國立交通大學

應用數學系

碩　士　論　文

運用在分解定義域之淺水方程模型的

切比雪夫排列法

Chebyshev Collocation Method for Shallow Water Models with Domain Decomposition

研 究 生： 張永潔

指導教授： 郭鴻基、賴明治　教授

中 華 民 國 九 十 八 年 八 月

運用在分解定義域之淺水方程模型的切比雪夫排列法

Chebyshev Collocation Method for Shallow Water Models with Domain Decomposition

研 究 生：張永潔　　Student：Yung-Chieh Chang

指導教授：郭鴻基、賴明治　Advisor：Hung-Chi Kuo, Ming-Chih Lai

國 立 交 通 大 學

應 用 數 學 系

碩 士 論 文

A Thesis

Submitted to Department of Applied Mathematics

College of Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Applied Mathematics

August 2009

Hsinchu, Taiwan, Republic of China

中 華 民 國 九 十 八 年 八 月

運用在分解定義域之淺水方程模型的切比雪夫排列法

學生：張永潔　　指導教授：郭鴻基、賴明治

國立交通大學應用數學系（研究所）碩士班

摘　要

本篇論文介紹使切比雪夫排列法加速的分解定義域方法。我們將定義域分割成許多子定義域，使其可以進行平行運算，並且在完成每一時步的積分後在子定義域之間的邊界進行資訊的交換。我們先使用此方法運作在三個一維的方程做為例子，分別是傳導方程、擴散方程以及 inviscid Burgers 方程，並且提供了指數收斂的計算結果。另外再延伸至更實際的二維淺水方程做為例子，並且獲得與未分割定義域之前十分相近的計算結果。因此我們也得到結論，對於大氣或海洋的模式，分解定義域的切比雪夫排列法是一個十分有效率的數值方法。

Chebyshev Collocation Method for Shallow Water Models with Domain Decomposition

Student：Yung-Chieh Chang　　Advisors：Hung-Chi Kuo, Ming-Chih Lai

Department (Institute) of Applied Mathematics

National Chiao Tung University

Abstract

The spectral methods seek the numerical solutions by a set of known polynomials. The main advantage of using spectral methods for solving atmospheric problems is the high efficiency and conservations of important quadratic quantities such as kinetic energy and enstrophy. Namely, we can get very high accuracy through the exponential convergence. The conservation of the quadratic quantities are important to model the turbulence under strong rotation and stratification. In this paper, we introduce the domain decomposition method to speed up the Chebyshev collocation method. The domain decomposition is to divide the domain into many sub-domains to run the computation in parallel and to exchange the information through the sub-domain boundaries during the time integration. We implement the domain decomposition Chebyshev collocation method with overlapping the sub-domains in one grid spacing interval for 1-D tests such as advection, diffusion and inviscid Burgers equations. We show the exponential convergence property and error characteristics in these tests. In a more realistic atmospheric modeling, we study the spectral method with 2-D shallow water equations. The domain decomposition results compared favorably with that of the single domain calculations. Thus, Chebyshev domain decomposition method may be an efficient alternative method for the atmospheric/oceanic limited area modeling.

誌　　謝

關於本篇論文的完成，首先要感謝我的兩位指導老師　郭鴻基教授與賴明治教授。從大四開始修賴老師的課，帶領我讓我體會到科學計算的有趣之處，而後在碩士班一年級，幸運地在賴老師的指導下，更加認識這片領域以及在實作之中找到成就感。碩士二年級時有幸接受郭老師的指導，在這一年郭老師耐心地教導與引導我，讓我習得了許多大氣科學的知識，並且進一步認識了科學計算的強大應用功能以及數值計算與大氣科學的結合。除了指導教授之外，交大與台大的實驗室伙伴們也給多我許多幫助。交大的陳冠羽學長是我常請教的對象，學長深厚的科學計算與數學分析能力，常常為迷惘的我指引一條明路。一路上我從應用數學跨領域到大氣科學，台大的徐理寰、蔡禹明學長、陳珮雯學姐以及姜浩天學弟也常常提供我許多建議，並且帶領我將模型建立得更完整。還有陳昱丞、黃義閔與許哲維同學們和我在課業與研究上面互相鼓勵與切磋，以及老師們與學長姐們留下來的資料，都成為我在碩士生涯之中不斷前進的助力。在論文口試期間，承蒙楊肅煜教的費心審閱以及提供許多寶貴意見與指導，使得本篇論文更加齊備，學生永誌於心。

研究所求學過程之中，我也要感謝我的研究室夥伴陳玉雯、游舜婷與曾慧棻同學們對我的關心與協助以及相互砥礪，與妳們一同的點點滴滴將會是我難忘的回憶；也感謝簡文昱、陳佳玲、葉彬以及多位研究所同學平時和我話家常、交換心得與想法以及相互鼓勵，一同度過這段歲月。

最後，要感謝我的家人，在一路上給我最多的支持以及自由的學習環境，讓我可以專心且順利地完全學業。再次感謝所有曾經幫助我、關心我的人，謝謝你們！

目　　錄

1 Introduction

With the advent of the fast Fourier transform(FFT) and the spectral transform method ([1]), spectral methods have been used successfully in global atmospheric models(e.g., [2] and [3]). Comparing with finite difference and finite element methods, global spectral models can eliminate pole problems and have high accuracy and efficiency that comes from the "exponential-convergence" property. The spectral methods also offer the discrete conservations of kinetic energy and enstropy, which are very important for the two-dimensional turbulence modeling. In addition to the popularity of spectral method for global models, finite difference or finite element methods are usually used to deal with limited-area models in atmospheric sciences. The main barrier of spectral methods used in limited area models is the time-dependent boundary conditions. Some discussions about computing atmospheric model which are restricted in smaller-scale and limited domain can be found in [4], [5] and [6]. Domain decomposition method is one of the efficient ways for improving the computational efficiency through the parallel processing. After the domain being divided, the information in sub-domains can be calculated in parallel with different Central Processing Units(CPUs) simultaneously thus increase the efficiency. Appendix B give a general discussion of parallel computing and Amdahls Law.

A suitable spectral method for the limited area atmospheric and oceanic modeling is the Chebyshev spectral method ([5] and [6]). The method handles successfully the time dependent boundary conditions while retain all the advantage of the global spectral methods. The earliest reference that to maintain the exponential convergence with domain decomposition for spectral methods is [7]. Later, Kopriva proposed an idea which is making hyperbolic equations on complicated geometries to be on squared sub-domains thus easier to execute spectral methods([8] and [9]). Kopriva also developed a conservative staggered-grid Chebyshev multi-domain method for compressible flows in papers [10] and [11]. Others, in the paper [12], the Chebyshev pseudospectral(collocation) method with domain decomposition is applied to deal with viscous flow calculation for lessening the influence of Gibbs phenomena on entire domain. These previous jobs about performing spectral methods with domain decompositon, were solving problems such as hyperbolic equations([8], [9]), viscous flow([12]), and compressible flows([10], [11]). None of these applications is directly related to the atmospheric science modelings.

In this paper, we implement Chebyshev collocation method with domain decomposition in the atmospheric sciences, based on [8] and [9]. We solve the shallow water equations in a limited-area of [6]. In Section 2, the Chebyshev collocation method is introduced briefly. We also talk about the way to demonstrate the domain decomposition and the advantages. In Section 3, we study 1-D cases using the Chebyshev collocation method with domain decomposition for advection equation, diffusion equation and inviscid Burger's equation. The error characteristics as well as exponential convergence are discussed. In Section 4, we present the shallow water equations and its numerical results. The MPI programming is employed to solve some multi-domain conditions here. The discussion on the time-splitting method for the model is also in this section. Section 5 gives concluding remarks.

2 Chebyshev Collocation Method and Domain Decomposition

2.1 Chebyshev Collocation Method

2.1.1 Analysis of boundary effects for limited domain

Spectral methods seek solutions in term of a series of known basis functions. Spectral methods are global type of method where the computation at given points depend not only on information at neighboring points, but on the entire domain. On the other hand, finite difference methods are local methods. The local refers to the use of nearby grid-points to approximate the function of its derivative at given points.

The essence of the choice of basis function is the property of the "completeness". Namely, the solution can be represented by the set of the functions. Practical consideration of the spectral methods are the basis functions are orthogonal and the projection or inner product can be calculated efficiently. The typical projection operators to find the spectral coefficients of the expansion are Galerkin, and tau methods. The basis functions often satisfy the prescribed boundary conditions in the Galerkin method. The tau method is more flexible in that the boundary condition need not be satisfied by the basis functions.

The "orthogonality" property of basis functions for spectral methods is practically useful in the computations. It makes the coefficients in the expansion independent of each other. The eigenfunctions of Sturm-Liouville equation are often chosen to be the basis. For demonstrating atmospheric spectral modeling, Spherical harmonics, Chebyshev and Fourier series are often used in the atmospheric spectral models. The Chebyshev and Fourier series allows a fast transforms to find spectral coefficients. These spectral methods get great accuracy and efficiency by the rapid convergence of fast transform.

The follow discussions are on the convergence property of the series expansion based on the Sturm-Liouville equation. More detailed analysis can be found in [15], [16], and [5].

Consider the Sturm-Liouville equation in limited domain [a,b]

$$L\phi(x) = -[p(x)\phi'(x)]' + q(x)\phi(x) = \lambda w(x)\phi(x) \tag{2.1}$$

with determined equations $p(x)$,$q(x)$ and $w(x)$. The prime represents the differentiation with respect to x. Eq.(2.1) has infinite and countable set of solutions $\phi(x)_{n=0}^{\infty}$ corresponding to eigenvalues $\lambda_{n=0}^{\infty}$. The eigenfunctions ϕ_n form an complete and orthonormal system under the inner product

$$(\phi_i, \phi_j)_w = \int_a^b \phi_i(x)\phi_j(x)w(x)dx = \delta_{ij} \tag{2.2}$$

where $\delta_{ij} = 1$ if $i = j$ and 0 otherwise. Therefore, any suitable smooth function $u(x)$ can be expanded by proper coefficients with basis $\{\phi_n\}_{n=0}^{\infty}$ as

$$u(x) = \sum_{n=0}^{\infty} \hat{u}_n \phi_n(x) \tag{2.3}$$

where

$$\hat{u}_n = (u, \phi_n)_w. \tag{2.4}$$

To estimate the magnitude of $\hat{u}_n$, we substitute for ϕ_n in $\hat{u}_n = (u, \phi_n)_w$ from Eq.(2.1), i.e.,

$$\begin{aligned} \phi_n(x) &= \lambda_n^{-1} w^{-1}(x) L\phi_n(x) \\ &= \lambda_n^{-1} w^{-1}(x)\{-[p(x)\phi_n'(x)]' + q(x)\phi_n(x)\}. \end{aligned} \tag{2.5}$$

Then we get $\hat{u}_n = \lambda_n^{-1}(u, w^{-1}L\phi_n)_w$. Next, doing integration by parts twice,

$$\begin{aligned} \lambda_n^{-1}(u, w^{-1}L\phi_n)_w &= \lambda_n^{-1}\{\int_a^b u(x)w^{-1}(x)\{-[p(x)\phi_n'(x)]' + q(x)\phi_n(x)\}w(x)dx\} \\ &= \lambda_n^{-1}[B(u, \phi_n) + (v, \phi_n)_w]. \end{aligned} \tag{2.6}$$

Here $B(u, \phi_n) = p(x)[u'(x)\phi_n(x) - u(x)\phi_n'(x)]|_{x=a}^{x=b}$ and $v = w^{-1}Lu$.

The Chebyshev polynomials are the solutions of Chebyshev differential equation where this is a special case of the Sturm-Liouville equation with $p(x) = (1-x^2)^{1/2}$, $q(x) = 0$ and $w(x) = (1-x^2)^{-1/2}$. About this case in domain $[-1, 1]$ and $p(-1) = p(1) = 0$, that means $B(u, \phi_n)$ equals to 0 no matter what bounded function u is. Thus we can do integration by parts for the $\lambda_n^{-1}[(v, \phi_n)_w]$ term repeatedly as long as the function is smooth enough after each integration. From the property that $(v, \phi_n)_w$ term is bounded independent of n and the asymptotic behavior of eigenvalues $\lambda_n = O(n^2)$ and eigenvectors $\phi_n(x) = O(1)$, $\phi_n'(x) = O(n)$ as $n \to \infty$, by [17], we get $\hat{u}_n < O(n^{-m})$ if u is m times differentiable. It satisfies the definition of exponential convergence, i.e., the convergence rate of Chebyshev series only depends on the smoothness of the expanded function and has nothing to do with the boundaries.

On the other hand, consider the case of the Fourier series where $p(x) = 1 \neq 0$. Here the function u must be periodic and smooth enough to maintain the exponential convergence property. Usually the u does not satisfy the periodic condition and it causes the convergence rate to be $B(u, \phi_n) = O(n)$ and $\hat{u}_n = O(n^{-1})$ which is very slow. If we give the boundary conditions to be $u(a) = u(b) = 0$, the convergence rates must be $B(u, \phi_n) = O(1)$ and $\hat{u}_n = O(n^{-2})$. The slow convergence rate is because of the reflection of the Gibbs phenomenon since the boundary conditions of u at a and b do not satisfy by the expansion function ϕ_n.

2.1.2 Chebyshev polynomial

From the previous discussion, Chebyshev polynomials $T_n(x)$ are appropriate basis function for boundary independent problems. There are n zeros of the Chebyshev polynomial T_n defined on $[-1, 1]$ by

$$T_n(x) = \cos n\theta, \text{ where } \cos\theta = x \tag{2.7}$$

Thus, the Chebyshev points are

$$x_j = \cos(\theta_j), \text{ with } \theta_j = \frac{(j + \frac{1}{2})\pi}{n}, \ 0 \leq j \leq n-1. \tag{2.8}$$

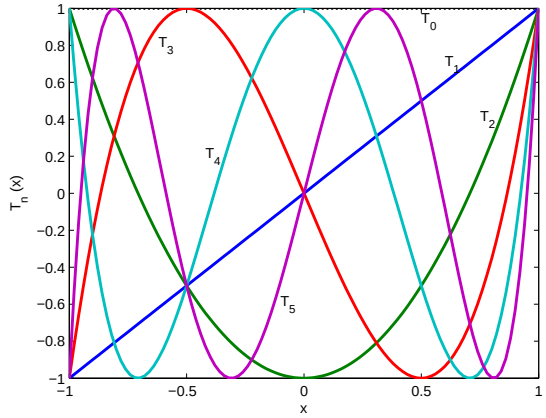

Figure 1: The Chebyshev polynomials of order 0 to 5.

