Multigrid and Cubic Spline Collocation Methods for Advection Equations

by

Zheng Zeng

A thesis submitted in conformity with the requirements
for the degree of Master of Science
Graduate Department of Computer Science
University of Toronto

Abstract

Multigrid and Cubic Spline Collocation Methods for Advection Equations

Zheng Zeng

Master of Science

Graduate Department of Computer Science

University of Toronto

2005

This thesis describes numerical methods for the solution of advection equations in one-dimensional space. The methods are based on combining the multigrid and cubic spline collocation (CSC) methodologies. Multigrid methods for cubic splines are presented. Appropriate restriction and extension operators are developed for cubic splines that satisfy various boundary conditions (BCs), i.e., Dirichlet, Neumann, periodic and general BCs. The multigrid methods are applied to CSC equations arising from the discretization of one-dimensional second-order differential equations. The convergence analysis of a two-grid method integrated with a damped Richardson relaxation scheme as smoother shows the rate of convergence is equal to or faster than $1/2$ for a model second order equation with periodic BCs. Multigrid methods for cubic splines are then extended to the solution of one-dimensional shallow water equations (SWEs). The SWEs are discretized in time with a semi-Lagrangian semi-implicit scheme and in space with CSC methods. At each time step, there are three different space discretization approaches: Analytic Elimination - Discretization (AED), Discretization -Algebraic Elimination (DAE), and Discretization - No Elimination (DNE). We discuss advantages and disadvantages of each, and develop new numerical methods based on multigrid and CSC for solving the SWEs discretized with either the AED or the DNE approaches. We compare our methods with Jacobi's iterative method applied to the same problems. Through comparison and convergence discussion, we show that our multigrid methods for CSC are convergent and efficient. The numerical results confirm our analysis.

Dedication

To my parents and dear sister

Acknowledgements

First of all, I would like to thank my supervisor Professor Christina Christara for her kindness, hard working, patience and dedication in guiding me throughout this thesis. She was always there when I needed help.

Secondly, I would ike to thank Professor Wayne Enright for being the second reader and providing valuable comments.

Thirdly, I would like to thank Professor Anita Layton from Duke University for her helpful comments on the experimental results, and Professor Justin Wan from University of Waterloo for his help on multigrid methods.

I would also like to thank University of Toronto and Professor Christina Christara for the financial support.

Last but not least, I would like to thank all my friends, particularly Vicky, Jessi, Qingqing, Yongyi, Michael, Daniel, Meng and Chloe for the great time we had together. Memories last forever ...

Contents

List of Figures

Chapter 1

Introduction

Numerical weather prediction (NWP) not only provides an important guidance to the short-term weather forecasts or natural disaster prediction, but also helps us understand the long-term behaviors and effects of the atmosphere, ocean water and sea ice. Since the solution of a NWP model is computationally very expensive and time-consuming, there is tremendous interest in the scientific community in finding accurate and efficient numerical methods. A simple but realistic model for NWP is based on the shallow water equations (SWEs), which are derived from simple forms of the equations of motion, and describe the inviscid flow of a thin layer of fluid in two dimensions, where the vertical length scale is much smaller than the horizontal length scale [23]. The solution of SWEs has long been the focus of study and used for development of promising numerical methods for solving atmospheric and oceanic problems.

Spectral methods, finite difference methods and finite element methods are the most commonly used spatial discretization methods for the integration of NWP models. Spectral methods produce high order solutions, but suffer from prohibitive expense at high resolution associated with the cost of the Legendre transforms, and from limited scalability in the parallel implementation due to the arising dense system [5]. On the other side, finite difference and finite element methods currently used are limited to low-order methods [18, 22, 6]. One way to achieve a more accurate solution is to use a high order discretization scheme. Recently, Layton, Christara and Jackson used high order methods, such as quadratic spline Galerkin [14] and quadratic spline collocation methods [13]. Layton [12] uses cubic spline collocation as a high order spatial discretization method. These pieces of work demonstrate that high order finite element methods are viable alternatives to spectral methods. However, when discretizing NWP models on realistic domains with high order finite element methods, one eventually faces the problem of solving large but sparse linear systems. The large size of the system makes direct methods (e.g., LU factorization with forward and backward substitutions) quite unsuitable. Classical iterative methods converge slowly on these type of systems. Multigrid methods, which are known to be efficient for the iterative solution of linear systems arising from the discretization of partial differential equations (PDEs), may provide a way to solve the systems up to a prescribed accuracy in a fixed number of iterations. Multigrid methods have primarily been developed and studied for linear systems arising from finite difference or Galerkin type finite element methods. Related work for spline collocation methods is relatively limited [10]. Christara and Smith [4] develop multigrid methods for quadratic spline collocation and prove that the convergence of a two-grid method for one-dimensional boundary value problems is independent of the discretization step-size.

In this thesis, multigrid methods are developed for cubic splines. Appropriate restriction and extension operators for cubic splines that satisfy various boundary conditions (BCs) are proposed. The multigrid methods are applied to cubic spline collocation (CSC) equations arising from the discretization of one-dimensional second-order differential equations. We prove mathematically that the convergence of a two-grid method applied to a certain one-dimensional boundary value problem (BVP) with periodic BCs is independent of the discretization step-size. Then, we extend our multigrid methods for CSC to the solution of one-dimensional SWEs. The SWEs are first discretized in time with a semi-Lagrangian semi-implicit scheme, which allows large time step-size while maintaining stability. At each time step, a system of two PDEs with two unknown functions has to be solved. We use CSC to discretize the system of PDEs in space. There are three approaches to perform the CSC discretization, which give rise to three different types of linear systems. The first approach is what we call Analytic Elimination - Discretization (AED), where we analytically eliminate one unknown function variable, and obtain a Helmholtz equation, which is then discretized in space. The second approach is Discretization - No Elimination (DAE), where we first discretize the system of two PDEs with the CSC method, then algebraically eliminate one unknown function variable from the arising linear system, thus obtaining a half size linear system involving only one unknown function variable. The last approach is Discretization - No Elimination (DNE). Similarly to DAE, the DNE approach first discretizes the system of PDEs. But after discretization, instead of performing algebraic elimination, the DNE assumes the arising linear system is to be solved as one whole entity and leaves it as it is. Thus the DNE approach produces a linear system twice as big as those of the first two approaches. The AED approach gives rise to a symmetric and diagonal dominant linear system, but it does not preserve some physical properties of SWEs, e.g., the stability of Rossby waves [19] and [20]. The DAE does not suffer from the instability problem of the AED approach, but the linear system generated is nonstandard and computationally expensive to solve. The DNE approach enjoys the same nice stability properties as the DAE approach, but it gives rise to a larger (double) size linear system than the other two approaches. An efficient linear solver is critical for the success of the DNE approach. Fortunately, multigrid methods provide us a way to solve linear systems efficiently so that we could bear the burden of solving the increased linear system of the DNE approach for the reason of stability. In this thesis, we focus primarily on the DNE approach and develop new numerical methods based on multigrid methods and CSC for solving the one-dimensional SWEs. We compare the performance of multigrid when applied to the systems arising from the DNE and AED approaches. For comparison, we also implement Jacobi's iterative method and apply it to the same problems. We perform a convergence analysis and discuss advantages and disadvantages of each technique. Through the analysis and the numerical results, we demonstrate that the multigrid methods proposed are convergent and efficient even when using the DNE approach.

The rest of the thesis is organized as follows. In Chapter 2, we first provide the background materials regarding CSC and multigrid methods, then develop multigrid methods for cubic splines, in terms of defining the coarse grid correction scheme characterized by restriction and extension operators for different types of BCs. In Chapter 3, we apply multigrid methods for CSC to one-dimensional two-point second order boundary value problems (BVPs) with periodic BCs and prove that the two-grid method integrated with a damped Richardson relaxation scheme as smoother converges with a contraction factor less than or equal to $1/2$ in the Euclidean norm, independently of the discretization step-size. The convergence analysis of Jacobi's iterative method for the same problems is presented in the same chapter. The discretization of the one-dimensional SWEs is presented in Chapter 4. In particular, we describe the three approaches for space discretization of the SWEs by CSC, and discuss the properties of the arising linear systems. In the same chapter, multigrid methods for CSC are developed for solving one-dimensional SWEs with the DNE and AED approaches. A convergence analysis of the multigrid and Jacobi's iterative methods for the same problems is presented. The analysis leads us to believe that the multigrid methods are convergent and efficient. Numerical results are shown in Chapter 5 and confirm our analysis. Finally, we conclude in Chapter 6.

Chapter 2

Multigrid Methods for Cubic Spline Collocation

2.1 Cubic Spline Collocation

Cubic splines (piecewise cubic polynomials with C^2 continuity) are commonly used for approximating functions [16]. In this section, we review the cubic spline collocation method and introduce some discretization matrices arising from boundary value problems (BVPs) with various boundary conditions (BCs), i.e., Dirichlet, Neumann, periodic or general BCs.

Consider a BVP in a one-dimensional domain $\Omega \equiv (\omega_1, \omega_2)$ described by a linear second-order elliptic PDE,

$$\mathrm{L}u \equiv au_{xx} + bu_x + cu = g \text{ in } \Omega, \tag{2.1}$$

and some BCs defined on the boundary, $\partial\Omega$, of Ω,

$$\mathrm{B}u \equiv \alpha u_x + \beta u = \gamma \text{ on } \partial\Omega, \tag{2.2}$$

where a, b, c, g, α, β and γ are given functions of x, and u is the unknown function of x.

Let $\Delta \equiv \{\omega_1 = x_0 < x_1 < \ldots < x_n = \omega_2\}$ be a uniform partition of Ω and $h = \frac{\omega_2 - \omega_1}{n}$ be the discretization step-size.

Given the partition Δ of Ω, we choose as approximation space the space S_Δ of piecewise cubic polynomials of continuity C^2 on the nodes of partition Δ. A set of basis functions for S_Δ is given by $\{\phi_i^h\}_{i=-1}^{n+1}$, where

$$\phi_i^h = \phi(\frac{x-\omega_1}{h} - i + 2), i = -1, \ldots, n+1, \tag{2.3}$$

and the cubic spline model function ϕ is defined by

$$\phi(x) = \frac{1}{6}\begin{cases} x^3 & \text{for } 0 \le x \le 1 \\ x^3 - 4(x-1)^3 & \text{for } 1 \le x \le 2 \\ x^3 - 4(x-1)^3 + 6(x-2)^3 & \text{for } 2 \le x \le 3 \\ x^3 - 4(x-1)^3 + 6(x-2)^3 - 4(x-3)^3 & \text{for } 3 \le x \le 4 \\ 0 & \text{elsewhere.} \end{cases} \tag{2.4}$$

Any cubic spline $S \in S_\Delta$ can be written as $S(x) = \Sigma_{i=-1}^{n+1} c_i^h \phi_i^h(x)$, where the scalars $c_i^h, i = -1, \ldots, n+1$, are called *coefficients* or *degree of freedoms* (DOFs). Note that $n+3$ DOFs are required to uniquely define $S(x)$.

In order to uniquely determine the $n+3$ DOFs, $n+3$ linearly independent conditions are required. Therefore, we define a set of data points, called *collocation* points, and impose appropriate conditions at the points. The cubic spline collocation method determines a piecewise cubic polynomial approximation $u_\Delta \in S_\Delta$ to u by requiring that the residuals $\mathrm{L}u - g$ and $\mathrm{B}u - \gamma$ of the differential operator L and boundary operator B, respectively, are zero at the collocation points. In our work, we choose the nodal points $x_0, x_1, \ldots, x_n$ of the partition as collocation points. In the rest of the thesis, for convenience, we denote $f(x_0), f(x_1), \ldots, f(x_n)$ by $f_0, f_1, \ldots, f_n$, respectively, where f can be any function of one variable, e.g., g, a, γ, etc.

2.1.1 Boundary Conditions

There are some cases in which we can adjust the basis functions to incorporate the boundary conditions into the approximation space. Methods using such adjusted basis functions are referred to as *interior* collocation methods. Usually, interior collocation methods produce more efficient formulations; more specifically, the linear system arising is smaller and most importantly, the matrix usually possesses better properties, such as diagonal dominance, etc. In the next subsection, we will be seeing various CSC matrices arising from BVPs subject to homogeneous Dirichlet, homogeneous Neumann and periodic BCs with unadjusted and adjusted basis functions, which respectively correspond to the general and the interior formulations of the CSC method.

2.1.2 The CSC Matrices

First we consider the BVP (2.1) in one-dimensional domain $\Omega \equiv (\omega_1, \omega_2)$ subject to homogeneous Dirichlet boundary conditions,

$$\mathrm{B}u \equiv u = 0 \text{ on } \partial\Omega, \tag{2.5}$$

and its discretization by the CSC method implemented with the basis functions defined by (2.3).

The second, first and zero order terms of L (i.e., u_{xx}, u_x and u, respectively) associated with BCs (2.5) give rise to $(n+3) \times (n+3)$ tridiagonal matrices $\mathbf{C}_2^D$, $\mathbf{C}_1^D$, and $\mathbf{C}_0^D$, respectively, where

$$\mathbf{C}_2^D = \frac{1}{h^2} \begin{pmatrix} \frac{1}{6}h^2 & \frac{2}{3}h^2 & \frac{1}{6}h^2 & & & \\ 1 & -2 & 1 & & & \\ & 1 & -2 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & -2 & 1 \\ & & & \frac{1}{6}h^2 & \frac{2}{3}h^2 & \frac{1}{6}h^2 \end{pmatrix}, \tag{2.6}$$

$$\mathbf{C}_1^D = \frac{1}{2h}\begin{pmatrix} \frac{1}{3}h & \frac{4}{3}h & \frac{1}{3}h & & & \\ -1 & 0 & 1 & & & \\ & -1 & 0 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 0 & 1 \\ & & & \frac{1}{3}h & \frac{4}{3}h & \frac{1}{3}h \end{pmatrix}, \quad \mathbf{C}_0^D = \frac{1}{6}\begin{pmatrix} 1 & 4 & 1 & & & \\ 1 & 4 & 1 & & & \\ & 1 & 4 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & 4 & 1 \\ & & & 1 & 4 & 1 \end{pmatrix}. \tag{2.7}$$

Thus, the collocation equations are written in a linear system form as $\mathbf{A}\mathbf{c}^h = \mathbf{g}$, where

$$\mathbf{A} = diag\{1, a_0, a_1, \ldots, a_n, 1\}\mathbf{C}_2^D + diag\{0, b_0, b_1, \ldots, b_n, 0\}\mathbf{C}_1^D + diag\{0, c_0, c_1, \ldots, c_n, 0\}\mathbf{C}_0^D,$$

$$\mathbf{c}^h = [c_{-1}^h, c_0^h, c_1^h, \ldots, c_n^h, c_{n+1}^h]^T, \qquad \mathbf{g} = [0, g_0, g_1, \ldots, g_n, 0]^T,$$

where the notation $diag\{1, a_0, a_1, \ldots, a_n, 1\}$ stands for the diagonal matrix with diagonal elements 1, $a_0, a_1, \ldots, a_n$ and 1. The notation $diag\{0, b_0, b_1, \ldots, b_n, 0\}$ and $diag\{0, c_0, c_1, \ldots, c_n, 0\}$ are defined similarly.

Notice that, in the way we have presented the matrices $\mathbf{C}_2^D$, $\mathbf{C}_1^D$, $\mathbf{C}_0^D$ and $\mathbf{A}$, the first and last rows of $\mathbf{C}_1^D$ and $\mathbf{C}_0^D$ do not affect the entries of $\mathbf{A}$ since the boundary conditions are incorporated into $\mathbf{A}$ through $\mathbf{C}_2^D$.

Based on $\{\phi_i^h\}_{i=-1}^{i=n+1}$, we construct a set of adjusted basis functions $\{\psi_i^h\}_{i=0}^{i=n}$ by

$$\begin{aligned} \psi_0^h &= \phi_0^h - 4\phi_{-1}^h; \\ \psi_1^h &= \phi_1^h - \phi_{-1}^h; \\ \psi_i^h &= \phi_i^h, \ i = 1, \ldots, n-2; \\ \psi_{n-1}^h &= \phi_{n-1}^h - \phi_{n+1}^h; \\ \psi_n^h &= \phi_n^h - 4\phi_{n+1}^h, \end{aligned} \tag{2.8}$$

satisfying homogeneous Dirichlet BCs by construction, and spanning the space $S_\Delta^D = \{S \in S_\Delta, \text{ and } S(x_0) = S(x_n) = 0\}$.

