78
5.1.2	Parameterization of Turbulence Transportation . . . . .	78
5.1.3	Mixing Length Theory . . . . .	79
5.1.4	Surface Boundary Layer . . . . .	80
5.1.5	Vertical Flux . . . . .	81
5.2	Calculations of Atmospheric Boundary Layer, Ground Surface Process and Soil Temperature	84
5.2.1	Cloudiness . . . . .	85
5.2.2	Solar Radiation and Downward Long Wave Radiation . . . . .	87
5.2.3	Flux at the Ground Surface . . . . .	89
5.2.4	Atmospheric Boundary Layer Process . . . . .	97
5.2.5	Ground Temperature . . . . .	102
<b>6</b>	<b>Numerical computation method</b>	<b>107</b>
6.1	The discretization of basic equation system . . . . .	108
6.1.1	The outline of the numerical solution . . . . .	108
6.1.2	Grid in the model and setting of variables . . . . .	108
6.1.3	Discretization by finite difference method . . . . .	109
6.2	Artificial numerical viscosity term . . . . .	123
6.2.1	Necessity of numerical viscosity term . . . . .	123
6.2.2	Numerical viscosity with the second-order or fourth-order diffusion term . . . . .	124
6.3	Boundary condition . . . . .	127
6.3.1	Lateral boundary condition . . . . .	127
6.3.2	Boundary conditions at the top and bottom . . . . .	132
6.3.3	Sponge layer . . . . .	134
<b>II</b>	<b>Technical Description</b>	<b>137</b>
<b>7</b>	<b>Application of a parallel processing</b>	<b>139</b>
7.1	Technique for parallelization . . . . .	140
7.1.1	Two-dimensional domain decomposition . . . . .	140
7.1.2	Example of parallel computing . . . . .	141
7.2	Test of the parallel program . . . . .	143
7.2.1	Verification of coincidence of a calculation result . . . . .	143
7.2.2	Efficiency of parallel processing . . . . .	143
<b>8</b>	<b>Program lists and tree diagram</b>	<b>145</b>
8.1	Program list . . . . .	146
8.1.1	Main programs of <i>CReSS</i> . . . . .	146
8.1.2	List of subroutines . . . . .	146
8.1.3	list of include file . . . . .	152
8.1.4	MPI subroutines list . . . . .	152
8.2	Tree diagrams of programs . . . . .	153
8.2.1	<i>solver</i> . . . . .	153
8.2.2	<i>gridata</i> . . . . .	178
8.2.3	<i>terrain</i> . . . . .	181
8.2.4	<i>gather</i> . . . . .	182
<b>9</b>	<b>Configuration and Execution</b>	<b>183</b>
9.1	User configuration file . . . . .	184
9.1.1	Notifications . . . . .	184
9.1.2	Details of the user configuration file . . . . .	184
9.2	Data flow in <i>CReSS</i> . . . . .	198
9.2.1	Outline of data flow . . . . .	198
9.2.2	Data flows of each configuration . . . . .	200

---

9.3	Formats of I/O data . . . . .	203
9.3.1	Format of the sounding data . . . . .	203
9.3.2	Formats of the history dumped file and the geographic information file . . . . .	204
9.4	Execution of <i>CReSS</i> . . . . .	207
9.4.1	Execution of <i>solver</i> . . . . .	207
9.4.2	Execution of <i>gather</i> . . . . .	208
9.4.3	Execution of <i>terrain</i> . . . . .	209
9.4.4	Execution of <i>gridata</i> . . . . .	211



**Part I**

**Fundamental Theory and  
Formulations**



# Chapter 1

## Summary of *CReSS*

*CReSS* (Cloud Resolving Storm Simulator) is the non-hydrostatic meteorological model to resolving cloud developed to the purpose to simulate meteorological phenomena from cloud scale to meso scale highly.

Clouds, especially cumulonimbus cloud, giving heavy precipitation, and organized cumulonimbus clouds, have very complex system and the processes of their development are determined by the flow field and the very complex non-linear interaction. It is important that to simulate such precipitation system for numerical model is to calculate the process of cloud physics in detail with the process of flow field.

*CReSS* is made to be able to calculate for massive parallel calculator effectively and time developing simulation of cloud in detail can do by massive parallel calculator.

In this chapter, we explain Progress of development of *CReSS* and Outline of model.

## 1.1 Progress of development

*CReSS* (Cloud Resolving Storm Simulator) is developed by Kazuhisa Tsuboki, Hydrospheric Atmospheric Research Center, the University of Nagoya, and Atsushi Sakakibara, Research Organization for Information Science and Technology, for one of 'Study for Advanced Simulation of Down Burst Phenomenon' for Special Coordination Funds for Promoting Science and Technology, 'to develop parallel software for large-scale simulation that enables us to predict the global climate change with high resolution' and under the guidance by Prof. Nobuhiko Kamiya, Aichi Gakusen University.

In the plans of this work, It is the purposes that we develop the numerical model which can do the numerical simulation of down burst and micro burst by parallel computers and investigate the incidence and the behavior of descending flow and diverging flow by it.

Down burst rise from much developing cumulonimbus, which formation relates to the proses of cloud physics. The model which simulates them sufficiently should be able to describe in detail to phenomenon for cumulonimbus in dynamical and cloud physics. In the climate model, it is important for cloud to be the source of drive of atmosphere. But the model which simulates such as down burst is expected to utilization to make a model of the constitute of cloud with time variation and to appear the parameter of cloud for a climate model. It is necessary for the simulation of micro burst, which is the down burst below 4km in diameter, to be the model which can resolve cloud because the phenomena of micro burst has the scale in individual cumulonimbus. The generation of negative buoyancy occurring descending flow occurs that atmospheric cooling by the phase change of precipitation particle and weight by precipitation particle drag down atmosphere. For the simulation of generation of micro burst, it is necessary to simulate exactly how precipitation particles are formed cumulonimbus and how they behave cumulonimbus(fig.1.1).

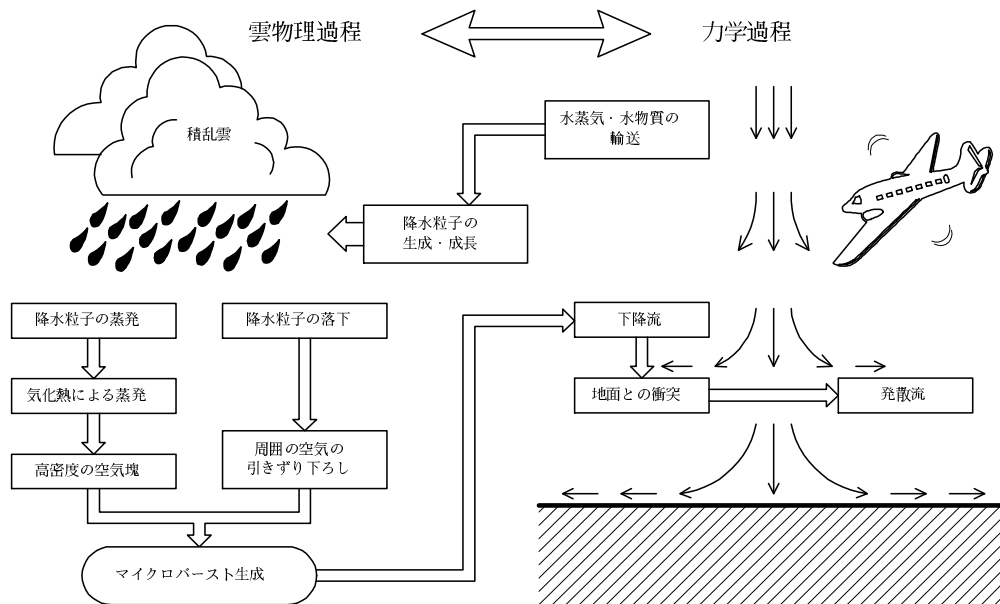


Figure 1.1. Conceptual model of the simulation model of micro burst.

It is the purpose that, to develop *CReSS*, we can simulate such environment and the behavior of precipitation particles and can practice the model best by parallel computers.

## 1.2 Outline of model

*CReSS* is the model of cloud resolution which can simulate down burst and micro burst, which has the field of calculation in meso-scale.



The characters of *CReSS* are given below.

- *CReSS* is designed for parallel computers, so it can be performed massive calculation. On the other hand, it has the version for one processor element, so it can also be performed by PC-UNIX.
- *CReSS* is the cloud model included in possible detail the processes of cloud physics. On the other hand, it is also possible to perform the intimate fluid field for dry model.
- *CReSS* is designed not only for the experiment of cloud simulation separately but also for the experiment to forecast, giving *CReSS* realistic terrain, initiation and boundary condition developing in time.
- *CReSS* is written by the coda of FORTRAN77, but using the extended function, so you can decode easily and can perform in almost computers.

Next, the concrete functions equipped with *CReSS* are given below.

- The basic equations of the process of dynamics are non-hydrostatic and compressible and performed in the field of three dimensional grid beside terrain.
- The dependent various values of the process of dynamics are the three components of velocity, the deviation of potential temperature, the deviation of pressure and the energy of turbulence.
- The processes of cloud physics introduce the bulk parametrization of warm rain and the bulk parametrization with ice phase
- The dependent various values of the process of cloud physics are the mixing ratio of water vapor, cloud water and rain water. In addition, in the bulk parametrization with ice phase, they are added the mixing ratio and numerical density of cloud ice, snow and graupel.
- Difference method is utilized to the method of differential of space and you can use the explicit method for both horizontal and vertical grids(HE-VE) or the explicit method for horizontal grid and the implicit method for vertical grid(HE-VI).
- The time differential method which divides the term related to the sound of wave and the other terms is utilized to the method of time integration. The method of leap frog is utilized to the integration with using Asselin filter together except the term related to the sound of wave.
- The parameterizations of the first order closure by Smagorinsky or the 1.5 order closure used the energy of turbulence are utilized to turbulence.
- The central difference having fourth or second order precision using fourth or second order numerical viscosity together is utilized to the calculation of advection.
- In the initialized condition, the first dimensional profile by upper-air observation and so on is given to the model horizontally and uniformly, or, for the nesting to wide area model, the three dimensional unequal data compensating the output values is given to the model.
- The condition of fixed boundary (mirror boundary), the condition of periodic boundary, the condition of zero-inclination and the condition of radial boundary are utilized to the boundary conditions.
- The sponge layer can be utilized to upper layer.
- MPI(Message Passing Interface) is utilized to parallel computers, so the model can do parallel calculation by the two dimensional deviation of rectangular field.

By this User's Guide, you can appear the detail of the functions(dynamical and physical processes) given and the method of equipment of program for parallel computers and can see the concrete method of practice of model and the result of inspecting experiment in order.



## Chapter 2

# Formulation of the system of basic equations

The governing equations of *CReSS* is expressed by the equation of motion which is Navier-Stokes equation considering the rotation of the earth, the equation of thermodynamics, the continuity equation with compression, the equation of water vapor mixing ratio, the equation of cloud mixing ratio and precipitation particle and the equation of number density of cloud and precipitation particle. The model is constituted by the formulation of various physical processes and the formulation of boundary value besides their equations. In this chapter, we collect the formulation of basic equations in the constitution.

*CReSS* can adopt the effect of terrain by taking the grid along terrain. In *CReSS*, this grid is transformed to the square grid for the calculation. To understand the model easily, first we describe the case of grid in terrain-excluding coordinates. Secondly, we describe the transformation of the grid in terrain-following coordinates and the equations in the case.

## 2.1 The basic equations system (in terrain-excluding coordinates)

The independent variables of the model are the space coordinates,  $x, y, z$  and time,  $t$ . In the equations of semi-compression adopted in *CReSS*, the dependent variables which defined as a function of the independent variables are horizontal velocity,  $u, v$ , vertical velocity,  $w$ , deviation from the basic status of potential temperature,  $\theta'$ , deviation from the basic status of pressure,  $p'$ , mixing ratio of water vapor,  $q_v$ , mixing ratio of water contents,  $q_x$ , and number density of water contents  $N_x$ . Here,  $q_x, N_x$  are for water contents except water vapor and they are determined how the processes of cloud and precipitation are expressed. Corresponding to this, the number of time-developing equations are varied. Here, dependent variables of potential temperature, pressure and density considering water contents and water vapor are fulfilled by hydrostatic equilibrium,

$$\frac{\partial \bar{p}}{\partial z} = -\bar{\rho}g \quad (2.1)$$

The relation between the basic states and the deviation from them are given as,

$$\theta = \bar{\theta} + \theta' \quad (2.2)$$

$$p = \bar{p} + p' \quad (2.3)$$

$$\rho = \bar{\rho} + \rho' \quad (2.4)$$

The density is given by the equation of state diagnostically,

$$\rho = \frac{p}{R_d T} \left( 1 - \frac{q_v}{\epsilon + q_v} \right) \left( 1 + q_v + \sum q_x \right) \quad (2.5)$$

Here,  $g$  is the gravity acceleration,  $\epsilon$  is the ratio of molecule number between water vapor and dry air and  $R_d$  is gas constant of dry air.

All of the dependent variables except density are described in time-developing equations. In the case of the terrain-excluding coordinate, the time-developing equations of the dependent variables are given as follows. In the actual model, these equations are coordinated beside the basic equations which contain terrain in Section 2.2. The terrain-excluding coordinate is used to understand the basic equations easily which result in as follows,

### Equation of motion

$$\frac{\partial \bar{\rho}u}{\partial t} = -\bar{\rho} \left( u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} \right) - \frac{\partial p'}{\partial x} + \bar{\rho} (f_s v - f_c w) + \text{Turb}.u \quad (2.6)$$

$$\frac{\partial \bar{\rho}v}{\partial t} = -\bar{\rho} \left( u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + w \frac{\partial v}{\partial z} \right) - \frac{\partial p'}{\partial y} - f_s \bar{\rho}u + \text{Turb}.v \quad (2.7)$$

$$\frac{\partial \bar{\rho}w}{\partial t} = -\bar{\rho} \left( u \frac{\partial w}{\partial x} + v \frac{\partial w}{\partial y} + w \frac{\partial w}{\partial z} \right) - \frac{\partial p'}{\partial z} - \bar{\rho} \text{Buoy}.w + f_c u + \text{Turb}.w \quad (2.8)$$

Here,  $f_s, f_c$  is Coriolis-parameter and  $\text{Buoy}.w$  is the term of buoyancy.

### Equation of pressure

$$\begin{aligned} \frac{\partial p'}{\partial t} = & - \left( u \frac{\partial p'}{\partial x} + v \frac{\partial p'}{\partial y} + w \frac{\partial p'}{\partial z} \right) + \bar{\rho} g w \\ & - \bar{\rho} c_s^2 \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right) + \bar{\rho} c_s^2 \left( \frac{1}{\theta} \frac{d\theta}{dt} - \frac{1}{Q} \frac{dQ}{dt} \right) \end{aligned} \quad (2.9)$$

Here,  $c_s$  is the speed of sound in atmosphere and  $Q = 1 + 0.61q_v + \sum q_x$ .

### Equation of potential temperature

$$\frac{\partial \bar{\rho} \theta'}{\partial t} = -\bar{\rho} \left( u \frac{\partial \theta'}{\partial x} + v \frac{\partial \theta'}{\partial y} + w \frac{\partial \theta'}{\partial z} \right) - \bar{\rho} w \frac{\partial \bar{\theta}}{\partial z} + \text{Turb.}\theta + \bar{\rho} \text{Src.}\theta \quad (2.10)$$

### Equations of water vapor and mixing ratio of wear contents

$$\frac{\partial \bar{\rho} q_v}{\partial t} = -\bar{\rho} \left( u \frac{\partial q_v}{\partial x} + v \frac{\partial q_v}{\partial y} + w \frac{\partial q_v}{\partial z} \right) + \text{Turb.}q_v + \bar{\rho} \text{Src.}q_v \quad (2.11)$$

$$\frac{\partial \bar{\rho} q_x}{\partial t} = -\bar{\rho} \left( u \frac{\partial q_x}{\partial x} + v \frac{\partial q_x}{\partial y} + w \frac{\partial q_x}{\partial z} \right) + \text{Turb.}q_x + \bar{\rho} \text{Src.}q_x + \bar{\rho} \text{Fall.}q_x \quad (2.12)$$

### Equation of number density of water contents

$$\begin{aligned} \frac{\partial N_x}{\partial t} = & -\bar{\rho} \left[ u \frac{\partial}{\partial x} \left( \frac{N_x}{\bar{\rho}} \right) + v \frac{\partial}{\partial y} \left( \frac{N_x}{\bar{\rho}} \right) + w \frac{\partial}{\partial z} \left( \frac{N_x}{\bar{\rho}} \right) \right] \\ & + \text{Turb.} \frac{N_x}{\bar{\rho}} + \bar{\rho} \text{Src.} \frac{N_x}{\bar{\rho}} + \bar{\rho} \text{Fall.} \frac{N_x}{\bar{\rho}} \end{aligned} \quad (2.13)$$

The detail of other dependent variables are described in Section 2.2. The term of diffusion in sub-grid scale  $\text{Turb.}\phi$  which appears except in equation of pressure is described in Chapter ??, and the terms of generation and disappearance  $\text{Src.}\phi$  and falling  $\text{Fall.}\phi$  which appear in equation of potential temperature and water contents are described in Chapter 4.

## 2.2 The basic equations system (in terrain-following coordinates)

### 2.2.1 General curvilinear coordinates

**CReSS** adopts the terrain-following coordinate to comprise terrain effect. The basis of this coordinate system doesn't necessarily become to orthonormal, while the vector in Cartesian coordinate is expressed by orthogonal basis. This is categorized in curvilinear coordinate of liner algebra. Here, the basic background is summarized.

#### Contravariant components and covariant components

Introducing general linear independent bases ( $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$ ), which are neither regular nor orthogonal, facultative vector  $\mathbf{A}$  is expressed as the linear combination,

$$\mathbf{A} = A^1 \mathbf{e}_1 + A^2 \mathbf{e}_2 + A^3 \mathbf{e}_3 \quad (2.14)$$

On the other hand, since the inverse bases ( $\mathbf{f}^1, \mathbf{f}^2, \mathbf{f}^3$ ), is defined as

$$\mathbf{f}^i \cdot \mathbf{e}_j = \delta_j^i \quad (2.15)$$

then, the components are,

$$\mathbf{f}^i \cdot \mathbf{A} = A^i \quad (2.16)$$

Here, Kronecker's delta is defined as follows,

$$\delta_j^i = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \quad (2.17)$$

In the case of orthonormal basis, coordinate components are given by scalar product of basis vector and facultative vector. In the case of general basis, however, the components must be given by scalar product of inverse basis vector and facultative vector. Thus components of coordinates for original basis are named contravariant components and expressed in a superscript. On the other hand, components of coordinates for inverse basis are named covariant components, expressed in a subscript.

The facultative vector  $\mathbf{B}$  is expressed by inverse basic vector as

$$\mathbf{B} = B_1 \mathbf{f}^1 + B_2 \mathbf{f}^2 + B_3 \mathbf{f}^3 \quad (2.18)$$

The solution of scalar product of them becomes

$$\mathbf{A} \cdot \mathbf{B} = A^i B_i \quad (2.19)$$

Thus, scalar product is expressed easily by the sum of product of components in the relative coordinate when both contravariant and covariant components are used. In the case of orthonormal basis, as inverse basis becomes same to original basis, there are no difference between contravariant and covariant components.

$$\mathbf{A} \cdot \mathbf{B} = A^i B^i \quad (2.20)$$

Scalar product of facultative vector  $\mathbf{A}$  and bases  $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$  gives covariant components for bases of vector  $\mathbf{A}$ ,  $(e_1, e_2, e_3)$ ,

$$A_i = \mathbf{A} \cdot \mathbf{e}_i \quad (2.21)$$

Next, for bases  $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$ , 9 scalar product can be made,

$$G_{ij} = \mathbf{e}_i \cdot \mathbf{e}_j \quad (2.22)$$

This matrix is named metric matrix. Moreover, this is quadratic tensor which is named metric tensor. Using this, the relationship between contravariant and covariant components is expressed as follows.

$$A^i = G_{ji} A_j \quad (2.23)$$

**curvilinear coordinates**

As a function of Cartesian coordinate, three function  $F^i(x, y, z)$  which defined at a certain region of a space is considered. The differentiation of them for  $(x, y, z)$  is possible for any number of times.

$$\xi = F^1(x, y, z) \quad (2.24)$$

$$\eta = F^2(x, y, z) \quad (2.25)$$

$$\zeta = F^3(x, y, z) \quad (2.26)$$

$(\xi, \eta, \zeta)$  correlates to the each point in a region  $P(x, y, z)$ . When this correlation is 1 to 1, it can be said that curvilinear coordinates is made in a region. Furthermore, the following condition is assumed.

$$\frac{\partial(x, y, z)}{\partial(\xi, \eta, \zeta)} = \begin{vmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} & \frac{\partial x}{\partial \zeta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} & \frac{\partial y}{\partial \zeta} \\ \frac{\partial z}{\partial \xi} & \frac{\partial z}{\partial \eta} & \frac{\partial z}{\partial \zeta} \end{vmatrix} \neq 0 \quad (2.27)$$

For Cartesian coordinate, vector  $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$  is

$$\mathbf{e}_1 = \begin{pmatrix} \frac{\partial x}{\partial \xi} \\ \frac{\partial y}{\partial \xi} \\ \frac{\partial z}{\partial \xi} \end{pmatrix}, \quad \mathbf{e}_2 = \begin{pmatrix} \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \eta} \\ \frac{\partial z}{\partial \eta} \end{pmatrix}, \quad \mathbf{e}_3 = \begin{pmatrix} \frac{\partial x}{\partial \zeta} \\ \frac{\partial y}{\partial \zeta} \\ \frac{\partial z}{\partial \zeta} \end{pmatrix} \quad (2.28)$$

Here, vector  $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$  is named 'basis' or 'fundamental vector' of curvilinear coordinates  $(\xi, \eta, \zeta)$ . Using above, facultative vector field  $\mathbf{A}$  is expressed as the linear combinatio of them.

$$\mathbf{A} = A^1 \mathbf{e}_1 + A^2 \mathbf{e}_2 + A^3 \mathbf{e}_3 \quad (2.29)$$

Then,  $A^i$  is named 'contravariant components' of curvilinear coordinates  $(\xi, \eta, \zeta)$  of vector field  $\mathbf{A}$ . The covariant component is mentioned in (2.21). Samely, for the bases  $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$ , 9 scalar product can be made.

$$G_{ij} = \mathbf{e}_i \cdot \mathbf{e}_j \quad (2.30)$$

Matrix  $G_{ij}$  is named metric matrix of curvilinear coordinates  $(\xi, \eta, \zeta)$ . The relationship between contravariant and covariant components is given by (2.23).

### 2.2.2 Coordinate along terrain

Many cloud models adopt terrain-following coordinate system to consider the effect of terrain. Such coordinate system are oriented on the special relationship of curvilinear coordinates.

*CReSS* adopts the terrain-following coordinate as well as NHM and ARPS.

$$\xi = x \quad (2.31)$$

$$\eta = y \quad (2.32)$$

$$\zeta = \zeta(x, y, z) \quad (2.33)$$

In this case, velocity vector of Cartesian coordinate can be expressed as well as (2.29) by components of velocity vector (contravariant components)  $(U, V, W)$  in the terrain-following coordinate. (Ordinally,  $(U, V, W)$  are contravariant components and they should be expressed in the superscript  $(u^1, u^2, u^3)$ . However, since it is easy, it is written in this way.)

$$u = U \frac{\partial x}{\partial \xi} + V \frac{\partial x}{\partial \eta} + W \frac{\partial x}{\partial \zeta} \quad (2.34)$$

$$v = U \frac{\partial y}{\partial \xi} + V \frac{\partial y}{\partial \eta} + W \frac{\partial y}{\partial \zeta} \quad (2.35)$$

$$w = U \frac{\partial z}{\partial \xi} + V \frac{\partial z}{\partial \eta} + W \frac{\partial z}{\partial \zeta} \quad (2.36)$$

Since it is assumed that condition (2.27) is realized, the inverse of the velocity vector can be asked by solving (2.34)~(2.36) about  $(U, V, W)$ .

$$UG^{\frac{1}{2}} = uJ_{\eta\zeta}^{yz} + vJ_{\eta\zeta}^{zx} + wJ_{\eta\zeta}^{xy} \quad (2.37)$$

$$VG^{\frac{1}{2}} = uJ_{\zeta\xi}^{yz} + vJ_{\zeta\xi}^{zx} + wJ_{\zeta\xi}^{xy} \quad (2.38)$$

$$WG^{\frac{1}{2}} = uJ_{\xi\eta}^{yz} + vJ_{\xi\eta}^{zx} + wJ_{\xi\eta}^{xy} \quad (2.39)$$

Here,  $J$  is Jacobian. For example, it is defined as

$$J_{\eta\zeta}^{yz} \equiv \frac{\partial(y, z)}{\partial(\eta, \zeta)} = \begin{vmatrix} \frac{\partial y}{\partial \eta} & \frac{\partial y}{\partial \zeta} \\ \frac{\partial z}{\partial \eta} & \frac{\partial z}{\partial \zeta} \end{vmatrix} \quad (2.40)$$

$G^{\frac{1}{2}}$  is defined as Jacobian of the coordinate conversion between  $(\xi, \eta, \zeta)$  and  $(x, y, z)$  and expressed as

$$G^{\frac{1}{2}} \equiv \frac{\partial(x, y, z)}{\partial(\xi, \eta, \zeta)} = \begin{vmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} & \frac{\partial x}{\partial \zeta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} & \frac{\partial y}{\partial \zeta} \\ \frac{\partial z}{\partial \xi} & \frac{\partial z}{\partial \eta} & \frac{\partial z}{\partial \zeta} \end{vmatrix} \quad (2.41)$$



In the case of terrain-following coordinate defined by (2.31)~(2.33), Jacobians which appear in (2.37)~(2.39) are as follows.

$$\begin{aligned}
J_{\eta\zeta}^{yz} &= \frac{\partial z}{\partial \zeta}, & J_{\eta\zeta}^{zx} &= 0, & J_{\eta\zeta}^{xy} &= 0, \\
J_{\zeta\xi}^{yz} &= 0, & J_{\zeta\xi}^{zx} &= \frac{\partial z}{\partial \zeta}, & J_{\zeta\xi}^{xy} &= 0, \\
J_{\xi\eta}^{yz} &= -\frac{\partial z}{\partial \xi}, & J_{\xi\eta}^{zx} &= -\frac{\partial z}{\partial \eta}, & J_{\xi\eta}^{xy} &= 1
\end{aligned} \tag{2.42}$$

In the case of three-dimension, Jacobian of the coordinate conversion between  $(\xi, \eta, \zeta)$  and  $(x, y, z)$  is

$$G^{\frac{1}{2}} = \left| \frac{\partial z}{\partial \zeta} \right| \tag{2.43}$$

Here, variable components of Jacobian's components are defined as

$$J_{31} \equiv J_{\xi\eta}^{yz} = -\frac{\partial z}{\partial \xi} \tag{2.44}$$

$$J_{32} \equiv J_{\xi\eta}^{zx} = -\frac{\partial z}{\partial \eta} \tag{2.45}$$

$$J_d \equiv J_{\eta\zeta}^{yz} = J_{\zeta\xi}^{zx} = \frac{\partial z}{\partial \zeta} \tag{2.46}$$

and  $\zeta$  is defined by using the altitude of surface  $z_{sfc}(x, y)$  and the height of the region of model  $z_{top}$ ,

$$\zeta(x, y, z) = \frac{z_{top}[z - z_{sfc}(x, y)]}{z_{top} - z_{sfc}(x, y)} \tag{2.47}$$

or

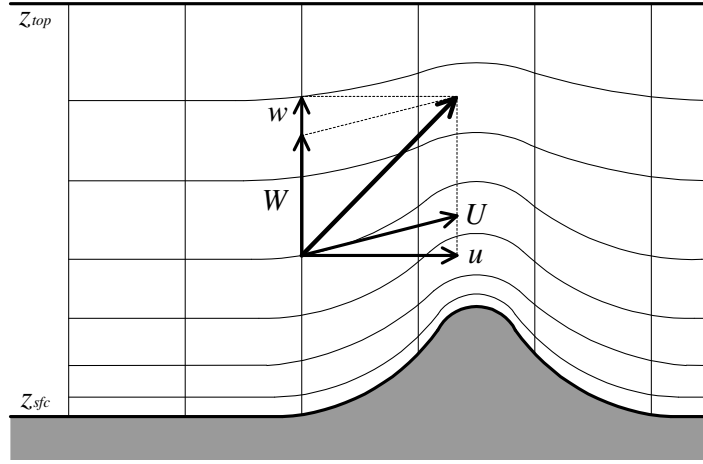
$$z(\xi, \eta, \zeta) = z_{sfc}(\xi, \eta) + \zeta \left[ 1 - \frac{z_{sfc}(\xi, \eta)}{z_{top}} \right] \tag{2.48}$$

In this case, Jacobian's various components are

$$J_{31} = -\frac{\partial z}{\partial \xi} = \left( \frac{\zeta}{z_{top}} - 1 \right) \frac{\partial z_{sfc}(\xi, \eta)}{\partial \xi} \tag{2.49}$$

$$J_{32} = -\frac{\partial z}{\partial \eta} = \left( \frac{\zeta}{z_{top}} - 1 \right) \frac{\partial z_{sfc}(\xi, \eta)}{\partial \eta} \tag{2.50}$$

$$J_d = \frac{\partial z}{\partial \zeta} = 1 - \frac{z_{sfc}(\xi, \eta)}{z_{top}} \tag{2.51}$$



**Figure 2.1.** Terrain-following coordinates and direction of the vector.

Like in this case, when  $\zeta$  is a monotonically increasing function about  $z$ , it is expressed that

$$G^{\frac{1}{2}} = |J_d| = J_d \quad (2.52)$$

Velocity of the terrain-following coordinate (contravariant velocity) which is given in (2.37)~(2.39) is transformed as follows,

$$U = u \quad (2.53)$$

$$V = v \quad (2.54)$$

$$W = (uJ_{31} + vJ_{32} + w) / G^{\frac{1}{2}} \quad (2.55)$$

When the transformation from Cartesian coordinate to the coordinate along terrain is performed using above relation, the space differential of a various value  $\phi$  is transformed as follows,

$$\frac{\partial \phi}{\partial x} = \frac{1}{G^{\frac{1}{2}}} \left[ \frac{\partial}{\partial \xi} (J_d \phi) + \frac{\partial}{\partial \zeta} (J_{31} \phi) \right] \quad (2.56)$$

$$\frac{\partial \phi}{\partial y} = \frac{1}{G^{\frac{1}{2}}} \left[ \frac{\partial}{\partial \eta} (J_d \phi) + \frac{\partial}{\partial \zeta} (J_{32} \phi) \right] \quad (2.57)$$

$$\frac{\partial \phi}{\partial z} = \frac{1}{G^{\frac{1}{2}}} \frac{\partial \phi}{\partial \zeta} \quad (2.58)$$

### 2.2.3 The basic dynamical equations in terrain-following coordinates

In the coordinates of terrain-following, three factors of dependent variables are separated into the values of base state and deviations from the values of base state, as well as in those of terrain-excluding. Those three factors are potential temperature, pressure and density which considered effects of water contents and water vapor. The values of base state are defined so that those are in hydrostatic balance taken account of effects of terrain, which balance is

$$\frac{\partial \bar{p}}{\partial \zeta} = -G^{\frac{1}{2}} \bar{\rho} g \quad (2.59)$$

To simplify those representation, we use

$$\rho^* = G^{\frac{1}{2}} \bar{\rho} \quad (2.60)$$

and transform each predicted variable as follows:

$$u^* = \rho^* u \quad (2.61)$$

$$v^* = \rho^* v \quad (2.62)$$

$$w^* = \rho^* w \quad (2.63)$$

$$W^* = \rho^* W \quad (2.64)$$

$$\theta^* = \rho^* \theta' \quad (2.65)$$

$$q_v^* = \rho^* q_v \quad (2.66)$$

$$q_x^* = \rho^* q_x \quad (2.67)$$

Using the above, the basic equations system in terrain-excluding coordinates shown in Section 2.1 of this chapter are transformed in terrain-following coordinates as follows.

### Equation of motion

$$\begin{aligned} \frac{\partial u^*}{\partial t} = & - \left( u^* \frac{\partial u}{\partial \xi} + v^* \frac{\partial u}{\partial \eta} + W^* \frac{\partial u}{\partial \zeta} \right) \\ & - \left[ \frac{\partial}{\partial \xi} \{J_d(p' - \alpha Div^*)\} + \frac{\partial}{\partial \zeta} \{J_{31}(p' - \alpha Div^*)\} \right] + (f_s v^* - f_c w^*) + G^{\frac{1}{2}} \text{Turb}.u \end{aligned} \quad (2.68)$$

$$\begin{aligned} \frac{\partial v^*}{\partial t} = & - \left( u^* \frac{\partial v}{\partial \xi} + v^* \frac{\partial v}{\partial \eta} + W^* \frac{\partial v}{\partial \zeta} \right) \\ & - \left[ \frac{\partial}{\partial \eta} \{J_d(p' - \alpha Div^*)\} + \frac{\partial}{\partial \zeta} \{J_{32}(p' - \alpha Div^*)\} \right] - f_s u^* + G^{\frac{1}{2}} \text{Turb}.v \end{aligned} \quad (2.69)$$

$$\begin{aligned} \frac{\partial w^*}{\partial t} = & - \left( u^* \frac{\partial w}{\partial \xi} + v^* \frac{\partial w}{\partial \eta} + W^* \frac{\partial w}{\partial \zeta} \right) \\ & - \frac{\partial}{\partial \zeta} (p' - \alpha Div^*) - \rho^* \text{Buoy}.w + f_c u^* + G^{\frac{1}{2}} \text{Turb}.w \end{aligned} \quad (2.70)$$

where the buoyancy term  $\text{Buoy}.w$  is expressed as

$$\text{Buoy}.w = -g \frac{\rho'}{\bar{\rho}} = g \left( \frac{\theta'}{\bar{\theta}} - \frac{p'}{\bar{\rho} c_s^2} + \frac{q'_v}{\epsilon + \bar{q}_v} - \frac{q'_v + \sum q_x}{1 + \bar{q}_v} \right) \quad (2.71)$$

Here  $q'_v$  is not the deviation from the value of base state but the deviation from the initial value,  $\epsilon$  is the ratio of molecular weight to water vapor and dry air.  $c_s$  is the speed of sound in air given by

$$c_s = \sqrt{\gamma R_d \bar{T}}, \quad \gamma \equiv C_p / C_v \quad (2.72)$$

where  $g$  is the gravity acceleration,  $\bar{T}$  is the temperature of base state and  $R_d$  is the gas constant for dry air.  $C_p, C_v$  are the specific heat at constant pressure and the specific heat at constant volume for dry air, respectively.  $f_s, f_c$  are the Coriolis coefficients:

$$f_s = 2\omega \sin \varphi \quad (2.73)$$

$$f_c = 2\omega \cos \varphi \quad (2.74)$$

where  $\omega$  is the angular velocity of the earth and  $\varphi$  is the latitude. Furthermore,  $\alpha Div^*$  shown in the pressure term is the divergence damping to suppress soundwaves, which is given by

$$Div^* = \frac{1}{G^{\frac{1}{2}}} \left( \frac{\partial u^*}{\partial \xi} + \frac{\partial v^*}{\partial \eta} + \frac{\partial W^*}{\partial \zeta} \right) \quad (2.75)$$

### Equation of pressure

$$\begin{aligned} \frac{\partial G^{\frac{1}{2}} p'}{\partial t} = & - \left( G^{\frac{1}{2}} u \frac{\partial p'}{\partial \xi} + G^{\frac{1}{2}} v \frac{\partial p'}{\partial \eta} + G^{\frac{1}{2}} W \frac{\partial p'}{\partial \zeta} \right) + G^{\frac{1}{2}} \bar{\rho} g w \\ & - \bar{\rho} c_s^2 \left( \frac{\partial G^{\frac{1}{2}} u}{\partial \xi} + \frac{\partial G^{\frac{1}{2}} v}{\partial \eta} + \frac{\partial G^{\frac{1}{2}} W}{\partial \zeta} \right) + G^{\frac{1}{2}} \bar{\rho} c_s^2 \left( \frac{1}{\theta} \frac{d\theta}{dt} - \frac{1}{Q} \frac{dQ}{dt} \right) \end{aligned} \quad (2.76)$$

where  $Q = 1 + 0.61q_v + \sum q_x$  was used.

### Equation of potential temperature

$$\frac{\partial \theta^*}{\partial t} = - \left( u^* \frac{\partial \theta'}{\partial \xi} + v^* \frac{\partial \theta'}{\partial \eta} + W^* \frac{\partial \theta'}{\partial \zeta} \right) - \bar{\rho} w \frac{\partial \bar{\theta}}{\partial \zeta} + G^{\frac{1}{2}} \text{Turb.} \theta + \rho^* \text{Src.} \theta \quad (2.77)$$

### Equations of mixing ratio of water vapor and water contents

$$\frac{\partial q_v^*}{\partial t} = - \left( u^* \frac{\partial q_v}{\partial \xi} + v^* \frac{\partial q_v}{\partial \eta} + W^* \frac{\partial q_v}{\partial \zeta} \right) + G^{\frac{1}{2}} \text{Turb.} q_v + \rho^* \text{Src.} q_v \quad (2.78)$$

$$\frac{\partial q_x^*}{\partial t} = - \left( u^* \frac{\partial q_x}{\partial \xi} + v^* \frac{\partial q_x}{\partial \eta} + W^* \frac{\partial q_x}{\partial \zeta} \right) + G^{\frac{1}{2}} \text{Turb.} q_x + \rho^* \text{Src.} q_x + \rho^* \text{Fall.} q_x \quad (2.79)$$

### Equations of number concentration per unit volume

$$\begin{aligned} \frac{\partial G^{\frac{1}{2}} N_x}{\partial t} = & - \left[ u^* \frac{\partial}{\partial \xi} \left( \frac{N_x}{\bar{\rho}} \right) + v^* \frac{\partial}{\partial \eta} \left( \frac{N_x}{\bar{\rho}} \right) + W^* \frac{\partial}{\partial \zeta} \left( \frac{N_x}{\bar{\rho}} \right) \right] \\ & + G^{\frac{1}{2}} \text{Turb.} \frac{N_x}{\bar{\rho}} + \rho^* \text{Src.} \frac{N_x}{\bar{\rho}} + \rho^* \text{Fall.} \frac{N_x}{\bar{\rho}} \end{aligned} \quad (2.80)$$

Similar to terrain-excluding, there are prognostic equations of turbulence kinetic energy  $E$  in addition to these equations, which are discussed in Chapter ?? Diffusion of Sub-grid scale. So is the diffusion term of sub-grid scale  $\text{Turb.} \phi$  shown in those equations. On the other hand, the production or loss term  $\text{Src.} \phi$  and the falling term  $\text{Fall.} \phi$ , shown in the equations of potential temperature and water contents, are discussed in Chapter ?? Physical Processes of Clouds and Precipitations.

## 2.3 基本方程式系 — 地図投影

これまでの雲の数値モデルは地球の曲率を十分無視できる程度の領域で用いられることがほとんどであったので、地球の曲率の効果や地図投影によるゆがみの効果を考慮する必要はなかった。しかしながら、近年の並列コンピューターの大規模化・高速化とともに、雲を解像しながらでも、それらが無視できないほど広い領域での計算が可能になってきた。例えば、日本海を含む領域の計算や、台風を十分広い領域でシミュレーションしようとする、地球の曲率の効果は無視できなくなり、地図投影が必要になる。その効果を取り入れるために、基本方程式系に地図係数 (map factor) を導入する。

通常、気象学で用いる地図投影図法や緯度経度座標系は、直交曲線座標系の一つと考えることができ、地形を含まない場合にはデカルト座標から直交曲線座標系への変換を行えばよい。しかしながら、地形を含む場合は水平方向の座標は直交しているが、それらに対して鉛直座標はもはや直交しておらず、一般の曲線座標となる。

そこで、ここでは一般の曲線座標として座標変換のテンソルを計算するのではなく、簡単のために、まず直交曲線座標系における地形を考えない地図投影の方程式系を求めておいてから、それに地形の効果を取り入れる方法をとる。まずはじめに、直交曲線座標系を用いる場合の微分作用素について整理しておく。

### 2.3.1 直交曲線座標

第 2.2 節にあるように、直交直線座標系  $(x, y, z)$  の関数として空間のある領域で必要なだけ微分可能な 3 つの一価関数が定義されており、式 (2.27) のようにその関数行列式が 0 でない場合は、曲線座標が定義される。その特別な場合として、基底が作る計量テンソル  $G_{ij}$  の対角成分のみがゼロでない場合、その基底は直交しており、そのような曲線座標を直交曲線座標という。

緯度経度座標や気象学で通常用いる地図投影座標系は、2次元の直交曲線座標系とみなすことができ、座標変換のテンソル解析を用いなくてもベクトル解析の範囲で定式化ができる。地図投影は2次元であるが、ここでは便宜上水平座標に直交する鉛直座標  $z$  を含めて3次元の直交曲線座標を考える。ここでは一般的に直交曲線座標を考えるので、次のような座標系を定義する。

$$\xi = \xi(x, y, z) \quad (2.81)$$

$$\eta = \eta(x, y, z) \quad (2.82)$$

$$\zeta = \zeta(x, y, z) \quad (2.83)$$

さて、第 2.2 節では、正規でも直交でもない 1 次独立な基底 ( $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$ ) を導入したが、ここでは直交曲線座標を考えるので、これらは単位ベクトルでかつお互いに直交していると仮定する。すなわち、

$$\mathbf{e}_i \cdot \mathbf{e}_j = \delta_{ij} \quad (2.84)$$

である。ここでは、基底が直交している、反変成分と共変成分は同じになり区別する必要はない。このとき、デカルト座標での線元素  $ds^2 = dx^2 + dy^2 + dz^2$  は、

$$ds^2 = (h_1 d\xi)^2 + (h_2 d\eta)^2 + (h_3 d\zeta)^2 \quad (2.85)$$

と与えられる。ここで、 $h_i$  はメトリック係数 (第 1 基本量) とよばれ、

$$h_1 = \left[ \left( \frac{\partial x}{\partial \xi} \right)^2 + \left( \frac{\partial y}{\partial \xi} \right)^2 + \left( \frac{\partial z}{\partial \xi} \right)^2 \right]^{\frac{1}{2}} \quad (2.86)$$

$$h_2 = \left[ \left( \frac{\partial x}{\partial \eta} \right)^2 + \left( \frac{\partial y}{\partial \eta} \right)^2 + \left( \frac{\partial z}{\partial \eta} \right)^2 \right]^{\frac{1}{2}} \quad (2.87)$$

$$h_3 = \left[ \left( \frac{\partial x}{\partial \zeta} \right)^2 + \left( \frac{\partial y}{\partial \zeta} \right)^2 + \left( \frac{\partial z}{\partial \zeta} \right)^2 \right]^{\frac{1}{2}} \quad (2.88)$$

と与えられるので、 $\xi$  曲線、 $\eta$  曲線、 $\zeta$  曲線の弧長  $s_1, s_2, s_3$  は、それぞれ、

$$ds_1 = h_1 d\xi \quad (2.89)$$

$$ds_2 = h_2 d\eta \quad (2.90)$$

$$ds_3 = h_3 d\zeta \quad (2.91)$$

と表わされる。

これらを用いて、直交曲線座標  $(\xi, \eta, \zeta)$  における勾配、発散、回転などを与えることができる。まず、任意のスカラー関数を  $\phi(\xi, \eta, \zeta)$  とすると、直交曲線座標における勾配は、

$$\nabla \phi = \frac{1}{h_1} \frac{\partial \phi}{\partial \xi} \mathbf{e}_1 + \frac{1}{h_2} \frac{\partial \phi}{\partial \eta} \mathbf{e}_2 + \frac{1}{h_3} \frac{\partial \phi}{\partial \zeta} \mathbf{e}_3 \quad (2.92)$$

と与えられる。ただし、 $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$  は、直交曲線座標  $(\xi, \eta, \zeta)$  についての正規直交基底（単位接ベクトル）である。

さて、一般に、直交曲線座標の場合、任意のベクトル  $\mathbf{A} = A^\xi \mathbf{e}_1 + A^\eta \mathbf{e}_2 + A^\zeta \mathbf{e}_3$  の座標成分  $(A^\xi, A^\eta, A^\zeta)$  だけでなく、その単位接ベクトル  $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$  も座標  $(\xi, \eta, \zeta)$  の関数であるので、ベクトル量  $\mathbf{A}$  に微分演算子を作用させるときは単位接ベクトルも微分されなければならない、例えば、

$$\frac{\partial A^\xi \mathbf{e}_1}{\partial \eta} = \mathbf{e}_1 \frac{\partial A^\xi}{\partial \eta} + A^\xi \frac{\partial \mathbf{e}_1}{\partial \eta} \quad (2.93)$$

のようになる。また、単位接ベクトルの 9 個の微分  $\frac{\partial \mathbf{e}_i}{\partial \xi_i}$  は、それぞれ、

$$\begin{aligned} \frac{\partial \mathbf{e}_1}{\partial \xi} &= -\frac{1}{h_2} \frac{\partial h_1}{\partial \eta} \mathbf{e}_2 - \frac{1}{h_3} \frac{\partial h_1}{\partial \zeta} \mathbf{e}_3, & \frac{\partial \mathbf{e}_1}{\partial \eta} &= \frac{1}{h_1} \frac{\partial h_2}{\partial \xi} \mathbf{e}_2, & \frac{\partial \mathbf{e}_1}{\partial \zeta} &= \frac{1}{h_1} \frac{\partial h_3}{\partial \xi} \mathbf{e}_3, \\ \frac{\partial \mathbf{e}_2}{\partial \xi} &= \frac{1}{h_2} \frac{\partial h_1}{\partial \eta} \mathbf{e}_1, & \frac{\partial \mathbf{e}_2}{\partial \eta} &= -\frac{1}{h_3} \frac{\partial h_2}{\partial \zeta} \mathbf{e}_3 - \frac{1}{h_1} \frac{\partial h_2}{\partial \xi} \mathbf{e}_1, & \frac{\partial \mathbf{e}_2}{\partial \zeta} &= \frac{1}{h_2} \frac{\partial h_3}{\partial \eta} \mathbf{e}_3, \\ \frac{\partial \mathbf{e}_3}{\partial \xi} &= \frac{1}{h_3} \frac{\partial h_1}{\partial \zeta} \mathbf{e}_1, & \frac{\partial \mathbf{e}_3}{\partial \eta} &= \frac{1}{h_3} \frac{\partial h_2}{\partial \zeta} \mathbf{e}_2, & \frac{\partial \mathbf{e}_3}{\partial \zeta} &= -\frac{1}{h_1} \frac{\partial h_3}{\partial \xi} \mathbf{e}_1 - \frac{1}{h_2} \frac{\partial h_3}{\partial \eta} \mathbf{e}_2 \\ & & & & \dots \dots & \end{aligned} \quad (2.94)$$

と与えられる。よって、任意のベクトル  $\mathbf{A}$  の直交曲線座標における発散は、

$$\nabla \cdot \mathbf{A} = \frac{1}{h_1 h_2 h_3} \left[ \frac{\partial}{\partial \xi} (h_2 h_3 A^\xi) + \frac{\partial}{\partial \eta} (h_3 h_1 A^\eta) + \frac{\partial}{\partial \zeta} (h_1 h_2 A^\zeta) \right] \quad (2.95)$$

のように、また、回転は次のように与えられる。

$$\begin{aligned} \nabla \times \mathbf{A} &= \begin{vmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \\ h_2 h_3 & h_3 h_1 & h_1 h_2 \\ \frac{\partial}{\partial \xi} & \frac{\partial}{\partial \eta} & \frac{\partial}{\partial \zeta} \\ h_1 A^\xi & h_2 A^\eta & h_3 A^\zeta \end{vmatrix} \\ &= \frac{\mathbf{e}_1}{h_2 h_3} \left[ \frac{\partial (h_3 A^\zeta)}{\partial \eta} - \frac{\partial (h_2 A^\eta)}{\partial \zeta} \right] + \frac{\mathbf{e}_2}{h_3 h_1} \left[ \frac{\partial (h_1 A^\xi)}{\partial \zeta} - \frac{\partial (h_3 A^\zeta)}{\partial \xi} \right] + \frac{\mathbf{e}_3}{h_1 h_2} \left[ \frac{\partial (h_2 A^\eta)}{\partial \xi} - \frac{\partial (h_1 A^\xi)}{\partial \eta} \right] \\ &\quad \dots\dots (2.96) \end{aligned}$$

最後に、任意のスカラー関数  $\phi(\xi, \eta, \zeta)$  のラプラシアンは、

$$\nabla^2 \phi = \frac{1}{h_1 h_2 h_3} \left[ \frac{\partial}{\partial \xi} \left( \frac{h_2 h_3}{h_1} \frac{\partial \phi}{\partial \xi} \right) + \frac{\partial}{\partial \eta} \left( \frac{h_3 h_1}{h_2} \frac{\partial \phi}{\partial \eta} \right) + \frac{\partial}{\partial \zeta} \left( \frac{h_1 h_2}{h_3} \frac{\partial \phi}{\partial \zeta} \right) \right] \quad (2.97)$$

と与えられる。

### 2.3.2 地図投影座標系における基本方程式系 — 地形を含まない場合

地球上の各点は、緯度経度座標や地図投影上の点と1対1の対応関係を与えることができるので、曲線座標とみなすことができる。さらに、緯度経度座標や気象学で用いる地図投影は、水平方向の座標はお互いに直交しており、鉛直方向を第3の座標とすると、地形を含まない基礎方程式系の場合には、これらは3次元の直交曲線座標として扱うことができる。ここでは、水平方向の座標を  $(\xi, \eta)$ 、鉛直方向の座標を  $z$ 、それぞれの単位接ベクトルを  $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$  とし、地図投影座標系における基礎方程式を求める。

まず、メトリック係数（拡大因数）を、

$$h_1 = \frac{1}{m} \quad (2.98)$$

$$h_2 = \frac{1}{n} \quad (2.99)$$

$$h_3 = 1 \quad (2.100)$$

としておく。さて、 $m, n$  は次の節で地図係数として扱われるが、ここでは、一般性を持たせるために拡大因数としておく。例えば、球面座標であれば、

$$m = \frac{1}{a \cos \phi} \quad (2.101)$$

$$n = \frac{1}{a} \quad (2.102)$$

である。ただし、 $a$  は地球の半径、 $\phi$  は緯度である。拡大因数がこれらの形をとると、後に求める拡大因数を含む方程式系は球面の方程式系に帰着できる。

さて、このとき、ある点Pとその近傍の座標を  $(\xi, \eta, z)$  と  $(\xi + d\xi, \eta + d\eta, z + dz)$  とすると、各座標軸上の距離  $ds_1, ds_2, ds_3$  は、

$$ds_1 = \frac{d\xi}{m} \quad (2.103)$$

$$ds_2 = \frac{d\eta}{n} \quad (2.104)$$

$$ds_3 = dz \quad (2.105)$$

また、単位接ベクトルの座標軸方向の微分は、

$$\begin{aligned} \frac{\partial \mathbf{e}_1}{\partial \xi} &= -n \frac{\partial}{\partial \eta} \left( \frac{1}{m} \right) \mathbf{e}_2 - \frac{1}{am} \mathbf{e}_3, & \frac{\partial \mathbf{e}_1}{\partial \eta} &= m \frac{\partial}{\partial \xi} \left( \frac{1}{n} \right) \mathbf{e}_2, & \frac{\partial \mathbf{e}_1}{\partial z} &= 0, \\ \frac{\partial \mathbf{e}_2}{\partial \xi} &= n \frac{\partial}{\partial \eta} \left( \frac{1}{m} \right) \mathbf{e}_1, & \frac{\partial \mathbf{e}_2}{\partial \eta} &= -\frac{1}{an} \mathbf{e}_3 - m \frac{\partial}{\partial \xi} \left( \frac{1}{n} \right) \mathbf{e}_1, & \frac{\partial \mathbf{e}_2}{\partial z} &= 0, \\ \frac{\partial \mathbf{e}_3}{\partial \xi} &= \frac{1}{am} \mathbf{e}_1, & \frac{\partial \mathbf{e}_3}{\partial \eta} &= \frac{1}{an} \mathbf{e}_2, & \frac{\partial \mathbf{e}_3}{\partial z} &= 0 \end{aligned} \quad (2.106)$$

のようになる。以下ではこれらを用いて、運動方程式、熱力学方程式、圧縮系の連続方程式、水蒸気混合比の式、雲・降水粒子の混合比の式、および、雲・降水粒子の数密度の式を書き換える。

まず、運動方程式について考える。地形を含まない場合の運動方程式 (2.6)~(2.8) をこの直交座標系  $(\xi, \eta, z)$  で表すことが、この目標である。

速度ベクトル  $\mathbf{u}$  を  $(\xi, \eta, z)$  の座標で表すと、

$$\mathbf{u} = u\mathbf{e}_1 + v\mathbf{e}_2 + w\mathbf{e}_3 \quad (2.107)$$

であるが、(2.103)~(2.105) を考慮すると速度の各成分は、次のようになる。

$$u = \frac{ds_1}{dt} = \frac{1}{m} \frac{d\xi}{dt} \quad (2.108)$$

$$v = \frac{ds_2}{dt} = \frac{1}{n} \frac{d\eta}{dt} \quad (2.109)$$

$$w = \frac{ds_3}{dt} = \frac{dz}{dt} \quad (2.110)$$



地図投影における全微分は、

$$\frac{d}{dt} = \frac{\partial}{\partial t} + \frac{d\xi}{dt} \frac{\partial}{\partial \xi} + \frac{d\eta}{dt} \frac{\partial}{\partial \eta} + \frac{dz}{dt} \frac{\partial}{\partial z} \quad (2.111)$$

であり、式 (2.108)~(2.110) を考慮すると、

$$\frac{d}{dt} = \frac{\partial}{\partial t} + mu \frac{\partial}{\partial \xi} + nv \frac{\partial}{\partial \eta} + w \frac{\partial}{\partial z} \quad (2.112)$$

となる。よって、曲線座標系における速度 (2.107) の時間微分は、

$$\frac{d\mathbf{u}}{dt} = \frac{du}{dt} \mathbf{e}_1 + \frac{dv}{dt} \mathbf{e}_2 + \frac{dw}{dt} \mathbf{e}_3 + u \frac{d\mathbf{e}_1}{dt} + v \frac{d\mathbf{e}_2}{dt} + w \frac{d\mathbf{e}_3}{dt} \quad (2.113)$$

となり、単位接ベクトルの時間微分を (2.106) および (2.112) を考慮すると、(2.113) の右辺後半の単位接ベクトルの時間微分の項は、

$$\begin{aligned} u \frac{d\mathbf{e}_1}{dt} + v \frac{d\mathbf{e}_2}{dt} + w \frac{d\mathbf{e}_3}{dt} &= -\mathbf{e}_1 mnv \left[ v \frac{\partial}{\partial \xi} \left( \frac{1}{n} \right) - u \frac{\partial}{\partial \eta} \left( \frac{1}{m} \right) \right] + \mathbf{e}_1 \frac{uw}{a} \\ &\quad + \mathbf{e}_2 mnu \left[ v \frac{\partial}{\partial \xi} \left( \frac{1}{n} \right) - u \frac{\partial}{\partial \eta} \left( \frac{1}{m} \right) \right] + \mathbf{e}_1 \frac{vw}{a} - \mathbf{e}_3 \frac{u^2 + v^2}{a} \end{aligned} \quad (2.114)$$

のようになる。これらの項は座標系が直線でなく、単位ベクトルが場所によって変化することによって現れる項で、曲率項あるいはメトリック項と呼ばれる。

次に、コリオリ力の項は、 $2\boldsymbol{\Omega}$  の座標系  $(\xi, \eta, z)$  の各成分を  $(f_\xi, f_\eta, f_z)$  で表すと、

$$\begin{aligned} 2\boldsymbol{\Omega} \times \mathbf{u} &= \begin{vmatrix} \mathbf{e}_1 & \mathbf{e}_2 & \mathbf{e}_3 \\ 2\Omega_\xi & 2\Omega_\eta & 2\Omega_z \\ u & v & w \end{vmatrix} \\ &= \mathbf{e}_1 (f_\eta w - f_z v) + \mathbf{e}_2 (f_z u - f_\xi w) + \mathbf{e}_3 (f_\xi v - f_\eta u) \end{aligned} \quad (2.115)$$

また、気圧傾度力は (2.92) より、

$$\nabla p' = m \frac{\partial p'}{\partial \xi} \mathbf{e}_1 + n \frac{\partial p'}{\partial \eta} \mathbf{e}_2 + \frac{\partial p'}{\partial z} \mathbf{e}_3 \quad (2.116)$$

と与えられる。

これらを用いると、運動方程式 (2.6)~(2.8) は、以下ようになる。

運動方程式

$$\bar{\rho} \frac{\partial u}{\partial t} = -\bar{\rho} \left( mu \frac{\partial u}{\partial \xi} + nv \frac{\partial u}{\partial \eta} + w \frac{\partial u}{\partial z} \right)$$

$$-m \frac{\partial p'}{\partial \xi} + \bar{\rho}(f_\eta w - f_z v) + \bar{\rho} m n v \left[ v \frac{\partial}{\partial \xi} \left( \frac{1}{n} \right) - u \frac{\partial}{\partial \eta} \left( \frac{1}{m} \right) \right] - \bar{\rho} \frac{uw}{a} + \text{Turb}.u \quad (2.117)$$

$$\begin{aligned} \bar{\rho} \frac{\partial v}{\partial t} &= -\bar{\rho} \left( mu \frac{\partial v}{\partial \xi} + nv \frac{\partial v}{\partial \eta} + w \frac{\partial v}{\partial z} \right) \\ &\quad -n \frac{\partial p'}{\partial \eta} + \bar{\rho}(f_z u - f_\xi w) - \bar{\rho} m n u \left[ v \frac{\partial}{\partial \xi} \left( \frac{1}{n} \right) - u \frac{\partial}{\partial \eta} \left( \frac{1}{m} \right) \right] - \bar{\rho} \frac{vw}{a} + \text{Turb}.v \end{aligned} \quad (2.118)$$

$$\begin{aligned} \bar{\rho} \frac{\partial w}{\partial t} &= -\bar{\rho} \left( mu \frac{\partial w}{\partial \xi} + nv \frac{\partial w}{\partial \eta} + w \frac{\partial w}{\partial z} \right) \\ &\quad - \frac{\partial p'}{\partial z} - \bar{\rho} \text{Buoy}.w + \bar{\rho}(f_\xi v - f_\eta u) + \bar{\rho} \frac{u^2 + v^2}{a} + \text{Turb}.w \end{aligned} \quad (2.119)$$

ここで、 $\bar{\rho} = \bar{\rho}(z)$  は基本場の密度で、鉛直座標のみの関数である。また、 $\text{Buoy}.w$  は浮力項である。

同様にして、気圧偏差の式 (2.9)、温位偏差の式 (2.10)、および、水蒸気や雲物理に関する式 (2.11)~(2.13) を書きかえると、以下ようになる。

気圧偏差の方程式

$$\begin{aligned} \frac{\partial p'}{\partial t} &= - \left( mu \frac{\partial p'}{\partial \xi} + nv \frac{\partial p'}{\partial \eta} + w \frac{\partial p'}{\partial z} \right) + \bar{\rho} g w \\ &\quad - \bar{\rho} c_s^2 \left[ mn \left( \frac{\partial}{\partial \xi} \frac{u}{m} + \frac{\partial}{\partial \eta} \frac{v}{n} \right) + \frac{\partial w}{\partial z} \right] + \bar{\rho} c_s^2 \left( \frac{1}{\theta} \frac{d\theta}{dt} - \frac{1}{Q} \frac{dQ}{dt} \right) \end{aligned} \quad (2.120)$$

温位偏差の方程式

$$\bar{\rho} \frac{\partial \theta'}{\partial t} = -\bar{\rho} \left( mu \frac{\partial \theta'}{\partial \xi} + nv \frac{\partial \theta'}{\partial \eta} + w \frac{\partial \theta'}{\partial z} \right) - \bar{\rho} w \frac{\partial \bar{\theta}}{\partial z} + \bar{\rho} \text{Src}.\theta + \text{Turb}.\theta \quad (2.121)$$

水蒸気および水物質の混合比の方程式

$$\bar{\rho} \frac{\partial q_v}{\partial t} = -\bar{\rho} \left( mu \frac{\partial q_v}{\partial \xi} + nv \frac{\partial q_v}{\partial \eta} + w \frac{\partial q_v}{\partial z} \right) + \bar{\rho} \text{Src}.q_v + \text{Turb}.q_v \quad (2.122)$$

$$\bar{\rho} \frac{\partial q_x}{\partial t} = -\bar{\rho} \left( mu \frac{\partial q_x}{\partial \xi} + nv \frac{\partial q_x}{\partial \eta} + w \frac{\partial q_x}{\partial z} \right) + \bar{\rho} \text{Fall}.q_x + \bar{\rho} \text{Src}.q_x + \text{Turb}.q_x \quad (2.123)$$

水物質の数密度の方程式

$$\begin{aligned} \frac{\partial N_x}{\partial t} = & -\bar{\rho} \left[ mu \frac{\partial}{\partial \xi} \left( \frac{N_x}{\bar{\rho}} \right) + nv \frac{\partial}{\partial \eta} \left( \frac{N_x}{\bar{\rho}} \right) + w \frac{\partial}{\partial z} \left( \frac{N_x}{\bar{\rho}} \right) \right] \\ & + \bar{\rho} \text{Src.} \frac{N_x}{\bar{\rho}} + \bar{\rho} \text{Fall.} \frac{N_x}{\bar{\rho}} + \text{Turb.} \frac{N_x}{\bar{\rho}} \end{aligned} \quad (2.124)$$

なお、この他に乱流に関わるものとして、乱流運動エネルギーの時間発展方程式と各式に現れる拡散項  $\text{Turb.}\phi$  も変更されなければならないが、これについては第??章「サブグリッドスケールの拡散」で述べる。

### 2.3.3 地図投影座標系における基本方程式系 — 地形を含む場合

前節までで、地図係数を入れた基本方程式系を求める準備ができたので、本節では、地図係数を含む地形に沿う座標系の基本方程式を求める。先に述べたように地形に沿う座標系を鉛直座標として選ぶと、その座標系はもはや直交座標系にはならず、一般の曲線座標系になる。しかしながら、地図係数は鉛直方向の座標の関数ではないので、ここでは前節で求めた直交曲線座標系に、第2.2節の地形の導入と同じ手続きで地形に沿う座標系に直すことができる。

さて、前節までは、 $m, n$  は拡大係数として扱ってきたが、ここでは通常気象学で用いられる地図投影の地図係数 (map factor) と考える。気象学で用いられる地図投影法は、平射図法 (ステレオ図法)、ランベルト正角円錐図法、正角円筒図法 (メルカトル図法) などの正角図法である。正角図法とは「角が正しい」または「形が正しい」もので、水平方向の2つの座標方向のメトリック係数が地図上のすべての点において等しいものである。すなわち、

$$\left( h_1 = \frac{1}{m} \right) = \left( h_2 = \frac{1}{n} \right) \quad (2.125)$$

である。一般には、これらの地図係数は緯度と経度の関数である。

この座標系での全微分は、(2.108)~(2.110) を考慮すると、

$$\begin{aligned} \frac{d}{dt} &= \frac{\partial}{\partial t} + \frac{d\xi}{dt} \frac{\partial}{\partial \xi} + \frac{d\eta}{dt} \frac{\partial}{\partial \eta} + \frac{d\zeta}{dt} \frac{\partial}{\partial \zeta} \\ &= \frac{\partial}{\partial t} + mu \frac{\partial}{\partial \xi} + mv \frac{\partial}{\partial \eta} + W \frac{\partial}{\partial \zeta} \end{aligned} \quad (2.126)$$

となり、鉛直速度  $W$  は、

$$\begin{aligned} W = \frac{d\zeta}{dt} &= mu \frac{\partial \zeta}{\partial \xi} + mv \frac{\partial \zeta}{\partial \eta} + w \frac{\partial \zeta}{\partial z} \\ &= \left[ mu \left( -\frac{\partial z}{\partial \xi} \right) + mv \left( -\frac{\partial z}{\partial \eta} \right) + w \frac{\partial \zeta}{\partial z} \right] \frac{\partial \zeta}{\partial z} \\ &= \frac{1}{G^{\frac{1}{2}}} (mu J_{31} + mv J_{32} + w) \end{aligned} \quad (2.127)$$

のようになる。また、ある変数  $\phi$  の空間微分は、

$$m \frac{\partial \phi}{\partial \xi} \rightarrow m \frac{1}{G^{\frac{1}{2}}} \left[ \frac{\partial}{\partial \xi} (J_d \phi) + \frac{\partial}{\partial \zeta} (J_{31} \phi) \right] \quad (2.128)$$

$$m \frac{\partial \phi}{\partial \eta} \rightarrow m \frac{1}{G^{\frac{1}{2}}} \left[ \frac{\partial}{\partial \eta} (J_d \phi) + \frac{\partial}{\partial \zeta} (J_{32} \phi) \right] \quad (2.129)$$

$$\frac{\partial \phi}{\partial z} \rightarrow \frac{1}{G^{\frac{1}{2}}} \frac{\partial \phi}{\partial \zeta} \quad (2.130)$$