$\langle\ ,\ \rangle$ denotes the Chebyshev inner product

$$\langle f, g\rangle = \int_{-1}^{1} \frac{f(x)g(x)}{\sqrt{1-x^2}}dx \tag{2.9}$$

and the Chebyshev polynomials have the orthogonality property

$$\langle T_m, T_n\rangle = \frac{\pi}{2}c_n\delta_{mn},\ c_n = \begin{cases} 2 & , n = 0 \\ 1 & , n > 0. \end{cases} \tag{2.10}$$

If a function can be expanded in Chebyshev series

$$\psi(x) = \sum_{n=0}^{\infty} \hat{\psi}_n T_n(x) \tag{2.11}$$

then we get the spectral coefficients $\hat{\psi}_n$ by the relation

$$\hat{\psi}_n = \frac{2}{\pi c_n}\langle \psi, T_n\rangle,\ n = 0, 1, ... \tag{2.12}$$

Hence, a function $\phi(x)$ for which we would like to find an approximation of its solution, the truncated Chebyshev series can be written as

$$\phi_N(x,t) = \sum_{n=0}^{N} \hat{\phi}_n T_n(x). \tag{2.13}$$

Eq.(2.11) and eq.(2.12) can be performed efficiently by the Fast Chebyshev Transform. Similar to the Fast Fourier Transform, the Fast Chebyshev Transform can perform N degree of freedom transformation in $O(N\log N)$ operations.

2.2 Domain Decomposition

2.2.1 Time speeding factor

The primal thought for us to develop methods for domain decomposition is saving computation time. Not only for paralleling computing, but for the larger Δt. After the

degree of freedom being decided in a fixed domain, the speed of demonstrating numerical method is mainly bounded by the Δt since the constrain of Courant-Friedrichs-Lewy condition. After doing domain decomposition, the Δt can be larger even keeping the same degree of freedom for whole domain since the domain has been cut to several sub-domains.

After setting up the Chebyshev grids, the minimal Δx is proportional to L/N^2, where L is the length of domain, N is the degree of freedom. Denote that Δt_s and Δt_m are the ideal time step for implementing some certain numerical methods in single domain and m sub-domains with domain decomposition, respectively. Since the time steps are proportional to the minimal Δx, it leads to

$$\Delta t_s \propto \frac{L}{N^2}. \tag{2.14}$$

For keeping the same degree of freedom N of the whole single domain, it should be set up for N/m Chebyshev grids at each sub-domain where the single domain has been cut to m sub-domains. Note that the length of each sub-domain is L/m, and the minimal Δx in sub-domains is proportional to $1/(\frac{N}{m})^2$. These cause that

$$\Delta t_m \propto \frac{\frac{L}{m}}{(\frac{N}{m})^2} = m\frac{L}{N^2}. \tag{2.15}$$

It implies that we can use the time step which is m times larger after doing domain decomposition than the original time step at single domain, and keep the stability still. For the idealist condition, if we put the data in each sub-domain in different computers separately, the speed of computing are just $1/m^2$ than we put data in single domain in one computer where the computers are with the same equipments. This is the main advantage of domain decomposition. In our preliminary tests, the Δt (the additional speed up factor) can be up to 4 in 8 sub-domains. In the following section, we will concentrated mainly on the spatial discretization errors of domain decomposition.

2.2.2 The overlapping boundaries

We have to introduce a proper method for exchanging the information of the boundaries between sub-domains at first. Here we use the overset method at those overlapped boundaries. It causes the importance of setting up the grids for dealing with domain decomposition and overset boundary problems. The grids can be set after L, N_a, N_D and N are given where L is the length of the whole domain, N_a+1 is the number of overlapped grids between sub-domains and $N_a \geq 1$, N_D is the number of sub-domains and N is the degree of freedom for each sub-domain. After these parameters being decided, the length of each sub-domain is

$$L_a = \frac{L}{N_D - 0.5(N_D - 1)(1 - cos(\frac{N_a\pi}{N}))} \tag{2.16}$$

Note that the setting makes the grids match well at the overlapped boundaries. x_{M_N} is donated to be the N^{th} Chebyshev collocation point at M^{th} sub-domain. The information

exchange in the overlapping boundaries simultaneously. Namely we assign $u(x_{M_{N_a}})$ value to $u(x_{(M+1)_N})$ and assign $u(x_{(M+1)_{(N-N_a)}})$ to $u(x_{M_0})$. The scheme of assignments is shown in Figure 2.

Figure 2: The information exchange at the overlapped boundary with $N_a = 1$.

3 1-D Test Problems

3.1 1-D Linear Advection Equation

Consider the one-dimensional linear advection equation as following

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0 \tag{3.17}$$

in the domain $[-1, 1]$. The initial and boundary conditions have been determined under the given analytical solution

$$u(x,t) = Aexp[-(\frac{x - x_0 - t}{h})^2] \tag{3.18}$$

where $h = 0.2$, $x_0 = -0.5$ and $A = h^{-1/2}(\pi/2)^{-1/4}$. Those conditions are mainly from [5].

For discussing the accuracy of Chebyshev collocation method, the finite difference fourth-order scheme(FD4) is introduced to do the comparison with it where the FD4 scheme about advection equation is

$$\frac{d\tilde{u}_j}{dt} + \frac{-\tilde{u}_{j+2} + 8\tilde{u}_{j+1} - 8\tilde{u}_{j-1} + \tilde{u}_{j-2}}{\Delta x} = \tilde{f}_j. \tag{3.19}$$

Note that $\tilde{(\,)}_j$ denotes the values at the grid points $\tilde{x}_j = -1 + j\Delta x$. At boundary points where $N = 0$, 1, $N - 1$, and N, we use the fourth-order one-sided finite differences where the derivation can be get by the idea of Taylor series.

The up in Figure 3 shows the results of analytical solution and the approximation by Chebyshev collocation method and FD4 with domain decomposition at time $t = 1.0$. Here the domain is divided into two sub-domains and the number of overlapped grids are two points, e.g., overlapped with one grid width, as Figure 2. The degree of freedom of each sub-domain is 24. It makes that the degree of freedom of the whole domain is double. As Figure 2, the approximation of Chebyshev collocation method with domain decomposition is identical to the analytic solution about 1-D advection problem.

The down in Figure 3 shows the L_2 error of the numerical results of Chebyshev collocation method with single domain and double domain and FD4 method with analytical

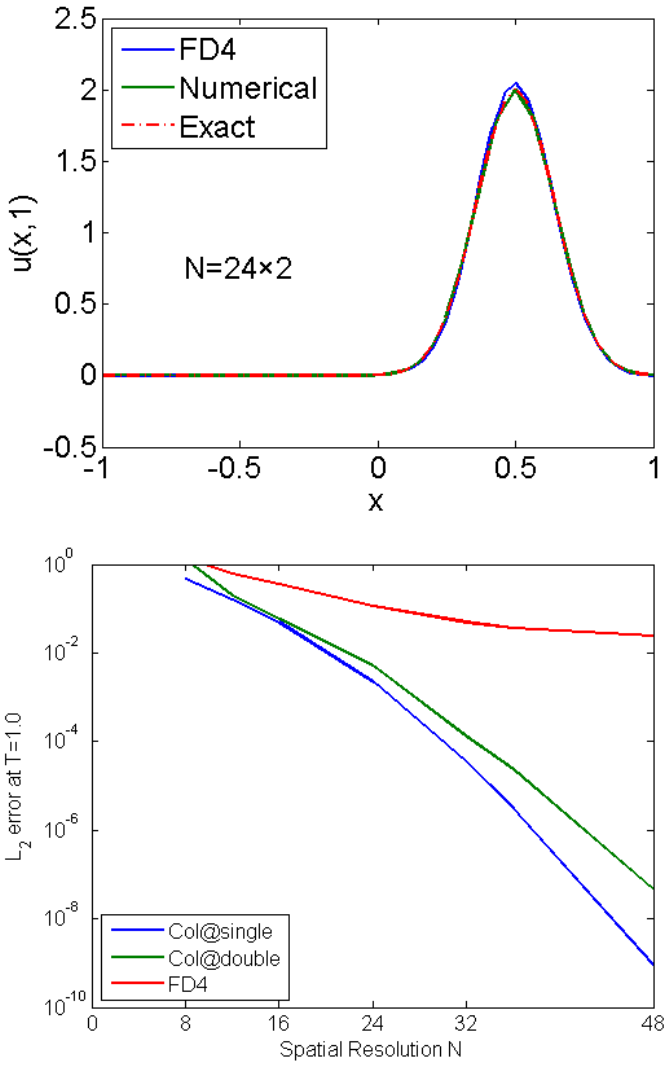

Figure 3: Analytical solution and numerical results of eq.(3.17) under condition eq.(3.18) and the L_2 error.

solution at $t = 1.0$. We use the norm

$$\|u - u_N\|_2 = \{\frac{2}{N}\sum_{n=0}^{N}\frac{1}{\bar{c}_j}[u(\tilde{x}_j, t) - u_N(\tilde{x}_j, t)]\}^{1/2} \tag{3.20}$$

to help us calculating the relative L_2 error. The error in the spectral solutions are decreasing like $10^{(-N/4)}$ as N approaches 64, while the error in FD4 decreases in order N. Note that the property of exponential convergence of the spectral method didn't lose where the domain decomposition is demonstrating.

3.2 1-D Diffusion Equation

Let's take a look of 1-D linear diffusion model with domain decomposition. The 1-D diffusion equation is

$$\frac{\partial u}{\partial t} = \kappa\frac{\partial^2 u}{\partial x^2} \tag{3.21}$$

in the domain [-5,5], where $\kappa = 0.1$. The initial condition is

$$u(x,0) = exp(-\frac{x^2}{0.64}). \tag{3.22}$$

The up in Figure 4 illustrates the comparison between the analytical solution and Chebyshev collocation method with domain decomposition and FD4 method at time $t = 1.0$. Here the FD4 scheme of diffusion equation is

$$\frac{d\tilde{u}_j}{dt} = \kappa \frac{-\tilde{u}_{j+2} + 16\tilde{u}_{j+1} - 30\tilde{u}_j + 16\tilde{u}_{j-1} - \tilde{u}_{j-2}}{12(\Delta x)^2} \tag{3.23}$$

Note that the values at $j = 1$, 1, $N-1$, and N, the fourth-order one-sided finite difference is used here. Each time step is $dt = 0.001$ and degree of freedom is $N = 24 \times 2$. We set the overlapped boundaries condition like the advection model at previous paragraph. Namely, they are overlapped with one grid width. The analytical solution and the numerical approximation are identical in Figure 4.

The down in Figure 4 shows the convergence rate of demonstrating FD4, Chebyshev collocation method with single domain, and with double domains. The same conclusion we can get from the rate of convergence about advection equation, the property of exponential convergence of error also maintain very well when we demonstrating the Chebyshev collocation method with double domain.

3.3 1-D inviscid Burgers Equation

Consider the inviscid Burgers equation

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = 0 \tag{3.24}$$

in the limited domain $[-1, 1]$ with the initial condition

$$u(x,0) = f(x) = \overline{u} - tan^{-1}(x - x_0). \tag{3.25}$$

The boundary conditions at $x = -1$ and $x = 1$ are decided by the general solution of eq.(3.24) which is

$$u(x,t) = f(x - u(x,t)t). \tag{3.26}$$

Thus, the analytical solution under the initial condition eq.(3.25) is

$$u = \overline{u} - \tan^{-1}(x - ut - x_0). \tag{3.27}$$

Furthermore, we differentiate eq.(3.27) with respect to x to get the time of scale-collapse which gives

$$\frac{\partial u}{\partial x} = -\frac{1 - t\frac{\partial u}{\partial x}}{1 + (x - x_0 - ut)^2}.$$

Consider $x = x_0 + \overline{u}t$ and $u = \overline{u}$, we obtained

$$(\frac{\partial u}{\partial x})_{x=x_0+\overline{u}t} = -(1 - t(\frac{\partial u}{\partial x})_{x=x_0+\overline{u}t}) \tag{3.28}$$

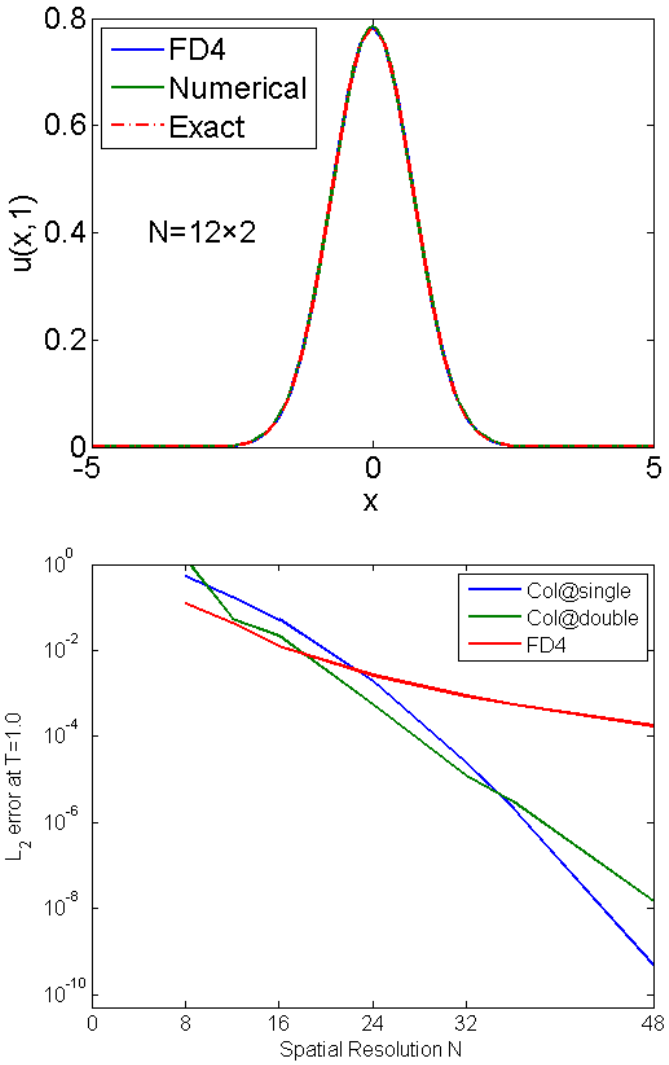

Figure 4: Analytical solution and numerical results of eq.(3.21) under condition eq.(3.22) and the L_2 error.

and then we have

$$(\frac{\partial u}{\partial x})_{x=x_0+\overline{u}t} = \frac{1}{t-1} \to -\infty \text{ as } t \to 1 \tag{3.29}$$

From the above derivation, the time of scale-collapse of u is 1 with the position of scale-collapse at $x_0 + \overline{u}$. Given the decided parameters $\overline{u}$ and x_0, the analytical solution of eq.(3.24) with the determined initial condition and boundary condition at certained x and t is found numerically for desired accuracy by fixed point iteration on eq.(3.27).