If the BVP (2.1)–(2.5) is discretized by the CSC method implemented with the adjusted basis functions (2.8), the corresponding discretization matrices $\tilde{\mathbf{C}}_2^D$, $\tilde{\mathbf{C}}_1^D$, and $\tilde{\mathbf{C}}_0^D$ are $(n+1) \times$

$(n+1)$ tridiagonal matrices, where

$$\tilde{\mathbf{C}}_2^D = \frac{1}{h^2}\begin{pmatrix} -6 & & & & & \\ 1 & -2 & 1 & & & \\ & 1 & -2 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & -2 & 1 \\ & & & & & -6 \end{pmatrix},$$

$$\tilde{\mathbf{C}}_1^D = \frac{1}{2h}\begin{pmatrix} 4 & 2 & & & & \\ -1 & 0 & 1 & & & \\ & -1 & 0 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 0 & 1 \\ & & & & -2 & -4 \end{pmatrix}, \quad \tilde{\mathbf{C}}_0^D = \frac{1}{6}\begin{pmatrix} 0 & 0 & 0 & & & \\ 1 & 4 & 1 & & & \\ & 1 & 4 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & 4 & 1 \\ & & & 0 & 0 & 0 \end{pmatrix}.$$

Therefore, CSC with the adjusted basis functions, satisfying the homogeneous Dirichlet BCs (2.5), results in a $(n+1)\times(n+1)$ linear system of the form, $\tilde{\mathbf{A}}\tilde{\mathbf{c}}^h = \tilde{\mathbf{g}}$, where

$$\tilde{\mathbf{A}} = diag\{a_0, a_1, \ldots, a_n\}\tilde{\mathbf{C}}_2^D + diag\{b_0, b_1, \ldots, b_n\}\tilde{\mathbf{C}}_1^D + diag\{c_0, c_1, \ldots, c_n\}\tilde{\mathbf{C}}_0^D,$$

$$\tilde{\mathbf{c}}^h = [c_0^h, c_1^h, \ldots, c_n^h]^T \quad \text{and} \quad \tilde{\mathbf{g}} = [g_0, g_1, \ldots, g_n]^T.$$

The scalars $\{\tilde{c}_i^h\}_{i=0}^n$ are the DOFs corresponding to the adjusted basis functions.

Secondly, we consider the same PDE as (2.1) but with homogeneous Neumann boundary conditions,

$$\mathrm{B}u \equiv u_x = 0 \text{ on } \partial\Omega, \tag{2.9}$$

and its discretization by the CSC method implemented with the basis functions defined by (2.3).

The second, first and zero order terms of L (i.e., u_{xx}, u_x and u, respectively) associated with BCs (2.9) give rise to $(n+3)\times(n+3)$ tridiagonal matrices $\mathbf{C}_2^N$, $\mathbf{C}_1^N$, and $\mathbf{C}_0^N$, respectively, where

$$\mathbf{C}_2^N = \frac{1}{h^2}\begin{pmatrix} -\frac{h}{2} & 0 & \frac{h}{2} & & & \\ 1 & -2 & 1 & & & \\ & 1 & -2 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & -2 & 1 \\ & & & -\frac{h}{2} & 0 & \frac{h}{2} \end{pmatrix},$$

$$\mathbf{C}_1^N = \frac{1}{2h}\begin{pmatrix} -1 & 0 & 1 & & & \\ -1 & 0 & 1 & & & \\ & -1 & 0 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 0 & 1 \\ & & & -1 & 0 & 1 \end{pmatrix}, \quad \mathbf{C}_0^N = \frac{1}{6}\begin{pmatrix} -\frac{3}{h} & 0 & \frac{3}{h} & & & \\ 1 & 4 & 1 & & & \\ & 1 & 4 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & 4 & 1 \\ & & & -\frac{3}{h} & 0 & \frac{3}{h} \end{pmatrix}.$$

Thus, the collocation equations are written in a linear system form as $\mathbf{A}\mathbf{c}^h = \mathbf{g}$, where

$$\mathbf{A} = diag\{1, a_0, a_1, \ldots, a_n, 1\}\mathbf{C}_2^N + diag\{0, b_0, b_1, \ldots, b_n, 0\}\mathbf{C}_1^N + diag\{0, c_0, c_1, \ldots, c_n, 0\}\mathbf{C}_0^N,$$

$$\mathbf{c}^h = [c_{-1}^h, c_0^h, c_1^h, \ldots, c_n^h, c_{n+1}^h]^T \quad \text{and} \quad \mathbf{g} = [0, g_0, g_1, \ldots, g_n, 0]^T.$$

Based on $\{\phi_i^h\}_{i=-1}^{i=n+1}$, we construct a set of adjusted basis functions $\{\psi_i^h\}_{i=0}^{i=n}$ by

$$\begin{aligned} \psi_0^h &= \phi_0^h; \\ \psi_1^h &= \phi_1^h + \phi_{-1}^h; \\ \psi_i^h &= \phi_i^h,\ i = 1, \ldots, n-2; \\ \psi_{n-1}^h &= \phi_{n-1}^h + \phi_{n+1}^h; \\ \psi_n^h &= \phi_n^h, \end{aligned} \tag{2.10}$$

satisfying homogeneous Neumann BCs by construction, and spanning the space $S_\Delta^N = \{S \in S_\Delta,$ and $S'(x_0) = S'(x_n) = 0\}$.

If the BVP (2.1) with (2.9) is discretized by the CSC method implemented with the adjusted basis functions (2.10), the corresponding discretization matrices $\tilde{\mathbf{C}}_2^N$, $\tilde{\mathbf{C}}_1^N$ and $\tilde{\mathbf{C}}_0^N$ are $(n+1) \times$

$(n+1)$ tridiagonal matrices, where

$$\tilde{\mathbf{C}}_2^N = \frac{1}{h^2}\begin{pmatrix} -2 & 2 & & & & \\ 1 & -2 & 1 & & & \\ & 1 & -2 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & -2 & 1 \\ & & & & 2 & -2 \end{pmatrix},$$

$$\tilde{\mathbf{C}}_1^N = \frac{1}{2h}\begin{pmatrix} 0 & 0 & & & & \\ -1 & 0 & 1 & & & \\ & -1 & 0 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & -1 & 0 & 1 \\ & & & & 0 & 0 \end{pmatrix}, \qquad \tilde{\mathbf{C}}_0^N = \frac{1}{6}\begin{pmatrix} 4 & 2 & & & & \\ 1 & 4 & 1 & & & \\ & 1 & 4 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & 4 & 1 \\ & & & & 2 & 4 \end{pmatrix}.$$

Therefore, CSC with the adjusted basis functions, satisfying the homogeneous Neumann BCs (2.9), results in a linear system of the form, $\tilde{\mathbf{A}}\tilde{\mathbf{c}}^h = \tilde{\mathbf{g}}$, where

$$\tilde{\mathbf{A}} = diag\{a_0, a_1, \ldots, a_n\}\tilde{\mathbf{C}}_2^N + diag\{b_0, b_1, \ldots, b_n\}\tilde{\mathbf{C}}_1^N + diag\{c_0, c_1, \ldots, c_n\}\tilde{\mathbf{C}}_0^N,$$

$$\tilde{\mathbf{c}}^h = [c_0^h, c_1^h, \ldots, c_n^h]^T, \quad \text{and} \quad \tilde{\mathbf{g}} = [g_0, g_1, \ldots, g_n]^T.$$

Finally, we consider the same PDE as (2.1) but with the periodic boundary condition,

$$u(\omega_1) = u(\omega_2). \tag{2.11}$$

For consistency with the periodic cubic splines, we also assume periodicity for the first and second derivative,

$$\begin{aligned} u_x(\omega_1) &= u_x(\omega_2); \\ u_{xx}(\omega_1) &= u_{xx}(\omega_2). \end{aligned} \tag{2.12}$$

If we discretize the problem by the CSC method implemented with the basis functions defined by (2.3), we need $n+3$ linearly independent conditions to solve for $n+3$ DOFs.

Since we assume periodicity on the boundary, that is, relations (2.11)–(2.12) hold for the cubic spline approximation as well, the collocation conditions on the collocation points x_0 and x_n are correlated. Therefore, we can only choose one of the two end points as the collocation point. In our case, we choose x_n. Thus the collocation points are $x_1, x_2, \ldots, x_n$. Combining them with the 3 periodic BCs, we get the $(n+3) \times (n+3)$ discretization matrices $\mathbf{C}_2^P$, $\mathbf{C}_1^P$, and $\mathbf{C}_0^P$, where

$$\mathbf{C}_2^P = \frac{1}{h^2}\begin{pmatrix} \frac{1}{6}h^2 & \frac{2}{3}h^2 & \frac{1}{6}h^2 & & & -\frac{1}{6}h^2 & -\frac{2}{3}h^2 & -\frac{1}{6}h^2 \\ -\frac{h}{2} & 0 & \frac{h}{2} & & & \frac{h}{2} & 0 & -\frac{h}{2} \\ 1 & -2 & 1 & & & & & \\ & 1 & -2 & 1 & & & & \\ & & \ddots & \ddots & \ddots & & & \\ & & & \ddots & \ddots & \ddots & & \\ & & & & \ddots & \ddots & \ddots & \\ & & & & & 1 & -2 & 1 \\ 1 & -2 & 1 & & & -1 & 2 & -1 \end{pmatrix},$$

$$\mathbf{C}_1^P = \frac{1}{2h}\begin{pmatrix} \frac{1}{3}h & \frac{4}{3}h & \frac{1}{3}h & & & -\frac{1}{3}h & -\frac{4}{3}h & -\frac{1}{3}h \\ -1 & 0 & 1 & & & 1 & 0 & -1 \\ -1 & 0 & 1 & & & & & \\ & -1 & 0 & 1 & & & & \\ & & \ddots & \ddots & \ddots & & & \\ & & & \ddots & \ddots & \ddots & & \\ & & & & \ddots & \ddots & \ddots & \\ & & & & & -1 & 0 & 1 \\ \frac{2}{h} & -\frac{4}{h} & \frac{2}{h} & & & -\frac{2}{h} & \frac{4}{h} & -\frac{2}{h} \end{pmatrix},$$

$$
\mathbf{C}_0^P = \frac{1}{6}\begin{pmatrix}
1 & 4 & 1 & & & -1 & -4 & -1 \\
-\frac{3}{h} & 0 & \frac{3}{h} & & & \frac{3}{h} & 0 & -\frac{3}{h} \\
1 & 4 & 1 & & & & & \\
 & 1 & 4 & 1 & & & & \\
 & & \ddots & \ddots & \ddots & & & \\
 & & & \ddots & \ddots & \ddots & & \\
 & & & & \ddots & \ddots & \ddots & \\
 & & & & & 1 & 4 & 1 \\
\frac{6}{h^2} & -\frac{12}{h^2} & \frac{6}{h^2} & & & -\frac{6}{h^2} & \frac{12}{h^2} & -\frac{6}{h^2}
\end{pmatrix}.
$$

Thus, the collocation equations are written in a linear system form as $\mathbf{A}\mathbf{c}^h = \mathbf{g}$, where

$$\mathbf{A} = diag\{1, 1, a_1, \ldots, a_n, 1\}\mathbf{C}_2^P + diag\{0, 0, b_1, \ldots, b_n, 0\}\mathbf{C}_1^P + diag\{0, 0, c_1, \ldots, c_n, 0\}\mathbf{C}_0^P,$$

$$\mathbf{c}^h = [c_{-1}^h, c_0^h, c_1^h, \ldots, c_n^h, c_{n+1}^h]^T, \quad \text{and} \quad \mathbf{g} = [0, 0, g_1, \ldots, g_n, 0]^T.$$

Actually, from the periodic BCs (2.11)–(2.12), we can get such relations as $c_{-1} = c_{n-1}, c_0 = c_n,$ and $c_1 = c_{n+1}$. Therefore we can condense our system to solve only for $c_1, \ldots, c_n$, then recover c_{-1}, c_0, and c_1.

Alternatively, we can adjust our basis functions to incorporate the periodic BCs. Based on

$\{\phi_i^h\}_{i=-1}^{i=n+1}$, we construct a set of adjusted basis functions $\{\psi_i^h\}_{i=0}^{i=n}$ by

$$
\begin{aligned}
\psi_1^h(x) &= \begin{cases} \phi_1^h(x), & x \in (x_0, x_3] \\ \phi_{n+1}^h(x), & x \in [x_{n-1}, x_n] \\ 0, & \text{otherwise;} \end{cases} \\
\psi_i^h(x) &= \phi_i^h(x), i = 2, \ldots, n-2; \\
\psi_{n-1}^h(x) &= \begin{cases} \phi_{-1}^h(x), & x \in (x_0, x_1] \\ \phi_{n-1}^h(x), & x \in [x_{n-3}, x_n] \\ 0, & \text{otherwise;} \end{cases} \\
\psi_n^h(x) &= \begin{cases} \phi_0^h(x), & x \in (x_0, x_2] \\ \phi_n^h(x), & x \in [x_{n-2}, x_n] \\ 0, & \text{otherwise,} \end{cases}
\end{aligned}
\tag{2.13}
$$

satisfying periodic BCs by construction and spanning the space $S_\Delta^P = \{S \in S_\Delta, \text{ and } S(x_0) = S(x_n), S'(x_0) = S'(x_n), S''(x_0) = S''(x_n)\}$.

If the BVP (2.1) with periodic BCs (2.11)–(2.12) is discretized by the CSC method implemented with the adjusted basis functions (2.13), the corresponding discretization matrices $\tilde{\mathbf{C}}_2^P$, $\tilde{\mathbf{C}}_1^P$ and $\tilde{\mathbf{C}}_0^P$ are $n \times n$ tridiagonal matrices, where

$$
\tilde{\mathbf{C}}_2^P = \frac{1}{h^2} \begin{pmatrix} -2 & 1 & & & 1 \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ 1 & & & 1 & -2 \end{pmatrix}, \tag{2.14}
$$

$$
\tilde{\mathbf{C}}_1^P = \frac{1}{2h} \begin{pmatrix} 0 & 1 & & & -1 \\ -1 & 0 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 0 & 1 \\ 1 & & & -1 & 0 \end{pmatrix}, \quad \tilde{\mathbf{C}}_0^P = \frac{1}{6} \begin{pmatrix} 4 & 1 & & & 1 \\ 1 & 4 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 4 & 1 \\ 1 & & & 1 & 4 \end{pmatrix}. \tag{2.15}
$$

Therefore, CSC with the adjusted basis functions, satisfying the periodic BCs (2.11)–(2.12), results in a linear system of the form, $\tilde{\mathbf{A}}\tilde{\mathbf{c}}^h = \tilde{\mathbf{g}}$, where

$$\tilde{\mathbf{A}} = diag\{a_1, a_2, \ldots, a_n\}\tilde{\mathbf{C}}_2^P + diag\{b_1, b_2, \ldots, b_n\}\tilde{\mathbf{C}}_1^P + diag\{c_1, c_2, \ldots, c_n\}\tilde{\mathbf{C}}_0^P,$$

$$\tilde{\mathbf{c}}^h = [c_1^h, c_2^h, \ldots, c_n^h]^T, \quad and \quad \tilde{\mathbf{g}} = [g_1, g_2, \ldots, g_n]^T. \tag{2.16}$$

2.1.3 The Standard and Optimal CSC Methods

The standard CSC method applied to (2.1)-(2.2) produces $O(h^2)$ approximations. However, cubic spline interpolation gives $O(h^4)$ approximations. Based on an appropriate perturbation of the differential equation, two types of CSC methods, deferred-correction [7, 1] and extrapolated CSC methods [9], achieve optimal fourth order convergence. The extrapolated or one-step method applies a perturbation to the operator or to the arising matrix. The deferred-correction or two-step method first obtains a second order approximation through standard CSC, then applies a perturbation to the right hand side. In this thesis, we refer to the deferred-correction or two-step method as our optimal CSC method.

2.2 Multigrid Methods

The multigrid method exploits the connection between the physical problem and its matrix analogue in order to accelerate the convergence of an iterative method. The multigrid method is a iterative linear solver based on the multilevel paradigm, and restricting and extrapolating solutions between a series of grids, from the finest to the coarsest. The underlying principle of the multigrid method is that many classical iterative methods, if appropriately applied to solving a discrete elliptic problem, have a strong smoothing effect on the high frequency components of the error, while the low frequency components (the smooth error) are well approximated on a coarse grid [21]. This suggests that a coarse grid should be used to damp the low to middle-frequency components of the error while a fine grid should be used to damp the middle to high-frequency components of the error. The power of the multigrid method lies in the fact that the coarse grid, which acts as a preconditioner, allows the information to pass from a point of the problem domain to another point in a few steps, while the fine grid maintains the accuracy required. Consider solving a collocation linear system $\mathbf{Ac} = \mathbf{g}$ on some fine grid. We can view the multigrid method as a preconditioning technique, consisting of the following steps:

1. apply one (or a few) iterations(s) of some basic iterative method (smoother), e.g., Jacobi's iterative method, to $\mathbf{Ac} = \mathbf{g}$ to obtain an approximate solution vector $\mathbf{w}$. Let $\mathbf{r}$ be the residual vector and $\mathbf{r} = \mathbf{g} - \mathbf{Aw}$.

2. Restrict (project) $\mathbf{r}$ to a coarse grid to obtain the coarse-grid residual vector $\hat{\mathbf{r}}$. Let $\hat{\mathbf{A}}$ be the collocation matrix on the coarse grid.

3. Solve $\hat{\mathbf{A}}\hat{\mathbf{s}} = \hat{\mathbf{r}}$ to get the preconditioned coarse-grid residual vector. This can be done by applying a few iterations of a basic iterative method, or by recursively applying the multigrid method, or by using direct solver since $\hat{\mathbf{A}}$ is a small matrix.

4. Extend $\hat{\mathbf{s}}$ to the find grid to obtain the preconditioned fine-grid residual vector $\mathbf{s}$.

5. Add $\mathbf{s}$ to $\mathbf{w}$ to obtain a new approximate solution vector.

We can repeat the above process until convergence. Depending on different ways of repeating the above process, there are typically three different algorithms for the multigrid method, the two-grid method, the V-cycle method, and the full multigrid method [21].

The multigrid method is characterized by so-called components, which include the smoother, the restriction operator, the extension operator and the coarse grid operator.

Smoother

Classical iterative methods, often called relaxation methods, are commonly used for the purpose of error smoothing in multigrid methods. Examples of smoother are Jacobi's iterative method, Gauss-Seidel method, successive over-relaxation method, and conjugate gradient method.