のように変換される。

これらを用いて方程式系を書き換えると、以下のように与えられる。

運動方程式

$$\begin{aligned} \bar{\rho} \frac{\partial u}{\partial t} = & -\bar{\rho} \left( mu \frac{\partial u}{\partial \xi} + mv \frac{\partial u}{\partial \eta} + W \frac{\partial u}{\partial \zeta} \right) - m \frac{1}{G^{\frac{1}{2}}} \left[ \frac{\partial}{\partial \xi} (J_d p') + \frac{\partial}{\partial \zeta} (J_{31} p') \right] \\ & + \bar{\rho} (f_\eta w - f_z v) + \bar{\rho} m^2 v \left[ v \frac{\partial}{\partial \xi} \left( \frac{1}{m} \right) - u \frac{\partial}{\partial \eta} \left( \frac{1}{m} \right) \right] - \bar{\rho} \frac{uw}{a} + \text{Turb.}u \end{aligned} \quad (2.131)$$

$$\begin{aligned} \bar{\rho} \frac{\partial v}{\partial t} = & -\bar{\rho} \left( mu \frac{\partial v}{\partial \xi} + mv \frac{\partial v}{\partial \eta} + W \frac{\partial v}{\partial \zeta} \right) - m \frac{1}{G^{\frac{1}{2}}} \left[ \frac{\partial}{\partial \eta} (J_d p') + \frac{\partial}{\partial \zeta} (J_{32} p') \right] \\ & + \bar{\rho} (f_z u - f_\xi w) - \bar{\rho} m^2 u \left[ v \frac{\partial}{\partial \xi} \left( \frac{1}{m} \right) - u \frac{\partial}{\partial \eta} \left( \frac{1}{m} \right) \right] - \bar{\rho} \frac{vw}{a} + \text{Turb.}v \end{aligned} \quad (2.132)$$

$$\begin{aligned} \bar{\rho} \frac{\partial w}{\partial t} = & -\bar{\rho} \left( mu \frac{\partial w}{\partial \xi} + mv \frac{\partial w}{\partial \eta} + W \frac{\partial w}{\partial \zeta} \right) \\ & - \frac{1}{G^{\frac{1}{2}}} \frac{\partial p'}{\partial \zeta} - \bar{\rho} \text{Buoy.}w + \bar{\rho} (f_\xi v - f_\eta u) + \bar{\rho} \frac{u^2 + v^2}{a} + \text{Turb.}w \end{aligned} \quad (2.133)$$

気圧偏差の方程式

$$\begin{aligned} \frac{\partial p'}{\partial t} = & - \left( mu \frac{\partial p'}{\partial \xi} + mv \frac{\partial p'}{\partial \eta} + W \frac{\partial p'}{\partial \zeta} \right) + \bar{\rho} g w \\ & - \bar{\rho} c_s^2 \frac{1}{G^{\frac{1}{2}}} \left[ m^2 \left( \frac{\partial}{\partial \xi} \frac{G^{\frac{1}{2}} u}{m} + \frac{\partial}{\partial \eta} \frac{G^{\frac{1}{2}} v}{m} \right) + \frac{\partial G^{\frac{1}{2}} w}{\partial \zeta} \right] + \bar{\rho} c_s^2 \left( \frac{1}{\theta} \frac{d\theta}{dt} - \frac{1}{Q} \frac{dQ}{dt} \right) \end{aligned} \quad (2.134)$$

温位偏差の方程式

$$\bar{\rho} \frac{\partial \theta'}{\partial t} = -\bar{\rho} \left( mu \frac{\partial \theta'}{\partial \xi} + mv \frac{\partial \theta'}{\partial \eta} + W \frac{\partial \theta'}{\partial \zeta} \right) - \bar{\rho} w \frac{1}{G^{\frac{1}{2}}} \frac{\partial \bar{\theta}}{\partial \zeta} + \bar{\rho} \text{Src.} \theta + \text{Turb.} \theta \quad (2.135)$$

水蒸気および水物質の混合比の方程式

$$\bar{\rho} \frac{\partial q_v}{\partial t} = -\bar{\rho} \left( mu \frac{\partial q_v}{\partial \xi} + mv \frac{\partial q_v}{\partial \eta} + W \frac{\partial q_v}{\partial \zeta} \right) + \bar{\rho} \text{Src.} q_v + \text{Turb.} q_v \quad (2.136)$$

$$\bar{\rho} \frac{\partial q_x}{\partial t} = -\bar{\rho} \left( mu \frac{\partial q_x}{\partial \xi} + mv \frac{\partial q_x}{\partial \eta} + W \frac{\partial q_x}{\partial \zeta} \right) + \bar{\rho} \text{Fall.} q_x + \bar{\rho} \text{Src.} q_x + \text{Turb.} q_x \quad (2.137)$$

水物質の数密度の方程式

$$\begin{aligned} \frac{\partial N_x}{\partial t} = & -\bar{\rho} \left[ mu \frac{\partial}{\partial \xi} \left( \frac{N_x}{\bar{\rho}} \right) + mv \frac{\partial}{\partial \eta} \left( \frac{N_x}{\bar{\rho}} \right) + W \frac{\partial}{\partial \zeta} \left( \frac{N_x}{\bar{\rho}} \right) \right] \\ & + \bar{\rho} \text{Src.} \frac{N_x}{\bar{\rho}} + \bar{\rho} \text{Fall.} \frac{N_x}{\bar{\rho}} + \text{Turb.} \frac{N_x}{\bar{\rho}} \end{aligned} \quad (2.138)$$

以上で地形に沿う地図投影座標の方程式系が求められたが、これらの表記を簡便にするために、アスタリスクの付いた変数(??)~(??)を用い、また、

$$m^2 \left[ v \frac{\partial}{\partial \xi} \left( \frac{1}{m} \right) - u \frac{\partial}{\partial \eta} \left( \frac{1}{m} \right) \right] = u \frac{\partial m}{\partial \eta} - v \frac{\partial m}{\partial \xi} \quad (2.139)$$

であることに注意して、これらを書き換えて形を整えると、以下のようなになる。

運動方程式

$$\begin{aligned} \frac{\partial u^*}{\partial t} = & - \left( mu^* \frac{\partial u}{\partial \xi} + mv^* \frac{\partial u}{\partial \eta} + W^* \frac{\partial u}{\partial \zeta} \right) - m \left[ \frac{\partial}{\partial \xi} (J_d p') + \frac{\partial}{\partial \zeta} (J_{31} p') \right] \\ & + (f_\eta w^* - f_z v^*) + v^* \left[ u \frac{\partial m}{\partial \eta} - v \frac{\partial m}{\partial \xi} \right] - u^* \frac{w}{a} + G^{\frac{1}{2}} \text{Turb.} u \end{aligned} \quad (2.140)$$

$$\begin{aligned} \frac{\partial v^*}{\partial t} = & - \left( mu^* \frac{\partial v}{\partial \xi} + mv^* \frac{\partial v}{\partial \eta} + W^* \frac{\partial v}{\partial \zeta} \right) - m \left[ \frac{\partial}{\partial \eta} (J_d p') + \frac{\partial}{\partial \zeta} (J_{32} p') \right] \\ & + (f_z u^* - f_\xi w^*) - u^* \left[ u \frac{\partial m}{\partial \eta} - v \frac{\partial m}{\partial \xi} \right] - v^* \frac{w}{a} + G^{\frac{1}{2}} \text{Turb.} v \end{aligned} \quad (2.141)$$

$$\begin{aligned} \frac{\partial w^*}{\partial t} = & - \left( mu^* \frac{\partial w}{\partial \xi} + mv^* \frac{\partial w}{\partial \eta} + W^* \frac{\partial w}{\partial \zeta} \right) \\ & - \frac{\partial p'}{\partial \zeta} - \rho^* \text{Buoy.} w + (f_\xi v^* - f_\eta u^*) + \frac{u^* u + v^* v}{a} + G^{\frac{1}{2}} \text{Turb.} w \end{aligned} \quad (2.142)$$

気圧偏差の方程式

$$\begin{aligned} \frac{\partial G^{\frac{1}{2}} p'}{\partial t} = & -G^{\frac{1}{2}} \left( mu \frac{\partial p'}{\partial \xi} + mv \frac{\partial p'}{\partial \eta} + W \frac{\partial p'}{\partial \zeta} \right) + gw^* \\ & - \bar{\rho} c_s^2 \left[ m^2 \left( \frac{\partial}{\partial \xi} \frac{G^{\frac{1}{2}} u}{m} + \frac{\partial}{\partial \eta} \frac{G^{\frac{1}{2}} v}{m} \right) + \frac{\partial G^{\frac{1}{2}} w}{\partial \zeta} \right] + G^{\frac{1}{2}} \bar{\rho} c_s^2 \left( \frac{1}{\theta} \frac{d\theta}{dt} - \frac{1}{Q} \frac{dQ}{dt} \right) \end{aligned} \quad (2.143)$$

温位偏差の方程式

$$\frac{\partial \theta^*}{\partial t} = - \left( mu^* \frac{\partial \theta'}{\partial \xi} + mv^* \frac{\partial \theta'}{\partial \eta} + W^* \frac{\partial \theta'}{\partial \zeta} \right) - \bar{\rho} w \frac{\partial \bar{\theta}}{\partial \zeta} + \rho^* \text{Src.} \theta + G^{\frac{1}{2}} \text{Turb.} \theta \quad (2.144)$$

水蒸気および水物質の混合比の方程式

$$\frac{\partial q_v^*}{\partial t} = - \left( mu^* \frac{\partial q_v}{\partial \xi} + mv^* \frac{\partial q_v}{\partial \eta} + W^* \frac{\partial q_v}{\partial \zeta} \right) + \rho^* \text{Src.} q_v + G^{\frac{1}{2}} \text{Turb.} q_v \quad (2.145)$$

$$\frac{\partial q_x^*}{\partial t} = - \left( mu^* \frac{\partial q_x}{\partial \xi} + mv^* \frac{\partial q_x}{\partial \eta} + W^* \frac{\partial q_x}{\partial \zeta} \right) + \rho^* \text{Fall.} q_x + \rho^* \text{Src.} q_x + G^{\frac{1}{2}} \text{Turb.} q_x \quad (2.146)$$

水物質の数密度の方程式

$$\begin{aligned} \frac{\partial G^{\frac{1}{2}} N_x}{\partial t} = & - \left[ mu^* \frac{\partial}{\partial \xi} \left( \frac{N_x}{\bar{\rho}} \right) + mv^* \frac{\partial}{\partial \eta} \left( \frac{N_x}{\bar{\rho}} \right) + W^* \frac{\partial}{\partial \zeta} \left( \frac{N_x}{\bar{\rho}} \right) \right] \\ & + \rho^* \text{Src.} \frac{N_x}{\bar{\rho}} + \rho^* \text{Fall.} \frac{N_x}{\bar{\rho}} + G^{\frac{1}{2}} \text{Turb.} \frac{N_x}{\bar{\rho}} \end{aligned} \quad (2.147)$$

なお、実際の計算では音波による計算不安定を抑えるため、気圧偏差  $p'$  は人工的に入れた音波の減衰項  $\alpha \text{Div}^*$  を含む、 $p' - \alpha \text{Div}^*$  で置き換えられる。この項の地図投影座標系における表記は、次のように与えられる。

$$\text{Div}^* = \frac{1}{G^{\frac{1}{2}}} \left[ m^2 \left( \frac{\partial u^*}{\partial \xi} \frac{1}{m} + \frac{\partial v^*}{\partial \eta} \frac{1}{m} \right) + \frac{\partial W^*}{\partial \zeta} \right] \quad (2.148)$$

前節と同様に、乱流運動エネルギーの時間発展方程式と各式に現れる拡散項  $\text{Turb.} \phi$  の変換については、第??章「サブグリッドスケールの拡散」で述べる。

### 2.3.4 正角投影図法

先にも述べたとおり、一般に領域気象モデルで採用される地図投影は、正角投影法である。それは、拡大係数  $m, n$  の方向性を考えなくてもよくなり、基礎方程式系への導入が容易になるからである。

しかしながら、投影方法により歪の分布は異なるので、地球のどの一部分を切り出しても計算できるようにするためには、種々の投影法を採用しなければならない。**CRess** では、計算領域の緯度分布に対して次の3種類の投影図法を採用しており、ここでは、それらの詳細について述べる。

- 平射図法（ステレオ図法）：高緯度における計算に用いる。
- ランベルト正角円錐図法：中緯度における計算に用いる。
- 正角円筒図法（メルカトル図法）：低緯度における計算に用いる。

平射図法（ステレオ図法）

図より、緯度・経度方向のそれぞれの拡大率は、 $p = \frac{\pi}{2} - \phi$  を用いて、

$$m_\lambda = \frac{1}{a} \frac{\delta r}{\delta p} \quad (2.149)$$

$$n_\phi = \frac{r \lambda}{a \lambda \sin p} = \frac{r}{a \sin p} \quad (2.150)$$

であるが、この図法は正角投影であり、 $m_\lambda = n_\phi$  でなければならないので、

$$\frac{1}{a} \frac{\delta r}{\delta p} = \frac{r}{a \sin p} \quad (2.151)$$

であり、結局、次の微分方程式が得られる。

$$\frac{\delta r}{r} = \frac{\delta p}{\sin p} \quad (2.152)$$

これを解くと、 $c$ を積分定数として、

$$r = c \left( \tan \frac{p}{2} \right) \quad (2.153)$$

が得られるが、ある基準緯度  $p$  では歪がないので、式 (2.150) を用いて、

$$n_\phi = \frac{c \tan \frac{p}{2}}{2a \sin \frac{p}{2} \cos \frac{p}{2}} = \frac{c}{2a \cos^2 \frac{p}{2}} = 1 \quad (2.154)$$

となる。

### ランベルト正角円錐図法

#### 正角円筒図法（メルカトル図法）

緯度  $\phi$  における緯線方向の拡大率は、

$$n_\phi = \frac{2\pi a}{2\pi a \cos \phi} = \sec \phi \quad (2.155)$$

である。この図法は正角投影であるので、経度方向の拡大率についても、

$$m_\lambda = \sec \phi \quad (2.156)$$

でなければならない。

さて、正角円筒図法では、図で見るとおり、経度方向を  $x$ 、緯度方向を  $y$  に単純に置き換えることができる。今、地図上における基準経度からの経度  $\lambda$  までの距離を  $x$ 、赤道からの緯度  $\phi$  までの距離を  $y$  とすれば、

$$x = a\lambda \quad (2.157)$$

$$\delta y = a \sec \phi \delta \phi \quad (2.158)$$

書くことができる。 $y$  方向については、微分方程式を解いて、

$$y = a \ln \left[ \tan \left( \frac{\pi}{4} + \frac{\phi}{2} \right) \right] \quad (2.159)$$

となる。





## Chapter 3

# Subgrid Scale Diffusion

Atmosphere is represented by grided point value in numerical simulation. However, air motions of smaller scale than that grid distance always exist in actual atmosphere. No matter how small we may get grid distance, such motions can exist. Those are called the subgrid scale motions, which act as diffusion in general. In addition, those correspond to turbulence and are often called ‘turbulence diffusion’.

If the subgrid scale motion cannot be expressed when we use smaller grid distance, it is possible for us to obtain prognostic equations of the subgrid scale motion theoretically. For example, we can separate velocity into the corresponding means and deviation from that means. In the equations of averaged variables, the second-degree correlation of deviation appears as unknown variables which is called Reynolds stress. So we have to get their prognostic equations. However, the third-degree correlation appears in the prognostic equations. To repeat such operation produces many unknown variables, and a system of equations is not closed. Such problem is caused by nonlinearity of turbulent flow. Kellar and Friedmann (1924) got first recognition on this problem, which is called ‘closure problem’.

One of solution is that you rewrite odd unknown variables by known variables using equations limited. This solution is called ‘closure assumptions’. Degree of predicted correlation decides what the solution is called, for example, one order closure, two order closure and so on. Furthermore, another solution is that you use only part of momentum equations system as closure assumptions. Thus, when it comes to usage of two-degree correlation, an expression of subgrid scale motion is made two separation.

- Modeling of unknown variables in the prognostic equations by dealing with two-degree correlation
- Modeling of the prognostic equations on the two-degree correlation represented by scalar, which can indicate averaged velocity, turbulence kinetic energy and turbulent flow of dissipation ratio through a concept of eddy viscosity

In this chapter, we discuss the formulation of subgrid scale motion with these two manner.

### 3.1 Parameteriaation of Turbulence Transport

There are various scale motion in atomosphere. The motion, which can be expressed with grid of numerical simulation, is called grid-scale motion, mean motion, subgrid scale motion or eddy motion.

To separate these motions, we assume that the field variables  $A$ , velocity, temperature, mixing ratio and so on, can be separated into mean field and deviation components.

$$A = \bar{A} + A'' \quad (3.1)$$

where the corresponding means are indicated by  $\bar{\quad}$  and the deviation components by  $''$ .

There are various ways when we average variables, but we do not show them in detail. For your information, average of deviations and average of products of two variabeles are given by

$$\overline{A''} = 0 \quad (3.2)$$

$$\overline{AB} = \bar{A}\bar{B} + \overline{A''B''} \quad (3.3)$$

In other words, if you average products, you do not always get the variable which correspond with products of each averaged, and the second term of above equation appears. We apply them to  $x$  components in equations of motion. Here, to simplify we discuss incompressible fluid  $\rho = const$ , and we can obtain a variable consisting of mean and deviation components.

$$\begin{aligned} \frac{\partial \bar{u}}{\partial t} + \bar{u} \frac{\partial \bar{u}}{\partial x} + \bar{v} \frac{\partial \bar{u}}{\partial y} + \bar{w} \frac{\partial \bar{u}}{\partial z} - f\bar{v} &= -\frac{1}{\rho} \frac{\partial \bar{p}}{\partial x} \\ &- \frac{1}{\rho} \left( \frac{\partial}{\partial x} \overline{\rho u'' u''} + \frac{\partial}{\partial y} \overline{\rho u'' v''} + \frac{\partial}{\partial z} \overline{\rho u'' w''} \right) + \nu \nabla^2 \bar{u} \end{aligned} \quad (3.4)$$

The terms  $-\overline{\rho u'' u''}$ ,  $-\overline{\rho u'' v''}$ ,  $-\overline{\rho u'' w''}$  in this equation represent stress by turbulent flow, which is called eddy stress or Reynolds stress. We can regard them as transport of momentum, so stress are made by trasport of momentum by eddy.

Similarly, the equations on potential temperature and mixing ratio are given by

$$\frac{\partial \bar{\theta}}{\partial t} + \bar{u} \frac{\partial \bar{\theta}}{\partial x} + \bar{v} \frac{\partial \bar{\theta}}{\partial y} + \bar{w} \frac{\partial \bar{\theta}}{\partial z} = -\frac{\partial}{\partial x} \overline{u'' \theta''} - \frac{\partial}{\partial y} \overline{v'' \theta''} - \frac{\partial}{\partial z} \overline{w'' \theta''} \quad (3.5)$$

Even if you use grid, the variables indicated by prime cannot be expressed. If you use them in numerical simulation, you have to represent them with the variables indicated by overbar (mean components). If you not, you cannot get effects of transport by eddy motion which has smaller scale than grid distance. The problem of turbulent parametarization is how we have to express such deviation by eddy, which is indicated by prime, by using limited grid. In addition, you need to pay attention to a difference between the definition of these variables indicated by  $\bar{\quad}$  and the definition of variables indicated by  $\quad''$  shown at the second chapter.

## 3.2 Eddy Viscosity Model

### 3.2.1 Formulation of Diffusion Term

In this section, we formulate diffusion term (the term of turbulence mixing)  $G^{\frac{1}{2}}\text{Turb.}\phi$ . The diffusion term appears in Equations of motion (??)(??), Equation of potential temperature (??), Equations of mixing ratio of water vapor and hydrometeor (??) and Equations of number concentration per unit volume (??), which shown in the basic dynamical equations in terrain-following coordinates at the section ??.

The diffusion term in the equation of motions is expressed using stress tensor  $\tau_{ij}$  as follows:

$$\begin{aligned} G^{\frac{1}{2}}\text{Turb.}u &= G^{\frac{1}{2}} \left( \frac{\partial\tau_{11}}{\partial x} + \frac{\partial\tau_{12}}{\partial y} + \frac{\partial\tau_{13}}{\partial z} \right) \\ &= \frac{\partial}{\partial\xi} (J_d\tau_{11}) + \frac{\partial}{\partial\eta} (J_d\tau_{12}) + \frac{\partial}{\partial\zeta} (\tau_{13} + J_{31}\tau_{11} + J_{32}\tau_{12}) \end{aligned} \quad (3.6)$$

$$\begin{aligned} G^{\frac{1}{2}}\text{Turb.}v &= G^{\frac{1}{2}} \left( \frac{\partial\tau_{21}}{\partial x} + \frac{\partial\tau_{22}}{\partial y} + \frac{\partial\tau_{23}}{\partial z} \right) \\ &= \frac{\partial}{\partial\xi} (J_d\tau_{21}) + \frac{\partial}{\partial\eta} (J_d\tau_{22}) + \frac{\partial}{\partial\zeta} (\tau_{23} + J_{31}\tau_{21} + J_{32}\tau_{22}) \end{aligned} \quad (3.7)$$

$$\begin{aligned} G^{\frac{1}{2}}\text{Turb.}w &= G^{\frac{1}{2}} \left( \frac{\partial\tau_{31}}{\partial x} + \frac{\partial\tau_{32}}{\partial y} + \frac{\partial\tau_{33}}{\partial z} \right) \\ &= \frac{\partial}{\partial\xi} (J_d\tau_{31}) + \frac{\partial}{\partial\eta} (J_d\tau_{32}) + \frac{\partial}{\partial\zeta} (\tau_{33} + J_{31}\tau_{31} + J_{32}\tau_{32}) \end{aligned} \quad (3.8)$$

where stress tensor  $\tau_{ij}$  consists of shear stress and Reynolds stress. Reynolds stress consists of fluctuation from averaged variables, so we need modeling in any way to the form with averaged variables. Reynolds stress can express in the form of gradient diffusion using viscosity coefficient from an analogy of shear stress.

$$\tau_{11} = \bar{\rho}\nu_{\tau h} \left( S_{11} - \frac{2}{3}Div \right) \quad (3.9)$$

$$\tau_{12} = \bar{\rho}\nu_{\tau h} S_{12} \quad (3.10)$$

$$\tau_{13} = \bar{\rho}\nu_{\tau v} S_{13} \quad (3.11)$$

$$\tau_{21} = \bar{\rho}\nu_{\tau h} S_{12} \quad (3.12)$$

$$\tau_{22} = \bar{\rho}\nu_{\tau h} \left( S_{22} - \frac{2}{3}Div \right) \quad (3.13)$$

$$\tau_{23} = \bar{\rho}\nu_{\tau v} S_{23} \quad (3.14)$$

$$\tau_{31} = \bar{\rho}\nu_{\tau h} S_{13} \quad (3.15)$$

$$\tau_{32} = \bar{\rho}\nu_{\tau h} S_{23} \quad (3.16)$$

$$\tau_{33} = \bar{\rho}\nu_{\tau v} \left( S_{33} - \frac{2}{3}Div \right) \quad (3.17)$$

where  $\nu_{\tau h}$  and  $\nu_{\tau v}$  are horizontal and vertical eddy viscosity coefficients regarding kinetic momentum, respectively. The molecular viscosity coefficient of shear stress is so small to eddy viscosity coefficient that it can be neglected.  $S_{ij}$  is deformation rate tensor. In curvilinear coordinate system, it is given by

$$S_{11} = 2 \frac{\partial u}{\partial x} = \frac{2}{G^{\frac{1}{2}}} \left[ \frac{\partial}{\partial \xi} (J_d u) + \frac{\partial}{\partial \zeta} (J_{31} u) \right] \quad (3.18)$$

$$S_{22} = 2 \frac{\partial v}{\partial y} = \frac{2}{G^{\frac{1}{2}}} \left[ \frac{\partial}{\partial \eta} (J_d v) + \frac{\partial}{\partial \zeta} (J_{32} v) \right] \quad (3.19)$$

$$S_{33} = 2 \frac{\partial w}{\partial z} = \frac{2}{G^{\frac{1}{2}}} \frac{\partial w}{\partial \zeta} \quad (3.20)$$

$$S_{12} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} = \frac{1}{G^{\frac{1}{2}}} \left[ \frac{\partial}{\partial \eta} (J_d u) + \frac{\partial}{\partial \xi} (J_d v) + \frac{\partial}{\partial \zeta} (J_{32} u + J_{31} v) \right] \quad (3.21)$$

$$S_{13} = \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} = \frac{1}{G^{\frac{1}{2}}} \left[ \frac{\partial}{\partial \xi} (J_d w) + \frac{\partial}{\partial \zeta} (u + J_{31} w) \right] \quad (3.22)$$

$$S_{23} = \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} = \frac{1}{G^{\frac{1}{2}}} \left[ \frac{\partial}{\partial \eta} (J_d w) + \frac{\partial}{\partial \zeta} (v + J_{32} w) \right] \quad (3.23)$$

and  $Div$  is divergence.

$$Div = \frac{1}{G^{\frac{1}{2}}} \left[ \frac{\partial}{\partial \xi} (G^{\frac{1}{2}} u) + \frac{\partial}{\partial \eta} (G^{\frac{1}{2}} v) + \frac{\partial}{\partial \zeta} (G^{\frac{1}{2}} W) \right] \quad (3.24)$$

The diffusion terms of potential temperature, mixing ration of hydrometeor and water vapor and the number concentration per unit volume are formalized by using  $\phi$  as follows:

$$\begin{aligned} G^{\frac{1}{2}} \text{Turb.} \phi &= G^{\frac{1}{2}} \left( \frac{\partial H_{\phi 1}}{\partial x} + \frac{\partial H_{\phi 2}}{\partial y} + \frac{\partial H_{\phi 3}}{\partial z} \right) \\ &= \frac{\partial}{\partial \xi} (J_d H_{\phi 1}) + \frac{\partial}{\partial \eta} (J_d H_{\phi 2}) + \frac{\partial}{\partial \zeta} (H_{\phi 3} + J_{31} H_{\phi 1} + J_{32} H_{\phi 2}) \end{aligned} \quad (3.25)$$

where  $H_{\phi 1}$ ,  $H_{\phi 2}$  and  $H_{\phi 3}$  are the molecular diffusions of the corresponding scalar  $\phi$  and turbulent fluxes in the  $x$ ,  $y$  and  $z$  directions, respectively. They have forms of the gradient diffusion similar to velocity

$$H_{\phi 1} = \bar{\rho} \nu_{Hh} \frac{\partial \phi}{\partial x} = \bar{\rho} \nu_{Hh} \frac{1}{G^{\frac{1}{2}}} \left[ \frac{\partial}{\partial \xi} (J_d \phi) + \frac{\partial}{\partial \zeta} (J_{31} \phi) \right] \quad (3.26)$$

$$H_{\phi 2} = \bar{\rho} \nu_{Hh} \frac{\partial \phi}{\partial y} = \bar{\rho} \nu_{Hh} \frac{1}{G^{\frac{1}{2}}} \left[ \frac{\partial}{\partial \eta} (J_d \phi) + \frac{\partial}{\partial \zeta} (J_{32} \phi) \right] \quad (3.27)$$

$$H_{\phi 3} = \bar{\rho} \nu_{Hv} \frac{\partial \phi}{\partial z} = \bar{\rho} \nu_{Hv} \frac{1}{G^{\frac{1}{2}}} \frac{\partial \phi}{\partial \zeta} \quad (3.28)$$

where  $\nu_{Hh}$  and  $\nu_{Hv}$  are horizontal and vertical eddy viscosity coefficients about scalar, respectively. The molecular diffusion coefficient is so small that it is neglected.

We have the modeling of Reynolds to the form with eddy viscosity coefficient. The manner is called eddy viscosity model, with which we can appreciate the eddy viscosity coefficients  $\nu_{\tau h}$  and  $\nu_{\tau v}$  and the eddy diffusion coefficients  $\nu_{Hh}$  and  $\nu_{Hv}$  appear in above equations.

In following section, two eddy viscosity model are explained, which are actually used in **CRess**.

- One order closure of Smagorinsky
- One and a half order closure with turbulence kinetic energy

### 3.2.2 One order closure of Smagorinsky

Smagorinsky (1963) and Lilly (1962) give the eddy viscosity coefficient in the case it is isotropic vertically and horizontally, using  $\nu_{\tau h} = \nu_{\tau v} = \nu_{\tau}$

$$\nu_{\tau} = \begin{cases} (C_S \Delta)^2 \left( Def^2 - \frac{N^2}{Pr} \right), & \nu_{\tau} > 0 \\ 0, & \nu_{\tau} \leq 0 \end{cases} \quad (3.29)$$

where  $C_S$  is smagorinsky constant,  $C_S = 0.21$  by Deardorff (1972a).  $\Delta$  is averaged grid interval of numerical simulation.

$$\Delta = (\Delta x \Delta y \Delta z)^{\frac{1}{3}} \quad (3.30)$$

$Def$ , which is measurement of transformation, can be obtained

$$Def^2 = \frac{1}{2} (S_{11}^2 + S_{22}^2 + S_{33}^2) + S_{12}^2 + S_{13}^2 + S_{23}^2 - \frac{2}{3} Div^2 \quad (3.31)$$

and

$$N^2 = \begin{cases} \frac{g}{G^{\frac{1}{2}}} \frac{\partial \ln \theta}{\partial \zeta}, & q_v < q_{vsw} \\ \frac{g}{G^{\frac{1}{2}}} \left[ \frac{1 + \frac{\mathcal{L}_v q_{vsw}}{R_d T}}{1 + \frac{\mathcal{L}_v^2 q_{vsw}}{C_p R_v T^2}} \left( \frac{\partial \ln \theta}{\partial \zeta} + \frac{\mathcal{L}_v}{C_p T} \frac{\partial q_{vsw}}{\partial \zeta} \right) - \frac{\partial q_w}{\partial \zeta} \right], & q_v \geq q_{vsw} \end{cases} \quad (3.32)$$

where  $N$  is a constant Brunt-Väisälä frequency, and

$$Pr = \frac{\nu_\tau}{\nu_H}, \quad \nu_{Hh} = \nu_{Hv} = \nu_H \quad (3.33)$$

where  $Pr$  is Turbulent Prandtl number. So we can obtain the eddy viscosity coefficient regarding scalar  $\phi$ .  $g$  is gravity acceleration,  $T$  is temperature,  $R_d$  and  $R_v$  are gas constants for dry air and wet air, respectively.  $C_p$  is specific heat at constant pressure and  $q_w$  is a sum of mixing ratio of molecular weight for water vapor, cloud liquid water and rainwater. With the equation of Tetens, mixing ratio of water saturation  $q_{vs_w}$  is given by

$$q_{vs_w} = \epsilon \frac{610.78}{p} \exp\left(17.269 \frac{T - 273.16}{T - 35.86}\right) \quad (3.34)$$

and latent heat for water evaporation  $\mathcal{L}_v$  is given by

$$\mathcal{L}_v = 2.50078 \times 10^6 \left(\frac{273.16}{T}\right)^{(0.167+3.67 \times 10^{-4}T)} \quad (3.35)$$

where  $\epsilon$  is the ratio of molecular weight of water vapor and of dry air.

Next, in the case it is anisotropic vertically and horizontally, the eddy viscosity coefficients of each direction are represented as

$$\nu_{\tau h} = \begin{cases} (C_S \Delta_h)^2 \left(Def^2 - \frac{N^2}{Pr}\right), & \nu_{\tau h} > 0 \\ 0, & \nu_{\tau h} \leq 0 \end{cases} \quad (3.36)$$

$$\nu_{\tau v} = \begin{cases} (C_S \Delta_v)^2 \left(Def^2 - \frac{N^2}{Pr}\right), & \nu_{\tau v} > 0 \\ 0, & \nu_{\tau v} \leq 0 \end{cases} \quad (3.37)$$

where  $\Delta_h$  and  $\Delta_v$  are given by

$$\Delta_h = (\Delta x \Delta y)^{\frac{1}{2}} \quad (3.38)$$

$$\Delta_v = \Delta z \quad (3.39)$$

Furthermore,

$$Pr = \frac{\nu_{\tau h}}{\nu_{Hh}} = \frac{\nu_{\tau v}}{\nu_{Hv}} \quad (3.40)$$

Thus, we obtain the eddy viscosity coefficients  $\nu_{Hh}, \nu_{Hv}$  of each direction regarding scalar  $\phi$ , using the Turbulent Prandtl number  $Pr$ .

### 3.2.3 One and a half order closure with turbulence kinetic energy

With regard to one and a half order closure, we use prognostic equations on turbulence kinetic energy. We mark deviation from averaged flow each velocity component with "", so this turbulence kinetic energy is represented as

$$E = \frac{1}{2} \left( \overline{u''^2 + v''^2 + w''^2} \right) \quad (3.41)$$

and, the prognostic equations are given by

$$\begin{aligned} \frac{\partial \rho^* E}{\partial t} = & - \left( u^* \frac{\partial E}{\partial \xi} + v^* \frac{\partial E}{\partial \eta} + W^* \frac{\partial E}{\partial \zeta} \right) + \text{Buoy.}E + \rho^* \left( \frac{1}{2} \nu_E \text{Def}^2 - \frac{2}{3} E \text{Div} \right) - \rho^* \frac{C_e}{l_h} E^{\frac{3}{2}} \\ & + \left[ \frac{\partial}{\partial \xi} (J_d H_{E1}) + \frac{\partial}{\partial \eta} (J_d H_{E2}) + \frac{\partial}{\partial \zeta} (H_{E3} + J_{31} H_{E1} + J_{32} H_{E2}) \right] \end{aligned} \quad (3.42)$$

Buoy. $E$  which appear in equations (3.42), is given as follows with the converting term of potential energy and kinetic energy.

$$\text{Buoy.}E = \begin{cases} -g \rho^* \nu_{Hv} \left( A \frac{\partial \theta_e}{\partial \zeta} - \frac{\partial q_{all}}{\partial \zeta} \right), & q_v \geq q_{vsw} \text{ or } q_c + q_i > 0 \\ -g \rho^* \nu_{Hv} \left( \frac{1}{\theta} \frac{\partial \theta}{\partial \zeta} + 0.61 \frac{\partial q_v}{\partial \zeta} \right), & q_v < q_{vsw} \text{ or } q_c + q_i = 0 \end{cases} \quad (3.43)$$

where,  $A$  is expressed as

$$A = \frac{1}{\theta} \frac{1 + \frac{1.61 \epsilon \mathcal{L}_v q_v}{R_d T}}{1 + \frac{\epsilon \mathcal{L}_v^2 q_v}{C_p R_d T^2}} \quad (3.44)$$

where,  $g$  is gravity acceleration,  $T$  is temperature,  $\epsilon$  is the ratio of molecular weight for water vapor and dry air,  $q_{all}$  is a sum of mixing ratio of molecular weight for water vapor, cloud liquid water and cloud ice, and  $\theta_e$  is equivalent potential temperature.  $C_p$  and  $R_d$  are gas constants for dry air and wet air, respectively.  $\mathcal{L}_v$  is latent heat for water evaporation.

Next, the third term of right hand  $Def$ ,  $Div$  is shown at the section 3.2.2. The coefficient  $C_e$  of the fourth term, the dissipating term is represented as

$$C_e = \begin{cases} 3.9, & \text{the lowest layer} \\ 0.93, & \text{the other layer} \end{cases} \quad (3.45)$$

In addition, the last term of right hand, flux of turbulence kinetic energy is given by

$$H_{E1} = \bar{\rho} \nu_E \frac{\partial E}{\partial x} = \bar{\rho} \nu_E \frac{1}{G^{\frac{1}{2}}} \left[ \frac{\partial}{\partial \xi} (J_d E) + \frac{\partial}{\partial \zeta} (J_{31} E) \right] \quad (3.46)$$

$$H_{E2} = \bar{\rho} \nu_E \frac{\partial E}{\partial y} = \bar{\rho} \nu_E \frac{1}{G^{\frac{1}{2}}} \left[ \frac{\partial}{\partial \eta} (J_d E) + \frac{\partial}{\partial \zeta} (J_{32} E) \right] \quad (3.47)$$

$$H_{E3} = \bar{\rho} \nu_E \frac{\partial E}{\partial z} = \bar{\rho} \nu_E \frac{1}{G^{\frac{1}{2}}} \frac{\partial E}{\partial \zeta} \quad (3.48)$$

where  $\nu_E$  is the eddy viscosity coefficient for turbulence kinetic energy.

Thus, the eddy viscosity coefficients  $\nu_{\tau h}, \nu_{\tau v}$  are represented as a function of turbulent kinetic energy  $E$ ,

$$\nu_{\tau h} = 0.1E^{\frac{1}{2}}l_h \quad (3.49)$$

$$\nu_{\tau v} = 0.1E^{\frac{1}{2}}l_v \quad (3.50)$$

where  $l_h$  and  $l_v$  are horizontal and vertical mixing length scales. In the case where the difference is small between horizontal and vertical grid interval,

$$l = l_h = l_v = \begin{cases} \Delta s, & \text{unstable or neutral} \\ \min(\Delta s, l_s), & \text{stable} \end{cases} \quad (3.51)$$

where  $\Delta s$  and  $l_s$  are expressed as

$$\Delta s = \Delta s_h = \Delta s_v = (\Delta x \Delta y \Delta z)^{\frac{1}{3}} \quad (3.52)$$

$$l_s = 0.76E^{\frac{1}{2}} \left| \frac{g}{\theta} \frac{\partial \bar{\theta}}{\partial z} \right|^{-\frac{1}{2}} \quad (3.53)$$

On the other hand, in the case where the difference is large between horizontal and vertical grid interval,

$$l_h = \Delta s_h \quad (3.54)$$

$$l_v = \begin{cases} \Delta s_v, & \text{unstable or neutral} \\ \min(\Delta s_v, l_s), & \text{stable} \end{cases} \quad (3.55)$$

where  $\Delta s_h$  and  $\Delta s_v$  are expressed as

$$\Delta s_h = (\Delta x \Delta y)^{\frac{1}{2}} \quad (3.56)$$

$$\Delta s_v = \Delta z \quad (3.57)$$

Finally, the eddy viscosity coefficients  $\nu_{\tau h}$  and  $\nu_{\tau v}$  are represented as a function of turbulence kinetic energy  $E$ ,

$$\nu_{\tau h} = \max \left( 0.1E^{\frac{1}{2}}l_h, \alpha \Delta s_h^2 \right) \quad (3.58)$$

$$\nu_{\tau v} = \max \left( 0.1E^{\frac{1}{2}}l_v, \alpha \Delta s_v^2 \right) \quad (3.59)$$

where  $\alpha$  is smaller number like  $\alpha = 10^{-6}$ . The eddy viscosity coefficient  $\nu_{Hv}, \nu_{Hh}$  for scalar  $\phi$  and  $\nu_E$  for the turbulence kinetic energy  $E$  can be obtained as follows:



In the case of the small difference between horizontal and vertical grid interval,

$$\frac{\nu_{\tau h}}{\nu_{Hh}} = \frac{\nu_{\tau v}}{\nu_{Hv}} = \frac{1}{1 + 2l/\Delta s} \quad ( = Pr ) \quad (3.60)$$

$$\nu_E = 2\nu_{\tau h} = 2\nu_{\tau v} \quad (3.61)$$

In the case of the large difference between horizontal and vertical grid interval,

$$\frac{\nu_{\tau h}}{\nu_{Hh}} = \frac{1}{1 + 2l_h/\Delta s_h} = \frac{1}{3} \quad (3.62)$$

$$\frac{\nu_{\tau v}}{\nu_{Hv}} = \frac{1}{1 + 2l_v/\Delta s_v} \quad ( = Pr ) \quad (3.63)$$

$$\nu_E = 2\nu_{\tau h} \quad (3.64)$$



## Chapter 4

# Physical Processes of Cloud and Precipitation

Cloud physics is roughly divided into cloud microphysics and cloud dynamics. Because they are closely concerned in each other, knowledges of cloud physics is necessary for understanding cloud dynamics.

Ascending a humid air parcel in atmosphere, clouds generate in association with a conversion from water vapor to cloud and precipitation particles and precipitation is induced. We generally classify the formation processes of precipitation into “warm rain” and “cold rain (ice-phase rain).” The “warm rain” is rainfall which has no ice-phase process, and clouds form below the altitude of  $0^{\circ}\text{C}$ . Such clouds is called “warm clouds.” The “cold rain” is rainfall of which ice-phase processes considerably contribute for the growth of precipitation particles. Such clouds is called “cold clouds.” The clouds form not only below the altitude of  $0^{\circ}\text{C}$  but also above the altitude. The clouds have both a liquid phase and an ice-phase. There are the following categories in modeling of these clouds.

- Parametarization of “warm rain” with a bulk method
- Parametarization containing an ice-phase with a bulk method
- A method predicting time development of size distribution with dividing the sizes of liquid-phase particles into several bins (a bin method)
- A bin method as same as the above but containing an ice-phase
- A bin method containing categorized aerosols
- A method of which a bin method is used for a liquid phase and a bulk method for an ice-phase (A hybrid type)

Bulk method parametarizations of “warm rain” and “cold rain” are used in the *CReSS*. Hereafter, modelings of these two methods are described.

## 4.1 Parameterization of “Warm Rain” with a Bulk Method

### 4.1.1 Equations for “Warm Rain” Process of Cloud and Precipitation

The parameterization of “warm rain” with a bulk method has the following three categories of water substance<sup>1</sup>.

Sign	Meaning	Contents
$q_v$	Mixing ratio of water vapor	Water which exists in the atmosphere in the state of gas
$q_c$	Mixing ratio of cloud water	Its fall velocity can be regarded as zero. It corresponds to cloud drops in the actual atmosphere. The cloud drop is ordinary a small particle of which its diameter is less than 100 $\mu\text{m}$ .
$q_r$	Mixing ratio of rain water	It corresponds to raindrops, which have a meaningful fall velocity, in the actual atmosphere.

Modeling variables involved in the “warm rain” process are three variables for water substance [ $\text{kg kg}^{-1}$ ] and potential temperature (temperature) [K]. Using equations shown in section 2.1 (we can explain by using the equations in which orography is not considered), the equations for the “warm rain” process are simply written as follows.

$$\frac{\partial \bar{\rho}\theta}{\partial t} = \text{Adv.}\theta + \text{Turb.}\theta - \bar{\rho}w \frac{\partial \bar{\theta}}{\partial z} + \frac{\bar{\rho}\mathcal{L}_v}{C_p\Pi} (CN_{vc} - EV_{cv} - EV_{rv}) \quad (4.1)$$

$$\frac{\partial \bar{\rho}q_v}{\partial t} = \text{Adv.}q_v + \text{Turb.}q_v - \bar{\rho} (CN_{vc} - EV_{cv} - EV_{rv}) \quad (4.2)$$

$$\frac{\partial \bar{\rho}q_c}{\partial t} = \text{Adv.}q_c + \text{Turb.}q_c + \bar{\rho} (CN_{vc} - EV_{cv} - CN_{cr} - CL_{cr}) \quad (4.3)$$

$$\frac{\partial \bar{\rho}q_r}{\partial t} = \text{Adv.}q_r + \text{Turb.}q_r + \bar{\rho} (CN_{cr} + CL_{cr} - EV_{rv}) + \frac{\partial}{\partial z} (\bar{\rho}U_r q_r) \quad (4.4)$$

where  $\text{Adv.}\phi$  and  $\text{Turb.}\phi$  represent the terms of advection and subgrid-scale turbulence.  $\mathcal{L}_v$ ,  $C_p$  and  $\Pi$  are latent heat of evaporation of water [ $\text{J kg}^{-1}$ ], specific heat at constant pressure in a dry air [ $\text{J K kg}^{-1}$ ] and the Exner function, respectively. The last term of the right side of Eq. (4.4) represents a flux divergence of  $q_r$  in association with falling of rainwater. Microphysical processes considered here are as follows.

Sign	Contents
$CN_{vc}$	Conversion from water vapor to cloud water by condensation (condensation)
$EV_{cv}$	Conversion from cloud water to water vapor by evaporation (evaporation)
$EV_{rv}$	Conversion from rain water to water vapor by evaporation (evaporation)
$CN_{cr}$	Conversion from cloud water to rain water by coalescence growth. It corresponds to growing from a cloud droplet to the size of a raindrop by coalescence or diffusion of water vapor (autoconversion).
$CL_{cr}$	Conversion from cloud water to rain water by collision. A process of which a large waterdrop captures a small waterdrop (collection).

All of these amounts are defined as a positive value. A process of condensation from water vapor to rain

<sup>1</sup>Rain water which has a diameter of 0.1-0.5 mm is sometimes called drizzle. Here, drizzle is contained in a category of rain.

water is neglected. Calculations of those processes are shown in the next section.

### 4.1.2 Microphysical Processes

**Conversion between Water Vapor and Cloud Water:**  $-CN_{vc} + EV_{cv}$

As the same as Klemp and Wilhelmson (1978), we use the method of the moist saturation adjustment by Soong and Ogura (1973). This method will be explained in Section 4.2.5.

**Saturation Mixing Ratio (Tetens Equation):**  $q_{vs_w}$

Saturation mixing ratio over water  $q_{vs_w}$  is represented by using the Tetens equation.

$$q_{vs_w} = \epsilon \frac{610.78}{p} \exp \left( 17.269 \frac{\Pi\theta - 273.16}{\Pi\theta - 35.86} \right) \quad (4.5)$$

Here,  $\epsilon$  is the ratio with the molecular weight of water vapor and the molecular weight of dry air.

**Conversion from Cloud Water to the Rain Water:**  $CN_{cr}, CL_{cr}$

The conversion from cloud water to rain water with coalescence growth ( $CN_{cr}$ ) and conversion from cloud water to rain water ( $CL_{cr}$ ) is calculated by using the Kessler (1969)’s parameterization.

$$CN_{cr} = k_1 (q_c - a) \quad (4.6)$$

$$CL_{cr} = k_2 q_c q_r^{0.875} \quad (4.7)$$

where

$$k_1 = 0.001 \text{ [s}^{-1}\text{]} \quad (4.8)$$

$$a = 0.001 \text{ [kg kg}^{-1}\text{]} \quad (4.9)$$

$$k_2 = 2.2 \text{ [s}^{-1}\text{]}. \quad (4.10)$$

**Evaporation of Rain Water:**  $EV_{rv}$

In the same way as Ogura and Takahashi (1971) and Klemp and Wilhelmson (1978), evaporation of rain water is represented as a following.

$$EV_{rv} = \frac{1}{\bar{\rho}} \frac{(1 - q_v / q_{vs_w}) C (\bar{\rho} q_r)^{0.525}}{5.4 \times 10^5 + 2.55 \times 10^6 / (p q_{vs_w})} \quad (4.11)$$

where  $C$  is a ventilation factor given by

$$C = 1.6 + 124.9 (\bar{\rho} q_r)^{0.2046}. \quad (4.12)$$

### Fall Velocity of Rain: $U_r$

Adding change of density to the equation of Soong and Ogura (1973), fall velocity  $U_r$  of the last term in the right side of the equation (4.4) is given by

$$U_r = 36.34 (\bar{\rho} q_r)^{0.1346} \left( \frac{\rho_0}{\bar{\rho}} \right). \quad (4.13)$$

where  $\rho_0$  is density [ $\text{kg m}^{-3}$ ] at the surface in the basic state and the unit of  $U_r$  is [ $\text{m s}^{-1}$ ]. Precipitation at the surface is calculated by using this fall velocity. Note that a difference in  $z^*$  coordinate system ( $\zeta$  coordinate system) is multiplied by a matrix, as shown in equations (2.58).

## 4.2 Parameterization of Cloud and Precipitation with the Mixed-phase Processes

### 4.2.1 Parameterization of the Mixed-phase Processes with a Bulk Method

We formulate the parameterization of the cloud and precipitation processes containing ice-phase with a bulk method which are used in a cloud model. Here, conversions of water substance and changes of temperature and mixing ratio of water vapor by the conversions are considered. When we use a bulk method, the water in the atmosphere is divided into some categories (e.g. rain, snow and graupel) The categories are formulized with the typical variables (usually, mixing ratio or mixing ratio and number concentration), and their time developments are calculated. Therefore, when we use a bulk method, the definition of each variable must be clear. The way of categorization and variables are different in each model.

The following physical processes are considered in formulization of parameterization containing an ice-phase. In the formulization, the type of a particle must be considered.

- Primary and secondary nucleation of ice crystals
- Growth and decline of the particle with water vapor diffusion.
- Growth with collision between particles
- Breakup of a particle (breakup of a raindrop)
- Conversion to another category (Conversion from the cloud water to rain water, and from cloud water to snow, and from snow to graupel. etc.)
- Freezing and melting
- Shedding of un-freezing water
- Gravity falling

Although the definitions of the cloud physical variables and its treatment are different in each model, we show two types of models containing an ice-phase in the following.

- A model which calculates only the time development equation of the mixing ratio of each category
- A model which calculates the graupel, cloud ice, snow, and their number concentrations adding the above equation

Hereafter, we summarize the formulizations by Murakami (1999), Murakami et al. (1994) and Murakami (1990).

We consider the six categories for the physical processes of cloud and precipitation: water vapor, cloud water, rain water, cloud ice, snow and graupel. In addition to the six categories, there is a model which makes fog water and hail another categories.

Sign	Meaning	Contents
$\theta$	Potential temperature	$\theta = \bar{\theta} + \theta'$
$q_v$	Mixing ratio of water vapor	Water which exists in the atmosphere in the state of gas.
$q_c$	Mixing ratio of cloud water	Liquid water of which a particle diameter is small and fall velocity is negligible. It can move with atmospheric motion.
$q_r$	Mixing ratio of rain water	Usually, the particle of a liquid water with a diameter more than 100 $\mu\text{m}$ is called rain. It is expressed as "rain water" in a model. Fall velocity depending on the size of a particle is meaningful. Although it moves in association with the atmospheric motion in horizontal, it falls and drops out of an air mass in vertical.
$q_i$	Mixing ratio of cloud ice	A minute crystal of ice which is called "ice crystal" in cloud physics. Usually, a diameter is less than 100 $\mu\text{m}$ . Fall velocity is so small that it is negligible.
$q_s$	Mixing ratio of snow	In a model, it means a solid precipitation particle with density of about 0.1 $\text{g cm}^{-3}$ and fall velocity of about 1 $\text{m s}^{-1}$ . It corresponds to a snow crystal or snow flakes, etc.
$q_g$	Mixing ratio of graupel	In a model, it means a solid precipitation particle with density of about 0.4 $\text{g cm}^{-3}$ and fall velocity of about 1~4 $\text{m s}^{-1}$ . It corresponds to a snow crystal with cloud droplets, snow flakes with cloud droplets, graupel, etc.
$q_h$	Mixing ratio of hail	In a model, it means a solid precipitation particle with density of about 0.9 $\text{g cm}^{-3}$ and fall velocity of about 10 $\text{m s}^{-1}$ fall speed. In an actual cloud particle, frozen rain, hail, etc. correspond. hail is contained in graupel in CReSS.
$N_i$	number concentration of cloud ice	Expressing cloud ice in a model, number concentration is sometimes regarded as a variable.
$N_s$	number concentration of snow	The same in case of snow.
$N_g$	number concentration of graupel	The same in case of graupel.

Here, the units of potential temperature, mixing ratio and number concentration are [K], [ $\text{kg kg}^{-1}$ ] and [ $\text{m}^{-3}$ ], respectively. We often use [ $\text{g kg}^{-1}$ ] as the unit of mixing ratio.

#### 4.2.2 The Equation System for Cloud and Precipitation Processes

The time development equations of potential temperature, water vapor and particles of cloud and precipitation are used here. Considering number concentrations of particles, the time development equations of number concentration of cloud ice, snow, and graupel are needed. These describe briefly the equation system shown in Section 2.1.

The time development equation of a potential temperature and a mixing ratio of water is expressed as

$$\frac{\partial \bar{\rho} \theta}{\partial t} = \text{Adv.} \theta + \text{Turb.} \theta - \bar{\rho} w \frac{\partial \bar{\theta}}{\partial z} + \bar{\rho} (\text{Src.} \theta_V + \text{Src.} \theta_S + \text{Src.} \theta_F) \quad (4.14)$$

$$\frac{\partial \bar{\rho} q_v}{\partial t} = \text{Adv.} q_v + \text{Turb.} q_v + \bar{\rho} \text{Src.} q_v \quad (4.15)$$

$$\frac{\partial \bar{\rho} q_c}{\partial t} = \text{Adv.} q_c + \text{Turb.} q_c + \bar{\rho} \text{Src.} q_c + \bar{\rho} \text{Fall.} q_c \quad (4.16)$$

$$\frac{\partial \bar{\rho} q_r}{\partial t} = \text{Adv.} q_r + \text{Turb.} q_r + \bar{\rho} \text{Src.} q_r + \bar{\rho} \text{Fall.} q_r \quad (4.17)$$

$$\frac{\partial \bar{\rho} q_i}{\partial t} = \text{Adv.} q_i + \text{Turb.} q_i + \bar{\rho} \text{Src.} q_i + \bar{\rho} \text{Fall.} q_i \quad (4.18)$$

$$\frac{\partial \bar{\rho} q_s}{\partial t} = \text{Adv.} q_s + \text{Turb.} q_s + \bar{\rho} \text{Src.} q_s + \bar{\rho} \text{Fall.} q_s \quad (4.19)$$

$$\frac{\partial \bar{\rho} q_g}{\partial t} = \text{Adv.} q_g + \text{Turb.} q_g + \bar{\rho} \text{Src.} q_g + \bar{\rho} \text{Fall.} q_g \quad (4.20)$$

where characters of  $v, c, r, i, s$ , and  $g$ , which are attached at the bottom, express cloud water, rain water, cloud ice, snow, and graupel, respectively. Hereafter, we sometimes use  $x$  or  $y$  as representative characters.

The meaning of each term is shown below.

Adv. $\phi$	Advection term of potential temperature or mixing ratio of water substance
Turb. $\phi$	Diffusion term of potential temperature or mixing ratio of water substance by a subgrid-scale turbulence
Src. $\theta_V$	Source term of potential temperature in association with condensation and evaporation
Src. $\theta_S$	Source term of potential temperature in association with sublimation
Src. $\theta_F$	Source term of potential temperature in association with freezing and melting
Src. $q_x$	Source term of the mixing ratio of water
Fall. $q_x$	Term of falling water substance (precipitation)

The time development equations of number concentrations of cloud ice, snow, and graupel are

$$\frac{\partial N_i}{\partial t} = \text{Adv.} \frac{N_i}{\bar{\rho}} + \text{Turb.} \frac{N_i}{\bar{\rho}} + \bar{\rho} \text{Src.} \frac{N_i}{\bar{\rho}} + \bar{\rho} \text{Fall.} \frac{N_i}{\bar{\rho}} \quad (4.21)$$

$$\frac{\partial N_s}{\partial t} = \text{Adv.} \frac{N_s}{\bar{\rho}} + \text{Turb.} \frac{N_s}{\bar{\rho}} + \bar{\rho} \text{Src.} \frac{N_s}{\bar{\rho}} + \bar{\rho} \text{Fall.} \frac{N_s}{\bar{\rho}} \quad (4.22)$$

$$\frac{\partial N_g}{\partial t} = \text{Adv.} \frac{N_g}{\bar{\rho}} + \text{Turb.} \frac{N_g}{\bar{\rho}} + \bar{\rho} \text{Src.} \frac{N_g}{\bar{\rho}} + \bar{\rho} \text{Fall.} \frac{N_g}{\bar{\rho}} \quad (4.23)$$

. where characters of  $i, s, g$ , which are attached at the bottom, express water vapor, cloud ice, snow, and graupel, respectively. Hereafter, we sometimes use  $x$  or  $y$  as representative characters.

The meaning of each term is shown below.

Adv. $N_x / \bar{\rho}$	Advection term of number concentration of water in the solid state
Turb. $N_x / \bar{\rho}$	Diffusion term of number concentration of water in the solid state by a subgrid-scale turbulence
Src. $N_x / \bar{\rho}$	Source term of the number concentration of water in the solid state
Fall. $N_x / \bar{\rho}$	Term of change of number concentration of water in the solid state in association with precipitation



Source terms of those equations are as follows.

**Source term of the equation (4.14) of potential temperature  $\theta$ :**  $\text{Src.}\theta_V + \text{Src.}\theta_S + \text{Src.}\theta_F$

$$\text{Src.}\theta_V = \frac{\mathcal{L}_v}{C_p \Pi} VD_{vr} \quad (4.24)$$

$$\text{Src.}\theta_S = \frac{\mathcal{L}_s}{C_p \Pi} (NUA_{vi} + VD_{vi} + VD_{vs} + VD_{vg}) \quad (4.25)$$

$$\begin{aligned} \text{Src.}\theta_F = \frac{\mathcal{L}_f}{C_p \Pi} & (NUF_{ci} + NUC_{ci} + NUH_{ci} + CL_{cs} + CL_{cg} + CL_{ri} + CL_{rs} + CL_{rg} \\ & - ML_{ic} - ML_{sr} - ML_{gr} + FR_{rg} - SH_{sr} - SH_{gr}) \end{aligned} \quad (4.26)$$

**Source term of the equation for mixing ratio of water vapor  $q_v$  (4.15):**

$$\text{Src.}q_v = -NUA_{vi} - VD_{vr} - VD_{vi} - VD_{vs} - VD_{vg} \quad (4.27)$$

**Source term of the equation for mixing ratio of cloud water  $q_c$  (4.16):**

$$\text{Src.}q_c = -NUF_{ci} - NUC_{ci} - NUH_{ci} - CL_{cr} - CL_{cs} - CL_{cg} - CN_{cr} + ML_{ic} \quad (4.28)$$

**Source term of the equation for mixing ratio of rain water  $q_r$  (4.17):**

$$\begin{aligned} \text{Src.}q_r = & VD_{vr} + CL_{cr} - CL_{ri} - CL_{rs} - CL_{rg} + CN_{cr} \\ & + ML_{sr} + ML_{gr} - FR_{rg} + SH_{sr} + SH_{gr} \end{aligned} \quad (4.29)$$

**Source term of the equation for mixing ratio of cloud ice  $q_i$  (4.18):**

$$\begin{aligned} \text{Src.}q_i = & NUA_{vi} + NUF_{ci} + NUC_{ci} + NUH_{ci} \\ & + VD_{vi} - CL_{ir} - CL_{is} - CL_{ig} - CN_{is} - ML_{ic} + SP_{si} + SP_{gi} \end{aligned} \quad (4.30)$$

**Source term of the equation for mixing ratio of snow  $q_s$  (4.19):**

$$\begin{aligned} \text{Src.}q_s = & -SP_{si} + VD_{vs} + CL_{cs} + CL_{rs}\alpha_{rs} + CL_{is} - CL_{sr}(1 - \alpha_{rs}) - CL_{sg} \\ & + CN_{is} - CN_{sg} - ML_{sr} - SH_{sr} \end{aligned} \quad (4.31)$$

**Source term of the equation for mixing ratio of graupel  $q_g$  (4.20):**

$$\begin{aligned} \text{Src.}q_g = & -SP_{gi} + VD_{vg} + PG_g + CL_{ri} + CL_{ir} + (CL_{rs} + CL_{sr})(1 - \alpha_{rs}) \\ & + CN_{sg} - ML_{gr} + FR_{rg} - SH_{gr} \end{aligned} \quad (4.32)$$

**Source term of the equation for number concentration of cloud ice  $\frac{N_i}{\rho}$  (4.21):**

$$\begin{aligned} \text{Src.}\frac{N_i}{\rho} = & \frac{1}{m_{i0}} NUA_{vi} + \frac{N_c}{\rho q_c} (NUF_{ci} + NUC_{ci} + NUH_{ci}) + SP_{si}^N + SP_{gi}^N \\ & + \frac{N_i}{\rho q_i} (VD_{vi} - CL_{ir} - CL_{is} - CL_{ig} - ML_{ic}) - AG_i^N - \frac{1}{m_{s0}} CN_{is} \end{aligned} \quad (4.33)$$

**Source term of the equation for number concentration of snow  $\frac{N_s}{\rho}$  (4.22):**

$$\text{Src.}\frac{N_s}{\rho} = \frac{N_s}{\rho q_s} (VD_{vs} - ML_{sr}) - CL_{sr}^N (1 - \alpha_{rs}) - CL_{sg}^N - AG_s^N + \frac{1}{m_{s0}} CN_{is} - CN_{sg}^N \quad (4.34)$$

Source term of the equation for number concentration of graupel  $\frac{N_g}{\bar{\rho}}$  (4.23):

$$\text{Src.} \frac{N_g}{\bar{\rho}} = \frac{N_g}{\bar{\rho} q_g} (V D_{vg} - M L_{gr}) + C L_{ri}^N + C L_{rs}^N (1 - \alpha_{rs}) + C N_{sg}^N + F R_{rg}^N \quad (4.35)$$

were  $\mathcal{L}_v, \mathcal{L}_s$ , and  $\mathcal{L}_f$  are latent heat with evaporation, sublimation, and melting [ $\text{J kg}^{-1}$ ], respectively.  $C_p$  is specific heat at the constant pressure [ $\text{J K kg}^{-1}$ ] in dry air.  $\Pi$  is the Exner function.  $m_{i0}$  and  $m_{s0}$  are the minimum masses of cloud ice and snow [ $\text{kg}$ ], respectively. The next table shows the meaning of each term in the above equations. Figure 4.1 shows the correlation between categories. In Section 4.2.4 conversion terms, which consist of these source terms, are formulated.

Sign	Contents
$NUA_{vi}$	Deposition or sorption nucleation
$NUF_{ci}$	Condensation-freezing nucleation
$NUC_{ci}$	Contact-freezing nucleation
$NUH_{ci}$	Homogeneous-freezing nucleation
$SP$	Secondary nucleation of ice crystals
$VD$	Vapor deposition, evaporation and sublimation
$CL$	Collision collection
$PG$	Growth of graupel with collisional collection (graupel production)
$AG$	Aggregation
$CN$	Conversion from a certain category to the other categories (conversion)
$ML$	Melting
$FR$	Freezing
$SH$	Shedding of liquid water
$SP^N$	Secondary nucleation of ice crystals for number concentration
$CL^N$	Collisional collection for number concentration (collection)
$AG^N$	Aggregation for number concentration (aggregation)
$CN^N$	Conversion from a certain category to the other categories for number concentration (conversion)
$FR^N$	Freezing for number concentration (freezing)
$\alpha_{rs}$	$1 - \alpha_{rs}$ : Production rate of graupel with the collision of raindrops and snow

There is no exchange term  $VD_{vc}$  between water vapor and an cloud water in (??), (4.27), and (4.28). The exchange term is calculated with the moist saturation adjustment method. Section 4.2.5 describes about the calculation.

### 4.2.3 Expression of Particles of Cloud and Precipitation

#### Particle-size Distribution

When we use a bulk method, we need to give particle-size distribution with a certain suitable function because only mixing ratio, or mixing ratio and number concentration can be directly calculated. A particle-size distribution is needed for the calculations of averaged mass and averaged fall velocity.

The particle-size distribution in the bulk method is often given with the exponential function. This is given by

$$\underbrace{n_x(D_x)}_{[m^{-4}]} = \underbrace{n_{x0}}_{[m^{-4}]} \exp(\underbrace{-\lambda_x D_x}_{[m^{-1}]}) \quad (4.36)$$

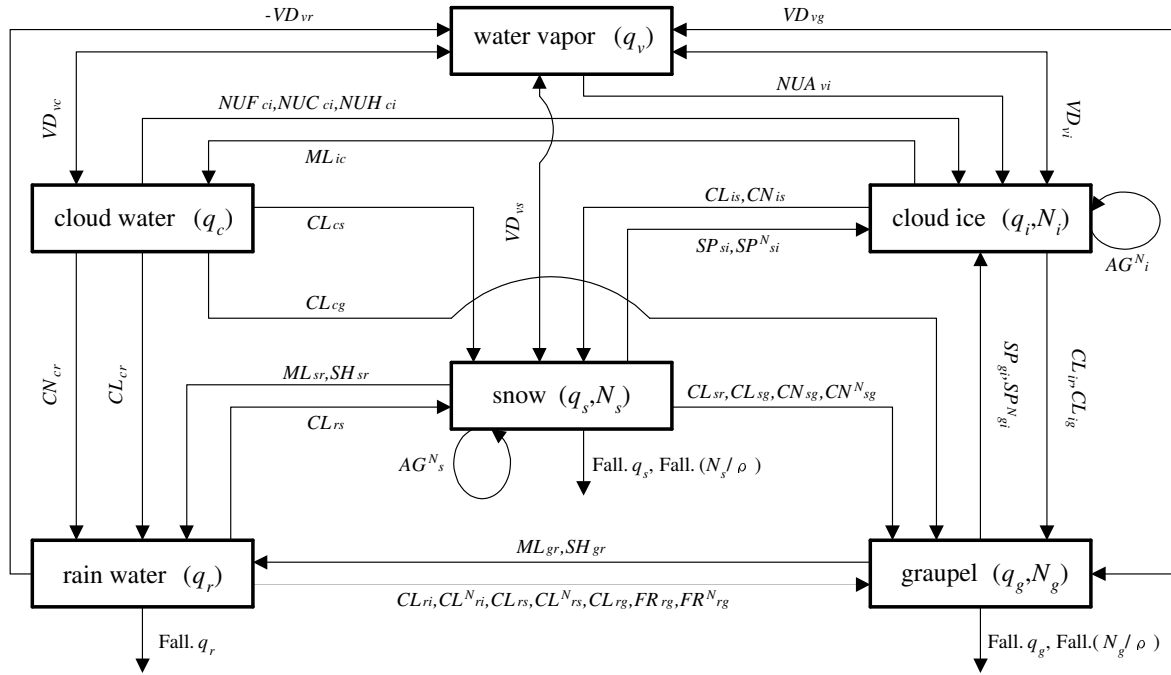


Figure 4.1. Correlation of cloud microphysical processes in a bulk method

using the parameters of an inclination in an exponential function for particle size distribution  $\lambda_x$  and  $y$ -section density  $n_{x0}$ . The parameters are based on Marshall and Palmer (1948). This distribution is sometimes called the Marshall-Palmer distribution. The distribution is sometimes expressed by using a

gamma function<sup>2</sup>

A gamma function distribution<sup>3</sup> expressing particle size distribution of a category is

$$\underbrace{f_x(D_x)}_{[m^{-1}]} = \frac{1}{\Gamma(\nu_x)} \left( \frac{D_x}{D_{nx}} \right)^{\nu_x - 1} \frac{1}{D_{nx}} \exp\left(-\frac{D_x}{D_{nx}}\right) \quad (4.43)$$

where  $D_x$  is the diameter of particle [m] and  $\Gamma(\nu_x)$  is a standardization (Integral from 0 to  $\infty$  is made to be set to 1.) constant,  $\nu_x$  is the form parameter of a gamma function and  $D_{nx}$  is a characteristic diameter. Number concentrations of particles of cloud and precipitation are expressed as

$$\underbrace{n_x(D_x)}_{[m^{-4}]} = \underbrace{n_{xt}}_{[m^{-3}]} \underbrace{f_x(D_x)}_{[m^{-1}]} \quad (4.44)$$

The averaged diameter of a particle  $\bar{D}_x$  in this distribution is

$$\bar{D}_x = \int_0^{\infty} D_x f_x(D_x) dD_x = \frac{\Gamma(\nu_x + 1)}{\Gamma(\nu_x)} D_{nx} = \nu_x D_{nx} \quad (4.45)$$

Here, the relation of (4.38) is used for the last conversion. Generally, the P'th moment of a gamma function distribution is expressed as follows (a complex number is sufficient as P).