Figure 5 to 7 shows the numerical approximation of domain decomposition method could sketch the general picture of the analytical solution even though the degree of freedom is just 16×2 and the general error of numerical approximation for $t = 1$ at double domains is smaller than it at single domain with different $\overline{u}$ and x_0. For these case, the numerical errors are mainly from the position of scale-collapse at $x = x_0 + \overline{u}$. About $x_0 = 0$ and $\overline{u} = 0$, the collocation grids at single domain have the lowest density at this location. If we use double domains, the collocation grids have the highest density

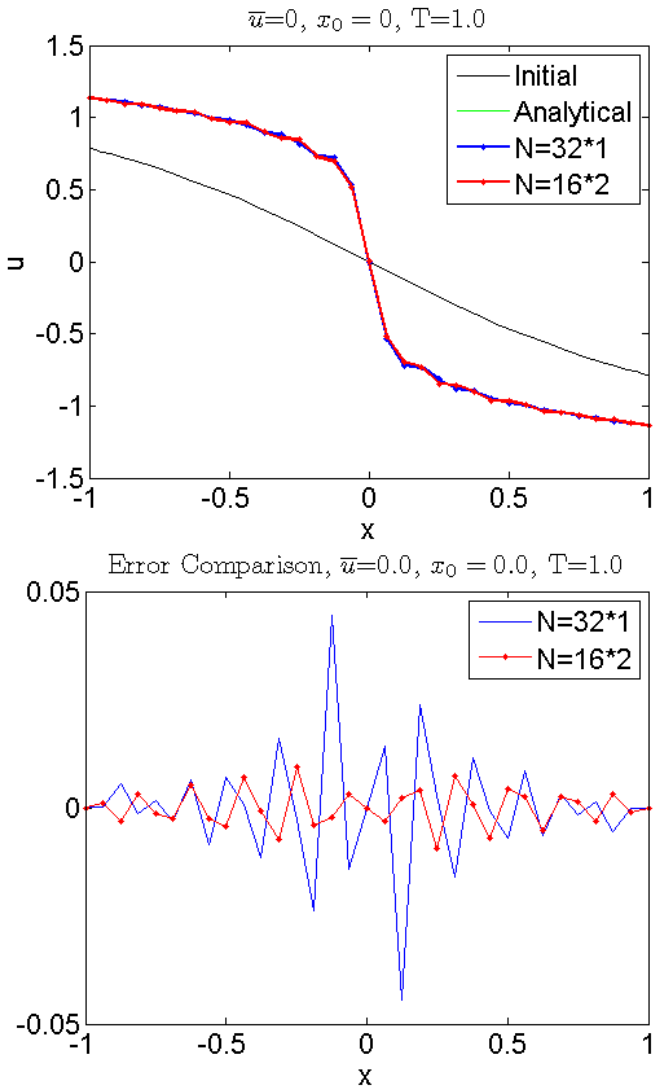

Figure 5: Analytical solution and numerical results in double domains of equation eq.(3.24) with $x_0 = 0$ and $\overline{u} = 0$ under conditions eq.(3.25) and eq.(3.27).

at $x = 0$ which cause the general error is smaller than it at single domain.

Let's take a look with another case $\overline{u} = 0.5$ and $x_0 = 0$. It means the scale-collapse moving with a background advection at speed $\overline{u} = 0.5$ from the initial position $x = 0$. There are more oscillations about the numerical approximation in this case. The general error of numerical approximation at double domains is larger than it at single domain, which is the opposite result to $\overline{u} = 0$ and $x_0 = 0$. It is because that the scale-collapse moving to $x = 0.5$ at $t = 1$ where the grids density at double domains is lower than it at single domain since $\overline{u} = 0.5$. About the case $\overline{u} = 0.5$ and $x_0 = -0.5$, the scale-collapse moves to $x = 0$ at $t = 1$. The error of results in double domain is smaller than in single domain has the same reason as the case $\overline{u} = 0$ and $x_0 = 0$.

To confirm our idea, Figure 8 shows the convergence rates while the scale-collapse at $x_0 = 0$ with $\overline{u} = 0.5$ at each time respectively. Apparently, since the time of scale-collapse of u is 1, the convergence rates are getting worse as the time approaching to $T = 1$ generally. Otherwise, the convergence rates at double domain are better than the results

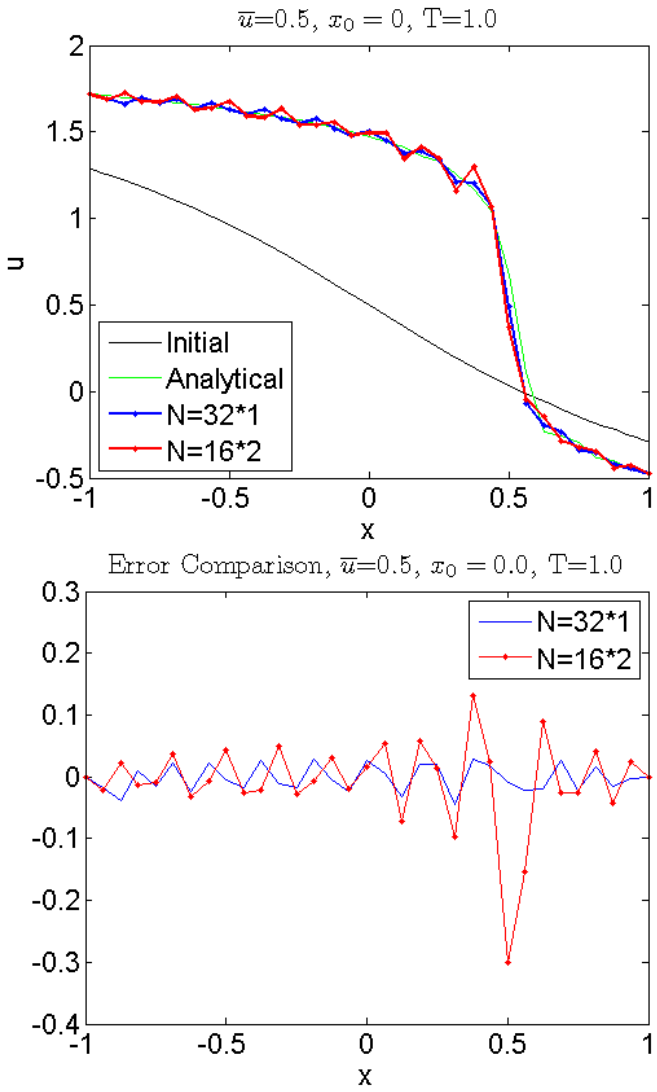

Figure 6: Analytical solution and numerical results in double domains of equation eq.(3.24) with $x_0 = 0$ and $\overline{u} = 0.5$under conditions eq.(3.25) and eq.(3.27).

at single domain. It verifies our idea that the magnitude of error is determined by the density of grids and the results of double domain condition is more excellent than of single domain under this condition.

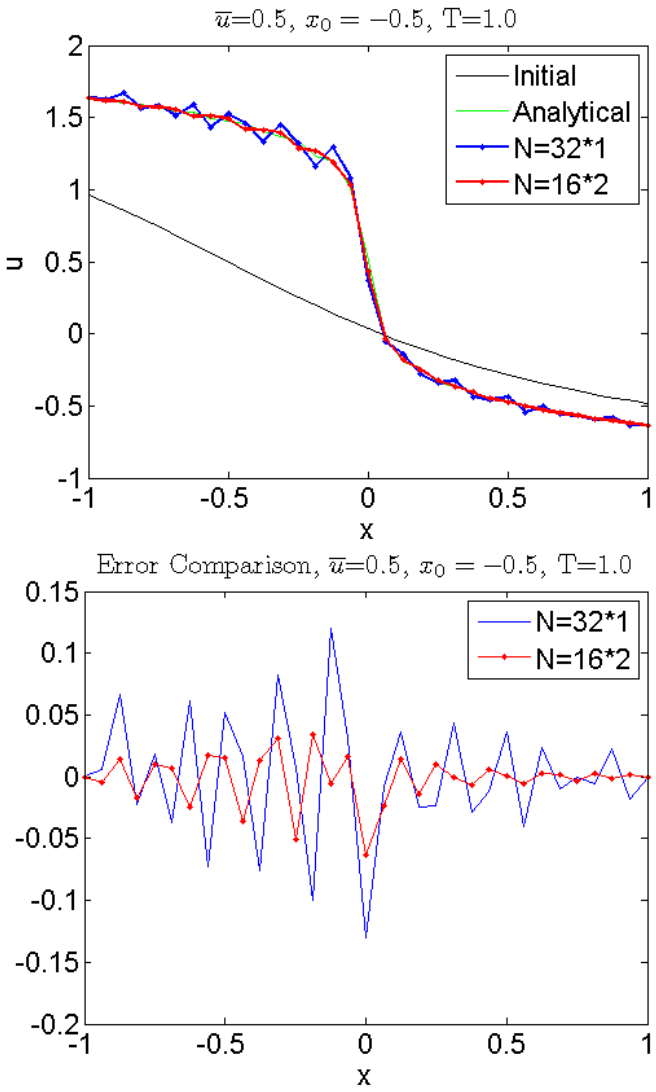

Figure 7: Analytical solution and numerical results in double domains of equation eq.(3.24) with $x_0 = -0.5$ and $\overline{u} = 0.5$ under conditions eq.(3.25) and eq.(3.27).

4 2-D Nonlinear Shallow Water Model

For testing the Chebyshev collocation method on a more realistic atmospheric model in two-dimensional, we introduce the nonlinear shallow water equations in Cartesian coordinates (x,y) :

$$\begin{aligned} \frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} - fv + \frac{\partial h}{\partial x} &= 0 \\ \frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + fu + \frac{\partial h}{\partial y} &= 0 \\ \frac{\partial h}{\partial t} + u\frac{\partial h}{\partial x} + v\frac{\partial h}{\partial y} + (\overline{h}+h)(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}) &= Q(x,y) \end{aligned} \tag{4.30}$$

Here u and v represent the velocity components in x and y directions. $\overline{h}$ is a constant basic state of geopotential and h is the deviation from $\overline{h}$. Note that the gradient of h causes the acceleration of gravity from the first and second equations. f is the Coriolis

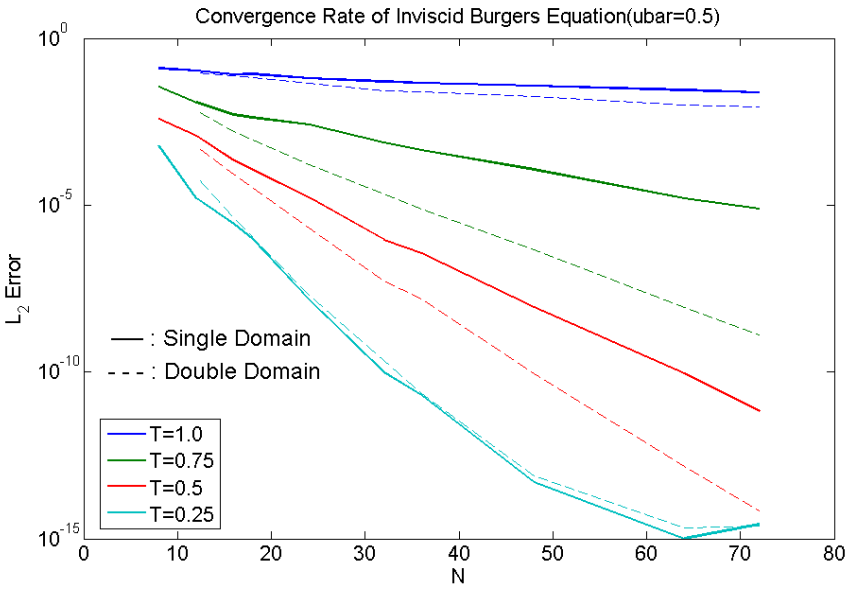

Figure 8: Convergence rate of eq.(3.24) while the scale-collapse at $x_0 = 0$ with $\overline{u} = 0.5$ at each time respectively.

force from the rotation of earth. $Q(x, y)$ represents the outer force of this system.

4.1 Chebyshev spectral discretizations

On the domain $x_a \le x \le x_b$, $y_a \le y \le y_b$, we demonstrate the Chebyshev collocation method based on the expansion

$$\begin{bmatrix} u(x,y,t) \\ v(x,y,t) \\ h(x,y,t) \end{bmatrix} \approx \begin{bmatrix} u_{MN}(x,y,t) \\ v_{MN}(x,y,t) \\ h_{MN}(x,y,t) \end{bmatrix} = \sum_{m=0}^{M} \sum_{n=0}^{N} \begin{bmatrix} \hat{u}_{mn}(t) \\ \hat{v}_{mn}(t) \\ \hat{h}_{mn}(t) \end{bmatrix} T_m(x') T_n(y').$$

Here M, N are spectral truncations in x and y respectively, $\hat{u}_{mn}$, $\hat{v}_{mn}$, $\hat{h}_{mn}$ are spectral coefficients, T_n denotes the Chebyshev polynimial of degree n, and $x' = 2(x - x_a)/(x_b - x_a)$, $y' = 2(y - y_a)/(y_b - y_a)$. We introduce the Chebyshev collocation points $\overline{x}_j, \overline{y}_k$ corresponding to $\overline{x}' = \cos(j\pi/M)$ where $j = 0, \cdots, M$ and $\overline{y}' = \cos(k\pi/N)$ where $k = 0, \cdots, N$. From those previous things, the collocation equations of shallow water system eq.(4.30) can be written as

$$\begin{aligned} \frac{d\overline{u}_{jk}}{dt} + \overline{u}_{jk}\overline{u}_{jk}^{(1,0)} + \overline{v}_{jk}\overline{u}_{jk}^{(0,1)} - f\overline{v}_{jk} + \overline{h}_{jk}^{(1,0)} &= 0 \\ \frac{d\overline{v}_{jk}}{dt} + \overline{u}_{jk}\overline{v}_{jk}^{(1,0)} + \overline{v}_{jk}\overline{v}_{jk}^{(0,1)} + f\overline{u}_{jk} + \overline{h}_{jk}^{(0,1)} &= 0 \\ \frac{d\overline{h}_{jk}}{dt} + \overline{u}_{jk}\overline{h}_{jk}^{(1,0)} + \overline{v}_{jk}^{(0,1)} + (\overline{h} + \overline{h}_{jk})(\overline{u}_{jk}^{(1,0)} + \overline{v}_{jk}^{(0,1)}) &= \overline{Q}_{jk} \end{aligned} \tag{4.31}$$

where the subscript jk denotes a value at the collocation point $(\overline{x}_j, \overline{y}_k)$, and the superscripts $(1,0)$ and $(0,1)$ denote the x and y derivative, respectively. The procedures of demonstrating Chebyshev collocation method about shallow water model are transforming items to spectral space, doing the derivative there, and then transforming back to physical space. About this model, there are 12 Chebyshev transforms at each time step. All the transforms are one-dimensional.

4.2 Overlapping boundaries

The handling of boundaries which are overlapping by sub-domains is similar to the 1-D problem, we give the overset boundary condition with one-grid width to exchange the information at those joints. Note there are two ways for us to divide the whole domain into sub-domains and we show both results to do comparison in following figures. One is dividing $x-$axis to two domain, then the 2-D single domain becomes 2×1 sub-domains(DD 2×1); the other is dividing $y-$axis additional than the previous way, namely, the original domain is cut to be 2×2 sub-domains(DD 2×2).