Restriction operator and Extension operator

A restriction operator maps functions (vectors) from a fine grid to a coarse grid. An extension operator maps functions (vectors) from a coarse grid to a fine grid. Various interpolation methods can be used as restriction and extension operators. We will develop specific restriction and extension operators for CSC systems in the next section.

Coarse grid operator

After restricting the residual to the coarse grid, the coarse grid operator obtains the matrix of the linear system on the coarse grid. There are various approaches, such as direct discretization, Galerkin approach [24], and Petrov-Galerkin approach [3, 25, 8], etc. In this thesis, we use the direct discretization method as the coarse grid operator, since for our problems, the direct discretization on the coarse grid is the simplest and fastest and it naturally reflects the physical connection between coarse and fine grids.

2.3 Multigrid Methods for CSC Equations

In this section, we introduce the multigrid methods for CSC and develop the extension and restriction operators for cubic splines with various boundary conditions.

2.3.1 The Restriction and Extension Operators for Cubic Splines with General BCs

We first develop restriction and extension operators for cubic splines assuming no specific BCs. Let $\{\phi_i^h\}_{i=-1}^{n+1}$ be the cubic spline basis functions constructed with step-size h and $\{\phi_i^{2h}\}_{i=-1}^{n/2+1}$ be the cubic spline basis functions constructed with step-size $2h$.

In the following lemma, we show that the cubic spline basis functions corresponding to the coarse grid can be represented as a particular linear combination of basis functions on the fine grid (i.e. the underlying spaces are nested). From this we can calculate explicit "interpolation-like" formulas to move the coefficients directly from the coarse to fine grid.

LEMMA 2.3.1 *If $s(x) = \Sigma_{i=-1}^{n/2+1} c_i^{2h} \phi_i^{2h}(x)$ is the representation of any cubic spline $s(x)$ with respect to the basis functions $\{\phi_i^{2h}\}_{i=-1}^{n/2+1}$, then $s(x) = \Sigma_{i=-1}^{n+1} c_i^h \phi_i^h(x)$ is the representation of $s(x)$ with respect to the basis functions $\{\phi_i^h\}_{i=-1}^{n+1}$, where the following relations hold:*

$$
\begin{aligned}
c_{2j+1}^h &= \frac{1}{8}(4c_j^{2h} + 4c_{j+1}^{2h}), \quad j = -1, \ldots, n/2; && (2.17)\\
c_{2j}^h &= \frac{1}{8}(c_{j-1}^{2h} + 6c_j^{2h} + c_{j+1}^{2h}), \quad j = 0, \ldots, n/2. && (2.18)
\end{aligned}
$$

PROOF. From the definition of cubic spline basis functions as in (2.3)–(2.4), it is easy to prove that $\{\phi_i^h\}_{i=-1}^{n+1}$ and $\{\phi_i^{2h}\}_{i=-1}^{n/2+1}$ are related by

$$
\begin{aligned}
\phi_{-1}^{2h} &= \frac{1}{8}(4\phi_{-1}^h + \phi_0^h); && (2.19)\\
\phi_0^{2h} &= \frac{1}{8}(4\phi_{-1}^h + 6\phi_0^h + 4\phi_1^h + \phi_2^h); && (2.20)\\
\phi_i^{2h} &= \frac{1}{8}(\phi_{2i-2}^h + 4\phi_{2i-1}^h + 6\phi_{2i}^h + 4\phi_{2i+1}^h + \phi_{2i+2}^h), \quad i = 1, \ldots, n/2-1; && (2.21)\\
\phi_{n/2}^{2h} &= \frac{1}{8}(\phi_{n-2}^h + 4\phi_{n-1}^h + 6\phi_n^h + 4\phi_{n+1}^h); && (2.22)\\
\phi_{n/2+1}^{2h} &= \frac{1}{8}(\phi_n^h + 4\phi_{n+1}^h). && (2.23)
\end{aligned}
$$

Then

$$
\begin{aligned}
s(x) &= \Sigma_{i=-1}^{n/2+1} c_i^{2h}\phi_i^{2h}(x) \\
&= \tfrac{1}{8}c_{-1}^{2h}(4\phi_{-1}^h + \phi_0^h) \\
&\quad + \tfrac{1}{8}c_0^{2h}(4\phi_{-1}^h + 6\phi_0^h + 4\phi_1^h + \phi_2^h) \\
&\quad + \tfrac{1}{8}\Sigma_{i=1}^{n/2-1} c_i^{2h}(\phi_{2i-2}^h + 4\phi_{2i-1}^h + 6\phi_{2i}^h + 4\phi_{2i+1}^h + \phi_{2i+2}^h) \\
&\quad + \tfrac{1}{8}c_{n/2}^{2h}(\phi_{n-2}^h + 4\phi_{n-1}^h + 6\phi_n^h + 4\phi_{n+1}^h) \\
&\quad + \tfrac{1}{8}c_{n/2+1}^{2h}(\phi_n^h + 4\phi_{n+1}^h) \qquad (2.24) \\
&= \tfrac{1}{8}(4c_{-1}^{2h} + 4c_0^{2h})\phi_{-1}^h + \tfrac{1}{8}(c_{-1}^{2h} + 6c_0^{2h} + c_1^{2h})\phi_0^h \\
&\quad + \tfrac{1}{8}(4c_0^{2h} + 4c_1^{2h})\phi_1^h + \tfrac{1}{8}(c_0^{2h} + 6c_1^{2h} + c_2^{2h})\phi_2^h \\
&\quad + \ldots \\
&\quad + \tfrac{1}{8}(4c_{n/2-1}^{2h} + 4c_{n/2}^{2h})\phi_{n-1}^h + \tfrac{1}{8}(c_{n/2-1}^{2h} + 6c_{n/2}^{2h} + c_{n/2+1}^{2h})\phi_n^h \\
&\quad + \tfrac{1}{8}(4c_{n/2}^{2h} + 4c_{n/2+1}^{2h})\phi_{n+1}^h.
\end{aligned}
$$

Relations (2.17)–(2.18) are derived directly from Relation (2.24). $\diamondsuit$

Relations (2.17)–(2.18) lead to the following extension operator matrix $E^G \in \mathbf{R}^{(n+3)\times(\frac{n}{2}+3)}$ for the coefficients of the finite element representation of a cubic spline:

$$
E^G = \frac{1}{8}\begin{pmatrix}
4 & 4 & & & & \\
1 & 6 & 1 & & & \\
 & 4 & 4 & & & \\
 & 1 & 6 & 1 & & \\
 & & \ddots & \ddots & \ddots & \\
 & & & 4 & 4 & \\
 & & & 1 & 6 & 1 \\
 & & & & 4 & 4
\end{pmatrix}. \qquad (2.25)
$$

In order to construct appropriate restriction operators for cubic splines, we consider the structure of the cubic spline basis functions. Strictly speaking, the cubic spline basis functions are *not* nodal, in the sense that the values of the coefficients do not represent function values at the grid points, or other particular points. However, we know that the restriction operator will be applied to the components of the residual vector, which represent residual values on the grid points. The grid points of the coarse grid are a subset of the grid points of the fine grid. Therefore, when transferring from the fine to the coarse grid, we can just pick the residual values corresponding to every other grid point. Thus, we can define a very simple restriction operator matrix $R^G \in \mathbf{R}^{(\frac{n}{2}+3)\times(n+3)}$, where

$$R^G = \begin{pmatrix} 1 & & & & & & & & \\ 0 & 1 & & & & & & & \\ & 0 & 0 & 1 & & & & & \\ & & & 0 & 0 & 1 & & & \\ & & & & & & \ddots & & \\ & & & & & & & 1 & \\ & & & & & & & 0 & 1 \end{pmatrix}. \tag{2.26}$$

Thus, if $\mathbf{c}^h$ is a vector of size $n+3$ corresponding to step-size h, the respective restricted vector $\mathbf{c}^{2h}$ corresponding to step-size $2h$ is defined by

$$\begin{aligned} c_1^{2h} &= c_1^h; \\ c_{i/2}^{2h} &= c_{i-2}^h, \quad i = 4, 6, 8, \ldots, n+4; \\ c_{n/2+3}^{2h} &= c_{n+3}^h. \end{aligned} \tag{2.27}$$

2.3.2 The Restriction and Extension Operators for Cubic Splines with Dirichlet BCs

In this subsection, we develop restriction and extension operators for cubic splines satisfying homogeneous Dirichlet BCs. Let $\{\phi_i^h\}_{i=-1}^{n+1}$ be the cubic spline basis functions constructed with step-size h and $\{\phi_i^{2h}\}_{i=-1}^{n/2+1}$ be the cubic spline basis functions constructed with step-size $2h$. Let $\{\psi_i^h\}_{i=0}^{n}$ be the adjusted basis functions constructed based on $\{\phi_i^h\}_{i=-1}^{n+1}$ in (2.8), and let $\{\psi_i^{2h}\}_{i=0}^{n/2}$ be the adjusted basis functions constructed the same way but based on $\{\phi_i^{2h}\}_{i=-1}^{n/2+1}$. In such way, $\{\psi_i^h\}_{i=0}^{n/2}$ and $\{\psi_i^{2h}\}_{i=0}^{n/2}$ satisfy homogeneous Dirichlet BCs by construction.

In the following lemma we show similar relations as in Lemma (2.3.1) between the cubic spline basis functions adjusted for homogeneous Dirichlet BCs on the coarse grid and the respective basis functions on the fine grid.

LEMMA 2.3.2 *If $s(x) = \Sigma_{i=0}^{n/2} c_i^{2h} \psi_i^{2h}(x)$ is the representation of any cubic spline $s(x) \in S_\Delta^D$ with respect to the basis functions $\{\psi_i^{2h}\}_{i=0}^{n/2}$, then $s(x) = \Sigma_{i=0}^{n} c_i^h \psi_i^h(x)$ is the representation of $s(x)$ with respect to the basis functions $\{\psi_i^h\}_{i=0}^{n}$, where the following relations hold:*

$$c_0^h = \frac{1}{4} c_0^{2h}; \tag{2.28}$$
$$c_{2j+1}^h = \frac{1}{8}(4c_j^{2h} + 4c_{j+1}^{2h}), \quad j = 0, \ldots, n/2 - 1; \tag{2.29}$$
$$c_{2j}^h = \frac{1}{8}(c_{j-1}^{2h} + 6c_j^{2h} + c_{j+1}^{2h}), \quad j = 1, \ldots, n/2 - 1; \tag{2.30}$$
$$c_n^h = \frac{1}{4} c_{n/2}^{2h}. \tag{2.31}$$

PROOF. From (2.19)–(2.23), it is easy to see that

$$\begin{aligned}
\psi_0^{2h} &= \phi_0^{2h} + 4\phi_{-1}^{2h} && (2.32)\\
&= \frac{1}{8}(4\phi_{-1}^h + 6\phi_0^h + 4\phi_1^h + \phi_2^h) - 4\frac{1}{8}(4\phi_{-1}^h + \phi_0^h) && (2.33)\\
&= \frac{4}{8}(-3\phi_{-1}^h) + \frac{1}{8}(2\phi_0^h) + \frac{4}{8}\phi_1^h + \frac{1}{8}\phi_2^h && (2.34)\\
&= \frac{1}{8}2(\phi_0^h - 4\phi_{-1}^h) + \frac{1}{8}4(\phi_1^h - \phi_{-1}^h) + \frac{1}{8}\phi_2^h && (2.35)\\
&= \frac{1}{4}\psi_0^h + \frac{1}{2}\psi_1^h + \frac{1}{8}\psi_2^h && (2.36)\\
&= \frac{1}{8}(2\psi_0^h + 4\psi_1^h + \psi_2^h); && (2.37)
\end{aligned}$$

$$\begin{aligned}
\psi_1^{2h} &= \phi_1^{2h} - \phi_{-1}^{2h} && (2.38)\\
&= \frac{1}{8}(\phi_0^h + 4\phi_1^h + 6\phi_2^h + 4\phi_3^h + \phi_4^h) - \frac{1}{8}(4\phi_{-1}^h + \phi_0^h) && (2.39)\\
&= \frac{4}{8}(\phi_1^h - \phi_{-1}^h) + \frac{1}{8}(6\phi_2^h + 4\phi_3^h + \phi_4^h) && (2.40)\\
&= \frac{4}{8}\psi_1^h + \frac{6}{8}\phi_2^h + \frac{4}{8}\phi_3^h + \frac{1}{8}\phi_4^h) && (2.41)\\
&= \frac{1}{8}(4\psi_1^h + 6\phi_2^h + 4\phi_3^h + \phi_4^h); && (2.42)
\end{aligned}$$

$$\psi_i^{2h} = \frac{1}{8}(\psi_{2i-2}^h + 4\psi_{2i-1}^h + 6\psi_{2i}^h + 4\psi_{2i+1}^h + \psi_{2i+2}^h), \quad i = 2, \ldots, n/2 - 2, \tag{2.43}$$

and similarly,

$$\psi^{2h}_{n/2-1} = \frac{1}{8}(\psi^h_{n-4} + 4\psi^h_{n-3} + 6\psi^h_{n-2} + 4\psi^h_{n-1}); \tag{2.44}$$

$$\psi^{2h}_{n/2} = \frac{1}{8}(\psi^h_{n-2} + 4\psi^h_{n-1} + 2\psi^h_{n}). \tag{2.45}$$

Then

$$\begin{aligned}
s(x) &= \Sigma_{i=0}^{n/2} c_i^{2h}\psi_i^{2h}(x) \\
&= \tfrac{1}{8}c_0^{2h}(2\psi_0^h + 4\psi_1^h + \psi_2^h) \\
&\quad + \tfrac{1}{8}c_1^{2h}(4\psi_1^h + 6\psi_2^h + 4\psi_3^h + \psi_4^h) \\
&\quad + \tfrac{1}{8}\Sigma_{i=2}^{n/2-2} c_i^{2h}(\psi^h_{2i-2} + 4\psi^h_{2i-1} + 6\psi^h_{2i} + 4\psi^h_{2i+1} + \psi^h_{2i+2}) \\
&\quad + \tfrac{1}{8}c^{2h}_{n/2-1}(\psi^h_{n-4} + 4\psi^h_{n-3} + 6\psi^h_{n-2} + 4\psi^h_{n-1}) \\
&\quad + \tfrac{1}{8}c^{2h}_{n/2}(\psi^h_{n-2} + 4\psi^h_{n-1} + 2\psi^h_{n}) \\
&= \tfrac{1}{8}(2c_0^{2h})\psi_0^h \\
&\quad + \tfrac{1}{8}(4c_0^{2h} + 4c_1^{2h})\psi_1^h + \tfrac{1}{8}(c_0^{2h} + 6c_1^{2h} + c_2^{2h})\psi_2^h \\
&\quad + \dots \\
&\quad + \tfrac{1}{8}(4c^{2h}_{n/2-2} + 4c^{2h}_{n/2-1})\psi^h_{n-3} + \tfrac{1}{8}(c^{2h}_{n/2-2} + 6c^{2h}_{n/2-1} + c^{2h}_{n/2})\psi^h_{n-2} \\
&\quad + \tfrac{1}{8}(4c^{2h}_{n/2-1} + 4c^{2h}_{n/2})\psi^h_{n-1} \\
&\quad + \tfrac{1}{8}(2c^{2h}_{n/2})\psi^h_{n}.
\end{aligned} \tag{2.46}$$

Relations (2.28)–(2.31) are derived directly from relation (2.46). ♢

Relations (2.28)–(2.31) lead to the following extension operator matrix $E^D \in \mathbf{R}^{(n+1)\times(\frac{n}{2}+1)}$ for the coefficients of the finite element representation of a cubic spline that satisfies homogeneous Dirichlet BCs:

$$E^D = \frac{1}{8}\begin{pmatrix} 2 & & & & & \\ 4 & 4 & & & & \\ 1 & 6 & 1 & & & \\ & \ddots & \ddots & \ddots & & \\ & & 4 & 4 & & \\ & & 1 & 6 & 1 & \\ & & & 4 & 4 & \\ & & & & 2 & \end{pmatrix}. \tag{2.47}$$

As for the restriction operator, we still define a simple restriction operator matrix $R^D \in \mathbf{R}^{(\frac{n}{2}+1)\times(n+1)}$, by

$$R^D = \begin{pmatrix} 1 & & & & & & & & \\ 0 & 0 & 1 & & & & & & \\ & & 0 & 0 & 1 & & & & \\ & & & & & \ddots & & & \\ & & & & & & 1 & & \\ & & & & & & & 0 & 1 \end{pmatrix}. \tag{2.48}$$

Thus, if $\mathbf{c}^h$ is a vector of size $n+1$ corresponding to step-size h, the respective restricted vector $\mathbf{c}^{2h}$ corresponding to step-size $2h$ is defined by

$$c_{i/2}^{2h} = c_{i-1}^{h}, \quad i = 2, 4, \ldots, n+2. \tag{2.49}$$

2.3.3 The Restriction and Extension Operators for Cubic Splines with Neumann BCs

In this subsection, we develop restriction and extension operators for cubic splines satisfying homogeneous Neumann BCs. Let $\{\phi_i^h\}_{i=-1}^{n+1}$ be the cubic spline basis functions constructed with step-size h and $\{\phi_i^{2h}\}_{i=-1}^{n/2+1}$ be the cubic spline basis functions constructed with step-size $2h$. Let $\{\psi_i^h\}_{i=0}^{n}$ be the adjusted basis functions constructed based on $\{\phi_i^h\}_{i=-1}^{n+1}$ as in (2.10), and let $\{\psi_i^{2h}\}_{i=0}^{n/2}$ be the adjusted basis functions constructed the same way but based on $\{\phi_i^{2h}\}_{i=-1}^{n/2+1}$. In such way, $\{\psi_i^{h}\}_{i=0}^{n/2}$ and $\{\psi_i^{2h}\}_{i=0}^{n/2}$ satisfy homogeneous Neumann BCs by construction.