<sup>2</sup>A gamma function is one of the special functions which extended Factorial  $n!$  even to the complex number. Using this function, we can mechanically solve a certain kind of definite integral. Especially, we often integrated using the exponential function distribution (4.36) in cloud physics. In such a case, definite integral can be solved mechanically. Although there are some definitions for the gamma function  $\Gamma(x)$ , it is defined with integration as follows.

$$\Gamma(x) = \int_0^{\infty} \exp(-t) t^{x-1} dt \quad (4.37)$$

$$\Gamma(x+1) = x\Gamma(x) \quad (4.38)$$

$$\Gamma(1) = 1 \quad (4.39)$$

where especially  $x$  is the positive integer NNN, it is set to

$$\Gamma(n+1) = n(n-1)(n-2)\cdots 2 \cdot 1 \cdot \Gamma(1) = n! \quad (4.40)$$

and is

$$\Gamma\left(\frac{1}{2}\right) = \sqrt{\pi} \quad (4.41)$$

For example, if this is used instead of carrying out integration by parts 3 times, in the definite integral of cloud physics, the following is mechanically calculable like

$$\int_0^{\infty} D_x^3 \exp(-\lambda_x D_x) dD_x = \frac{1}{\lambda_x^4} \Gamma(4) = \frac{6}{\lambda_x^4} \quad (4.42)$$

<sup>3</sup>Murakami (1999)

$$\int_0^{\infty} D_x^P f_x(D_x) dD_x = \frac{\Gamma(\nu_x + P)}{\Gamma(\nu_x)} D_{nx}^P. \quad (4.46)$$

The exponential function distribution (4.36) is one of the special form of a gamma function distribution (4.43). The distribution (4.43) is the exponential function distribution (4.36) when we set as follows.

$$\nu_x = 1 \quad (4.47)$$

$$D_{nx} = \frac{1}{\lambda_x} \quad (4.48)$$

In this case, (4.46) can be expressed as

$$\int_0^{\infty} D_x^P \lambda_x \exp(-\lambda_x D_x) dD_x = \frac{1}{\lambda_x^P} \Gamma(P+1). \quad (4.49)$$

It is an availability of a gamma function that definite integral is made simply. The average radius of a particle  $\bar{D}_x$  is

$$\bar{D}_x = \frac{1}{\lambda_x}. \quad (4.50)$$

A single distribution is used for an cloud water and cloud ice, and the exponential function distribution<sup>4</sup> is used for precipitation particles of rain water, snow, and graupel. Here, the averaged diameters of cloud water and cloud ice are

$$\bar{D}_c = \left( \frac{6\bar{\rho}q_c}{\pi\rho_w N_c} \right)^{\frac{1}{3}} \quad (4.51)$$

$$\bar{D}_i = \left( \frac{6\bar{\rho}q_i}{\pi\rho_i N_i} \right)^{\frac{1}{3}}. \quad (4.52)$$

where  $\rho_w$  is density of cloud water [ $\text{kg m}^{-3}$ ] and  $\rho_i$  is density of cloud-ice [ $\text{kg m}^{-3}$ ]. The constant value  $1 \times 10^8 \text{ m}^{-3}$  is used for the number concentration of cloud water  $N_c$ .

Particle size distributions of rain water, snow, and graupel are given as

$$n_r(D_r) = n_{r0} \exp(-\lambda_r D_r) \quad (4.53)$$

$$n_s(D_s) = n_{s0} \exp(-\lambda_s D_s) \quad (4.54)$$

$$n_g(D_g) = n_{g0} \exp(-\lambda_g D_g). \quad (4.55)$$

Note that the unit of  $n_x$  is [ $\text{m}^{-4}$ ].

<sup>4</sup>The exponential function distribution is used by Murakami (1999), Ikawa and Saito (1991), Murakami et al. (1994), Ikawa et al. (1991), Lin et al. (1983). Ferrier (1994) uses the exponential function distribution for cloud water and the gamma function distribution for the other precipitation particles.

### Averaged Mass and Averaged Fall Velocity

The mass of a particle of cloud or precipitation, which depends on the form of the particle, is given by the empirical equation of an exponentiation of a particle diameter.

$$m_x(D_x) = \alpha_{ux} D_x^{\beta_{ux}} \quad (4.56)$$

$\beta_{ux} = 3$  in the case of a spherical particle. Using a gamma function distribution, the averaged mass is

$$\bar{m}_x = \alpha_{ux} D_{nx}^{\beta_{ux}} \frac{\Gamma(\nu_x + \beta_{ux})}{\Gamma(\nu_x)}. \quad (4.57)$$

The fall velocity of a particle is also given by the empirical equation of the exponentiation of a diameter.

$$U_x(D_x) = \alpha_{ux} D_x^{\beta_{ux}} \left( \frac{\rho_0}{\bar{\rho}} \right)^{\gamma_{ux}} \quad (4.58)$$

In the same way, using a gamma function distribution, the averaged fall velocity of which the weight with the particle number concentration per unit diameter [ $\text{m}^{-4}$ ] is multiplied is

$$\bar{U}_{xN} = \alpha_{ux} D_{nx}^{\beta_{ux}} \frac{\Gamma(\nu_x + \beta_{ux})}{\Gamma(\nu_x)} \left( \frac{\rho_0}{\bar{\rho}} \right)^{\gamma_{ux}}. \quad (4.59)$$

The averaged fall velocity of which the weight of mass is multiplied is

$$\bar{U}_{xq} = \alpha_{ux} D_{nx}^{\beta_{ux}} \frac{\Gamma(\nu_x + 3 + \beta_{ux})}{\Gamma(\nu_x + 3)} \left( \frac{\rho_0}{\bar{\rho}} \right)^{\gamma_{ux}}. \quad (4.60)$$

These relations include an exponential function distribution as a special case. Using the relational expression with a gamma function in the cases of (4.47), (4.60) and (4.59) are

$$\bar{U}_{xN} = \alpha_{ux} \frac{\Gamma(1 + \beta_{ux})}{\lambda_x^{\beta_{ux}}} \left( \frac{\rho_0}{\bar{\rho}} \right)^{\gamma_{ux}} \quad (4.61)$$

$$\bar{U}_{xq} = \alpha_{ux} \frac{\Gamma(4 + \beta_{ux})}{6\lambda_x^{\beta_{ux}}} \left( \frac{\rho_0}{\bar{\rho}} \right)^{\gamma_{ux}}. \quad (4.62)$$

where  $\rho_0$  is the air density at the surface in the basic state [ $\text{kg m}^{-3}$ ].

In the exponential function distribution<sup>5</sup> (4.53)~(4.55), the total particle number for categories  $x = r, s, g$ :  $N_x$  is

$$N_x = \int_0^\infty n_{x0} \exp(-\lambda_x D_x) dD_x = \frac{n_{x0}}{\lambda_x}. \quad (4.63)$$

The total mass of water substance  $x$  per unit volume is

<sup>5</sup>Ikawa and Saito (1991)

$$\bar{\rho}q_x = \int_0^\infty \frac{\pi}{6} \rho_x D_x^3 n_{x0} \exp(-\lambda_x D_x) dD_x = \frac{\pi \rho_x n_{x0}}{\lambda_x^4}. \quad (4.64)$$

From these two equations, an inclination parameter in an exponential function particle size distribution  $\lambda_x$  and the  $y$ -section density  $n_{x0}$  are

$$\lambda_x = \left( \frac{\pi \rho_x N_x}{\bar{\rho} q_x} \right)^{\frac{1}{3}} \quad (4.65)$$

$$n_{x0} = N_x \left( \frac{\pi \rho_x N_x}{\bar{\rho} q_x} \right)^{\frac{1}{3}}. \quad (4.66)$$

Then, the terminal fall velocity of particle in category  $x$  which is weighted by number concentration is

$$\begin{aligned} \bar{U}_{xN} &= \frac{1}{N_x} \int_0^\infty U_x(D_x) n_{x0} \exp(-\lambda_x D_x) dD_x \\ &= \alpha_{ux} \frac{\Gamma(1 + \beta_{ux})}{\lambda_x^{\beta_{ux}}} \left( \frac{\rho_0}{\bar{\rho}} \right)^{\gamma_{ux}} \end{aligned} \quad (4.67)$$

and this is consistent with (4.61). In the same way, the mass-weighted terminal fall speed of category  $x$  is

$$\begin{aligned} \bar{U}_{xq} &= \frac{1}{\bar{\rho} q_x} \int_0^\infty \frac{\pi}{6} U_x(D_x) D_x^3 \rho_x n_{x0} \exp(-\lambda_x D_x) dD_x \\ &= \alpha_{ux} \frac{\Gamma(4 + \beta_{ux})}{6 \lambda_x^{\beta_{ux}}} \left( \frac{\rho_0}{\bar{\rho}} \right)^{\gamma_{ux}}. \end{aligned} \quad (4.68)$$

and this is consistent with (4.62). These averaged fall velocities are used for calculations of change of mixing ratio and number concentration in association with a falling, as described in Section 4.2.6.

Finally, the form parameter of each category is summarized below.

variable	the density of $y$ -section [m <sup>-4</sup> ]	the form parameter of fall speed	density [kg m <sup>-3</sup> ]
$q_c$	—	$(\alpha_{uc} = 2.98 \times 10^7, \beta_{uc} = 2.0, \gamma_{uc} = 1.0)$	$\rho_w = 1.0 \times 10^3$
$q_r$	$n_{r0} = 8.0 \times 10^6$	$\alpha_{ur} = 842, \beta_{ur} = 0.8, \gamma_{ur} = 0.5$	$\rho_w = 1.0 \times 10^3$
$q_i$	—	$(\alpha_{ui} = 700, \beta_{ui} = 1.0, \gamma_{ui} = 0.33)$	$\rho_i = 5.0 \times 10^2$
$q_s$	$(n_{s0} = 1.8 \times 10^6)$	$\alpha_{us} = 17, \beta_{us} = 0.5, \gamma_{us} = 0.5$	$\rho_s = 8.4 \times 10^1$
$q_g$	$(n_{g0} = 1.1 \times 10^6)$	$\alpha_{ug} = 124, \beta_{ug} = 0.64, \gamma_{ug} = 0.5$	$\rho_g = 3.0 \times 10^2$

#### 4.2.4 Formulations of Physical Processes of Source terms

Hereafter, we will explain physical processes of source terms. Signs are summarized into table in each section (Sections 4.2.5 and 4.2.7 are the same as this section).



**Nuclear Formation of Primary Ice :**  $NUA_{vi}, NUF_{ci}, NUC_{ci}, NUH_{ci}$ 

The primary ice nucleation are shown as below.

homogeneous nucleation	sublimation nucleation		water vapor → ice crystals
	freezing nucleation		supercooled water drops → ice crystals
heterogeneous nucleation	sublimation nucleation	sublimation nuclei	water vapor → ice crystals
	freezing nucleation	sorption nuclei	
		contact-freezing nuclei	supercooled water drops → ice crystals
		immersion-freezing nuclei	supercooled water drops → ice crystals

Our model adopts  $NUA_{vi}, NUF_{ci}, NUC_{ci}, NUH_{ci}$ .

**(1) Sublimation Nucleation:**  $NUA_{vi}$ **(a) Number Concentration of Sublimation Nuclei as a Function of Supercooling Temperature**<sup>6</sup>

Number concentration of sublimation nuclei as a function of supercooled temperature  $T_s$  is written by

$$NUA_{vi} = \frac{m_{i0}}{\bar{\rho}} \beta_2 N_{i0} \exp(\beta_2 T_s) \left( \frac{S_i - 1}{S_{wi} - 1} \right)^B \frac{\partial T_s}{\partial z} w \quad (4.69)$$

$$NUA_{vi}^N = \frac{NUA_{vi}}{m_{i0}}. \quad (4.70)$$

Note that a difference in  $z^*$  coordinate system ( $\zeta$  coordinate system) is multiplied by a matrix, as shown in equations (2.58) since the vertical difference is a difference in a real space.

**(b) Number Concentration of Sublimation Nuclei as a Function of Supersaturation**<sup>7</sup>

Number concentration of sublimation nuclei as a function of supersaturation is written by

$$NUA_{vi} = \frac{m_{i0}}{\bar{\rho}} 15.25 \exp(5.17 + 15.25 SS_i) \frac{\partial SS_i}{\partial z} w \quad (4.71)$$

$$NUA_{vi}^N = \frac{NUA_{vi}}{m_{i0}} \quad (4.72)$$

Note that a difference in  $z^*$  coordinate system ( $\zeta$  coordinate system) is multiplied by a matrix, as well as the same way in (a).

**(c) Considering both Supercooling Temperature and Supersaturation**<sup>8</sup>

Ferrier (1994) adopted a method that uses two equations owing to temperature for heterogeneous sublimation nucleation. In the method, the equation of Murakami (1990), Cotton et al. (1986) is used when temperature is more than  $-5$  °C, and that of Meyers et al. (1992) is used when temperature is less than  $-5$  °C.

$$NUA_{vi} = \frac{m_{i0}}{\bar{\rho}} w \frac{\partial N_i}{\partial z} \quad (4.73)$$

$$NUA_{vi}^N = \frac{NUA_{vi}}{m_{i0}} \quad (4.74)$$

<sup>6</sup>Ikawa and Saito (1991), Cotton et al. (1986), Murakami (1990), Ikawa et al. (1991), Murakami et al. (1994), 村上 (1999)

<sup>7</sup>Meyers et al. (1992), 村上 (1999)

<sup>8</sup>Ferrier (1994)

where  $N_i$  is divided by  $-5$  °C as follows.

$$N_i = \begin{cases} N_{i01} \exp(\beta_2 T_s) \left( \frac{S_i - 1}{S_{wi} - 1} \right)^B, & T \geq -5 \text{ [}^\circ\text{C]} \\ N_{i02} \exp(a_1 S S_i - b_1), & T < -5 \text{ [}^\circ\text{C]} \end{cases} \quad (4.75)$$

Note that a difference in  $z^*$  coordinate system ( $\zeta$  coordinate system) is multiplied by a matrix, as well as the same way in (a) and (b).

Meanings of the signs used in (a)~(c) are as follows.

$a_1$	Coefficient of the Ferrier equation in below $-5$ °C	12.96	
$b_1$	Coefficient of the Ferrier equation in below $-5$ °C	0.639	
$B$	Coefficient of the Huffmann and Vail equation	4.5	
$m_{i0}$	Minimum mass of cloud ice	$10^{-12}$	kg
$N_{i0}$	Coefficient of Fletcher equation	$10^{-2}$	$\text{m}^{-3}$
$N_{i01}$	Number of particles of the Ferrier equation in more than $-5$ °C	$10^3$	$\text{m}^{-3}$
$N_{i02}$	Number of particles of the Ferrier equation in below $-5$ °C	50	$\text{m}^{-3}$
$q_{vsi}$	Saturation mixing ratio over ice		$\text{kg kg}^{-1}$
$q_{vsw}$	Saturation mixing ratio over water		$\text{kg kg}^{-1}$
$T$	Temperature		K
$T_0$	Melting point of ice	273.16	K
$T_s$	Supercooling temperature ( $T_0 - T$ )		K
$S_i$	Ratio of mixing ratio of water vapor and saturation mixing ratio of ice in an air parcel		
$S_{wi}$	Ratio of the saturation mixing ratios of water and ice		
$SS_i$	Ice supersaturation ( $S_i - 1$ )		
$w$	Vertical motion in $z$ coordinate system		$\text{m s}^{-1}$
$\beta_2$	Coefficient of the Fletcher equation	0.6	$\text{K}^{-1}$
$\bar{\rho}$	Air density in the basic state		$\text{kg m}^{-3}$

## (2) Immersion Freezing Nucleation: $NUF_{ci}$

Heterogeneous nucleation of cloud droplets depends on a size, physical and chemical features of freezing nucleus and temperature and size of a cloud drop. We use the nucleation that is extrapolated outside to the size of a cloud droplet with the Bigg (1953) empirical equation<sup>9</sup>.

$$NUF_{ci} = B' [\exp(A'T_s) - 1] \frac{\bar{\rho} q_c^2}{\rho_w N_c} \quad (4.76)$$

$$NUF_{ci}^N = B' [\exp(A'T_s) - 1] \frac{q_c}{\rho_w} \quad (4.77)$$

Meanings of the signs used here are as follows.

$A'$	Coefficient of the Bigg empirical equation	0.66	$\text{K}^{-1}$
$B'$	Coefficient of the Bigg empirical equation	100.0	$\text{m}^{-3} \text{s}^{-1}$
$N_c$	Number concentration of cloud droplets	$1 \times 10^8$	$\text{m}^{-3}$
$T$	Temperature		K
$T_0$	Melting point of ice	273.16	K

<sup>9</sup>村上 (1999), Ikawa and Saito (1991)

---

$T_s$	Supercooled temperature ( $T_0 - T$ )		K
$\bar{\rho}$	Air density in the basic state		kg m <sup>-3</sup>
$\rho_w$	Density of water	$1 \times 10^3$	kg m <sup>-3</sup>

---

### (3) Contact Freezing Nucleation : $NUC_{ci}$

The contact freezing nucleation is occurred with a collision of an ice particle core (freezing core) and a supercooled cloud droplet. The following three processes are considered to it.

- Brownian diffusion
- diffusiophoresis
- thermophoresis

The generation rate of ice particles by these processes is given by

$$\left[ \frac{dN_c}{dt} \right]_b = F_1 \mathcal{D}_{ar} \quad (4.78)$$

$$\left[ \frac{dN_c}{dt} \right]_v = F_1 F_2 \frac{R_v T}{\mathcal{L}_v} \quad (4.79)$$

$$\left[ \frac{dN_c}{dt} \right]_t = F_1 F_2 f_t \quad (4.80)$$

where

$$F_1 = 2\pi D_c N_c N_{ar} \quad (4.81)$$

$$F_2 = \frac{\kappa_a}{p} (T - T_{cl}) \quad (4.82)$$

$$f_t = \frac{0.4 [1 + 1.45K_n + 0.4 \exp(-1/K_n)] (\kappa + 2.5K_n\kappa_a)}{(1 + 3K_n)(2\kappa + 5\kappa_a K_n + \kappa_a)}. \quad (4.83)$$

$K_n$  in the equations is the Knudsen number which is defined as

$$K_n = \lambda_{a0} \frac{p_{00}}{T_{00} R_a} \frac{T}{p}. \quad (4.84)$$

The diffusion coefficient of an aerosol particle is

$$\mathcal{D}_{ar} = \frac{kT_{cl}}{6\pi R_a \mu} (1 + K_n). \quad (4.85)$$

The number concentration  $N_{ar}$  of the contact freezing nuclei activated by temperature  $T_{cl}$ <sup>10</sup> is

$$N_{ar} = N_{a0} (270.16 - T_{cl})^{1.3}. \quad (4.86)$$

The formation of an ice particle with the contact freezing nuclei is finally represented by the sum of the three processes, as shown in follows.

$$NUC_{ci}^N = \frac{1}{\bar{\rho}} \left( \left[ \frac{dN_c}{dt} \right]_b + \left[ \frac{dN_c}{dt} \right]_v + \left[ \frac{dN_c}{dt} \right]_t \right) \quad (4.87)$$

$$NUC_{ci} = \frac{\bar{\rho} q_c}{N_c} NUC_{ci}^N \quad (4.88)$$

Meanings of the signs used here are as follows.

$D_c$	Diameter of a cloud droplet		m
$k$	The Boltzmann number	$1.380658 \times 10^{-23}$	J K <sup>-1</sup>
$\mathcal{L}_v$	Latent heat of evaporation of water		J kg <sup>-1</sup>
$N_{a0}$	Coefficient in the number concentration of contact freezing nuclei	$2 \times 10^5$	m <sup>-3</sup>
$N_c$	Number concentration of cloud droplets	$1 \times 10^8$	m <sup>-3</sup>
$p$	Pressure		Pa
$p_{00}$	Standard pressure	101325	Pa
$R_a$	Radius of an aerosol particle	$3 \times 10^{-7}$	m
$R_v$	The gas constant of water vapor	461.0	J K <sup>-1</sup> kg <sup>-1</sup>

<sup>10</sup>Cotton et al. (1986)

$T$	Temperature		K
$T_{00}$	Temperature in the basic state	293.15	K
$T_{cl}$	Temperature of a cloud droplet		K
$\kappa$	Thermal conductivity of air	$2.4 \times 10^{-2}$	$\text{J m}^{-1} \text{s}^{-1} \text{K}^{-1}$
$\kappa_a$	Thermal conductivity of an aerosol particle		$\text{J m}^{-1} \text{s}^{-1} \text{K}^{-1}$
$\lambda_{a0}$	Mean free path at $p_{00}, T_{00}$	$6.6 \times 10^{-8}$	m
$\mu$	Coefficient of viscosity of air		$\text{kg m}^{-1} \text{s}^{-1}$
$\bar{\rho}$	Air density in the basic state		$\text{kg m}^{-3}$

#### (4) Homogeneous Freezing Nuclei : $NUH_{ci}$

We assume that cloud water freezes momentarily when temperature is below  $-40$  °C <sup>11</sup>. Rate of the homogeneous freezing nuclei is written by

$$NUH_{ci}^N = \frac{1}{\bar{\rho}} \frac{N_c}{2\Delta t} \quad (4.89)$$

$$NUH_{ci} = \frac{q_c}{2\Delta t} \quad (4.90)$$

where  $\Delta t$  is the time step in the leap frog method. Note that generation of cloud water must be given by the moist saturation adjustment before this calculation (Ferrier, 1994).

#### Secondary Nucleation of Ice Crystals: $SP$

Although there are still many points which are not known about the nucleation of secondary ice crystals, the following processes are mainly known.

- While snow and graupel grow up with collection of cloud droplets, minute particles of ice are generated (Hallett and Mossop, 1974).
- Snow and graupel collide each other while they are falling, and the fragment of small ice is generated (Vardiman, 1978).
- When the large supercooled water droplet, which is generated near the cloud top, freezes, ice crystals of high-concentration are generated (Hobbs and Rangno, 1985).

We consider only the Hallett-Mossop rime splintering mechanism<sup>12</sup>. Generation rates of the secondary nucleation of ice crystals are

$$SP_{si}^N = \frac{1}{\bar{\rho}} \times 3.5 \times 10^8 f(T_s) CL_{cs} \quad (4.91)$$

$$SP_{si} = m_{i0} SP_{si}^N \quad (4.92)$$

$$SP_{gi}^N = \frac{1}{\bar{\rho}} \times 3.5 \times 10^8 f(T_g) CL_{cg} \quad (4.93)$$

$$SP_{gi} = m_{i0} SP_{gi}^N. \quad (4.94)$$

Note that the secondary nucleation of ice crystals has never occurred in the wet growth of graupel. Function of the temperature of a particle  $f(T_x)$  ( $T_x$  is  $T_s$  or  $T_g$ ) is defined as follows.

<sup>11</sup> Ikawa and Saito (1991), Ferrier (1994)

<sup>12</sup> 村上 (1999), Ikawa and Saito (1991), Cotton et al. (1986)

$$f(T_x) = \begin{cases} 0, & T_x > 270.16 [K] \\ \frac{T_x - 268.16}{2}, & 268.16 \leq T_x \leq 270.16 [K] \\ \frac{268.16 - T_x}{3}, & 265.16 \leq T_x \leq 268.16 [K] \\ 0, & T_x < 265.16 [K] \end{cases} \quad (4.95)$$

Although  $f(T_x)$  will become a negative value in (72) of Cotton et al. (1986), if an absolute value is taken, it will become the same as the equation (4.95). Although (4.95) is a function that it may be  $f(T_x) = 0$  at  $T_x = 268.16 [K]$ , Ikawa et al. (1991), Ikawa and Saito (1991) use a function that it may be  $f(T_x) = 1$  at  $T_x = 268.16[K]$  as follows.

$$f(T_x) = \begin{cases} 0, & T_x \geq 270.16 [K] \\ \frac{270.16 - T_x}{2}, & 268.16 < T_x < 270.16 [K] \\ 1, & T_x = 268.16 [K] \\ \frac{T_x - 265.16}{3}, & 265.16 \leq T_x < 268.16 [K] \\ 0, & T_x < 265.16 [K] \end{cases} \quad (4.96)$$

This function has the maximum value at  $-5^\circ\text{C}$  and is considered to be more reasonable.

Meanings of the signs used here are as follows.

$CL_{cg}$	Growth rate of graupel in the collision with cloud droplets		$\text{s}^{-1}$
$CL_{cs}$	Growth rate of snow in the collision with cloud droplets		$\text{s}^{-1}$
$m_{i0}$	The mass of the minimum cloud ice	$10^{-12}$	kg
$T_s$	Temperature of snow		K
$T_g$	Temperature of graupel		K
$T_x$	Temperature of snow or graupel		K
$\bar{\rho}$	Air density in the basic state		$\text{kg m}^{-3}$

#### Water Vapor Diffusion Growth : $VD$

Diffusion growth is the generation, the growth, disappearance, and consumption by exchange of the water molecule between moisture and a particle. There are the following processes and some of them are considered in our model.

gaseous phase – liquid phase	condensation	water vapor $\rightarrow$ cloud water	calculated by the moist saturation a
		water vapor $\rightarrow$ rain water	neglected because of small amount
	evaporation	cloud water $\rightarrow$ water vapor	calculated by the moist saturation a
		rain water $\rightarrow$ water vapor	$VD_{vr} < 0$
gaseous phase – solid phase	sublimation	water vapor $\rightarrow$ cloud ice	$VD_{vi} > 0$
		water vapor $\rightarrow$ snow	$VD_{vs} > 0$
		water vapor $\rightarrow$ graupel	$VD_{vg} > 0$
		water vapor $\rightarrow$ hail	$VD_{vh} > 0$
	sublimation evaporation	cloud ice $\rightarrow$ water vapor	$VD_{vi} < 0$
		snow $\rightarrow$ water vapor	$VD_{vs} < 0$

	graupel $\rightarrow$ water vapor	$VD_{vg} < 0$
	hail $\rightarrow$ water vapor	$VD_{vh} < 0$

**(1)Evaporation of Rain Water :**  $VD_{rv}$ 

Variations of mixing ratio and number concentration of rain water <sup>13</sup> are represented by

$$VD_{vr} = -VD_{rv} = \frac{2\pi}{\bar{\rho}} (S_w - 1) G_w(T, p) VENT_r \quad (4.97)$$

where

$$G_w(T, p) = \left( \frac{\mathcal{L}_v^2}{\kappa R_v T^2} + \frac{1}{\bar{\rho} q_{vs w} \mathcal{D}_v} \right)^{-1}. \quad (4.98)$$

The term of the ventilation coefficient of rain water is represented by

$$VENT_r = n_{r0} \left[ 0.78 \lambda_r^{-2} + 0.31 S_c^{\frac{1}{3}} \nu^{-\frac{1}{2}} \alpha_{ur}^{\frac{1}{2}} \Gamma \left( \frac{5 + \beta_{ur}}{2} \right) \lambda_r^{-\frac{(5 + \beta_{ur})}{2}} \left( \frac{\rho_0}{\bar{\rho}} \right)^{\frac{1}{4}} \right]. \quad (4.99)$$

**(2)Sublimation Condensation of Snow and Graupel (Sublimation Evaporation) :**  $VD_{vs}, VD_{vg}$   
<sup>14</sup>

Sublimation condensation (sublimation evaporation) rates of snow and graupel are formulized like rain water. However, they are divided in the cases more and less than the melting point of ice. Here, snow and graupel are simultaneously formulized as  $x = s, g$ . That is,

considering the latent heat of freezing in a collecting supercooled cloud droplets at  $T < T_0$ ,

$$VD_{vx} = \frac{2\pi}{\bar{\rho}} (S_i - 1) G_i(T, p) VENT_x - \frac{\mathcal{L}_s \mathcal{L}_f}{\kappa R_v T^2} G_i(T, p) CL_{cx}. \quad (4.100)$$

At  $T > T_0$ ,

when a melting process occurs,

$$VD_{vx} = 2\pi \mathcal{D}_v (q_v - q_{vs}(T_0)) VENT_x \quad (4.101)$$

when a melting process does not occur,

$$VD_{vx} = \frac{2\pi}{\bar{\rho}} (S_w - 1) G_w(T, p) VENT_x. \quad (4.102)$$

<sup>13</sup>村上 (1999)

<sup>14</sup>村上 (1999), Ikawa and Saito (1991), Lin et al (1983)

where

$$G_i(T, p) = \left( \frac{\mathcal{L}_s^2}{\kappa R_v T^2} + \frac{1}{\bar{\rho} q_{vsi} \mathcal{D}_v} \right)^{-1} \quad (4.103)$$

where  $G_w(T, p)$  is given by the equation (4.98). A ventilation coefficient at  $x = s, g$  is as follows.

$$VENT_x = n_{x0} \left[ 0.78 \lambda_x^{-2} + 0.31 S_c^{\frac{1}{3}} \nu^{-\frac{1}{2}} \alpha_{ux}^{\frac{1}{2}} \Gamma \left( \frac{5 + \beta_{ux}}{2} \right) \lambda_x^{-\frac{(5 + \beta_{ux})}{2}} \left( \frac{\rho_0}{\bar{\rho}} \right)^{\frac{1}{4}} \right] \quad (4.104)$$

### (3) Growth of Cloud Ice in Gaseous Phase : $VD_{vi}$ <sup>15</sup>

Variation of growth of cloud ice in gaseous phase is represented as follows.

$$VD_{vi} = \frac{q_v - q_{vsi}}{q_{vsw} - q_{vsi}} a_1 (\bar{m}_i)^{a_2} \frac{N_i}{\bar{\rho}} \quad (4.105)$$

where  $a_1, a_2$  is a parameter depending on temperature (König (1971)) and  $\bar{m}_i$  is mean mass of cloud ice given as following equation.

$$\bar{m}_i = \frac{q_i \bar{\rho}}{N_i} \quad (4.106)$$

---

<sup>15</sup>Ikawa and Saito (1991), Ikawa et al. (1991)



Meanings of the signs used in (1)-(3) are as follows.

$CL_{cg}$	Growth rate of graupel in coalescence with cloud droplets		$s^{-1}$
$CL_{cs}$	Growth rate of snow in coalescence with cloud droplets		$s^{-1}$
$\mathcal{D}_v$	Diffusion coefficient of water vapor		$m^2 s^{-1}$
$\mathcal{L}_f$	Latent heat of melting of water		$J kg^{-1}$
$\mathcal{L}_s$	Latent heat of sublimation of water		$J kg^{-1}$
$\mathcal{L}_v$	Latent heat of evaporation of water		$J kg^{-1}$
$n_{g0}$	Density of graupel in the $y$ section		$m^{-4}$
$n_{r0}$	Density of rain water in the $y$ section	$8.0 \times 10^6$	$m^{-4}$
$n_{s0}$	Density of snow in the $y$ section		$m^{-4}$
$q_{vs}(T_0)$	Saturation mixing ratio of the melting point of water		$kg kg^{-1}$
$q_{vsi}$	Ice saturation mixing ratio		$kg kg^{-1}$
$q_{vsw}$	Water saturation mixing ratio		$kg kg^{-1}$
$R_v$	The gas constant of water vapor	461.0	$J K^{-1} kg^{-1}$
$S_c$	Schmidt number	0.6	
$S_i - 1$	Ice supersaturation of an air parcel		
$S_w - 1$	Water supersaturation of an air parcel		
$T$	Temperature		K
$T_0$	Melting point of ice	273.16	K
$\alpha_{ug}$	Coefficient in the relational expression of fall velocity and diameter of graupel	124	$m^{1-\beta_{ug}} s^{-1}$
$\alpha_{ur}$	Coefficient in the relational expression of fall velocity and diameter of rain water	842	$m^{1-\beta_{ur}} s^{-1}$
$\alpha_{us}$	Coefficient in the relational expression of fall velocity and diameter of snow	17	$m^{1-\beta_{us}} s^{-1}$
$\beta_{ug}$	Coefficient in the relational expression of fall velocity and diameter of graupel	0.64	
$\beta_{ur}$	Coefficient in the relational expression of fall velocity and diameter of rain water	0.8	
$\beta_{us}$	Coefficient in the relational expression of fall velocity and diameter of snow	0.5	
$\lambda_g$	Inclination in a reversed exponential function expressing particle size distribution of graupel		$m^{-1}$
$\lambda_r$	Inclination in a reversed exponential function expressing particle size distribution of rain water		$m^{-1}$
$\lambda_s$	Inclination in a reversed exponential function expressing particle size distribution of snow		$m^{-1}$
$\kappa$	Thermal conductivity of air	$2.4 \times 10^{-2}$	$J m^{-1} s^{-1} K^{-1}$
$\nu$	Coefficient of kinematic viscosity of air		$m^2 s^{-1}$
$\bar{\rho}$	Air density in the basic state		$kg m^{-3}$
$\rho_0$	Density at the surface in the basic state		$kg m^{-3}$

### Particle Collection: $CL$

In this section, we formulize the growth process of the particle which captures a certain particle by the collision between various particles. For the particle collision, the following processes can be respectively considered to the categories of the particles which are defined before.

process	sink	source	sign	meaning	model
Collection of cloud water	$q_c$	$q_i$	$CL_{ci}$	Growth of cloud ice by collection with cloud water	$\times$
	$q_c$	$q_s$	$CL_{cs}$	Growth of snow by collection with cloud water	$\bigcirc$
	$q_c$	$q_g$	$CL_{cg}$	Growth of graupel by collection with cloud water	$\bigcirc$
	$q_c$	$q_h$	$CL_{ch}$	Growth of hail by collection with cloud water	$\triangle$
	$q_c$	$q_r$	$CL_{cr}$	Growth of rain water by collection with cloud water	$\bigcirc$

Collection of rain water	$q_r$	$q_i$	$CL_{ri}$	Growth of cloud ice by collection with rain water	○
	$q_r$	$q_s$	$CL_{rs}$	Growth of snow by collection with rain water	○
	$q_r$	$q_g$	$CL_{rg}$	Growth of graupel by collection with rain water	○
	$q_r$	$q_h$	$CL_{rh}$	Growth of hail by collection with rain water	△
Collection of cloud ice	$q_i$	$q_r$	$CL_{ir}$	Growth of rain water by collection with cloud ice	○
	$q_i$	$q_s$	$CL_{is}$	Growth of snow by collection with cloud ice	○
	$q_i$	$q_g$	$CL_{ig}$	Growth of graupel by collection with cloud ice	○
	$q_i$	$q_h$	$CL_{ih}$	Growth of hail by collection with cloud ice	△
Collection of snow	$q_s$	$q_r$	$CL_{sr}$	Growth of rain water by collection with snow	○
	$q_s$	$q_g$	$CL_{sg}$	Growth of graupel by collection with snow	○
	$q_s$	$q_h$	$CL_{sh}$	Growth of hail by collection with snow	△

In this table, "○" and "×" denote the signs of which it is included or is not included in our model, respectively, and "△" is the sign of suspending for including in our model at present.

The number concentration will change for the collection of snow particles (aggregation) and that of rain water (accretion) but the mixing ratio do not change. These are described in the other section.

In the following section, we formulize the coalescences between large particles (rain water, snow, and graupel), which have a sufficient value of fall velocity, a large particle and a small particle (cloud water and cloud ice) of which its fall velocity can be neglected, and the growth of graupel by collection of supercooled rain water and ice crystals<sup>16</sup>.

**(1) Coalescence between Rain Water, Snow and Graupel :**  $CL_{xy}$  ( $x, y = r, s, g; x \neq y$ )

Variations of mixing ratio and number concentration by the collection between precipitation particles (rain water, snow, and hail) are formulized as follows.

$$CL_{xy} = \pi^2 \frac{\rho_x}{\rho} E_{xy} \sqrt{(\bar{U}_x - \bar{U}_y)^2 + \alpha \bar{U}_x \bar{U}_y n_{x0} n_{y0}} \left( \frac{5}{\lambda_x^6 \lambda_y} + \frac{2}{\lambda_x^5 \lambda_y^2} + \frac{0.5}{\lambda_x^4 \lambda_y^3} \right) \quad (4.107)$$

$$CL_{xy}^N = \frac{\pi}{2\rho} E_{xy} \sqrt{(\bar{U}_x - \bar{U}_y)^2 + \alpha \bar{U}_x \bar{U}_y n_{x0} n_{y0}} \left( \frac{1}{\lambda_x^3 \lambda_y} + \frac{1}{\lambda_x^2 \lambda_y^2} + \frac{1}{\lambda_x \lambda_y^3} \right) \quad (4.108)$$

where  $x, y = r, s, g; x \neq y$ . Meanings of the signs used are as follows.

$E_{xy}$	Collection efficiency of a particle	
$n_{x0}$	Density of a category $x$ in the $y$ section	$m^{-4}$
$n_{y0}$	Density of a category $y$ in the $y$ section	$m^{-4}$
$\bar{U}_x$	Mean fall velocity weighted by a mass of a category $x$	$m \text{ s}^{-1}$
$\bar{U}_y$	Mean fall velocity weighted by mass of a category $y$	$m \text{ s}^{-1}$
$\alpha$	Coefficient of an adjustment term	0.04
$\lambda_x$	Inclination in a reversed exponential function expressing particle size distribution of a category $x$	$m^{-1}$
$\lambda_y$	Inclination in a reversed exponential function expressing particle size distribution of a category $y$	$m^{-1}$
$\bar{\rho}$	Air density in the basic state	$\text{kg m}^{-3}$
$\rho_x$	Density of a particle of a category $x$	$\text{kg m}^{-3}$

<sup>16</sup>Murakami (1999), Lin et al. (1983), Murakami (1990), Ikawa and Saito (1991)

(2) Collection of Rain water, Snow, and Graupel with Cloud Water and Cloud Ice:  $CL_{cy}, CL_{iy}$  ( $y = r, s, g$ )

Because fall velocities of cloud water and cloud ice are relatively small to those of rain water, snow and graupel, the collection is represented as follows.

$$CL_{xy} = \frac{\pi}{4} \bar{E}_{xy} n_{y0} q_x \alpha_{uy} \Gamma(3 + \beta_{uy}) \lambda_y^{-(3+\beta_{uy})} \left( \frac{\rho_0}{\bar{\rho}} \right)^{\frac{1}{2}} \quad (4.109)$$

The mean collection efficiency for coalescence with cloud ice  $\bar{E}_{iy}$  is constant, and that of coalescence with cloud water  $\bar{E}_{cy}$  is given by

$$\bar{E}_{cy} = \frac{Stk^2}{(Stk + 0.5)^2} \quad (4.110)$$

where  $Stk$  is the Stokes number which is given with the average radius of cloud water, cloud ice, and precipitation particle. Ikawa and Saito (1991) calculates as follows.

$$Stk = \bar{D}_c^2 \rho_w \frac{\bar{U}_y}{9\mu \bar{D}_y} \quad (4.111)$$

Meanings of the signs used here are as follows.

$D_c$	Diameter of a particle of cloud water	m
$D_y$	Diameter of a particle of a category $y$	m
$n_{y0}$	Density of a category $y$ in the $y$ section	$\text{m}^{-4}$
$\bar{U}_y$	Mean fall velocity multiplied by an weight coefficient of mass of a category $y$	$\text{m s}^{-1}$
$\alpha_{uy}$	Coefficient in the relational expression of fall velocity and diameter of a category $y$	$\text{m}^{1-\beta_{uy}} \text{s}^{-1}$
$\beta_{uy}$	Coefficient in the relational expression of fall velocity and diameter of a category $y$	
$\lambda_y$	Inclination in a reversed exponential function expressing particle size distribution of a category $y$	$\text{m}^{-1}$
$\mu$	Coefficient of viscosity of air	$\text{kg m}^{-1} \text{s}^{-1}$
$\bar{\rho}$	Air density in the basic state	$\text{kg m}^{-3}$
$\rho_0$	Density at the surface in the basic state	$\text{kg m}^{-3}$
$\rho_w$	Density of water	$1 \times 10^3 \text{ kg m}^{-3}$

(3) Formation Process of Graupel by Collection of a Supercooled Raindrop and Ice Crystals :  $CL_{ri}$

When it is assumed that a raindrop freezes instantaneously, variations of mixing ratio and number concentration of graupel with its formation by the collision of a supercooled raindrop and ice crystals are given as follows<sup>17</sup>.

$$CL_{ri} = \frac{\pi^2}{24} E_{ir} N_i n_{r0} \alpha_{ur} \Gamma(6 + \beta_{ur}) \lambda_r^{-(6+\beta_{ur})} \left( \frac{\rho_0}{\bar{\rho}} \right)^{\frac{1}{2}} \quad (4.112)$$

<sup>17</sup>村上 (1999)

$$CL_{ri}^N = \frac{\pi}{4\bar{\rho}} E_{ir} N_i n_{r0} \alpha_{ur} \Gamma(3 + \beta_{ur}) \lambda_r^{-(3+\beta_{ur})} \left(\frac{\rho_0}{\bar{\rho}}\right)^{\frac{1}{2}} \quad (4.113)$$

Meanings of the signs used here are as follows.

$E_{ir}$	Collection efficiency of a particle	1.0	
$n_{r0}$	Density of rain water in the $y$ section	$8.0 \times 10^6$	$\text{m}^{-4}$
$\alpha_{ur}$	Coefficient in the relational expression of fall velocity and diameter of rain water	842	$\text{m}^{1-\beta_{ur}} \text{s}^{-1}$
$\beta_{ur}$	Coefficient in the relational expression of fall velocity and diameter of rain water	0.8	
$\lambda_r$	Inclination in a reversed exponential function expressing particle size distribution of rain water		$\text{m}^{-1}$
$\bar{\rho}$	Air density in the basic state		$\text{kg m}^{-3}$
$\rho_0$	Density at the surface in the basic state		$\text{kg m}^{-3}$

#### (4) Summary of Collection Efficiencies for Coalescences : $E_{xy}$ <sup>18</sup>

Collection efficiencies  $E_{xy}$  for coalescences between particles shown in (1)-(3) are summarized in the following table.

$E_{cr}$	Collection efficiency for the coalescence of rain water with cloud water	$Stk^2 / (Stk + 0.5)^2$
$E_{cs}$	Collection efficiency for the coalescence of snow with cloud water	$Stk^2 / (Stk + 0.5)^2$
$E_{cg}$	Collection efficiency for the coalescence of graupel with cloud water	$Stk^2 / (Stk + 0.5)^2$
$E_{rs}$	Collection efficiency for the coalescence of snow with rain water	1.0
$E_{rg}$	Collection efficiency for the coalescence of graupel with rain water	1.0
$E_{ir}$	Collection efficiency for the coalescence of rain water with cloud ice	1.0
$E_{is}$	Collection efficiency for the coalescence of snow with cloud ice	1.0
$E_{ig}$	Collection efficiency for the coalescence of graupel with cloud ice	0.1
$E_{sr}$	Collection efficiency for the coalescence of rain water with snow	1.0
$E_{sg}$	Collection efficiency for the coalescence of graupel with snow	0.001

#### (5) Category Allocation Ratios of Snow and Raindrop after the Coalescence : $\alpha_{rs}$

In the coalescence of snow and raindrop, it is difficult that we decide an adequate category a particle formed by the coalescence in the layer less than 0 °C . Here, the allocation ratios of the snow and graupel, which appear in the equations (4.31) and (4.32), is given by using the average masses of rain water  $\bar{m}_r$  and snow  $\bar{m}_s$  as follows.

$$\alpha_{rs} = \frac{\bar{m}_s^2}{\bar{m}_s^2 + \bar{m}_r^2} \quad (4.114)$$

<sup>18</sup>Ikawa and Saito (1991), Ikawa et al. (1991)

The average masses of rain water  $\bar{m}_r$  and snow  $\bar{m}_s$  are given as follows.

$$\bar{m}_r = \rho_r \left( \frac{4}{\lambda_r} \right)^3 \quad (4.115)$$

$$\bar{m}_s = \rho_s \left( \frac{4}{\lambda_s} \right)^3 \quad (4.116)$$

Using these equations, we can find that the generation ratio of graupel by the coalescence of a raindrop and snow is  $(1 - \alpha_{rs})$ . Note that the ratios in (4.114) are tentative values.

### Production of Graupel : $PG$

Production of graupel is an important and complicated problem, since graupel is produced by the collision between other particles. In this section, production process of graupel is explained in detail especially, although it should be explained in the section of "Collision between particles".

Production of graupel has two processes: dry growth and wet growth. By the former process, all supercooled cloud droplets freeze in an instant on contact with graupel, and surface of graupel remains dry. By the latter process, not all supercooled cloud droplets freeze by the latent heat which supercooled cloud droplets emit when they collided with graupel. Surface of graupel gets wet.

In dry growth process, although all supercooled cloud droplets contribute to growth, cloud ice and snow hardly contribute to growth because the collection efficiency of cloud ice or snow colliding with graupel is small. On the contrary, in wet growth process, a freezing amount of supercooled water is decided by the budget of sensible and latent heat. Cloud ice or snow are efficiently captured by graupel.

Dry growth process is defined as following<sup>19</sup>.

$$PG_{dry} = CL_{cg} + CL_{rg} + CL_{ig} + CL_{sg} \quad (4.117)$$

On the contrary, wet growth process is defined as following.

$$PG_{wet} = \frac{2\pi [\kappa T_s + \mathcal{L}_v \mathcal{D}_v \bar{\rho} (q_{vs}(T_0) - q_v)]}{\bar{\rho} (\mathcal{L}_f - C_w T_s)} VENT_g + (CL'_{ig} + CL'_{sg}) \left( 1 + \frac{C_i T_s}{\mathcal{L}_f - C_w T_s} \right) \quad (4.118)$$

Where  $VENT_g$  is ventilation efficiency (4.104). Whether graupel grows by dry growth process or wet growth process is decided by following relation of  $PG_{dry}$  and  $PG_{wet}$ .

$$PG_g = PG_{dry}, \quad PG_{dry} \leq PG_{wet} \quad (4.119)$$

$$PG_g = PG_{wet}, \quad PG_{dry} > PG_{wet} \quad (4.120)$$

Meanings of signs used in this section are shown as follows.

$CL_{cg}$	Growth rate of cloud water by coalescence with cloud water	$s^{-1}$
$CL_{ig}$	Growth rate of cloud ice by coalescence with cloud water	$s^{-1}$
$CL'_{ig}$	Growth rate of cloud ice by coalescence with cloud water in wet growth process	$s^{-1}$
$CL_{rg}$	Growth rate of rain by collecting cloud water	$s^{-1}$
$CL_{sg}$	Growth rate of snow by collecting cloud water	$s^{-1}$

<sup>19</sup>村上 (1999)

$CL'_{sg}$	Growth rate of snow by collecting cloud water in wet growth process		$s^{-1}$
$C_i$	Specific heat at constant pressure of ice	$2.0 \times 10^3$	$J K^{-1} kg^{-1}$
$C_w$	Specific heat at constant pressure of water	$4.17 \times 10^3$	$J K^{-1} kg^{-1}$
$\mathcal{D}_v$	Diffusion coefficient of water vapor		$m^2 s^{-1}$
$\mathcal{L}_f$	Latent heat of fusion of water		$J kg^{-1}$
$\mathcal{L}_v$	Latent heat of evaporation of water		$J kg^{-1}$
$q_{vs}(T_0)$	Saturation mixing ratio at melting point of water point of water		$kg kg^{-1}$
$T$	Temperature		K
$T_0$	Melting point of ice	273.16	K
$T_s$	Temperature in supercooling ( $T_0 - T$ )		K
$\kappa$	Efficiency of conduction of heat of air	$2.4 \times 10^{-2}$	$J m^{-1} s^{-1} K^{-1}$
$\bar{\rho}$	Air Density in the basic state		$kg m^{-3}$

### Aggregation : $AG$

Aggregation, as well as production of graupel, should be categorized in "Collision between particles," however, we explain it especially in this section.

We consider the following two phenomenons for aggregation. First, the number of cloud ice per unit volume decreases by the collision between cloud ices (ice crystals). Second, the number of snow per unit volume decreases by the collision between snows (snowflakes).

By aggregation process, only the number concentration changes but mixing ratio does not change.

#### (1) Aggregation between Cloud Ice Particles : $AG_i^N$ <sup>20</sup>

Decrease of the number concentration of cloud ices (ice crystals) by aggregation process is defined as following.

$$AG_i^N = \left[ \frac{d}{dt} \left( \frac{N_i}{\bar{\rho}} \right) \right]_{aggr} = -\frac{c_1}{2\bar{\rho}} N_i \quad (4.121)$$

where  $c_1$  is defined as following.

$$c_1 = \frac{\bar{\rho} q_i \alpha_{ui} E_{ii} X}{\rho_i} \left( \frac{\rho_0}{\bar{\rho}} \right)^{\frac{1}{3}} \quad (4.122)$$

Meanings of signs used in this section are shown as follows.

$E_{ii}$	Collection efficiency between ice crystals	0.1	
$X$	Spectrum dispersion of fall velocity of ice crystal	0.25	
$\alpha_{ui}$	The coefficient which appears in the equation of fall velocity and diameter of ice crystal	700	$m^{1-\beta_{ui}} s^{-1}$
$\beta_{ui}$	The coefficient which appears in the equation of fall velocity and diameter of ice crystal	1.0	
$\bar{\rho}$	Air density in the basic state		$kg m^{-3}$
$\rho_0$	Air density at the surface in the basic state		$kg m^{-3}$
$\rho_i$	Density of cloud ice	$5.0 \times 10^2$	$kg m^{-3}$

<sup>20</sup>村上 (1999), Ikawa and Saito (1991)

**(2) Aggregation between Snow Particles :  $AG_s^N$** <sup>21</sup>

A decrease of the number concentration of snow (snowflakes) by aggregation process is defined as following. By this process, only the number concentration of snow ( $N_s$ ) changes but mixing ratio of snow ( $q_s$ ) does not change.

$$AG_s^N = \left[ \frac{d}{dt} \left( \frac{N_s}{\bar{\rho}} \right) \right]_{aggr} = -\frac{1}{\bar{\rho}} \frac{\alpha_{us} E_{ss} I(\beta_{us})}{4 \times 720} \pi^{\frac{1-\beta_{us}}{3}} \bar{\rho}^{\frac{2+\beta_{us}}{3}} \rho_s^{\frac{-2-\beta_{us}}{3}} q_s^{\frac{2+\beta_{us}}{3}} N_s^{\frac{4-\beta_{us}}{3}} \quad (4.123)$$

where  $I(\beta_{us})$  is defined as following.

$$I(\beta_{us}) = \int_0^\infty \int_0^\infty x^3 y^3 (x+y)^2 |x_{us}^\beta - y_{us}^\beta| \exp[-(x+y)] dx dy \quad (4.124)$$

This equation is solved with Gauss's hypergeometric function as following.<sup>22</sup>

$$I(\beta_{us}) = \Gamma(\beta_{us}) 2^{1-d} \sum_{i=1}^3 C_i \left[ \frac{F(1, d; 8-i; 0.5)}{7-i} - \frac{F(1, d; 4+\beta_{us}; 0.5)}{3+\beta_{us}+i} \right] \quad (4.129)$$

where

$$\begin{aligned} d &= 10 + \beta_{us} \\ C_1 &= 1 \\ C_2 &= 3 \\ C_3 &= 1. \end{aligned}$$

We adopted the following values as typical values of  $I(\beta_{us})$  (Ikawa and Saito, 1991; Mizuno, 1990).

$\beta_{us}$	0.4	0.5	0.6
$I(\beta_{us})$	1108	1610	2566

<sup>21</sup>村上 (1999), Ikawa and Saito (1991)

<sup>22</sup>The hypergeometric function is one of solutions of hypergeometric equation which is two order linear ordinary differential equation, and has regular singularity at  $x = 0, 1, \infty$ .

$$F(x, a; b; c) = 1 + \frac{a \cdot b}{c} \frac{x}{1!} + \frac{a(a+1)b(b+1)}{c(c+1)} \frac{x^2}{2!} + \dots \quad (4.125)$$

where  $c \neq 0, -1, -2, \dots$ . With Pochhimer's sign,

$$(a)_n = a(a+1)(a+2)\dots(a+n-1) = \frac{(a+n-1)!}{(a-1)!} \quad (4.126)$$

$$(a)_0 = 1 \quad (4.127)$$

$$F(x, a; b; c) = \sum_{n=0}^{\infty} \frac{(a)_n (b)_n}{(c)_n} \frac{x^n}{n!} \quad (4.128)$$

As special limit of this hypergeometric function, there is confluent hypergeometric function

Meanings of signs used in these equations are shown as follows.

$E_{ss}$	Collection efficiency between snows	0.1	
$\alpha_{us}$	The coefficient which appears in the equation of fall velocity and diameter of snow	17	$\text{m}^{1-\beta_{us}} \text{s}^{-1}$
$\beta_{us}$	The coefficient which appears in the equation of fall velocity and diameter of snow	0.5	
$\bar{\rho}$	Air density in the basic state		$\text{kg m}^{-3}$
$\rho_s$	Density of snow	$8.4 \times 10^1$	$\text{kg m}^{-3}$

### Conversion : $CN$

Particle conversion from a certain category 'x' into other category 'y' is expressed by  $CN_{xy}$ .

For example, Kessler (1969)'s expression of conversion from cloud water into rain water is one of the earliest formulation of category conversion.

We consider the following types of category conversions.

process	consumed particle	growing particle	sign	meaning
conversion from cloud water to rain water	$q_c$	$q_r$	$CN_{cr}$	by aggregation
conversion from cloud ice to snow	$q_i$	$q_s$	$CN_{is}$	by aggregation and sublimation
conversion from snow to graupel	$q_s$	$q_g$	$CN_{sg}$	by growth with capturing cloud
conversion from snow to hail	$q_s$	$q_h$	$CN_{sh}$	by growth with capturing cloud
conversion from graupel to hail	$q_g$	$q_h$	$CN_{gh}$	by growth with capturing cloud
conversion from snow to graupel	$q_s$	$q_g$	$CN_{sg}$	by freezing of rain water
conversion from snow to hail	$q_s$	$q_h$	$CN_{sh}$	by freezing of rain water
conversion from graupel to hail	$q_g$	$q_h$	$CN_{gh}$	by freezing of rain water

Hereafter, we consider about conversion processes of  $CN_{cr}, CN_{is}, CN_{sg}$ .

#### (1) Conversion from Cloud to Rain : $CN_{cr}$ <sup>23</sup>

The conversion from cloud water to rain water is based on coalescence growth of cloud droplets. Kessler (1969) parameterized this process first. This process is investigated in detail by Berry (1968), and Berry and Reinhardt (1974). Recently, the method for the conversion based on this process is used in many cases.

#### (a) Conversion Based on Berry(1968), Berry and Reinhardt(1974)<sup>24</sup>

The method for the conversion based on Berry (1968), Berry and Reinhardt (1974) is shown as following.

$$CN_{cr} = \begin{cases} \frac{0.104gE_{cc}}{\mu(N_c\rho_w)^{\frac{1}{3}}} (\bar{\rho}^4 q_c^7)^{\frac{1}{3}}, & q_c \geq q_{cm} \\ 0, & q_c < q_{cm} \end{cases} \quad (4.130)$$

where the collection efficiency between cloud water is  $E_{cc} = 0.55$ . The critical value of mixing ratio of cloud water to convert from cloud water to rain water ( $q_{cm}$ ) is shown as following.

$$q_{cm} = \frac{\rho_w}{6\bar{\rho}} \pi D_{cm}^3 N_c \quad (4.131)$$

where the critical mean value of diameter of cloud is  $D_{cm} = 20 [\mu\text{m}]$ . We adopt a typical value ( $N_c = 10^8 [\text{m}^{-3}]$ ) of the number concentration of cloud ( $N_c$ ). However, it is better for this to be calculated by time integration as a forecast variable of a time development equation essentially.

<sup>23</sup>村上 (1999), Lin et al. (1983), Ferrier (1994), Ikawa and Saito (1991)

<sup>24</sup>村上 (1991)



**(b) Kessler (1969) 's Definition of Coalescence Growth** <sup>25</sup>

Kessler's parameterization is the most classic parameterization, and is also used in many models now.

$$CN_{cr} = a (q_c - q_{cm}) H (q_c - q_{cm}) \quad (4.132)$$

where  $H$  is set function. Usually  $a$  is selected as  $10^{-3} [s^{-1}]$ ,  $q_{cm}$  is selected as  $10^{-3} [\text{kg kg}^{-1}]$ . In Cotton and Anthes (1989),  $q_{cm}$  and  $a$  are given by the function of  $q_c$  as following.

$$q_{cm} = \frac{4\pi\rho_w N_c D_{cm}^3}{3\bar{\rho}} = 4 \times 10^{-12} N_c, \quad D_{cm} = 10^{-5} [m] \quad (4.133)$$

$$a = \pi E_{cc} U_{dc} N_c D_c^2 = 1.3 \times q_c^{\frac{4}{3}} N_c^{-\frac{1}{3}} \left( \frac{\rho_0}{\bar{\rho}} \right) \quad (4.134)$$

**(c) Lin et al.(1983)'s way**

Lin et al. (1983) improved Berry (1968)'s method as following.

$$CN_{cr} = \bar{\rho} (q_c - q_{cm})^2 \left[ 1.2 \times 10^{-4} + 1.569 \times 10^{-12} \frac{N_c}{\sigma^2 (q_c - q_{cm})} \right] \quad (4.135)$$

where  $\sigma^2$  indicates dispersion of the number concentration of cloud (= 0.15).  $q_{cm}$  is selected as  $2 \times 10^{-3} [\text{kg kg}^{-1}]$ .

Meanings of signs used in (a)~(c) are shown as following.

$g$	Acceleration of gravity	9.8	$\text{m s}^{-2}$
$N_c$	Number concentration of cloud particle	$1 \times 10^8$	$\text{m}^{-3}$
$\mu$	Viscosity Coefficient of air		$\text{kg m}^{-1} \text{s}^{-1}$
$\bar{\rho}$	Air density in the basic state		$\text{kg m}^{-3}$
$\rho_0$	Air density at the surface in the basic state		$\text{kg m}^{-3}$
$\rho_w$	Density of water	$1 \times 10^3$	$\text{kg m}^{-3}$

**(2) Conversion from Cloud Ice to Snow :  $CN_{is}$** 

The conversion from cloud ice to snow is formulized with an assumption of which the conversion occurs by two processes of sublimation growth of cloud ice and condensation<sup>26</sup>.

By sublimation condensation growth, it takes  $\Delta t_{is1}$  for ice crystal with  $\bar{R}_i$  radius to grow into snow with  $R_{s0}$  radius.  $\Delta t_{is1}$  is

$$\Delta t_{is1} = \frac{R_{s0}^2 - \bar{R}_i^2}{2a_1} \rho_i \quad (4.136)$$

where  $a_1$ , as well as (4.100) and (4.97), is expressed as following.

$$a_1 = (S_i - 1) \left( \frac{\mathcal{L}_s^2}{\kappa R_v T^2} + \frac{1}{\bar{\rho} q_{vsi} \mathcal{D}_v} \right)^{-1} \quad (4.137)$$

<sup>25</sup>Ikawa and Saito (1991)

<sup>26</sup>村上 (1999), Murakami (1990), Ikawa and Saito (1991)

Variation of mixing ratio from ice cloud to snow per unit time by sublimation condensation growth (conversion rate)  $CN_{is}^{dep}$  is showed as following.

$$CN_{is}^{dep} = \frac{q_i}{\Delta t_{is1}} \quad (4.138)$$

By aggregation growth, it takes  $\Delta t_{is1}$  for cloud ice with  $\bar{R}_i$  radius to grow into snow with  $R_{s0}$  radius. We suppose that the  $\Delta t_{is1}$  is the same time as it takes for the number concentration of cloud ice to decrease from  $N_i$  to  $N_i (R_i / R_{s0})^3$  on condition  $\rho_i = const..$  Therefore, the  $\Delta t_{is1}$  is expressed as following.

$$\Delta t_{is2} = \frac{2}{c_1} \log \left( \frac{R_{s0}}{R_i} \right)^3 \quad (4.139)$$

where  $c_1$  is given by the equation (4.122). Variation of mixing ratio from ice cloud to snow per unit time by aggregation growth (conversion rate) is showed as following.

$$CN_{is}^{agg} = \frac{q_i}{\Delta t_{is2}} \quad (4.140)$$

Finally, the conversion rate from cloud ice to snow ( $CN_{is}$ ) is given as following.

$$CN_{is} = CN_{is}^{dep} + CN_{is}^{agg} \quad (4.141)$$

Meaning of signs used here are shown as following.

$\mathcal{D}_v$	Diffusion coefficient of water vapor		$\text{m}^2 \text{s}^{-1}$
$\mathcal{L}_s$	Latent heat of sublimation of water		$\text{J kg}^{-1}$
$q_{vsi}$	Saturation mixing ratio over ice		$\text{kg kg}^{-1}$
$R_v$	Gas constant of water vapor	461.0	$\text{J K}^{-1} \text{kg}^{-1}$
$S_i - 1$	Supersaturation of air to ice		
$T$	Temperature		K
$\kappa$	Conduction efficiency of heat of air	$2.4 \times 10^{-2}$	$\text{J m}^{-1} \text{s}^{-1} \text{K}^{-1}$
$\bar{\rho}$	Air density in the basic state		$\text{kg m}^{-3}$
$\rho_i$	Density of cloud ice	$5.0 \times 10^2$	$\text{kg m}^{-3}$

### (3) Conversion from Snow to Graupel: $CN_{sg}$

We suppose that the conversion from snow to graupel takes place when  $CL_{cs}$  (growth rate by capturing cloud droplets), which is given in (4.109), exceeds  $VD_{vs}$  (growth rate by sublimation condensation), which is given in (4.100)<sup>27</sup>. When the conversion takes place, the rate of change of mixing ratio per unit time by converting from snow to graupel is shown as following.

When  $CL_{cs} > VD_{vs}$ ,

$$CN_{sg} = CL_{cs} \frac{\rho_s}{\rho_g - \rho_s}. \quad (4.142)$$

The variation of the number concentration is shown here,

$$CN_{sg}^N = \frac{3\alpha_{us} E_{cs} q_c (\rho_0 \bar{\rho})^{\frac{1}{2}} \Gamma(\beta_{us})}{2\bar{\rho} (\rho_g - \rho_s)} \lambda_s^{1-\beta_{us}} N_s. \quad (4.143)$$

Meanings of signs used here are shown as following.

$E_{cs}$	Collection efficiency of snow in the coalescence with cloud ice	$Stk^2 / (Stk + 0.5)^2$	
$\alpha_{us}$	The coefficient which appears in the equation of fall velocity and a diameter of snow.	17	$\text{m}^{1-\beta_{us}} \text{s}^{-1}$
$\beta_{us}$	The coefficient which appears in the equation of fall velocity and a diameter of snow	0.5	
$\lambda_s$	Inclination in the reverse exponential function expressing particle size distribution of snow		$\text{m}^{-1}$
$\bar{\rho}$	Air density in the basic state		$\text{kg m}^{-3}$
$\rho_0$	Air density at the surface in the basic state		$\text{kg m}^{-3}$
$\rho_g$	Density of graupel	$3.0 \times 10^2$	$\text{kg m}^{-3}$
$\rho_s$	Density of snow	$8.4 \times 10^1$	$\text{kg m}^{-3}$

## Melting of Solid Particle: $ML$

### (1) Melting of Cloud Ice: $ML_{ic}$

<sup>27</sup>村上 (1999), Murakami (1990)

We suppose that cloud ice melts and is converted into cloud water in an instant when  $T > T_0$ , because the radius of cloud ice (ice crystal) is very small.

When  $T_c > T_0$ ,

$$ML_{ic} = \frac{q_i}{2\Delta t}. \quad (4.144)$$

**(2) Melting of Snow and Graupel:  $ML_{sr}, ML_{gr}$**

With regard to snow and graupel,  $ML_{xr}$  ( $x = s, g$ ), which indicates the conversion rate from solid particles to rain, is controlled by heat budget<sup>28</sup>, that is,

When  $T > T_0$ ,

$$ML_{xr} = \frac{2\pi}{\bar{\rho}\mathcal{L}_f} [\kappa T_c + \mathcal{L}_v \mathcal{D}_v \bar{\rho} (q_v - q_{vs}(T_0))] VENT_x + \frac{C_w T_c}{\mathcal{L}_f} (CL_{cx} + CL_{rx}). \quad (4.145)$$

If  $ML_{xr} > 0$ , snow or graupel melts and is converted to rain, however if  $ML_{xr} < 0$ , even if  $T > T_0$ , melting does not take place, that is,  $ML_{xr} = 0$ .  $VENT_g$  is ventiration efficiency(4.104).

Meanings of signs used in (1),(2) are shown as following.

$CL_{cx}$	Growth rate of a particle in x category by collision with cloud water		$s^{-1}$
$CL_{rx}$	Growth rate of a particle in x category by collision with rain water		$s^{-1}$
$C_w$	Specific heat at constant pressure of water	$4.17 \times 10^3$	$J K^{-1} kg^{-1}$
$\mathcal{D}_v$	Diffusion coefficient of water vapor		$m^2 s^{-1}$
$\mathcal{L}_f$	Latent heat of fusion of water		$J kg^{-1}$
$\mathcal{L}_v$	Latent heat of evaporation of water		$J kg^{-1}$
$q_{vs}(T_0)$	Saturated mixing ratio at melting point of water		$kg kg^{-1}$
$T$	Temperature		K
$T_0$	Melting point of ice	273.16	K
$T_c$	Temperature in Celsius scale		$^{\circ}C$
$\kappa$	Efficiency of conduction of heat of air	$2.4 \times 10^{-2}$	$J m^{-1} s^{-1} K^{-1}$
$\bar{\rho}$	Air density in the basic state		$kg m^{-3}$

<sup>28</sup>Ikawa and Saito (1991)

**Freezing of Rain:  $FR$** 

We suppose that freezing of rain takes place in an instant. All frozen rain is categorized into graupel because we do not consider hail. If there is a category of hail, frozen rain is categorized into the category of hail.

The rate of variation of mixing ratio by freezing rain ( $FR_{rg}$ ) is based on Bigg (1953)'s empirical equation<sup>29</sup>, and is given as following.

$$FR_{rg} = 20\pi^2 B' n_{r0} \frac{\rho_w}{\bar{\rho}} [\exp(A'T_s) - 1] \lambda_r^{-7} \quad (4.146)$$

The rate of variation of the number concentration is given as following.

$$FR_{rg}^N = \frac{\pi}{6\bar{\rho}} B' n_{r0} [\exp(A'T_s) - 1] \lambda_r^{-4} \quad (4.147)$$

Meanings of signs used here are shown as following.