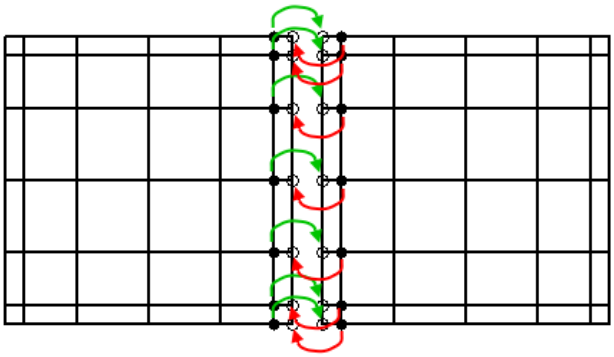

Figure 9: The information exchange at the overlapped boundary in 2-D shallow water model with double domain.

4.3 Numerical results

Here u and v represent the velocity components in x and y directions. $\overline{h}$ is a constant basic state of geopotential where $g\overline{h} = c^2$ is $2500m^2/s^2$ and h is the deviation from $\overline{h}$. Note that the gradient of h causes the acceleration of gravity from the first and second equations. f is the Coriolis force from the rotation of earth and we consider the $\beta-$effect ($f = f_0 + \beta y$ where f_0 and β are both constants) of model. All the results are presented at the domain $[x_a, x_b]\times[y_a, y_b] = [-2000\text{km}, 2000\text{km}]\times[-2000\text{km}, 2000\text{km}]$. We set in $y = 0$ means located at $30°N$ which makes $f_0 = 2\Omega sin\frac{\pi}{6}$ and $\beta = \frac{2\Omega}{R}cos\frac{\pi}{6}$ where $\Omega = \frac{2\pi}{86400\text{ sec}}$ is the rotating rate of earth and $R = 6378100\text{m}$ is the radius of earth. $Q(x,y)$ represents the outer force of this system. We give

$$Q(x,y,t) = q_0 exp[-(\frac{x-x_c}{x_0})^2 - (\frac{y-y_c}{y_0})^2]4t^2t_0^{-3}e^{-2t/t_0} \tag{4.32}$$

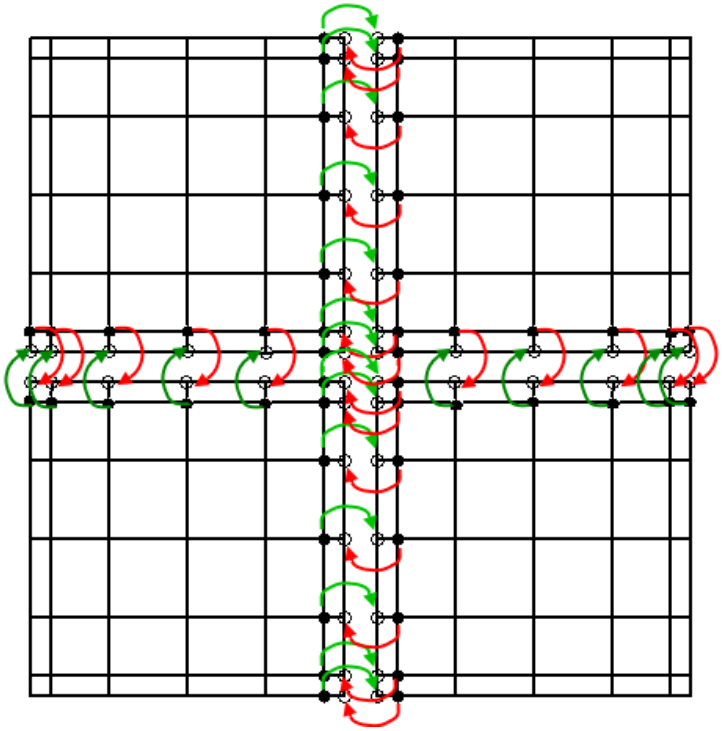

Figure 10: The information exchange at the overlapped boundary in 2-D shallow water model with DD2 × 2.

where the amplitude $q_0 = 6250m^2s^{-2}$, time scale $t_0 = 6$ hours $= 21600$ sec, e-folding width $x_0 = y_0 = 200$km and centered at $(x_c, y_c) = (1000 \text{ km}, -1000 \text{ km})$. Note that $Q(x, y, t)$ reaches its maximum when $t = t_0$ at each point.

For demonstrating the following cases, we set the initial condition to be

$$u(x, y, 0) = -Ucos[\pi \frac{y - y_a}{y_b - y_a}] \tag{4.33}$$

with $U = 7.5\text{m}s^{-1}$ and $v(x, y, 0) = 0$. The vortex which caused mainly from Q-force has variation as time goes by and the advection from u and v. We make h in geopotential balance on $\beta-$plane, i.e.,

$$\frac{\partial h}{\partial y}(x, y, 0) = -(f_0 + \beta y)u. \tag{4.34}$$

If we set $Q(x, y, t) = 0$ and this initial condition for h, the system is in geostrophic balance state on a $\beta-$plane continuously. About the boundary conditions, we set periodic overset condition for one grid width at $x = x_a$ and $x = x_b$. Namely we assign the u, v and h values at x_1 to the values at x_N, and the values at x_{N-1} to the values at x_0. The wall-condition is applied at $y = y_a$ and $y = y_b$, i.e., $v = 0$ at $y = y_a$ and $y = y_b$.

In this simulation, we pay attention to the vortex formation by the Q-forcing at $(1000 \text{ km}, -1000 \text{ km})$. The forced vortex is like the typhoon atmosphere. We will observe the vortex drift from the easterly background flow to the westerly background flow in our calculations.

The Chebyshev collocation method is used to calculate the derivatives which are $\partial u/\partial x$, $\partial u/\partial y$, $\partial v/\partial x$, $\partial u/\partial y$, $\partial h/\partial x$, $\partial h/\partial y$. Once the derivatives have been calculated,

we use the RK4 method to do the integration of time.

The results in Figure 11 to 16 are computed by the Chebyshev collocation method about the eq.(4.30) at 1.5, 3, 4.5, 6, 7.5 and 9 days respectively as labeled. The contour lines represent the geopotential field h/c. Figure 17 to 19 shows the analysis of the average velocity, vorticity, and pressure respectively at 3, 6, 9 days as labeled.

Since the shallow water equations are nonlinear with chaos, the model cannot be stable integrated forever. The model in single domain and DD2 $\times$ 2 blow up after 9 days and in DD 2 $\times$ 1 blows up after 10 days. We use 96×96 in single domain as a benchmark and compare with the degree of freedom $(48+48, 96)$ in double domain and $(48+48, 48+48)$ in DD2 $\times$ 2. We found the L_2 error of the domain decomposition with respect to single domain are order of 10^{-4} in these calculations.

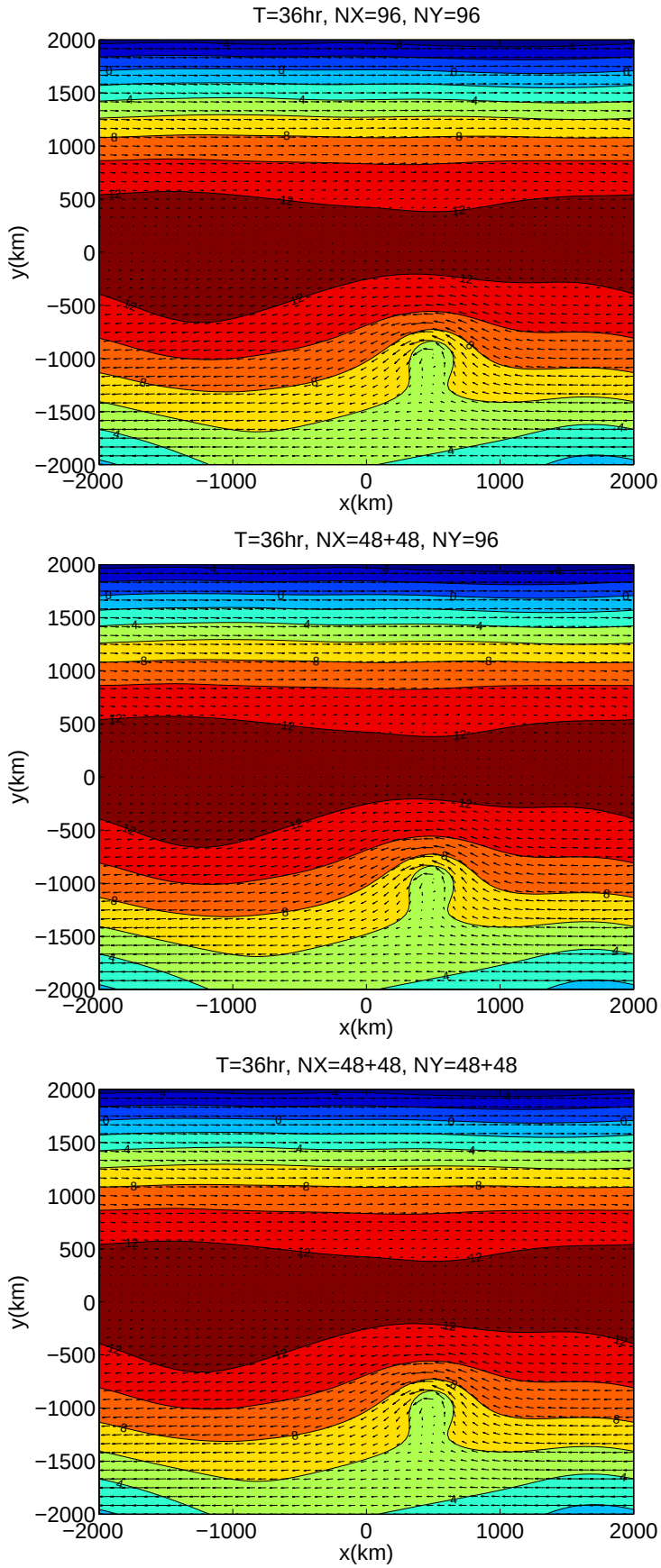

Figure 11: The results for shallow water model when T=36 hr. Up to down : single domain, DD 2×1, DD 2×2.

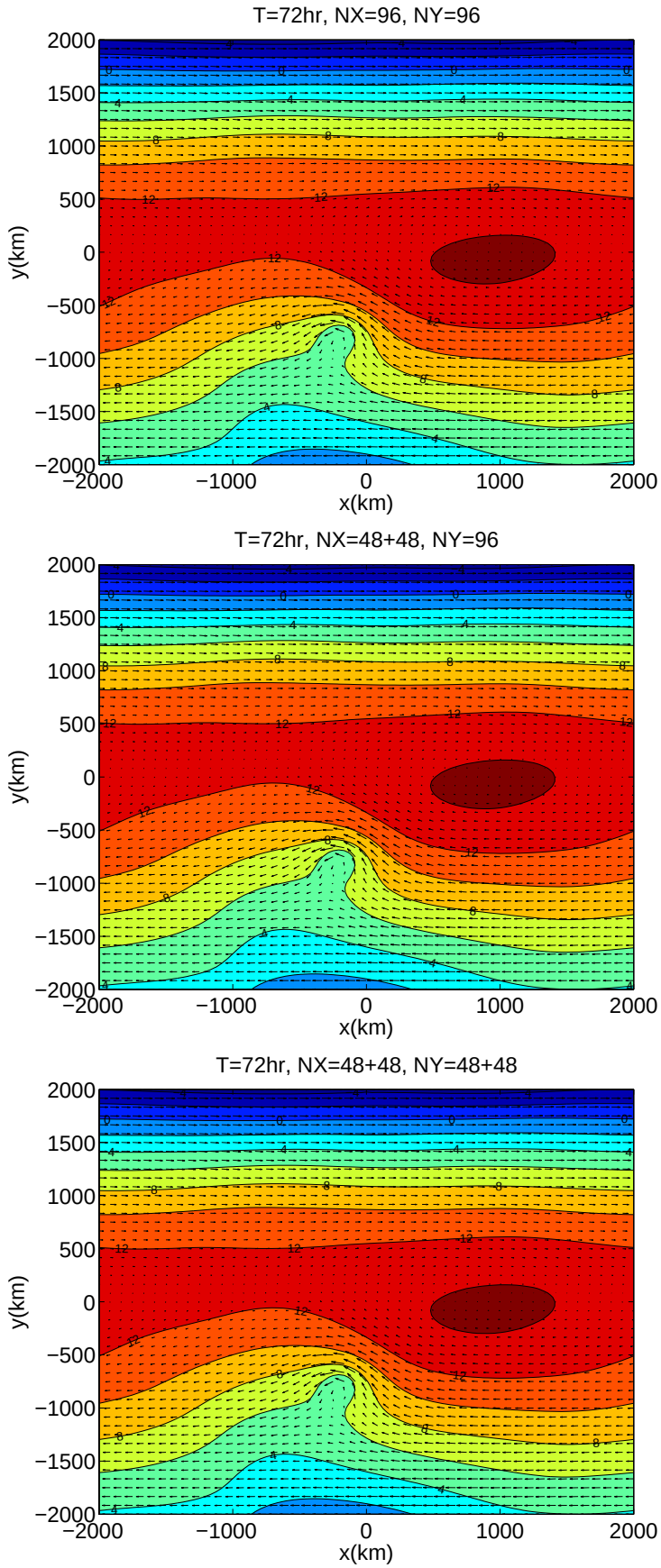

Figure 12: The results for shallow water model when T=72 hr. Up to down : single domain, DD 2×1, DD 2×2.