In the following lemma we show similar relations as in Lemma (2.3.1) between the cubic spline basis functions adjusted for homogeneous Neumann BCs on the coarse grid and the respective basis functions on the fine grid.

LEMMA 2.3.3 *If $s(x) = \Sigma_{i=0}^{n/2} c_i^{2h}\psi_i^{2h}(x)$ is the representation of any cubic spline $s(x) \in S_\Delta^N$ with respect to the basis functions $\{\psi_i^{2h}\}_{i=0}^{n/2}$, then $s(x) = \Sigma_{i=0}^{n} c_i^{h}\psi_i^{h}(x)$ is the representation of $s(x)$ with respect to the basis functions $\{\psi_i^{h}\}_{i=0}^{n}$, where the following relations hold:*

$$
\begin{aligned}
c_0^h &= \frac{1}{8}(6c_0^{2h} + 2c_1^{2h}); && (2.50)\\
c_{2j+1}^h &= \frac{1}{8}(4c_j^{2h} + 4c_{j+1}^{2h}), \quad j = 0, \ldots, n/2-1; && (2.51)\\
c_{2j}^h &= \frac{1}{8}(c_{j-1}^{2h} + 6c_j^{2h} + c_{j+1}^{2h}), \quad j = 1, \ldots, n/2-1; && (2.52)\\
c_n^h &= \frac{1}{8}(2c_{n/2-1}^{2h} + 6c_{n/2}^{2h}). && (2.53)
\end{aligned}
$$

PROOF. From (2.19)–(2.23), it is easy to see that

$$
\begin{aligned}
\psi_0^{2h} &= \phi_0^{2h} && (2.54)\\
&= \frac{1}{8}(4\phi_{-1}^h + 6\phi_0^h + 4\phi_1^h + \phi_2^h) && (2.55)\\
&= \frac{4}{8}(\phi_{-1}^h + \phi_1^h) + \frac{6}{8}\phi_0^h + \frac{1}{8}\phi_2^h && (2.56)\\
&= \frac{4}{8}\psi_1^h + \frac{6}{8}\psi_0^h + \frac{1}{8}\psi_2^h && (2.57)\\
&= \frac{1}{8}(6\psi_0^h + 4\psi_1^h + \psi_2^h); && (2.58)
\end{aligned}
$$

$$
\begin{aligned}
\psi_1^{2h} &= \phi_1^{2h} + \phi_{-1}^{2h} && (2.59)\\
&= \frac{1}{8}(\phi_0^h + 4\phi_1^h + 6\phi_2^h + 4\phi_3^h + \phi_4^h) + \frac{1}{8}(4\phi_{-1}^h + \phi_0^h) && (2.60)\\
&= \frac{4}{8}(\phi_1^h + \phi_{-1}^h) + \frac{1}{8}(2\phi_0^h + 6\phi_2^h + 4\phi_3^h + \phi_4^h) && (2.61)\\
&= \frac{1}{8}(2\psi_0^h + 4\psi_1^h + 6\psi_2^h + 4\psi_3^h + \psi_4^h); && (2.62)
\end{aligned}
$$

$$
\psi_i^{2h} = \frac{1}{8}(\psi_{2i-2}^h + 4\psi_{2i-1}^h + 6\psi_{2i}^h + 4\psi_{2i+1}^h + \psi_{2i+2}^h), \quad i = 2, \ldots, n/2-2, \quad (2.63)
$$

and similarly,

$$\psi^{2h}_{n/2-1} = \frac{1}{8}(\psi^h_{n-4} + 4\psi^h_{n-3} + 6\psi^h_{n-2} + 4\psi^h_{n-1} + 2\psi^h_n); \tag{2.64}$$

$$\psi^{2h}_{n/2} = \frac{1}{8}(\psi^h_{n-2} + 4\psi^h_{n-1} + 6\psi^h_n). \tag{2.65}$$

Then

$$\begin{aligned}
s(x) &= \Sigma_{i=0}^{n/2} c_i^{2h}\psi_i^{2h}(x) \\
&= \tfrac{1}{8}c_0^{2h}(6\psi_0^h + 4\psi_1^h + \psi_2^h) \\
&\quad + \tfrac{1}{8}c_1^{2h}(2\psi_0^h + 4\psi_1^h + 6\psi_2^h + 4\psi_3^h + \psi_4^h) \\
&\quad + \tfrac{1}{8}\Sigma_{i=2}^{n/2-2} c_i^{2h}(\psi^h_{2i-2} + 4\psi^h_{2i-1} + 6\psi^h_{2i} + 4\psi^h_{2i+1} + \psi^h_{2i+2}) \\
&\quad + \tfrac{1}{8}c^{2h}_{n/2-1}(\psi^h_{n-4} + 4\psi^h_{n-3} + 6\psi^h_{n-2} + 4\psi^h_{n-1} + 2\psi^h_n) \\
&\quad + \tfrac{1}{8}c^{2h}_{n/2}(\psi^h_{n-2} + 4\psi^h_{n-1} + 6\psi^h_n) \\
&= \tfrac{1}{8}(6c_0^{2h} + 2c_1^{2h})\psi_0^h \\
&\quad + \tfrac{1}{8}(4c_0^{2h} + 4c_1^{2h})\psi_1^h + \tfrac{1}{8}(c_0^{2h} + 6c_1^{2h} + c_2^{2h})\psi_2^h \\
&\quad + \dots \\
&\quad + \tfrac{1}{8}(4c^{2h}_{n/2-2} + 4c^{2h}_{n/2-1})\psi^h_{n-3} + \tfrac{1}{8}(c^{2h}_{n/2-2} + 6c^{2h}_{n/2-1} + c^{2h}_{n/2})\psi^h_{n-2} \\
&\quad + \tfrac{1}{8}(4c^{2h}_{n/2-1} + 4c^{2h}_{n/2})\psi^h_{n-1} \\
&\quad + \tfrac{1}{8}(2c^{2h}_{n/2-1} + 6c^{2h}_{n/2})\psi^h_n.
\end{aligned} \tag{2.66}$$

Relations (2.50)–(2.53) are derived directly from relation (2.66). ♢

Relations (2.50)–(2.53) lead to the following extension operator matrix $E^N \in \mathbf{R}^{(n+1)\times(\frac{n}{2}+1)}$ for the coefficients of the finite element representation of a cubic spline that satisfies homogeneous Neumann BCs:

$$E^N = \frac{1}{8}\begin{pmatrix} 6 & 2 & & & \\ 4 & 4 & & & \\ 1 & 6 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 4 & 4 & \\ & & 1 & 6 & 1 \\ & & & 4 & 4 \\ & & & 2 & 6 \end{pmatrix}. \tag{2.67}$$

For Neumann conditions, we define the same restriction operator matrix $R^N \in \mathbf{R}^{(\frac{n}{2}+1)\times(n+1)}$, as for Dirichlet conditions, i.e.,

$$R^N = \begin{pmatrix} 1 & & & & & & & & & \\ 0 & 0 & 1 & & & & & & & \\ & & 0 & 0 & 1 & & & & & \\ & & & & & \ddots & & & & \\ & & & & & & & 1 & & \\ & & & & & & & & 0 & 1 \end{pmatrix}. \tag{2.68}$$

2.3.4 The Restriction and Extension Operators for Cubic Splines with Periodic BCs

Finally, we develop restriction and extension operators for periodic cubic splines. Let $\{\phi_i^h\}_{i=-1}^{n+1}$ be the cubic spline basis functions constructed with step-size h and $\{\phi_i^{2h}\}_{i=-1}^{n/2+1}$ be the cubic spline basis functions constructed with step-size $2h$. Let $\{\psi_i^h\}_{i=1}^{n}$ be the adjusted basis functions constructed based on $\{\phi_i^h\}_{i=-1}^{n+1}$ as in (2.13), and let $\{\psi_i^{2h}\}_{i=1}^{n/2}$ be the adjusted basis functions constructed the same way but based on $\{\phi_i^{2h}\}_{i=-1}^{n/2+1}$. In such way, $\{\psi_i^h\}_{i=1}^{n/2}$ and $\{\psi_i^{2h}\}_{i=1}^{n/2}$ satisfy periodic BCs by construction.

In the following lemma we show similar relations as in Lemma (2.3.1) between the cubic spline basis functions adjusted for periodic BCs on the coarse grid and the respective basis functions on the fine grid.

LEMMA 2.3.4 *If* $s(x) = \Sigma_{i=1}^{n/2} c_i^{2h} \psi_i^{2h}(x)$ *is the representation of any cubic spline* $s(x) \in S_\Delta^P$ *with respect to the basis functions* $\{\psi_i^{2h}\}_{i=1}^{n/2}$*, then* $s(x) = \Sigma_{i=1}^{n} c_i^h \psi_i^h(x)$ *is the representation of* $s(x)$ *with respect to the basis functions* $\{\psi_i^h\}_{i=1}^{n}$*, where the following relations hold:*

$$c_1^h = \frac{1}{8}(4c_{n/2}^{2h} + 4c_1^{2h}); \tag{2.69}$$

$$c_2^h = \frac{1}{8}(c_{n/2}^{2h} + 6c_1^{2h} + c_2^{2h}); \tag{2.70}$$

$$c_{2j+1}^h = \frac{1}{8}(4c_j^{2h} + 4c_{j+1}^{2h}), \quad j = 1, \ldots, n/2-2; \tag{2.71}$$

$$c_{2j}^h = \frac{1}{8}(c_{j-1}^{2h} + 6c_j^{2h} + c_{j+1}^{2h}), \quad j = 2, \ldots, n/2-1; \tag{2.72}$$

$$c_{n/2-1}^h = \frac{1}{8}(4c_{n/2-1}^{2h} + 4c_{n/2}^{2h}); \tag{2.73}$$

$$c_{n/2}^h = \frac{1}{8}(c_{n/2-1}^{2h} + 6c_{n/2}^{2h} + c_1^{2h}); \tag{2.74}$$

PROOF. From (2.19)–(2.23), we can get

$$\psi_1^{2h}(x) = \begin{cases} \phi_1^{2h}(x), & x \in (x_0, x_3] \\ \phi_{n/2+1}^{2h}(x), & x \in [x_{n-1}, x_n] \\ 0, & \text{otherwise} \end{cases} \tag{2.75}$$

$$= \begin{cases} \frac{1}{8}(\phi_0^h + 4\phi_1^h + 6\phi_2^h + 4\phi_3^h + \phi_4^h), & x \in (x_0, x_3] \\ \frac{1}{8}(\phi_n^h + 4\phi_{n+1}^h), & x \in [x_{n-1}, x_n] \\ 0, & \text{otherwise} \end{cases} \tag{2.76}$$

$$= \frac{1}{8}(\psi_n^h + 4\psi_1^h + 6\psi_2^h + 4\psi_3^h + \psi_4^h); \tag{2.77}$$

$$\psi_i^{2h} = \frac{1}{8}(\psi_{2i-2}^h + 4\psi_{2i-1}^h + 6\psi_{2i}^h + 4\psi_{2i+1}^h + \psi_{2i+2}^h), \quad i = 2, \ldots, n/2-2, \tag{2.78}$$

and similarly,

$$\psi_{n/2-1}^{2h} = \frac{1}{8}(\psi_{n-4}^h + 4\psi_{n-3}^h + 6\psi_{n-2}^h + 4\psi_{n-1}^h + \psi_n^h); \tag{2.79}$$

$$\psi_{n/2}^{2h} = \frac{1}{8}(\psi_{n-2}^h + 4\psi_{n-1}^h + 6\psi_n^h + 4\psi_1^h + \psi_2^h). \tag{2.80}$$

Then

$$
\begin{aligned}
s(x) &= \Sigma_{i=1}^{n/2} c_i^{2h} \psi_i^{2h}(x) \\
&= \tfrac{1}{8} c_1^{2h} (\psi_n^h + 4\psi_1^h + 6\psi_2^h + 4\psi_3^h + \psi_4^h) \\
&\quad + \tfrac{1}{8} \Sigma_{i=2}^{n/2-2} c_i^{2h} (\psi_{2i-2}^h + 4\psi_{2i-1}^h + 6\psi_{2i}^h + 4\psi_{2i+1}^h + \psi_{2i+2}^h) \\
&\quad + \tfrac{1}{8} c_{n/2-1}^{2h} (\psi_{n-4}^h + 4\psi_{n-3}^h + 6\psi_{n-2}^h + 4\psi_{n-1}^h + \psi_n^h) \\
&\quad + \tfrac{1}{8} c_{n/2}^{2h} (\psi_{n-2}^h + 4\psi_{n-1}^h + 6\psi_n^h + 4\psi_1^h + \psi_2^h) \\
&= \tfrac{1}{8}(4c_{n/2}^{2h} + 4c_1^{2h})\psi_1^h + \tfrac{1}{8}(c_{n/2}^{2h} + 6c_1^{2h} + c_2^{2h})\psi_2^h \\
&\quad + \tfrac{1}{8}(4c_1^{2h} + 4c_2^{2h})\psi_3^h + \tfrac{1}{8}(c_1^{2h} + 6c_2^{2h} + c_3^{2h})\psi_4^h \\
&\quad + \dots \\
&\quad + \tfrac{1}{8}(4c_{n/2-2}^{2h} + 4c_{n/2-1}^{2h})\psi_{n-3}^h + \tfrac{1}{8}(c_{n/2-2}^{2h} + 6c_{n/2-1}^{2h} + c_{n/2}^{2h})\psi_{n-2}^h \\
&\quad + \tfrac{1}{8}(4c_{n/2-1}^{2h} + 4c_{n/2}^{2h})\psi_{n-1}^h + \tfrac{1}{8}(c_{n/2-1}^{2h} + 6c_{n/2}^{2h} + c_1^{2h})\psi_n^h.
\end{aligned}
\tag{2.81}
$$

Relations (2.70)–(2.74) are derived directly from relation (2.81). ◇

Relations (2.70)–(2.74) lead to the following extension operator matrix $E^P \in \mathbf{R}^{n\times(\frac{n}{2})}$ for the coefficients of the finite element representation of a periodic cubic spline:

$$
E^P = \frac{1}{8}\begin{pmatrix}
4 & & & & 4 \\
6 & 1 & & & 1 \\
4 & 4 & & & \\
1 & 6 & 1 & & \\
 & \ddots & \ddots & \ddots & \\
 & & 4 & 4 & \\
 & & 1 & 6 & 1 \\
 & & & 4 & 4 \\
1 & & & 1 & 6
\end{pmatrix}. \tag{2.82}
$$

We define a simple restriction operator matrix $R^P \in \mathbf{R}^{(\frac{n}{2})\times n}$, by

$$R^P = \begin{pmatrix} 0 & 1 & & & & & & & & \\ 0 & 0 & 0 & 1 & & & & & & \\ & & 0 & 0 & 0 & 1 & & & & \\ & & & & & & \ddots & & & \\ & & & & & & & 1 & & \\ & & & & & & & 0 & 0 & 1 \end{pmatrix}. \tag{2.83}$$

Thus, if $\mathbf{c}^h$ is a vector of size n corresponding to step-size h, the respective restricted vector $\mathbf{c}^{2h}$ corresponding to step-size $2h$ is defined by

$$c_{i/2}^{2h} = c_i^h, \quad i = 2, 4, \ldots, n. \tag{2.84}$$

Chapter 3

Multigrid methods for CSC on Second Order Equations

3.1 The Second Order Equation with Periodic BCs

In this chapter, we consider the second order BVP

$$\mathrm{L}u \equiv u_{xx} - cu = g \quad \text{in } \Omega \equiv [\omega_1, \omega_2], \tag{3.1}$$

subject to periodic boundary conditions

$$u(\omega_1) = u(\omega_2), \quad u_x(\omega_1) = u_x(\omega_2), \quad u_{xx}(\omega_1) = u_{xx}(\omega_2), \tag{3.2}$$

where c is a constant.