$A'$	A coefficient in Bigg's empirical equation	0.66	$K^{-1}$
$B'$	A coefficient in Bigg's empirical equation	100.0	$m^{-3} s^{-1}$
$n_{r0}$	Density of rain in $y$ section	$8.0 \times 10^6$	$m^{-4}$
$T$	Temperature		K
$T_0$	Melting point of ice	273.16	K
$T_s$	Supercooled temperature ( $T_0 - T$ )		K
$\lambda_r$	Inclination in the reverse exponential function expressing particle size distribution of rain		$m^{-1}$
$\bar{\rho}$	Air density in the basic state		$kg m^{-3}$
$\rho_w$	Density of water	$1 \times 10^3$	$kg m^{-3}$

**Shedding of Water from Snow and Graupel:  $SH$** 

In our model, we do not consider that snow and graupel partly contain liquid water in them. Therefore, melted water sheds from snow or ice, and becomes rain water. The process is expressed as following.

When  $T > T_0$ ,

$$SH_{sr} = CL_{cs} + CL_{rs} \quad (4.148)$$

$$SH_{gr} = CL_{cg} + CL_{rg} \quad (4.149)$$

If graupel grows by the wet growth process (even when  $T \leq T_0$ ),

$$SH_{gr} = CL_{cg} + CL_{rg} + CL'_{ig} + CL'_{sg} - PG_{wet}. \quad (4.150)$$

<sup>29</sup>Lin et al. (1983), 村上 (1999)

Meanings of signs used here are shown as following.

$CL_{cg}$	Growth rate of graupel by collision with cloud water	$s^{-1}$
$CL_{cs}$	Growth rate of snow by collision with cloud water	$s^{-1}$
$CL_{rg}$	Growth rate of graupel by collision with rain water	$s^{-1}$
$CL_{rs}$	Growth rate of snow by collision with rain water	$s^{-1}$
$CL'_{ig}$	Growth rate of graupel by collision with cloud water in wet growth process	$s^{-1}$
$CL'_{sg}$	Growth rate of graupel by collision with snow in wet growth process	$s^{-1}$
$PG_{wet}$	Growth rate of graupel in wet growth process	$s^{-1}$
$T$	Temperature	K
$T_0$	Melting point of ice	273.16 K

Ferrier (1994) formulizes the wet growth process of snow, graupel and hail (frozen ice) with a time development equation, and takes the wet snow and wet graupel and wet hail into consideration.

Such formulization will be required in the future.

### Breakup of Waterdrop

It is known that if it becomes large to some extent (it is 8mm at the diameter of a sphere), rain water becomes unstable and breaks up.

From this reason, the waterdrop more than the size of 8 mm does not fall down in an actual nature. Although the breakup of waterdrop does not change mixing ratio of rain water, it changes number concentration of rain water. However, since the number concentration of rain water is not predicted in this model, the process is not considered.

### 4.2.5 Moist Saturation Adjustment

The exchange between water vapor and cloud water is expressed by the moist saturation adjustment<sup>30</sup>.

We add the mark \* to the variable which is calculated by forecast equation. The supersaturated mixing ratio which is defined as

$$\Delta q_c = q_v^* - q_{vsw}^*. \quad (4.151)$$

If  $\Delta q_c > 0$ , the moist saturation adjustment is applied. The the moist saturation adjustment calculates  $\theta$ ,  $q_v$ ,  $q_c$  with following way.

$$\theta^{t+\Delta t} = \theta^* + \gamma (q_v^* - q_{vsw}^*) \left/ \left[ 1 + \gamma q_{vsw}^* \frac{4093\Pi}{(\Pi\theta^* - 36)^2} \right] \right. \quad (4.152)$$

$$q_v^{t+\Delta t} = q_v^* + (\theta^* - \theta^{t+\Delta t}) / \gamma \quad (4.153)$$

$$q_c^{t+\Delta t} = q_v^* + q_c^* - q_v^{t+\Delta t} \quad (4.154)$$

First, provisional  $\theta$ ,  $q_v$ ,  $q_c$  are calculated with these equations. If  $q_c^{t+\Delta t} > 0$ , the calculation with these equations (4.152)~(4.154) is repeatedly done until when the value of  $\theta$ ,  $q_v$ ,  $q_c$  converges. Therefore, final  $\theta$ ,  $q_v$ ,  $q_c$  is decided by the moist saturation adjustment. Here, we defined  $\gamma$  as  $\gamma \equiv \mathcal{L}_v / (C_p \Pi)$ .

If  $\Delta q_c \leq 0$ , we apply following equations.

<sup>30</sup>Soong and Ogura (1973)

$$\theta^{t+\Delta t} = \theta^* - \gamma q_c^* \quad (4.155)$$

$$q_v^{t+\Delta t} = q_v^* + q_c^* \quad (4.156)$$

$$q_c^{t+\Delta t} = 0 \quad (4.157)$$

Meanings of signs used here are shown as following.

$C_p$	Specific heat at constant pressure of dry air	1004	J K kg <sup>-1</sup>
$\mathcal{L}_v$	Latent heat of evaporation of water		J kg <sup>-1</sup>
$q_{vsw}$	Saturation mixing ratio over water		kg kg <sup>-1</sup>
$\Pi$	Exner function		

#### 4.2.6 Formulization of Variational Terms on Mixing Ratio and Number Concentration with Precipitation

The rate of variation of mixing ratio by fall of particles of cloud and precipitation are shown as following.

$$\text{Fall.} q_x = \frac{1}{\bar{\rho}} \frac{\partial \bar{\rho} \bar{U}_{xq} q_x}{\partial z} \quad (4.158)$$

The mass-weighted terminal fall velocity of particle in category x ( $\bar{U}_{xq}$ ) is given by the equation (4.68) in the section 4.2.3.

The rate of variation of the number concentration is shown as following.

$$\text{Fall.} \frac{N_x}{\bar{\rho}} = \frac{1}{\bar{\rho}} \frac{\partial N_x \bar{U}_{xN}}{\partial z} \quad (4.159)$$

$\bar{U}_{xN}$  indicates mean fall velocity of particle in category x. Similarly, it is given by the equation (4.67) in the section 4.2.3.

When the grid interval in the vertical direction is small,  $\Delta t$  must be enough small so that CFL condition is filled, and so that a falling precipitation particle should not pass next grid for 1 time step.

Since the differentiation in the vertical direction is the differentiation in real space, the differentiation in  $z^*$  coordinate ( $\zeta$  coordinate) must be multiplied by a metric, as shown in equation (2.58).

### 4.2.7 The other constants

Here, we show variables and constants, which are used in equations but are not explained.

**Water Saturation Mixing Ratio and Ice Saturation Mixing Ratio :**  $q_{vsw}, q_{vsi}$ <sup>31</sup>

$$q_{vsw} = \epsilon \frac{610.78}{p} \exp\left(17.269 \frac{T - T_0}{T - 35.86}\right) \quad (4.160)$$

$$q_{vsi} = \epsilon \frac{610.78}{p} \exp\left(21.875 \frac{T - T_0}{T - 7.86}\right) \quad (4.161)$$

**Latent Heat with Evaporation, Sublimation and Melting of Water :**  $\mathcal{L}_v, \mathcal{L}_s, \mathcal{L}_f$  [J kg<sup>-1</sup>]

$$\mathcal{L}_v = 2.50078 \times 10^6 \left(\frac{T_0}{T}\right)^{(0.167+3.67 \times 10^{-4}T)} \quad (4.162)$$

$$\mathcal{L}_s = 2.834 \times 10^6 + 100(T - T_0) \quad (4.163)$$

$$\mathcal{L}_f = 3.34 \times 10^5 + 2500(T - T_0) \quad (4.164)$$

**Coefficient of Kinematic Viscosity of Air :**  $\nu$  [m<sup>2</sup> s<sup>-1</sup>]

$$\nu = 1.328 \times 10^{-5} \frac{p_0}{p} \left(\frac{T}{T_0}\right)^{1.754} \quad (4.165)$$

**Coefficient of Viscosity of Air:**  $\mu$  [kg m<sup>-1</sup> s<sup>-1</sup>]

$$\mu = \rho \nu \quad (4.166)$$

**Coefficient of Water Vapor Diffusion:**  $\mathcal{D}_v$  [m<sup>2</sup> s<sup>-1</sup>]

$$\mathcal{D}_v = 2.23 \times 10^{-5} \frac{p_0}{p} \left(\frac{T}{T_0}\right)^{1.81} \quad (4.167)$$

Meanings of signs used here are shown as following.

$p$	Pressure		Pa
$p_0$	Standard pressure	101325	Pa
$T$	Temperature		K
$T_0$	Melting point of ice	273.16	K
$\epsilon$	Ratio of molecular weight of water vapor to molecular weight of dry air	0.622	
$\rho$	Air density		kg m <sup>-3</sup>

<sup>31</sup>Orville and Kopp (1977), Murray (1966)



## Chapter 5

# Atmospheric Boundary Layer, Ground Surface Process and Ground Temperature

The integration time of a cloud model is short ranging from few to ten hours. Therefore, in the model, most of phenomena provided by the atmospheric processes. When energy flux from the ground surface (land or sea) is large in a cloud / precipitation system as in the case of a thunderstorm in summer or a snow cloud over Japan Sea in winter, however, ground surface process is substantially important for the development of the system.

Following processes are considered as the ground surface processes and they are expressed by simple bulk methods.

- vertical one-dimensional diffusion in the atmospheric boundary layer
- fluxes of sensible heat and latent heat from the ground to atmosphere
- momentum flux (friction of the ground surface)
- soil temperature variation

In CReSS, these processes are expressed with vertical one-dimensional equations. We adopt the ground surface flux with a bulk method and soil temperature forecasting which are used in JSM (Japan Spectral Model; Segami et al., 1989). Complicated soil and vegetation models are incorporated in some cloud models (e.g. RAMS, ARPS). However, they are important only when integration for a long time (month order) in a wide area (e.g. the region covering all the Eurasian Continent) is conducted. Therefore, the complicated soil and vegetation models will be added to *CReSS* in the future.

## 5.1 Basic Theories of Atmospheric Boundary Layer

Here, we summarize briefly basic theories for the atmospheric boundary layer processes<sup>1</sup>.

### 5.1.1 Structure of Atmospheric Boundary Layer

Atmosphere is bordered on the lower boundary by the ground surface. Generally, effects of the ground surface which includes various conditions like land and sea become stronger as the altitude becomes lower. The effects exist from the surface to about 1-2 km high though the depth is affected with conditions of the ground surface and atmosphere. Such atmospheric layer which is thermodynamically and dynamically affected with the ground surface is called atmospheric boundary layer or planetary boundary layer; PBL.

Atmospheric boundary layer is divided into two layers; surface boundary layer which ranges from the surface to 20-50 m high and is affected by the ground surface, and Ekman layer or outer boundary layer above the surface boundary layer. Atmosphere above the atmospheric boundary layer is called free atmosphere.

The surface boundary layer has a property that vertical fluxes of sensible heat, latent heat, momentum and so on are uniform vertically and equal to their values on the ground. In other words, the atmospheric layer where these vertical fluxes are equal to the values on the ground can be called the surface boundary layer. The surface boundary layer is sometimes called constant flux layer. Because momentum flux is vertically constant, stress of turbulence is vertically constant in the surface boundary layer, that is, wind direction is also constant vertically.

### 5.1.2 Parameterization of Turbulence Transportation

Motions of various scale exist in the atmosphere. Among them, motion which can be expressed on the grid points of a numerical model is called grid-scale motion or mean motion. Motion which has a scale smaller than a grid size is called subgrid-scale motion or eddy motion.

To classify these motions, physical quantity  $A$  (e.g. wind speed, temperature, mixing ratio of water vapor) is divided into two quantity; averaged quantity which can be expressed on the grid points and deviation from that, as follows.

$$A = \bar{A} + A'' \quad (5.1)$$

There are some ways to average physical quantities, we will not mention about them here. Average of deviation and that of product of two physical quantities are expressed as follows;

$$\overline{A''} = 0 \quad (5.2)$$

$$\overline{AB} = \bar{A}\bar{B} + \overline{A''B''} \quad (5.3)$$

The average of the products of some quantities is not always equal to the product of averages of each quantity, and the second term of the equation above appears. This equation is applied to the  $x$  component of equations of motion. Considering an uncompressible fluid ( $\rho = \text{const.}$ ), each term of the equation can be expressed with the average and deviation from that as follows;

$$\begin{aligned} \frac{\partial \bar{u}}{\partial t} + \bar{u} \frac{\partial \bar{u}}{\partial x} + \bar{v} \frac{\partial \bar{u}}{\partial y} + \bar{w} \frac{\partial \bar{u}}{\partial z} - f\bar{v} = -\frac{1}{\rho} \frac{\partial \bar{p}}{\partial x} \\ - \frac{1}{\rho} \left( \frac{\partial}{\partial x} \overline{\rho u'' u''} + \frac{\partial}{\partial y} \overline{\rho u'' v''} + \frac{\partial}{\partial z} \overline{\rho u'' w''} \right) + \nu \nabla^2 \bar{u}. \end{aligned} \quad (5.4)$$

<sup>1</sup>Introduction to dynamic meteorology (Ogura, 1978) and Atmospheric Science Course 1 - Atmosphere near the ground surface (Takeuchi and Kondo, ????) are mainly referred here.

The terms of  $-\overline{\rho u'' u''}$ ,  $-\overline{\rho u'' v''}$  and  $-\overline{\rho u'' w''}$  express the stress of turbulence and are called eddy stress or Reynolds stress. These stresses can be regarded as a momentum transportation, and the momentum transportation by eddy is the stress.

Similarly, equations of potential temperature and mixing ratio of water vapor are as follows.

$$\frac{\partial \bar{\theta}}{\partial t} + \bar{u} \frac{\partial \bar{\theta}}{\partial x} + \bar{v} \frac{\partial \bar{\theta}}{\partial y} + \bar{w} \frac{\partial \bar{\theta}}{\partial z} = -\frac{\partial}{\partial x} \overline{u'' \theta''} - \frac{\partial}{\partial y} \overline{v'' \theta''} - \frac{\partial}{\partial z} \overline{w'' \theta''} \quad (5.5)$$

The quantities with a prime cannot be expressed with the grid points. It is necessary to express them with quantities on the grid points (averaged quantities) for including effects of momentum transportation by eddy which is smaller than the grid size. A problem of turbulent parameterization is to decide the way to express deviation of quantity by eddy with prime on the grid points. Note that definitions of quantities with  $\bar{\quad}$  used in this chapter and in the other chapter (e.g. Chapter 2) are different.

### 5.1.3 Mixing Length Theory

As mentioned in the chapter of turbulence, atmospheric motions should have a motion which is smaller than the grid size. Therefore, the equation system is not closed due to correlation quantities of deviation as in (5.4) and (5.5). This is called closure problem and the terms of transportation by turbulence must be expressed as functions of quantities on grid points to close the equation system. One of the turbulent parameterization methods is mixing length theory by Prandtl (1925). This is explained by an analogy of kinetic theory of gases as follows.

We assume that the mean vertical speed of air mass at the altitude of  $z - l''$  is  $\bar{u}(z - l'')$ , and gives momentum to the environment when the air mass moves upward for  $l''$  reaching to the altitude of  $z$ . The variation of the speed is

$$u'' = \bar{u}(z - l'') - \bar{u}(z) \cong -l'' \frac{\partial \bar{u}}{\partial z}. \quad (5.6)$$

Raynolds tension is

$$\tau_{zx} = -\overline{\rho u'' w''} = \rho \overline{l'' w''} \frac{\partial \bar{u}}{\partial z} = \rho K_{mz} \frac{\partial \bar{u}}{\partial z} \quad (5.7)$$

where

$$K_{mz} = \overline{l'' w''} \quad (5.8)$$

is eddy viscosity coefficient.

Considering fluid with neutral stratification, the effect of buoyancy can be neglected, and it can be assumed  $|w''| \sim |u''|$ . We can assume the following relation.

$$w'' \sim u'' \cong -l'' \frac{\partial \bar{u}}{\partial z} \quad (5.9)$$

Then,

$$\tau_{zx} = -\overline{\rho u'' w''} = \rho \overline{l''^2} \frac{\partial \bar{u}}{\partial z} \left| \frac{\partial \bar{u}}{\partial z} \right|. \quad (5.10)$$

and,

$$K_{mz} = \overline{l'^2} \left| \frac{\partial \bar{u}}{\partial z} \right|. \quad (5.11)$$

Using these equations,

$$-\frac{1}{\rho} \frac{\partial}{\partial z} \overline{\rho u'' w''} = -\frac{1}{\rho} \frac{\partial}{\partial z} \left( \rho K_{mz} \frac{\partial \bar{u}}{\partial z} \right) \quad (5.12)$$

$$-\frac{\partial}{\partial z} \overline{w'' \theta''} = -\frac{\partial}{\partial z} \left( K_{hz} \frac{\partial \bar{\theta}}{\partial z} \right) \quad (5.13)$$

are gained.

As mentioned above, the transportation of physical quantities with turbulence could be explained by using mean field based on an analogy of kinetic theory of gases. That is called Prandtl's mixing length theory. It is difficult to apply generally this theory for turbulent transportation. For the surface boundary layer, we can apply the theory as a basic theory to calculate ground surface flux. The other theories are needed to solve a closure problem for the whole of the atmospheric boundary layer including Ekman layer.

#### 5.1.4 Surface Boundary Layer

The surface boundary layer is called constant flux layer where a value of vertical momentum flux at all altitudes is equal to that of the ground surface, and stress is constant. Therefore, wind direction is constant in the surface boundary layer. We set that the direction of an averaged wind is the x direction. Considering (5.10),

$$\frac{\partial \bar{u}}{\partial z} = \frac{1}{l} \sqrt{\frac{\tau_{zx}}{\rho}} \quad (5.14)$$

where  $l = \sqrt{\overline{l'^2}}$ . The size of turbulence which transports momentum within the surface boundary layer is restricted by the ground surface. The mixing length  $l$  is assumed as follows;

$$l = k z \quad (5.15)$$

where  $z$  is the distance from the surface and  $k$  is nondimensional quantity called Karman constant.  $k$  is decided as  $k = 0.4$  from wind tunnel experiments and observations<sup>2</sup>.

We define  $u_*$  as follows;

$$u_*^2 \equiv \frac{\tau_{zx}}{\rho} = -\overline{u'' w''}. \quad (5.16)$$

This is called friction velocity and has a dimension of velocity, corresponding to the scale of turbulent strength.

Observations showed that the amount of vertical transportation of momentum (or stress) is constant to altitudes within the surface boundary layer. Using these equations,

<sup>2</sup>Karman constant is decided as  $k = 0.4 \pm 0.01$  by simultaneous observations of Reynolds stress  $\tau = -\overline{u'' w''}$  and vertical profile of wind speed (Meteorology of water environment, p99, Kondo, ???).

$$u_* = \sqrt{\frac{\tau_{zx}}{\rho}} = kz \frac{\partial \bar{u}}{\partial z} = \text{const.} \quad (5.17)$$

is gained. Integrating this equation for  $z$ ,

$$\bar{u} = \frac{u_*}{k} \ln z + C. \quad (5.18)$$

Since the actual ground surface has various irregularity, it can not say that  $\bar{u} = 0$  at  $z = 0$ . Then, the constant of integration  $C$  is decided as the value which makes  $\bar{u} = 0$  at  $z = Z_{0m}$ .

$$\bar{u} = \frac{u_*}{k} \ln \frac{z}{Z_{0m}} \quad (5.19)$$

This is the logarithmic profile of wind speed.  $Z_{0m}$  means the extent of roughness on the ground and is called aerodynamical roughness parameter (or length) or ground surface roughness for wind profile. The logarithmic profile expressed with this equation corresponds well with the actual wind profile when the atmosphere has a neutral stratification.

Eddy viscosity coefficient in the surface boundary layer  $K_{mz}$  can be expressed by (5.11) and (5.17) as follows;

$$K_{mz} = (kz)^2 \frac{u_*}{kz} = kzu_*. \quad (5.20)$$

### 5.1.5 Vertical Flux

#### Momentum flux and Bulk coefficient

In the surface boundary layer, vertical flux is constant for altitude and it is enough to estimate flux at an altitude in the surface boundary layer for the estimation of flux on the ground. Momentum flux, i.e., wind stress on the ground is as follows;

$$\tau_0 = -\overline{\rho u'' w''} = \rho u_*^2 = \rho \bar{u}^2 \left[ \frac{k}{\ln \frac{z}{Z_{0m}}} \right]^2 = \rho C_m \bar{u}^2 \quad (5.21)$$

$\bar{u}$  can be gained by measuring average wind speed at an altitude (usually 10 m) in the surface boundary layer.  $C_m$  defined as

$$C_m = \left[ \frac{k}{\ln \frac{z}{Z_{0m}}} \right]^2 \quad (5.22)$$

is nondimensional quantity and called drag coefficient or bulk transfer coefficient.

**Potential Temperature Flux**

In the surface boundary layer, it can be considered that the amount of transportations (fluxes) of sensible heat and latent heat  $H_s, H_L$  are constant at all altitudes. If turbulence is decided by  $u_*$  and  $z$  in the surface boundary layer, equation which appears in (5.13);

$$K_{mz} \frac{\partial \bar{\theta}}{\partial z} = \overline{w''\theta''} \tag{5.23}$$

is expressed as follows by using eddy viscosity coefficient (??) as in the case of momentum;

$$\frac{\partial \bar{\theta}}{\partial z} = \frac{\overline{w''\theta''}}{kz u_*} \tag{5.24}$$

Sensible heat flux from the ground  $H_s$  [ $J\ m^{-2}\ s^{-1}$ ] is offered as follows;

$$H_s = -\rho C_p \overline{w''\theta''} \tag{5.25}$$

where potential temperature scale  $\theta_*$  which is defined by

$$\theta_* = \frac{\overline{w''\theta''}}{u_*} \tag{5.26}$$

and potential temperature flux is expressed as follows;

$$\frac{H_s}{\rho C_p} = -\overline{w''\theta''} = -kz \frac{\partial \bar{\theta}}{\partial z} u_* = -\theta_* u_* \tag{5.27}$$

$\theta_*$  defined here is constant in the surface boundary layer. Therefore,

$$\frac{\partial \bar{\theta}}{\partial z} = \frac{\theta_*}{kz} \tag{5.28}$$

is obtained. By integrating this vertically and defining  $Z_{0h}$  as the altitude of  $\bar{\theta} = \theta_G$  as in the case of momentum,

$$\bar{\theta} - \theta_G = \frac{\theta_*}{k} \ln \frac{z}{Z_{0h}} \tag{5.29}$$

is gained. Potential temperature flux is expressed with this as follows;

$$\frac{H_s}{\rho C_p} = -\overline{w''\theta''} = -\theta_* u_* = -\frac{k}{\ln \frac{z}{Z_{0h}}} \frac{k}{\ln \frac{z}{Z_{0m}}} (\bar{\theta} - \theta_G) \bar{u} \tag{5.30}$$

where

$$\frac{H_s}{\rho C_p} = -\frac{k}{\ln \frac{z}{Z_{0h}}} \frac{k}{\ln \frac{z}{Z_{0m}}} (\bar{\theta} - \theta_G) \bar{u} = -C_h (\bar{\theta} - \theta_G) \bar{u}. \quad (5.31)$$

Bulk coefficients of sensible heat and potential temperature can be defined as follows;

$$C_h = \frac{k}{\ln \frac{z}{Z_{0h}}} \frac{k}{\ln \frac{z}{Z_{0m}}}. \quad (5.32)$$

### Mixing Ratio of Water Vapor Flux

Latent heat flux  $H_L$  can be regarded to be constant in the surface boundary layer,

$$H_L = {}_lE = -\mathcal{L}_v \overline{w'' q_v''} \quad (5.33)$$

and

$$\frac{\partial \bar{q}_v}{\partial z} = \frac{\overline{w'' q_v''}}{k z u_*} \quad (5.34)$$

where  $L$  is latent heat of evaporation. Flux of mixing ratio of water vapor is expressed as follows.

$$E = -\overline{w'' q_v''} = -k z u_* \frac{\partial \bar{q}_v}{\partial z} \quad (5.35)$$

In the same way, we define the scale of mixing ratio of water vapor (friction mixing ratio)  $q_{v*}$  as

$$q_{v*} = \frac{\overline{w'' q_v''}}{u_*} \quad (5.36)$$

and flux of mixing ratio of water vapor is expressed as

$$E = -\overline{w'' q_v''} = -k z \frac{\partial \bar{q}_v}{\partial z} u_* = -q_{v*} u_* \quad (5.37)$$

where  $q_{v*}$  defined here is also constant in the surface boundary layer. By integrating this vertically and defining an altitude  $Z_{0h}$  of which  $\bar{q}_v = q_{vsfc}$  as same as the case of momentum and potential temperature,

$$\bar{q}_v - q_{vsfc} = \frac{q_{v*}}{k} \ln \frac{z}{Z_{0h}} \quad (5.38)$$

is given. This equation is expressed with Mixing ratio of water vapor as follows;

$$-\overline{w''q_v''} = -q_{v*}u_* = -\frac{k}{\ln \frac{z}{Z_{0h}}} \frac{k}{\ln \frac{z}{Z_{0m}}} (\bar{q}_v - q_{vsfc})\bar{u} \quad (5.39)$$

This is expressed by using the bulk coefficient of potential temperature defined by (5.32) as follows;

$$-\frac{k}{\ln \frac{z}{Z_{0h}}} \frac{k}{\ln \frac{z}{Z_{0m}}} (\bar{q}_v - q_{vsfc})\bar{u} = -C_h(\bar{q}_v - q_{vsfc})\bar{u} \quad (5.40)$$

Vertical flux of momentum, potential temperature and water vapor mixing ratio in the atmospheric boundary layer have been described with bulk coefficient. In the ground surface process of **CRess**, the exchange of sensible and latent heat between land and atmosphere is described by calculating these bulk coefficients. The effect of ground surface friction is included in a calculation of momentum flux.

## 5.2 Calculations of Atmospheric Boundary Layer, Ground Surface Process and Soil Temperature

The ways of calculation used in JSM (Japan Spectral Model) by Segami et al. (1989) are basically adopted in the calculations of atmospheric boundary layer, ground surface process and soil temperature in **CRess**. Their calculations are one-dimensional vertically. However, following main differences exist between the calculations in **CRess** and those in JSM.

- On the calculation of soil temperature, a four layer model is used in JSM but we can select an arbitrary number ( $m$ ) of layers in **CRess**.
- On the calculation of soil temperature, it starts at  $n - 1$  step and uses a complicated calculation method adopting differential of flux on temperature in JSM. On the other hand in **CRess**, the calculation method is a simple implicit one, just solving simultaneous equations.
- Cloudiness is estimated from relative humidity by the method of Ohno and Isa (1983) in JSM, but it is estimated from mixing ratio of cloud water and cloud ice in **CRess**.
- In JSM, quantities of atmosphere are calculated at first and after that, sensible heat flux and latent heat flux on the ground are calculated from temperature and relative humidity of the atmosphere which are modified by the first calculation of the atmosphere. In **CRess** on the other hand, they are calculated from temperature and mixing ratio of water vapor of  $n - 1$  step and the result is applied for the atmosphere and the ground.
- The calculation of vertical diffusion is conducted for all layers of the atmosphere at every step in JSM and flux entering the atmosphere from the ground affects instantly (within the time step of  $\Delta t$ ) to top of the atmosphere. In **CRess**, the layers affected by the flux can be designed arbitrary.
- On the calculation of vertical diffusion, an implicit method modifying dependent variables is used in JSM but an explicit method is used in **CRess**. Rate of change for time can be calculated and it is added as a forcing term for a time integration.

The calculations of the atmospheric boundary layer, ground surface process and soil temperature are conducted mainly as follows. They are independently conducted for the vertical one-dimension on each horizontal grid.

- Cloudiness is calculated. Usually cloudinesses are calculated for upper, middle and lower layers.
- Solar radiation and downward long wave radiation are calculated.



- Coefficients of fluxes of momentum, sensible heat and latent heat on the ground surface are calculated.
- Horizontal wind speed, potential temperature and mixing ratio of water vapor are modified by calculation of the atmospheric boundary layer process.
- Soil temperature is calculated.

It is notable that, on the ground surface process, flux is not calculated but its coefficient is calculated. The reason is that sensible and latent heat fluxes,

$$H_S = -\rho_a C_p C_h |V_a| (T_a - T_G) \quad (5.41)$$

$${}_l E = -\rho_a \mathcal{L}_v C_h |V_a| \beta [q_{va} - q_{vs}(T_G)] \quad (5.42)$$

are given as external forces on the ground surface, and potential temperature flux<sup>3</sup> and flux of mixing ratio of water vapor,

$$F_\theta = -\rho_a C_h |V_a| (\theta_{va} - \theta_{vG}) \quad (5.43)$$

$$F_{q_v} = -\rho_a C_h |V_a| \beta [q_{va} - q_{vs}(T_G)] \quad (5.44)$$

are given as external forces from the ground surface in the atmosphere. Coefficients common to the two above  $\rho_a C_h |V_a|$  (actually it is different between for potential temperature and for mixing ratio of water vapor) are calculated in the ground surface process and are given to the processes of the atmospheric boundary layer and soil temperature. Then sensible and latent heat fluxes are included in the process of soil temperature, while the effects of fluxes of potential temperature and mixing ratio of water vapor are included in the process of the atmospheric boundary layer. Mixing ratio of water vapor corresponding to the ground surface temperature is calculated in the ground surface process and given to each process simultaneously.

Processes for the calculation of the atmospheric boundary layer, the ground surface process and soil temperature are explained in the next section.

### 5.2.1 Cloudiness

The amount of solar radiation reaching to the ground decreases by the effect of cloud. Cloudiness is used to estimate this effect. The way of Ohno and Isa (1984) to estimate it by an empirical formula of cloudiness and relative humidity is explained in this section.

Cloudiness is nondimensional quantity from 0 to 1. Cloudiness is given as a function of relative humidity for upper, middle and lower layers, respectively. The height of each layer is not exact and 400hPa is for the upper layer, 500 and 700hPa for the middle layer and 850hPa for the lower layer in Ohno and Isa (1984). Upper layer corresponds to 7-7.5km, middle corresponds to 5-6 and 3km, and lower corresponds to 1.5km.

Cloudiness is given by 21 data every 5% ranging from 0 to 100% from the result of Ohno and Isa (1984). When average relative humidity in each layer is defined as  $\overline{Rh}[\%]$ , an integer  $k_n$  is defined as follows;

$$k_n = 1 + 20 \times \frac{\overline{Rh}}{100} \quad (5.45)$$

and cloudiness of lower layer  $CD_L$ , middle layer  $CD_M$  and upper layer  $CD_H$  is defined as follows (written in the format of `data` description of Fortran);

<sup>3</sup>Virtual potential temperature is used here instead of potential temperature. It makes no difference because flux is proportional to the difference of quantities. However, it is notable that it makes big difference when they are used to calculate CAPE.

```

data cdldef / 0.e0, 0.e0, 0.e0, 0.e0, 0.e0, 0.e0, 0.e0, 0.e0,
.           0.e0, 0.e0, 0.e0, 0.e0, .07e0, .11e0, .19e0, .40e0,
.           .85e0, 1.e0, 1.e0, 1.e0, 1.e0 /

data cdmdef / 0.e0, 0.e0, 0.e0, 0.e0, 0.e0, 0.e0, 0.e0, 0.e0,
.           0.e0, 0.e0, 0.e0, .05e0, .12e0, .30e0, .40e0, .50e0,
.           .70e0, .95e0, 1.e0, 1.e0, 1.e0 /

data cdhdef / 0.e0, 0.e0, 0.e0, 0.e0, 0.e0, 0.e0, 0.e0, 0.e0,
.           0.e0, .05e0, .15e0, .30e0, .55e0, .75e0, .87e0, .95e0,
.           1.e0, 1.e0, 1.e0, 1.e0, 1.e0 /

```

---

To calculate relative humidity from mixing ratio of water vapor  $q_v$  and air temperature  $T_a$ , partial pressure of water vapor estimated from  $q_v$  and saturated water vapor pressure at the air temperature is calculated by using Tetens (1930)'s equation. Relative humidity is given from the ratio of the partial pressure of water vapor and saturated water vapor pressure.

Then, air pressure, air temperature and mixing ratio of water vapor are defined as  $p (= \bar{p} + p')$ ,  $T_a$  and  $q_v$ , respectively, partial pressure of water vapor is given as follows;

$$e_a = \frac{q_v p}{\epsilon + (1 - \epsilon) q_v} \tag{5.46}$$

where  $\epsilon$  is the ratio of molecular mass of water vapor to that of dry air, which is 0.622 when mixing ratio of water vapor is provided with the unit of  $[\text{kg kg}^{-1}]$ , and is 622.0 when it is provided with the unit of  $[\text{g kg}^{-1}]$ . The unit of air pressure  $p$  and partial pressure of water vapor  $e_a$  is  $[\text{Pa}]$ .

We assume that air temperature  $T[^\circ\text{C}]$  (note that the unit is degrees Celsius) is given, saturated water vapor pressure of the air temperature (Tetens (1930)'s equation) becomes

$$e_{as} = e_s \cdot 10^{\frac{a \cdot T}{b+T}} \tag{5.47}$$

or

$$e_{as} = e_s \exp\left(\frac{a \cdot T}{b+T} \ln 10\right) \tag{5.48}$$

where  $e_s (= 610.78 \text{ Pa})$  is saturated water vapor pressure at  $0^\circ\text{C}$ . The constant values  $a$  and  $b$  are defined as follows;

	$a$	$b$
$(T \geq 0^\circ\text{C})$ on the water	7.5	237.3
$(T < 0^\circ\text{C})$ on the ice	9.5	265.3

By using these, relative humidity  $Rh$  is gained as follows.

$$Rh = \frac{e_a}{e_{as}} \times 100 \tag{5.49}$$

Cloudiness is derived with this.

By using the method above, cloudiness is decided irrespective of the actual cloud distribution. Because mixing ratios of water substances are calculated in *CReSS*, cloudiness should be estimated from these values. The total amount of cloud water [ $\text{m}^{-2}$ ] is calculated as follows.

$$q_L = \int_{z_s}^{z_L} \rho (q_c + q_r + q_i + q_s + q_g) dz \quad (5.50)$$

$$q_M = \int_{z_L}^{z_M} \rho (q_c + q_r + q_i + q_s + q_g) dz \quad (5.51)$$

$$q_H = \int_{z_M}^{z_t} \rho (q_c + q_r + q_i + q_s + q_g) dz \quad (5.52)$$

where,  $z_L$ ,  $z_M$  are the heights of interfaces between upper, middle and lower layer, respectively. Then, cloudiness is estimated from the relation between these total amount of cloud water and the solar radiation<sup>4</sup>.

## 5.2.2 Solar Radiation and Downward Long Wave Radiation

Solar radiation (short wave radiation) and downward long wave radiation from atmosphere are calculated as a function of cloudiness, zenith angle of the sun, near-surface temperature, water vapor pressure and albedo at the ground surface.

The estimation of solar radiation and atmospheric long wave radiation are summarized as follows.

- Partial pressure of water vapor  $e_a$  is obtained by the temperature and mixing ratio of water vapor  $q_v$  at the near surface layer (around the first or second layer).
- The zenith angle of the sun for each grid at the time is calculated.
- Downward short wave radiation is estimated by the partial pressure of water vapor  $e_a$ , zenith angle and cloud amount with the consideration for albedo.
- Downward long wave radiation is represented as a function of atmospheric temperature  $T_a$ , partial pressure of water vapor  $e_a$  and cloud amount.

Following subsections explain about net (downward) short wave radiation and downward long wave radiation, respectively.

### Partial Pressure of Water Vapor at the Ground Surface

Partial pressure of water vapor at lower layer of atmosphere is estimated utilizing temperature and mixing ratio of water vapor at the lowest atmospheric layer or averaged among some lower atmospheric layers<sup>5</sup>.

### Zenith Angle of the Sun

Zenith angle of the sun is calculated as a function of the number of days from the first day of January  $jday$ , local time  $T_{LC}$  and latitude at the point  $lat$ .

Local time is a function of longitude based on the universal standard time  $UTC$ . It is expressed by

$$T_{LC} = UTC + lon/15 \quad (5.53)$$

<sup>4</sup>Kondo (2000)

<sup>5</sup>In JSM, averaged value of first and second atmospheric layers is used

Angle of the sun  $\phi_s$  is expressed as

$$\phi_s = 23.44 \cos(172 - jday) \quad (5.54)$$

then, zenith angle of the sun is expressed as follows (see fig.5.1):

$$\zeta = \sin(lat) \sin(\phi_s) + \cos(lat) \cos(\phi_s) \cos[0.2618(T_{LC} - 12)]. \quad (5.55)$$

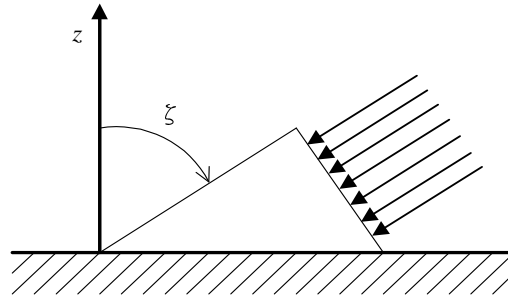


Figure 5.1. Zenith angle of the sun  $\zeta$ .

### Net (Downward) Short Wave Radiation

The solar radiation which reaches at the ground surface is called downward short wave radiation. Considering its reflection from the ground surface with a rate of albedo, downward short wave radiation which is absorbed into the ground is estimated. It is called net downward short wave solar radiation.

In order to estimate total solar radiation which reaches at the ground surface, it is needed to estimate solar radiation which arrive at the top of atmosphere  $S_\infty$ <sup>6</sup>.

$$S_\infty = S_0 \cos \zeta \quad (S_0 = 1,367[\text{W m}^{-2}] : \text{solar constant}) \quad (5.56)$$

where  $\zeta$  is zenith angle of the sun. If we now let

$$a = 1.12 - b - 0.06 \log_{10} e_a \quad (1 \leq e_a \leq 3000[\text{Pa}] : \text{water vapor pressure near surface}) \quad (5.57)$$

$$b = 0.43 + 0.00016e_a \quad (5.58)$$

then solar radiation of the clear weather is

$$S^\downarrow = \begin{cases} S_\infty (a + b \times 10^{-0.13 \sec \zeta}) & \cos \zeta > 0 \\ 0 & \cos \zeta \leq 0 \end{cases} \quad (5.59)$$

Adding the effect of the cloud and its albedo, net downward short wave radiation which is absorbed into the ground is

$$RS^\downarrow = (1 - A_l) S^\downarrow (1 - cd_L \cdot CD_L) (1 - cd_M \cdot CD_M) (1 - cd_H \cdot CD_H) \quad (5.60)$$

where

<sup>6</sup>in actual,  $S_\infty = S_0 (\bar{r}_E / r_E)^2 \cos \zeta$ .  $r_E$  is distance between earth and sun,  $\bar{r}_E$  is its average

$cd_L$	absorbing and refracting effects with the lower cloud	0.7
$cd_M$	absorbing and refracting effects with the middle cloud	0.6
$cd_H$	absorbing and refracting effects with the upper cloud	0.3

As mentioned before, a cloud amount is calculated by empirical formula using relative humidity (Ohno and Isa, 1984). Albedo  $A_l$  is shown in the following table.

land surface	given from data
sea ice or snow surface	constant to 0.6

At the sea surface, there is no need to consider solar radiation because sea temperature is constant during a calculation, so that there is no need to give the albedo.

### Downward Long Wave Radiation from the Atmosphere at the Ground Surface

Using empirical formula of Kondo (1976), downward long wave radiation from atmosphere including effects of cloud and water vapor is

$$L^\downarrow = \varepsilon_G \sigma T_a^4 [1 + (-0.49 + 0.0066\sqrt{e_a})(1 - CD \cdot C \cdot A_c)] \quad (5.61)$$

where  $T_a$  is estimated by a temperature at the lowest atmospheric layer or averaged for some lower atmospheric layers<sup>7</sup> and

$$C = 0.75 - 0.005e_a \quad (5.62)$$

$$A_c = \frac{CD_L + 0.85CD_M + 0.5CD_H}{CD} + 0.1 \frac{N_r}{N} \quad (5.63)$$

$$CD = CD_L + CD_M + CD_H \quad (5.64)$$

<sup>8</sup> Then, utilized constants are expressed in the following table.

$\varepsilon_G$	emissivity for infrared region at the ground surface	0.95	
$\sigma$	Stefan-Boltzman constant	$5.67 \times 10^{-8}$	$\text{W m}^{-2}\text{K}^{-4}$

### 5.2.3 Flux at the Ground Surface

Momentum flux at the ground surface  $\tau_x, \tau_y$  [ $\text{kg m s}^{-1} \text{ m}^{-2} \text{ s}^{-1} = \text{N m}^{-2}$ ] is expressed as follows with bulk coefficient and the absolute value of wind speed.

$$\tau_x = \rho_a C_m |V_a| u_a \quad (5.65)$$

$$\tau_y = \rho_a C_m |V_a| v_a \quad (5.66)$$

Sensible and latent heat flux  $H_S, {}_lE$  [ $\text{J m}^{-2} \text{ s}^{-1} = \text{W m}^{-2}$ ] is given as

<sup>7</sup>In JSM, averaged value of first and second atmospheric layers is used

<sup>8</sup> $N_r/N$  means appearance rate of precipitating time. In temporal integration, it is set to 0 in case of no precipitation and to 1 in case of precipitation through the time. In JMS, on the other hand, it is set to 0 constantly though there is no confidence to be finest method.

$$H_S = -\rho_a C_p C_h |V_a| (T_a - T_G) \quad (5.67)$$

$${}_l E = -\rho_a \mathcal{L}_v C_h |V_a| \beta [q_{va} - q_{vs}(T_G)] \quad (5.68)$$

$$(5.69)$$

where  $a$  and  $G$  mean the first layer of the atmosphere and the ground surface (first layer of the ground temperature),  $C_m$  and  $C_h$  are the bulk coefficient (with no dimension) of the momentum, and heat and water vapor,  $q_{vs}(T_G)$  is saturated mixed ratio of ground temperature  $T_G$ ,  $\beta$  is evapotranspiration coefficient and  $\mathcal{L}_v$  is latent heat of evaporation of water. They are multiplied atmospheric density and rewritten to equivalent temperature flux  $F_\theta$  [ $\text{kg K m}^{-2} \text{s}^{-1}$ ] and water vapor mixed ratio flux  $F_{q_v}$  [ $\text{kg m}^{-2} \text{s}^{-1}$ ] as

$$\begin{aligned} F_\theta &= \frac{H_S}{C_p} \left( \frac{p_0}{p} \right)^{\frac{R}{C_p}} = -\rho_a C_h |V_a| (\theta_{va} - \theta_{vG}) \\ &= -\rho_a u_* \theta_* \end{aligned} \quad (5.70)$$

$$\begin{aligned} F_{q_v} &= E = -\rho_a C_h |V_a| \beta [q_{va} - q_{vs}(T_G)] \\ &= -\rho_a u_* q_{v*} \end{aligned} \quad (5.71)$$

where  $\theta_*$  is scale of potential temperature (frictional temperature) and  $q_{v*}$  is scale of mixing ratio (frictional mixing ratio). Friction velocity  $u_*$  is defined as

$$u_*^2 = C_m |V_a|^2 \quad (5.72)$$

where  $|V_a|$  is absolute value of ground surface wind speed which expressed as

$$|V_a| = (u_a^2 + v_a^2)^{\frac{1}{2}} \quad (5.73)$$

Virtual potential temperature  $\theta_v$  is

$$\theta_v = \frac{1 + \epsilon_{iv} q_v}{1 + q_v} \theta \quad (5.74)$$

where  $\epsilon_{iv}$  is ratio of molecular weight of water vapor to that of dry air.

Non-dimensional bulk coefficients  $C_m, C_h$  are expressed by the scheme of Louis et al. (1980) considering momentum, heat and roughness parameter of water vapor as follows:

$$C_m = \left[ \frac{k}{\ln \left( \frac{z_a}{Z_{0m}} \right)} \right]^2 f_m(Ri, z_a, Z_{0m}) \quad (5.75)$$

$$C_h = \frac{k}{\ln \left( \frac{z_a}{Z_{0m}} \right)} \frac{k}{\ln \left( \frac{z_a}{Z_{0h}} \right)} f_h(Ri, z_a, Z_{0m}, Z_{0h}) \quad (5.76)$$

where  $k$  ( $=0.4$ ),  $z_a$ ,  $Z_{0m}$ ,  $Z_{0h}$  and  $Ri$  are Kalman coefficient, height of the first atmospheric layer, roughness of momentum, roughness of heat and water vapor, and Richardson number, respectively.

In unstable case ( $Ri < 0$ ),

$$f_m = 1 - \frac{2b \cdot Ri}{1 + 3bc \left[ \frac{k}{\ln\left(\frac{z_a}{Z_{0m}}\right)} \right]^2 \left( -\frac{Ri \cdot z_a}{Z_{0m}} \right)^{\frac{1}{2}}} \quad (5.77)$$

$$f_h = 1 - \frac{3b \cdot Ri}{1 + 3bc \frac{k}{\ln\left(\frac{z_a}{Z_{0m}}\right)} \frac{k}{\ln\left(\frac{z_a}{Z_{0h}}\right)} \left( -\frac{Ri \cdot z_a}{Z_{0h}} \right)^{\frac{1}{2}}} \quad (5.78)$$

$$b = c = 5 \quad (5.79)$$

In stable case ( $Ri > 0$ ),

$$f_m = \frac{1}{1 + 2b \cdot Ri \cdot \sqrt{1 + d \cdot Ri}} \quad (5.80)$$

$$f_h = \frac{1}{1 + 3b \cdot Ri \cdot \sqrt{1 + d \cdot Ri}} \quad (5.81)$$

$$b = d = 5 \quad (5.82)$$

Roughness parameters of momentum  $Z_{0m}$  and of heat and water vapor  $Z_{0h}$  are shown in the following table. Roughness of the sea surface is refined at the every calculating step.

roughness	land surface	sea surface	sea ice or snow surface
$Z_{0m}$	given from data set	calculating as a function of $u_*$	optional constant value
$Z_{0h}$	constant value (0.1 m)	same with $Z_{0m}$	same with $Z_{0m}$

Standing on the above, actual calculation of land surface flux is shown as follows.

**(1) Calculation of Absolute Value of Horizontal Wind Speed**

Its absolute value  $|V_a|$  is calculated by using horizontal wind speed at the first atmospheric layer ( $u_a, v_a$ ) (??).

**(2) Calculataion of Saturated Mixing Ratio of Water Vapor Corresponding to Ground Surface Temperature**

Saturated mixing ratio of water vapor  $q_{vs}(T_G)$  corresponding to ground temperature at the first ground layer or sea surface temperature is obtained by the equation of Tetens (1930) or (5.48).

From partial pressure of saturated water vapor by the equation of Tetens (1930), saturated mixing ratio of water vapor [kg/kg] is expressed as

$$q_{vs}(T_G) = \epsilon \frac{e_{as}}{p} \tag{5.83}$$

Originally this is the equation to define specific humidity, but we can regard it as mixing ratio with a small error (difference).  $p$  is expressed as  $p = \bar{p} + p'$ , and  $\epsilon$  is ratio of molecular weight of dry air to water vapor.

**(3) Definition of Mixing Ratio of Water Vapor at the Ground Surface**

Mixing ratio of water vapor at the ground surface is defined for land, sea, sea ice and snow surfaces, respectively. Considering evapotranspirate efficiency  $WET$  which is given to land surface only, it is defined as follows:

for land surface,

$$q_{vsfc} = \beta [q_{vs}(T_G) - q_{va}] + q_{va} \tag{5.84}$$

for sea, sea ice and snow surfaces,

$$q_{vsfc} = q_{vs}(T_G) \tag{5.85}$$



**(4) Calculation of Richardson Number**

Richardson number of the ground surface is calculated in order to decide stability at the ground surface and to calculate  $f_m, f_h$  in the scheme of Louis et al. (1980).

Richardson number  $Ri$  can be expressed as

$$Ri = \frac{gz_a \Delta\theta_v}{\theta_{va} |V_a|^2} \tag{5.86}$$

where  $\epsilon_{iv}$  indicate ratio of the molecular weight between dry air and water vapor, then

$$\theta_{vG} = T_G \frac{1 + \epsilon_{iv} q_{vsfc}}{1 + q_{vsfc}} \left( \frac{p_0}{p} \right)^{\frac{R}{c_p}} \tag{5.87}$$

$$\Delta\theta_v = \theta_{va} - \theta_{vG} \tag{5.88}$$

and  $z_a$  is the height of first atmospheric layer and  $g$  is the acceleration of gravity.

**(5) Calculation of  $Z_{0m}$  and  $u_*$  at the Sea Surface at the First Step by Iteration**

Though both roughnesses at the land surface  $Z_{0m}$  and  $Z_{0h}$  are constant value during the calculation, those at the sea surface are estimated by the iteration with frictional velocity considering its dependence on wind speed. The iteration method is used at the only first step and  $Z_{0m}$  is estimated by frictional velocity of the one step before;  $u_*^{n-1}$  on and after the second step.

$Z_{0m}$  is expressed as a function of  $u_*$  based on the formula of Kondo (1975) as follows:

$$Z_{0m} = -34.7 \times 10^{-6} + 8.28 \times 10^{-4} u_* \quad \text{for } u_* \leq 1.08 [\text{m s}^{-1}] \tag{5.89}$$

$$Z_{0m} = -0.277 \times 10^{-2} + 3.39 \times 10^{-3} u_* \quad \text{for } u_* > 1.08 [\text{m s}^{-1}] \tag{5.90}$$

where maximum of  $Z_{0m}$  is  $1.5 \times 10^{-5}$  m. From (5.72),  $u_*$  would be

$$u_* = |V_a| \sqrt{C_m} \tag{5.91}$$

Although bulk coefficient  $C_m$  is estimated by (5.75), its coefficient  $f_m$  is depended on Richardson number which given by (5.86); in unstable case, it is estimated by (5.77) while in stable case, it by (5.80).

$Z_{0m}$  is calculated with 5.89 or 5.90 using the friction velocity  $u_*$ . The calculation using the roughness is iterated until the value of  $Z_{0m}, u_*$  converges. Then, we can decide  $Z_{0m}, u_*$  at the sea surface.

**(6) Calculation of  $f_m, f_h$  for the Whole of a Region**

In the subsection (5), the roughness parameter of momentum at the sea surface  $Z_{0m}$  is estimated. Roughness parameters of heat and water vapor  $Z_{0h}$  are defined as  $Z_{0h} = Z_{0m}$ . At the land surface, the roughness of momentum  $Z_{0m}$  is given as a data and the roughnesses of heat and water vapor  $Z_{0h}$  are defined as constant value 0.1 m. In case of sea ice and snow cover, roughnesses of momentum, heat and water vapor are defined as constant values.

Roughness parameters of momentum, heat and water vapor for the whole of a region are decided, so coefficients  $f_m, f_h$  which are used in the scheme of Louis et al. (1980) and multiply by bulk coefficient are calculated by using (5.77), (5.78), (5.80), and (5.81).

**(7) Calculation of Bulk Coefficients  $C_m, C_h$  for the Whole of a Region**

Using  $f_m, f_h$  and roughness parameters  $Z_{0m}$  and  $Z_{0h}$  which are estimated from (5) and (6), bulk coefficients  $C_m, C_h$  are calculated with (5.75) and (5.76).

**(8) Calculation of Friction Velocity  $u_*$  for the Whole of a Region**

According to (5.91), friction velocity  $u_*$  is estimated for the whole of a region using bulk coefficients of momentum  $C_m$  and  $|V_a|$ .

**(9) Calculation of Correction Term for Potential Temperature and Flux of Mixing Ratio Used in Kondo(1975).**

The correction term by Kondo (1975) is estimated to calculate fluxes of water vapor and heat at the sea surface.

Kondo (1975) suggest the following equations of correction factor  $B_h^{-1}$  and  $B_e^{-1}$  by the observation.

At the sea surface,

$$B_h^{-1} = \frac{1}{k} \ln \left( \frac{\nu + ku_* Z_{0m}}{\mathcal{D}_a} \right) \quad (5.92)$$

$$B_e^{-1} = \frac{1}{k} \ln \left( \frac{\nu + ku_* Z_{0m}}{\mathcal{D}_v} \right) = B_h^{-1} + \frac{1}{k} \ln \left( \frac{\mathcal{D}_a}{\mathcal{D}_v} \right) \quad (5.93)$$

or,

$$B_h^{-1} = \frac{1}{k} \ln (0.71 + 4.64 \times 10^4 ku_* Z_{0m}) \quad (5.94)$$

$$B_e^{-1} = B_h^{-1} - 0.168 \frac{1}{k} \quad (5.95)$$

At the land surface, sea ice and snow cover,

$$B_h^{-1} = 0 \quad (5.96)$$

$$B_e^{-1} = 0 \quad (5.97)$$

where  $\nu$  is kinetic viscosity coefficient of air [ $\text{m}^2 \text{s}^{-1}$ ],  $\mathcal{D}_a$  is diffusion coefficient of air [ $\text{m}^2 \text{s}^{-1}$ ] and  $\mathcal{D}_v$  is diffusion coefficient of water vapor [ $\text{m}^2 \text{s}^{-1}$ ]<sup>9, 10</sup>.

<sup>9</sup>Kinetic viscosity coefficient of air  $\nu$  [ $\text{m}^2 \text{s}^{-1}$ ] is expressed by,

$$\nu = \nu_0 \frac{101325}{p} \left( \frac{T}{273.16} \right)^{1.754} \quad (5.98)$$

$$\nu_0 = 1.328 \times 10^{-5} \quad (5.99)$$

Diffusion coefficient of air  $\mathcal{D}_a$  [ $\text{m}^2 \text{s}^{-1}$ ] is expressed by,

$$\mathcal{D}_a = \mathcal{D}_{a0} \frac{101325}{p} \left( \frac{T}{273.16} \right)^{1.78} \quad (5.100)$$

$$\mathcal{D}_{a0} = 1.87 \times 10^{-5} \quad (5.101)$$

Diffusion coefficient of water vapor  $\mathcal{D}_v$  [ $\text{m}^2 \text{s}^{-1}$ ] is expressed by,

$$\mathcal{D}_v = \mathcal{D}_{v0} \frac{101325}{p} \left( \frac{T}{273.16} \right)^{1.81} \quad (5.102)$$

$$\mathcal{D}_{v0} = 2.23 \times 10^{-5} \quad (5.103)$$

However, it is enough to use approximation in the text.

<sup>10</sup> $B_h$  is Stanton number at the bottom layer and  $B_e$  is Dalton number at the bottom layer (大気科学講座 1 P95).

**(10) Calculation of Flux Coefficients of Potential Temperature and Mixing Ratio of Water Vapor**

In order to simplify the estimation of fluxes of potential temperature and mixing ratio of water vapor, the coefficients  $E_\theta$  and  $E_{q_v}$  are calculated.

$$E_\theta = \frac{\theta_*}{\theta_{va} - \theta_{vG}} = \left[ \frac{u_*}{C_h |V_a|} + B_h^{-1} \right]^{-1} \quad (5.104)$$

$$E_{q_v} = \frac{q_{v*}}{q_{va} - q_{vsfc}} = \left[ \frac{u_*}{C_h |V_a|} + B_e^{-1} \right]^{-1} \quad (5.105)$$

Though same bulk coefficient  $C_h$  has been used in case of water vapor and heat fluxes, the use of above coefficients adopting  $B_h^{-1}$  and  $B_e^{-1}$  make different coefficient value in case of the sea surface. In cases of the land surface, sea ice and snow cover, however, these values become equally 0.

**(11) Calculation of Flux**

Standing on the above, fluxes of momentum, sensible heat, latent heat, potential temperature and mixing ratio of water vapor at the ground surface are given as follows.

At first, absolute value of momentum flux is expressed by

$$\tau = \rho_a u_*^2 \quad (5.106)$$

and its  $x$  and  $y$  components are

$$\tau_x = \tau \frac{u_a}{|V_a|} \quad (5.107)$$

$$\tau_y = \tau \frac{v_a}{|V_a|} \quad (5.108)$$

Secondly, sensible heat and latent heat fluxes are expressed by

$$H_S = -\rho_a C_p u_* E_\theta (T_a - T_G) \quad (5.109)$$

$$L E = -\rho_a \mathcal{L} u_* E_{q_v} (q_{va} - q_{vsfc}) \quad (5.110)$$

and, fluxes of potential temperature  $F_\theta$  [ $\text{kg K m}^{-2} \text{s}^{-1}$ ] and mixing ratio of water vapor  $F_{q_v}$  [ $\text{kg m}^{-2} \text{s}^{-1}$ ] which are multiplied by atmospheric density are expressed as follows:

$$F_\theta = -\rho_a u_* E_\theta (\theta_{va} - \theta_{vG}) \quad (5.111)$$

$$F_{q_v} = -\rho_a u_* E_{q_v} (q_{va} - q_{vsfc}) \quad (5.112)$$

At land surface,  $q_{va} - q_{vsfc} = \beta [q_{va} - q_{vs}(T_G)]$  is obtained using (5.84). Then, (5.110) and (5.112) take account of evaporation efficiency  $\beta$  through  $q_{vsfc}$ .

The above is the calculation of ground surface flux, however, (5.109) and (5.110) are used in the calculating process of ground temperature actually. (5.111) and (5.112) are used in the calculating process of atmospheric boundary layer. Therefore, this process returns following three coefficients which are common to these equations,

$$C_w = \rho_a u_*^2 \frac{1}{|V_a|} \tag{5.113}$$

$$C_\theta = \rho_a u_* E_\theta \tag{5.114}$$

$$C_{q_v} = \rho_a u_* E_{q_v} \tag{5.115}$$

and mixing ratio at the ground surface  $q_{vsfc}$  which include  $\beta$  in case of the land surface. Utilising these coefficients,  $x, y$  components of momentum flux and flux of mixing ratio are expressed as follows:

$$\tau_x = C_w u_a \tag{5.116}$$

$$\tau_y = C_w v_a \tag{5.117}$$

$$F_\theta = C_\theta (\theta_{va} - \theta_{vG}) \tag{5.118}$$

$$F_{q_v} = C_{q_v} (q_{va} - q_{vsfc}) \tag{5.119}$$

When we don't use an implicit method in the calculation of atmospheric boundary layer process, we can calculate fluxes of momentum, sensible heat, latent heat, potential temperature and mixing ratio, instead.

**(12) Calculation of Monitor Data at the Ground Surface**

Temperature and wind speed at the ground surface are important to be compared with observed physical parameter at the ground surface. Surface boundary layer touching the ground surface is called as constant flux layer where vertical flux is constant regardless of height. According to the nature, temperature and mixing ratio of water vapor at a height of 1.5 m and wind speed at 10 m are calculated as a monitor value.

The method of calculation is simple.  $u_*$  is estimated previously and wind speed at a height of 10 m is calculated by (5.72) or (5.91) as

$$|V_{a10}| = \frac{u_*}{\sqrt{C_{m10}}} \tag{5.120}$$

$C_{m10}$  is estimated by (5.75)–(5.81) in which  $f_m$  and  $C_m$  are calculated with application of  $z_a = 10$ . Since wind direction is constant in vertical within the surface ground layer,  $x$  and  $y$  components are

$$u_{x10} = |V_{a10}| \frac{u_a}{|V_a|} \tag{5.121}$$

$$v_{y10} = |V_{a10}| \frac{v_a}{|V_a|} \tag{5.122}$$

In case of temperature and mixing ratio of water vapor,  $z_a = 1.5$  is used to calculate  $f_m$  and  $C_m$  by (5.75)–(5.81). Using the obtained bulk coefficient  $C_{m1.5}$  and friction velocity  $u_*$  at a height of 1.5 m, wind speed at a height of 1.5 m is calculated as follows.

$$|V_{a1.5}| = \frac{u_*}{\sqrt{C_{m1.5}}} \tag{5.123}$$

Simultaneously, bulk coefficient  $C_{h1.5}$  of potential temperature and mixing ratio at a height of 1.5m is calculated. Substituting these coefficients into (5.104) and (5.105), we get

$$E_{\theta1.5} = \left[ \frac{u_*}{C_{h1.5}|V_{a1.5}|} + B_h^{-1} \right]^{-1} \tag{5.124}$$

$$E_{q_v1.5} = \left[ \frac{u_*}{C_{h1.5}|V_{a1.5}|} + B_e^{-1} \right]^{-1} \tag{5.125}$$

Virtual potential temperature at a height of 1.5m  $\theta_{v1.5}$  is related to that at the surface layer  $\theta_{va}$  and at the ground  $\theta_{vG}$  as

$$\frac{\theta_{v1.5} - \theta_{vG}}{\theta_{va} - \theta_{vG}} = \frac{E_{\theta}}{E_{\theta1.5}} \quad (5.126)$$

Then, virtual potential temperature at height of 1.5m is given by

$$\theta_{v1.5} = \theta_{vG} + (\theta_{va} - \theta_{vG}) \frac{E_{\theta}}{E_{\theta1.5}} \quad (5.127)$$

Similarly, mixing ratio of water vapor at height of 1.5m  $q_{va1.5}$  is also related to that at the surface layer  $q_{va}$  and at the ground  $q_{vsfc}$  as

$$\frac{q_{va1.5} - q_{vsfc}}{q_{va} - q_{vsfc}} = \frac{E_{qv}}{E_{qv1.5}} \quad (5.128)$$

Then, mixing ratio of water vapor at a height of 1.5m  $q_{va1.5}$  is given by

$$q_{va1.5} = q_{vsfc} + (q_{va} - q_{vsfc}) \frac{E_{qv}}{E_{qv1.5}} \quad (5.129)$$

where  $q_{vsfc}$  includes the effect of evapotranspiration coefficient.

### (13) Calculation of the Roughness Parameter at the Sea Surface $Z_{0m}$ for the Next Step

At the end of this calculation process, roughness  $Z_{0m}$  which relate to the momentum at the sea surface is calculated for the next step. Adopting the friction velocity  $u_*$  into (5.89) and (5.90),  $Z_{0m}$  at the sea surface is calculated and conserved until the next step.

## 5.2.4 Atmospheric Boundary Layer Process

Momentum, sensible heat and latent heat which are given to the atmosphere from the ground surface are transported upward by the vertical diffusion. Although diffusion is caused by the turbulent flow, we should consider at some points for the diffusion in the boundary layer.

In CRESS, the vertical diffusion other than the usual turbulent flow is taken into consideration in the atmospheric boundary layer process. The height of boundary layer is specified by the user. Since vertical diffusion becomes so small that it is far from the ground surface, the height of which diffusion attains to should just be given. Vertical diffusion of the atmospheric boundary layer process is fundamentally calculated using level 2 of the turbulent closure model by Mellor and Yamada (1974) which is used in JSM (Segami et al. (1989)), and that is also calculation about only 1 dimension in vertical.

The equation of vertical diffusion is expressed with a  $z$  coordinate as follows.

$$\frac{\partial u}{\partial t} = \frac{1}{\rho} \frac{\partial}{\partial z} \left( \rho K_m \frac{\partial u}{\partial z} \right) \quad (5.130)$$

$$\frac{\partial v}{\partial t} = \frac{1}{\rho} \frac{\partial}{\partial z} \left( \rho K_m \frac{\partial v}{\partial z} \right) \quad (5.131)$$

$$\frac{\partial \theta_v}{\partial t} = \frac{1}{\rho} \frac{\partial}{\partial z} \left( \rho K_h \frac{\partial \theta_v}{\partial z} \right) \quad (5.132)$$

$$\frac{\partial q_v}{\partial t} = \frac{1}{\rho} \frac{\partial}{\partial z} \left( \rho K_h \frac{\partial q_v}{\partial z} \right) \quad (5.133)$$

where the turbulent mixing coefficient added  $K_0 = 1.0[\text{m}^2 - \text{s}^{-1}]$  was used. That is,

$$K_m = K_0 + l^2 \left| \frac{\partial \mathbf{V}}{\partial z} \right| S_m \tag{5.134}$$

$$K_h = K_0 + l^2 \left| \frac{\partial \mathbf{V}}{\partial z} \right| S_h \tag{5.135}$$

where  $S_m, S_h$  are the function of flux Richardson number (Mellor and Yamada, 1974) and length scale  $l$  is

$$l = \frac{kz}{1 + \frac{kz}{l_0}} \tag{5.136}$$

$k$  is Karman's constant and  $l_0$  is expressed by making  $E$  into turbulent energy as follows.

$$l_0 = 0.10 \times \frac{\int_{z_s}^{\infty} \rho E z dz}{\int_{z_s}^{\infty} \rho E dz} \tag{5.137}$$

Based on the above, the calculation method of boundary layer process is as follows.

**(1) Calculation of Air Density between Grids**

Air density  $\rho_k^w$  is calculated between each vertical grid which is needed in the case of difference specialization (see Fig. ref fig:pbl grid setting).

**(2) Calculation of the square of vertical shear**

Because of the use for following calculation of Richardson number, or of flux, the absolute value of vertical shear is calculated here.

$$\left| \frac{\partial \mathbf{V}_k}{\partial z} \right|^2 = \left( \frac{\partial u_k}{\partial z} \right)^2 + \left( \frac{\partial v_k}{\partial z} \right)^2 \tag{5.138}$$

**(3) Gradient Richardson Number**

In the level 2 by Mellor and Yamada (1974), flux Richardson number  $R_f$  is given as a function of gradient Richardson number  $R_i$ . It is defined as follows.

$$R_i = \frac{g \frac{\partial \theta_v}{\partial z}}{\bar{\theta}_v \left| \frac{\partial \mathbf{V}_k}{\partial z} \right|^2} \quad (5.139)$$

where  $\bar{\theta}_v = \frac{1}{2}(\theta_{vk+1} + \theta_{vk})$ .

**(4) Flux Richardson Number**

Following to Mellor and Yamada (1974), flux Richardson number  $R_f$  is given as follows.

$$R_f = 0.725 \left( R_i + 0.186 - \sqrt{R_i^2 - 0.316R_i + 0.0346} \right) \quad (5.140)$$

**(5) Calculation of  $\tilde{S}_H$  and  $\tilde{S}_M$**

The constants used by Mellor and Yamada (1974) are as follows.