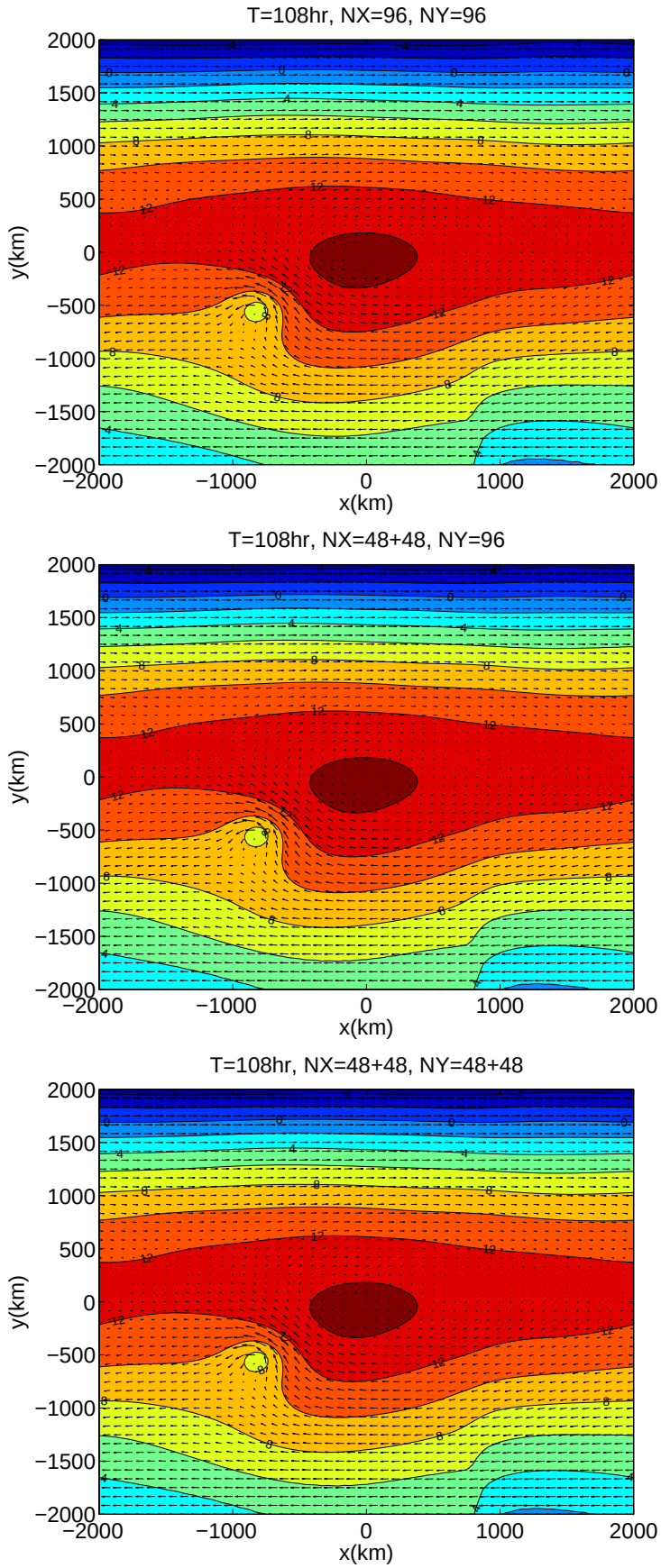

Figure 13: The results for shallow water model when T=108 hr. Up to down : single domain, DD 2×1, DD 2×2.

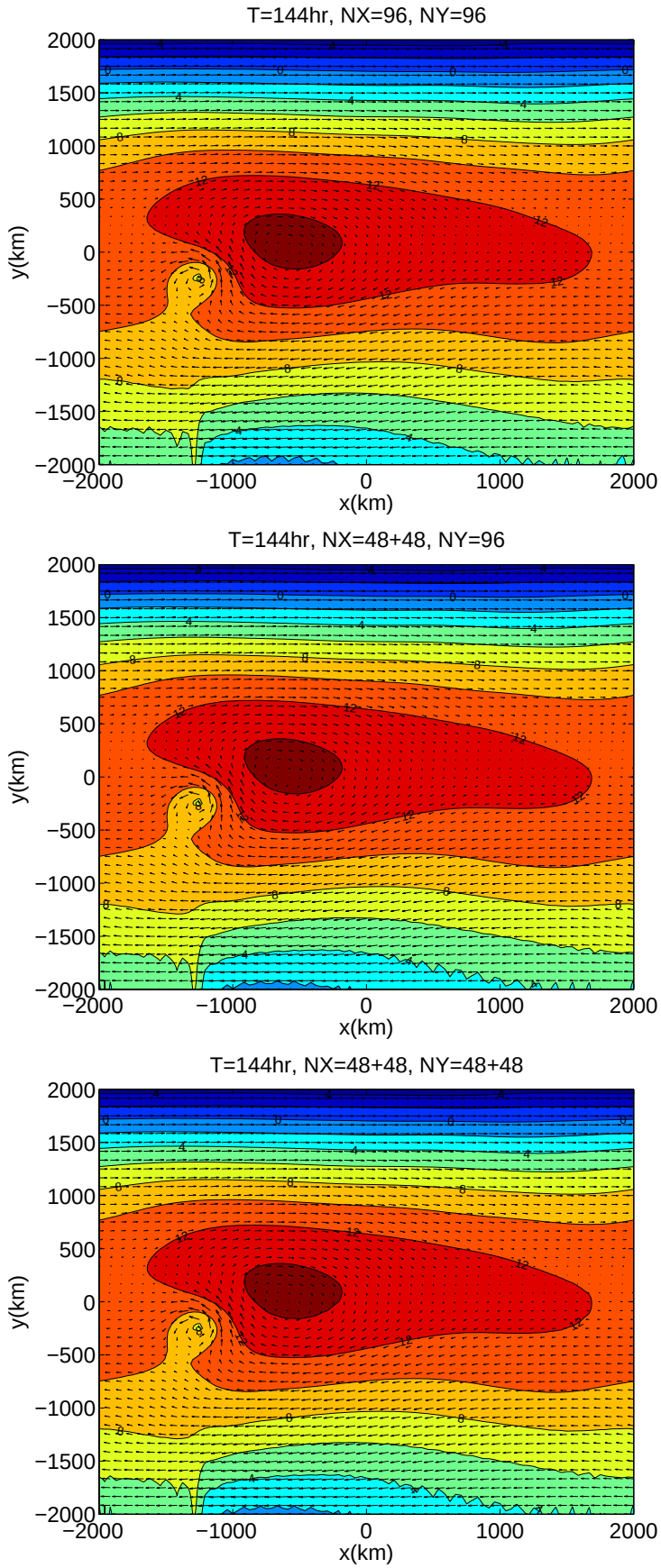

Figure 14: The results for shallow water model when T=144 hr. Up to down : single domain, DD 2×1, DD 2×2.

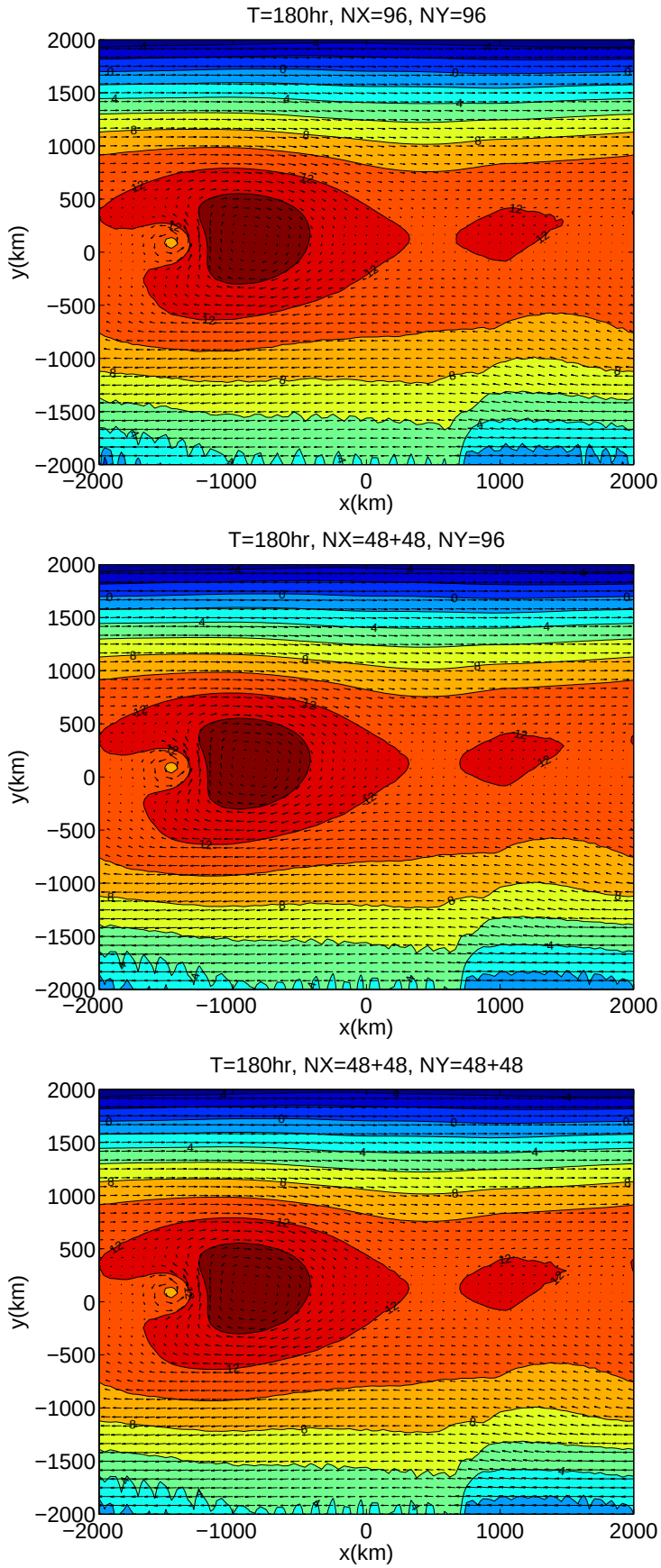

Figure 15: The results for shallow water model when T=180 hr. Up to down : single domain, DD 2×1, DD 2×2.

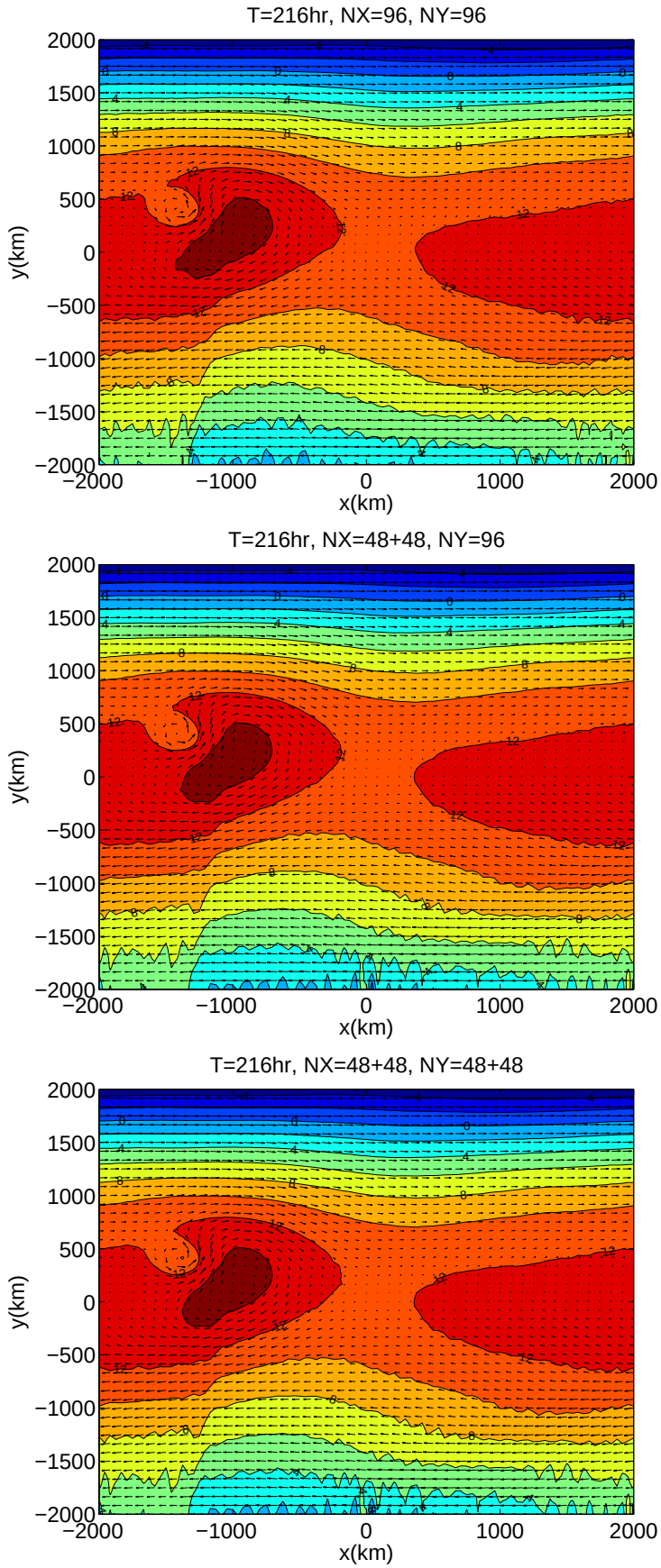

Figure 16: The results for shallow water model when T=216 hr. Up to down : single domain, DD 2×1, DD 2×2.

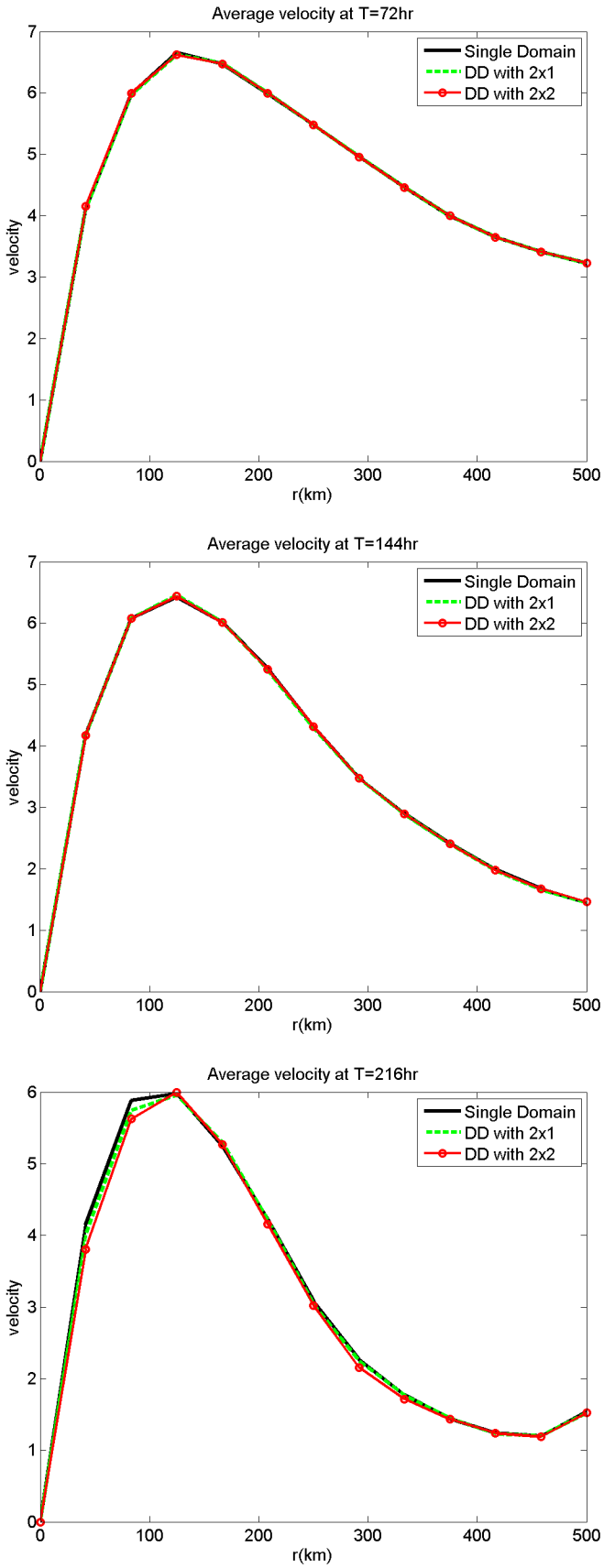

Figure 17: The comparison of average velocity near the vortex at $T = 72$ hr, $T = 144$ hr, $T = 216$ hr.