For the above BVP, the CSC method implemented with the adjusted basis functions defined by (2.13) results in an almost tridiagonal linear system of the form

$$A\mathbf{c}^h = \mathbf{g}, \tag{3.3}$$

where

$$A = \tilde{\mathbf{C}}_2^P - c\tilde{\mathbf{C}}_0^P, \tag{3.4}$$

$\tilde{\mathbf{C}}_2^P$ and $\tilde{\mathbf{C}}_0^P$ are defined as in (2.14)–(2.15),

$$\mathbf{c}^h = [c_1^h, c_2^h, \ldots, c_n^h]^T \quad \text{and} \quad \mathbf{g} = [g_1, g_2, \ldots, g_n]^T.$$

Since $\tilde{\mathbf{C}}_2^P$ is a circulant matrix, its eigenvalues are given by

$$\lambda_j^{\{2\}} = \frac{1}{h^2}(-2 + 2\cos\frac{2(j-1)\pi}{n}), \quad j = 1, \ldots, n. \tag{3.5}$$

Similarly, the eigenvalues of $\tilde{\mathbf{C}}_0^P$ are given by

$$\lambda_j^{\{0\}} = \frac{1}{6}(4 + 2\cos\frac{2(j-1)\pi}{n}), \quad j = 1, \ldots, n. \tag{3.6}$$

Moreover, matrices A, $\tilde{\mathbf{C}}_2^P$ and $\tilde{\mathbf{C}}_0^P$ have the same orthonormal eigenvectors $\delta_j, j = 1, \ldots, n$, which are defined as the columns of the $n \times n$ matrix

$$U = \frac{1}{\sqrt{n}} \begin{pmatrix} 1 & 1 & 1 & \ldots & 1 \\ 1 & \omega & \omega^2 & \ldots & \omega^{n-1} \\ 1 & \omega^2 & \omega^4 & \ldots & \omega^{2(n-1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \omega^{n-1} & \omega^{2(n-1)} & \ldots & \omega^{(n-1)(n-1)} \end{pmatrix}, \tag{3.7}$$

where

$$\omega = e^{2\pi\imath/n} = \cos\frac{2\pi}{n} + \imath\sin\frac{2\pi}{n}, \tag{3.8}$$

with $\imath = \sqrt{-1}$. Therefore, the eigenvalues of A are

$$\lambda_j = \lambda_j^{\{2\}} - c\lambda_j^{\{0\}} = -(\frac{2}{h^2} + \frac{2c}{3}) + (\frac{2}{h^2} - \frac{c}{3})\cos\frac{2(j-1)\pi}{n}, \quad j = 1, \ldots, n. \tag{3.9}$$

3.2 Convergence Analysis of Jacobi's Iterative Method

In the multigrid methods for CSC equations, we adopt the Jacobi's iterative method as smoother. Therefore, we study its convergence properties when applied to CSC equations. Consider using Jacobi's iterative method to solve the linear system (3.3)–(3.4) arising from CSC discretization of (3.1)–(3.2). Let $D = diag[(-\frac{2}{h^2} - \frac{2c}{3})d]$, where d is a damping factor. Jacobi's iterative method gives rise to the relation

$$D\mathbf{c}^{h[k+1]} = (D-A)\mathbf{c}^{h[k]} + \mathbf{g} \tag{3.10}$$

at each iteration k of the method. Let $\zeta = (-\frac{2}{h^2} - \frac{2c}{3})d$ and $\eta = \frac{1}{h^2} - \frac{c}{6}$.

The iteration matrix G of Jacobi's iterative method is given by

$$\begin{aligned}
G &= I - D^{-1}A \\
&= I - \begin{pmatrix} \frac{1}{d} & \frac{\eta}{\zeta} & & & \frac{\eta}{\zeta} \\ \frac{\eta}{\zeta} & \frac{1}{d} & \frac{\eta}{\zeta} & & \\ & \ddots & \ddots & \ddots & \\ & & \frac{\eta}{\zeta} & \frac{1}{d} & \frac{\eta}{\zeta} \\ \frac{\eta}{\zeta} & & & \frac{\eta}{\zeta} & \frac{1}{d} \end{pmatrix} \\
&= -\frac{\eta}{\zeta}\begin{pmatrix} (\frac{1}{d}-1)\frac{\zeta}{\eta} & 1 & & & 1 \\ 1 & (\frac{1}{d}-1)\frac{\zeta}{\eta} & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & (\frac{1}{d}-1)\frac{\zeta}{\eta} & 1 \\ 1 & & & 1 & (\frac{1}{d}-1)\frac{\zeta}{\eta} \end{pmatrix}.
\end{aligned}$$

Notice that G is circulant. Its eigenvalues are given by

$$\begin{aligned}
\lambda_j^G &= -\frac{\eta}{\zeta}((\frac{1}{d}-1)\frac{\zeta}{\eta} + 2c_j) \\
&= 1 - \frac{1}{d} - 2\frac{\eta}{\zeta}c_j \\
&= 1 - \frac{1}{d} + \frac{ch^2-6}{-6-2ch^2}\frac{1}{d}c_j,
\end{aligned}$$

where $c_j \equiv \cos\frac{2(j-1)\pi}{n}$, $j = 1, \dots, n$.

We want to have $|\lambda_j^G| < \alpha$, for some $\alpha < 1$, which implies that the Jacobi's iterative method converges independently of the initial guess vector.

For damping factor $d = 1$,

$$\begin{aligned}
|\lambda_j^G| &< \alpha \\
\Leftrightarrow\ |\frac{ch^2-6}{6+2ch^2}c_j| &< \alpha
\end{aligned}$$

A sufficient condition to the above is

$$\left|\tfrac{ch^2-6}{6+2ch^2}\right| \quad < \quad \alpha.$$

For sufficiently small h, we have $0 < ch^2 < 6$, so a sufficient condition to the above is

$$\begin{array}{rrcl} & \tfrac{6-ch^2}{6+2ch^2} & < & \alpha \\ \Leftrightarrow & 6-ch^2 & < & 6\alpha + 2ch^2\alpha \\ \Leftrightarrow & (2\alpha+1)ch^2 & > & 6-6\alpha \\ \Leftrightarrow & ch^2 & > & \tfrac{6-6\alpha}{2\alpha+1}. \end{array}$$

Therefore, for $\alpha = 1/2$ and sufficient small h, a sufficient condition for convergence of Jacobi's iterative method with damping factor $d = 1$ is $ch^2 > 3/2$ or $h^2 > \frac{3}{2c}$.

3.3 Convergence Analysis of the Two-grid Method

In this section, we study the behavior of the error at any iteration of the two-grid method for the CSC equations.

The effect of the coarse grid correction scheme on the error is expressed as the result of the application of the operator

$$M_c = \mathbf{I} - E \cdot A'^{-1} \cdot R \cdot A \tag{3.11}$$

to the error vector, where $\mathbf{I}$ is the identity operator of appropriate dimension; A and A' are the CSC matrices for the step-sizes h and $2h$, respectively, as defined in (3.4); E is the extension operator as defined in (2.82); R is the restriction operator as defined in (2.83).

As a simple relaxation scheme, we choose Richardson's iteration, damped by the factor $d \cdot \mathrm{diag}(A)$, i.e., $-d(\frac{2}{h^2} + \frac{2c}{3})$, where d is a constant and the notation $\mathrm{diag}(B)$, for a matrix with constant diagonal, indicates the diagonal term of matrix B. Thus, the effect of one iteration of the relaxation scheme on the error can be expressed as the result of the application of the operator

$$M_r = \mathbf{I} + \frac{1}{d(\frac{2}{h^2} + \frac{2c}{3})} \cdot A. \tag{3.12}$$

Therefore, the effect of one iteration of the two-grid method on the error can be expressed as the result of the application of the operator

$$M = M_c \cdot M_r. \tag{3.13}$$

THEOREM 3.3.1 *Let an iteration of the two-grid method for CSC equations (3.3) consist of a Richardson iteration damped by $-d(\frac{2}{h^2} + \frac{2c}{3})$ and a coarse grid correction scheme characterized by the extension operator (2.82) and the restriction operator (2.83), and applied to the residual of the damped Richardson iteration approximation. Then, for $d = 2$ and sufficiently small step-size h, the two-grid method converges with a contraction factor less than or equal to 1/2 in the Euclidean norm, independently of the step-size.*

PROOF. The proof follows the lines of [4]. Let $\delta_j, j = 1, \ldots, n$, and $\delta'_j, j = 1, \ldots, n/2$, be the eigenvalues of A and A', respectively. Let Q be the matrix with its column formed by the eigenvectors of A in the following order:

$$Q \equiv [\delta_1, \delta_{n/2+1}, \delta_2, \delta_{n/2+2}, \ldots, \delta_{n/2-1}, \delta_{n-1}, \delta_{n/2}, \delta_n]. \tag{3.14}$$

Let Q' be the matrix with its columns formed by the eigenvectors of A' in the natural order:

$$Q' \equiv [\delta'_1, \delta'_2, \ldots, \delta'_{n/2-1}, \delta'_{n/2}]. \tag{3.15}$$

Let $s_j \equiv \sin \frac{2(j-1)\pi}{n}$ and $c_j \equiv \cos \frac{2(j-1)\pi}{n}$, $j = 1, \ldots, n$.
Consider the transformed matrices

$$\tilde{M}_r \equiv Q^T \cdot M_r \cdot Q, \qquad \tilde{M}_c \equiv Q^T \cdot M_c \cdot Q$$

and

$$\tilde{M} \equiv Q^T \cdot M \cdot Q = \tilde{M}_c \cdot \tilde{M}_r = (\mathbf{I} - \tilde{E} \cdot \tilde{A}'^{-1} \cdot \tilde{R} \cdot \tilde{A}) \cdot (\mathbf{I} + \frac{1}{d(\frac{2}{h^2} + \frac{2c}{3})} \cdot \tilde{A}) \tag{3.16}$$

with

$$\tilde{E} \equiv Q^T \cdot E \cdot Q', \qquad \tilde{A} \equiv Q'^T \cdot A'^{-1} \cdot Q', \qquad \tilde{R} \equiv Q'^T \cdot R \cdot Q$$

and

$$\tilde{A} \equiv Q^T \cdot A \cdot Q.$$

Now we show that the transformed matrices have block-diagonal structures.

First consider $\tilde{A}$,

$$\begin{aligned}
\tilde{A} &= Q^T \cdot A \cdot Q \\
&= \begin{pmatrix} \bar{\delta}_1 \cdot \delta_1 \lambda_1 & & & & \\ & \bar{\delta}_{n/2+1} \cdot \delta_{n/2+1} \lambda_{n/2+1} & & & \\ & & \ddots & & \\ & & & \bar{\delta}_{n/2} \cdot \delta_{n/2} \lambda_{n/2} & \\ & & & & \bar{\delta}_n \cdot \delta_n \lambda_n \end{pmatrix} \\
&= \begin{pmatrix} \lambda_1 & & & & \\ & \lambda_{n/2+1} & & & \\ & & \ddots & & \\ & & & \lambda_{n/2} & \\ & & & & \lambda_n \end{pmatrix} \\
&= \text{blockdiag}\{A_1, A_{n/2+1}, A_2, A_{n/2+2}, \ldots, A_{n-1}, A_n\},
\end{aligned}$$

with

$$A_j = [\lambda_j] = [-(\frac{2}{h^2} + \frac{2c}{3}) + (\frac{2}{h^2} - \frac{c}{3})c_j], \quad j = 1, \ldots, n,$$

where the notation blockdiag$\{B_1, \ldots, B_n\}$ denotes a block-diagonal matrix with blocks $B_1, \ldots, B_n$ on the diagonal. Also,

$$\tilde{M}_r = \text{blockdiag}\{\tilde{M}_{r1}, \tilde{M}_{r(n/2+1)}, \tilde{M}_{r2}, \tilde{M}_{r(n/2+2)}, \ldots, \tilde{M}_{r(n/2)}, \tilde{M}_{rn}\}$$

with

$$\tilde{M}_{rj} = [1 + \frac{1}{d(\frac{2}{h^2} + \frac{2c}{3})}\lambda_j] = [1 - \frac{1}{d} + \frac{6 - ch^2}{d(6 + 2ch^2)}c_j], \quad j = 1, \ldots, n.$$

Noting that $c_j = -c_{n/2+j}, \quad j = 1, \ldots, n/2$, we have

$$\tilde{M}_r = \text{blockdiag}\{M_{r1}, M_{r2}, \ldots, M_{r(n/2)}\}$$

with

$$M_{rj} = \begin{bmatrix} m_j & 0 \\ 0 & e - m_j \end{bmatrix}, \quad j = 1, \ldots, n/2. \tag{3.17}$$

where

$$e = 2 - \frac{2}{d},$$

and

$$m_j = 1 + \frac{1}{d(\frac{2}{h^2} + \frac{2c}{3})}\lambda_j = 1 - \frac{1}{d} + \frac{6 - ch^2}{d(6 + 2ch^2)}c_j, \quad j = 1, \ldots, n/2.$$

In addition,

$$\tilde{A}'^{-1} = \text{blockdiag}\{A'_1, A'_2, \ldots, A'_{n/2}\}$$

with

$$A'_j = [\frac{1}{-(\frac{1}{2h^2} + \frac{2c}{3}) + (\frac{1}{2h^2} - \frac{c}{3})(c_j^2 - s_j^2)}], \quad j = 1, \ldots, n/2,$$

and

$$\tilde{R} = Q'^T \cdot R \cdot Q = \frac{\sqrt{2}}{2}\text{blockdiag}\{R_1, R_2, \ldots, R_{n/2}\}$$

with

$$R_j = [c_j + is_j \quad - c_j - is_j], \quad j = 1, \ldots, n/2.$$

Noting that

$$\begin{gathered}\sin\frac{(2i-3)(j-1)2\pi}{n} + 4\sin\frac{(2i-2)(j-1)2\pi}{n} + 6\sin\frac{(2i-1)(j-1)2\pi}{n} \\ +4\sin\frac{2i(j-1)2\pi}{n} + \sin\frac{(2i+1)(j-1)2\pi}{n} \\ = 16\cos^4\frac{(j-1)\pi}{n} \cdot \sin\frac{(2i-1)(j-1)2\pi}{n}, \quad i = 1, \ldots, n/2, \quad j = 1, \ldots, n,\end{gathered}$$

we get

$$Q^T \cdot E \cdot Q' = (Q'^T \cdot E^T \cdot Q)^T = \text{blockdiag}\{E_1, E_2, \ldots, E_{n/2}\}$$

with

$$E_j = \sqrt{2}\begin{bmatrix} \cos^4(\frac{(j-1)\pi}{n})(c_j - is_j) \\ -\sin^4(\frac{(j-1)\pi}{n})(c_j - is_j) \end{bmatrix}, j = 1, \ldots, n/2.$$

Thus,

$$\tilde{M}_c = \text{blockdiag}\{M_{c1}, M_{c2}, \ldots, M_{cn/2}\}$$

with

$$M_{cj} = \begin{bmatrix} 1 - \frac{(\frac{1+c_j}{2})^2(-(\frac{2}{h^2}+\frac{2c}{3})+(\frac{2}{h^2}-\frac{c}{3})c_j)}{-(\frac{1}{2h^2}+\frac{2c}{3})+(\frac{1}{2h^2}-\frac{c}{3})(c_j^2-s_j^2)} & -\frac{(\frac{1+c_j}{2})^2((\frac{2}{h^2}+\frac{2c}{3})+(\frac{2}{h^2}-\frac{c}{3})c_j)}{-(\frac{1}{2h^2}+\frac{2c}{3})+(\frac{1}{2h^2}-\frac{c}{3})(c_j^2-s_j^2)} \\ \frac{(\frac{1-c_j}{2})^2(-(\frac{2}{h^2}+\frac{2c}{3})+(\frac{2}{h^2}-\frac{c}{3})c_j)}{-(\frac{1}{2h^2}+\frac{2c}{3})+(\frac{1}{2h^2}-\frac{c}{3})(c_j^2-s_j^2)} & 1 + \frac{(\frac{1-c_j}{2})^2((\frac{2}{h^2}+\frac{2c}{3})+(\frac{2}{h^2}-\frac{c}{3})c_j)}{-(\frac{1}{2h^2}+\frac{2c}{3})+(\frac{1}{2h^2}-\frac{c}{3})(c_j^2-s_j^2)} \end{bmatrix},$$
$$j = 1, \ldots, n/2.$$

It is easy to see that

$$M_{cj}(1,1) + M_{cj}(2,2) = 1,$$

$$M_{cj}(1,1) \cdot M_{cj}(2,2) - M_{cj}(2,1) \cdot M_{cj}(1,2) = 0,$$

$$j = 1, \ldots, n/2.$$

So if we let $\alpha_j = M_{cj}(1,1)$ and $\beta_j = M_{cj}(1,2)$, then M_{cj} can be simplified as

$$M_{cj} = \begin{bmatrix} \alpha_j & \beta_j \\ \frac{\alpha_j(1-\alpha_j)}{\beta_j} & 1-\alpha_j \end{bmatrix}, \quad j = 1, \ldots, n/2. \tag{3.18}$$

From relations (3.17) and (3.18), we get

$$\tilde{M} = \text{blockdiag}\{M_1, M_2, \ldots, M_{n/2}\}$$

with

$$\begin{aligned} M_j &= M_{cj} \cdot M_{rj} \\ &= \begin{bmatrix} \alpha_j & \beta_j \\ \frac{\alpha_j(1-\alpha_j)}{\beta_j} & 1-\alpha_j \end{bmatrix} \cdot \begin{bmatrix} m_j & 0 \\ 0 & e-m_j \end{bmatrix} \\ &= \begin{bmatrix} \alpha_j m_j & \beta_j(e-m_j) \\ \frac{\alpha_j(1-\alpha_j)}{\beta_j} m_j & (1-\alpha_j)(e-m_j) \end{bmatrix}, \end{aligned} \tag{3.19}$$
$$j = 1, 2, \ldots, n/2.$$

We want to force the eigenvalues of matrices $M_j, j = 1, \ldots, n/2$, to have absolute values less than or equal to 1/2. The characteristic equation of M_j is

$$\det(M_j - \lambda I) = 0,$$

or

$$\lambda^2 - (\alpha_j m_j + (1-\alpha_j)(e-m_j))\lambda = 0.$$

So the eigenvalues of M_j are

$$\lambda_j^{[1]} = 0$$

and

$$\lambda_j^{[2]} = \alpha_j m_j + (1-\alpha_j)(e-m_j).$$

It is trivial to see that

$$\begin{aligned}
\lambda_j^{[2]} &= \alpha_j m_j + (1-\alpha_j)(e-m_j) \\
&= 1 + (2\alpha_j m_j + e - m_j - \alpha_j e + \alpha_j m_j - 1) \\
&= 1 - \tfrac{(6(1-c_j)+ch^2(2+c_j))(6(1+c_j)+ch^2(2-c_j))(1+c_j^2)}{2d(3+ch^2)(6(1-c_j^2)+2ch^2(1+2c_j^2))}.
\end{aligned}$$