$A_1$	$A_2$	$B_1$	$B_2$	$C_1$
0.78	0.78	15.0	8.0	0.056

Using these values, the following constants are estimated.

$$\gamma_1 \equiv \frac{1}{3} - \frac{2A_1}{B_1} = 0.2293333 \quad (5.141)$$

$$\gamma_2 \equiv \frac{B_2}{B_1} + \frac{6A_1}{B_1} = 0.8453333 \quad (5.142)$$

and,

$$\Gamma \equiv \frac{R_f}{1 - R_f} \quad (5.143)$$

Using these values,  $\tilde{S}_H$  and  $\tilde{S}_M$  are expressed as follows.

$$\tilde{S}_H = 3A_2(\gamma_1 - \gamma_2\Gamma) \quad (5.144)$$

$$\tilde{S}_M = 3A_1(\gamma_1 - \gamma_2\Gamma) \frac{\gamma_1 - C_1 - (6A_1 + 3A_2) \frac{\Gamma}{B_1}}{\gamma_1 - \gamma_2\Gamma + 3A_1 \frac{\Gamma}{B_1}} \quad (5.145)$$

These can be simplified more in the actual calculation. Eliminating  $\Gamma$  and considering  $A_1 = A_2$ , they can express as

$$\tilde{S}_H = 3A_2 \frac{\gamma_1 - (\gamma_1 + \gamma_2)R_f}{1 - R_f} \quad (5.146)$$

$$\tilde{S}_M = \tilde{S}_H \frac{X_1 - X_2 R_f}{X_3 - X_4 R_f} \quad (5.147)$$

The constants used here are as follows.

$$\gamma_1 + \gamma_2 = 1.074667 \quad (5.148)$$

$$3A_2 = 2.34 \quad (5.149)$$

$$X_1 = 0.173333 \quad (5.150)$$

$$X_2 = 0.641333 \quad (5.151)$$

$$X_3 = 0.229333 \quad (5.152)$$

$$X_4 = 0.918667 \quad (5.153)$$

#### (6) Calculation of $S_H$ and $S_M$

Using  $\tilde{S}_H$  and  $\tilde{S}_M$  which are calculated previously,  $S_H$  and  $S_M$  are estimated. As the factor common to both  $S_H$  and  $S_M$ ,

$$S_e = \sqrt{B_1(1 - R_f) \left| \frac{\partial \mathbf{V}}{\partial z} \right|^2} \tilde{S}_M \quad (5.154)$$

is defined. Then,  $S_H$  and  $S_M$  are expressed as follows.

$$S_H = S_e \tilde{S}_H \quad (5.155)$$

$$S_M = S_e \tilde{S}_M \quad (5.156)$$

#### (7) Calculation of Standard of a Length Scale $l_0$

In Mellor and Yamada (1974), standard of a length scale  $l_0$  is calculated as a function of turbulent kinetic energy (5.137). A turbulent kinetic energy which is estimated in Chapter ?? "diffusion in sub-grid scale" is used.

#### (8) Calculation of a Length Scale $l$

Length scale  $l$  is given by (5.136) as follows.

$$l = \frac{z}{\frac{1}{k} + \frac{z}{l_0}} \quad (5.157)$$

#### (9) Calculation of $K_m$ and $K_h$

Although  $K_m$  and  $K_h$  are defined as (5.134) and (5.135), the absolute value of vertical shear is already applied to  $S_M$  and  $S_H$  in the above-mentioned calculation. Then, they are rewritten as follows.

$$K_m = K_0 + l^2 S_M \quad (5.158)$$

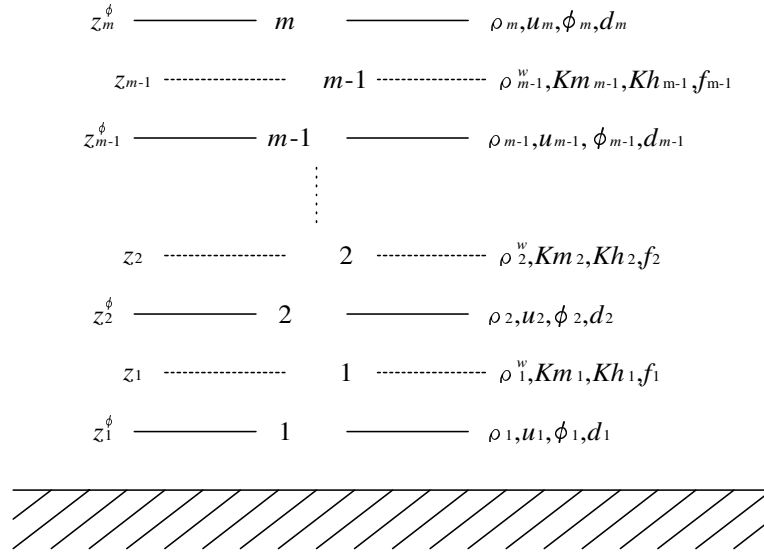
$$K_h = K_0 + l^2 S_H \quad (5.159)$$



Here,  $K_0 = 1.0 \text{ [m}^2 \text{ s}^{-1}]$  .

**(10)The Finite Difference Method for Equation of Vertical Diffusion**

The vertical distributions of  $K_m$  and  $K_h$  in boundary layer were calculated on the level between grids. Then, the vertical diffusion equations (5.130)–(5.133) are solved and a time changing rate and a correction value (integration value) are calculated in the cases of the explicit method and the implicit method, respectively.



**Figure 5.2.** vertical grid setting used for atmospheric boundary layer process

We conduct a discretization for these equations with the arrangement as shown in Fig.5.2.

$$\frac{u_k^n - u_k^{n-1}}{\Delta t} = \frac{1}{\rho_k} \frac{1}{z_k - z_{k-1}} \left( \rho_k^w K_{m_k} \frac{u_{k+1}^N - u_k^N}{z_{k+1}^\phi - z_k^\phi} - \rho_{k-1}^w K_{m_{k-1}} \frac{u_k^N - u_{k-1}^N}{z_k^\phi - z_{k-1}^\phi} \right) \quad (5.160)$$

$$\frac{\phi_k^n - \phi_k^{n-1}}{\Delta t} = \frac{1}{\rho_k} \frac{1}{z_k - z_{k-1}} \left( \rho_k^w K_{m_k} \frac{\phi_{k+1}^N - \phi_k^N}{z_{k+1}^\phi - z_k^\phi} - \rho_{k-1}^w K_{m_{k-1}} \frac{\phi_k^N - \phi_{k-1}^N}{z_k^\phi - z_{k-1}^\phi} \right) \quad (5.161)$$

where  $u_k^n$  and  $\phi_k^n$  (subscript with bottom  $k$  represents a vertical grid number and that with top  $n$  represents the time step) are used as a representative of the velocity component and of the scalar, respectively.  $N$  which appears in the right-hand side of difference equation becomes discretization of the explicit method in the case of  $N = n - 1$ , and becomes discretization of the implicit method in the case of  $N = n$ .

Coefficients of surface fluxes;  $C_w$  in (5.113),  $C_\theta$  in (5.114) and  $C_{q_v}$  in (5.115) are used at the bottom layer ( $k = 1$ ).

$$\frac{u_1^n - u_1^{n-1}}{\Delta t} = \frac{1}{\rho_1} \frac{1}{z_1} \left[ \rho_1^w K_{m_1} \frac{u_2^N - u_1^N}{z_2^\phi - z_1^\phi} - C_w (u_1^N - u_{sfc}^N) \right] \quad (5.162)$$

$$\frac{\phi_1^n - \phi_1^{n-1}}{\Delta t} = \frac{1}{\rho_1} \frac{1}{z_1} \left[ \rho_1^w K_{m_1} \frac{\phi_2^N - \phi_1^N}{z_2^\phi - z_1^\phi} - C_\theta (\phi_1^N - \phi_{sfc}^N) \right] \quad (5.163)$$

where

$$u_{sfc}^N = 0 \quad (5.164)$$

and representative of the scalar is  $\phi$ . In these equations,  $z_0 = 0$  can define the following signs,

$$d_k = \frac{1}{\rho_k} \frac{1}{z_k - z_{k-1}} \quad (5.165)$$

$$f_k = \rho_k^w K_{m_k} \frac{1}{z_{k+1}^\phi - z_k^\phi} \quad (5.166)$$

We can understand  $K_{h_k}$  using  $f_k$ .

In the case of  $k = 1$ ,

$$\frac{u_1^n - u_1^{n-1}}{\Delta t} = d_1 [f_1(u_2^N - u_1^N) - C_w u_1^N] \quad (5.167)$$

$$\frac{\phi_1^n - \phi_1^{n-1}}{\Delta t} = d_1 [f_1(\phi_2^N - \phi_1^N) - C_\phi(\phi_1^N - \phi_{sfc}^N)] \quad (5.168)$$

In the case of  $k \geq 2$ ,

$$\frac{u_k^n - u_k^{n-1}}{\Delta t} = d_k [f_k(u_{k+1}^N - u_k^N) - f_{k-1}(u_k^N - u_{k-1}^N)] \quad (5.169)$$

$$\frac{\phi_k^n - \phi_k^{n-1}}{\Delta t} = d_k [f_k(\phi_{k+1}^N - \phi_k^N) - f_{k-1}(\phi_k^N - \phi_{k-1}^N)] \quad (5.170)$$

About  $k = m - 1$ ,

$$\frac{u_{m-1}^n - u_{m-1}^{n-1}}{\Delta t} = d_{m-1} [f_{m-1}(u_m^{n-1} - u_{m-1}^N) - f_{m-2}(u_{m-1}^N - u_{m-2}^N)] \quad (5.171)$$

$$\frac{\phi_{m-1}^n - \phi_{m-1}^{n-1}}{\Delta t} = d_{m-1} [f_{m-1}(\phi_m^{n-1} - \phi_{m-1}^N) - f_{m-2}(\phi_{m-1}^N - \phi_{m-2}^N)] \quad (5.172)$$

where since  $u_m^{n-1}$  and  $\phi_m^{n-1}$  in the case of  $k = m - 1$  are given as boundary condition, this process is eternal.

In the case of  $N = n - 1$ , these equations are solved by the explicit method and time change rate is acquired. In the case of  $N = n$ , they are solved by the implicit method and integration value is acquired.

### 5.2.5 Ground Temperature

The ground temperature is calculated by the model with  $m$  layers.

Figure 5.3 shows a ground grid setting used for calculation of ground temperature. In the calculation, only the vertical heat conduction is taken into consideration, while the horizontal heat diffusion does not take into consideration.

Here, time evolution equation system of the ground temperature in the model with  $m$  layer is expressed as follows.

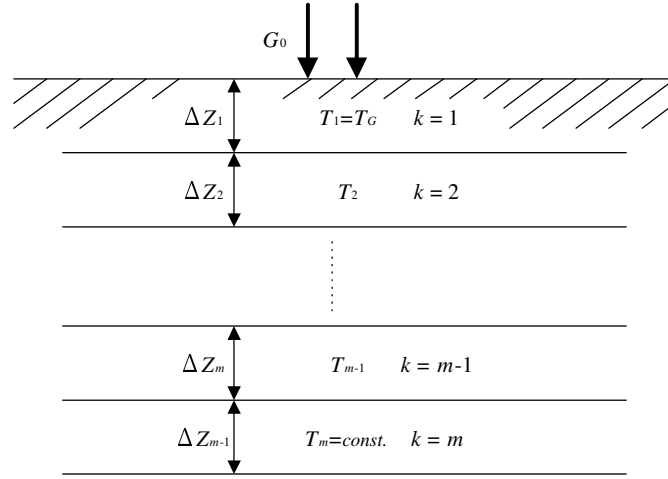


Figure 5.3. A underground setup used for calculation of ground temperature.

$$\begin{aligned}
 \frac{\partial T_1}{\partial t} &= \frac{G_0}{\rho_G C_G \Delta Z_1} + \frac{2\nu_G}{\Delta Z_1(\Delta Z_2 + \Delta Z_1)}(T_2 - T_1) \\
 \frac{\partial T_2}{\partial t} &= -\frac{2\nu_G}{\Delta Z_2(\Delta Z_2 + \Delta Z_1)}(T_2 - T_1) + \frac{2\nu_G}{\Delta Z_2(\Delta Z_3 + \Delta Z_2)}(T_3 - T_2) \\
 &\dots\dots\dots \\
 \frac{\partial T_k}{\partial t} &= -\frac{2\nu_G(T_k - T_{k-1})}{\Delta Z_k(\Delta Z_k + \Delta Z_{k-1})} + \frac{2\nu_G(T_{k+1} - T_k)}{\Delta Z_k(\Delta Z_{k+1} + \Delta Z_k)} \\
 &\dots\dots\dots \\
 \frac{\partial T_{m-1}}{\partial t} &= -\frac{2\nu_G(T_{m-1} - T_{m-2})}{\Delta Z_{m-1}(\Delta Z_{m-1} + \Delta Z_{m-2})} + \frac{2\nu_G(T_m - T_{m-1})}{\Delta Z_{m-1}(\Delta Z_m + \Delta Z_{m-1})}
 \end{aligned} \tag{5.173}$$

In the  $m$  layer, temperature  $T_m$  is constant during a calculation period. Heat capacity of the ground per unit volume  $\rho_G C_G$  and heat diffusion coefficient  $\nu_G$  are

$$\rho_G C_G = 2.3 \times 10^6 \quad [\text{Jm}^{-3}\text{K}^{-1}] \tag{5.174}$$

$$\nu_G = 7.0 \times 10^{-7} \quad [\text{m}^2\text{s}^{-1}] \tag{5.175}$$

In (5.173),  $G_0$  which is appeared in the first ground layer ( $k = 1$ ) is heat flux which goes to the ground. It is sum of net radiation  $R_{net}$ , sensible heat  $H_S$  and latent heat  $lE$ .

$$G_0 = R_{net} - H_S - lE \tag{5.176}$$

They are given as follows, respectively.

$$R_{net} = RS^\downarrow + L^\downarrow - L^\uparrow \tag{5.177}$$

$$H_S = -C_p C_\theta (T_a - T_G) \tag{5.178}$$

$$lE = -\mathcal{L}_v C_{qv} \beta [q_{va} - q_{vs}(T_G)] \tag{5.179}$$

where the last term of net radiation  $R_{net}$  shows net upward longwave radiation  $T_1 = T_G$  and,

$$L^\uparrow = \varepsilon_G \sigma T_G^4 \tag{5.180}$$

$\beta$  in the equation of latent heat  ${}_lE$  is evapotranspiration coefficient which is supposed to take constant value during the integration period. Then, radiation process is used only for calculation of the heat balance at the ground surface.

Time integration of the ground temperature solved by implicit scheme. In order to disperse the time evolution equation of ground temperature (5.173),

about  $k = 1 \sim m - 1$ , the followings are defined.

$$a_1 = 0 \tag{5.181}$$

$$a_k = -\Delta t \frac{2\nu_G}{\Delta z_k(\Delta z_k + \Delta z_{k-1})} \tag{5.182}$$

$$b_k = -\Delta t \frac{2\nu_G}{\Delta z_k(\Delta z_{k+1} + \Delta z_k)} \tag{5.183}$$

Then, (5.173) is dispersed as follows.

$$\begin{aligned} T_1^n &= T_1^{n-1} + \frac{\Delta t G_0}{\rho_G C_G \Delta Z_1} + b_1(T_1^n - T_2^n) \\ T_2^n &= T_2^{n-1} + a_2(T_2^n - T_1^n) + b_2(T_2^n - T_3^n) \\ &\dots\dots\dots \\ T_k^n &= T_k^{n-1} + a_k(T_k^n - T_{k-1}^n) + b_k(T_k^n - T_{k+1}^n) \\ &\dots\dots\dots \\ T_{m-1}^n &= T_{m-1}^{n-1} + a_{m-1}(T_{m-1}^n - T_{m-2}^n) + b_{m-1}(T_{m-1}^n - T_m^n) \end{aligned} \tag{5.184}$$

The simultaneous equations can be expressed using a procession as follows.

$$\begin{pmatrix} T_1^n \\ T_2^n \\ \vdots \\ T_k^n \\ \vdots \\ T_{m-1}^n \end{pmatrix} = \begin{pmatrix} T_1^{n-1} + \frac{\Delta t G_0}{\rho_G C_G \Delta Z_1} \\ T_2^{n-1} \\ \vdots \\ T_k^{n-1} \\ \vdots \\ T_{m-1}^{n-1} - b_{m-1} T_m^n \end{pmatrix} + \begin{pmatrix} b_1 & -b_1 & 0 & \dots & \dots & \dots & 0 \\ -a_2 & a_2 + b_2 & -b_2 & 0 & \dots & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & -a_k & a_k + b_k & -b_k & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & \dots & \dots & \dots & -a_{m-1} & a_{m-1} + b_{m-1} \end{pmatrix} \begin{pmatrix} T_1^n \\ T_2^n \\ \vdots \\ T_k^n \\ \vdots \\ T_{m-1}^n \end{pmatrix} \tag{5.185}$$

Furthermore, it can be expressed briefly like

$$\mathbf{T}^n = \mathbf{F} + A\mathbf{T}^n \quad (5.186)$$

If the term of  $\mathbf{T}^n$  is shifted using unit procession  $I$  of  $(m - 1) \times (m - 1)$ ,

$$(I - A)\mathbf{T}^n = \mathbf{F} \quad (5.187)$$

is obtained. While it can be solved easily by the elimination of a gauss (refer to the 6.1.3 for explanation of a concrete solution method), ground temperature of all layers at the time of  $n$  are calculated.



## Chapter 6

# Numerical computation method

Generally, the equation of motion, the equation of potential temperature derived by the equation of thermodynamics, the equation of pressure derived by the continuity equation and the equation of gaseous state, and the equations of water vapor and hydrometeors are solved simultaneously and integrated with respect to time under appropriate boundary and initial values in cloud resolving models.

In *CReSS*, the equation of motion, the equations of the potential temperature perturbation and the pressure perturbation, and the equations of hydrometeors and water vapor which are shown in Chapter 2 'Formulation of the system of basic equations' are also temporally and spatially integrated by using finite difference method. There are many studies about the methods which are discretized and integrated these equation systems and the various calculation techniques are established.

In this chapter, the structure of a grid point, the method of discretization, the method of time integration and so on are explained about the numerical computation method used by *CReSS*.

## 6.1 The discretization of basic equation system

### 6.1.1 The outline of the numerical solution

There are various techniques from the selection of dependent variables to the method of time integration in the numerical solution by cloud resolving models. The outline of the numerical solution used in *CReSS* is as follows.

- The basic equation system is the equation of motion, potential temperature perturbation, pressure perturbation, water vapor and hydrometeors.
- The dependent variables are three-dimensional velocity components  $u, v, w$ , potential temperature perturbation  $\theta'$ , pressure perturbation  $p'$ , mixing ratio of water vapor, and mixing ratios and number concentrations of hydrometeors.
- The basic equations are defined by a terrain-following curvilinear coordinates are used to include the effect of orography.
- The finite differences are used for the derivation of the dependent variable with respect to space. The types of Arakawa-C and Lorenz staggered grids are used for horizontal and vertical grid arrangement, respectively.
- The sound wave is contained as a solution by using compressible equation system. The terms related to the sound wave are integrated with small time step  $\Delta\tau$ , the other terms are integrated with large time step  $\Delta t$ .
- The leap-frog scheme with the Asselin time filter is used for time integration with respect to the large time step.
- The explicit forward-backward scheme or the implicit Crank-Nicolson scheme in only vertical is used for time integration with respect to the small time step.
- The effects of sound wave are weakened by adding the divergence terms explicit scheme with the small time step.

Furthermore,  $\xi, \eta$  are expressed as  $x, y$  in this chapter as it is  $\xi = x, \eta = y$  in a terrain-following coordinates.

### 6.1.2 Grid in the model and setting of variables

As the space are represented by the method of grid points in *CReSS*, finite difference method is used for the spatial derivation. Here, the structure of grid points and the setting of variables in *CReSS* are shown.

The structure of grid points and the setting of variables are set on the staggered grid<sup>1</sup> as Fig6.1 in horizontal and vertical. Arakawa-C grid and Lorenz grid are used for the settings of horizontal and vertical grid points, respectively.

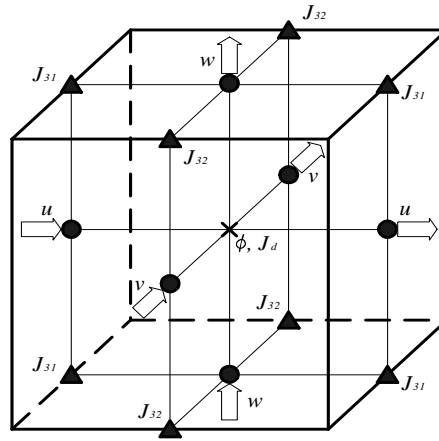
In this case, the point of the all scalars, such as pressure perturbations, potential temperature perturbation, mixing ratios of water vapor and physical quantities of hydrometeors, are located at the center of the grid box<sup>2</sup>.

The point of  $x$ -component velocity  $u$  is dislocated to a half interval from the center of the grid in the  $x$ -direction, the point of  $y$ -component velocity  $v$  is dislocated to a half interval from the center of the grid in the  $y$ -direction in horizontal. The point of  $u$  and  $v$  are defined at the same level as the scalars points. The point of  $z$ -component velocity  $w$  is vertically dislocated at a half interval just above and below the scalar variables points.

<sup>1</sup>There are various setting in the staggered grid which variables are defined alternately.

<sup>2</sup>The point of potential temperature (temperature) and the point of pressure are located at the same level in Lorenz grid. On the other hand, the point of potential temperature (temperature) is dislocated at a half interval below the pressure point, and defined at the same level as the vertical velocity point in the Charney Phillips' grid.





**Figure 6.1.** Structure of the staggered grid and setting of dependent variables.

Furthermore, as a terrain-following coordinate is used in *CReSS*, the Jacobian with transform for coordinates  $G^{\frac{1}{2}}$  and the metric tensor which is not zero or 1, that is,  $J_{31}, J_{32}, J_d (= G^{\frac{1}{2}})$  are shown as follows. The  $G^{\frac{1}{2}} (= J_d)$  point is defined at the same level as the scalars points. The  $J_{31}$  point is dislocated to a half interval from the center of the grid in the  $x$ -direction and the  $z$ -direction. The  $J_{32}$  is dislocated to a half interval from the center of the grid in the  $y$ -direction and the  $z$ -direction.

The coordinates  $x, y, \zeta$  which are set on the staggered grid as shown in Fig 6.1 are defined as the vectors points. The domain in a thick line except the most outside grid points is the physical domain as shown in Fig6.2. So four grid points in the direction without a dimension need to be calculated at the time of the 2-dimensional calculation.

### 6.1.3 Discretization by finite difference method

Generally, since elastic equations without any approximations are used in the cloud resolving model as the basic equations, all waves in the atmosphere are represented in the model. Waves in the atmosphere are divided into the sound wave, the gravity wave and the Rossby wave by phase velocity or frequency, and the phase velocities become small in this order. The phase velocity of sound wave is the fastest but we have no meteorological interest for it. On the other hand, the gravity wave and the Rossby wave are important for a cloud and a mesoscale meteorological phenomena<sup>3</sup> However, the sound wave is contained in models with elastic equation system, and the special technique is needed to be used for the calculation about the wave. So the time step is very small to satisfy CFL condition for the sound wave which the phase velocity is fastest if the time integration is calculated explicitly in meteorological models the general numerical integration for fluid. However, the time steps for the gravity and the Rossby waves which are meteorologically important are satisfied with the much larger time step than that of the sound wave, since the phase velocities of these waves are much slower than the sound wave.

There are various calculation techniques not to make the time step of the sound wave small<sup>4</sup>. The well-known calculation techniques are as follows.

**HI-VI method** The terms of the sound wave are solved implicitly, and the other terms are solved explicitly. This technique is called semi-implicit time integration method<sup>5</sup>.

<sup>3</sup>The Rossby wave is important for larger scale phenomena in mesoscale meteorology, synoptic meteorological phenomena and planetary meteorological phenomena. Weather is influenced by the Rossby-scale atmospheric phenomena every day, and usual forecast models represent these phenomena.

<sup>4</sup>If the anelastic equation system by Ogura and Phillips (1962) is used for the basic equation system in a numerical model, the time step can be defined by the phase velocity of gravity wave because the sound wave is not contained in this equation system. In this case, pressure is calculated by the equation of motion and the continuity equation. There are the advantage that the sound wave is removed, but the demerits of which the equation of pressure is complicated and a error is large at the place with large variations of density in this techniques.

<sup>5</sup>The terms with large time step are explicitly calculated and the other terms are implicitly solved in the semi-implicit

**HE-VE method** The terms of the sound wave and the other terms are solved explicitly using different time steps, respectively. This technique which is called the mode-splitting technique (Klemp and Wilhelmson, 1978) is often used.

**HE-VI method** This technique is almost the same as the mode-splitting technique. But the terms of the sound wave are solved implicitly in vertical.

The schematic representation of this mode-splitting technique is shown in Fig6.3.

Many data are exchanged between some nodes in the case of using parallel computer since large simultaneous equations are solved in the HE-VI method. The HE-VE method or HE-VI method are used in *CRESS* since it is assumed to be mainly used on a parallel computer. These techniques are easy to be applied to the large parallel computing since the physical space are corresponded to the setting of a node and only the data in halo region of a node communicates between nodes. The detail description is shown in 7 'Mounting of a parallel processing'. The efficiency of calculation does not become so bad to integrate the terms with respect to sound wave in small time step.

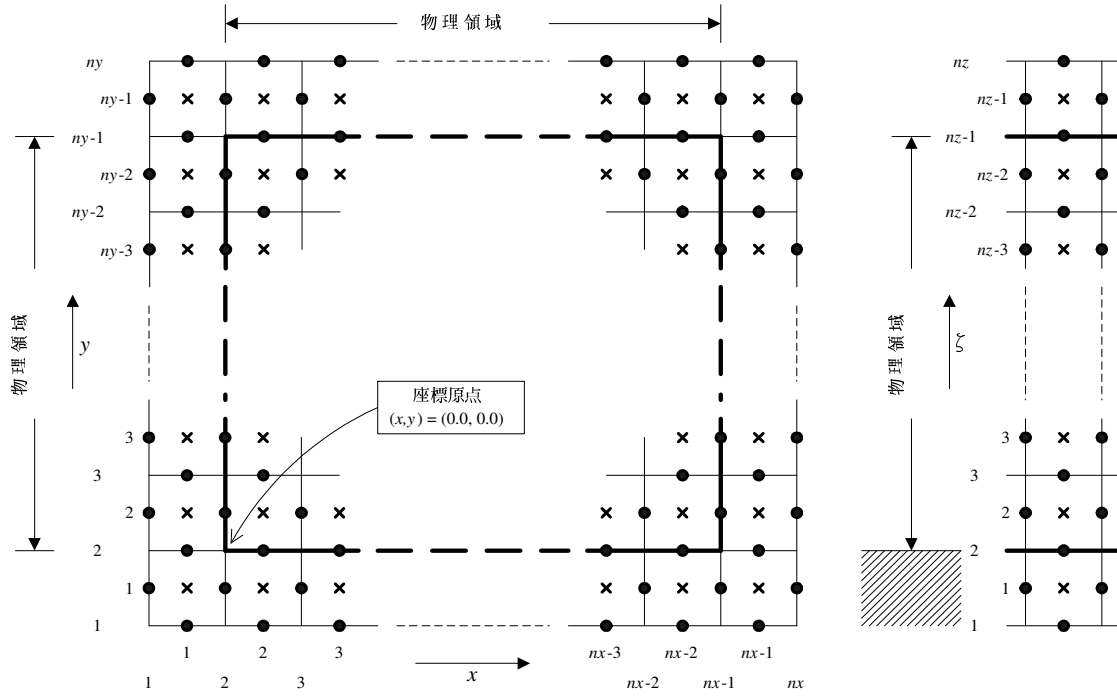


Figure 6.2. Setting of variables in a calculation domain (● represents a point for velocities. × represents a point for scalars).

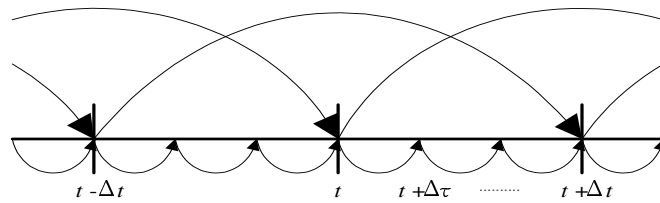


Figure 6.3. The schematic representation of the mode-splitting technique.

### The method of discretization for physical quantities with the sound wave mode

The method of discretization for physical quantities with the sound wave mode is shown from here. The gravity wave mode is calculated with the large time step. In this case, the discretizations for potential temperature, mixing ratios of water vapor, mixing ratios and number concentrations of hydrometeors is not took into consideration as they are calculated in the large time step.

Physical quantities with the sound wave mode are shown in the basic equation system in *CReSS*. In the following equations,  $[rm]$  indicates terms which are related to the Rossby wave mode (rotational mode),  $[gm]$  the gravity wave mode (divergence mode) and  $[am]$  the sound wave mode, respectively. Terms of physical processes are indicated by  $[physics]$ .

### The equation of motion

$$\begin{aligned} \frac{\partial u^*}{\partial t} = & - \underbrace{\left( u^* \frac{\partial u}{\partial x} + v^* \frac{\partial u}{\partial y} + W^* \frac{\partial u}{\partial \zeta} \right)}_{[rm]} \\ & - \underbrace{\left[ \frac{\partial}{\partial x} \{ J_d (p' - \alpha Div^*) \} + \frac{\partial}{\partial \zeta} \{ J_{31} (p' - \alpha Div^*) \} \right]}_{[am]} + \underbrace{(f_s v^* - f_c w^*)}_{[rm]} + \underbrace{G^{\frac{1}{2}} \text{Turb.} u}_{[physics]} \end{aligned} \quad (6.1)$$

$$\begin{aligned} \frac{\partial v^*}{\partial t} = & - \underbrace{\left( u^* \frac{\partial v}{\partial x} + v^* \frac{\partial v}{\partial y} + W^* \frac{\partial v}{\partial \zeta} \right)}_{[rm]} \\ & - \underbrace{\left[ \frac{\partial}{\partial y} \{ J_d (p' - \alpha Div^*) \} + \frac{\partial}{\partial \zeta} \{ J_{32} (p' - \alpha Div^*) \} \right]}_{[am]} - \underbrace{f_s u^*}_{[rm]} + \underbrace{G^{\frac{1}{2}} \text{Turb.} v}_{[physics]} \end{aligned} \quad (6.2)$$

$$\begin{aligned} \frac{\partial w^*}{\partial t} = & - \underbrace{\left( u^* \frac{\partial w}{\partial x} + v^* \frac{\partial w}{\partial y} + W^* \frac{\partial w}{\partial \zeta} \right)}_{[rm]} \\ & - \underbrace{\frac{\partial}{\partial \zeta} (p' - \alpha Div^*)}_{[am]} - \rho^* g \left( \underbrace{\frac{\theta'}{\theta}}_{[gm]} - \underbrace{\frac{p'}{\bar{\rho} c_s^2}}_{[am]} + \underbrace{\frac{q'_v}{\epsilon + \bar{q}_v} - \frac{q'_v + \sum q_x}{1 + \bar{q}_v}}_{[physics]} \right) + \underbrace{f_c u^*}_{[rm]} + \underbrace{G^{\frac{1}{2}} \text{Turb.} w}_{[physics]} \end{aligned} \quad (6.3)$$

### The equation of pressure

$$\begin{aligned} \frac{\partial G^{\frac{1}{2}} p'}{\partial t} = & - \underbrace{\left( G^{\frac{1}{2}} u \frac{\partial p'}{\partial x} + G^{\frac{1}{2}} v \frac{\partial p'}{\partial y} + G^{\frac{1}{2}} W \frac{\partial p'}{\partial \zeta} \right)}_{[rm]} + \underbrace{G^{\frac{1}{2}} \bar{\rho} g w}_{[am]} \\ & - \underbrace{\bar{\rho} c_s^2 \left( \frac{\partial G^{\frac{1}{2}} u}{\partial x} + \frac{\partial G^{\frac{1}{2}} v}{\partial y} + \frac{\partial G^{\frac{1}{2}} W}{\partial \zeta} \right)}_{[am]} + \underbrace{G^{\frac{1}{2}} \bar{\rho} c_s^2 \left( \frac{1}{\theta} \frac{d\theta}{dt} - \frac{1}{Q} \frac{dQ}{dt} \right)}_{[am]} \end{aligned} \quad (6.4)$$

These equation systems are made discrete by using finite difference method and integrated for the grid system as Fig 6.1. However, the equalization is required to calculate at the same point as using the

staggered grid. So the equalization operator is defined as follows. With respect to some physical quantities  $\phi$ , the equalization operator in the  $x$ -direction is defined as follows. Moreover, the  $i$ -th grid point number from the side is represented as the subindex.

$$\overline{\phi}_i^x = \frac{1}{2} \left( \phi_{i-\frac{1}{2}} + \phi_{i+\frac{1}{2}} \right) \quad (6.5)$$

Furthermore, the average of four grid points in the 4th order central difference is defined as follows.

$$\overline{\phi}_i^{2x} = \frac{1}{4} \left( \phi_{i-\frac{3}{2}} + \phi_{i-\frac{1}{2}} + \phi_{i+\frac{1}{2}} + \phi_{i+\frac{3}{2}} \right) \quad (6.6)$$

In the case with two indexes of equalization, the equalization operator for  $x, z$  is defined as follows (The indexes of grid point number are represented as  $i$  and  $k$  in the  $x$ -direction and the  $z$ -direction, respectively.).

$$\overline{\phi}_{ik}^{xz} = \frac{1}{4} \left( \phi_{i-\frac{1}{2}, k-\frac{1}{2}} + \phi_{i+\frac{1}{2}, k-\frac{1}{2}} + \phi_{i-\frac{1}{2}, k+\frac{1}{2}} + \phi_{i+\frac{1}{2}, k+\frac{1}{2}} \right) \quad (6.7)$$

Moreover, the differential operator is defined by representing the grid interval as  $\Delta x$  in  $x$ -direction as follows. The  $i$ -th grid point number from the side is also represented as the subindex.

$$(\partial_x \phi)_i = \frac{1}{\Delta x} \left( \phi_{i+\frac{\Delta x}{2}} - \phi_{i-\frac{\Delta x}{2}} \right) \quad (6.8)$$

$$(\partial_{2x} \phi)_i = \frac{1}{2\Delta x} \left( \phi_{i+\Delta x} - \phi_{i-\Delta x} \right) \quad (6.9)$$

Dependent variables in the numerical computing are defined by using these values as follows.

$$\rho^* = \overline{\rho G^{\frac{1}{2}}} \quad (6.10)$$

$$u^* = \overline{\rho G^{\frac{1}{2}} u} \quad (6.11)$$

$$v^* = \overline{\rho G^{\frac{1}{2}} v} \quad (6.12)$$

$$w^* = \overline{\rho G^{\frac{1}{2}} w} \quad (6.13)$$

$$W^* = \overline{\rho G^{\frac{1}{2}} W} \quad (6.14)$$

Here, the density  $\bar{\rho}$  and the Jacobian  $G^{\frac{1}{2}}$  for coordinate transformation are defined at the same point as the scalars. The vertical velocity  $W$  in a terrain-following coordinates and the vertical velocity  $w$  in the Cartesian coordinates are defined at the same point.

$$W = \frac{\left( \overline{u^\zeta J_{31}^x} + \overline{v^\zeta J_{32}^y} + w \right)}{\overline{G^{\frac{1}{2}}}} \quad (6.15)$$

In the case of explicit scheme, discretization of the equations of (6.1) ~ (??) are given with respect to  $u, v, w, p'$  as follows.

$$\overline{\rho}^{*x} \frac{u^{\tau+\Delta\tau} - u^\tau}{\Delta\tau} = - \left[ \partial_x (J_d P_\alpha) + \partial_\zeta \left( J_{31} \overline{P}_\alpha^{\zeta x} \right) \right]^\tau + F_u^t \quad (6.16)$$

$$\overline{\rho}^{*y} \frac{v^{\tau+\Delta\tau} - v^\tau}{\Delta\tau} = - \left[ \partial_y (J_d P_\alpha) + \partial_\zeta \left( J_{32} \overline{P}_\alpha^{\zeta y} \right) \right]^\tau + F_v^t \quad (6.17)$$

$$\overline{\rho}^{*\zeta} \frac{w^{\tau+\Delta\tau} - w^\tau}{\Delta\tau} = - [\partial_\zeta P_\alpha]^\tau - \left[ g G^{\frac{1}{2}} \frac{p'^\zeta}{c_s^2} \right]^\tau + F_w^t \quad (6.18)$$

$$\begin{aligned} G^{\frac{1}{2}\zeta} \frac{p'^{\tau+\Delta\tau} - p'^\tau}{\Delta\tau} &= -\overline{\rho} c_s^2 \left[ \partial_x \left( \overline{G}^{\frac{1}{2}x} u \right) + \partial_\zeta \left( \overline{J}_{31} \overline{w}^{\zeta x} \right) + \partial_y \left( \overline{G}^{\frac{1}{2}y} v \right) + \partial_\zeta \left( \overline{J}_{32} \overline{v}^{\zeta y} \right) + \partial_\zeta w \right]^{\tau+\Delta\tau} \\ &\quad + g \overline{\rho}^* \left[ \overline{w}^\zeta \right]^{\tau+\Delta\tau} + F_p^t \end{aligned} \quad (6.19)$$

where

$$P_\alpha = p' - \alpha \text{Div}^* \quad (6.20)$$

$$\text{Div}^* = \frac{1}{G^{\frac{1}{2}}} \left( \frac{\partial u^*}{\partial x} + \frac{\partial v^*}{\partial y} + \frac{\partial W^*}{\partial \zeta} \right) \quad (6.21)$$

where  $\alpha$  is a factor of damping for divergence terms to weaken the effects of the sound wave. The terms concerning the sound wave, which are integrated with a small time step  $\Delta\tau$ , are explicitly written, the other terms are included in  $F_\phi^t$  ( $\phi = u, v, w, p'$ ). The first term on the right side in the equations of (6.16) and (6.17), and the second term on the right side in the equation of (6.18) are pressure gradient force term. Buoyancy force with pressure perturbations and divergence damping are shown by the first and third term on the right side in the equation (6.18), respectively. The first term on the right side in the equation of (6.19) is a divergence term and the second term of that is a vertical advection term.

When terms for sound wave are solved explicitly,  $u, v$ , and  $w$  are calculated by the equations of (6.16) ~ (6.18) with forward-difference for  $\Delta\tau$ . Using the values of  $u, v$ , and  $w$ , pressure perturbation  $p'$  is calculated by the equation of (6.19) with backward-difference for  $\Delta\tau$ . During this integration with a small time step for  $2\Delta\tau$  ( $= n\Delta\tau$ ),  $F_\phi^t$  is constant. Terms of the sound wave in most of non-hydrostatic models are not solved explicitly in vertical and horizontal. Although some models (e.g. ARPS) include the HE-VE scheme as an option, the NHM of Meteorological Research Institute adopts the HI-VI and HE-VI schemes. The technique to solve implicitly in only vertical is described later. However, it is not necessarily that terms for sound wave are calculated implicitly in vertical when the horizontal resolution is the same as the vertical resolution as an object of **CRess**. There is an option to calculate the gravity mode in the small time step in **CRess**. This technique is also shown later. The gravity wave mode is integrated in the small time step in ARPS and in the large time step in NHM.

In the case of using vertical implicit scheme, the equations of vertical velocity  $w$  and pressure perturbation  $p'$  in the equations (6.18 and 6.19) are replaced by the following equations.

$$\begin{aligned} \overline{\rho^*}^\zeta \frac{w^{\tau+\Delta\tau} - w^\tau}{\Delta\tau} &= [\partial_\zeta(\alpha Div^*)]^\tau - \left[ \beta (\partial_\zeta p')^{\tau+\Delta\tau} + (1-\beta) (\partial_\zeta p')^\tau \right] \\ &\quad - \left[ \beta \left( \overline{gG^{\frac{1}{2}} \frac{p'}{c_s^2}} \right)^{\tau+\Delta\tau} + (1-\beta) \left( \overline{gG^{\frac{1}{2}} \frac{p'}{c_s^2}} \right)^\tau \right] + F_w^t \end{aligned} \quad (6.22)$$

$$\begin{aligned} \overline{G^{\frac{1}{2}}}^\zeta \frac{p'^{\tau+\Delta\tau} - p'^\tau}{\Delta\tau} &= -\bar{\rho}c_s^2 \left[ \partial_x \left( \overline{G^{\frac{1}{2}} u} \right) + \partial_\zeta \left( \overline{J_{31} \bar{u}^\zeta} \right) + \partial_y \left( \overline{G^{\frac{1}{2}} v} \right) + \partial_\zeta \left( \overline{J_{32} \bar{v}^\zeta} \right) \right]^{\tau+\Delta\tau} \\ &\quad - \bar{\rho}c_s^2 \left[ \beta (\partial_\zeta w)^{\tau+\Delta\tau} + (1-\beta) (\partial_\zeta w)^\tau \right] \\ &\quad + g\rho^* \left[ \beta (\bar{w}^\zeta)^{\tau+\Delta\tau} + (1-\beta) (\bar{w}^\zeta)^\tau \right] + F_p^t \end{aligned} \quad (6.23)$$

These equations are gained by averaging for  $\tau$  and  $\tau + \Delta\tau$  of the equations (6.18) and (6.19) weighted by  $\beta$ . The equations of  $w$  (6.22) with  $\beta = 0$  and  $p'$  (6.23) with  $\beta = 1$  result in the equations of (6.18) and (6.19), respectively. When  $\beta = 0$ , (6.22) and (6.23) are the simultaneous equations for  $w^{\tau+\Delta\tau}$  and  $p'^{\tau+\Delta\tau}$ . They are implicitly calculated in vertical with the Crank-Nicolson scheme.

In the vertical implicit scheme, the forward difference for the equations of (6.16) and (6.17) are calculated and the values of  $u$  and  $v$  at  $\tau + \Delta\tau$  are gained. In the vertical implicit scheme, the forward difference for the equations of (6.16) and (6.17) are calculated and the values of  $u$  and  $v$  at  $\tau + \Delta\tau$  are gained. Concerning  $p'^{\tau+\Delta\tau}$  from (6.22) and (6.23), we get the following equations for  $w$  and  $p$ .

$$\begin{aligned} \overline{\rho^*}^\zeta \frac{w^{\tau+\Delta\tau} - w^\tau}{\Delta\tau} &= -\Delta\tau\beta^2\partial_\zeta \left( \frac{g\rho^*}{G^{\frac{1}{2}}}\bar{w}^\zeta - \frac{\bar{\rho}c_s^2}{G^{\frac{1}{2}}}\partial_\zeta w \right)^{\tau+\Delta\tau} \\ &\quad - \Delta\tau\beta^2g \left( \frac{g\rho^*}{c_s^2}\bar{w}^\zeta - \bar{\rho}\partial_\zeta w^\zeta \right)^{\tau+\Delta\tau} + F_w' \end{aligned} \quad (6.24)$$

where  $F_w'$  and  $F_p'$  are

$$\begin{aligned} F_p' &= \frac{\Delta\tau}{G^{\frac{1}{2}}} \left[ F_p^t + (1-\beta) (g\rho^*\bar{w}^\zeta - \bar{\rho}c_s^2\partial_\zeta w)^\tau \right] \\ &\quad - \frac{\Delta\tau}{G^{\frac{1}{2}}}\bar{\rho}c_s^2 \left[ \partial_x \left( \overline{G^{\frac{1}{2}} u} \right) + \partial_\zeta \left( \overline{J_{31} \bar{u}^\zeta} \right) + \partial_y \left( \overline{G^{\frac{1}{2}} v} \right) + \partial_\zeta \left( \overline{J_{32} \bar{v}^\zeta} \right) \right]^{\tau+\Delta\tau} \end{aligned} \quad (6.25)$$

$$F_w' = -[\partial_\zeta P_\alpha]^\tau - \left[ \overline{gG^{\frac{1}{2}} \frac{p'}{c_s^2}} \right]^\tau - \beta \left[ \frac{\overline{gG^{\frac{1}{2}} F_p'}}{c_s^2} + \partial_\zeta F_p' \right] + F_w^t. \quad (6.26)$$

The second-order discretization in vertical (6.24) is ordered as follows for the unknown  $w^{\tau+\Delta\tau}$ .

$$A_k w_{k-1}^{\tau+\Delta\tau} + B_k w_k^{\tau+\Delta\tau} + C_k w_{k+1}^{\tau+\Delta\tau} = F_k \quad (6.27)$$

Simultaneous equations whose factors are a triple diagonal matrix are obtained as this equation. This equation is solved simply if the top and bottom boundary conditions are given as follows, for example.

$$w = 0 \quad k = nk - 1 \quad (6.28)$$

$$w = u \frac{\partial z_s}{\partial x} + v \frac{\partial z_s}{\partial y} \quad k = 2 \quad (6.29)$$

where  $A_k, B_k, C_k, F_k$  are known quantities at  $\tau$ . These quantities are given under the boundary conditions of (6.28) and (6.29) as follows.

$$A_k = \begin{cases} 0 & k = 3 \\ \left( -Q_k c_{s_{k-1}}^2 + P_k G_{k-1}^{\frac{1}{2}} \right) (\mathcal{R}_{k-1} + \mathcal{S}_{k-1}), & 4 \leq k \leq nk - 2 \end{cases} \quad (6.30)$$

$$B_k = 1 + Q_k \{ (\mathcal{R}_k + \mathcal{S}_k) c_{s_k}^2 - (\mathcal{R}_{k-1} - \mathcal{S}_{k-1}) c_{s_{k-1}}^2 \} \\ + P_k \{ (\mathcal{R}_k + \mathcal{S}_k) G_k^{\frac{1}{2}} - (\mathcal{R}_{k-1} - \mathcal{S}_{k-1}) G_{k-1}^{\frac{1}{2}} \}, \quad 3 \leq k \leq nk - 2 \quad (6.31)$$

$$C_k = \begin{cases} \left( Q_k c_{s_k}^2 + P_k G_k^{\frac{1}{2}} \right) (\mathcal{R}_k + \mathcal{S}_k), & 3 \leq k \leq nk - 3 \\ 0, & k = nk - 2 \end{cases} \quad (6.32)$$

$$F_k = \begin{cases} F'_{w_k} + w_k^\tau + \bar{u}_k^\zeta (\partial_x z_s)_k + \bar{v}_k^\zeta (\partial_y z_s)_k, & k = 3 \\ F'_{w_k} + w_k^\tau, & 4 \leq k \leq nk - 2 \end{cases} \quad (6.33)$$

$$P_k = \frac{\Delta \tau^2 \beta^2 g}{2 \rho_k^* \zeta}, \quad Q_k = \frac{\Delta \tau^2 \beta^2}{\Delta \zeta_k \rho_k^* \zeta}, \quad \mathcal{R}_k = \frac{g \rho_k^*}{2 G_k^{\frac{1}{2}} c_{s_k}^2}, \quad \mathcal{S}_k = \frac{\rho_k^*}{\Delta \zeta_k G_k} \quad (6.34)$$



This simultaneous equation (6.27) is represented by a matrix by replacing  $w_k^{\tau+\Delta\tau}$  with  $w_k$  as follows.

$$\begin{pmatrix} B_3 & C_3 & 0 & \cdots & \cdots & 0 \\ A_4 & B_4 & C_4 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & A_k & B_k & C_k & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & \cdots & A_{nk-2} & B_{nk-2} \end{pmatrix} \begin{pmatrix} w_3 \\ w_4 \\ \vdots \\ w_k \\ \vdots \\ w_{nk-2} \end{pmatrix} = \begin{pmatrix} F_3 \\ F_4 \\ \vdots \\ F_k \\ \vdots \\ F_{nk-2} \end{pmatrix} \quad (6.35)$$

The forward elimination and the backward substitution are carried out to solve this matrix by the Gaussian Elimination. The first line in (6.35) are divided by  $B_3$  to carry out the forward elimination.

$$\begin{pmatrix} 1 & c_3 & 0 & \cdots & \cdots & 0 \\ A_4 & B_4 & C_4 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & A_k & B_k & C_k & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & \cdots & A_{nk-2} & B_{nk-2} \end{pmatrix} \begin{pmatrix} w_3 \\ w_4 \\ \vdots \\ w_k \\ \vdots \\ w_{nk-2} \end{pmatrix} = \begin{pmatrix} f_3 \\ F_4 \\ \vdots \\ F_k \\ \vdots \\ F_{nk-2} \end{pmatrix} \quad (6.36)$$

where  $c_3 = C_3/B_3$  and  $f_3 = F_3/B_3$ . The diagonal components of the values which subtracted from the second line to the values which multiplied the first line by  $A_4$  are standardized as follows.

$$\begin{pmatrix} 1 & c_3 & 0 & \cdots & \cdots & 0 \\ 0 & 1 & c_4 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & A_k & B_k & C_k & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & \cdots & A_{nk-2} & B_{nk-2} \end{pmatrix} \begin{pmatrix} w_3 \\ w_4 \\ \vdots \\ w_k \\ \vdots \\ w_{nk-2} \end{pmatrix} = \begin{pmatrix} f_3 \\ f_4 \\ \vdots \\ F_k \\ \vdots \\ F_{nk-2} \end{pmatrix} \quad (6.37)$$

where  $c_4 = C_4/(B_4 - A_4c_3)$  and  $f_4 = (F_4 - A_4f_3)/(B_4 - A_4c_3)$ . Similarly, using  $c_k = C_k/(B_k - A_kc_{k-1})$  and  $f_k = (F_k - A_kf_{k-1})/(B_k - A_kc_{k-1})$ ,

$$\begin{pmatrix} 1 & c_3 & 0 & \cdots & \cdots & 0 \\ 0 & 1 & c_4 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & 1 & c_k & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & \cdots & 0 & 1 \end{pmatrix} \begin{pmatrix} w_3 \\ w_4 \\ \vdots \\ w_k \\ \vdots \\ w_{nk-2} \end{pmatrix} = \begin{pmatrix} f_3 \\ f_4 \\ \vdots \\ f_k \\ \vdots \\ f_{nk-2} \end{pmatrix} \quad (6.38)$$

Finally, solutions for all  $k$  are obtained to carry out the backward substitution in order of  $k = nk - 2, nk - 3$  and  $\cdots, 4$ .

$$\begin{aligned}
w_{nk-2} &= f_{nk-2} \\
w_{nk-3} &= f_{nk-3} - c_{nk-3}w_{nk-2} \\
&\vdots \\
w_k &= f_k - c_k w_{k+1} \\
&\vdots \\
w_4 &= f_4 - c_4 w_5 \\
w_3 &= f_3 - c_3 w_4
\end{aligned} \tag{6.39}$$

The term ( $F_\phi^t$ ) which is calculated with the large time step is represented as follows.

$$F_u^t = -\text{Adv}.u^t + \left[ \overline{\rho^* f_s \bar{v}^y}^x - \overline{\rho^* f_c \bar{w}^\zeta}^x \right]^t + \left[ G^{\frac{1}{2}} \text{Turb}.u \right]^{t-\Delta t} \tag{6.40}$$

$$F_v^t = -\text{Adv}.v^t - \left[ \overline{\rho^* f_s \bar{u}^x}^y \right]^t + \left[ G^{\frac{1}{2}} \text{Turb}.v \right]^{t-\Delta t} \tag{6.41}$$

$$F_w^t = -\text{Adv}.w^t + \left[ \overline{\rho^* (\text{Buoy}.\theta + \text{Buoy}.q)}^\zeta + \overline{\rho^* f_c \bar{u}^x}^\zeta \right]^t + \left[ G^{\frac{1}{2}} \text{Turb}.w \right]^{t-\Delta t} \tag{6.42}$$

$$F_p^t = -\text{Adv}.p^t \tag{6.43}$$

where Buoy. $\theta$  and Buoy. $q$  are buoyancy terms for temperature perturbations, water vapor and hydrometeors, and are defined as follows.

$$\text{Buoy}.\theta \equiv g \frac{\theta'}{\bar{\theta}} \tag{6.44}$$

$$\text{Buoy}.q \equiv g \left( \frac{q'_v}{\epsilon + \bar{q}_v} - \frac{q'_v + \sum q_x}{1 + \bar{q}_v} \right) \tag{6.45}$$

The advection term is calculated with the second or fourth-order accuracy by using the values at the time ( $t$ ). The second-order advection term is discretized as follows.

$$\text{Adv}.u^t = \overline{u^{*x}}^x \partial_x u + \overline{v^{*x}}^y \partial_y u + \overline{W^{*x}}^\zeta \partial_\zeta u \tag{6.46}$$

$$\text{Adv}.v^t = \overline{u^{*y}}^x \partial_x v + \overline{v^{*y}}^y \partial_y v + \overline{W^{*y}}^\zeta \partial_\zeta v \tag{6.47}$$

$$\text{Adv}.w^t = \overline{u^{*\zeta}}^x \partial_x w + \overline{v^{*\zeta}}^y \partial_y w + \overline{W^{*\zeta}}^\zeta \partial_\zeta w \tag{6.48}$$

$$\text{Adv}.p^t = G^{\frac{1}{2}} \overline{u^{*x}}^x \partial_x p' + G^{\frac{1}{2}} \overline{v^{*y}}^y \partial_y p' + G^{\frac{1}{2}} \overline{W^{*y}}^\zeta \partial_\zeta p' \tag{6.49}$$

The fourth-order advection term is discretized as follows.

$$\begin{aligned}
\text{Adv}.u^t &= \frac{4}{3} \left[ \overline{u^{*x}}^x \partial_x u + \overline{v^{*x}}^y \partial_y u + \overline{W^{*x}}^\zeta \partial_\zeta u \right] \\
&\quad - \frac{1}{3} \left[ \overline{u^{*2x}}^{2x} \partial_{2x} u + \overline{v^{*xy}}^{2y} \partial_{2y} u + \overline{W^{*x\zeta}}^{2\zeta} \partial_{2\zeta} u \right]
\end{aligned} \tag{6.50}$$

$$\begin{aligned} \text{Adv}.v^t &= \frac{4}{3} \left[ \overline{u^{*y}} \partial_x v + \overline{v^{*y}} \partial_y v + \overline{W^{*y}} \partial_\zeta v \right] \\ &\quad - \frac{1}{3} \left[ \overline{u^{*xy}} \partial_{2x} v + \overline{v^{*2y}} \partial_{2y} v + \overline{W^{*y\zeta}} \partial_{2\zeta} v \right] \end{aligned} \quad (6.51)$$

$$\begin{aligned} \text{Adv}.w^t &= \frac{4}{3} \left[ \overline{u^{*\zeta}} \partial_x w + \overline{v^{*\zeta}} \partial_y w + \overline{W^{*\zeta}} \partial_\zeta w \right] \\ &\quad - \frac{1}{3} \left[ \overline{u^{*x\zeta}} \partial_{2x} w + \overline{v^{*y\zeta}} \partial_{2y} w + \overline{W^{*\zeta}} \partial_{2\zeta} w \right] \end{aligned} \quad (6.52)$$

$$\begin{aligned} \text{Adv}.p^t &= \frac{4}{3} \left[ \overline{G^{\frac{1}{2}x}} u \partial_x p' + \overline{G^{\frac{1}{2}y}} v \partial_y p' + \overline{G^{\frac{1}{2}\zeta}} W \partial_\zeta p' \right] \\ &\quad - \frac{1}{3} \left[ \overline{G^{\frac{1}{2}x}} u \partial_{2x} p' + \overline{G^{\frac{1}{2}y}} v \partial_{2y} p' + \overline{G^{\frac{1}{2}\zeta}} W \partial_{2\zeta} p' \right] \end{aligned} \quad (6.53)$$

The computing instability is caused in the calculation with discretization of central difference. So an artificial numerical viscosity term which is shown in the following section 6.2 is added to the above terms.

Variables except for pressure in a turbulent diffusion term are applied in the turbulent diffusion terms. The tensor for deformation velocity which are indicated in (??) ~ (??) are solved as follows by using calculus of finite differences.

$$S_{11} = \frac{2}{G^{\frac{1}{2}}} \left[ \partial_x \left( \overline{J_d^x} u \right) + \partial_\zeta \left( \overline{J_{31} \bar{w}^\zeta} \right) \right] \quad (6.54)$$

$$S_{22} = \frac{2}{G^{\frac{1}{2}}} \left[ \partial_y \left( \overline{J_d^y} v \right) + \partial_\zeta \left( \overline{J_{32} \bar{v}^\zeta} \right) \right] \quad (6.55)$$

$$S_{33} = \frac{2}{G^{\frac{1}{2}}} \partial_\zeta w \quad (6.56)$$

$$S_{12} = \frac{1}{G^{\frac{1}{2}xy}} \left[ \partial_y \left( \overline{J_d^x} u \right) + \partial_x \left( \overline{J_d^y} v \right) + \partial_\zeta \left( \overline{J_{32} \bar{u}^\zeta} + \overline{J_{31} \bar{v}^\zeta} \right) \right] \quad (6.57)$$

$$S_{13} = \frac{1}{G^{\frac{1}{2}x\zeta}} \left[ \partial_x \left( \overline{J_d^\zeta} w \right) + \partial_\zeta \left( u + \overline{J_{31} \bar{w}^\zeta} \right) \right] \quad (6.58)$$

$$S_{23} = \frac{1}{G^{\frac{1}{2}y\zeta}} \left[ \partial_y \left( \overline{J_d^\zeta} w \right) + \partial_\zeta \left( v + \overline{J_{32} \bar{w}^\zeta} \right) \right] \quad (6.59)$$

The divergence term  $Div$  which is indicated in 3.24 are solved as follows by using calculus of finite differences.

$$Div = \frac{1}{G^{\frac{1}{2}}} \left[ \partial_x \left( \overline{G^{\frac{1}{2}x}} u \right) + \partial_y \left( \overline{G^{\frac{1}{2}y}} v \right) + \partial_\zeta \left( \overline{G^{\frac{1}{2}\zeta}} W \right) \right] \quad (6.60)$$

Using these, the stress tensor which are indicated in (??) ~ (??) are written by finite differences as follows.

$$\tau_{11} = \bar{\rho} \nu_{\tau h} \left( S_{11} - \frac{2}{3} Div \right) \quad (6.61)$$

$$\tau_{12} = \overline{\bar{\rho} \nu_{\tau h}^{xy}} S_{12} \quad (6.62)$$

$$\tau_{13} = \overline{\bar{\rho} \nu_{\tau v}^{x\zeta}} S_{13} \quad (6.63)$$

$$\tau_{21} = \overline{\bar{\rho} \nu_{\tau h}^{yx}} S_{21} \quad (6.64)$$

$$\tau_{22} = \bar{\rho} \nu_{\tau h} \left( S_{22} - \frac{2}{3} Div \right) \quad (6.65)$$

$$\tau_{23} = \overline{\bar{\rho} \nu_{\tau v}^{y\zeta}} S_{23} \quad (6.66)$$

$$\tau_{31} = \overline{\bar{\rho} \nu_{\tau h}^{\zeta x}} S_{31} \quad (6.67)$$

$$\tau_{32} = \overline{\bar{\rho} \nu_{\tau h}^{\zeta y}} S_{32} \quad (6.68)$$

$$\tau_{33} = \bar{\rho} \nu_{\tau v} \left( S_{33} - \frac{2}{3} Div \right) \quad (6.69)$$

Finally, the turbulent diffusion terms are indicated in differences as follows.

$$G^{\frac{1}{2}} \text{Turb.} u = \partial_x (J_d \tau_{11}) + \partial_y \left( \overline{J_d^{xy}} \tau_{12} \right) + \partial_\zeta \left( \tau_{13} + J_{31} \overline{\tau_{11}^{x\zeta}} + \overline{J_{32}^x \tau_{12}^{\zeta y}} \right) \quad (6.70)$$

$$G^{\frac{1}{2}} \text{Turb.} v = \partial_x \left( \overline{J_d^{yx}} \tau_{21} \right) + \partial_y (J_d \tau_{22}) + \partial_\zeta \left( \tau_{23} + \overline{J_{31}^y \tau_{21}^{\zeta x}} + J_{32} \overline{\tau_{22}^{y\zeta}} \right) \quad (6.71)$$

$$G^{\frac{1}{2}} \text{Turb.} w = \partial_x \left( \overline{J_d^{\zeta x}} \tau_{31} \right) + \partial_y \left( \overline{J_d^{\zeta y}} \tau_{32} \right) + \partial_\zeta \left( \tau_{33} + \overline{J_{31}^{\zeta} \tau_{31}^{\zeta x}} + \overline{J_{32}^{\zeta} \tau_{32}^{\zeta y}} \right) \quad (6.72)$$

### The method of discretization for physical quantities without the sound wave mode

When the gravity wave mode is calculated with the large time step, the equations of potential temperature, water vapor and hydrometeors are solved with only large time step. These equations are shown by indicating water vapor and hydrometeors as  $x$  as follows.

$$\rho^* \frac{\theta^{t+\Delta t} - \theta^{t-\Delta t}}{2\Delta t} = F_\theta^t \quad (6.73)$$

$$\rho^* \frac{q_x^{t+\Delta t} - q_x^{t-\Delta t}}{2\Delta t} = F_q^t \quad (6.74)$$

$$G^{\frac{1}{2}} \frac{N_x^{t+\Delta t} - N_x^{t-\Delta t}}{2\Delta t} = F_N^t \quad (6.75)$$

where

$$F_\theta^t = -\text{Adv.}\theta^t + \left[ G^{\frac{1}{2}} \text{Turb.}\theta \right]^{t-\Delta t} + [\rho^* \text{Src.}\theta]^t - \left[ \overline{\rho^\zeta w \partial_\zeta \theta}^\zeta \right]^t \quad (6.76)$$

$$F_q^t = -\text{Adv.}q_x^t + \left[ G^{\frac{1}{2}} \text{Turb.}q_x \right]^{t-\Delta t} + [\rho^* \text{Src.}q_x]^t + [\rho^* \text{Fall.}q_x]^t \quad (6.77)$$

$$F_N^t = -\text{Adv.}\frac{N_x}{\rho^*}^t + \left[ G^{\frac{1}{2}} \text{Turb.}\frac{N_x}{\rho^*} \right]^{t-\Delta t} + \left[ \rho^* \text{Src.}\frac{N_x}{\rho^*} \right]^t + \left[ \rho^* \text{Fall.}\frac{N_x}{\rho^*} \right]^t. \quad (6.78)$$

The fourth term on the right side of the equation of potential temperature perturbation (6.76) is The term of gravity wave with vertical advection of potential temperature in the basic state.

The advection term is calculated as physical quantities with the sound wave mode by using the values at the time  $t$ . The advection term is solved by using the second-order scheme as follows.

$$\text{Adv.}\theta^t = \overline{u^* \partial_x \theta^x} + \overline{v^* \partial_y \theta^y} + \overline{W^* \partial_\zeta \theta^\zeta} \quad (6.79)$$

$$\text{Adv.}q_x^t = \overline{u^* \partial_x q_x^x} + \overline{v^* \partial_y q_x^y} + \overline{W^* \partial_\zeta q_x^\zeta} \quad (6.80)$$

$$\text{Adv.}\frac{N_x}{\rho^*}^t = \overline{u^* \partial_x \frac{N_x}{\rho^*}^x} + \overline{v^* \partial_y \frac{N_x}{\rho^*}^y} + \overline{W^* \partial_\zeta \frac{N_x}{\rho^*}^\zeta} \quad (6.81)$$

The advection terms are solved by using the fourth-order scheme as follows.

$$\begin{aligned} \text{Adv.}\theta^t &= \frac{4}{3} \left[ \overline{u^* \partial_x \theta^x} + \overline{v^* \partial_y \theta^y} + \overline{W^* \partial_\zeta \theta^\zeta} \right] \\ &\quad - \frac{1}{3} \left[ \overline{u^{*x} \partial_{2x} \theta'^{2x}} + \overline{v^{*y} \partial_{2y} \theta'^{2y}} + \overline{W^{*\zeta} \partial_{2\zeta} \theta'^{2\zeta}} \right] \end{aligned} \quad (6.82)$$

$$\begin{aligned} \text{Adv.}q_x^t &= \frac{4}{3} \left[ \overline{u^* \partial_x q_x^x} + \overline{v^* \partial_y q_x^y} + \overline{W^* \partial_\zeta q_x^\zeta} \right] \\ &\quad - \frac{1}{3} \left[ \overline{u^{*x} \partial_{2x} q_x^{2x}} + \overline{v^{*y} \partial_{2y} q_x^{2y}} + \overline{W^{*\zeta} \partial_{2\zeta} q_x^{2\zeta}} \right] \end{aligned} \quad (6.83)$$

$$\text{Adv.}\frac{N_x}{\rho^*}^t = \frac{4}{3} \left[ \overline{u^* \partial_x \frac{N_x}{\rho^*}^x} + \overline{v^* \partial_y \frac{N_x}{\rho^*}^y} + \overline{W^* \partial_\zeta \frac{N_x}{\rho^*}^\zeta} \right]$$

$$-\frac{1}{3} \left[ \overline{\frac{N_x}{\rho^*}} \frac{N_x}{\rho^*} \partial_{2x}^2 + \overline{v^{*y}} \partial_{2y} \frac{N_x}{\rho^*} + \overline{W^{*\zeta}} \partial_{2\zeta} \frac{N_x}{\rho^*} \right] \quad (6.84)$$

The turbulent flux  $H_{\phi 1}$ ,  $H_{\phi 2}$ , and  $H_{\phi 3}$  for one scalar  $\phi$  which is represented by (??)  $\sim$  (??) are solved by using difference as follows.

$$H_{\phi 1} = \left( \overline{\bar{\rho} \nu_{Hh} \frac{1}{G^{\frac{1}{2}}}} \right)^x \left[ \partial_x (J_d \phi) + \partial_\zeta (J_{31} \bar{\phi}^{x\zeta}) \right] \quad (6.85)$$

$$H_{\phi 2} = \left( \overline{\bar{\rho} \nu_{Hh} \frac{1}{G^{\frac{1}{2}}}} \right)^y \left[ \partial_y (J_d \phi) + \partial_\zeta (J_{32} \bar{\phi}^{y\zeta}) \right] \quad (6.86)$$

$$H_{\phi 3} = \left( \overline{\bar{\rho} \nu_{Hv} \frac{1}{G^{\frac{1}{2}}}} \right)^\zeta \partial_\zeta \bar{\phi}^\zeta \quad (6.87)$$

Finally, the turbulent diffusion term is indicated as follows.

$$G^{\frac{1}{2}} \text{Turb.} \theta = \partial_x \left( \overline{J_d^x} H_{\theta 1} \right) + \partial_y \left( \overline{J_d^y} H_{\theta 2} \right) + \partial_\zeta \left( H_{\theta 3} + \overline{J_{31} \bar{H}_{\theta 1}^\zeta} + \overline{J_{32} \bar{H}_{\theta 2}^\zeta} \right) \quad (6.88)$$

$$G^{\frac{1}{2}} \text{Turb.} q_x = \partial_x \left( \overline{J_d^x} H_{q_x 1} \right) + \partial_y \left( \overline{J_d^y} H_{q_x 2} \right) + \partial_\zeta \left( H_{q_x 3} + \overline{J_{31} \bar{H}_{q_x 1}^\zeta} + \overline{J_{32} \bar{H}_{q_x 2}^\zeta} \right) \quad (6.89)$$

$$G^{\frac{1}{2}} \text{Turb.} \frac{N_x}{\rho^*} = \partial_x \left( \overline{J_d^x} H_{N_x 1} \right) + \partial_y \left( \overline{J_d^y} H_{N_x 2} \right) + \partial_\zeta \left( H_{N_x 3} + \overline{J_{31} \bar{H}_{N_x 1}^\zeta} + \overline{J_{32} \bar{H}_{N_x 2}^\zeta} \right) \quad (6.90)$$

### The method of discretization in the case of calculating the gravity wave mode with small time step

When the gravity wave mode is calculated with small time step, the equation of potential temperature is indicated as follows.

$$\frac{\partial \theta^*}{\partial t} = - \underbrace{\left( u^* \frac{\partial \theta'}{\partial x} + v^* \frac{\partial \theta'}{\partial y} + W^{*\zeta} \frac{\partial \theta'}{\partial \zeta} \right)}_{[rm]} - \underbrace{\bar{\rho} w \frac{\partial \bar{\theta}}{\partial \zeta}}_{[gm]} + \underbrace{G^{\frac{1}{2}} \text{Turb.} \theta}_{[physics]} + \underbrace{\rho^* \text{Src.} \theta}_{[physics]} \quad (6.91)$$

The second term on the right side of is calculated by using the values at the time step  $\Delta \tau$ . The time integration of this equation is shown as follows.

$$\rho^* \frac{\theta'^{\tau+\Delta\tau} - \theta'^\tau}{\Delta\tau} = - \left[ \overline{\bar{\rho}^\zeta w \partial_\zeta \bar{\theta}^\zeta} \right]^\tau + F_\theta^t \quad (6.92)$$

The time integration of vertical velocity ( $w$ ) is shown since the buoyancy terms are calculated by using the values at the time( $\Delta\tau$ ).

$$\begin{aligned} \overline{\rho^*}^\zeta \frac{w^{\tau+\Delta\tau} - w^\tau}{\Delta\tau} &= [\partial_\zeta(\alpha Div^*)]^\tau + g \left[ \overline{\rho^* \frac{\theta'^\zeta}{\theta}} \right]^\tau - \left[ \beta (\partial_\zeta p')^{\tau+\Delta\tau} + (1-\beta) (\partial_\zeta p')^\tau \right] \\ &\quad - \left[ \beta \left( \overline{g G^{\frac{1}{2}} \frac{p'}{c_s^2}} \right)^{\tau+\Delta\tau} + (1-\beta) \left( \overline{g G^{\frac{1}{2}} \frac{p'}{c_s^2}} \right)^\tau \right] + F_w^t \end{aligned} \quad (6.93)$$

Terms which are calculated with large time step are modified as follows by this.

$$F_w^t = -Adv.w^t + \left[ \overline{\rho^* Buoy.q}^\zeta + \overline{\rho^* f_c \bar{u}^x \zeta} \right]^t + \left[ G^{\frac{1}{2}} Turb.w \right]^{t-\Delta t} \quad (6.94)$$

$$F_\theta^t = -Adv.\theta^t + \left[ G^{\frac{1}{2}} Turb.\theta \right]^{t-\Delta t} + [\rho^* Src.\theta]^t \quad (6.95)$$

### time filter

The large time step integration is solved by using the leap-frog scheme. The separation of solutions between the odd time step and the even time step is caused in the case of using this technique. So the Asselin time filter (Asselin, 1972) is used to control the separation of solutions. This is the technique to filter physical quantities at the time  $t$  after physical quantities are calculated at the time  $t + \Delta t$ .