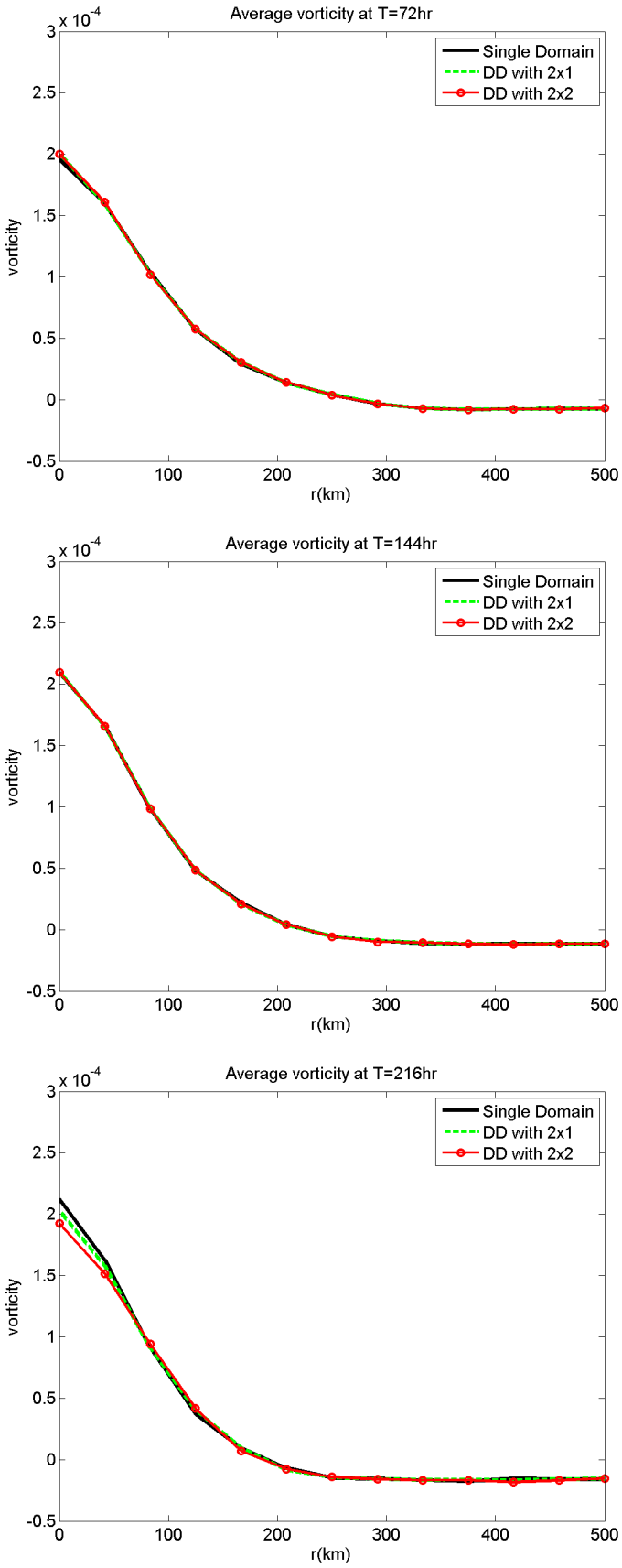

Figure 18: The comparison of average vorticity near the vortex at $T = 72$ hr, $T = 144$ hr, $T = 216$ hr.

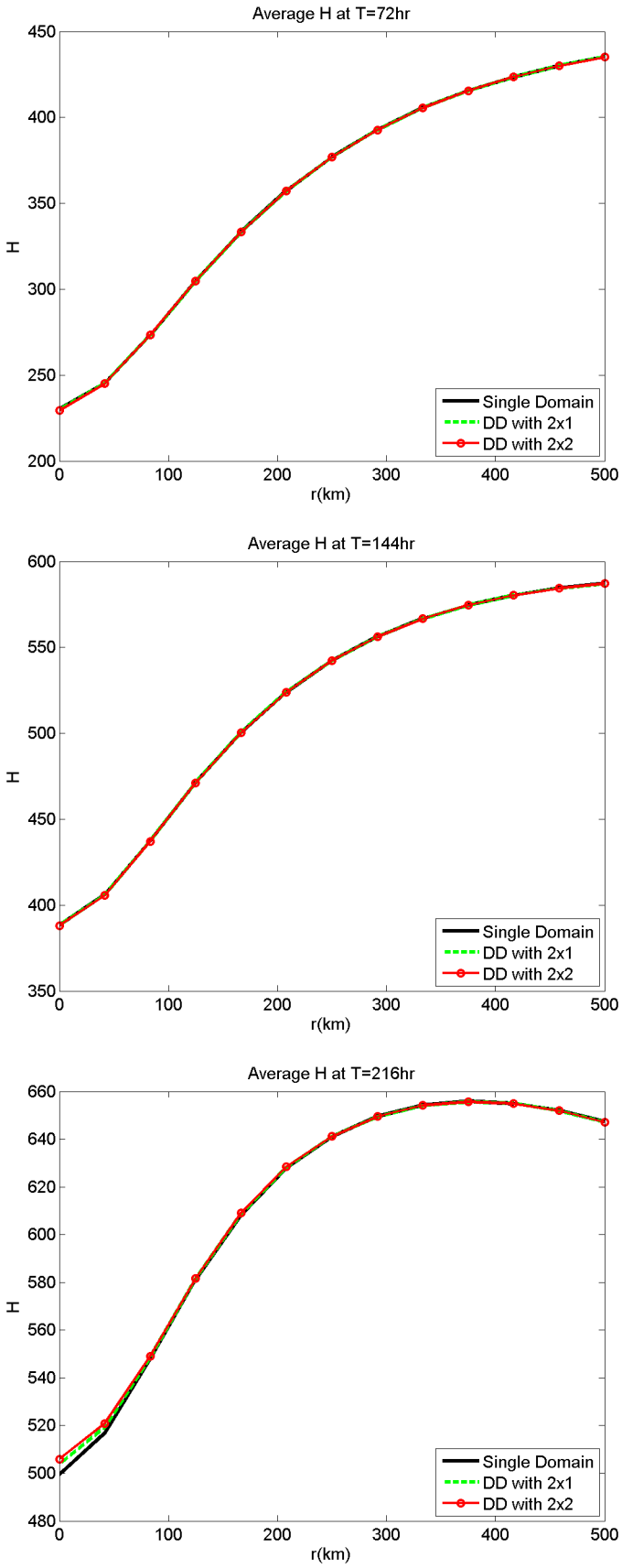

Figure 19: The comparison of average pressure near the vortex at $T = 72$ hr, $T = 144$ hr, $T = 216$ hr.

4.4 MPI implementation

To test and implement our numerical results on an MPI(Message-Passing Interface) environment, we use the DD 4×2 and DD 4×4. We initialize the MPI system in the beginning then the calculations are in different CPUs by the assignment of program. The information has to be exchanged between different CPUs after the integration of each time step has been finished. Here is the basic pseudo code of exchanging information at overlapping boundaries between southern and northern sub-domains:

```
 call MPI_SENDRECV(un(0,ny-numOver),nx+1,MPI_REAL8,south,1,
+         un(0,0),nx+1,MPI_REAL8,north,1,MPI_COMM_WORLD,
+         status,ierr)

 call MPI_SENDRECV(un(0,numOver),nx+1,MPI_REAL8,north,1,
+          un(0,ny),nx+1,MPI_REAL8,south,1,MPI_COMM_WORLD,
+         status,ierr)
```

About the overlapping boundaries between western and eastern sub-domains, because of the structure of Fortran array, we have to define a datatype first aimed to exchange information appropriately.

```
 call MPI_TYPE_VECTOR(ny+1,1,nx+2,MPI_REAL8,column,ierr)
 call MPI_TYPE_COMMIT(column,ierr)
```

The new datatype *culumn* specifies the method of picking data. Then the basic peudo code of southern and northern sub-domains is given here:

```
 call MPI_SENDRECV(un(numOver,0),1,column,east,1,
+         un(nx,0),1,column,west,1,MPI_COMM_WORLD,status,ierr)
 call MPI_SENDRECV(un(nx-numOver,0),1,column,west,1,
+         un(0,0),1,column,east,1,MPI_COMM_WORLD,status,ierr)
```

The results in Figure 20 to 25 are computed by the Chebyshev collocation method about the eq.(4.30) at 1.5, 3, 4.5, 6, 7.5 and 9 days respectively as labeled with MPI programming. They are very similar with Figure 11 to 16. Thus, we conclude our domain decomposition method is suitable for MPI environment.

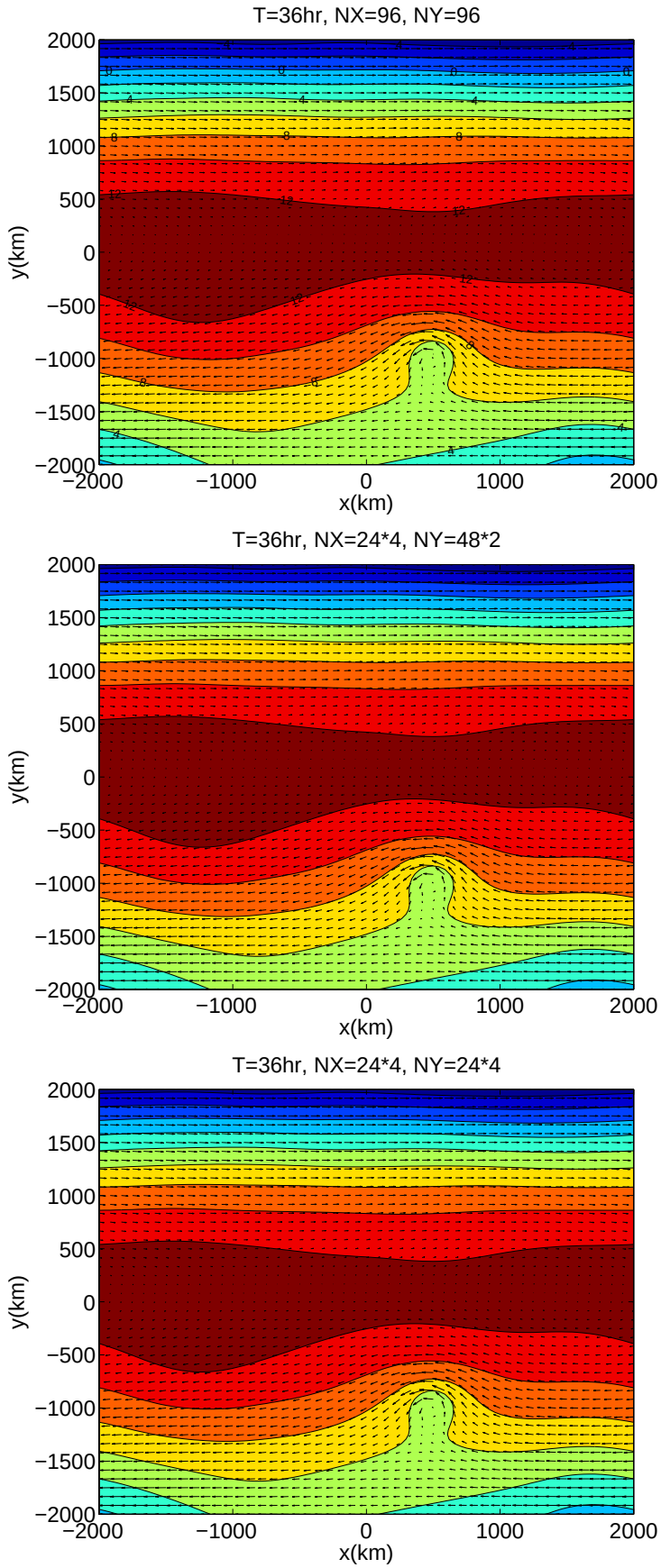

Figure 20: The results for shallow water model when T=36 hr. Up to down : single domain, DD 4×2, DD 4×4.

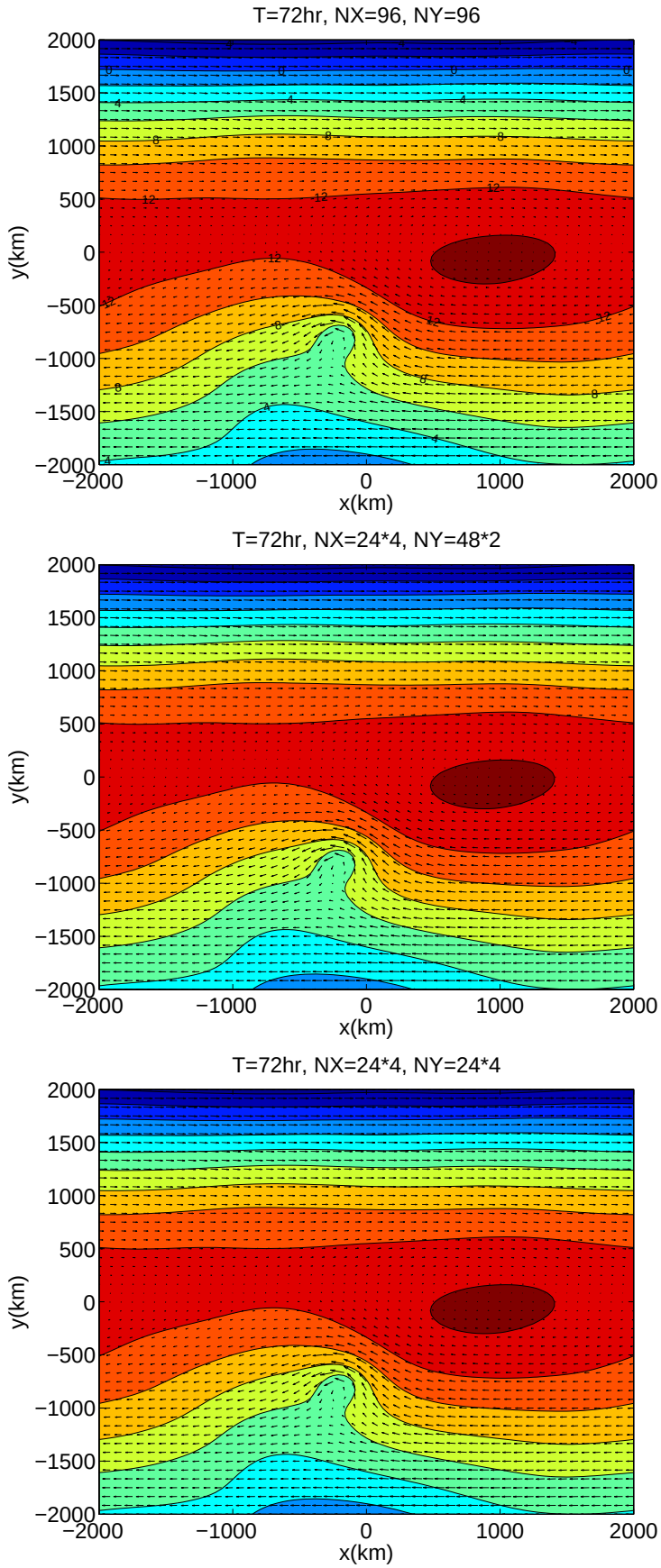

Figure 21: The results for shallow water model when T=72 hr. Up to down : single domain, DD 4×2, DD 4×4.