Let $\theta_1 = 6(1-c_j)+ch^2(2+c_j), \theta_2 = 6(1+c_j)+ch^2(2-c_j), \theta_3 = 3+ch^2, \theta_4 = 6(1-c_j^2) + 2ch^2(1+2c_j^2)$, and $\Theta = \frac{2(6(1-c_j)+ch^2(2+c_j))(6(1+c_j)+ch^2(2-c_j))(c_j^2+1)}{4d(3+ch^2)(6(1-c_j^2)+2ch^2(1+2c_j^2))}$. Therefore,

$$\lambda_j^{[2]} = 1 - \Theta, \tag{3.20}$$

and

$$\Theta = \frac{1+c_j^2}{2d} \cdot \frac{\theta_1\theta_2}{\theta_3\theta_4}. \tag{3.21}$$

In the following, we show that for $d = 2$,

$$\frac{1}{2} \leq \Theta \leq 1. \tag{3.22}$$

Since

$$\begin{aligned}
\theta_1\theta_2 &= (6(1-c_j)+ch^2(2+c_j))(6(1+c_j)+ch^2(2-c_j)) \\
&= 36 - 36c_j^2 + 24ch^2 + 12ch^2c_j^2 + c^2h^4(4-c_j^2),
\end{aligned} \tag{3.23}$$

and

$$\begin{aligned}
\theta_3\theta_4 &= (3+ch^2)(6(1-c_j^2)+2ch^2(1+2c_j^2)) \\
&= \tfrac{1}{2}[36 - 36c_j^2 + 24ch^2 + 12ch^2c_j^2 + c^2h^4(4-c_j^2)] + \tfrac{9}{2}c^2h^4c_j^2 \\
&= \tfrac{\theta_1\theta_2}{2} + \tfrac{9}{2}c^2h^4c_j^2,
\end{aligned} \tag{3.24}$$

we have

$$\frac{\theta_1\theta_2}{\theta_3\theta_4} = 2 - \frac{9c^2h^4c_j^2}{\theta_3\theta_4}. \tag{3.25}$$

Relations (3.21), (3.25) and $d = 2$ lead to

$$\Theta = \frac{1+c_j^2}{4}(2 - \frac{9c^2h^4c_j^2}{\theta_3\theta_4}). \tag{3.26}$$

If we assume that $ch^2 \leq \frac{3}{2}$, the minimum value of θ_4 is obtained for $c_j = 1$, then

$$0 < 6ch^2 \leq \theta_4. \tag{3.27}$$

Noticing also that $\theta_3 > 0$, it is easy to see that

$$\Theta \leq \frac{1+c_j^2}{4} \cdot 2 \leq 1. \tag{3.28}$$

Consider

$$\frac{h^2}{\theta_3} = \frac{h^2}{3+ch^2} = h^2(\frac{1}{3} - \frac{ch^2}{9} + O(h^4)), \tag{3.29}$$

and

$$\frac{9c^2h^2c_j^2}{\theta_4} \leq \frac{9c^2h^2c_j^2}{6ch^2} = \frac{3cc_j^2}{2}. \tag{3.30}$$

From relations (3.29) and (3.30), we get

$$\frac{9c^2h^4c_j^2}{\theta_3\theta_4} \leq \frac{3cc_j^2}{2}h^2(\frac{1}{3} - \frac{ch^2}{9} + O(h^4)) = \frac{1}{2}ch^2c_j^2 - \frac{c^2h^4c_j^2}{6} + O(h^6) \leq \frac{1}{2}ch^2c_j^2, \tag{3.31}$$

where the last inequality in (3.31) is obtained for sufficiently small h. From relations (3.31) and (3.26), it is easy to see that

$$\begin{aligned} \Theta &\geq \tfrac{1+c_j^2}{4}(2 - \tfrac{1}{2}ch^2c_j^2) \\ &= \tfrac{1}{2} + \tfrac{c_j^2}{2} - \tfrac{1}{8}ch^2c_j^2 - \tfrac{1}{8}ch^2c_j^4 \\ &= \tfrac{1}{2} + \tfrac{c_j^2}{2}(1 - \tfrac{ch^2}{4}(1+c_j^2)) \\ &\geq \tfrac{1}{2}, \end{aligned} \tag{3.32}$$

where the last inequality in (3.32) is obtained for $ch^2 \leq 2$. Relation (3.22) follows from relations (3.28) and (3.32).

Relations (3.20) and (3.22) lead to

$$0 \leq \lambda_j^{[2]} \leq \frac{1}{2},$$

for $c \leq \frac{2}{h^2}$ and sufficient small h. Therefore, the eigenvalues of matrices $M_j, j = 1, \ldots, n/2$, have absolute values less than or equal to 1/2. Thus, $S(M) = S(\tilde{M}) \leq 1/2$. So the spectral radius of M is bounded by 1/2, independently of the step-size. This proves the convergence of the two-grid method for CSC equations corresponding to the model problem (3.1)-(3.2). $\diamondsuit$

Chapter 4

Solution of One-Dimensional Shallow Water Equations

4.1 One-Dimensional Shallow Water Equations

The Shallow Water Equations (SWEs) describe the inviscid flow of a thin layer of fluid in two dimensions [11]. For many years, the SWEs have been used for investigating promising numerical methods for solving atmospheric and oceanic problems. In this thesis, we consider the one-dimensional SWEs in Cartesian coordinates, which are given by

$$\frac{du}{dt} + \phi_x = f_u, \tag{4.1}$$
$$\frac{d\phi}{dt} + \phi^* \cdot u_x = f_\phi, \tag{4.2}$$

where

$$\frac{du}{dt} = \frac{\partial u}{\partial t} + \frac{dx}{dt}\frac{\partial u}{\partial x}, \tag{4.3}$$
$$\frac{d\phi}{dt} = \frac{\partial \phi}{\partial t} + \frac{dx}{dt}\frac{\partial \phi}{\partial x}. \tag{4.4}$$

Recall that the subscript x denotes the partial derivative with respect to x. The velocity $u(x,t)$ and geopotential $\phi(x,t)$ are unknown functions; $f_u(x,t)$ and $f_\phi(x,t)$ are given functions; ϕ^* is a given constant corresponding to the geopotential height. The problem is defined on an interval $[L_1, L_2]$ with periodic boundary conditions. Given the initial states $u(x, t_0)$ and $\phi(x, t_0)$ at time t_0, $x \in [L_1, L_2]$, we would like to find the states $u(x, t_k)$ and $\phi(x, t_k)$ at a later time t_k.

4.2 Semi-Lagrangian Integration Scheme

As for the integration scheme, an explicit leapfrog scheme leads to limitation over the time step-size by the propagation speed of gravitational oscillations. To overcome this limitation, we adopt the two-level semi-Lagrangian semi-implicit integration schemes [17] by associating a semi-Lagrangian treatment of advection with a semi-implicit treatment of gravitational oscillations.

The basic idea of the semi-Lagrangian method can be illustrated by applying it to the one-dimensional advection equation

$$\frac{du}{dt} \equiv \frac{\partial u}{\partial t} + \frac{dx}{dt}\frac{\partial u}{\partial x} = 0, \tag{4.5}$$

where

$$\frac{dx}{dt} = \omega(x, t), \tag{4.6}$$

and $\omega(x, t)$ is a given function or just u itself. Equation (4.5) states that the scalar u is constant along a fluid path (or a trajectory). In Figure 4.1, we denote the exact trajectory in the $x - t$ plane of the fluid particle that arrives at mesh point x_m at time $t_n + \Delta t$ by the solid curve AC, where A and C are the departure and arrival point respectively. We approximate the trajectory by a straight line that is denoted by the dashed line $A'C$. Assuming that $u(x, t)$ is known at all mesh points x at time $t = t_n$, the semi-Lagrangian method tries to integrate (4.5) along the approximate fluid trajectory $A'C$. Thus,

$$\frac{u(x_m, t_n + \Delta t) - u(x_m - \alpha_m, t_n)}{\Delta t} = 0, \tag{4.7}$$

where α_m is the distance that the particle travels in x within time Δt, when following the approximate space-time trajectory $A'C$. If we know α_m, then the value of u at the arrival point

x_m at time $t_n + \Delta t$ is just the value at the upstream point $x_m - \alpha_m$ at time t_n. However, α_m is not known, so we need to compute it. Noting that ω evaluated at the point B is just the inverse of the slope of the straight line $A'C$, we get

$$\frac{\alpha_m}{\Delta t} = \omega(x_m - \frac{\alpha_m}{2}, t_n + \frac{\Delta t}{2}), \tag{4.8}$$

which is an $O(\Delta t^2)$ approximation to equation (4.6)

In order to solve the non-linear equation (4.8) for the displacement α_m, we use a fixed-point iteration scheme given by

$$\alpha_m^{[k+1]} = \Delta t \omega(x_m - \frac{\alpha_m^{[k]}}{2}, t_n + \frac{\Delta t}{2}), \tag{4.9}$$

where the velocity ω at the upstream midpoint $(x_m - \frac{\alpha_m}{2}, t_n + \frac{\Delta t}{2})$ is approximated by linear extrapolation between grid points:

$$\omega(x_m, t_n + \frac{\Delta t}{2}) = \frac{3}{2}\omega(x_m, t_n) - \frac{1}{2}\omega(x_m, t_n - \Delta t). \tag{4.10}$$

Consider the forced-advection equation (4.1), associated with (4.3). A semi-Lagrangian semi-implicit approximation to (4.1) is

$$\frac{u^{n+1} - u^n}{\Delta t} + \frac{1}{2}(\phi_x^{n+1} + \phi_x^n) = f_u^{n+\frac{1}{2}}, \tag{4.11}$$

or equivalently

$$u^{n+1} + \frac{\Delta t}{2}\phi_x^{n+1} = u^n - \frac{\Delta t}{2}\phi_x^n + \Delta t f_u^{n+\frac{1}{2}} \equiv w_u^n, \tag{4.12}$$

where the superscripts "$n + 1$", "$n + \frac{1}{2}$" and "n" denote, respectively, the evaluation at the arrival point $(x_m, t_n + \Delta t)$, the midpoint of the trajectory $(x_m - \frac{\alpha_m}{2}, t_n + \frac{\Delta t}{2})$ and the departure point $(x_m - \alpha_m, t_n)$.

Similarly, a semi-Lagrangian semi-implicit approximation to (4.2) is

$$\frac{\phi^{n+1} - \phi^n}{\Delta t} + \phi^* \frac{1}{2}(u_x^{n+1} + u_x^n) = f_\phi^{n+\frac{1}{2}}, \tag{4.13}$$

or equivalently

$$\phi^{n+1} + \phi^* \frac{\Delta t}{2} u_x^{n+1} = \phi^n - \phi^* \frac{\Delta t}{2} u_x^n + \Delta t f_\phi^{n+\frac{1}{2}} \equiv w_\phi^n. \tag{4.14}$$

Note that (4.11) and (4.13) are centered $O(\Delta t^2)$ approximations to (4.1) and (4.2), where ϕ_x and u_x are evaluated as the time average of their values at the end points of the trajectory; f_u and f_ϕ are evaluated at the midpoint of the trajectory.

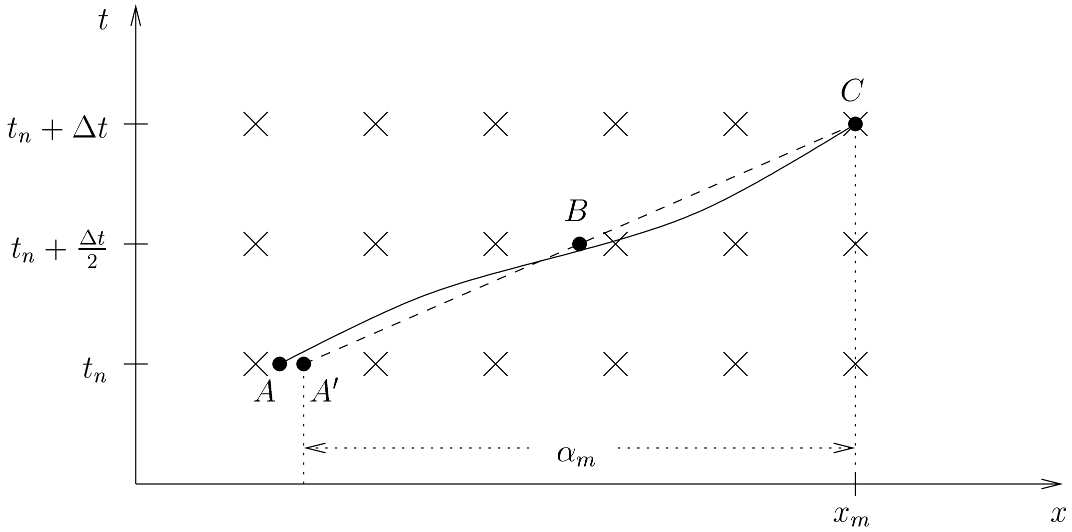


Figure 4.1: An illustration of semi-Lagrangian method for one-dimensional advection. The solid curve (AC) is the actual trajectory and the dashed line ($A'C$) is the approximated trajectory. α_m is the distance the particle is displaced in x in time Δt.

4.3 Algorithm

In summary, the semi-Lagrangian algorithm for the one-dimensional SWEs proceeds as follows:

(a) Approximate the velocity ω at the upstream midpoint $(x_m - \frac{\alpha_m}{2}, t_n + \frac{\Delta t}{2})$ by linear interpolation (4.10).

(b) Apply a few iterations as in (4.9) to solve for the displacements α_m for all mesh points x_m, using some initial guess (usually the respective displacement value at the previous time step).

(c) Interpolate to get u^n, u_x^n, ϕ^n and ϕ_x^n.

(d) Solve (4.12) and (4.14) for u^{n+1} and ϕ^{n+1}.

For step (d), the CSC method is used for spatial discretization, giving rise to a linear system of equations. In the following subsections, we discuss three different ways (analytic elimination - discretization (AED), discretization - algebraic elimination (DAE) and discretization - no elimination (DNE)) of generating a CSC linear system, each having some advantages and disadvantages, which we briefly discuss. We explain the reasons that we adopt the DNE approach and study Jacobi's iterative method and multigrid methods for solving the linear system arising from the CSC discretization.

4.3.1 Analytic Elimination - Discretization (AED)

A Helmholtz equation for ϕ^{n+1} may be derived by eliminating u^{n+1} before spatial discretization. Taking the derivative of (4.12) with respect to x yields

$$u_x^{n+1} + \frac{\Delta t}{2}\phi_{xx}^{n+1} = \Delta t f_{u_x}^{n+\frac{1}{2}} + u_x^n - \frac{\Delta t}{2}\phi_{xx}^n \tag{4.15}$$

or

$$u_x^{n+1} = -\frac{\Delta t}{2}\phi_{xx}^{n+1} + \Delta t f_{u_x}^{n+\frac{1}{2}} + u_x^n - \frac{\Delta t}{2}\phi_{xx}^n. \tag{4.16}$$

The above equation is used to eliminate u_x^{n+1} from (4.14) to yield a Helmholtz equation for ϕ^{n+1}:

$$-\phi^*\frac{\Delta t^2}{4}\phi_{xx}^{n+1} + \phi^{n+1} = w, \tag{4.17}$$

where

$$w = \phi^*\frac{\Delta t^2}{4}\phi_{xx}^n + \phi^n - \phi^*\Delta t u_x^n + \Delta t f_\phi^{n+\frac{1}{2}} - \phi^*\frac{\Delta t^2}{2} f_{u_x}^{n+\frac{1}{2}}.$$

Then we use the CSC method to discretize (4.17) which yields the collocation linear system

$$A\mathbf{c}_\phi^h = \mathbf{w}, \tag{4.18}$$

where

$$A = -\phi^*\frac{\Delta t^2}{4}\tilde{C}_2^P + \tilde{C}_0^P, \tag{4.19}$$

$\mathbf{c}_\phi^h$ is the vector of degrees of freedom of ϕ^{n+1} and $\mathbf{w}$ is the vector arising from evaluating w at the collocation points. Matrices $\tilde{C}_2^P$ and $\tilde{C}_0^P$ are defined as in Chapter 3.

After (4.18) is solved, we have the approximation to ϕ^{n+1}. The approximation to u^{n+1} can be obtained by first evaluating ϕ_x^{n+1} then computing u^{n+1} from (4.12) or first evaluating ϕ^{n+1} then solving (4.14) for u^{n+1} by a finite element method, e.g., the CSC method.

4.3.2 Discretization - Algebraic Elimination (DAE)

First we discretize the equations (4.12) and (4.14) by the CSC method to yield

$$A_0\mathbf{c}_u^h + \frac{\Delta t}{2}A_1\mathbf{c}_\phi^h = \mathbf{w}_u, \tag{4.20}$$

$$A_0\mathbf{c}_\phi^h + \phi^*\frac{\Delta t}{2}A_1\mathbf{c}_u^h = \mathbf{w}_\phi, \tag{4.21}$$

where

$$A_0 = \tilde{C}_0^P, A_1 = \tilde{C}_1^P, \tag{4.22}$$

$\mathbf{w}_u$ and $\mathbf{w}_\phi$ are vectors arising from evaluating w_u^n and w_ϕ^n, respectively, at the collocation points, and $\mathbf{c}_u^h$ and $\mathbf{c}_\phi^h$ are the vectors of degrees of freedom of u^{n+1} and ϕ^{n+1}, respectively.