The Asselin time filter is indicated by using a representative variable  $\psi$  for forecasted variables  $u, v, w, p', \theta', q_x$ .

$$\bar{\psi}^t = \psi^t + \mu_a \left( \overline{\psi^{t-\Delta t}} - 2\psi^t + \psi^{t+\delta t} \right) \quad (6.96)$$

where  $\mu_a$  is a factor of filter, and  $\mu_a = 0.1$  is a standard value.

## 6.2 Artificial numerical viscosity term

### 6.2.1 Necessity of numerical viscosity term

In previous section, it is shown that advection term is expressed by second-order or fourth-order central difference, and calculation instability will be generated if the artificial numerical viscosity term is not added. The reason for that is shown as follows.

One-dimensional linear wave equation :

$$\frac{\partial u}{\partial t} + c \frac{\partial u}{\partial x} = 0, \quad (c > 0) \quad (6.97)$$

The equation is discretized as follows. Arrangement of shifted grid points is not considered here.

$$\frac{\partial u}{\partial t} = \frac{u_i^{t+\Delta t} - u_i^t}{\Delta t}, \quad \frac{\partial u}{\partial x} = \frac{u_{i+\Delta x}^t - u_{i-\Delta x}^t}{2\Delta x} \quad (6.98)$$

The equation (6.97) is written by

$$u_i^{t+\Delta t} = u_i^t - \frac{c}{2} \left( \frac{\Delta t}{\Delta x} \right) (u_{i+\Delta x}^t - u_{i-\Delta x}^t) \quad (6.99)$$

From this equation, the value at time  $t + \Delta t$  is calculated by using the value at time  $t$ . However, when this calculation is performed, this calculation will break even if the relation of  $\Delta t$  and  $\Delta x$  satisfies the CFL condition.

As the other discrete method, we consider first-order windward difference because its calculation can be stable. However, it does not mean higher accuracy for the calculation. If this discretization is applied to an equation (6.97) and solved as follows.

$$u_i^{t+\Delta t} = u_i^t - c \left( \frac{\Delta t}{\Delta x} \right) (u_i^t - u_{i-\Delta x}^t) \quad (6.100)$$

However, if this is transformed into the form using the equation (6.99) which is the first discretization,

$$u_i^{t+\Delta t} = u_i^t - \frac{c}{2} \left( \frac{\Delta t}{\Delta x} \right) (u_{i+\Delta x}^t - u_{i-\Delta x}^t) + \frac{c}{2} \left( \frac{\Delta t}{\Delta x} \right) (u_{i+\Delta x}^t - 2u_i^t + u_{i-\Delta x}^t) \quad (6.101)$$

The added term is difference approximation of space second-degree differentiation  $\frac{\partial^2 u}{\partial x^2}$ , and affects as the diffusion term. Calculation is stably performed using the first-order forward difference from this reason. In central difference, however, for making the diffusion term, the artificial numerical viscosity term is added to the advection term.

### 6.2.2 Numerical viscosity with the second-order or fourth-order diffusion term

Since the eventh-degree differentiation generally has the effect of diffusion, the diffusion term explained in the previous subsection can change the degree of the differentiation. Here, numerical viscosities with the second-order and the fourth-order diffusion terms are explained.

Firstly, the second-order viscosity term is represented by the arbitrary physical quantities  $\phi$ ,

$$G^{\frac{1}{2}} \text{Diff.} \phi = \nu_{2h} \left[ \frac{\partial^2 (\bar{\rho} \phi')}{\partial x^2} + \frac{\partial^2 (\bar{\rho} \phi')}{\partial y^2} \right] + \nu_{2v} \frac{\partial^2 (\bar{\rho} \phi')}{\partial \zeta^2}. \quad (6.102)$$

When  $\phi$  are  $u, v, w, q_x, N_x / \rho^*$ ,  $\phi'$  is perturbation from an initial value. When  $\phi$  is  $\theta$ ,  $\phi$  is the perturbation from the basic state.



$\nu_{2h}$  and  $\nu_{2v}$  are viscosity coefficient, and are defined as follows.

$$\nu_{2h} = \frac{\alpha_{2h} \Delta_h^2}{\Delta t} \quad (6.103)$$

$$\nu_{2v} = \frac{\alpha_{2v} \Delta_v^2}{\Delta t} \quad (6.104)$$

where,

$$\Delta_h = (\Delta x \Delta y)^{\frac{1}{2}} \quad (6.105)$$

$$\Delta_v = \Delta \zeta \quad (6.106)$$

In these equations,  $\Delta_h$  and  $\Delta_v$  are the horizontal and vertical averaged grid interval in the domain. Moreover,  $\alpha_{2h}$  and  $\alpha_{2v}$  are the non-dimensional quantities. For a stable calculation, the following requirement needs to be satisfied.

$$\alpha_{2h} \leq \frac{1}{8}, \quad \alpha_{2v} \leq \frac{1}{8} \quad (6.107)$$

In this way, this viscosity term is applied to forecast variables other than pressure perturbation.

We define the second-order differentiation operator as

$$(\partial_x^2 \phi)_i = \frac{1}{\Delta x} (\phi_{i+\Delta x} - 2\phi_i + \phi_{i-\Delta x}) \quad (6.108)$$

When  $u', v', w', q'_x, N'_x / \rho^*$  are made into perturbation from the initial value and  $\theta'$  is made into perturbation from the basic state, difference notation are shown as follows.

$$G^{\frac{1}{2}} \text{Diff}.u = \nu_{2h} [\partial_x^2 (\bar{\rho}^x u') + \partial_y^2 (\bar{\rho}^x u')] + \nu_{2v} [\partial_\zeta^2 (\bar{\rho}^x u')] \quad (6.109)$$

$$G^{\frac{1}{2}} \text{Diff}.v = \nu_{2h} [\partial_x^2 (\bar{\rho}^y v') + \partial_y^2 (\bar{\rho}^y v')] + \nu_{2v} [\partial_\zeta^2 (\bar{\rho}^y v')] \quad (6.110)$$

$$G^{\frac{1}{2}} \text{Diff}.w = \nu_{2h} [\partial_x^2 (\bar{\rho}^\zeta w') + \partial_y^2 (\bar{\rho}^\zeta w')] + \nu_{2v} [\partial_\zeta^2 (\bar{\rho}^\zeta w')] \quad (6.111)$$

$$G^{\frac{1}{2}} \text{Diff}.\theta = \nu_{2h} [\partial_x^2 (\bar{\rho} \theta') + \partial_y^2 (\bar{\rho} \theta')] + \nu_{2v} [\partial_\zeta^2 (\bar{\rho} \theta')] \quad (6.112)$$

$$G^{\frac{1}{2}} \text{Diff}.q_x = \nu_{2h} [\partial_x^2 (\bar{\rho} q'_x) + \partial_y^2 (\bar{\rho} q'_x)] + \nu_{2v} [\partial_\zeta^2 (\bar{\rho} q'_x)] \quad (6.113)$$

$$G^{\frac{1}{2}} \text{Diff}.\frac{N_x}{\rho^*} = \nu_{2h} \left[ \partial_x^2 \left( \bar{\rho} \frac{N'_x}{\rho^*} \right) + \partial_y^2 \left( \bar{\rho} \frac{N'_x}{\rho^*} \right) \right] + \nu_{2v} \left[ \partial_\zeta^2 \left( \bar{\rho} \frac{N'_x}{\rho^*} \right) \right] \quad (6.114)$$

The fourth-order numerical viscosity term is considered as well as the second-order. When  $\phi$  is  $u, v, w, q_x, N_x / \rho^*$ ,  $\phi'$  is perturbation from an initial value. When  $\phi$  is  $\theta$ ,  $\phi$  is the perturbation from the basic state.

$$G^{\frac{1}{2}} \text{Diff}.\phi = -\nu_{4h} \left[ \frac{\partial^4 (\bar{\rho} \phi')}{\partial x^4} + \frac{\partial^4 (\bar{\rho} \phi')}{\partial y^4} \right] - \nu_{4v} \frac{\partial^4 (\bar{\rho} \phi')}{\partial \zeta^4} \quad (6.115)$$

$\nu_{2h}$  and  $\nu_{2v}$  are a viscosity coefficient, and are defined as follows.

$$\nu_{4h} = \frac{\alpha_{4h} \Delta_h^4}{\Delta t} \quad (6.116)$$

$$\nu_{4v} = \frac{\alpha_{4v} \Delta_v^4}{\Delta t} \quad (6.117)$$

where  $\alpha_{2h}$  and  $\alpha_{2v}$  are the non-dimensional quantities, and the values which are often used are as follows.

$$\alpha_{4h} = 0.001, \quad \alpha_{4v} = 0.001. \quad (6.118)$$

As well as the second-order, this viscosity term is applied to forecast variables other than pressure perturbation.

When  $u', v', w', q'_x, N'_x / \rho^*$  are made into perturbation from initial value and  $\theta'$  is made into perturbation from the basic state, difference notation are shown as follows.

$$G^{\frac{1}{2}} \text{Diff}.u = -\nu_{4h} \left[ \partial_x^2 \{ \partial_x^2 (\bar{\rho}^x u') \} + \partial_y^2 \{ \partial_y^2 (\bar{\rho}^x u') \} \right] - \nu_{4v} \partial_\zeta^2 \left[ \partial_\zeta^2 (\bar{\rho}^x u') \right] \quad (6.119)$$

$$G^{\frac{1}{2}} \text{Diff}.v = -\nu_{4h} \left[ \partial_x^2 \{ \partial_x^2 (\bar{\rho}^y v') \} + \partial_y^2 \{ \partial_y^2 (\bar{\rho}^y v') \} \right] - \nu_{4v} \partial_\zeta^2 \left[ \partial_\zeta^2 (\bar{\rho}^y v') \right] \quad (6.120)$$

$$G^{\frac{1}{2}} \text{Diff}.w = -\nu_{4h} \left[ \partial_x^2 \{ \partial_x^2 (\bar{\rho}^z w') \} + \partial_y^2 \{ \partial_y^2 (\bar{\rho}^z w') \} \right] - \nu_{4v} \partial_\zeta^2 \left[ \partial_\zeta^2 (\bar{\rho}^z w') \right] \quad (6.121)$$

$$G^{\frac{1}{2}} \text{Diff}.\theta = -\nu_{4h} \left[ \partial_x^2 \{ \partial_x^2 (\bar{\rho} \theta') \} + \partial_y^2 \{ \partial_y^2 (\bar{\rho} \theta') \} \right] - \nu_{4v} \partial_\zeta^2 \left[ \partial_\zeta^2 (\bar{\rho} \theta') \right] \quad (6.122)$$

$$G^{\frac{1}{2}} \text{Diff}.q_x = -\nu_{4h} \left[ \partial_x^2 \{ \partial_x^2 (\bar{\rho} q'_x) \} + \partial_y^2 \{ \partial_y^2 (\bar{\rho} q'_x) \} \right] - \nu_{4v} \partial_\zeta^2 \left[ \partial_\zeta^2 (\bar{\rho} q'_x) \right] \quad (6.123)$$

$$G^{\frac{1}{2}} \text{Diff}.\frac{N_x}{\rho^*} = -\nu_{4h} \left[ \partial_x^2 \left\{ \partial_x^2 \left( \bar{\rho} \frac{N'_x}{\rho^*} \right) \right\} + \partial_y^2 \left\{ \partial_y^2 \left( \bar{\rho} \frac{N'_x}{\rho^*} \right) \right\} \right] - \nu_{4v} \partial_\zeta^2 \left[ \partial_\zeta^2 \left( \bar{\rho} \frac{N'_x}{\rho^*} \right) \right] \quad (6.124)$$

It is more desirable to use this numerical viscosity term which has fourth-order diffusion term, since it can distinguish from the diffusion term  $\text{Turb}.\phi$  in Section ?? - "subgrid-scale diffusion", and can diffuse the ingredient of high wave number.

As a result, including these numerical viscosity term, the term  $F_\phi^t$  calculated with large time step  $t$  shown in (6.40)  $\sim$  (6.43), (6.76) and (6.77) are able to be expressed as follows.

$$F_u^t = -\text{Adv}.u^t + \left[ \overline{\rho^* f_s \bar{v}^{yx}} - \overline{\rho^* f_c \bar{w}^\zeta x} \right]^t + \left[ G^{\frac{1}{2}} \text{Turb}.u + G^{\frac{1}{2}} \text{Diff}.u \right]^{t-\Delta t} \quad (6.125)$$

$$F_v^t = -\text{Adv}.v^t - \left[ \overline{\rho^* f_s \bar{w}^{xy}} \right]^t + \left[ G^{\frac{1}{2}} \text{Turb}.v + G^{\frac{1}{2}} \text{Diff}.v \right]^{t-\Delta t} \quad (6.126)$$

$$F_w^t = -\text{Adv}.w^t + \left[ \overline{\rho^* (\text{Buoy}.\theta + \text{Buoy}.q)^\zeta} + \overline{\rho^* f_c \bar{u}^{x\zeta}} \right]^t + \left[ G^{\frac{1}{2}} \text{Turb}.w + G^{\frac{1}{2}} \text{Diff}.w \right]^{t-\Delta t} \quad (6.127)$$

$$F_\theta^t = -\text{Adv}.\theta^t + \left[ G^{\frac{1}{2}} \text{Turb}.\theta' + G^{\frac{1}{2}} \text{Diff}.\theta' \right]^{t-\Delta t} + [\rho^* \text{Src}.\theta']^t - \left[ \overline{\rho^\zeta w \partial_\zeta \bar{\theta}^\zeta} \right]^t \quad (6.128)$$

$$F_q^t = -\text{Adv}.q_x^t + \left[ G^{\frac{1}{2}} \text{Turb}.q_x + G^{\frac{1}{2}} \text{Diff}.q_x \right]^{t-\Delta t} + [\rho^* \text{Src}.q_x]^t + [\rho^* \text{Fall}.q_x]^t \quad (6.129)$$

$$F_N^t = -\text{Adv}.\frac{N_x}{\rho^*}^t + \left[ G^{\frac{1}{2}} \text{Turb}.\frac{N_x}{\rho^*} + G^{\frac{1}{2}} \text{Diff}.\frac{N_x}{\rho^*} \right]^{t-\Delta t} + \left[ \rho^* \text{Src}.\frac{N_x}{\rho^*} \right]^t + \left[ \rho^* \text{Fall}.\frac{N_x}{\rho^*} \right]^t \quad (6.130)$$

The terms  $F_w^t$  and  $F_\theta^t$  in the case of calculating gravity wave mode with small time step shown in (6.94) and (6.95) is able to be expressed as follows.

$$F_w^t = -\text{Adv}.w^t + \left[ \overline{\rho^* \text{Buoy}.q}^\zeta + \overline{\rho^* f_c \bar{u}^{x\zeta}} \right]^t + \left[ G^{\frac{1}{2}} \text{Turb}.w + G^{\frac{1}{2}} \text{Diff}.w \right]^{t-\Delta t} \quad (6.131)$$

$$F_\theta^t = -\text{Adv}.\theta^t + \left[ G^{\frac{1}{2}} \text{Turb}.\theta + G^{\frac{1}{2}} \text{Diff}.\theta \right]^{t-\Delta t} + [\rho^* \text{Src}.\theta]^t \quad (6.132)$$

## 6.3 Boundary condition

Finite difference of the basic equations shown in the previous section includes operators at  $i-1$ ,  $i+1$ ,  $j-1$ ,  $\dots$ . For calculation of forecast variables at the grid points which locate along each boundary:

$$\begin{aligned} u; & \quad i = 1, nx, & \quad j = 1, ny - 1, & \quad k = 1, nz - 1 \\ v; & \quad i = 1, nx - 1, & \quad j = 1, ny, & \quad k = 1, nz - 1 \\ w; & \quad i = 1, nx - 1, & \quad j = 1, ny - 1, & \quad k = 1, nz \\ \phi; & \quad i = 1, nx - 1, & \quad j = 1, ny - 1, & \quad k = 1, nz - 1, \end{aligned} \quad (6.133)$$

where  $\phi$  of the arbitrary scalar, and a boundary condition is needed to be given. We will explain about some kinds of a boundary condition in **CReSS** in the followings. In the fourth-order calculation, values at the grid points shifted one grid from the boundary are calculated with the second-order precision.

### 6.3.1 Lateral boundary condition

Four boundary conditions of periodic, fixed wall (mirror condition), non-gradient, and radiation, are applicable to a lateral boundary.

#### Periodic boundary condition

The periodic boundary condition is the condition that the values along the eastern (northern) boundary are equal to those along the western (southern) boundary during an integration.

In the cast that we use the periodic boundary condition in the east-west direction,

$$\begin{aligned}
 u_{1,j,k} &= u_{nx-2,j,k}, & u_{nx,j,k} &= u_{3,j,k} \\
 v_{1,j,k} &= v_{nx-2,j,k}, & v_{nx-1,j,k} &= v_{2,j,k} \\
 w_{1,j,k} &= w_{nx-2,j,k}, & w_{nx-1,j,k} &= w_{2,j,k} \\
 \phi_{1,j,k} &= \phi_{nx-2,j,k}, & \phi_{nx-1,j,k} &= \phi_{2,j,k} \\
 W_{1,j,k} &= W_{nx-2,j,k}, & W_{nx-1,j,k} &= W_{2,j,k},
 \end{aligned} \tag{6.134}$$

or in the cast that we use the periodic boundary condition in the north-south direction,

$$\begin{aligned}
 u_{i,1,k} &= u_{i,ny-2,k}, & u_{i,ny-1,k} &= u_{i,2,k} \\
 v_{i,1,k} &= v_{i,ny-2,k}, & v_{i,ny,k} &= v_{i,3,k} \\
 w_{i,1,k} &= w_{i,ny-2,k}, & w_{i,ny-1,k} &= w_{i,2,k} \\
 \phi_{i,1,k} &= \phi_{i,ny-2,k}, & \phi_{i,ny-1,k} &= \phi_{i,2,k} \\
 W_{i,1,k} &= W_{i,ny-2,k}, & W_{i,ny-1,k} &= W_{i,2,k},
 \end{aligned} \tag{6.135}$$

where  $\phi$  is the arbitrary scalar.

### Fixed wall boundary condition

In order to explain fixed wall boundary condition, for example, an east-and-west wall is shown here. Along the east and west boundaries,  $x$  differentiation in incompressible equation of conservation of mass and free-slip condition are

$$\partial \left( \frac{\partial u}{\partial x} + \frac{\partial w}{\partial \zeta} \right) / \partial x = 0, \quad \frac{\partial w}{\partial x} = 0. \tag{6.136}$$

It is drawn that the velocity  $u$  of  $x$  direction satisfies the following relation.

$$\frac{\partial^2 u}{\partial x^2} = 0 \quad \implies \quad u(x = -\Delta x, y, z) = u(x = \Delta x, y, z) \tag{6.137}$$

Under the situation, the fixed wall condition is determined as follows.

In the cast that we use the periodic boundary condition in the east-west direction,

$$\begin{aligned}
u_{1,j,k} &= -u_{3,j,k}, & u_{nx,j,k} &= -u_{nx-2,j,k} \\
v_{1,j,k} &= v_{2,j,k}, & v_{nx-1,j,k} &= v_{nx-2,j,k} \\
w_{1,j,k} &= w_{2,j,k}, & w_{nx-1,j,k} &= w_{nx-2,j,k} \\
\phi_{1,j,k} &= \phi_{2,j,k}, & \phi_{nx-1,j,k} &= \phi_{nx-2,j,k} \\
W_{1,j,k} &= W_{2,j,k}, & W_{nx-1,j,k} &= W_{nx-2,j,k},
\end{aligned} \tag{6.138}$$

or in the cast that we use the periodic boundary condition in the north-south direction,

$$\begin{aligned}
u_{i,1,k} &= u_{i,2,k}, & u_{i,ny-1,k} &= u_{i,nx-2,k} \\
v_{i,1,k} &= -v_{i,3,k}, & v_{i,ny,k} &= -v_{i,nx-2,k} \\
w_{i,1,k} &= w_{i,2,k}, & w_{i,ny-1,k} &= w_{i,nx-2,k} \\
\phi_{i,1,k} &= \phi_{i,2,k}, & \phi_{i,ny-1,k} &= \phi_{i,nx-2,k} \\
W_{i,1,k} &= W_{i,2,k}, & W_{i,ny-1,k} &= W_{i,nx-2,k},
\end{aligned} \tag{6.139}$$

where  $\phi$  is the arbitrary scalar.

### Non-gradient boundary condition

Non-gradient boundary condition is the condition that the gradient of forecast variables across boundary plane may be set to  $grad(\phi) = 0$ .

In the cast that we use the periodic boundary condition in the east-west direction,

$$\begin{aligned}
u_{1,j,k} &= u_{3,j,k}, & u_{nx,j,k} &= u_{nx-2,j,k} \\
v_{1,j,k} &= v_{2,j,k}, & v_{nx-1,j,k} &= v_{nx-2,j,k} \\
w_{1,j,k} &= w_{2,j,k}, & w_{nx-1,j,k} &= w_{nx-2,j,k} \\
\phi_{1,j,k} &= \phi_{2,j,k}, & \phi_{nx-1,j,k} &= \phi_{nx-2,j,k} \\
W_{1,j,k} &= W_{2,j,k}, & W_{nx-1,j,k} &= W_{nx-2,j,k},
\end{aligned} \tag{6.140}$$

or in the cast that we use the periodic boundary condition in the north-south direction,

$$\begin{aligned}
u_{i,1,k} &= u_{i,2,k}, & u_{i,ny-1,k} &= u_{i,nx-2,k} \\
v_{i,1,k} &= v_{i,3,k}, & v_{i,ny,k} &= v_{i,nx-2,k} \\
w_{i,1,k} &= w_{i,2,k}, & w_{i,ny-1,k} &= w_{i,nx-2,k} \\
\phi_{i,1,k} &= \phi_{i,2,k}, & \phi_{i,ny-1,k} &= \phi_{i,nx-2,k} \\
W_{i,1,k} &= W_{i,2,k}, & W_{i,ny-1,k} &= W_{i,nx-2,k},
\end{aligned} \tag{6.141}$$

where  $\phi$  is the arbitrary scalar.

### Radiational boundary condition

Radiational boundary condition is set up so that the radiational conditions with one-dimensional linear wave equation

$$\frac{\partial \phi}{\partial t} + c \frac{\partial \phi}{\partial x} = 0 \tag{6.142}$$

is satisfied for stopping the action of the unnatural solution produced in a boundary plane. Adding the attenuation term,

$$\frac{\partial \phi}{\partial t} + c \frac{\partial \phi}{\partial x} = -\gamma \phi \quad (6.143)$$

is used here. As nesting from a model output for a larger domain, such as objective analysis data, as an initial and a boundary values. The difference between **CRess** and the model for a larger domain for each physical amount is used for a boundary condition.

$$\frac{\partial (\phi_f - \phi_c)}{\partial t} + c \frac{\partial (\phi_f - \phi_c)}{\partial x} = -\gamma (\phi_f - \phi_c) \quad (6.144)$$

where  $\phi$  is the arbitrary physical quantities, bottom characters  $f$  and  $c$  indicate the values of the larger model and this model, respectively.

There are various methods in the way to solve phase velocity  $c$  in the above radiational condition. The three methods can be chosen in **CRess**.  $\phi$  and  $\phi_f - \phi_c$  is expressed with  $\Phi$  below.

[ (1) Method to calculate phase velocity for each grid point] (1) Method to calculate phase velocity for each grid point

In this case, equations excluded the attenuation term on the right side from the equations (6.143), (6.144) is solved about phase velocity  $c$ . Non-dimensional phase velocity is directly solved.

$$C_b^t = c_b^t \frac{\Delta t}{\Delta x} = -\frac{\Phi_b^{t+\Delta t} - \Phi_b^{t-\Delta t}}{2\Phi_{b-1}^t - \Phi_b^{t+\Delta t} - \Phi_b^{t-\Delta t}}, \quad -1 \leq C_b^t \leq 0, \quad \text{western edge or southern edge} \quad (6.145)$$

$$C_b^t = c_b^t \frac{\Delta t}{\Delta x} = -\frac{\Phi_b^{t+\Delta t} - \Phi_b^{t-\Delta t}}{\Phi_b^{t+\Delta t} + \Phi_b^{t-\Delta t} - 2\Phi_{b-1}^t}, \quad 0 \leq C_b^t \leq 1, \quad \text{eastern edge or northern edge} \quad (6.146)$$

These phase velocities should be calculated for each forecast variables. In the actual calculation, the following values are used.

$$C_b^t = c_b^t \frac{\Delta t}{\Delta x} = -\frac{\Phi_{b-1}^t - \Phi_{b-1}^{t-2\Delta t}}{2\Phi_{b-2}^{t-\Delta t} - \Phi_{b-1}^t - \Phi_{b-1}^{t-2\Delta t}}, \quad -1 \leq C_b^t \leq 0, \quad \text{westernmost or southernmost} \quad (6.147)$$

$$C_b^t = c_b^t \frac{\Delta t}{\Delta x} = -\frac{\Phi_{b-1}^t - \Phi_{b-1}^{t-2\Delta t}}{\Phi_{b-1}^t + \Phi_{b-1}^{t-2\Delta t} - 2\Phi_{b-2}^{t-\Delta t}}, \quad 0 \leq C_b^t \leq 1, \quad \text{easternmost or northernmost} \quad (6.148)$$

where bottom character  $b$  is the number of grid point on the boundary given by the equation (6.133).

[ (2) Method to calculate constant phase velocity in vertical direction] (2) Method to calculate constant phase velocity in vertical direction

Wave equation which assumed constant phase velocity in vertical is considered as follows.

$$\int_{z_{sfc}}^{z_{top}} \frac{\partial \Phi}{\partial t} dz + c \int_{z_{sfc}}^{z_{top}} \frac{\partial \Phi}{\partial x} dz = -\gamma \Phi \quad (6.149)$$

Using the value calculated at the inner grid point on preceding time step,

$$C_b^t = c_b^t \frac{\Delta t}{\Delta x} = - \frac{\sum_{k=kmin}^{kmax} \{ (\Phi_{b-1}^t - \Phi_{b-1}^{t-2\Delta t}) \operatorname{sgn} (2\Phi_{b-2}^{t-\Delta t} - \Phi_{b-1}^t - \Phi_{b-1}^{t-2\Delta t}) \}}{\sum_{k=kmin}^{kmax} |2\Phi_{b-2}^{t-\Delta t} - \Phi_{b-1}^t - \Phi_{b-1}^{t-2\Delta t}|},$$

.....  $-1 \leq C_b^t \leq 0$ , *western edge or southern* (6.150)

$$C_b^t = c_b^t \frac{\Delta t}{\Delta x} = - \frac{\sum_{k=kmin}^{kmax} \{ (\Phi_{b-1}^t - \Phi_{b-1}^{t-2\Delta t}) \operatorname{sgn} (\Phi_{b-1}^t + \Phi_{b-1}^{t-2\Delta t} - 2\Phi_{b-2}^{t-\Delta t}) \}}{\sum_{k=kmin}^{kmax} |\Phi_{b-1}^t + \Phi_{b-1}^{t-2\Delta t} - 2\Phi_{b-2}^{t-\Delta t}|},$$

.....  $0 \leq C_b^t \leq 1$ , *eastern edge or northern* (6.151)

where  $kmin$  and  $kmax$  are

$$u, v, \phi; \quad kmin = 2, \quad kmax = nz - 2 \quad (6.152)$$

$$w; \quad kmin = 2, \quad kmax = nz - 1. \quad (6.153)$$

[ (3) Method to add advectonal velocity to constant phase velocity](3) Method to add advectonal velocity to constant phase velocity

Phase velocities should be calculated for each forecast variable.  $c^*$  is given as the propagation velocity of typical gravity wave, such as about  $30 \text{ m s}^{-1}$ . Using  $x$ -component velocity  $u$  for the eastern and western boundaries and  $y$ -component velocity  $v$  for the northern and southern boundaries, a boundary condition is given as follows.

$$C_b^t = c_b^t \frac{\Delta t}{\Delta x} = (u - c^*) \frac{\Delta t}{\Delta x}, \quad -1 \leq C_b^t \leq 0, \quad \textit{westernmost} \quad (6.154)$$

$$C_b^t = c_b^t \frac{\Delta t}{\Delta x} = (u + c^*) \frac{\Delta t}{\Delta x}, \quad 0 \leq C_b^t \leq 1, \quad \textit{easternmost} \quad (6.155)$$

$$C_b^t = c_b^t \frac{\Delta t}{\Delta y} = (v - c^*) \frac{\Delta t}{\Delta y}, \quad -1 \leq C_b^t \leq 0, \quad \textit{southernmost} \quad (6.156)$$

$$C_b^t = c_b^t \frac{\Delta t}{\Delta y} = (v + c^*) \frac{\Delta t}{\Delta y}, \quad 0 \leq C_b^t \leq 1, \quad \textit{northernmost} \quad (6.157)$$

[ (4) Method to use constant phase velocity](4) Method to use constant phase velocity

Only  $c^*$  in (3) is used on assumption that a wave always goes away out of the calculation domain. Therefore, phase velocities at all grid points on the lateral boundary plane are the same value. However, a sign becomes reverse between eastern and western sides, and southern and northern sides.

In the case of (3) and (4), the same phase velocity as (2) is used for a vertical velocity component, such as  $u$  on the eastern and western planes and  $v$  on the southern and northern planes.

By using the phase velocity solved in (1), (2), (3), and (4), the value at a grid point on the boundary at next time step is,

for the variable integrated with small time step  $\Delta\tau$ ,

$$\Phi_b^{\tau+\Delta\tau} = \Phi_b^\tau - \frac{C_b^t}{1-C_b^t} (\Phi_{b-1}^t - \Phi_b^{t-\Delta t}) \frac{\Delta\tau}{\Delta t} - \gamma \Phi_b^{t-\Delta t} \Delta\tau, \quad \text{western edge or southern edge} \quad (6.158)$$

$$\Phi_b^{\tau+\Delta\tau} = \Phi_b^\tau + \frac{C_b^t}{1+C_b^t} (\Phi_{b-1}^t - \Phi_b^{t-\Delta t}) \frac{\Delta\tau}{\Delta t} - \gamma \Phi_b^{t-\Delta t} \Delta\tau, \quad \text{eastern edge or northern edge} \quad (6.159)$$

for the variable integrated with large time step  $\Delta t$ ,

$$\Phi_b^{t+\Delta t} = \Phi_b^{t-\Delta t} - \frac{2C_b^t}{1-C_b^t} (\Phi_{b-1}^t - \Phi_b^{t-\Delta t}) - \gamma \Phi_b^{t-\Delta t} \Delta t, \quad \text{westernmost or southernmost} \quad (6.160)$$

$$\Phi_b^{t+\Delta t} = \Phi_b^{t-\Delta t} + \frac{2C_b^t}{1+C_b^t} (\Phi_{b-1}^t - \Phi_b^{t-\Delta t}) - \gamma \Phi_b^{t-\Delta t} \Delta t, \quad \text{easternmost or northernmost} \quad (6.161)$$

where attenuation constant  $\gamma$  has the dimension of time inverse number. It is thought that the inverse number  $1/\gamma$  is suitable more than one hour and less than twelve hours.

### 6.3.2 Boundary conditions at the top and bottom

Two kinds of a boundary condition, fixed wall (mirror condition) and non-gradient, are applicable to boundary conditions at the top and bottom.

#### Fixed wall boundary condition

Basically, it can be considered as same as lateral fixed wall boundary condition. However, when we use the terrain-following coordinate system, the relation between the velocity on Cartesian coordinate system and contravariant velocity should be considered as follows.

$$W = (uJ_{31} + vJ_{32} + w) / G^{\frac{1}{2}} \quad (6.162)$$

Note that vertical velocity  $W$  when velocity is 0 on the boundary plane.



On the boundary plane at the top ( $k = nz - 1$ ),

$$W = 0, \quad w = 0 \quad (6.163)$$

and on the boundary plane at the bottom ( $k = 2$ ),

$$W = 0, \quad w = -uJ_{31} - vJ_{32}. \quad (6.164)$$

Therefore, boundary condition is given as follows.

$$\begin{aligned} u_{i,j,1} &= u_{i,j,2}, & u_{i,j,nz-1} &= u_{i,j,nz-2} \\ v_{i,j,1} &= v_{i,j,2}, & v_{i,j,nz-1} &= v_{i,j,nz-2} \\ \phi_{i,j,1} &= \phi_{i,j,2}, & \phi_{i,j,nz-1} &= \phi_{i,j,nz-2} \\ W_{i,j,2} &= 0, \quad W_{i,j,1} = -W_{i,j,3}, & W_{i,j,nz-1} &= 0, \quad W_{i,j,nz} = -W_{i,j,nz-2} \end{aligned} \quad (6.165)$$

where  $\phi$  is the arbitrary scalar. By replacing vertical velocity  $W$  at the top and bottom boundaries with the equation (6.162),

on boundary plane at the top ( $k = nz - 1$ ),

$$w_{i,j,nz-1} = 0 \quad (6.166)$$

$$w_{i,j,nz} = - \left( \overline{u^\zeta} J_{31}^x + \overline{v^\zeta} J_{32}^y + W \overline{G^{\frac{1}{2}}}^\zeta \right)_{i,j,nz-2} \quad (6.167)$$

on the boundary plane at the bottom ( $k = 2$ ),

$$w_{i,j,2} = - \left( \overline{u^\zeta} J_{31}^x + \overline{v^\zeta} J_{32}^y \right)_{i,j,2} \quad (6.168)$$

$$w_{i,j,1} = - \left( \overline{u^\zeta} J_{31}^x + \overline{v^\zeta} J_{32}^y + W \overline{G^{\frac{1}{2}}}^\zeta \right)_{i,j,3} \quad (6.169)$$

where pressure perturbation  $p'$  on the boundary plane at the bottom is given by extrapolation as follows.

$$p'_{i,j,1} = 2p'_{i,j,2} - p'_{i,j,3} \quad (6.170)$$

### Non-gradient boundary condition

Similarly to the lateral boundary,

$$\begin{aligned}
 u_{i,j,1} &= u_{i,j,2}, & u_{i,j,nz-1} &= u_{i,j,nz-2} \\
 v_{i,j,1} &= v_{i,j,2}, & v_{i,j,nz-1} &= v_{i,j,nz-2} \\
 w_{i,j,1} &= w_{i,j,2}, & w_{i,j,nz} &= w_{i,j,nz-1} \\
 \phi_{i,j,1} &= \phi_{i,j,2}, & \phi_{i,j,nz-1} &= \phi_{i,j,nz-2} \\
 W_{i,j,1} &= W_{i,j,2}, & W_{i,j,nz} &= W_{i,j,nz-1}
 \end{aligned} \tag{6.171}$$

where  $\phi$  is the arbitrary scalar.

### 6.3.3 Sponge layer

In *CRess*, the sponge layer set as arbitrary thickness is able to be set up. This is a layer for suppressing and absorbing wave reflections near the boundary plane. An additional term is added to the basic equation as follows.

$$\frac{\partial(\rho^*\phi)}{\partial t} = -\text{Adv.}\phi + \dots - \gamma_h(x, y)\rho^*(\phi - \phi_e) - \gamma_v(z)\rho^*(\phi - \phi_e) \tag{6.172}$$

where  $\phi$  is an arbitrary forecast variable,  $\phi_e$  is a variable of external data, such as an objective analysis,  $\gamma_h(x, y)$  is an attenuation coefficient which becomes small toward each horizontal boundary plane.  $\gamma_v(z)$  is attenuation coefficient which becomes small toward each vertical boundary plane.

For example, in the case of the west lateral boundary layer which made thickness of sponge layer  $d$  in horizontal,  $\gamma_h(x, y)$  is given as follows.

$$\gamma_h = \begin{cases} \alpha_h \left(1 - \frac{x}{d}\right)^3, & x \leq d \\ 0, & x > d \end{cases} \tag{6.173}$$

In the case of the bottom boundary layer which made minimum height of sponge layer in vertical,  $\gamma_v(z)$  is given as follows.

$$\gamma_v = \begin{cases} 0, & z < z_{low} \\ \alpha_v \left\{1 - \cos\left(\pi \frac{z - z_{low}}{z_{top} - z_{low}}\right)\right\}, & z \geq z_{low} \end{cases} \tag{6.174}$$

where  $\alpha_h$  and  $\alpha_v$  are horizontal and vertical attenuation constants, and have dimension of the inverse number of time. These inverse numbers  $1/\alpha_h$  and  $1/\alpha_v$  is called e-folding time. It is appropriate that these constants are set up in the range about  $1/100 \sim 1/300 \text{ s}^{-1}$ . In addition, it is suitable for the thickness of sponge layer to set up in the number of several grid points in horizontal and the thickness of vertical layer about the upper surface to 1/3 in vertical.

This term is calculated at  $t - \Delta t$ . About the forecast variable calculated at a small time step, it is treated as a value of a big time step  $2\Delta t$  like advection term and numerical viscosity term.

The adjusted value to adjust was external data's value  $\phi_e$ , such as objective analysis, in the equation (6.172). However, when a model is performed independently, and there is no objective analysis data for a forecast variable, it can adjust to initial value  $\phi_0$  or basic state value  $\bar{\phi}$  as follows.

---

$$\frac{\partial(\rho^* \phi)}{\partial t} = -\text{Adv.}\phi + \dots - \gamma_h(x, y) \rho^*(\phi - \phi_0) - \gamma_v(z) \rho^*(\phi - \phi_0) \quad (6.175)$$

$$\frac{\partial(\rho^* \phi)}{\partial t} = -\text{Adv.}\phi + \dots - \gamma_h(x, y) \rho^*(\phi - \bar{\phi}) - \gamma_v(z) \rho^*(\phi - \bar{\phi}) \quad (6.176)$$



## Part II

# Technical Description



# Chapter 7

## Application of a parallel processing

In this section, we will describe the computation method of *CR<sub>e</sub>SS* with a parallel computing.

The parallel program of *CR<sub>e</sub>SS* is fundamentally designed for running on a multiple memory type parallel computer. The parallel program is written with Message Passing Interface (MPI). *CR<sub>e</sub>SS* is available to almost all parallel computers, because the MPI is a defactor standard library. Using some workstations, *CR<sub>e</sub>SS* can also be performed in a cluster environment.

## 7.1 Technique for parallelization

### 7.1.1 Two-dimensional domain decomposition

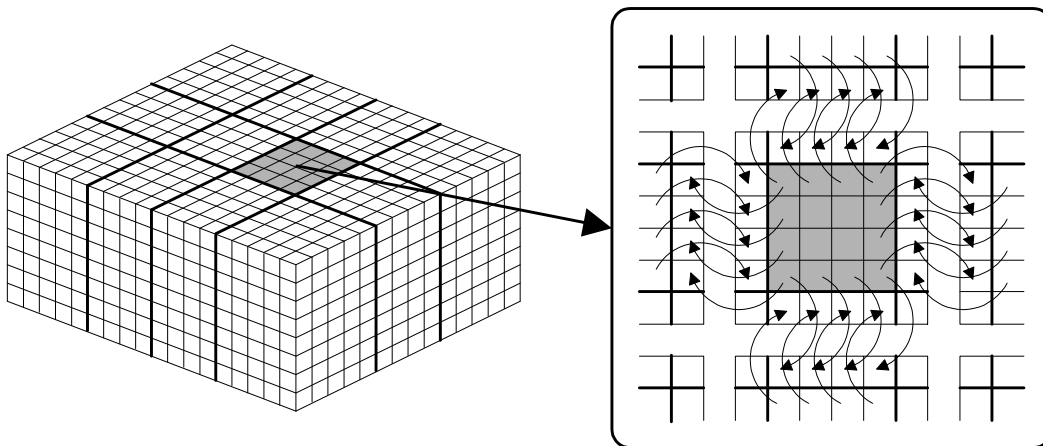
*CReSS* is fundamentally designed for a multiple memory type parallel computer. A parallel program is available for not only large multiple memory type parallel computers but also cluster systems of workstations and PC-UNIX computers.

It is easy to use the parallel processing by adopting two-dimensional domain decomposition in horizontal, because a spacial difference in horizontal is explicitly calculated in *CReSS*. Various physical processes, such as cloud microphysics, can be fundamentally calculated only with the physical quantities in the calculating cell, there are some references between upper and lower cells. A  $z^*$  coordinate system ( $\zeta$  system) makes parallelization easy that a coordinate system.

Firstly, we show the advantage of parallel processing by adopting two-dimensional domain decomposition in horizontal.

- It is easy to establish the boundary condition, because coordination system is  $z^*$ -system ( $\zeta$ -system) has no loss of data near the surface.
- It is easy to understand the technique of parallelization intuitively.
- It is available to use the same function on nodes except the node included the boundary condition.
- The load balance is high. (That is, a waiting time for other node's computation would be short.)
- Parallelization efficiency for a calculation with a number of grid points would be high.
- As a calculation domain for each node is larger, data communication between each node is relatively smaller.

We show a schematic illustration of two-dimensional domain decomposition as follows.



**Figure 7.1.** Schematic illustration of two-dimensional domain decomposition and exchange of the value in the halo region

Figure 7.1 shows a schematic illustration of the parallel computing in the case of calculating with the second-order precision. In the case of adopting central difference approximation with the second order precision, six grid points must be referred for a certain grid point. Considering the horizontal two-dimensional domain decomposition, there is no problem of reference in the vertical direction. In horizontal, however, when the approximate solution of the grid point on the boundary of decomposition is



calculated, the reference between nodes is needed. Then showing the enlarged figure on right side, we set the halo region in each node, and it is enable to obtain the same result as by one node on the computational domain computed by decomposed nodes had the halo region. For that purpose, communication between the nodes are performed only for getting the essential data from neighboring the node whenever the value in the halo region is needed (The detail is shown in subsection 7.1.2)

In addition, *CRess* is available to compute with fourth-order precision. For that purpose, it is necessary to communicate much the value which is needed by doubling a halo region like the case of second order precision. It is possible to change the precision of the differential equation by the same method, for example, sixth-order, eighth-order, ... and N-th-order.

### 7.1.2 Example of parallel computing

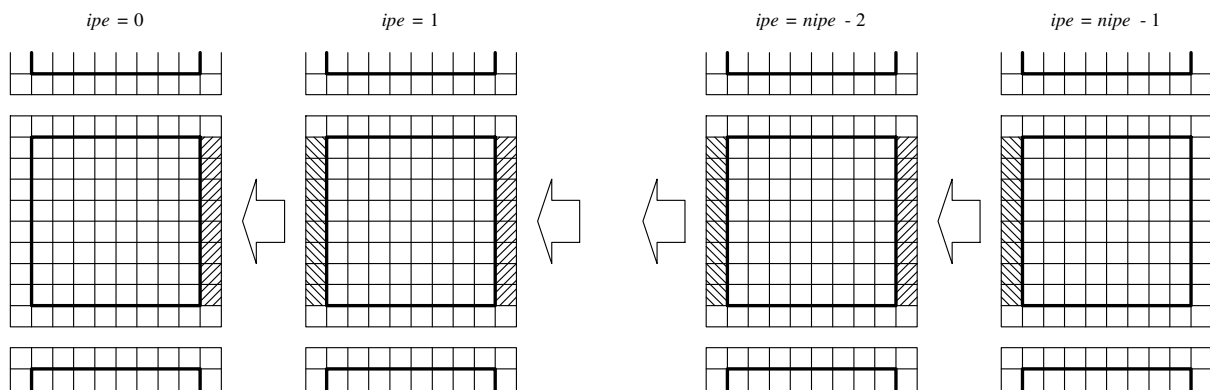
In this subsection, we describe the communication between halo regions explained in subsection 7.1.1.

The source code is written with Message Passing Interface (MPI). The MPI is the interface specifications for the call form of various communication routines required for a multiple memory type parallel computer and the function and subroutines of C and Fortran. But it is interface specification to the last, and the concrete application method is not determined.

As for the details about MPI, it is good to refer to the following homepages.

<http://www.mpi-forum.org>

For communicating by MPI such as figure 7.1, it is essential to repeat a shift in all directions except the boundary plane. Figure 7.2 is the example of a shift to the halo region of east side from the halo region of west side except the boundary plane. In this figure,  $nipe$  is numbers of node in the east-west direction and  $ipe$  is node number.



**Figure 7.2.** This is a example of a shift to the halo region of east side from the halo region of west side.

The source code of the MPI parts in *CReSS* is shown as following. The part of it is shown here, so for understanding the detail it is necessary to refer `exchansn.f` and `exchanwe.f` in the directory `Src`.

---

```
! A node at the westernmost end has no transmitting buffer
if(ipe.eq.0) then
  dst=mpi_proc_null
end if

! A node at the easternmost end has no transmitting buffer
if(ipe.eq.nipe-1) then
  src=mpi_proc_null
end if

! A setup of transmitting buffers other than a node at the westernmost end
if(ipe.ne.0) then

  do 140 k=kbmin,kbmax
  do 140 j=jbmin,jbmax
    ib=(k-kbmin)*jbm xn1-jbmin1+j

    sbuf(ib)=var(iwsend,j,k)

140    continue

  end if

! The call of the MPI routine for transmission and reception
call mpi_sendrecv(sbuf,siz,mpi_real,dst,tag,rbuf,siz,mpi_real,src,
.               tag,mpi_comm_world,stat,ierr)

! A setup of receiving buffers other than a node at the easternmost end
if(ipe.ne.nipe-1) then

  do 170 k=kbmin,kbmax
  do 170 j=jbmin,jbmax
    ib=(k-kbmin)*jbm xn1-jbmin1+j

    var(ierecv,j,k)=rbuf(ib)

170    continue

  end if
```

---

Although the shift is mounted by `mpi_sendrecv` here, it is better not to mount this by `mpi_send` and `mpi_recv`. Although it is not impossible, various programming can be considered and may carry out a deadlock depending on the case. Moreover, by the formal specification of the MPI, the guarantee of operation of a program will be impossible, since it seems that it has not defined whether a deadlock is carried out by this or it does not carry out.

## 7.2 Test of the parallel program

### 7.2.1 Inspection of coincidence of a calculation result

Since parallel programming adopted in *CRess* should just do a shift fundamentally shown in the previous section, the result performed using several nodes must be completely same as the result performed using one node (in calculation for the whole average value, it is not necessarily to get complete agreement. Since it does not apply operation to the value in shifting, disagreement cannot happen.). We checked the agreement by *CRess* as follows.

When several nodes are used for calculation, a result is outputted from every node.

```
ex. : exprim.dmpxxxxxx.pe0000.bin ~ exprim.dmpxxxxxx.peyyyyy.bin
```

For comparing to the result by one node and that by several nodes, we gather the above data files up to one file. (A post processor “unite” can gather the output files up to a binary unformatted file with direct access.) The file was compared with the result of calculation with one node. If the files of both one node and several nodes are text format, we can confirm their complete agreement as the following command.

```
ex. : % diff exprim.dmpxxxxxx.united.txt exprim.dmpxxxxxx.txt
```

As a result of executing this command, if nothing happens (0 byte of file will be made if a file is outputted to a standard input/output), the result must completely be in agreement. *exprim* of file name is an experiment name (see section 9.1).

We perform this verification to various experiments with *CRess*, and there is no problem in the parallelization.

### 7.2.2 Efficiency of parallel processing

Finally, the efficiency of parallel processing is shown by a comparison of calculations with the various numbers of processing elements (PEs). The performance of parallel processing of *CRess* was tested by a simulation which had  $67 \times 67 \times 35$  grid points and which was integrated for 50 steps on HITACHI SR2201.

As increase of the numbers of PEs, the calculation time decreased almost linearly (Fig. 7.3). The results of the test showed a sufficiently high performance of the parallel computing of *CRess*. In figure 7.4, it can see that the efficiency of parallel processing is falling little by little. When the number of PEs was 32, the efficiency decreased significantly. Because the number of grid points was too small to use the 32 PEs, the communication between PEs became relatively large for the present verification setting. This result of the verification shows that the efficiency of parallel processing of *CRess* is higher when each PE shares larger number of grid points. The calculation with a lot of PEs is more useful for larger calculation.

However, there is a problem. Especially, it is becoming more remarkable in the case of using the option for cloud microphysics. Because the program for cloud microphysics has so many `if` sentences. For example, when a node without clouds must wait for the processing of the node with clouds, efficiency would be small. In the present condition, *CRess* does not have advanced parallel processing changed from the first setup to the more efficient setup in the calculation.

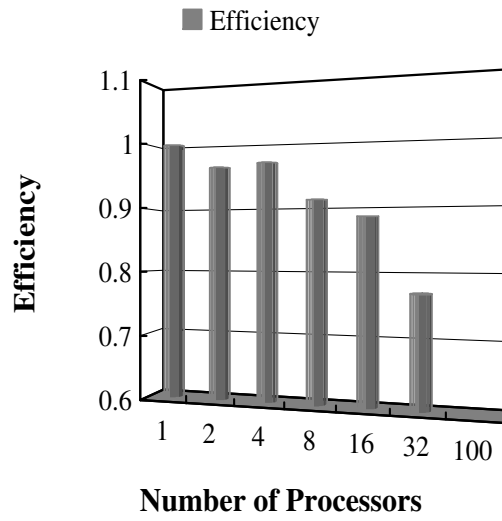
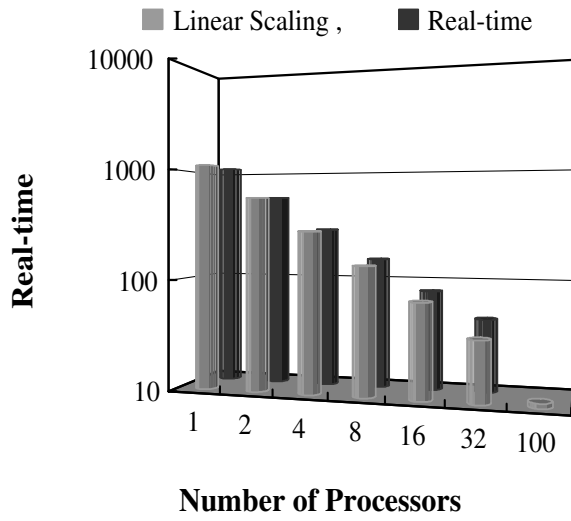


Figure 7.3. Time for parallel processing from a test experiment [s]

Figure 7.4. Efficiency of parallel processing from a test experiment

The values of efficiency in the graph is somewhat wavy in figure 7.4, since it is based on the difference between the ways of a decomposition on two-dimensional domain decomposition. (For example, when a calculation is performed with four nodes, there is the ways of dividing of  $1 \times 4$ ,  $2 \times 2$ , or  $4 \times 1$ .) However, it is sufficient for seeing the tendency of the efficiency of parallel processing.

## Chapter 8

# Program lists and tree diagram

In this chapter, general descriptions of the application programs of *CRess*, subroutines consisted the programs, include files and libraries of MPI will be described. The tree diagrams of the programs will be presented. Most descriptions are about the parallel version of *CRess*. Since the serial version of *CRess* is essentially the same as the parallel version, the readers will have no trouble with the descriptions even if they use the serial version.

Each source code of *CRess* includes a single subroutine and each file has the same name as subroutine. This is helpful to read the source code as well as the tree diagram.

## 8.1 Program list

In this section, the programs of *CReSS*, the composing subroutines, include files and the library subroutine of MPI will be described.

### 8.1.1 Main programs of *CReSS*

*CReSS* is composed of one solver (*solver*), two pre-processor (*gridata* and *terrain*) and one post-processor (*gather*) as listed below.

programs		description
<i>solver</i>	<code>solver.exe</code>	The solver program to perform the time integration from an initial value with boundary conditions.
<i>gridata</i>	<code>gridata.exe</code>	A pre-processor to produce an initial and boundary values for the solver by interpolating a grid point value to the grid points of the solver.
<i>terrain</i>	<code>terrain.exe</code>	A pre-processor to produce a model terrain data by interpolating of a topography data.
<i>gather</i>	<code>gather.exe</code>	A post-processor to combine multiple-output history files and/or the multiple terrain file in to a single file after parallel calculation.

### 8.1.2 List of subroutines

*CReSS* is composed of a few hundreds subroutines. All subroutines are listed below and their functions are described. Each subroutine file includes its corresponding subroutine. The names in the list will also refer to the file names.

subroutine	description
<code>adjstnc</code>	Adjust number density of ice phase to be consistent with mixing ratio.
<code>adjstq</code>	Adjust a negative mixing ratio to be zero.
<code>advbsp</code>	Calculate vertical advection term of the basic pressure field.
<code>advbspt</code>	Calculate vertical advection term of the basic potential temperature field.
<code>adv</code>	Calculate advection terms of scalar variables.
<code>advuvw</code>	Calculate advection terms of velocity vectors.
<code>aggregat</code>	Aggregation of number density of the ice phase.
<code>baserho</code>	Multiply Jacobian with the basic density.
<code>bc8u</code>	Calculate the boundary value defined at the points of u-velocity component.
<code>bc8v</code>	Calculate the boundary value defined at the points of v-velocity component.
<code>bc8w</code>	Calculate the boundary value defined at the points of w-velocity component.
<code>bcbase</code>	Calculate the boundary value of basic state variables.
<code>bcdfbt</code>	Calculate the upper and lower boundary values of the velocity deformation tensor.
<code>bcphi</code>	Calculate the boundary value of the Poisson equation in the retrieval.
<code>bcs2d</code>	Calculate the boundary value of two-dimensional scalar variables.
<code>bcsmsbt</code>	Calculate the upper and lower boundary values for the 4th-order numerical smoothing terms of scalar variables.
<code>bcsmsn</code>	Calculate the south and north boundary values for the 4th-order numerical smoothing terms of scalar variables.
<code>bcsmsw</code>	Calculate the east and west boundary values for the 4th-order numerical smoothing terms of scalar variables.

continued

subroutine	description
bcsmbt	Calculate the upper and lower boundary values for the 4th-order numerical smoothing terms of the x-component of velocity.
bcsmusn	Calculate the south and north boundary values for the 4th-order numerical smoothing terms of the x-component of velocity.
bcsmuwe	Calculate the east and west boundary values for the 4th-order numerical smoothing terms of the x-component of velocity.
bcsmbvt	Calculate the upper and lower boundary values for the 4th-order numerical smoothing terms of the y-component of velocity.
bcsmvsn	Calculate the south and north boundary values for the 4th-order numerical smoothing terms of the y-component of velocity.
bcsmvwe	Calculate the east and west boundary values for the 4th-order numerical smoothing terms of the y-component of velocity.
bcsmbwt	Calculate the upper and lower boundary values for the 4th-order numerical smoothing terms of the vertical velocity component.
bcsmwsn	Calculate the south and north boundary values for the 4th-order numerical smoothing terms of the vertical velocity component.
bcsmwwe	Calculate the east and west boundary values for the 4th-order numerical smoothing terms of the vertical velocity component.
bruntv	Calculation of the buoyancy frequency.
buoytke	Calculation of the buoyancy term in the equation of the turbulent kinetic energy.
buoywb	Calculation of the buoyancy term in the time integration of the large time step.
buoyws	Calculation of the buoyancy term in the time integration of the small time step.
cast1d	Broadcast the horizontally uniform data (the interpolated sounding data).
castexbc	Broadcast the boundary values.
castname	Broadcast the namelist variables.
chkfile	Check the configuration of calculation when the restart files are read in.
chkitr	Check conversion of the mass conservation model.
chkmoist	Check if the experiment is moist or dry.
chkopen	Check if a history file with the same extension is opened.
chksedim	Check if the terms of sedimentation of hydrometeors are calculated at the same time intervals.
closedmp	Close history files.
cloudamt	Calculation of cloud amount.
cloudphy	The main driver routine of the cloud micro-physics.
cnt2phy	Conversion of the vertical velocity component in the calculation coordinate to that in the physical coordinate.
coldrain	The driver routine of the bulk cold rain.
collc2r	Collision term between the cloud water and rain water.
collect	Collection terms between the hydrometeors including ice phase.
convc2r	Conversion term from the cloud water to rain water.
convers	Conversion terms between the hydrometeors including ice phase.
copy	Copy array variables to another array with the same dimension.
coriuv	Coriolis forces acting on the horizontal velocity components.
coriuvw	Coriolis forces acting on the all velocity components.
cpondall	Timing adjustment between the all processing elements.
currpe	Get the processing element number of the processing point in the calculation domain in the pre-processors.
deposit	Deposition term of hydrometeors.
destroy	Error processing in calculation.
dfomten	Calculate the deformation tensor.
diabat	Adiabatic term in the pressure equation.

continued to the next page

---

continued

subroutine	description
diffequa	Solve the poison equation in the retrieval.
disptke	Dissipation term of the equation of the turbulent kinetic energy.
diver2d	Calculate the horizontal divergence.
diver3d	Calculate the 3 dimensional divergence.
diverpe	Calculate the divergence term of the pressure equation in HE-VE scheme.
diverpih	Calculate the horizontal divergence term of the pressure equation in HE-VI scheme.
diverpiv	Calculate the vertical divergence term of the pressure equation in HE-VI scheme.
eddyvs1	Calculate the eddy viscous coefficient and the turbulent mixing length with respect to velocity.
eddyvs2	Calculate the eddy viscous coefficients of the scalar variables.
eddyvsfc	Calculate the eddy viscous coefficients of the vertical diffusion in the PBL process.
endmpi	Termination of MPI process.
estimp	Calculate the production term of graupel and shading term.
evapr2v	Evaporation from rain water.
exbcbs	Forcing the boundary value of scalar variables to the boundary data in the large time step.
exbcss	Forcing the boundary value of scalar variables to the boundary data in the small time step.
exbcu	Forcing the boundary value of the x-component of velocity to the boundary data.
exbcv	Forcing the boundary value of the y-component of velocity to the boundary data.
exbcw	Forcing the boundary value of the vertical component of velocity to the boundary data.
exchansn	Exchange data in the y-direction between the neighboring processing elements.
exchanwe	Exchange data in the x-direction between the neighboring processing elements.
findrij	Calculate real array indexes of the model grids in the data domain.
fituvwc	Adjustment of velocity in order to satisfy the mass conservation.
fltret	Asselin filter of scalar variables in retrieval.
forcep	Driver routine of the advection and forcing terms of pressure perturbation.
forcept	Driver routine of the advection and forcing terms of potential temperature perturbation.
forceq	Driver routine of the advection and forcing terms of mixing ratios of liquid water substances.
forceqv	Driver routine of the advection and forcing terms of mixing ratio of water vapor.
forceuv	Driver routine of the advection and forcing terms of velocity.
freezing	Calculate the freezing term of rain water.
g2kg	Convert the unit of mixing ratios of water substances from $[g\ kg^{-1}]$ to $[kg\ kg^{-1}]$ .
gaussel	The Gaussian elimination in HE-VI.
getbase1	Extract the base state which satisfies the hydrostatic balance from the external data in the data domain.
getbase2	Extract the base state which satisfies the hydrostatic balance from the external data in the model domain.
getcname	Get a namelist variable of character type.
getdate	Get a character variable of <i>yyyymmddhhmm</i> from model time in the <i>ssssss</i> form.
getdianc	Diagnostic calculation of the number density of ice when it is not prognostic.
getexner	Exner function.
getiname	Get a namelist variable of integer type.
getindx	Get a range of array index in the model domain to interpolate the external data.
getmxn	Find the maximum and minimum values of prognostic variables in the model domain.
getpt0	Set the initial perturbations of the potential temperature.
getref	Calculate horizontally averaged reference potential temperature.
getrname	Get a namelist variable of real type.
getrsfc	Calculate the reference density at the surface.