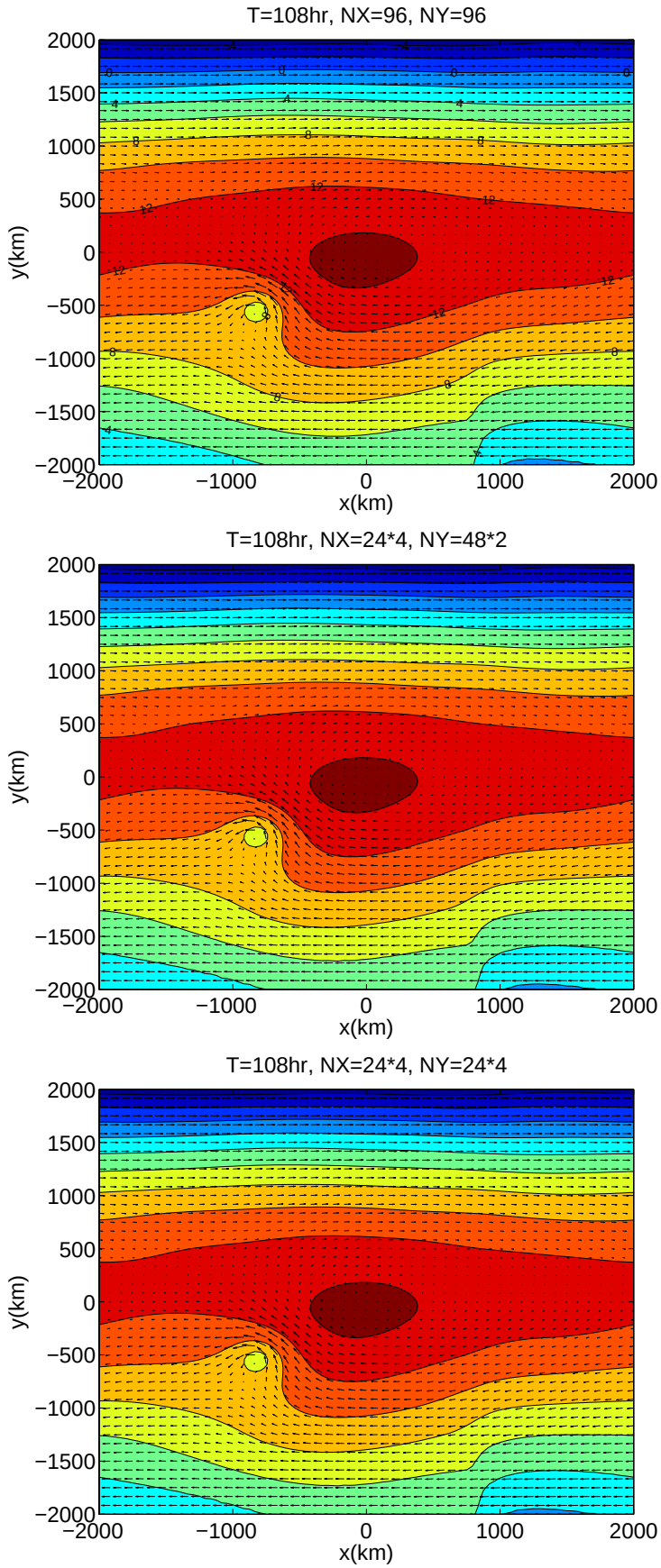

Figure 22: The results for shallow water model when T=96 hr. Up to down : single domain, DD 4×2, DD 4×4.

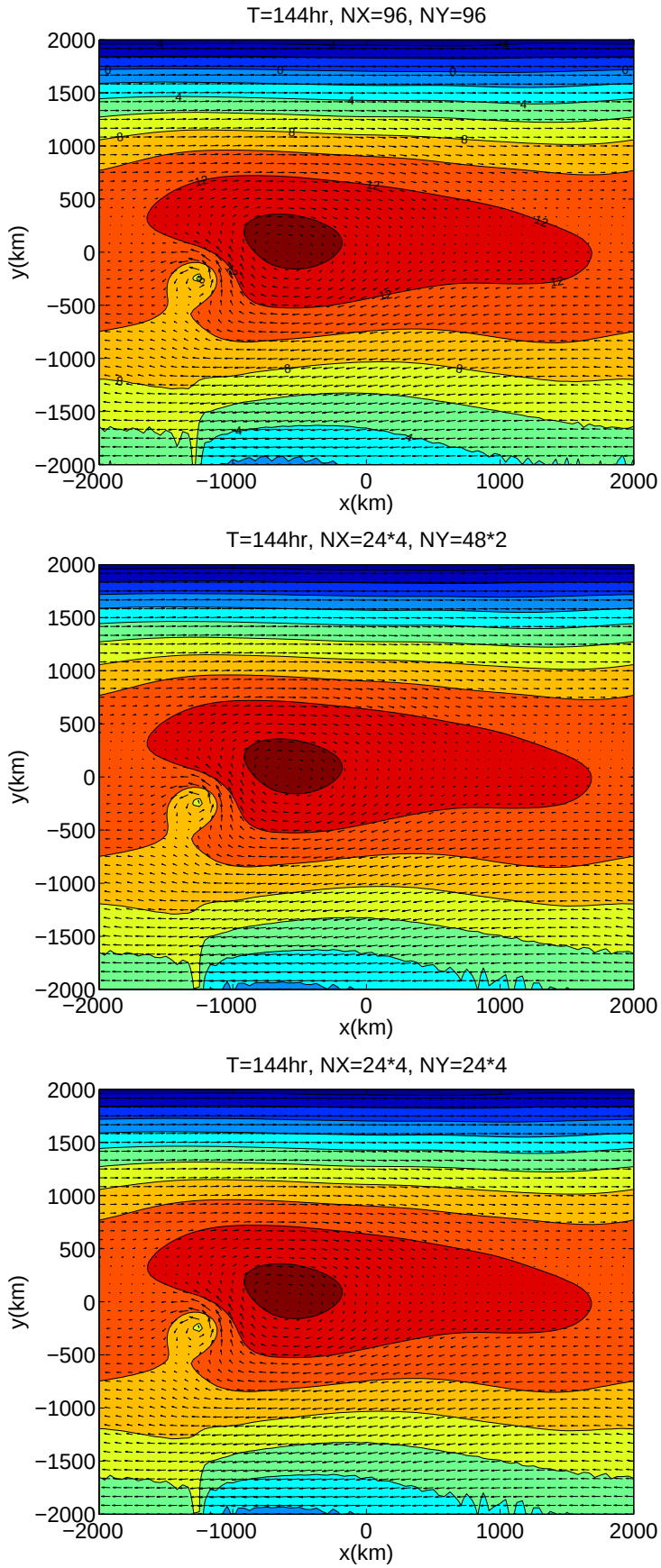

Figure 23: The results for shallow water model when T=144 hr. Up to down : single domain, DD 4×2, DD 4×4.

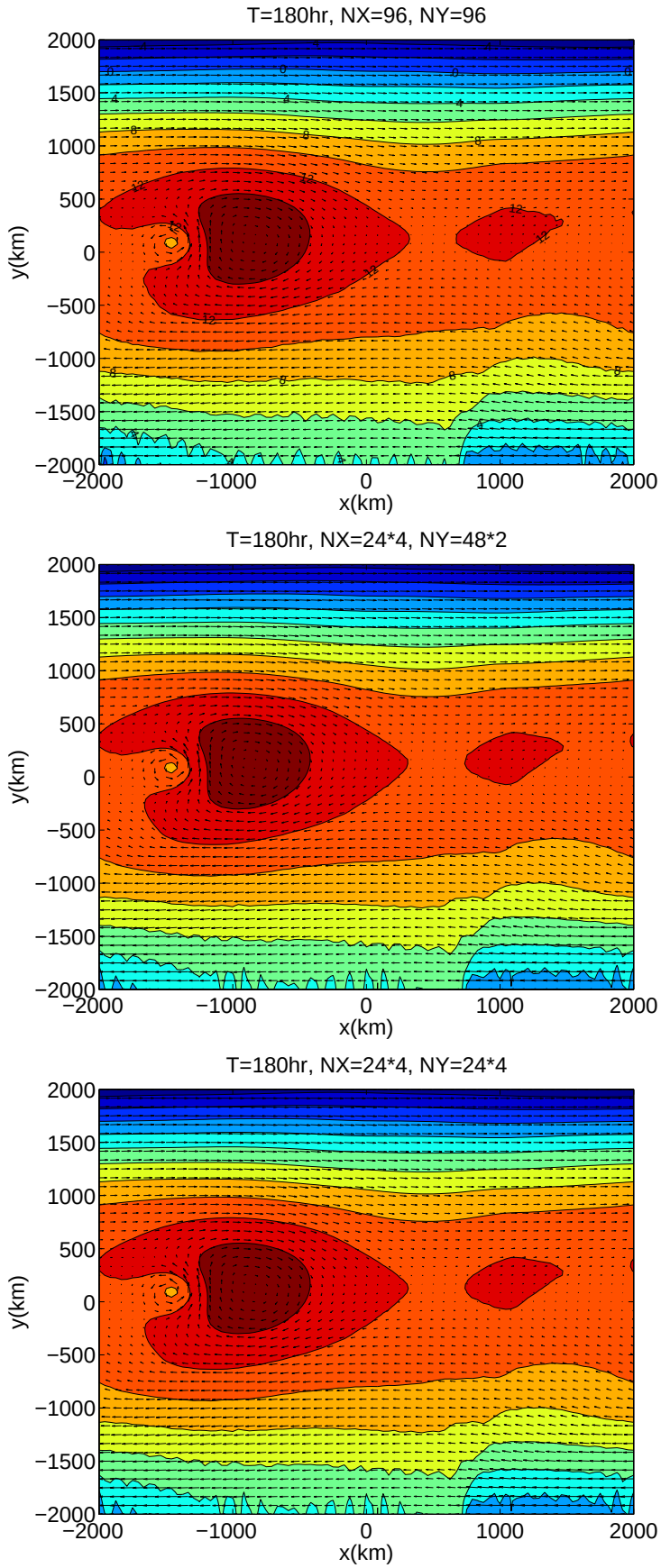

Figure 24: The results for shallow water model when T=180 hr. Up to down : single domain, DD 4×2, DD 4×4.

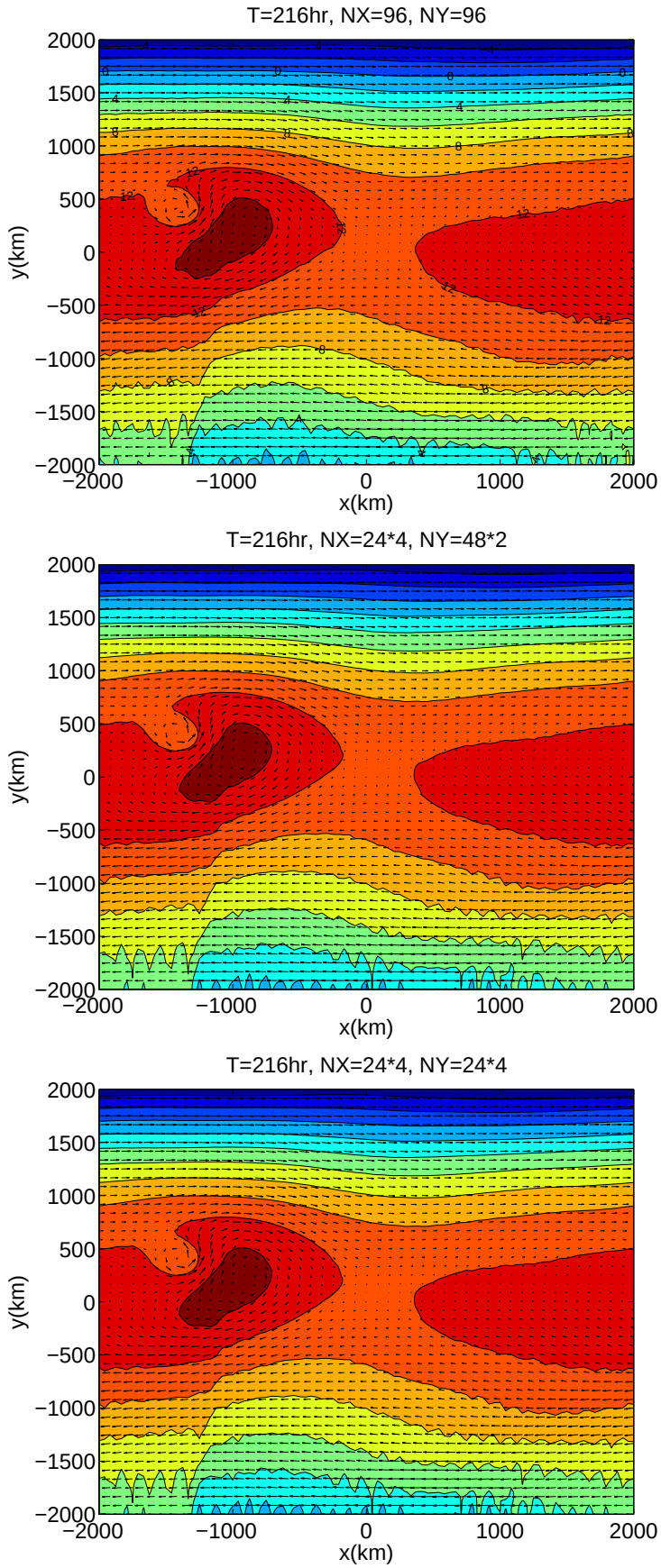

Figure 25: The results for shallow water model when T=216 hr. Up to down : single domain, DD 4×2, DD 4×4.

4.5 Time splitting method

In this thesis, we mainly discuss the domain decomposition method which could save the computation time. The shallow water equation allows multiple scales of oscillations in the model. There are slow type of motions such as vortex drifting (e.g., $7.5ms^{-1}$), Rossby waves and fast type of motion such as gravity waves (e.g., $50ms^{-1}$). We are usually interested in the slow motion and yet the fast motion limit the use of our time step size. In another words, the time step, as far as the slow type of motion is concerned, is limited by the stability and not by the accuracy. It is often treated in the atmospheric cloud models with the time splitting method for the efficiency. The essence of the time splitting is that use small time step for the terms in the equation govern the fast motion and use a larger time step for the terms govern the slow motion. Consequently, the time splitting method, in addition to the domain decomposition, can also improve our efficiency. Discussion on the time splitting method can be found in [14].