Eliminating $\mathbf{c}_u^h$ from (4.20)-(4.21), we get

$$(A_0 - \phi^*\frac{\Delta t^2}{4}A_1A_0^{-1}A_1)\mathbf{c}_\phi^h = \mathbf{w}_\phi - \phi^*\frac{\Delta t}{2}A_1A_0^{-1}\mathbf{w}_u. \tag{4.23}$$

Thus, we may solve the above equation for $\mathbf{c}_\phi^h$ first, then compute $\mathbf{c}_u^h$ by solving

$$A_0\mathbf{c}_u^h = \mathbf{w}_u - \frac{\Delta t}{2}A_1\mathbf{c}_\phi^h. \tag{4.24}$$

4.3.3 Discretization - No Elimination (DNE)

Similar to DAE, this approach discretizes (4.12) and (4.14) first. But after discretization, instead of performing algebraic elimination, the linear system arising from the CSC method is solved as one entity. The linear system is in the form of

$$A\mathbf{c}^h = \mathbf{w}, \tag{4.25}$$

where

$$A = \begin{pmatrix} A_0 & \frac{\Delta t}{2} A_1 \\ \phi^* \frac{\Delta t}{2} A_1 & A_0 \end{pmatrix}, \mathbf{c}^h = \begin{pmatrix} \mathbf{c}_u^h \\ \mathbf{c}_\phi^h \end{pmatrix}, \text{and}, \mathbf{w} = \begin{pmatrix} \mathbf{w}_u \\ \mathbf{w}_\phi \end{pmatrix}. \tag{4.26}$$

4.3.4 Comparison of the three approaches

The AED approach results in one Helmholtz equation, which can be easily discretized by using standard discretization methods, e.g., the CSC method or finite difference. The resulting linear system is of size $n \times n$, has nice properties, such as symmetry and diagonal dominance, and can be solved by fast direct or iterative solvers, for example, the Fast Fourier Transform or multigrid methods. However, this method does not preserve the stability of Rossby waves, as several researchers have indicated, [19] and [20], etc.

The DAE approach does not suffer from the instability problem of the first approach, still requires solving a linear system of size $n \times n$, but the coefficient matrix of the system is nonstandard. More specifically, the matrix $A_0 - \phi^* \frac{\Delta t^2}{4} \cdot A_1 \cdot A_0^{-1} A_1$ is not computed explicitly, due to the presence of A_0^{-1}. Iterative methods are usually applied for the solution of (4.23). Thus only the application of the matrix on a vector is required, and this is performed by solving a system with matrix A_0, however, this computation is required at each iteration. Recall that solving (4.23) is required for each time step and each fixed-point iteration, therefore we have three nested loops around the solution of the system with matrix A_0.

The DNE approach has the same nice stability properties as the second approach, but it results in solution of a double-sized linear system, i.e. a system of size $2n \times 2n$. Fortunately, iterative methods come into rescue. With efficient iterative methods, especially multigrid methods, we could bear the burden of solving the size-increased linear system for the purpose of stability.

In this thesis, we focus primarily on the DNE approach, and study the performance of multigrid and Jacobi's iterative methods when applied to (4.25), which is generated by DNE. We also consider the AED approach, apply Jacobi and multigrid to (4.17), and compare with the respective results of the DNE approach.

4.4 Jacobi's Iterative Method for CSC Discretization of SWEs

In this section, we use Jacobi's iterative method to solve the linear systems (4.18) and (4.25), which arise from the CSC method applied to the SWEs discretized by AED and DNE approaches respectively, and study its convergence.

4.4.1 Jacobi's Method for AED

In the solution of the one-dimensional SWEs, if we use the AED approach, we will have to solve the Helmholtz equation (4.17), which could be written in the form of second order equation (3.1), where $c = \frac{1}{\phi^*}\frac{4}{\Delta t^2} > 0$. Thus, we could apply Jacobi's iterative method to solving the CSC linear system (3.3)-(3.4).

Convergence Analysis

From the conclusion in section 3.2, a sufficient condition we have for Jacobi's iterative method with damping factor $d = 1$ to converge with rate bounded by α and a contraction factor less than α is $ch^2 > \frac{6-6\alpha}{2\alpha+1}$ for sufficiently small h, where $c = \frac{1}{\phi^*}\frac{4}{\Delta t^2}$, which leads to

$$\frac{\Delta t}{h} < 2\sqrt{\frac{2\alpha+1}{6-6\alpha}}\frac{1}{\sqrt{\phi^*}}. \tag{4.27}$$

If $\alpha = 1/2$ and $\phi^* = 10$, then

$$\frac{\Delta t}{h} < \frac{1}{\sqrt{\phi^*}}\frac{\sqrt{24}}{3} = 0.516.$$

We conducted experiments in solving the second order two-point BVP (3.1)-(3.2) with $\Omega \equiv [0, 2\pi]$ and the right hand side g was chosen so that the exact solution to the problem is $u(x) = \sin x + \cos x$. We used Jacobi's method for AED with damping factor $d = 1$, and chose relative residual as the stopping criteria and set the tolerance to be 10^{-10}.

We tabulate the spectral radius of the Jacobi iteration matrix and the number of iterations used with respect to different $\Delta t/h$ ratios in Table 4.1. The results confirm our convergence analysis and show that the Jacobi's method works best when $\Delta t/h$ ratio is around 0.255. Notice that the relation between the spectral radius of iteration matrix and the $\Delta t/h$ ratio is independent of the problem. In Figure 4.2(a) we plot the spectral radius vs. $\Delta t/h$. The figure shows that Jacobi's method is very sensitive to $\Delta t/h$ and works best for $\Delta t/h$ around 0.255, which is a relatively small value. It converges very slowly or even diverges with large $\Delta t/h$, e.g., greater than 4. Jacobi's method with damping factor $d = 2$ works worse than with $d = 1$. Therefore, Jacobi's method does not allow large time step-size.

n h	8 7.85-e01	16 3.93-e01	32 1.96-e01	64 9.82-e02	128 4.91-e02	256 2.45-e02
c = 2560.0 Δt = 1.25e-02	1.59e-02 0.50 23	3.18e-02 0.49 29	6.37e-02 0.46 29	1.27e-01 0.34 22	2.55e-01 0.01 5	5.09e-01 0.49 33
c = 640.0 Δt = 2.50e-02	3.18e-02 0.49 22	6.37e-02 0.46 27	1.27e-01 0.34 21	2.55e-01 0.01 5	5.09e-01 0.49 33	1.02e+00 0.83 50+
c = 160.0 Δt = 5.00e-02	6.37e-02 0.46 21	1.27e-01 0.34 20	2.55e-01 0.01 5	5.09e-01 0.49 33	1.02e+00 0.83 50+	2.04e+00 0.95 50+
c = 40.0 Δt = 1.00e-01	1.27e-01 0.34 17	2.55e-01 0.01 5	5.09e-01 0.49 32	1.02e+00 0.83 50+	2.04e+00 0.95 50+	4.07e+00 0.99 50+
c = 10.0 Δt = 2.00e-01	2.55e-01 0.01 5	5.09e-01 0.49 30	1.02e+00 0.83 50+	2.04e+00 0.95 50+	4.07e+00 0.99 50+	8.15e+00 1.00 -
c = 2.5 Δt = 4.00e-01	5.09e-01 0.49 22	1.02e+00 0.83 50+	2.04e+00 0.95 50+	4.07e+00 0.99 50+	8.15e+00 1.00 -	1.63e+01 1.00 -

Table 4.1: Spectral radius of iteration matrix and number of iterations with respect to $\Delta t/h$ for Jacobi's method for AED. In each cell, the first entry is $\Delta t/h$, the second is spectral radius of the iteration matrix, and the third is the number of iterations with tolerance 10^{-10}.

4.4.2 Jacobi's Method for DNE

Suppose D is the diagonal matrix holding d times the diagonal elements of A which is defined as in (4.26), where d is the damping factor. Then D is a $2n \times 2n$ matrix and

$$D = \frac{d}{6}\begin{pmatrix} 4 & & & & \\ & 4 & & & \\ & & \ddots & & \\ & & & 4 & \\ & & & & 4 \end{pmatrix}. \tag{4.28}$$

Thus, Jacobi's iterative method when applied to (4.25) gives the scheme

$$D\mathbf{c}^{h[k+1]} = (D - A)\mathbf{c}^{h[k]} + \mathbf{w}, \tag{4.29}$$

or

$$\mathbf{c}^{h[k+1]} = G\mathbf{c}^{h[k]} + D^{-1}\mathbf{w}, \tag{4.30}$$

where G is the iteration matrix

$$G = I - D^{-1}A = \begin{pmatrix} \tilde{A}_0 & -\frac{3\Delta t}{4}A_1 \\ -\phi^*\frac{3\Delta t}{4}A_1 & \tilde{A}_0 \end{pmatrix}, \tag{4.31}$$

with

$$\tilde{A}_0 = -\frac{1}{4}\begin{pmatrix} \frac{4}{d}-4 & 1 & & & 1 \\ 1 & \frac{4}{d}-4 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & \frac{4}{d}-4 & 1 \\ 1 & & & 1 & \frac{4}{d}-4 \end{pmatrix}. \tag{4.32}$$

Convergence Analysis

Taking into account that $\tilde{A}_0$ is a circulant matrix, it can be shown that eigenvalues of $\tilde{A}_0$ are given by

$$\lambda_j^{\{0\}} = -\frac{1}{4}(\frac{4}{d} - 4 + \cos\frac{2(j-1)\pi}{n}), \quad j = 1, \dots, n. \tag{4.33}$$

Similarly, the eigenvalues of A_1 are given by

$$\lambda_j^{\{1\}} = \frac{\imath}{h} \sin \frac{2(j-1)\pi}{n}, \quad j = 1, \ldots, n, \tag{4.34}$$

where $\imath = \sqrt{-1}$.

Note that $\lambda_j^{\{0\}}$ and $\lambda_j^{\{1\}}$ are related to the same eigenvector.

Consider a $2n \times 2n$ matrix B, which consists of four $n \times n$ blocks, i.e.,

$$B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix},$$

where $B_{ij}, i, j = 1, 2$, have the same eigenvectors. Then any eigenvalue of B is given by the formula [15]:

$$\lambda_B = \frac{1}{2}(\lambda_{11} + \lambda_{22} \pm \sqrt{(\lambda_{11} + \lambda_{22})^2 - 4(\lambda_{11}\lambda_{22} - \lambda_{12}\lambda_{21})}), \tag{4.35}$$

where λ_{ij} is an eigenvalue of block B_{ij} , respectively, for $i, j = 1, 2$, and λ_{ij} corresponds to the same eigenvector.

By applying (4.35), the eigenvalues of G are given by

$$\begin{aligned} \lambda_j^G &= \tfrac{1}{2}(2\lambda_j^{\{0\}} \pm \sqrt{(2\lambda_j^{\{0\}})^2 - 4({\lambda_j^{\{0\}}}^2 - \tfrac{9}{16}\Delta t^2 \phi^* {\lambda_j^{\{1\}}}^2)}) \\ &= \lambda_j^{\{0\}} \pm \tfrac{3}{4}\Delta t \sqrt{\phi^*} \lambda_j^{\{1\}} \\ &= -\tfrac{1}{4}(\tfrac{4}{d} - 4 + 2c_j) \pm \imath \tfrac{3}{4} \tfrac{\Delta t}{h} \sqrt{\phi^*} s_j, \end{aligned} \tag{4.36}$$

where $s_j \equiv \sin \frac{2(j-1)\pi}{n}$ and $c_j \equiv \cos \frac{2(j-1)\pi}{n}$, $j = 1, \ldots, n$.

We want to have $\left|\lambda_j^G\right| < \alpha$, for some $\alpha < 1$, which implies that Jacobi's iterative method converges for all initial guess vectors.

For damping factor $d = 1$,

$$\begin{aligned} & & \left|\lambda_j^G\right| &< \alpha \\ &\Leftrightarrow & \left|-\tfrac{c_j}{2} \pm \imath \tfrac{3}{4} \tfrac{\Delta t}{h} \sqrt{\phi^*} s_j\right| &< \alpha \\ &\Leftrightarrow & \tfrac{c_j^2}{4} + \tfrac{9}{16} \tfrac{\Delta t^2}{h^2} \phi^* s_j^2 &< \alpha^2 \\ &\Leftrightarrow & \tfrac{1-s_j^2}{4} + \tfrac{9}{16} \tfrac{\Delta t^2}{h^2} \phi^* s_j^2 &< \alpha^2 \\ &\Leftrightarrow & (9\tfrac{\Delta t^2}{h^2}\phi^* - 4)s_j^2 &< 16\alpha^2 - 4. \end{aligned} \tag{4.37}$$

A sufficent condition to the above is

$$\begin{aligned} 9\tfrac{\Delta t^2}{h^2}\phi^* - 4 &< 16\alpha^2 - 4 \\ \Leftrightarrow \quad (\tfrac{\Delta t}{h})^2\phi^* &< \tfrac{16\alpha^2}{9} \\ \Leftrightarrow \quad \tfrac{\Delta t}{h} &< \tfrac{4\alpha}{3}\tfrac{1}{\sqrt{\phi^*}}. \end{aligned} \tag{4.38}$$

Therefore, if we could keep $\frac{\Delta t}{h} < \frac{4\alpha}{3}\frac{1}{\sqrt{\phi^*}}$, Jacobi's iterative method would converge with a rate bounded by α and a contraction factor less than α when applied to (4.25).

If $\alpha = 1/2$ and $\phi^* = 10$, then

$$\frac{\Delta t}{h} < 0.211$$

We conducted experiments in solving the one-dimensional SWEs (4.1)- (4.2), with periodic boundary conditions in $x \in [0, 2\pi]$. The right hand side is chosen so that the exact solution to the problem is $u(x,t) = \sin x \sin t$ and $v(x,t) = \sin x \cos t$. We used Jacobi's method for DNE with damping factor $d = 1$, and chose the relative residual as the stopping criteria and set the tolerance to be 10^{-10}.

We tabulate the spectral radius of the Jacobi's iteration matrix and the number of iterations used with respect to different $\Delta t/h$ ratios into Table 4.2. The results confirm our convergence analysis and show that the Jacobi's method works best with small $\Delta t/h$ ratio. In Figure 4.2(b) we plot the spectral radius vs. $\Delta t/h$. The figure shows that Jacobi's method with damping factor $d = 1$ is sensitive to $\Delta t/h$ and works best for $\Delta t/h$ less than 0.21, which is a relatively small value. It diverges with big $\Delta t/h$, e.g., greater than 0.4. It shows no convergence with damping factor $d = 2$.

n h	8 7.85-e01	16 3.93-e01	32 1.96-e01	64 9.82-e02	128 4.91-e02	256 2.45-e02
Δt = 1.25e-02	1.59e-02 0.50 26	3.18e-02 0.50 32	6.37e-02 0.50 34	1.27e-01 0.50 35	2.55e-01 0.60 35	5.09e-01 1.21 -
Δt = 2.50e-02	3.18e-02 0.50 26	6.37e-02 0.50 31	1.27e-01 0.50 34	2.55e-01 0.60 35	5.09e-01 1.21 -	1.02e+00 2.42 -
Δt = 5.00e-02	6.37e-02 0.50 28	1.27e-01 0.50 33	2.55e-01 0.60 35	5.09e-01 1.21 -	1.02e+00 2.42 -	2.04e+00 4.83 -
Δt = 1.00e-01	1.27e-01 0.50 30	2.55e-01 0.60 38	5.09e-01 1.21 -	1.02e+00 2.42 -	2.04e+00 4.83 -	4.07e+00 9.66 -
Δt = 2.00e-01	2.55e-01 0.60 43	5.09e-01 1.21 -	1.02e+00 2.42 -	2.04e+00 4.83 -	4.07e+00 9.66 -	8.15e+00 19.33 -
Δt = 4.00e-01	5.09e-01 1.21 -	1.02e+00 2.42 -	2.04e+00 4.83 -	4.07e+00 9.66 -	8.15e+00 19.33 -	1.63e+01 38.65 -

Table 4.2: Spectral radius of iteration matrix and number of iterations with respect to $\Delta t/h$ for Jacobi's method for DNE. In each cell, the first entry is $\Delta t/h$, the second is spectral radius of the iteration matrix, and the third is the number of iterations with tolerance 10^{-10}.

4.5 Multigrid Methods for CSC Discretization of SWEs

In this section, we use multigrid methods integrated with a damped Richardson relaxation scheme as smoother to solve the linear systems (4.18) and (4.25), which arise from the CSC method applied to the SWEs discretized by AED and DNE approaches respectively.

4.5.1 Multigrid Methods for AED

In the solution of the one-dimensional SWEs, if we use the AED approach, we will have to solve Helmholtz equation (4.17), which could be written in the form of second order equation (3.1), where $c = \frac{1}{\phi^*}\frac{4}{\Delta t^2} > 0$. Thus, we could apply the multigrid method to solving the CSC linear system (3.3)-(3.4).