---



continued

subroutine	description
gettime	Find the model time.
gettrn	Set the model terrain.
getunit	Get the I/O number of file.
getxy	Calculate independent variable of the horizontal coordinate $x, y$ of model grid.
getz	Calculate independent variable of the vertical coordinate $\zeta$ of model grid.
gridrv	Main driver program of the pre-processor <i>gridata</i> .
gseidel	The Gauss-Seidel scheme in the time integration of HE-VI.
gthdrv	Main driver program of the pre-processor <i>gather</i> .
heve	The driver routine of HE-VE.
hevi	The driver routine of HE-VI.
hint2d	Interpolate 2 dimensional data to the model grid.
hint3d	Interpolate 3 dimensional data to the model grid.
hndmore0	Adjust total of terms to decrease each mixing ratio.
inichar	Initialize character variables with space.
inimpi	Initialize MPI process and parameters of parallel processing.(solver)
inipe	Initialize MPI process and parameters of parallel processing.(pre-processor)
inisfc	Initialize variables of the surface process.
iniunit	Initialize table of I/O (set from 11 to 64).
inivar	Driver routine to initialize the prognostic variables.
intgdrv	Driver routine of the time integration.
intrpdrv	Driver routine to interpolate data into model grid in the pre-processor <i>gridata</i> .
jacobian	Calculate Jacobian and its components $z, x$ and $z, y$ .
kg2g	Convert unit of water substance from $[\text{kg kg}^{-1}]$ to $[\text{g kg}^{-1}]$ .
lbcss	Calculate the lateral boundary values of the pressure perturbation.
lbcsb	Calculate the lateral boundary values of scalar variables except pressure.
lbcu	Calculate the lateral boundary values of the x-component of velocity.
lbcv	Calculate the lateral boundary values of the y-component of velocity.
lbcw	Calculate the lateral boundary values of the z-component of velocity.
l12xy	Calculate the coordinate number $x, y$ of longitude and latitude.
lspdmp	Relaxation coefficient of the lateral sponge layer.
lspht	Interpolate the model terrain in order to be linked smoothly with the terrain of outer data in the lateral sponge layer.
lspss	Lateral boundary forcing of scalar variables in the sponge layer.
lspuvw	Lateral boundary forcing of vector variables in the sponge layer.
mapfct	Calculate the map factor.
melting	Melting terms of solid water substance.
move2d	Perform the coordinate movement.
newcold	Post process of the potential temperature perturbation, mixing ratios and number densities after the microphysical process.
nuclet11	Nucleation due to deposition.
nuclet12	Nucleation due to condensation freezing, contact freezing and immersion freezing.
nuclet2	The secondary nucleation.
nudges	Nudging scalar variables.
nudgeuvw	Nudging vector variables.
numchar	Count character number of a character variables.
opendmp	Open the history files.
outdmp	Drive routine to output the history file.
outdmp2d	Output 2 dimensional variables to the history files.
outdmp3d	Output 3 dimensional variables to the history files.
outexbc	Output the lateral boundary values to the lateral boundary files.
outgeo	Output the geographical data (altitude of terrain, longitude and latitude).

continued to the next page

continued

subroutine	description
outgrd	Output 3 dimensional grid point values to the grid data files.
outmxn	Output maximum and minimum values of prognostic variables to the standard output.
outpre2d	Output 2 dimensional external data.
outrcs	Output the prognostic variables to the re-start files.
outsfc	Driver routine to output the variables of the surface process to the history files.
outstd1	Output execution status to the standard output. No.1.
outstd2	Output execution status to the standard output. No.2.
outstd3	Output execution status to the standard output. No.3.
outstd4	Output execution status to the standard output. No.4.
outstd5	Output execution status to the standard output. No.5.
outstd6	Output execution status to the standard output. No.6.
outstd7	Output execution status to the standard output. No.7.
outstd8	Output execution status to the standard output. No.8.
outstd9	Output execution status to the standard output. No.9.
outstd10	Output execution status to the standard output. No.10.
outstd11	Output execution status to the standard output. No.11.
pc2kg	Convert the relative humidity [%] to the water vapor mixing ratio [kg kg <sup>-1</sup> ].
pgrad	Calculate the pressure gradient force.
pgradiv	Calculate the vertical pressure gradient force in HE-VI.
pgratio	Calculate the partition ratio of rain to snow and graupel.
phasev	Drive routine to calculate the phase velocity difference between the external data and prognostic data at the lateral boundary.
phasevbc	Drive routine to calculate the phase velocity difference between the external data and prognostic data at the lateral boundary.
phvbc	Calculate the phase velocity of the difference between the external scalar data and prognostic scalar variables at the lateral boundary.
phvbcuvw	Calculate the phase velocity of the difference between the external velocity data and prognostic velocity at the lateral boundary.
phvs	Calculate the phase velocity of the prognostic scalar variables at the lateral boundary.
phvuvw	Calculate the phase velocity of the prognostic velocity at the lateral boundary.
phy2cnt	Calculate the contravariant velocities in the physical coordinates.
phycood	Calculate the coordinate values of $z$ in the physical coordinates.
productg	Production rate of grouple.
putunit	Return the I/O unit number to the table of number after it is used.
radiat	Solar radiation and downward long wave radiation.
rdexbc	Read in lateral boundary values.
rdgrd	Read in 3-dimensional grid point data.
rdini	Read in 3-dimensional initial data.
rdname	Read in namelist variables.
rdobj	Read in 3-dimensional grid point values (in the pre-processor <i>gridata</i> ). Users may change the statements to read in.
rdpre2d	Read in 2-dimensional data (for example, terrain data).
rdres	Read in the re-start files.
rdsnd	Read in the 1-dimensional sounding data.
rdstd	Input routine of the post-processor <i>gather</i> .
rdtrn	Read in the terrain data in the pre-processor <i>terrain</i> . Users may change the statement to read in.
resetag	Reset the tag of MPI function to zero.
retrvp	Main driver routine of the retrieval of the pressure perturbation.
retrvs	Main driver routine of the retrieval of scalar variables.
retrvuvw	Main driver routine of the retrieval of the velocity components.

continued to the next page

continued

subroutine	description
rotuvm2s	Rotate the horizontal velocity components from the model coordinate to the longitude-latitude coordinate.
rotuvs2m	Rotate the horizontal velocity components from the longitude-latitude coordinate to the model coordinate.
roughitr	Iteration calculation of the roughness over the sea.
roughnxt	Calculate the roughness over the sea for the next time step.
satadjst	The moist saturation adjustment.
sediment	The sedimentation terms of rain, snow and graupel.
sedimqr	The sedimentation term of rain.
set0grd	Initialize the array variables in the pre-processor <i>gridata</i> .
set0name	Initialize the array of namelist variables.
set0slv	Initialize the array variables in the solver <i>solver</i> .
set0trn	Initialize the array variables in the pre-processor <i>terrain</i> .
set1d	Set the horizontally uniform variables.
setbase	Calculate the base statement of prognostic variables at the model grid points.
setcnst	Set a constant variable to the array.
setcold	Calculate some physical quantities in the bulk cold rain process.
setgrid	The driver routine to set the model grid.
setproj	Calculate coefficients for map projections.
setsfc	Calculate virtual potential temperature and the water vapor mixing ration at the surface.
setsndbc	Set the lateral boundary condition when the sounding data is used.
setvar1	Initialization routine of the prognostic variables. No.1
setvar2	Initialization routine of the prognostic variables. No.2
sfcflx	Calculate the friction velocity and the coefficient of flux at the surface.
sfcphy	The main driver routine of the surface process.
sheartke	The shearing stress calculation.
shedding	Shading of water from snow and graupel.
slay	Halt the program after the process of <b>destroy</b> .
slvdrv	The main driver routine of the solver program <i>solver</i> .
smoo2s	Calculate the 2nd-order numerical smoothing terms of scalar variables.
smoo2uvw	Calculate the 2nd-order numerical smoothing terms of velocities.
smoo4s	Calculate the 4th-order numerical smoothing terms of scalar variables.
smoo4uvw	Calculate the 4th-order numerical smoothing terms of velocities.
sndwave	Calculation of the sound speed.
sparprt	Separate variables into the base state and perturbations.
steppe	Time integration of the pressure perturbation in HE-VE.
steppi	Time integration of the pressure perturbation in HE-VI.
stepptb	Time integration of the perturbation potential temperature in the large time step.
steppts	Time integration of the perturbation potential temperature in the small time step.
stepq	Time integration of the mixing ratios of solid and liquid water substances.
stepqv	Time integration of the mixing ratio of water vapor.
steptg	Time integration of the ground temperature.
steptke	Time integration of the turbulent kinetic energy.
stepu	Time integration of the x-component of velocity.
stepv	Time integration of the y-component of velocity.
stepwe	Time integration of the vertical velocity component in HE-VE.
stepwi	Time integration of the vertical velocity component in HE-VI.
stretch	Stretching of the model grid in the vertical direction.
strsten	The deformation tensor.
swp2nxt	Exchange the arrays of prognostic variables in the order of time.

continued to the next page

continued

subroutine	description
swpret	Exchange the arrays of scalar variables of the retrieval in the order of time.
t2pt	Conversion of temperature to potential temperature.
timeflt	Asselin's time filter.
tkequa	The driver routine of 1.5th-order closure.
trilat	Coriolis parameter.
trndrv	Main driver routine of the pre-processor <i>terrain</i> .
turbdrv	Main driver routine of the turbulence process.
turbflx	Calculate the turbulent flux of scalar prognostic variables.
turbret	Driver routine of turbulent process of scalar variables in retrieval.
turbs	The turbulent terms (subgrid-scale process) of scalar prognostic variables.
turbsfc	The turbulent terms of vertical diffusion.
turbuvw	The turbulent terms (subgrid-scale process) of velocities.
var8uvw	Interpolate of variables at the velocity grid point.
vbcp	The top and bottom boundary condition of the pressure perturbation.
vbcpt	The top and bottom boundary condition of the perturbation of potential temperature.
vbcsb	The top and bottom boundary condition of the mixing ratios.
vbcu	The top and bottom boundary condition of the x-component of velocity.
vbcv	The top and bottom boundary condition of the y-component of velocity.
vbcwp	The top and bottom boundary condition of the vertical velocity component.
vbcwc	The top and bottom boundary condition of the contravariant vertical velocity.
vint11d	Vertical interpolation of 1-dimensional data into 1-dimensional vertical fine grid.
vint13d	Vertical interpolation of 1-dimensional data into the model grid.
vint31d	Vertical interpolation of 3-dimensional data into 1-dimensional vertical fine grid.
vint33d	Vertical interpolation of 3-dimensional data into the model grid.
vspdmp	Calculate the dumping coefficients in the upper sponge layer.
vsps	Dumping process of the scalar variables in the upper sponge layer.
vspuvw	Dumping process of the velocity components in the upper sponge layer.
warmrain	The main driver routine of the bulk warm rain process.
xy2ll	Calculate the longitude and latitude from $x, y$ coordinate.
xy2ij	Calculate the real array number of $x, y$ coordinate.

### 8.1.3 list of include file

*CRess* has two types of include files; one is `common` statement of Fortran and the other is `namelist` statement. There are also temporal include files which include array dimensions.

file name	description
comdmp.h	Control parameters and <code>common</code> statement for history dump.
comindx.h	Indexes of the array of the namelist variables and their <code>common</code> statement.
commpi.h	Parameters and related <code>common</code> statement for MPI.
comname.h	Array of namelist variables and their <code>common</code> statement.
comstd.h	Control parameters and <code>common</code> statement for the standard I/O.
comunit.h	Array of the I/O unit number table and its <code>common</code> statement.
namelist.h	Namelist variables and their namelist statements.

### 8.1.4 MPI subroutines list

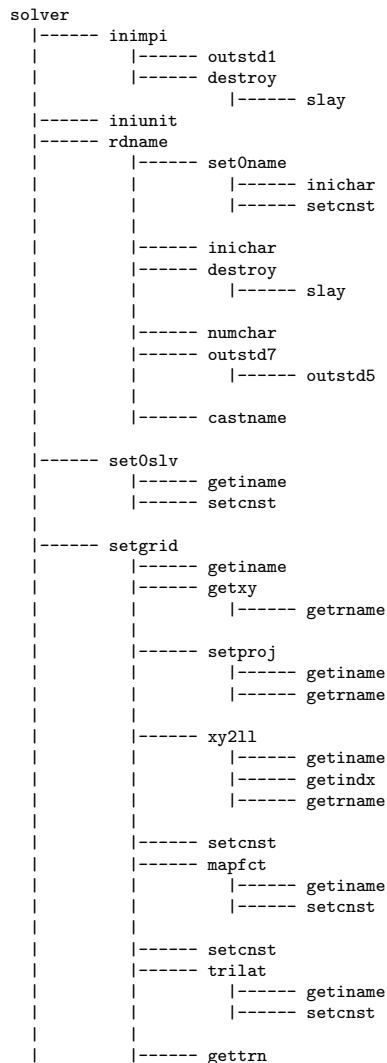
The program codes of *CRess* have MPI subroutines as well as the above subroutines. MPI subroutines used in *CRess* are listed in this subsection. The detailed description of MPI routine is found in the manuals of MPI.

subroutine	description
<code>mpi_abort</code>	Halt MPI processes.
<code>mpi_allreduce</code>	Perform the same process in all nodes.
<code>mpi_barrier</code>	Adjust timing between nodes.
<code>mpi_bcast</code>	Broadcast from a node to all nodes.
<code>mpi_comm_rank</code>	Get the number of node.
<code>mpi_comm_size</code>	Get a number of all nodes.
<code>mpi_comm_split</code>	Split communicator.
<code>mpi_finalize</code>	Terminate MPI.
<code>mpi_gather</code>	Gather data to a node from other nodes.
<code>mpi_init</code>	Initialize MPI.
<code>mpi_sendrecv</code>	Send and receive data simultaneously.

## 8.2 Tree diagrams of programs

In this section, tree diagrams of programs which are listed in Section 8.1.1. Note that MPI library routines are not included in the diagram because they are not original to *CReSS*.

### 8.2.1 *solver*

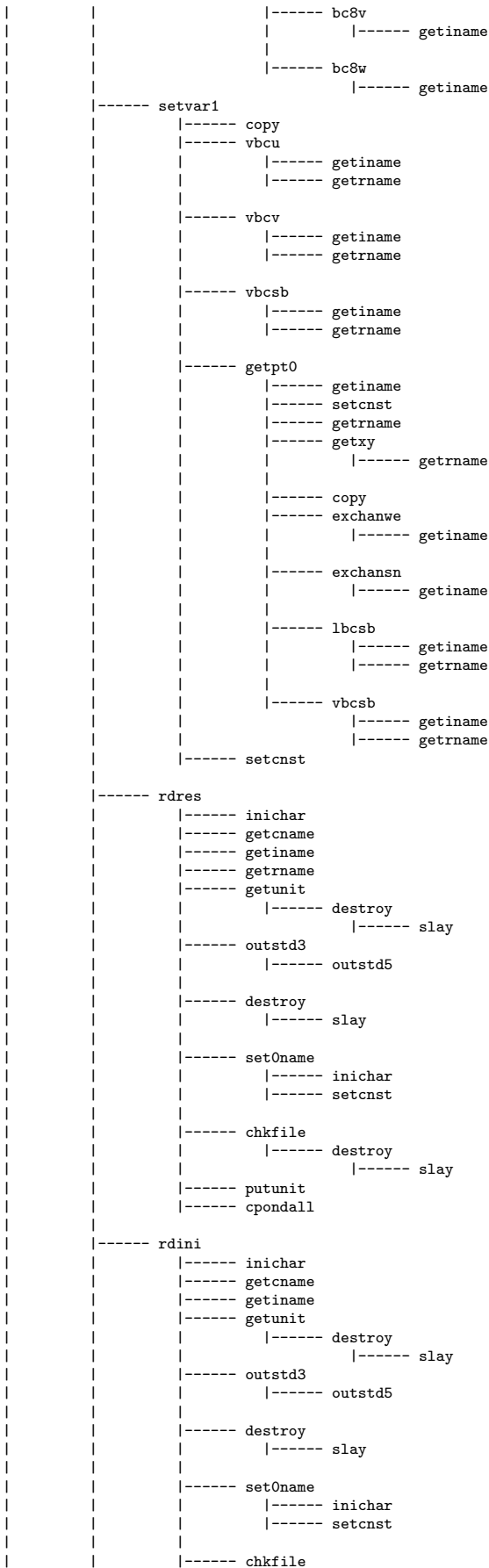


```

|----- getiname
|----- getrname
|----- setcnst
|----- rdpre2d
|----- inichar
|----- getcname
|----- getiname
|----- getunit
|----- destroy
|----- slay
|----- outstd3
|----- outstd5
|----- destroy
|----- slay
|----- set0name
|----- inichar
|----- setcnst
|----- chkfile
|----- destroy
|----- slay
|----- putunit
|----- cpondall
|----- exchanwe
|----- getiname
|----- exchansn
|----- getiname
|----- bcs2d
|----- getiname
|----- outgeo
|----- getrname
|----- inichar
|----- getcname
|----- getiname
|----- getunit
|----- destroy
|----- slay
|----- outstd3
|----- outstd5
|----- destroy
|----- slay
|----- putunit
|----- chkopen
|----- destroy
|----- slay
|----- cpondall
|----- getz
|----- getrname
|----- phycood
|----- getiname
|----- getrname
|----- copy
|----- stretch
|----- getiname
|----- getrname
|----- destroy
|----- slay
|----- jacobian
|----- getiname
|----- exchanwe
|----- getiname
|----- bc8u
|----- getiname
|----- exchansn
|----- getiname
|----- bc8v
|----- getiname
|----- var8uvw
|----- getiname
|----- exchanwe

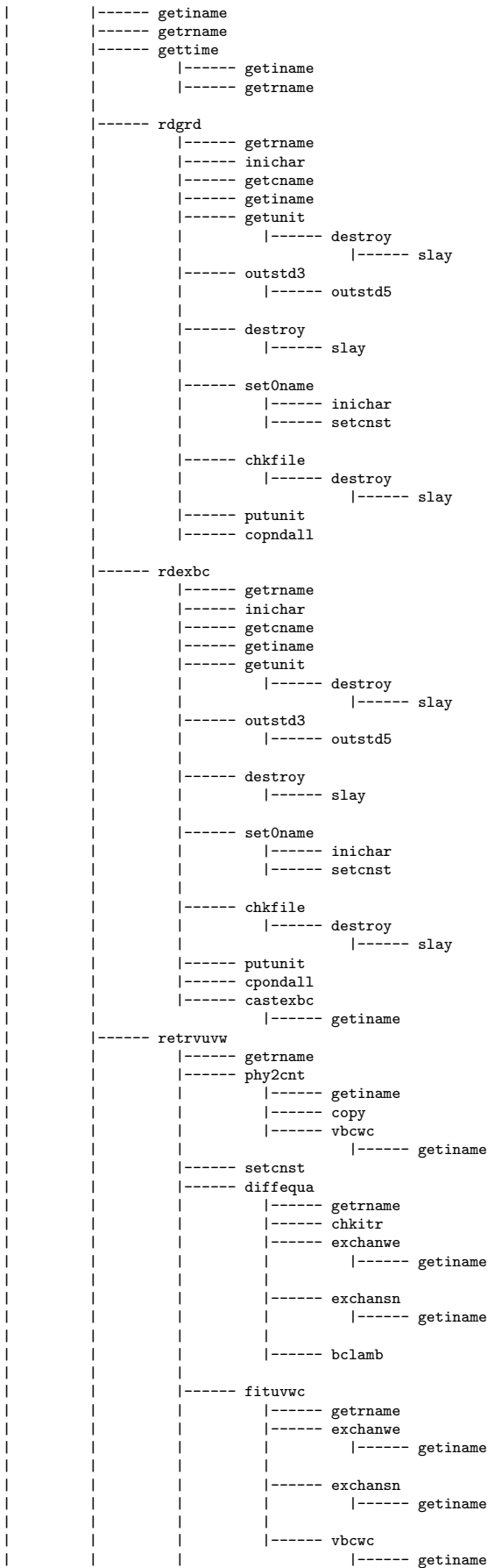
```

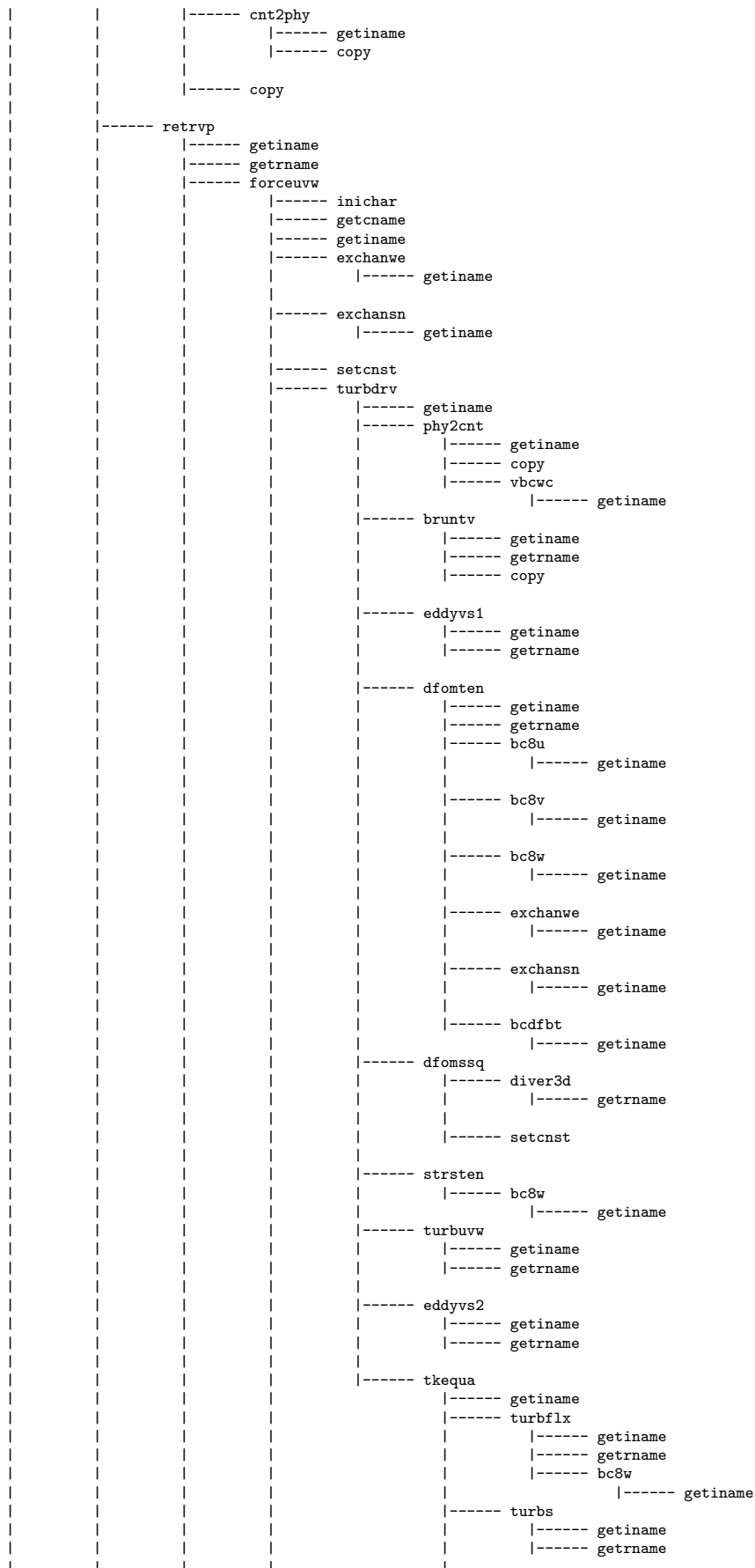






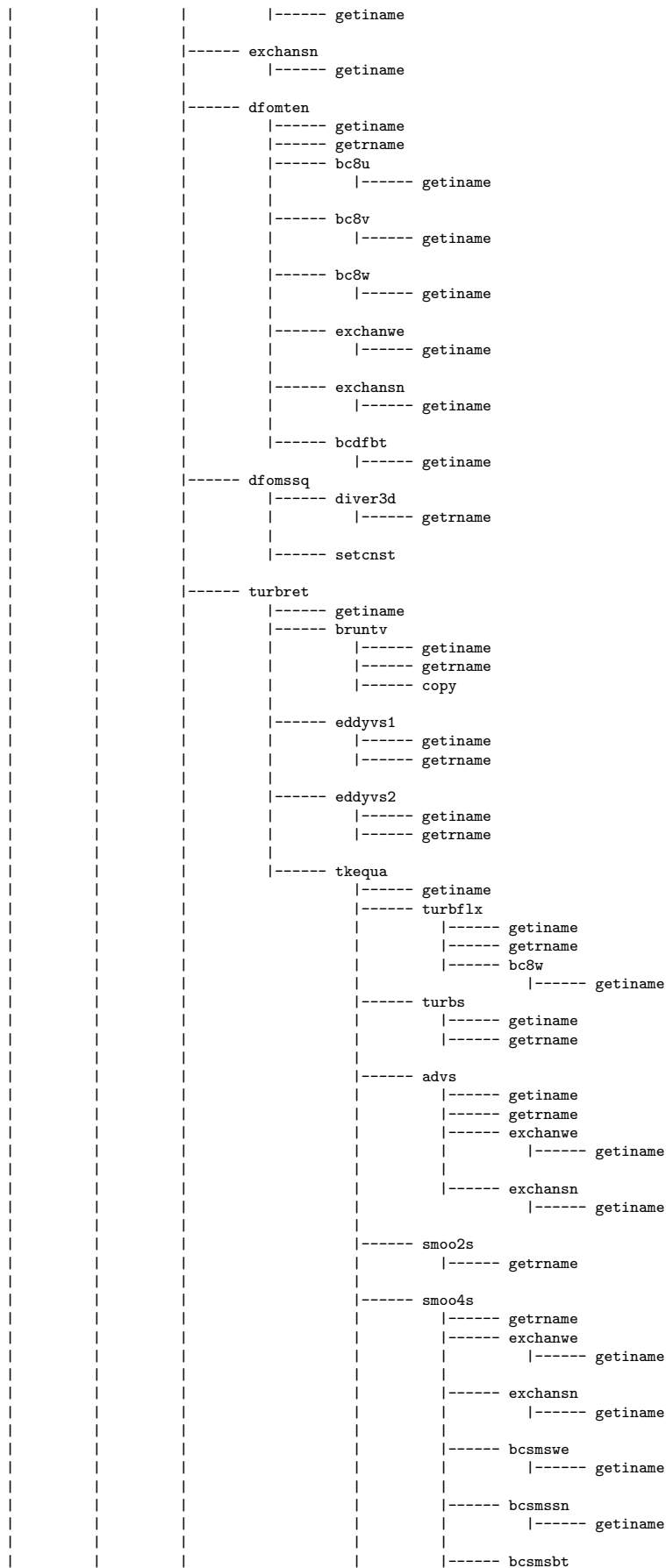


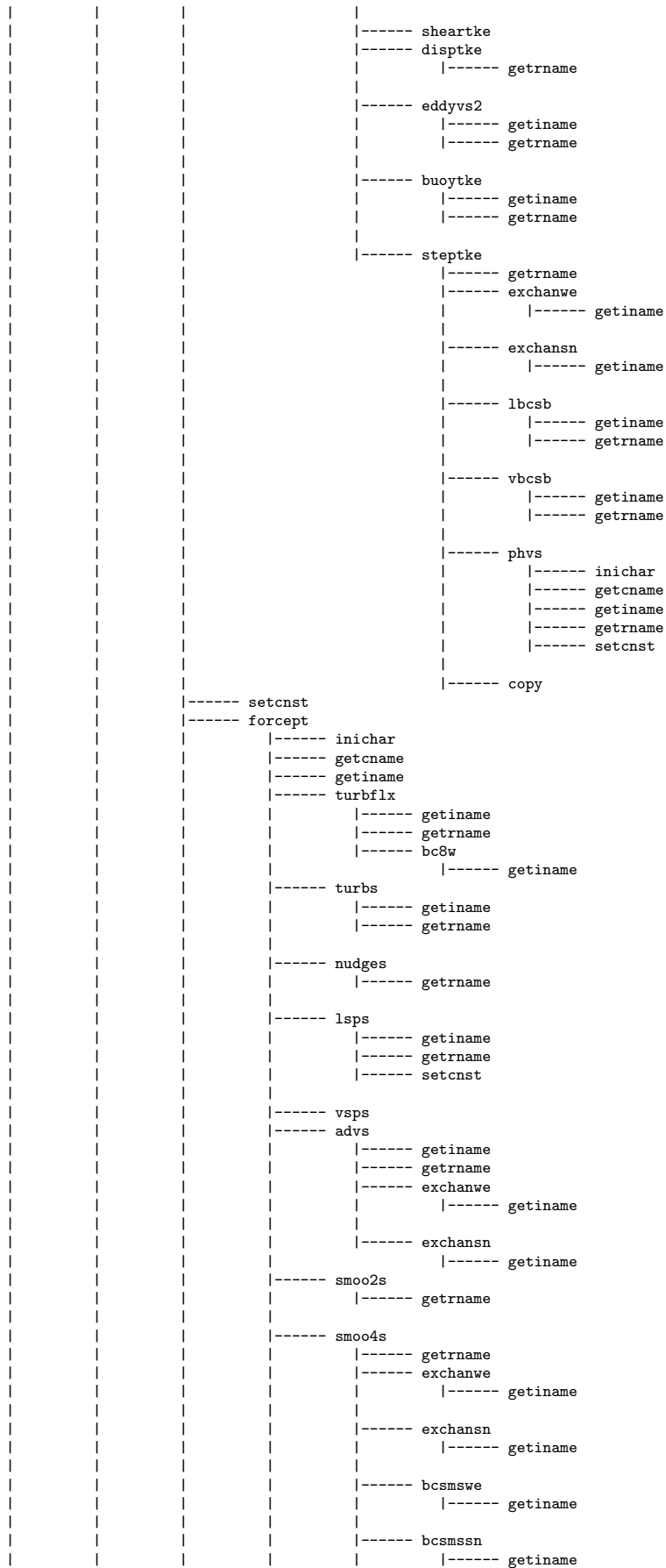


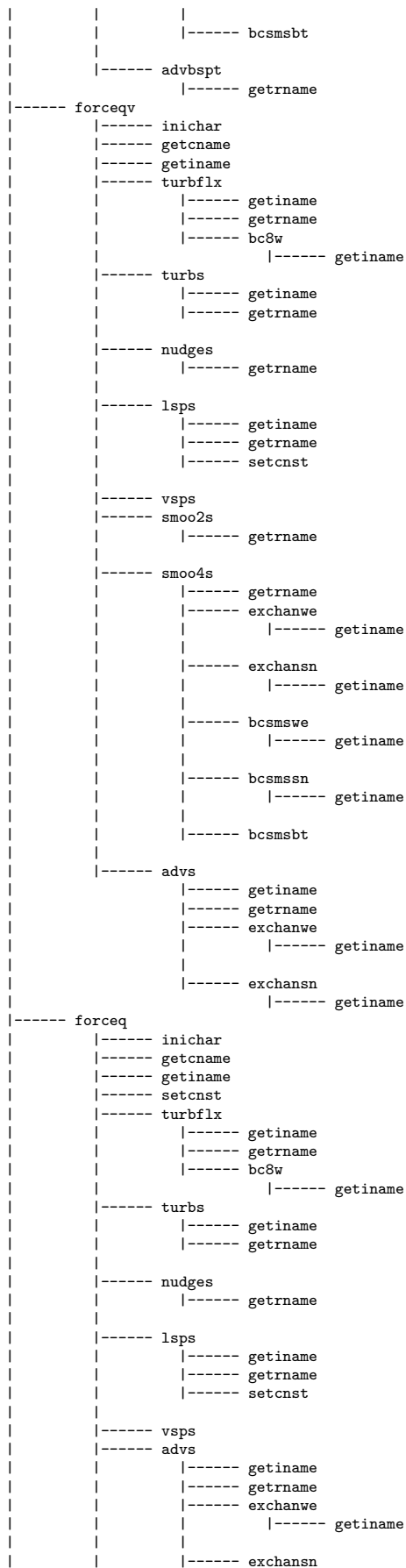




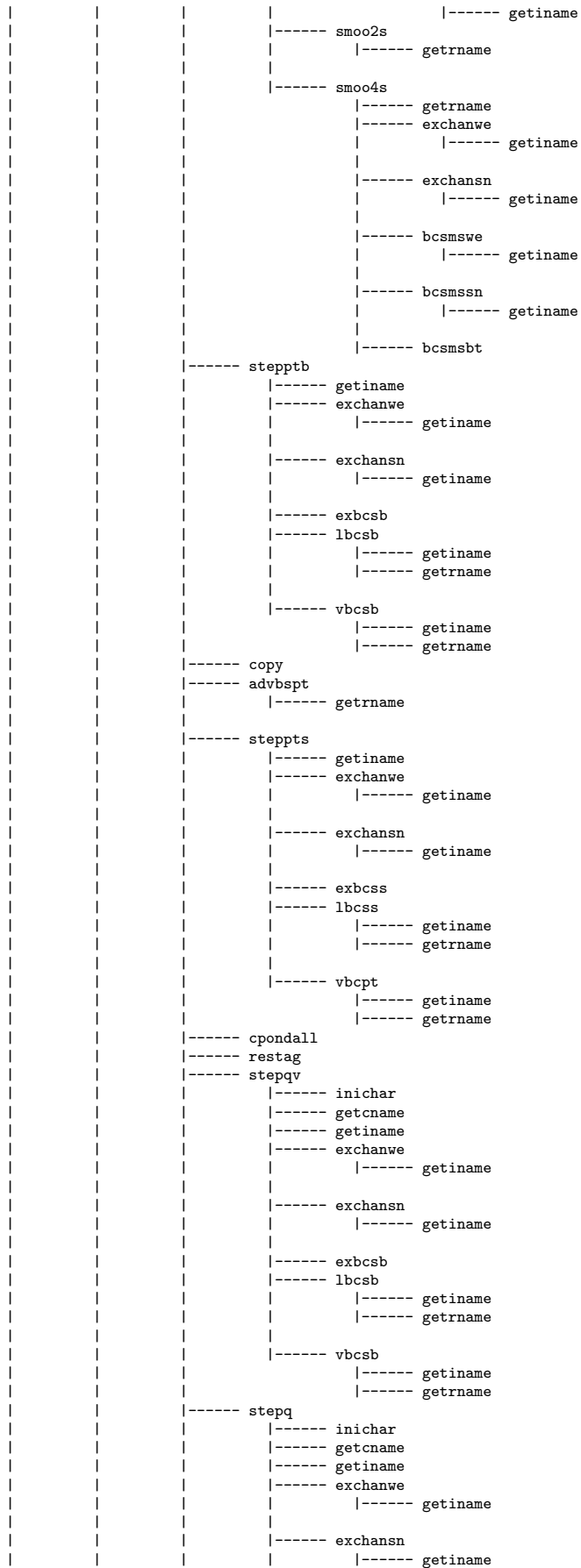












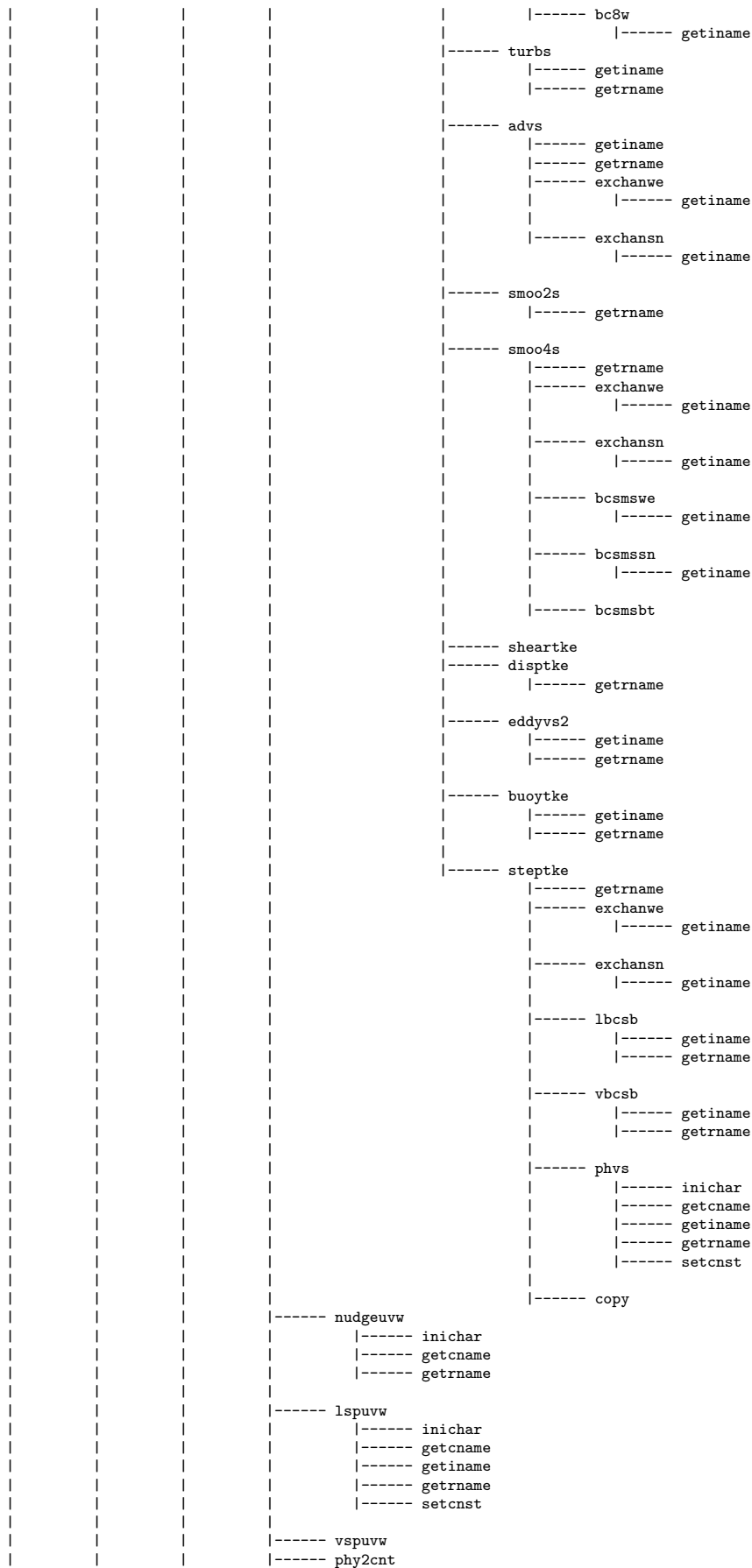
```

|----- exbsb
|----- lbsb
|----- getiname
|----- getrname
|----- vbcsb
|----- getiname
|----- getrname
|----- cloudphy
|----- getiname
|----- kg2g
|----- getiname
|----- getexner
|----- warmrain
|----- adjstq
|----- satadjst
|----- convc2r
|----- collc2r
|----- evapr2v
|----- sedimqr
|----- getiname
|----- chksedim
|----- coldrain
|----- getiname
|----- getsrsc
|----- adjstq
|----- adjstnc
|----- nuclct11
|----- getiname
|----- getrname
|----- satadjst
|----- nuclct12
|----- getiname
|----- getdianc
|----- setcold
|----- collect
|----- getiname
|----- productg
|----- aggregat
|----- getiname
|----- melting
|----- deposit
|----- convers
|----- getiname
|----- nuclct2
|----- freezing
|----- getiname
|----- shedding
|----- pgratio
|----- hndmore0
|----- getiname
|----- setcnst
|----- estimpq
|----- newcold
|----- getiname
|----- sediment
|----- getiname
|----- getrname
|----- chksedim
|----- g2kg
|----- getiname
|----- fltret
|----- getiname
|----- getrname
|----- swpret
|----- getiname
|----- copy
|----- opendmp
|----- getiname
|----- inichar
|----- getcname
|----- getiname
|----- getunit

```



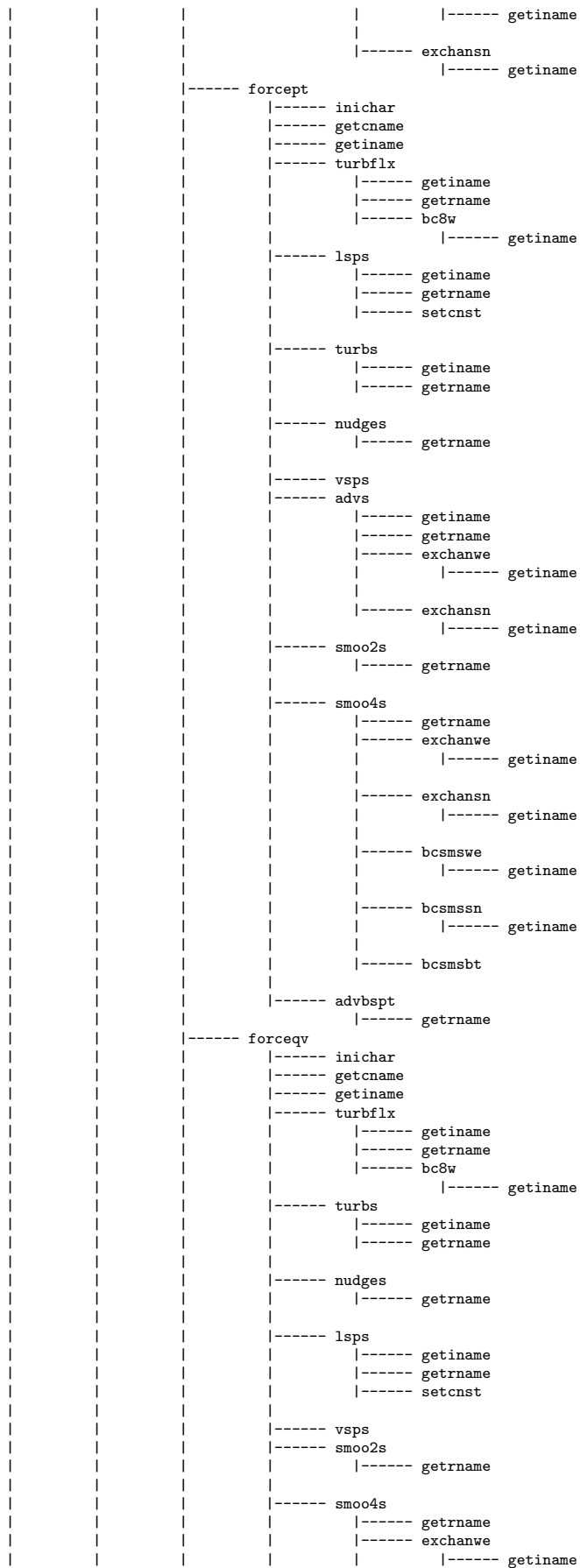




```

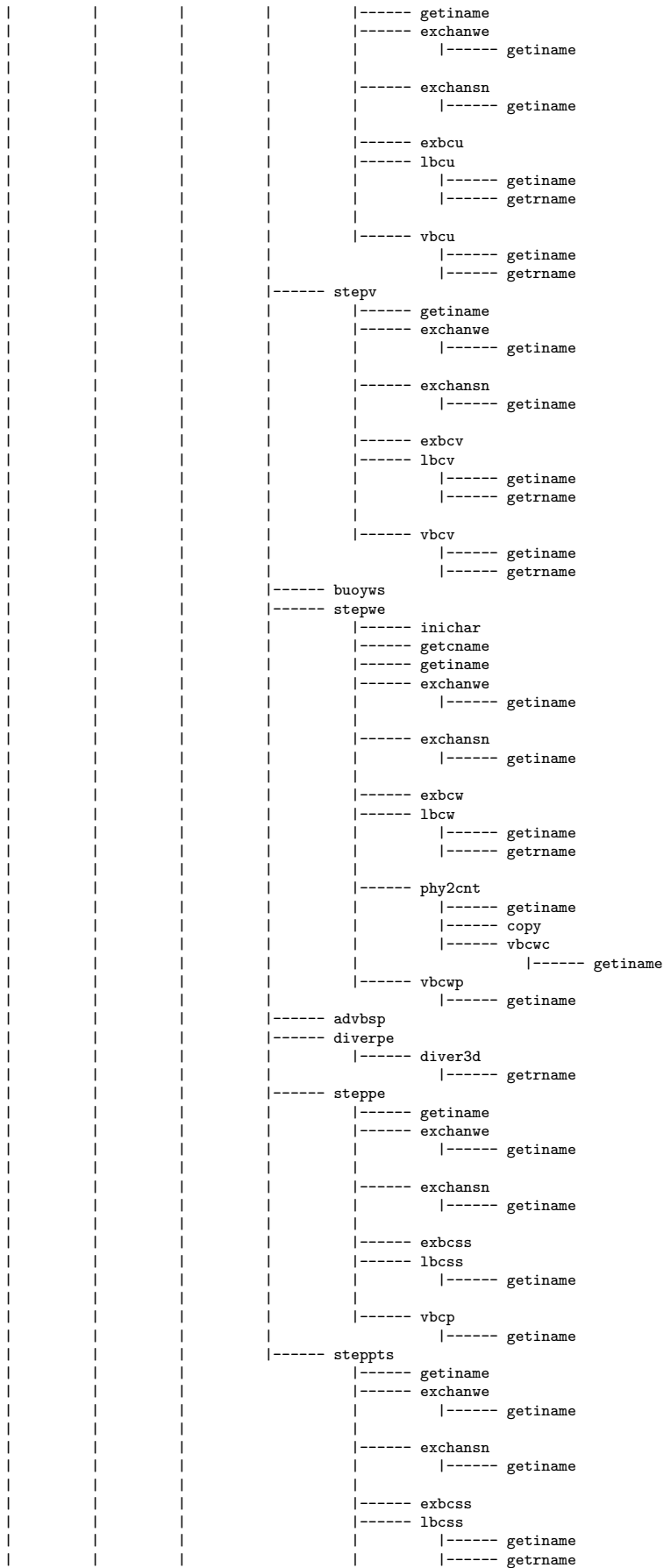
|----- getiname
|----- copy
|----- vbcwc
|----- getiname
|----- advuvw
|----- getiname
|----- getrname
|----- exchanwe
|----- getiname
|----- exchansn
|----- getiname
|----- var8uvw
|----- getiname
|----- exchanwe
|----- getiname
|----- bc8u
|----- getiname
|----- exchansn
|----- getiname
|----- bc8v
|----- getiname
|----- bc8w
|----- getiname
|----- smoo2uvw
|----- getrname
|----- smoo4uvw
|----- getrname
|----- exchanwe
|----- getiname
|----- exchansn
|----- getiname
|----- bcsmuwe
|----- getiname
|----- bcsmusn
|----- getiname
|----- bcsmubt
|----- bcsmvwe
|----- getiname
|----- bcsmvsn
|----- getiname
|----- bcsmvbt
|----- bcsmwwe
|----- getiname
|----- bcsmvsn
|----- getiname
|----- bcsmvbt
|----- getiname
|----- coriuv
|----- coriuvw
|----- buoywb
|----- getiname
|----- setcnst
|----- copy
|----- forcep
|----- inichar
|----- getcname
|----- getiname
|----- setcnst
|----- nudges
|----- getrname
|----- lsp
|----- getiname
|----- getrname
|----- setcnst
|----- advs
|----- getiname
|----- getrname
|----- exchanwe

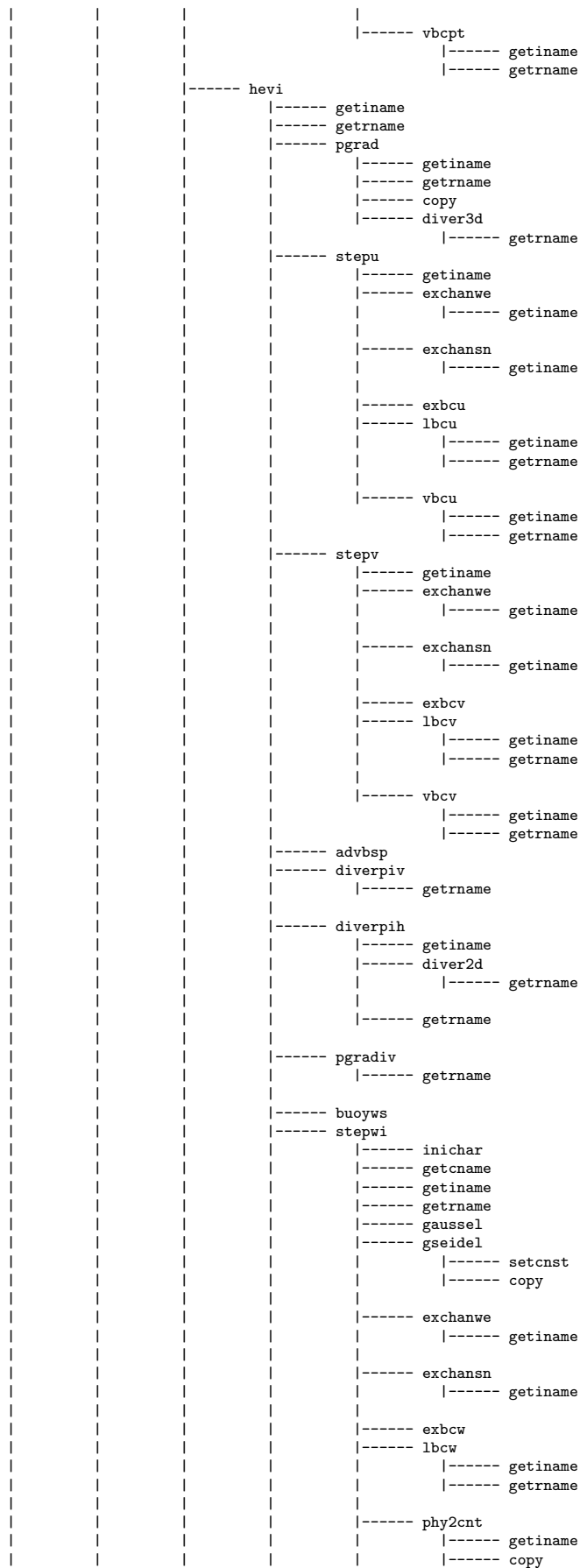
```



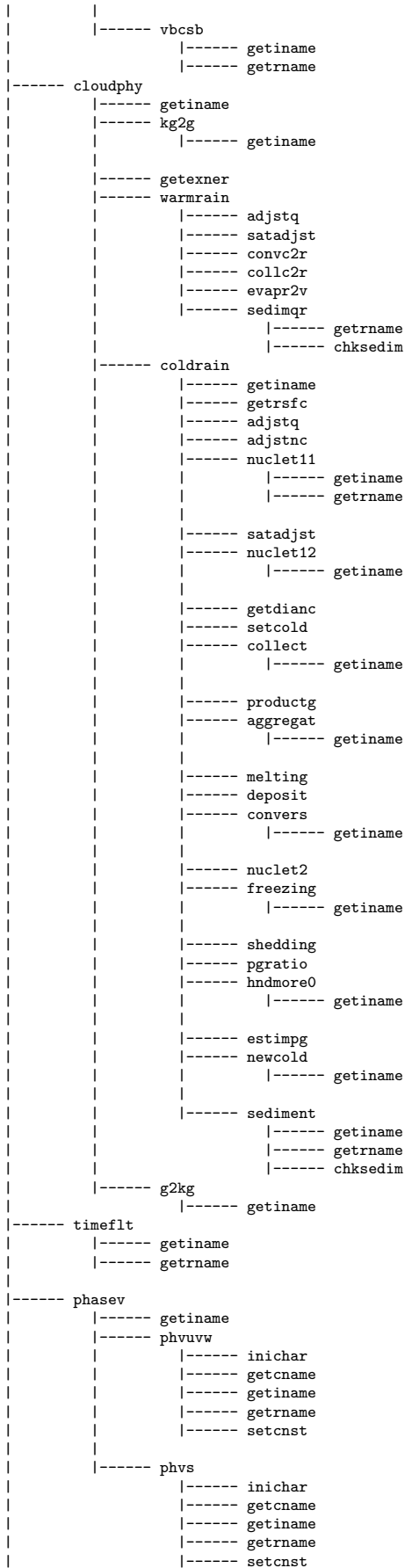


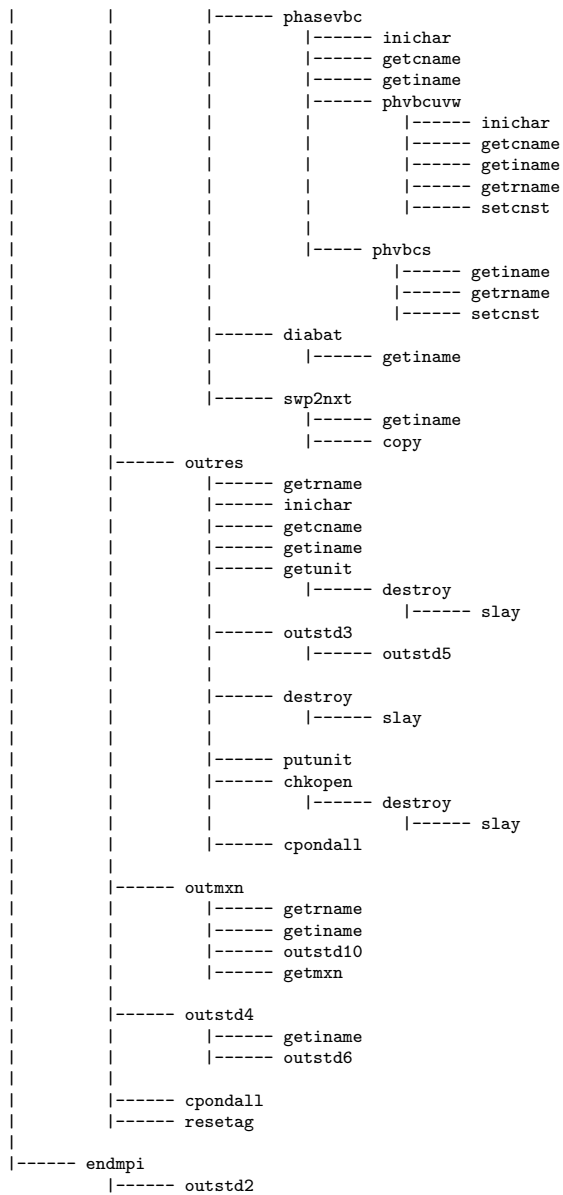




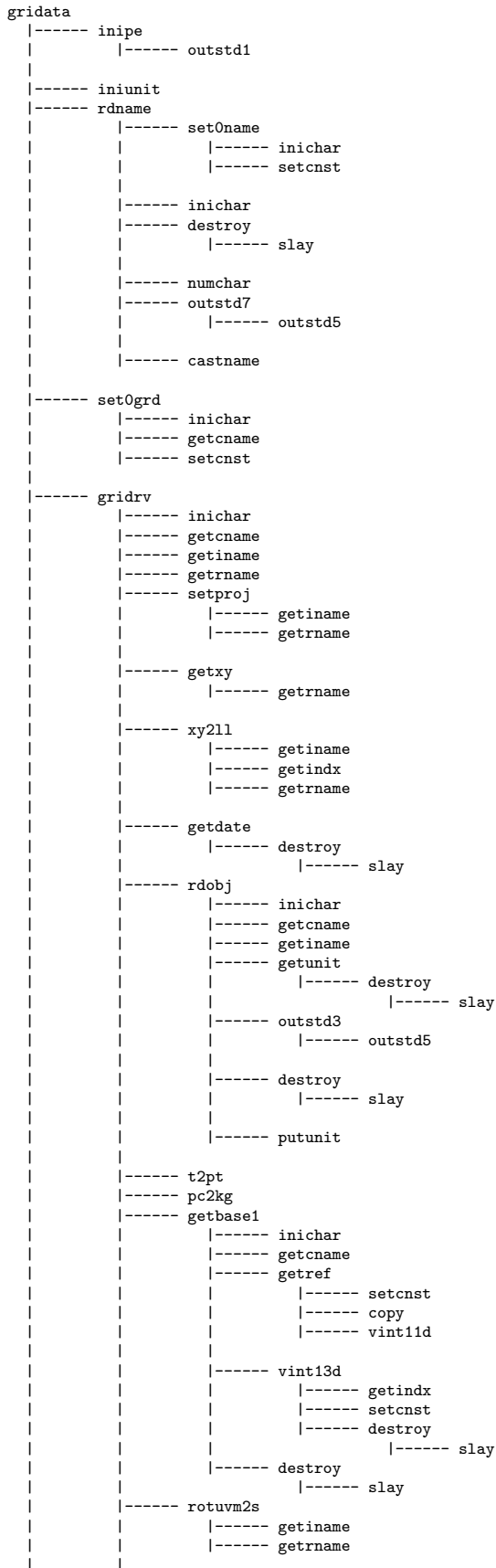


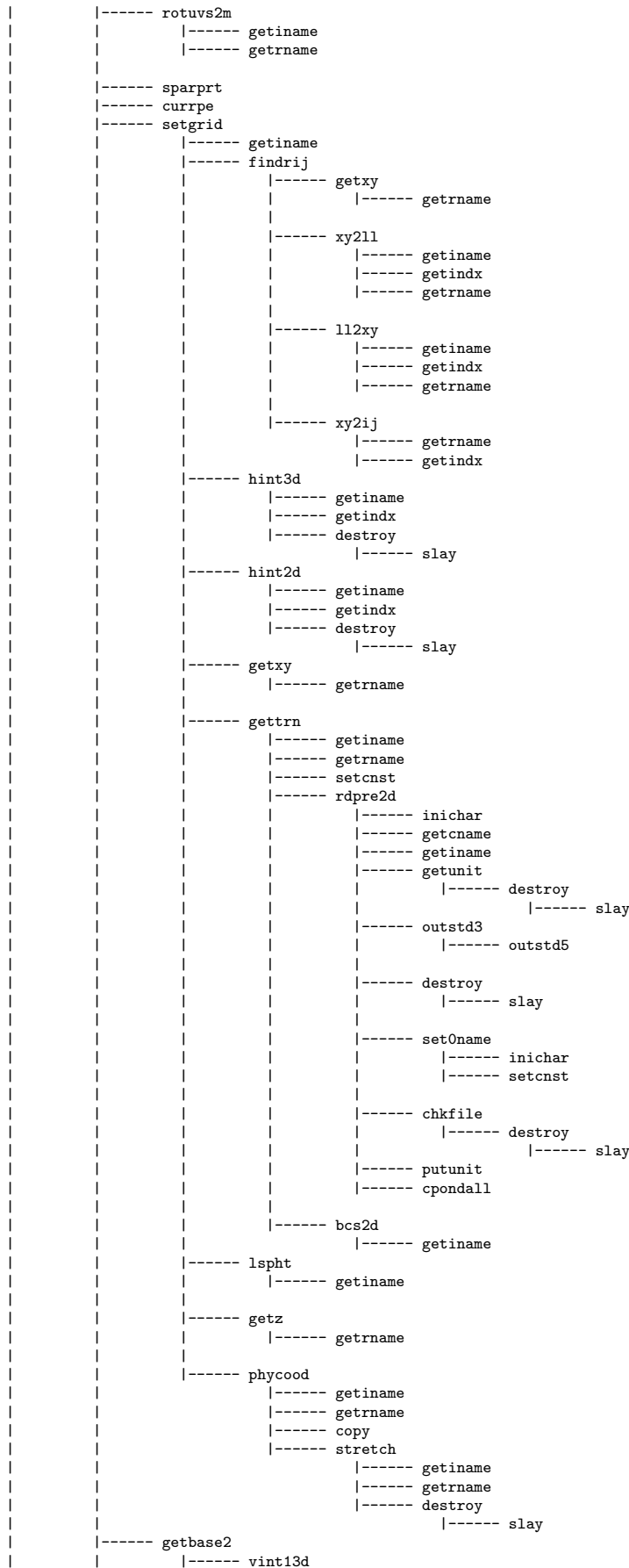






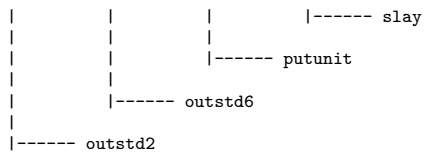
8.2.2 *gridata*



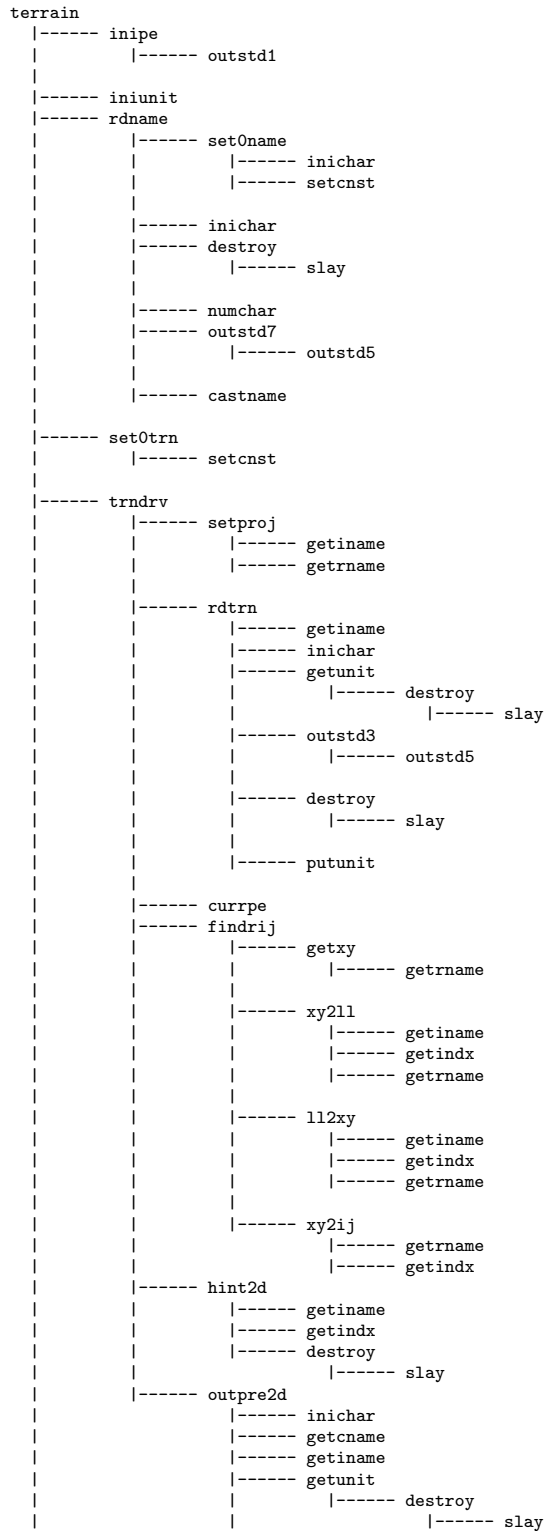


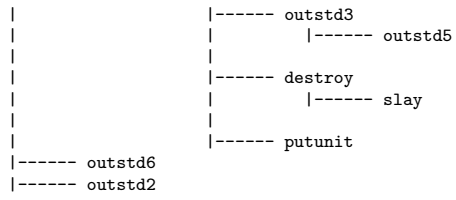




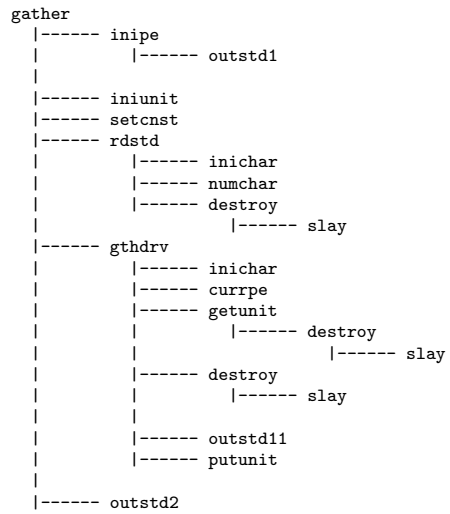


### 8.2.3 *terrain*





### 8.2.4 *gather*



## Chapter 9

# Configuration and Execution

In this chapter, configuration, flowchart of data and execution of *CRess* will be described. Two types of *CRess* are present: one is the serial version for a single processor and the other is the parallel version for multiple processors. This chapter describes mostly about the parallel version of *CRess*. The configuration and execution of the serial version are essentially the same as those of the parallel version and the differences between the parallel and serial versions will be unspecified.

*CRess* has two configuration files: one is `compile.conf` which configures the compile commands and their options, and the other is `user.conf` which controls both compile and run. The former is dependent on the machine to be used while the latter is independent of mostly except `& sysdep`. In this chapter, the latter configuration file will be explained in detail.

The configuration file "user.conf" is written as the format of the Fortran namelist. The user.conf is used for not only run but also the compile. All the dimensions of array arguments are specified by the user.conf. Users, therefore, will never change the source codes of *CRess*.

The flowchart of data will also be described as well as data formats. There are different data flows which depend on the configuration of calculation.

At the last part of this chapter, an example run of experiment of the Kelvin-Helmholtz waves will be shown: compilation of solver, run the solver and output the result. Compilations and executions of the pre-processors will be also presented.

## 9.1 User configuration file

### 9.1.1 Notifications

The user configuration file of *CReSS* is a namelist of Fortran. The followings are important notices to use the file.

- Since the configuration file is used by a shell script in the compilation of *CReSS*, a space is necessary to separate between a variable name and “=” and between a value and “=”.
- The solver and the pre-processors have dependency of some namelist variables. When a discrepancy of variables between pre-processors and the solver is found, the program will stop execution and output an error.

### 9.1.2 Details of the user configuration file

#### & sysdep

The namelist variable is depend on machines. All programs use the variable, while no dependence is present between the programs.

---

<code>wlength</code>	<i>integer * 4</i> Word length of binary data in the direct access and unformatted. Most machines take <code>wlength</code> as 4, while some others take 1 (eg. Digital UNIX, NEC SX6).
----------------------	--

---

#### & dmiset

This namelist variables set numbers of grid points of calculation domain and numbers of decomposition in parallel computing. All programs use the variables and have dependency. (**Recompiling is necessary if changed.**)

---

<code>xpedim, ypedim</code>	<i>integer * 4</i> Numbers of decomposition of the calculation domain in $x$ and $y$ directions. The total numbers of processing elements (CPUs) are $x \times y$ .
-----------------------------	--

<code>xdim, ydim</code>	<i>integer * 4</i> Numbers of grid points of the domain in the $x$ and $y$ directions, respectively. Since the horizontal grid system is the Arakawa C type, the numbers of the scalar grid points are $xdim - 1$ and $ydim - 1$ in the $x$ and $y$ directions, respectively. These numbers must satisfy the following relations in order to each processing element is loaded with the same number of grid points.
-------------------------	--

$$\begin{aligned} [xdim + 3(xpedim - 1)] / xpedim &= integer \\ [ydim + 3(ypedim - 1)] / ypedim &= integer \end{aligned}$$

These relations means that numbers of grid points of the physical domain ( $xdim - 3, ydim - 3$ ) must be divided by each number of decomposition.

<code>zdim</code>	<i>integer * 4</i> Number of grid point in the $z$ direction. Since the vertical grid system is the Lorenz type, the number of scalar grid points is $zdim - 1$ . Using the parameters of <code>cphopt = 2</code> or <code>cphopt = 3</code> , <code>zdim</code> must be larger or equal 12 because of use of temporary array.
-------------------	---

---

**& expname**

The namelist specifies the name of the experiment, which is used by *solver*, *gridata*, *terrain*. Dependency is present between these programs.

---

<b>exprim</b>	<i>character * 80</i> Name of numerical experiment. The number of character used in the variable must be less than or equal to 80 characters. The variable is used in input and output data file names. For example, the sounding data file will be <i>exprim.sounding.txt</i> . No special characters are allowed to be used.
---------------	---

---

**& project**

The namelists define the horizontal coordinate systems of the calculation domain. They are used *solver*, *gridata*, and *terrain*. Dependency is present between these programs.

---

<b>mpopt</b>	<i>integer * 4</i> Option for map projection of the domain. The following four projections are available. 1: polar stereo graphic projection 2: Lambert conformal projection 3: Mercator projection 4: No map projection (latitude and longitude)
<b>nspol</b>	<i>integer * 4</i> Switch to define origin of the coordinate in the northern or southern hemisphere. 1: the origin of coordinate in the northern hemisphere. -1: the origin of coordinate in the southern hemisphere.
<b>tlat1,tlat2</b>	<i>real * 4</i> The true latitudes of map projections. No deformation due to projection occurs at the latitudes. The unit is degree [°] and they are negative in the southern hemisphere. If the Lambert conformal projection is used, two true latitude ( <b>tlat1</b> and <b>tlat2</b> ) are necessary. If no map projection is used, they are not necessary to be specified.
<b>tlon</b>	<i>real * 4</i> The true longitude of coordinate. The unit is degree [°] and degree in the West Longitude is negative. The y-axis of the model domain corresponds to the longitude. Even in the case of <b>mpopt</b> = 4, <b>tlon</b> must be specified.

---

**& gridset**

The namelists determine grid intervals and reference longitude and latitude. They are used in *solver*, *gridata*, and *terrain*. Dependency is present between these programs.

---

<b>dx,dy,dz</b>	<i>real * 4</i> Grid intervals in the <i>x,y</i> and <i>z</i> directions, respectively. The unit is meter. If non-uniform vertical grid is used ( <b>sthopt</b> = 1,2) , <b>dz</b> is the averaged grid interval.
<b>ulat,ulon</b>	<i>real * 4</i>

Reference longitude and latitude, respectively, which corresponds to a single point within the domain. The unit is degree. Degrees in the southern hemisphere and in the West Longitude are negative. *CReSS* specifies the region of the domain using the reference longitude and latitude with the following real number of `riu` and verb "rju".