For the implementation of the time-splitting method in the shallow water model, we divided the equations into two parts : advection and gravity waves parts. eq.(4.30) can be written as

$$\begin{aligned} \frac{\partial u}{\partial t} &= U_a + U_g \\ \frac{\partial v}{\partial t} &= V_a + V_g \\ \frac{\partial h}{\partial t} &= H_a + H_g \end{aligned} \tag{4.35}$$

where the U_a, V_a, H_a dominate the advection, and

$$\begin{aligned} U_a &= -u\frac{\partial u}{\partial x} - v\frac{\partial u}{\partial y} + fv \\ V_a &= -u\frac{\partial v}{\partial x} - v\frac{\partial v}{\partial y} - fu \\ H_a &= -u\frac{\partial h}{\partial x} - v\frac{\partial h}{\partial y} - Q(x,y), \end{aligned} \tag{4.36}$$

the U_g, V_g, H_g are about gravity waves, and

$$\begin{aligned} U_g &= -\frac{\partial h}{\partial x} \\ V_g &= -\frac{\partial h}{\partial y} \\ H_g &= -(\overline{h} + h)(\frac{\partial u}{\partial x} - \frac{\partial v}{\partial y}) \end{aligned} \tag{4.37}$$

Thus, we introduce two different time steps Δt and $\Delta\tau$ where $\Delta\tau = \Delta t/n_s$. After the number n_s is decided, about the terms $\partial h/\partial x$, $\partial h/\partial y$, $\partial u/\partial x$, $\partial v/\partial x$ which cause the gravity waves, the smaller time step $\Delta\tau$ is used to do the integration. While the integration has been done for $n_s\Delta\tau$, the advection terms add to them and do the integration together. When $t = j\Delta t + k\Delta\tau$ and k is not divisible by n_s, we use the results of the advection term at $t = j\Delta t$.

Unfortunately, the predictability of this model by time-splitting method is weaken than without time-splitting, i.e., all the models blow up earlier than the previous results which with no time-splitting. All of them blow up earlier than 4 days.

There is also an important observation that why the time-splitting method for shallow water model could not be very efficient. In this model, 6 terms have to be calculate by Chebyshev collocation transform, they are $\partial h/\partial x$, $\partial h/\partial y$, $\partial u/\partial x$, $\partial v/\partial y$, $\partial u/\partial y$ and $\partial v/\partial x$. But there are 4 terms belonged to the gravity waves. Even though the time-splitting method can implemented, the time saving is just 1/3 and additional complication cause the model to loses the stability at the same time. It is recommended that the use of the time splitting method required carefully analysis in future.

5 Concluding Remarks and Future Works

We have introduced the Chebyshev collocation method with domain decomposition in the atmospheric modelings. The sub-domain boundary information exchange is by overlapping the sub-domains in one grid spacing interval. By the property of Chebyshev grids setting and consider the relation between Δx and Δt with CFL condition, we can enlarge the Δt with Chebyshev domain decomposition compared with single domain. Our domain decomposition Chebyshev collocation method indicates the exponential convergence property in 1-D linear advection and diffusion models. In the test of inviscid Burgers equation, we integrate the model up to the shock formation time. We show that the domain decomposition spectral method in general yields a smaller errors when compared to the single domain calculations. In a more realistic atmospheric modeling with a 2-D shallow water model, we find our domain decomposition Chebyshev method gives results identical to the single domain spectral method with a L_2 error on the order of 10^{-4} when 96 degree of freedom is considered. The domain decomposition spectral method is capable of a stable integration of 9 days in our test. It is prominent, considering the fact that the predictability of the typical atmospheric model is about 10 to 12 days. We also argued that the time-splitting method is not well applicable to the 2-D shallow water equation.

Our future work will be evaluate the overhead or the additional cost of the boundary information exchange in domain decomposition. We also may implement the method in the oceanic modeling by incorporating the immerse boundary condition method in the lateral continental shelf and using the Chebyshev domain decomposition method.

Appendix

A Verticle Transform and Shallow Water Equation

In this appendix we will show that hydorstatic atmosphere is equivalent to a set of shallow water equation by the vertical transform.

$$s = s(p) \equiv \frac{c_p \theta_0}{g}(1 - (\frac{p}{p_0})^\kappa) \tag{A.1}$$

$$\kappa = \frac{R}{c_p} \tag{A.2}$$

$$b = \frac{g}{\theta_0}\theta \tag{A.3}$$

Linearized equation in s coordinate (with J represents diabatic heating).

$$\frac{\partial \overrightarrow{v}}{\partial t} + f\hat{k} \times \overrightarrow{v} = -\nabla\Phi \tag{A.4}$$

$$\nabla \cdot \overrightarrow{v} + \frac{\partial \dot{s}}{\partial s} = 0 \tag{A.5}$$

$$\frac{\partial \Phi}{\partial s} = b \tag{A.6}$$

$$\frac{\partial b}{\partial t} + \dot{s}N^2 = J \tag{A.7}$$

Boundary condition: $w = dz/dt = 0$ at $s = 0$, $\dot{s} = dp/dt = 0$ at $s = H$.
$\frac{\partial \tilde{\Phi}}{\partial t}$ is the local height change that can be resulted from diabatic heating.
Boundary condition derivation.

$$J = \frac{\partial}{\partial s}(\frac{\partial \tilde{\Phi}}{\partial t}) \tag{A.8}$$

Substitute A.6 and A.8 into A.7 we could obtain:

$$\dot{s} = -\frac{1}{N^2}\frac{\partial}{\partial s}(\frac{\partial \Phi}{\partial t} - \frac{\partial \tilde{\Phi}}{\partial t}) \tag{A.9}$$

At $s = 0$, $w = dz/dt = 0$,

$$\frac{d\Phi}{dt} = 0 \tag{A.10}$$

$$\frac{\partial \Phi}{\partial t} + \dot{s}\frac{\partial \overline{\Phi}}{\partial s} = 0 \tag{A.11}$$

$$\frac{\partial \Phi}{\partial t} + b_0 \dot{s} = 0 \tag{A.12}$$

Substitute A.9 into A.12, and with $\frac{\partial \tilde{\Phi}}{\partial t} = 0$ at $s = 0$ we could obtain:

$$(\frac{\partial \Phi}{\partial t} - \frac{\partial \tilde{\Phi}}{\partial t}) - \alpha\frac{\partial}{\partial s}(\frac{\partial \Phi}{\partial t} - \frac{\partial \tilde{\Phi}}{\partial t}) = 0 \tag{A.13}$$

where

$$\alpha = \frac{b_0}{N^2}. \tag{A.14}$$

At $s = H$, $\dot{s} = dp/dt = 0$,

$$\frac{\partial}{\partial s}(\frac{\partial \Phi}{\partial t} - \frac{\partial \tilde{\Phi}}{\partial t}) = 0. \tag{A.15}$$

Inner product:

$$\frac{1}{H}\int_0^H uv \; ds = \langle u, v \rangle \tag{A.16}$$

$$\langle \mathcal{L}u, v \rangle = \langle u, \mathcal{L}v \rangle \tag{A.17}$$

Based on the Sturm-Liouville theorem, the basis function is complete, orthogonal, and the eigenvalue is real.

$$\mathcal{L}\Psi_n = \lambda_n \Psi_n \tag{A.18}$$

$$\Psi_n - \alpha \frac{\partial \Psi_n}{\partial s} = 0 \tag{A.19}$$

$$\frac{\partial \Psi_n}{\partial s} = 0 \tag{A.20}$$

If $\lambda_n = \frac{1}{c_n^2} > 0$, we could obtain the shallow water equation.

$$\frac{\partial \overrightarrow{v_n}}{\partial t} + f\hat{k} \times \overrightarrow{v_n} = -\nabla \Phi_n \tag{A.21}$$

$$\frac{\partial \Phi}{\partial t} + c_n^2 \nabla \cdot \overrightarrow{v_n} = -\frac{\partial \tilde{\Phi}}{\partial t} \tag{A.22}$$

The above derivation demonstrates that a hydrostatic atmosphere with suitable vertical boundary conditions may support free oscillations with several different structurs. The eigenvalue of each free oscillation is $c_n^2 = g\overline{h}$, which is related to the depth of the shallow water equation.

B Amdahl's Law

In this appendix, we will discuss the Amdahl's law in parallel computing. Let W be the amount of work to be done for a particular job, and let r be the rate at which it can be done by one processor. Then the computer time required for one processor to do the job is T_1, given by

$$T_1 = \frac{W}{r} \tag{B.1}$$

Now suppose that f fraction of the job, by time, must be done serially and the remaining $1 - f$ fraction can be done perfectly parallelized by p processors. Then the time, T_p, for parallel computation is given by

$$T_p = \frac{fW}{r} + \frac{(1-f)W}{pr} \tag{B.2}$$

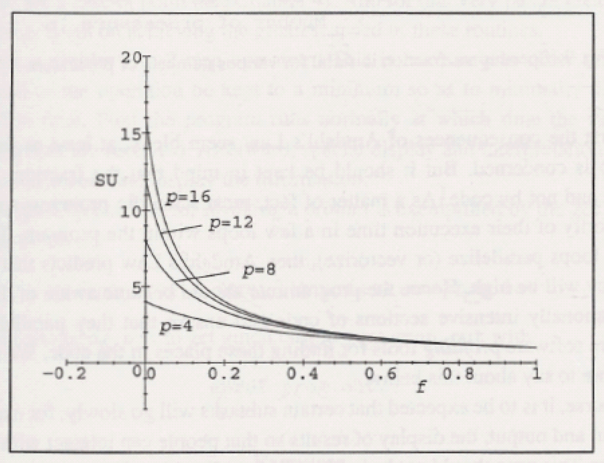

Figure B.1: Amdahl speedup as a function of f.

The above equation indicates that if the entire calculation can be parallelized, that is, $f = 0$, then all the work will be done in p fraction of the time. We then claim the *speedup* SU is p, and

$$SU = \frac{T_1}{T_p} = p \tag{B.3}$$

This is the well known *linear speedup*. But as the equation indicate, the speedup in general will be

$$SU = \frac{T_1}{T_p} = \frac{W/r}{(W/r)(f + \frac{(1-f)}{p})} = \frac{p}{f(p-1)+1}. \tag{B.4}$$

This relation is known in the field of Parallel Computing as the *Amdahl's Law*. We are interested from the above equation the speed up SU as a function of numbers of processors p. In particular, we want to know how the SU behave as a function of p and f. Figure B.1 shows the Amdahls speed up as a function of f for various p. It is obvious that the steepness near $f = 0$ means that the speedup falls off rapidly for the increase of f. For example, the SU does not change much with p processors for $f = 0.2$. Namely, the SU becomes insignificant when percentage of code that cannot be parallelized is about 20%. It may appear that the Amdahl's Law gives a bleak picture as far as the speedup is concerned. However, the fraction f is defined by computational time and not by computational code. As a matter of fact, most scientific programs spend the majority of their execution time in a few loops within the program. Thus if these loops parallelize (or vectorize), then Amdahl's Law predicts that the efficiency will be high. On the other hand, if we employed the domain decomposition method, the theoretical SU will be almost proportional to the number of processors p with the overhead of information exchange through the decomposed boundaries. The SU in the domain decomposition in general is not a function of the f. More details are on [18].

References

[1] Orszag, S.A., "Transform Method for calculation of Vector-coupled Sums: Application to the Spectral Form of the Vorticity equation" *Journal of the Atmospheric Sciences*, **Vol. 27**, pp. 890-895 (1970).

[2] Bourke, W., B. McAvaney, K. Puri and R. Thurling, "Global Modeling of Atmospheric Modeling of Atmospheric Flow by Spectral Methods." *Methods in Computational Physics*, **Vol. 17**, Academic Press, pp. 267-324 (1977).

[3] Machenhauer, B., "The Spectral Method." *Numerical Methods Used in Atmospheric Models*, **Vol. 2**, No. 17, pp. 121-275 (1979).

[4] Tatsumi, Y., "A Spectral Limited-area Model with Time-dependent Lateral Boundary Conditions and Its Application to a Multi-level Primitive Equation Model." *Meteorological Society of Japan, Journal*, **Vol. 64**, pp. 637-664 (1986).

[5] Scott R. Fulton and Wayne H. Schubert, "Chebyshev Spectral Methods for Limited-Area Models. Part I: Model Problem Analysis" *Monthly Weather Review*, **Vol. 115**, pp. 1940-1953 (1986).

[6] Scott R. Fulton and Wayne H. Schubert, "Chebyshev Spectral Methods for Limited-Area Models. Part II: Shallow Water Model" *Monthly Weather Review*, **Vol. 115**, pp. 1954-1965 (1986).

[7] Michele G. Macaraeg and Craig L. Streett, "Improvements in Spectral Collocation Discretization Through a Multiple Domain Technique" *Applied Numerical Mathematics 2*, pp. 95-108 (1986).

[8] David A. Kopriva, "A Spectral Multidomain Method For the Solution of Hyperbolic Systems" *Applied Numerical Mathematics 2*, pp. 221-241 (1986).

[9] David A. Kopriva, "Computation of Hyperbolic Equations on Complicated Domains with Patched and Overset Chebyshev Grids" *SIAM Journal on Scientific and Statistical Computing*, **Vol. 10**, No. 1, pp. 120-132 (1989).

[10] David A. Kopriva and John H. Kolias, "A Conservative Staggered-Grid Chebyshev Multidomain Method for Compressible Flows" *Journal of Computational Physics*, **Vol. 125**, pp. 244-261 (1996).

[11] David A. Kopriva, "A Conservative Staggered-Grid Chebyshev Multidomain Method for Compressible Flows. II. A Semi-Structured Method" *Journal of Computational Physics*, **Vol. 128**, No. 3, pp. 475-488 (1997).

[12] Henry H. Yang and Bernie Shizgal, "Chebyshev Pseudospectral Multi-domain Technique for Viscous Flow Calculation" *Computer Methods in Applied Mechanics and Engineering*, **Vol. 118**, pp. 47-61 (1995).

[13] J. S. Hesthaven, "A Stable Penalty Method For the Compressible Navier-Stokes Equations: II. One-Dimensional Domain Decomposition Schemes" *SIAM Journal on Scientific Computing*, **Vol. 18**, No. 3, pp. 658-685 (1999).

[14] Louis J. Wicker and William C. Skamarock, "A Time-Splitting Scheme for the Elastic Equations Incorporating Second-Order Runge-Kutta Differencing" *Monthly Weather Review*, **Vol. 126**, pp. 1992-1999 (1998).

[15] Lanczos, C., "Applied Analysis" *Prentice-Hall*, pp 539 (1956).

[16] Gottlieb,D., and S. A. Orszag, "Numerical Analysis of Spectral Methods" *NSF-CBMS Monogr.*, No. 26, pp. 172 (1977).

[17] Courant, R., and D. Hilbert, *Methods of Mathematical Physics*, **Vol. 1**. Wiley-Interscience, pp. 561 (1953).

[18] Ronald W. Shonkwiler, and Lew Lefton, *An Introduction to Parallel and Vector Scientific Computing*, Cambridge University Press, pp. 20-23 (2006).