`riu,rju`      *real \* 4*  
The real number of the reference point which corresponds to the point determined by the reference longitude and latitude (`ulat,ulon`).

---

### & gridsth

The namelists determine vertical grid spacings in non-uniform coordinate. They are used in *solver*, *gridata*. Dependency is present between these programs.

`zsfc`      *real \* 4*  
Altitude of the surface. The unit is meter. In most cases, `zsfc = 0.0` m.

`zflat`      *real \* 4*  
The lowest height of the flat level. Unit is meter.

`sthopt`      *integer \* 4*  
Option for vertical stretching.  
0: No stretching (uniform grid interval).  
1: Stretching using a cubic function.  
2: Stretching using a function of tanh.

`dzmin`      *real \* 4*  
Minimum grid interval at the lowest level. Unit is meter.

`layer1,layer2`      *real \* 4*  
In stretching, the grid interval of `dzmin` is used up to a height of `layer1`, the intervals determined by functions between heights of `layer1` and `layer2`, and the grid interval at a height of `layer2` is used above the height of `layer2`. Unit is meter.

---

### & terrain

The namelists specify terrain topography of the model. They are used in the program of *solver* and *gridata*. Dependency is present between these programs.

`trnopt`      *integer \* 4*  
Option for model terrain.  
0: the flat ground.  
1: the bell-shaped mountain.  
2: user specified terrain from the external data file.

`mntgh`      *real \* 4*  
Height (m) of the summit of the bell-shaped mountain, if `trnopt = 1`

`mntwx,mntwy`      *real \* 4*  
Half width (distance from the center of a half height) of the bell-shaped mountain in the *x* and *y* directions, respectively, if `trnopt = 1`. Unit is meter.

---

**mntcx, mntcy**      *real \* 4*  
 The coordinates  $x, y$  of the center of the bell-shaped mountain, if `trnopt = 1`. Unit is meter. The relationship between the coordinates and their array number  $i, j$  is as follows.

$$x = (i - 2) \times dx$$

$$y = (j - 2) \times dy$$


---

### & flength

The namelists are with respect to time of calculation. They are used in *solver* and *gridata*. The parameters with \* have dependency between these programs.

---

**sfcast\***            *character \* 16*  
 Date and time of the initial time of calculation in the Universal Time Coordinate (UTC). The format is `sfcast = 'yyyy/mm/dd hh:mm'`. For example, when the initial time is 13:00 UTC, 22 September 2000, `sfcast = '2000/09/22 13:00'`.

**stime**             *real \* 4*  
 Initial time of calculation. Unit is second. The time is measured from the time of `sfcast`. In the first calculation, `stime = 0.0`, and In a restart, `stime` is must be the restart time.

**etime**             *real \* 4*  
 Time of the termination of calculation. Unit is second. The time is also measured from the time of `sfcast`. For example, `etime` must be 5400.0 to calculate 1800.0 seconds from 3600.0 seconds.

---

### & extpram

The namelists are parameters of external forcing using an external data. They are used in *solver* and *gridata*. The parameters with \* have dependency between these programs.

---

**extvar\***            *character \* 80*  
 Specify variables to be inputted to the model from an external data. The first 7 characters of 80 are used. Each character from the first corresponds to the vertical velocity component, water vapor mixing ratio (or relative humidity), mixing ratios of cloud water, rain, cloud ice, snow and graupel. On the other hand,  $z$ -coordinate, horizontal velocity components, pressure, potential temperature (or temperature) are indispensable and included without this specification. For example, the vertical velocity component, water vapor mixing ratio (or relative humidity) and mixing ratio of rain are used (when they are available), then `extvar = 'ooxoxxx'`

o: the variable is inputted.  
 x: the variable is not inputted.

**extitv\***            *real \* 4*  
 Time interval of the external data. Unit is second.

**ndgopt**            *integer \* 4*  
 Option for nudging of an analysis data.

0: no nudging.  
 1: perform the nudging.

ndgvar	<p><i>character * 80</i></p> <p>Specify variables to be used in nudging. The first 11 characters of 80 are used. Each character from the first corresponds to u and v-components of velocity, vertical velocity component, pressure, potential temperature (or temperature), water vapor mixing ratio (or relative humidity), mixing ratios of cloud water, rain, cloud ice, snow and graupel. For example, when all the three components of velocity, pressure and potential temperature are used for nudging, <code>ndgvar = 'ooooooooxxxx'</code>. All variables besides the indispensable variables must be specified by <code>extvar</code>.</p> <ul style="list-style-type: none"> <li>o: the variable is used in nudging.</li> <li>x: the variable is not used in nudging.</li> </ul>
ndgcoe	<p><i>real * 4</i></p> <p>Coefficient of dumping (e-folding time of dumping) of the nudging of the analysis data.</p>
ngdelt	<p><i>real * 4</i></p> <p>Time interval of nudging. Unit is second. The parameter must be divided by <code>dtb</code> and larger than or equal to <code>dtb</code>.</p>
ngtime	<p><i>real * 4</i></p> <p>Time to terminate the nudging (measured from the initial). Unit is second.</p>
exbopt*	<p><i>integer * 4</i></p> <p>Option for forcing to external data at the lateral boundary. The forcing is performed with respect to the indispensable variables and variables determined by <code>extvar</code>.</p> <ul style="list-style-type: none"> <li>0: no forcing at the lateral boundary.</li> <li>1: forcing to the external data at the lateral boundary is performed.</li> <li>2: forcing to the external data at the lateral boundary is performed and the terrain is also smoothed to be the external terrain in the lateral sponge layer.</li> <li>3: forcing to the sounding data at the lateral boundary is performed.</li> </ul>
lspvar	<p><i>character * 80</i></p> <p>Specify variables to be forced in the lateral sponge layer. Variables besides the indispensable variables must be determined <code>extvar</code>. Specification is the same as <code>ndgvar</code>.</p>
lspspn*	<p><i>integer * 4</i></p> <p>Number of the lateral sponge layer. Dependency is present in the case of <code>refsfc = 1</code>.</p>
lspcoe	<p><i>real * 4</i></p> <p>Coefficient of the lateral forcing in the lateral sponge layer. The forcing decrease with the distance from the lateral boundary.</p>
vspopt	<p><i>integer * 4</i></p> <p>Option for dumping in the top sponge layer.</p> <ul style="list-style-type: none"> <li>0: no dumping in the top sponge layer.</li> <li>1: dumping in the top sponge layer is performed. The variables are forced to be those at the initial time.</li> </ul>
vspcoe	<p><i>real * 4</i></p> <p>Coefficient of the dumping in the top sponge layer. The forcing decreases with the distance from the top of the model.</p>
zsplow	<p><i>real * 4</i></p>



---

The lowest height of the upper dumping. Unit is meter.

---

### & boundary

The namelists determine the lateral boundary condition. They are used in *solver* and *gridata*. The parameters with \* have dependency between these programs.

---

wbc*, ebc*, sbc*, nbc*	<i>integer * 4</i> Option for lateral boundary conditions at the west, east, south and north boundaries, respectively. 1: periodic boundary condition. 2: rigid wall boundary condition. 3: zero-gradient boundary condition. 4: wave-radiating boundary condition. 5: wave-radiating boundary condition using vertically averaged phase velocity of normal velocity component.
bbc, tbc	<i>integer * 4</i> Option for bottom and top boundary condition, respectively. If <code>impopt = 1</code> , then these parameters are forced to be 3. 2: rigid wall boundary condition. 3: zero-gradient boundary condition. 4: wave-radiating boundary condition at the top boundary condition.

---

### & sfcphys

The namelists determine the surface conditions and the ground model. (**Recompiling is necessary if changed.**)

---

sfcopt	<i>integer * 4</i> Option for the surface process. 0: no surface process. 1: surface process is performed.
sfcdat*	<i>character * 80</i> Specify the external data of surface process. The first 5 characters of 80 are used. Each character from the first corresponds to land use, land surface temperature, sea surface temperature, surface temperature of sea ice, and surface temperature of snow. For example, when the land use and the land surface temperature are available, then, <code>sfcdat = 'oxxxx'</code> . o: the variable is available and will be used. x: the variable is not used.
levpbl	<i>integer * 4</i> Number of the planetary boundary layer. The thickness of each layer is $dz$ and $1 \leq \text{levpbl} \leq \text{zdim} - 2$ .
levgrd	<i>integer * 4</i> Number of layers of the ground model.
dzgrd	<i>real * 4</i> Thickness of the layers of the ground model. Unit is meter. The total depth of the ground model is desirable to be larger than a few meters.

---

<code>cbeta</code>	<i>real * 4</i> A constant evapotranspiration coefficient (moisture availability form the surface) used in the case of no land use data.
<code>calbe</code>	<i>real * 4</i> A constant albedo used in the case of no land use data.
<code>cz0m</code>	<i>real * 4</i> A constant roughness over the land used in the case of no land use data. Unit is meter.
<code>ctdeep</code>	<i>real * 4</i> A constant ground temperature at the deepest layer used in the case of no ground temperature data. Unit is K.
<code>ctsea</code>	<i>real * 4</i> A constant sea surface temperature used in case of no SST data. Unit is K.

---

**& iniotype**

The namelists determine the initialization of model. Only in *solver*. (**Recompiling is necessary if changed.**)

---

<code>iniopt</code>	<i>integer * 4</i> Option for the model initialization. 1: read the sounding data. 2: read the restart file. 3: read 3-dim non-uniform grid data. 4: In addition to 3, retrieval is performed with respect to the prognostic variables.
<code>snddim</code>	<i>integer * 4</i> Number of levels of the sounding data if <code>iniopt = 1</code> .
<code>sndhed</code>	<i>integer * 4</i> Number of comments in the header of the sounding data if <code>iniopt = 1</code> . The comment lines must be at the beginning of the sounding file.
<code>sndtyp</code>	<i>character * 80</i> Specify data type of the sounding if <code>iniopt = 1</code> . The following combinations are available except the horizontal velocity. <p> <code>ppk</code>: 1st pressure [Pa], 2nd <math>\theta</math> [K], 5th <math>q_v</math> [kg kg<sup>-1</sup>]  <code>zpk</code>: 1st height [m], 2nd <math>\theta</math> [K], 5th <math>q_v</math> [kg kg<sup>-1</sup>]  <code>ptk</code>: 1st pressure [Pa], 2nd temperature [K], 5th <math>q_v</math> [kg kg<sup>-1</sup>]  <code>ztk</code>: 1st height [m], 2nd temperature [K], 5th <math>q_v</math> [kg kg<sup>-1</sup>]  <code>ppp</code>: 1st pressure [Pa], 2nd <math>\theta</math> [K], 5th <math>Rh</math> [%]  <code>zpp</code>: 1st height [m], 2nd <math>\theta</math> [K], 5th <math>Rh</math> [%]  <code>ptp</code>: 1st pressure [Pa], 2nd temperature [K], 5th <math>Rh</math> [%]  <code>ztp</code>: 1st height [m], 2nd temperature [K], 5th <math>Rh</math> [%] </p>
<code>zsnd0,psnd0</code>	<i>real * 4</i> Height [m] and pressure [Pa] of the lowest level of the sounding data, respectively. If <code>sndtyp(1:1) = 'p'</code> , then <code>zsnd0</code> is used and if <code>sndtyp(1:1) = 'z'</code> , then <code>psnd0</code> is used.

---

<b>retvar</b>	<i>character * 80</i> Specify variables to be performed retrieval. First 3 characters of 80 are used. Each character corresponds to the 3-dim. velocity components, pressure, and potential temperature and mixing ratio of hydrometeors. For example, if the retrieval is performed with respect to the 3-dim. velocity components and pressure, then <b>retvar = 'oox'</b> . o: retrieval is performed. x: no retrieval is performed.
<b>retime</b>	<i>real * 4</i> Integration time of retrieval with respect to the potential temperature and mixing ratios of hydrometeors.
<b>alpha1,alpha2</b>	<i>real * 4</i> Weighting coefficients of horizontal and vertical directions using the static stability if <b>iniopt = 4</b> .

---

**& gridmove**

The namelists configure the horizontal displacement of the calculation domain and are used only in *solver*.

---

<b>movopt</b>	<i>integer * 4</i> Option for the displacement of the domain. This option is valid when <b>trnopt = 0</b> , <b>iniopt = 1</b> and <b>exbopt = 0</b> . 0: No displacement of the domain. 1: Displacement of the domain is switched on.
<b>umove,vmove</b>	<i>real * 4</i> Speeds of the displacement of the domain in the <i>x</i> and <i>y</i> directions, respectively. Unit is [m s <sup>-1</sup> ].

---

**& ptinicon**

The namelists set perturbations of the potential temperature at the initial time and are used only in *solver*.

---

<b>pt0opt</b>	<i>integer * 4</i> Option for perturbations of potential temperature at the initial time. This parameter is valid when <b>iniopt = 1</b> . 0: No initial perturbation of potential temperature. 1: One or more elliptic perturbations are placed in the <i>x</i> direction. 2: One or more elliptic perturbations are placed in the <i>y</i> direction. 3: Perturbation of trigonometric function in the <i>x</i> direction is set. 4: Perturbation of trigonometric function in the <i>y</i> direction is set. 5: Random perturbations are generated between the specified two levels.
<b>pt0num</b>	<i>integer * 4</i> Number of the elliptic perturbation of potential temperature when <b>pt0opt = 1</b> or <b>2</b> .
<b>ptp0</b>	<i>real * 4</i> Maximum potential perturbation at the initial time . Unit is Kelvin.
<b>pt0rx,pt0ry,</b>	<i>real * 4</i>

---

<code>pt0rz</code>	<i>real</i> * 4 Radii of the elliptic perturbations, or half wave length of the trigonometric function or half width of the random perturbation layer in $z$ direction ( <code>pt0opt = 5</code> ). Unit is meter.
<code>pt0cx,pt0cy, pt0cz</code>	<i>real</i> * 4 Center of the elliptic perturbations, or the origin of the trigonometric function or middle height of the random perturbation layer in $z$ direction ( <code>pt0opt = 5</code> ). Unit is meter.
<code>pt0ds</code>	<i>real</i> * 4 Distance between bubbles of potential temperature in the case of <code>pt0opt = 1</code> or <code>2</code> . Unit is meter.

---

**& integrat**

The namelists define the parameters of time integration such as time intervals and coefficients of the Asselin's time filter. Only in *solver*.

---

<code>dtbig,dtsml</code>	<i>real</i> * 4 Time intervals of the large time step and the small time step, respectively. The terms related with the acoustic waves are integrated in the small time step and others in the large time step. If <code>gwmopt = 1</code> , then the terms of the gravity waves are integrated in the small time step. Unit is second. These time intervals must satisfy the CFL condition.
<code>gwmopt</code>	<i>integer</i> * 4 Option for the time integration of the terms related with the gravity waves. 0: They are integrated in the large time step. 1: They are integrated in the small time step.
<code>impopt</code>	<i>integer</i> * 4 Option for time integration scheme of the acoustic wave mode in the vertical direction. <code>impopt = 1</code> is recommended. 0: Explicit scheme. 1: Implicit scheme using Gaussian elimination. 2: Implicit scheme using Gaussian elimination with pivot option. 3: Implicit scheme using Gauss-Zaidel method.
<code>weicoe</code>	<i>real</i> * 4 Weighting coefficient of the implicit scheme.
<code>filcoe</code>	<i>real</i> * 4 Weighting coefficient of the Asselin's time filter.

---

**& advction**

The namelist is the option of the advection scheme. Only in *solver*.

---

<code>ad4opt</code>	<i>integer</i> * 4 Option for the advection scheme. The center difference is used in the advection scheme. The numerical smoothing is, therefore, necessary to suppress the computational instability.
---------------------	---

- 
- 0: the second order center difference.
  - 1: the fourth order center difference.
- 

### & smoother

Configurations of the numerical smoothing are determined in the namelists. Only in *solver*.

---

<b>smtopt</b>	<i>integer * 4</i> Option for the numerical smoothing which works as an artificial viscosity to suppress the numerical noise of the advection scheme. <ul style="list-style-type: none"> <li>0: No numerical smoothing.</li> <li>1: The second order numerical smoothing.</li> <li>2: The fourth order numerical smoothing.</li> </ul>
<b>smndch</b>	<i>real * 4</i> Non-dimensional coefficient of the artificial viscosity in the horizontal direction. Recommended that <b>smndch/dtb</b> = 0.001. The coefficients of viscosity $\nu_{2h}, \nu_{4h}$ are defined as <ul style="list-style-type: none"> <li>Second order: <math>\nu_{2h} = \text{smndch} \times (\text{dxdy}) / \text{dtb}</math></li> <li>Fourth order: <math>\nu_{4h} = \text{smndch} \times (\text{dxdy})^2 / \text{dtb}</math></li> </ul>
<b>smndcv</b>	<i>real * 4</i> Non-dimensional coefficient of the artificial viscosity in the vertical direction. Recommended that <b>smndcv/dtb</b> = 0.001. The coefficients of viscosity $\nu_{2v}, \nu_{4v}$ are defined as <ul style="list-style-type: none"> <li>Second order: <math>\nu_{2v} = \text{smndcv} \times \text{dz}^2 / \text{dtb}</math></li> <li>Fourth order: <math>\nu_{4v} = \text{smndcv} \times \text{dz}^4 / \text{dtb}</math></li> </ul>

---

### & mapfcter

The namelist is an option for map factors. Only in *solver*. Map factors are not implemented in the version of 1.3 or earlier.

---

<b>mfcopt</b>	<i>integer * 4</i> Option for map factor. <ul style="list-style-type: none"> <li>0: No map factor is used.</li> <li>1: Map factor is used.</li> </ul>
---------------	--

---

### & coriolis

Configuration of the Coriolis force terms. Only in *solver*.

---

<b>coropt</b>	<i>integer * 4</i> Option for the Coriolis force terms. <ul style="list-style-type: none"> <li>0: No Coriolis force terms.</li> <li>1: Horizontal components of the Coriolis force are calculated.</li> <li>2: Both horizontal and vertical components of the Coriolis force are calculated.</li> </ul>
---------------	--

---

**& buoyancy**

Switching the buoyancy term of pressure perturbation in the vertical component of the momentum equation. This term is related to the acoustic waves. Only in *solver*.

---

**buyopt**            *integer \* 4*  
Option for the buoyancy term.  
0: The buoyancy is omitted.  
1: The buoyancy is included.

---

**& diabatic**

Switching the diabatic term in the pressure equation. Only in *solver*.

---

**diaopt**            *integer \* 4*  
Option for the diabatic term.  
0: The diabatic term is omitted.  
1: The diabatic term is included.

---

**& ddamping**

Configuration of the divergence dumping of the pressure gradient force in the momentum equations. Only in *solver*.

---

**divopt**            *integer \* 4*  
Option for the divergence dumping of the pressure gradient force, which is an artificial term to suppress the numerical instability.  
0: No divergence dumping is performed.  
1: Divergence dumping is performed.

**divndc**            *real \* 4*  
Non-dimensional dumping coefficient. The recommended value is 0.05.

---

**& turbulent**

The sub-grid scale turbulence process is configured in the namelist. Only in *solver*.

---

<b>tubopt</b>	<i>integer * 4</i> Option for the sub-grid turbulence process. 0: No turbulence process is performed. 1: The Smagorinsky scheme (1 order closure). 2: The 1.5 order closure with TKE.
<b>isoopt</b>	<i>integer * 4</i> Option for directionality of the grid. 1: Grid is directional in horizontal and vertical directions. 2: Grid is non-directional in horizontal and vertical directions.
<b>prnum</b>	<i>real * 4</i> Turbulent Prandtl number ( $Pr = \nu_\tau / \nu_H$ ). It is used only for stability check when <b>tubopt</b> = 2.
<b>csnum</b>	<i>real * 4</i> The Smagorinsky constant when <b>tubopt</b> = 1.

---

**& cloudphy**

Configuration for the cloud microphysics. Only in *solver*. (**Recompiling is necessary if changed.**)

---

<b>cphopt</b>	<i>integer * 4</i> Option for the cloud microphysics. 0: No cloud microphysics (the dry model). 1: The bulk warm rain parameterization. 2: The bulk cold rain parameterization. 3: The bulk cold rain parameterization with solving the tendency equation of number densities of ice phase hydrometeors (ice, snow and graupel).
---------------	---

---

**& outfomat**

The namelists determine the format and time intervals of outputs. Only in *solver*.

---

<b>dmpfmt</b>	<i>integer * 4</i> Option for the output format of the history and geographic files. 1: ASCII text format. 2: Binary data of the unformatted direct access.
<b>dmpcmp</b>	<i>integer * 4</i> Option of the vertical coordinate of the history file. When <b>dmpcmp</b> = 2 or 1, A small value of $-1.0 \times 10^{35}$ are outputted below the surface and above the top of the domain. 1 The vertical coordinate of output is $z^*$ ( $\zeta$ ). 2 The output variables are interpolated at horizontal levels with an interval of <b>dz</b> . 3 The output variables are interpolated at horizontal levels defined by the stretching functions.

---

<code>dmpitv</code>	<i>real * 4</i> Time interval of the history-file outputs. Unit is second. The interval is counted from the initial time even the calculation is started from a restart file. For example, if <code>dmpitv=300.0</code> seconds and restarted from 450.0 sec, the first output time will be 600.0 second from the initial time.
<code>resitv</code>	<i>real * 4</i> Time interval of the restart-file outputs. Unit is second. The time interval is also counted from the initial time even if started from a restart file.
<code>mxnitv</code>	<i>real * 4</i> Time interval to output maximum and minimum of prognostic variables to the standard output. Unit is second. The time interval is also counted from the initial time even if started from a restart file.

---

### & project0

The namelists describe the coordinate system of an external data. Although they used both *gridata* and *terrain*, no dependency was present between these programs.

---

<code>mpopt0</code>	<i>integer * 4</i> Option for the coordinate of the external data. 1: latitude and longitude coordinate 2: polar stereo graphic projection 3: Lambert conformal projection 4: Mercator projection 5: No map projection
<code>nspol0</code>	<i>integer * 4</i> Switch to define origin of the coordinate in the northern or southern hemisphere. 1: the origin of coordinate in the northern hemisphere. -1: the origin of coordinate in the southern hemisphere.
<code>tlat10,tlat20</code>	<i>real * 4</i> The true latitudes of map projections. No deformation due to projection occurs at the latitudes. The unit is degree [°] and they are negative in the southern hemisphere. If the Lambert conformal projection is used, two true latitude ( <code>tlat10</code> and <code>tlat20</code> ) are necessary. If no map projection is used, they are not necessary to be specified.
<code>tlon0</code>	<i>real * 4</i> The true longitude of coordinate. The unit is degree [°] and degree in the West Longitude is negative. The y-axis of the model domain corresponds to the longitude.

---

### & gridset0

The namelists describe grid intervals and the reference points. They are used in *gridata* and *terrain* while no dependency is present between the two programs.

---

<code>xdim0,ydim0, zdim0</code>	<i>integer * 4</i> Dimensions of external data in the <i>x</i> , <i>y</i> and <i>z</i> directions, respectively. The vertical dimension includes the surface. These namelist will be used for two-dimensional variables such as altitude. In this case, <code>zdim0</code> is not used.
-------------------------------------	--



---

<code>dx0,dy0</code>	<i>real * 4</i> Grid intervals of the external data in the $x$ and $y$ directions, respectively. Unit is degree if <code>mpopt = 0</code> and otherwise meter. The vertical levels should be included variables of the external data.
<code>ulat0,ulon0</code>	<i>real * 4</i> Longitude and latitude of a reference point of the external data. Unit is degree and they are negative in the southern hemisphere and/or the West Longitude. The point is corresponded to the real number of the external data ( <code>riu,rju</code> ) to determine the region in the pre-processors
<code>riu0,rju0</code>	<i>real * 4</i> The real number of the reference point which corresponds to the point determined by the reference longitude and latitude ( <code>ulat0,ulon0</code> ).

---

**& how2intp**

The namelist determines the scheme to interpolate the external data to the model grid. It is used in *gridata* and *terrain*, while no dependency is present in these programs.

---

<code>biiopt</code>	<i>integer * 4</i> Option for scheme of horizontal interpolation of the external data to the model grid. The vertical interpolation is always the linear interpolation. 0: a linear interpolation. 1: a interpolation using the square function.
---------------------	---

---

**& datacf3d**

Format of the 3-dimensional external data (such as GPV data) an option of *gridata* are configured in the namelists.

---

<code>datatype</code>	<i>character * 80</i> Specify data type of variables in the external data. The following combinations are available. tk: temperature [K] and water vapor mixing ratio [kg kg <sup>-1</sup> ] pk: potential temperature [K] and water vapor mixing ratio [kg kg <sup>-1</sup> ] tp: temperature [K] and relative humidity [%] pp: potential temperature [K] and relative humidity [%]
<code>refsfc</code>	<i>integer * 4</i> Flag to use or not the height and variables of the external data at the surface in the interpolation of the external data. If <code>refsfc =1</code> , the first level of <code>zdim0</code> is considered as the surface. This should be consistent with reading data in <code>rdobj.f</code> which must be modified by user. 0: The first level is the surface and used in interpolation. 1: The first level is not the surface.

---

## 9.2 Data flow in *CReSS*

### 9.2.1 Outline of data flow

The outline of the data flow in *CReSS* is shown in Fig.9.1. The figure includes all types of data flows. According to the configuration, some of programs and data depicted in the figure are not necessary. Each case of configuration will be described in the following subsections.

The italic character part of file names in Fig.9.1 are summarized in the following table.

<i>exprim</i>	The experimental name which is defined by user. The beginning part of all input and output files is the name.
<i>xxxxxx</i>	Elapsed time from the initial (second). For example, if the elapsed time is 1200.0 seconds, then <i>xxxxxx</i> = 001200.

These file names are used in the serial version of *CReSS*. In the parallel version, the number of processing element (CPU or nodes) *yyyy* is appended to the file names as *.peyyyy*.

Details of the data format will be described in Subsection 9.3.

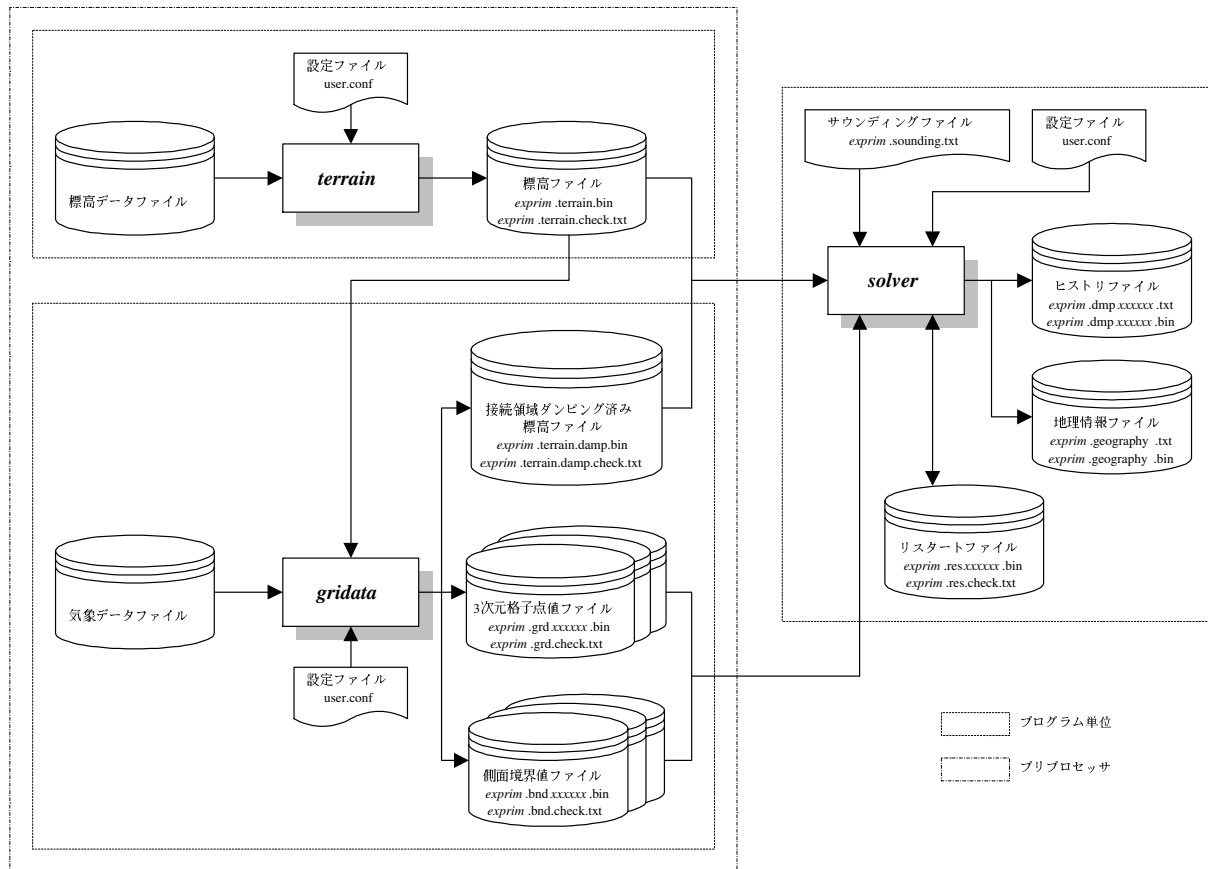


Figure 9.1. The outline of data flow in *CReSS*.

The data files in Fig.9.1 are summarized in the following table.

file name	description
<i>exprim.terrain.bin</i>	Altitude of the model surface. (unformatted direct access binary file).
<i>exprim.terrain.check.txt</i>	Parameter file to check the configuration of the above <i>exprim.terrain.bin</i> . (formatted text file).
<i>exprim.terrain.damp.bin</i>	Terrain data file which lateral boundary is smoothed to connect with the coarse outer terrain when the lateral boundary data is used. (unformatted direct access binary file).
<i>exprim.terrain.damp.check.txt</i>	Parameter file to check the configuration of the above <i>exprim.terrain.damp.bin</i> . (formatted text file).
<i>exprim.grdxxxxxx.bin</i>	GPV data coordinated at the model grid. Time series of the file is necessary when the nudging is performed. If nudging is not performed, the data file is used at the initial time. (unformatted direct access binary file).
<i>exprim.grd.check.txt</i>	Parameter file to check the configuration of the above <i>exprim.grdxxxxxx.bin</i> (formatted text file).
<i>exprim.bndxxxxxx.bin</i>	Boundary data coordinated at the model grid. Time series of this data is usually used. (unformatted direct access binary file).
<i>exprim.bnd.check.txt</i>	Parameter file to check the configuration of the above <i>.bndxxxxxx.bin</i> . (formatted text file).
<i>exprim.sounding.txt</i>	One-dimensional sounding data to provide the horizontally uniform initial field. (formatted text file).
<i>exprim.dmpxxxxxx.txt</i> or <i>.bin</i>	History dumped file. The format is optional: formatted text file or unformatted direct access binary file.
<i>exprim.geography.txt</i> or <i>.bin</i>	Geographic information of the domain. The format is optional: formatted text file or unformatted direct access binary file.
<i>exprim.resxxxxxx.bin</i>	File to restart. (unformatted direct access binary file).
<i>exprim.res.check.txt</i>	Parameter file to check the configuration of the above <i>exprim.resxxxxxx.bin</i> . (formatted text file).

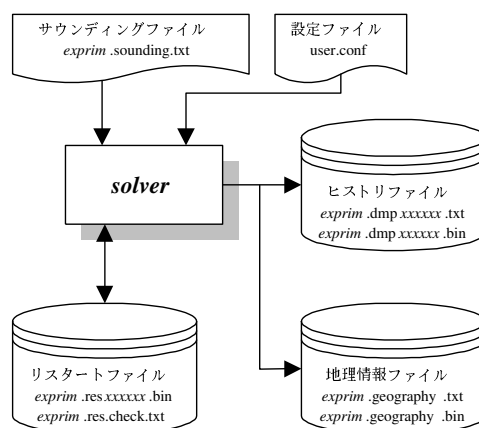
## 9.2.2 Data flows of each configuration

### Using sounding data

#### (1) In the case of no terrain data

The configuration in `user.conf` is summarized in the following table and the data flow is shown in the figure.

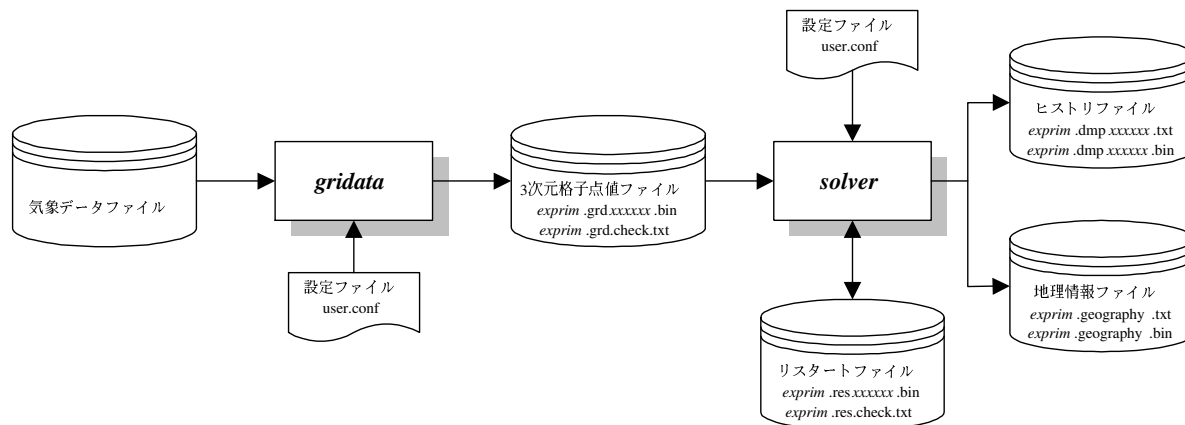
<code>iniopt = 1</code>	Initial data is provided by the sounding data.
<code>trnopt = 0 or 1</code>	No external terrain data is used.
<code>exbopt = 0 or 3</code>	No external boundary forcing is performed.



#### (2) Using the external terrain data

The Configuration in `user.conf` and the data flow are as follows.

<code>iniopt = 1</code>	Initial data is provided by the sounding data.
<code>trnopt = 0 or 1</code>	External terrain data is used for model topography.
<code>exbopt = 0 or 3</code>	No external boundary forcing is performed.

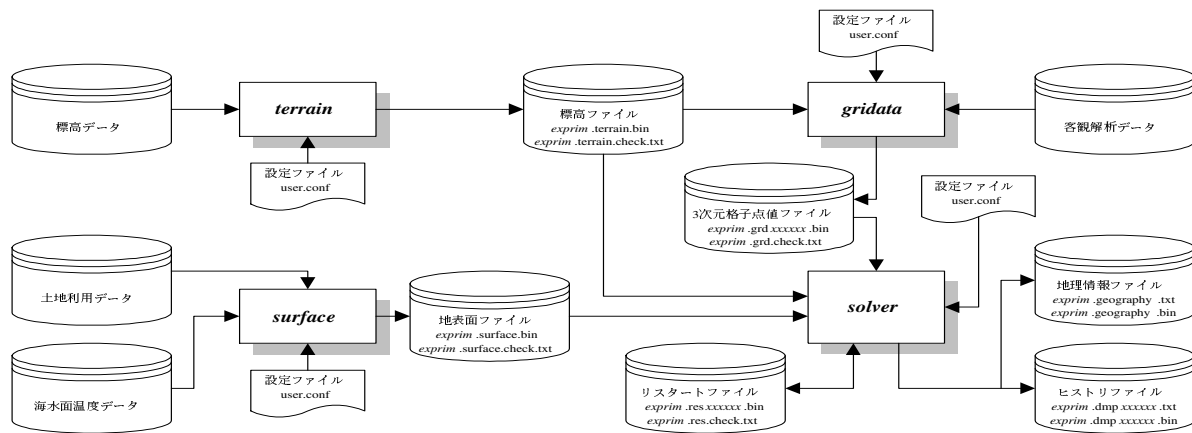


Using the 3-dimensional GPV data

(1) No external lateral boundary data and no terrain data

The Configuration in `user.conf` and the data flow are as follows. (This configuration will not generally used.)

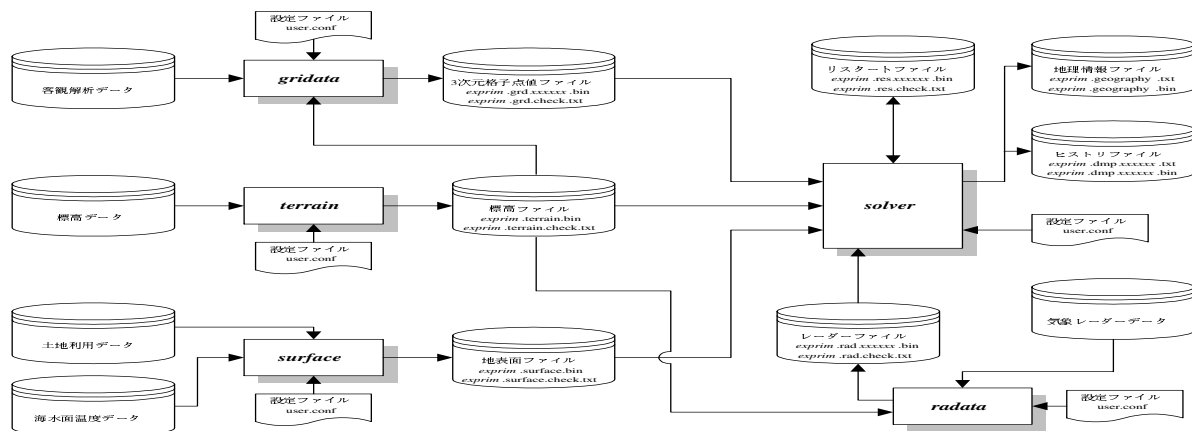
<code>iniopt = 3 or 4</code>	The initial data is a 3-dimensional GPV. Time series of the data is necessary if the nudging is performed.
<code>trnopt = 0 or 1</code>	No external terrain data is used.
<code>exbopt = 0</code>	No external boundary forcing is performed.



(2) A lateral boundary data is used while no external terrain data

The Configuration in `user.conf` and the data flow are as follows. (This configuration will not generally used.)

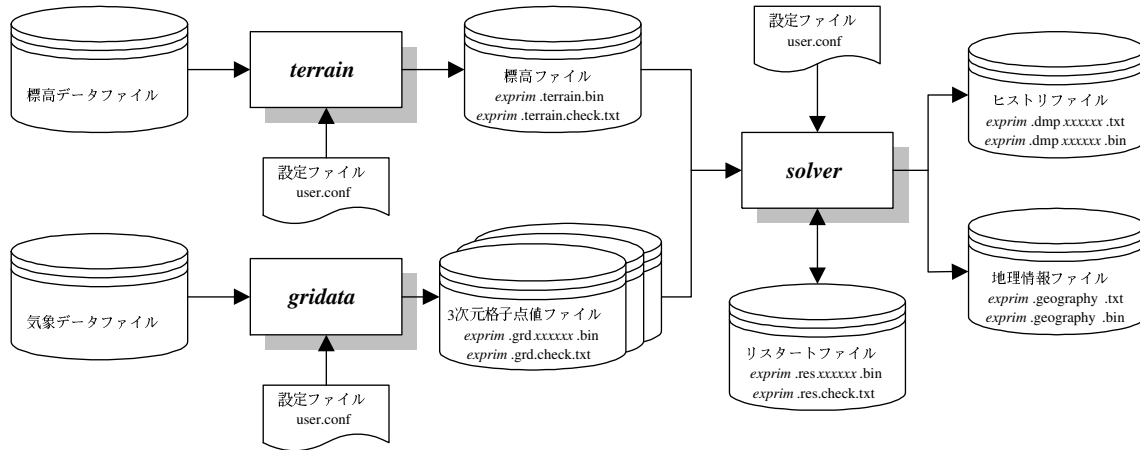
<code>iniopt = 3 or 4</code>	The initial data is a 3-dimensional GPV. Time series of the data is necessary if the nudging is performed.
<code>trnopt = 0 or 1</code>	No external terrain data is used.
<code>exbopt = 1 or 2</code>	External boundary forcing is performed.



(3) An external terrain data is used while no external lateral boundary data

The Configuration in `user.conf` and the data flow are as follows.

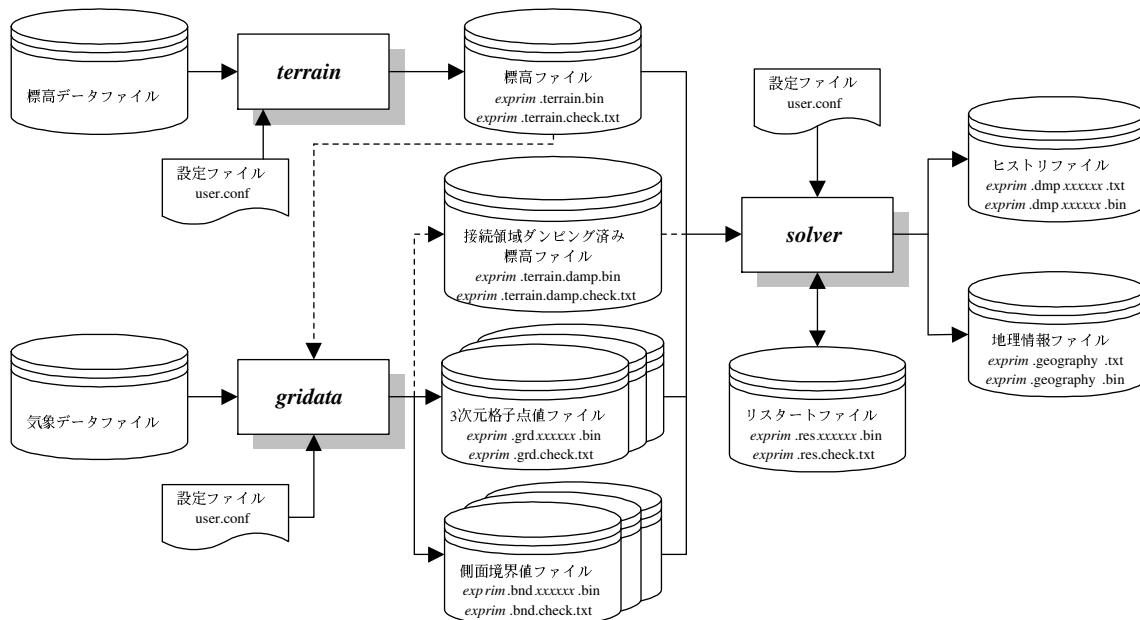
<code>iniopt = 3 or 4</code>	The initial data is a 3-dimensional GPV. Time series of the data is necessary if the nudging is performed.
<code>trnopt = 2</code>	External terrain data is used.
<code>exbopt = 0</code>	No external boundary forcing is performed.



(4) Using both external lateral boundary data and terrain data

The Configuration in `user.conf` and the data flow are as follows.

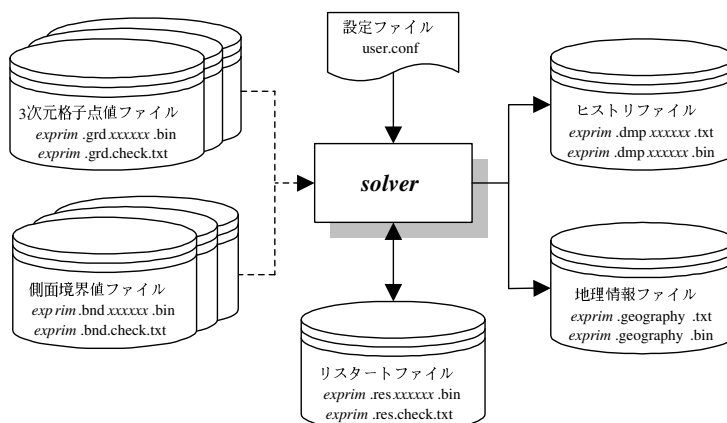
<code>iniopt = 3 or 4</code>	The initial data is a 3-dimensional GPV. Time series of the data is necessary if the nudging is performed.
<code>trnopt = 2</code>	External terrain data is used.(Boundary smoothing is performed optionally.)
<code>exbopt = 1 or 2</code>	External lateral boundary forcing is performed.



### Restarting from restart file

The Configuration in `user.conf` and the data flow are as follows.

<code>iniopt = 2</code>	Begin from the restart file. Time series of GPV is necessary if nudging is performed.
<code>trnopt = anything</code>	Terrain condition is already determined in the previous run. (No data is necessary.)
<code>exbopt = anything</code>	External lateral boundary data is necessary if the external lateral forcing is performed.



## 9.3 Formats of I/O data

In this section, formats of I/O data which controlled by users will be described: sounding data (`exprim.sounding.txt`), history dumped data (`exprim.dmpxxxxxx.txt` or `exprim.dmpxxxxxx.bin`) and geography data (`exprim.geography.txt` or `exprim.geography.bin`).

### 9.3.1 Format of the sounding data

The sounding data is a text data composed of five columns of variables. The data types of the first, second and fifth columns are defined by users. All types of combination are allowed. Therefore, there are eight combinations of variables.

First column	height [m] or pressure [Pa]
Second column	temperature [K] or potential temperature [K]
Third column	x-component of horizontal velocity [m s <sup>-1</sup> ]
fourth column	y-component of horizontal velocity [m s <sup>-1</sup> ]
fifth column	water vapor mixing ratio [kg kg <sup>-1</sup> ] or relative humidity [%]

This is an example of the sounding file. It is easy to make a sounding file following the example. Some samples are found in Form directory of *CRess*.

```
#####
#                                                                 #
#   One dimensional sounding input file, sounding.txt.cats.eye.form #
#                                                                 #
#   This is the cats eye simulation data.                         #
#                                                                 #
#   Author       : SAKAKIBARA Atsushi                            #
#   Date        : 1999/07/23                                     #
#   Modification : 1999/07/28                                     #
#               : 1999/11/19                                     #
#                                                                 #
#   First column: height [m]                                     #
#   Second column: tempereture [K]                              #
#   Third column: x components of velocity [m/s]                #
#   Fourth column: y components of velocity [m/s]              #
#   Fifth column: water vapor relative humidity [%]             #
#                                                                 #
#####
780.e0 300.200e0  8.0000e0 0.e0 0.e0
720.e0 300.200e0  7.9999e0 0.e0 0.e0
660.e0 300.200e0  7.9993e0 0.e0 0.e0
      :
      :
120.e0 299.800e0 -7.9946e0 0.e0 0.e0
 60.e0 299.800e0 -7.9993e0 0.e0 0.e0
  0.e0 299.800e0 -7.9999e0 0.e0 0.e0
```

### 9.3.2 Formats of the history dumped file and the geographic information file

Formatted text data or unformatted binary data is optionally chosen for the history dumped data and the geographic information data.

In the case of the formatted text data (`dmpfmt = 1`), a variable will be outputted as follows.

---

```
do xxx k=2,nk-2
  write(ionum,*,err=errnum) (variable(i,j,k),i=2,nx-2,j=2,ny-2)
xxx continue
```

---

If the unformatted binary data is selected (`dmpfmt = 2`), a variable will be outputted as follows.

---

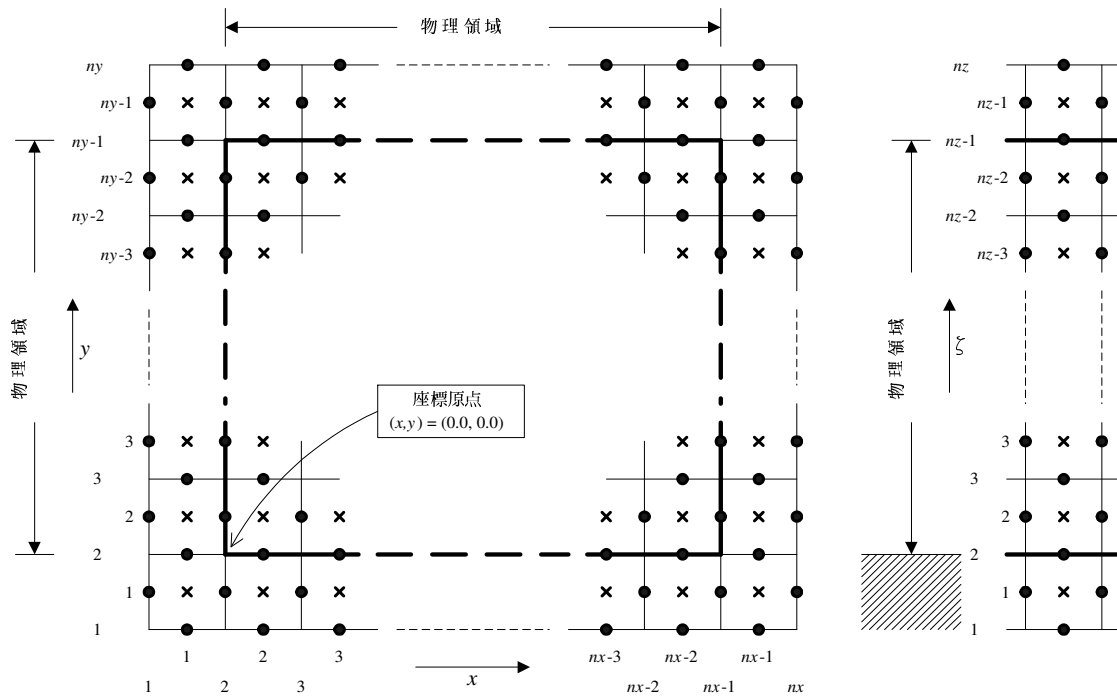
```
do xxx k=2,nk-2
  recnum=recnum+1
  write(ionum,rec=recnum,err=errnum)
    (variable(i,j,k),i=2,nx-2,j=2,ny-2)
xxx continue
```

---

(refer to `outdmp3d.f` and `outdmp2d.f`). where `nx,ny,nz` are numbers of the grid points in the  $x, y$  and  $z$  directions, respectively. The number of `nz` is unity in the geographic information file.

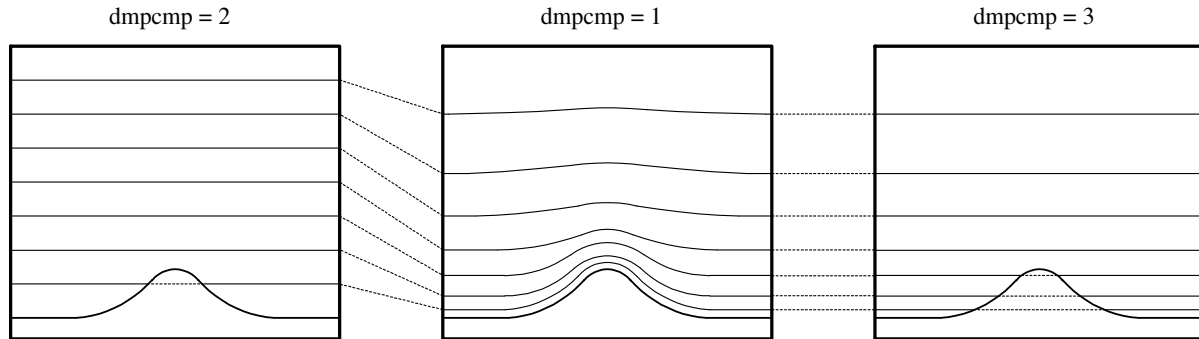


The region of the output data is indicated by the thick lines in Fig.9.2. The data points are indicated by the crosses which is the scalar points. The vector variables defined at the closed circles are interpolated to the scalar points. The dimensions of the  $x, y$  and  $z$  directions are respectively smaller than those of the model grid by three.



**Figure 9.2.** Setting of variables in the calculation domain. The closed circles are points of vector variables and the crosses are of scalar variables.

Options for vertical grid of the output data are shown in Fig.9.3.



**Figure 9.3.** Vertical coordinates of the history dumped data.

Output variables in the geographic information data and the history dumped data are summarized in the following tables. The types of output variables of the history dumped data are depend on the configurations.

Geographic information file			History dumped file		
<b>ht</b>	altitude of terrain	m	<b>u</b>	x-component of the horizontal velocity	$m s^{-1}$
<b>lat</b>	latitude	$^{\circ}$	<b>v</b>	y-component of the horizontal velocity	$m s^{-1}$
<b>lon</b>	longitude	$^{\circ}$	<b>w</b>	vertical component of velocity	$m s^{-1}$
<b>fs</b>	Coriolis parameter		<b>pbar</b>	basic state of pressure	Pa
<b>fc</b>	Coriolis parameter		<b>pp</b>	perturbation pressure	Pa
<b>mf</b>	map factor		<b>ptbar</b>	basic state of potential temperature	K
			<b>ptp</b>	perturbation potential temperature	K
			<b>qv</b>	mixing ratio of water vapor	$kg kg^{-1}$
			<b>qc</b>	mixing ratio of cloud water	$kg kg^{-1}$
			<b>qr</b>	mixing ratio of rain water	$kg kg^{-1}$
			<b>qi</b>	mixing ratio of cloud ice	$kg kg^{-1}$
			<b>qs</b>	mixing ratio of snow	$kg kg^{-1}$
			<b>qg</b>	mixing ratio of graupel	$kg kg^{-1}$
			<b>nci</b>	number density of cloud ice	$m^{-3}$
			<b>ncs</b>	number density of snow	$m^{-3}$
			<b>ncg</b>	number density of graupel	$m^{-3}$
			<b>zph</b>	height of grid points	m
			<b>prr</b>	rainfall intensity, total rainfall	$m s^{-1}, m$
			<b>prs</b>	snow intensity, total snowfall	$m s^{-1}, m$
			<b>prg</b>	graupel intensity, total graupel fall	$m s^{-1}, m$

## 9.4 Execution of *CReSS*

### 9.4.1 Execution of *solver*

Using an example experiment of the Kelvin-Helmholtz waves, the procedures to execute the solver program *solver* will be explained.

First, the user uncompress and expand the compressed archive file of *CReSS*. (This example is *CReSS*ver.1.1 while the essence is the same in other version.)

```
% ls
cress1.1m.tar.Z
% uncompress -c cress1.1m.tar.Z | tar xvf -
  messages
  :
% ls
CReSS1.1m      cress1.1m.tar.Z
%
```

The directory structure will be as follows.

```
% cd CReSS1.1m
% ls
Doc          Src          compile.conf
Form         Tmp          compile.csh
%
```

Directory begin with a capital letter while files with a small letter. These are summarized in the following table.

Doc	Documents and <b>Readme</b> are included.
Form	Examples of configuration and the related sounding data are included.
Src	All source codes are archived in the directory.
Tmp	A temporary directory used in compilation of <i>CReSS</i> .
compile.conf	Configuration file of compilation.
compile.csh	Shell script to control the compilation.

Some configuration files and the related sounding files are archived in the directory of **Form**. Copy the configuration file and the sounding file of the Kelvin-Helmholtz waves to the directory of **CReSS1.1m** as the following names. We will use **test** as the name of experiment. All I/O data must begin with **test..**

```
% cp Form/user.conf.cats.eye.form.2 user.conf
% cp Form/sounding.txt.cats.eye.form test.sounding.txt
% ls
Doc          Src          compile.conf      test.sounding.txt
Form         Tmp          compile.csh       user.conf
%
```

Second, the configuration file of compilation **user.conf** will be edited if necessary. If the compiler is different from **f90** or **mpif90** or special options of compilation are used, the file should be edited. Otherwise, the file is not changed. The word length of the real number is depend on the machine. If it is not 4, the parameter **wlength** in the user configuration file **user.conf** must be changed to the appropriate value (usually 1).

Third, compile the solver as follows. The dimension is defined in the user configuration file **user.conf** and include files are produced automatically. It is, therefore, not necessary to change the source codes of the solver.

```
% compile.csh solver user.conf
```

```
cd Src; messages
:
%
```

When the compilation is completed, the executable file `solver.exe` is generated. (Actually, `solver.exe` is a symbolic link to that in the directory of `Src`.)

```
% ls
Doc                Tmp                solver.exe
Form               compile.conf       test.sounding.txt
Src                compile.csh        user.conf
%
```

Forth, execute the solver with reading the user configuration file `user.conf` from the standard input and outputting `test.out` to the standard output. In order to use main frame computer, job script and NQS could be necessary. In that case, ask the system manager how to use NQS.

```
% solver.exe < user.conf > test.out &
%
```

If the calculation is terminated normally, *CRESS* leaves the message “This program stopped normally.” at the end of the standard output `test.out` and the history dumped files (the names include `dmp`), the geographic information file (the names include `geography`) and restart files (the names include `res`) are produced. The standard output includes history of calculation and maximum and minimum of prognostic variables.

```
% ls
Doc                test.dmp000120.pe0003.bin
Form               test.dmp000160.pe0000.bin
Src                test.dmp000160.pe0001.bin
Tmp                test.dmp000160.pe0002.bin
compile.conf       test.dmp000160.pe0003.bin
compile.csh        test.dmp000200.pe0000.bin
solver.exe         test.dmp000200.pe0001.bin
test.dmp000000.pe0000.bin test.dmp000200.pe0002.bin
test.dmp000000.pe0001.bin test.dmp000200.pe0003.bin
test.dmp000000.pe0002.bin test.dmp000240.pe0000.bin
test.dmp000000.pe0003.bin test.dmp000240.pe0001.bin
test.dmp000040.pe0000.bin test.dmp000240.pe0002.bin
test.dmp000040.pe0001.bin test.dmp000240.pe0003.bin
test.dmp000040.pe0002.bin test.geography.bin
test.dmp000040.pe0003.bin test.out
test.dmp000080.pe0000.bin test.res000240.check.txt
test.dmp000080.pe0001.bin test.res000240.pe0000.bin
test.dmp000080.pe0002.bin user.res000240.pe0001.bin
test.dmp000080.pe0003.bin test.res000240.pe0002.bin
test.dmp000120.pe0000.bin test.res000240.pe0003.bin
test.dmp000120.pe0001.bin test.sounding.txt
test.dmp000120.pe0002.bin user.conf
%
```

#### 9.4.2 Execution of *gather*

In parallel calculation, the history dumped files are outputted separately by each preprocessing element. To combine the files, the post-processor *gather* is used.

The compilation of *gather* is as follows.

```
% compile.csh gather user.conf
cd Src; messages
      :
%
```

Then the post-processor *gather.exe* is generated. *gather.exe* is an interactive executable. The following example combines the history dumped file at 240 seconds. The name of experiment is *test*.

```
% gather.exe
Now the program, gather start.

Input the history file name of PE #0000. readfl(character) = ?
  >> test.dmp000240.pe0000.bin
Input the word length for your machine. wlngh(integer) = 4

Input the selected cloud micro physics option. cphopt(integer) = 0

Input the selected history file z coordinates option. dmpcmp(integer) = 1

Can I delete history files after processing? "yes" or "no", fdelet = no

Your setting are:
  readfl = "test.dmp000240.pe0000.bin"
  wlngh = 4,  cphopt = 0,  dmpcmp = 1,  fdelet = "no"

Are these setting correct? "yes" or "no" >> yes
  messages
      :
This program stopped normally.
%
```

The combined file is *test.dmp000240.gather.bin*. The format of the file is described in Subsection 9.3.2. Types and order of output are also found in the standard output *test.out*.

The file is used in a graphic application such as Grads, then the result will be displayed as in Fig.9.4 and Fig.9.5.

### 9.4.3 Execution of *terrain*

The pre-processor *terrain* is a program to interpolate a terrain data into the model grids. This is necessary in a prediction experiment as well as a numerical experiment using a non-analytic topography.

While an external terrain data is necessary to produce the model terrain using *terrain*, the format of the external terrain data is not specified in *CReSS*. In order to read the data, it is necessary to modify the subroutine *rdtrn.f* which will be found in the directory *Src*. User will modify the parts of the *open* statement and *read* statement in *rdtrn.f* which is indicated by # to read the external terrain data.

---

```
* ##### You need to modify the following lines. #####
```

```
      siz=nid*njd*wlngh
```

```
      write(trnfl(1:16),'(a16)') 'data.terrain.bin'
```

```

        open(iotrn,iostat=stat,file=trnfl(1:16),status='old',
        .   access='direct',form='unformatted',recl=siz)

* #####

* ##### You need to modify the following lines. #####

        read(iotrn,rec=1,err=100) ((htdat(id,jd),id=1,nid),jd=1,njd)

* #####

```

---

The unit of the terrain data is meter. If the unit is not meter, it must be corrected here.

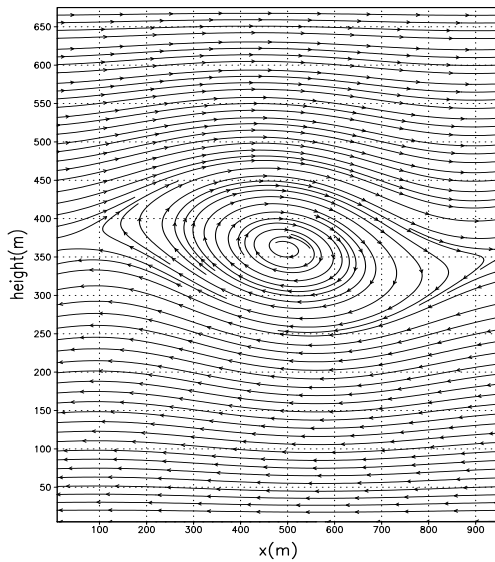
After the namelists of the user configuration file are specified (refer to Section 9.1), compile as follows.

```

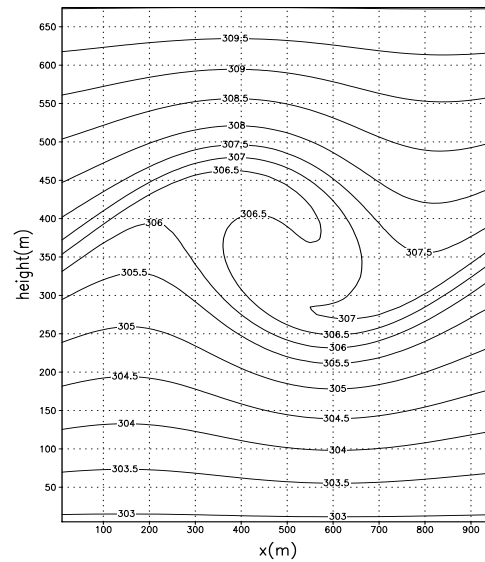
% compile.csh terrain user.conf
cd Src; messages
      :
%

```

When the compilation is completed, the executable program `terrain.exe` is generated. Actually, `terrain.exe` is a symbolic link to that in the directory of `Src`.



**Figure 9.4.** Stream function of  $(u - w)$  at 240 seconds from the initial.



**Figure 9.5.** Potential temperature [K] at 240 seconds from the initial.

The execution of `terrain.exe` is as follows. The input data file must be in the current directory.

```
% terrain.exe < user.conf > test.out.terrain &
%
```

If the execution succeeded, the terrain data file with the experimental name at the beginning (in the case, it is `test`) will be generated. (For example `test.terrain.bin`). The program `terrain.exe` of the serial and parallel versions is performed in a single processing element. A single file will be produced by the serial version while multiple files by the parallel version with an extension of `.peyyyy`. Where `yyyy` is the unit number of the processing element. This is necessary to execute the parallel version of `solver.exe`.

#### 9.4.4 Execution of *gridata*

The pre-processor *gridata* produces 3-dimensional grid point data files and the lateral boundary data files.

The format of the external GPV data such as an objective analysis data or a coarse model output is not specified in *CReSS*. In order to read the data, it is necessary to modify the subroutine `rdobj.f` which will be found in `Src`. User will change the `open` statement and `read` statement of the subroutine `rdobj.f`, which are indicated by `#`. The file names of the input data must include date and time of the Gregorian calendar `yyyymmddhhmm` (UTC).

```
* ##### You need to modify the following lines. #####

write(objfl(1:24), 'a8,a12,a4') 'data.obj', cdate(1:12), '.bin'

open(iobj, iostat=stat, file=objfl(1:24), status='old',
     .   access='sequential', form='unformatted')

* #####

* ##### You need to modify the following lines. #####

read(iobj, end=100, err=100) zdat

read(iobj, end=100, err=100) udat
read(iobj, end=100, err=100) vdat
```

<code>qrdat</code>	mixing ratio of rain water	kg kg <sup>-1</sup>
<code>qidat</code>	mixing ratio of cloud ice	kg kg <sup>-1</sup>
<code>qsdat</code>	mixing ratio of snow	kg kg <sup>-1</sup>
<code>qgdat</code>	mixing ratio of graupel	kg kg <sup>-1</sup>

If the units of the variable are different from the above, correction is necessary in the subroutine.

If data at the surface is present (`refsfc = 1`), the data is allocated at the first level (`kd = 1`). If the altitude of the surface is allocated at the first level of `zdat` and the corresponded data is allocated at the first level of each data, the level which is lower than the altitude will be ignored in the vertical interpolation when `refsfc = 1`. If `refsfc = 0`, the interpolation or extrapolation will be performed in the model grid even if the data is below the ground.

Compilation and execution of `gridata.exe` are essentially the same as `terrain.exe`. After the namelists are configured (refer to Section 9.1), compile the program as follows.

```
% compile.csh gridata user.conf
cd Src; messages
:
%
```

If the compilation is succeeded, the executable file `gridata.exe` is generated. Actually `gridata.exe` is also a symbolic link to that in the directory `Src`.

The execution of `terrain.exe` is as follows. The input data file must be in the current directory.

```
% gridata.exe < user.conf > test.out.gridata &
%
```

If the execution succeeded, 3-dimensional grid data files (in this case `test.grdxxxxxx.bin`), the lateral boundary data files (`test.bndxxxxxx.bin`), and optionally the model terrain data which is connected with the external terrain smoothly (`test.terrain.damp.bin`) are produced.

The program `gridata.exe` of the serial and parallel versions is performed in a single processing element. A single file will be produced by the serial version while multiple files by the parallel version with an extension of `.peyyyy`. Where `yyyy` is the unit number of the processing element. This is necessary to execute the parallel version of `solver.exe`.





## *CReSS* ユーザーズガイド 第2版

---

- 坪木 和久  
名古屋大学地球水循環研究センター  
e-mail: tsuboki@ihas.nagoya-u.ac.jp
  
- 榊原 篤志  
財団法人高度情報科学技術研究機構（名古屋大学地球水循環研究センター 駐在）  
e-mail: atsusi@ihas.nagoya-u.ac.jp
  
- 名古屋大学地球水循環研究センター  
464-8601 名古屋市千種区不老町
  
- 財団法人高度情報科学技術研究機構 東京事業所  
153-0061 東京都目黒区中目黒 2-2-54

上記は、平成 13 年 9 月現在のものです。