Convergence Analysis

From Theorem 3.3.1, the condition for spectral radius of the two-grid method's iteration matrix to be less than $\alpha = 1/2$ with $d = 2$ is $ch^2 \leq \frac{3}{2}$, where $c = \frac{1}{\phi^*}\frac{4}{\Delta t^2}$. If let $\phi^* = 10$, we have

$$\frac{\Delta t}{h} \geq \frac{1}{\sqrt{\phi^*}}\frac{\sqrt{24}}{3} = 0.516. \tag{4.39}$$

We conducted experiments in solving the second order two-point BVP (3.1)-(3.2) with $\Omega \equiv [0, 2\pi]$ and the right hand side g was chosen so that the exact solution to the problem is $u(x) = \sin x + \cos x$. We used the two-grid method for AED to solve the problem. We used diagonal preconditioning with damping factor 2 as the relaxation scheme and apply one relaxation per level and chose relative residual as the stopping criteria and set the tolerance to be 10^{-10}.

We tabulate the spectral radius of the iteration matrix and the number of iterations used with respect to different $\Delta t/h$ ratios into Table 4.3. In Figure 4.2(c) we plot the spectral radius against $\Delta t/h$. The results confirm our convergence analysis and show that, in contrary to Jacobi's method, the two-grid method with damping factor 2 works better with larger $\Delta t/h$. This is a big advantage over Jacobi's method, meaning that the two-grid method can solve big size (large grid size) problem fast and allows large time step-size. The two-grid method converges for almost all range of $\Delta t/h$, though the spectral radius are larger than those of Jacobi's method with $d = 1$ for small $\Delta t/h$, e.g., less than 0.25. However, if we choose the diagonal preconditioning damping factor to be 1, then the two-grid method shows almost the same convergence trend as Jacobi's method with $d = 1$. This gives us an idea to construct an efficient multigrid solver for AED: choosing damping factor 1 for smaller $\Delta t/h$ and damping factor 2 for larger $\Delta t/h$.

n h	8 7.85-e01	16 3.93-e01	32 1.96-e01	64 9.82-e02	128 4.91-e02	256 2.45-e02
c = 2560.0 Δt = 1.25e-02	1.59e-02 0.749 47	3.18e-02 0.744 47	6.37e-02 0.728 38	1.27e-01 0.669 25	2.55e-01 0.505 13	5.09e-01 0.500 6
c = 640.0 Δt = 2.50e-02	3.18e-02 0.744 46	6.37e-02 0.728 45	1.27e-01 0.669 32	2.55e-01 0.505 17	5.09e-01 0.500 8	1.02e+00 0.500 5
c = 160.0 Δt = 5.00e-02	6.37e-02 0.728 44	1.27e-01 0.669 38	2.55e-01 0.505 21	5.09e-01 0.500 10	1.02e+00 0.500 6	2.04e+00 0.500 4
c = 40.0 Δt = 1.00e-01	1.27e-01 0.669 39	2.55e-01 0.505 25	5.09e-01 0.500 13	1.02e+00 0.500 7	2.04e+00 0.500 5	4.07e+00 0.500 4
c = 10.0 Δt = 2.00e-01	2.55e-01 0.505 29	5.09e-01 0.500 15	1.02e+00 0.500 8	2.04e+00 0.500 6	4.07e+00 0.500 4	8.15e+00 0.500 3
c = 2.5 Δt = 4.00e-01	5.09e-01 0.500 20	1.02e+00 0.500 11	2.04e+00 0.500 7	4.07e+00 0.500 5	8.15e+00 0.500 4	1.63e+01 0.500 3

Table 4.3: Spectral radius of iteration matrix and number of iterations with respect to $\Delta t/h$ for the two-grid method for AED. In each cell, the first entry is $\Delta t/h$, the second is spectral radius of the iteration matrix, and the third is the number of iterations with tolerance 10^{-10}.

4.5.2 Multigrid Methods for DNE

Since (4.25) is a linear system arising from a system of two PDEs, the extension and restriction operators E and R, respectively, are given by

$$E = \begin{pmatrix} E^P & 0 \\ 0 & E^P \end{pmatrix}, \tag{4.40}$$

and

$$R = \begin{pmatrix} R^P & 0 \\ 0 & R^P \end{pmatrix}, \tag{4.41}$$

where E^P and R^P are, respectively, extension and restriction operators for cubic splines arising from a single differential equation with periodic boundary conditions, as defined in Relations (2.82) and (2.83).

Convergence Analysis

The iteration matrix of two-grid method for DNE is so complex, that we are unable to find the explicit form of the spectral radius of the iteration matrix. In spite of lack of theoretical proof of convergence, we could give a pretty good estimation of the convergence for multigrid methods by performing numerical experiments, plotting the spectral radius numerically, and comparing the results with the above three cases.

We conducted same experiments as Jacobi's method for DNE in solving the one-dimensional SWEs (4.1)- (4.2), with periodic boundary conditions in $x \in [0, 2\pi]$. The right hand side is chosen so that the exact solution to the problem is $u(x,t) = \sin x \sin t$ and $v(x,t) = \sin x \cos t$. We applied the two-grid method for DNE to solve the problem. We used diagonal preconditioning with damping factor either 1 or 2 as the relaxation scheme and apply one relaxation per level and chose relative residual as the stopping criteria and set the tolerance to be 10^{-10}.

We tabulate the numerically calculated spectral radius of the iteration matrix and the number of iterations with respect to different $\Delta t/h$ ratios into Table 4.4 for damping factor 1 and Table 4.5 for damping factor 2. In Figure 4.2(d) we plot the spectral radius against $\Delta t/h$. The results show that multigrid methods for DNE have some advantages over Jacobi's method for

DNE as multigrid methods for AED do over Jacobi's method for AED.

For damping factor 1, both the two-grid and Jacobi's method show similar convergence trend. The spectral radius is 0.5 if $\Delta t/h$ is less than about 0.21, then it increases as $\Delta t/h$ increases where the two-grid method is only slightly better than Jacobi's method in terms of having slightly smaller spectral radius. Eventually, both methods diverge when $\Delta t/h$ reaches about 0.41.

For damping factor 2, Jacobi's method shows no convergence at all by having spectral radius no less than 1. In the contrary, the two-grid method has spectral radius less than 1 if $\Delta t/h$ is less than about 0.71. Though its spectral radius is no less than 0.75, the two-grid method definitely outperforms the Jacobi's method.

From comparison, we could conclude that with damping factor 1, multigrid methods work similarly as Jacobi's method does; with damping factor 2, multigrid methods tend to work better for larger $\Delta t/h$ ratio. This give us the idea to construct an efficient multigrid solver for DNE: choosing damping factor 1 for $\Delta t/h < 0.21$ and damping factor 2 for $\Delta t/h > 0.21$. However we must keep $\Delta t/h$ less than about 0.7 in order to guarantee convergence. Thus, the two-grid method for DNE allows about twice as large time step-size as Jacobi's method for DNE. We could also increase the number of relaxations applied per level to reduce the number of iterations. For example, as shown in Table 4.6, if applying two relaxations with damping factor 1 at each level, we reduced the number of iterations by nearly half in all cases. Notice that each iteration of multigrid methods consists of relaxation(s), one restriction, one extension, and one solution of linear system on the coarse grid. By applying two relaxations instead of one, the number of iterations are reduced by half, which saves the computation cost from performing restriction, extension and solution of the linear system.

n h	8 7.85-e01	16 3.93-e01	32 1.96-e01	64 9.82-e02	128 4.91-e02	256 2.45-e02
Δt = 1.25e-02	1.59e-02 0.50 20	3.18e-02 0.50 22	6.37e-02 0.50 19	1.27e-01 0.50 16	2.55e-01 0.55 12	5.09e-01 1.46 -
Δt = 2.50e-02	3.18e-02 0.50 22	6.37e-02 0.50 22	1.27e-01 0.50 19	2.55e-01 0.55 17	5.09e-01 1.45 -	1.02e+00 5.83 -
Δt = 5.00e-02	6.37e-02 0.50 26	1.27e-01 0.50 22	2.55e-01 0.55 23	5.09e-01 1.45 -	1.02e+00 5.82 -	2.04e+00 23.31 -
Δt = 1.00e-01	1.27e-01 0.50 28	2.55e-01 0.55 32	5.09e-01 1.44 -	1.02e+00 5.80 -	2.04e+00 23.28 -	4.07e+00 93.25 -
Δt = 2.00e-01	2.55e-01 0.54 36	5.09e-01 1.42 -	1.02e+00 5.77 -	2.04e+00 23.22 -	4.07e+00 93.13 -	8.15e+00 373.02 -
Δt = 4.00e-01	5.09e-01 1.46 -	1.02e+00 5.73 -	2.04e+00 23.11 -	4.07e+00 92.89 -	8.15e+00 372.54 -	1.63e+01 1492.10 -

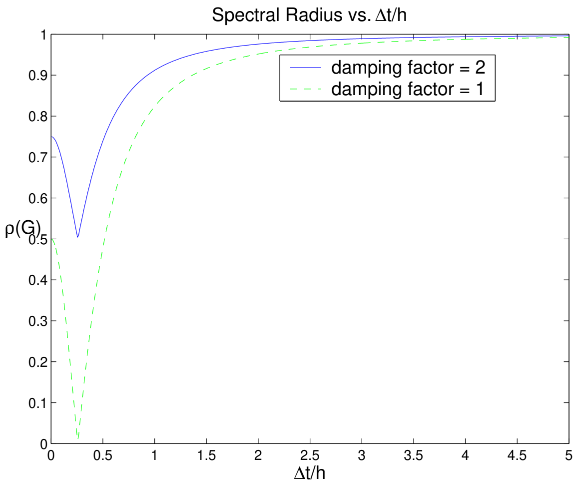
Table 4.4: Spectral radius of iteration matrix and number of iterations with respect to $\Delta t/h$ for the two-grid method for DNE. In each cell, the first entry is $\Delta t/h$, the second is spectral radius of the iteration matrix, and the third is the number of iterations with tolerance 10^{-10}. At each level, one relaxation with damping factor 1 is performed.

n h	8 7.85-e01	16 3.93-e01	32 1.96-e01	64 9.82-e02	128 4.91-e02	256 2.45-e02
Δt = 1.25e-02	1.59e-02 0.75 49	3.18e-02 0.75 50	6.37e-02 0.75 43	1.27e-01 0.75 33	2.55e-01 0.75 23	5.09e-01 0.79 13
Δt = 2.50e-02	3.18e-02 0.75 50	6.37e-02 0.75 50	1.27e-01 0.75 44	2.55e-01 0.75 35	5.09e-01 0.79 25	1.02e+00 2.42 -
Δt = 5.00e-02	6.37e-02 0.75 50	1.27e-01 0.75 48	2.55e-01 0.75 45	5.09e-01 0.79 36	1.02e+00 2.42 -	2.04e+00 11.16 -
Δt = 1.00e-01	1.27e-01 0.75 50+	2.55e-01 0.75 50+	5.09e-01 0.79 43+	1.02e+00 2.41 -	2.04e+00 11.15 -	4.07e+00 46.14 -
Δt = 2.00e-01	2.55e-01 0.75 50+	5.09e-01 0.80 50+	1.02e+00 2.41 -	2.04e+00 11.13 -	4.07e+00 46.09 -	8.15e+00 186.04 -
Δt = 4.00e-01	5.09e-01 0.79 50+	1.02e+00 2.41 -	2.04e+00 11.10 -	4.07e+00 46.00 -	8.15e+00 185.82 -	1.63e+01 745.60 -

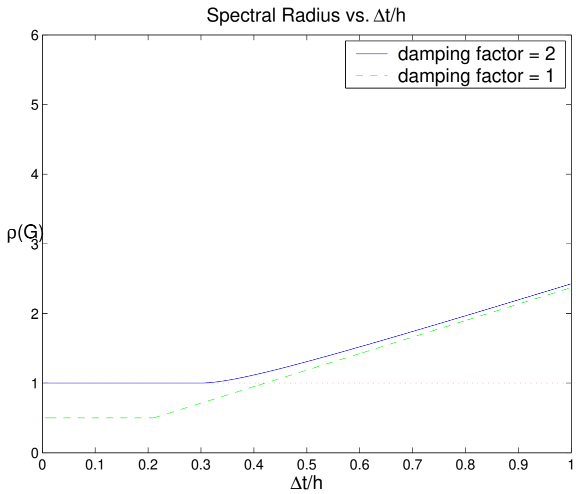
Table 4.5: Spectral radius of iteration matrix and number of iterations with respect to $\Delta t/h$ for the two-grid method for DNE. In each cell, the first entry is $\Delta t/h$, the second is spectral radius of the iteration matrix, and the third is the number of iterations with tolerance 10^{-10}. At each level, one relaxation with damping factor 2 is performed.

n h	8 7.85-e01	16 3.93-e01	32 1.96-e01	64 9.82-e02	128 4.91-e02	256 2.45-e02
Δt = 1.25e-02	1.59e-02 0.50 10	3.18e-02 0.50 11	6.37e-02 0.50 10	1.27e-01 0.50 8	2.55e-01 0.55 6	5.09e-01 1.46 -
Δt = 2.50e-02	3.18e-02 0.50 11	6.37e-02 0.50 11	1.27e-01 0.50 10	2.55e-01 0.55 8	5.09e-01 1.45 -	1.02e+00 5.83 -
Δt = 5.00e-02	6.37e-02 0.50 13	1.27e-01 0.50 11	2.55e-01 0.55 11	5.09e-01 1.45 -	1.02e+00 5.82 -	2.04e+00 23.31 -
Δt = 1.00e-01	1.27e-01 0.50 14	2.55e-01 0.55 16	5.09e-01 1.44 -	1.02e+00 5.80 -	2.04e+00 23.28 -	4.07e+00 93.25 -
Δt = 2.00e-01	2.55e-01 0.54 21	5.09e-01 1.42 -	1.02e+00 5.77 -	2.04e+00 23.22 -	4.07e+00 93.13 -	8.15e+00 373.02 -
Δt = 4.00e-01	5.09e-01 1.46 -	1.02e+00 5.73 -	2.04e+00 23.11 -	4.07e+00 92.89 -	8.15e+00 372.54 -	1.63e+01 1492.10 -

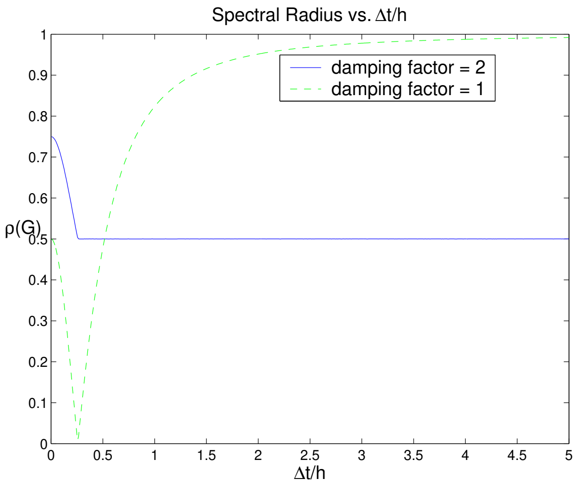
Table 4.6: Spectral radius of iteration matrix and number of iterations with respect to $\Delta t/h$ for the two-grid method for DNE. In each cell, the first entry is $\Delta t/h$, the second is spectral radius of the iteration matrix, and the third is the number of iterations with tolerance 10^{-10}. At each level, two relaxations with damping factor 1 is performed.



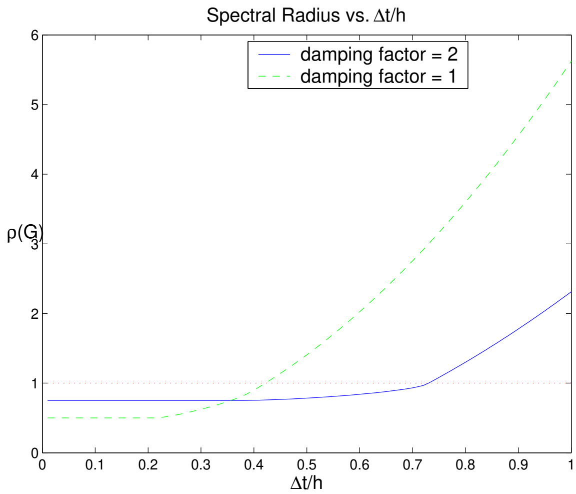
(a) Jacobi for AED



(b) Jacobi for DNE



(c) Multigrid for AED



(d) Multigrid for DNE

Figure 4.2: Plots of spectral radius of iteration matrix G with respect to $\Delta t/h$.

Chapter 5

Numerical Results

In this chapter, we present numerical results to demonstrate the performance of multigrid and cubic spline collocation methods for advection equations and verify the conclusions from the analysis.

The methods were implemented in Matlab (Version 6.5 Release 13) and all the computations were carried out on a timesharing server (Dell PowerEdge 2650 with 2 P4Xeon-2.8GHz CPUs running RedHat 7.3) in double precision.

When presenting the error, we use the notations $\|\cdot\|_{x_i}$, $\|\cdot\|_{\tau_i}$, $\|\cdot\|_{\lambda_i}$ and $\|\cdot\|_{\infty}$ to denote the maximum absolute value at nodal points, midpoints, Gaussian points, and global points, respectively. The midpoints are defined as the midpoints of the n subintervals. There are two Gaussian points in each subinterval, one is $\frac{3-\sqrt{3}}{6}h$ from the left nodal point and the other is the same distance from the right nodal point. The global points are defined as 1013 equally spaced evaluation points. The order of convergence is defined as $\frac{\log(e_1/e_2)}{\log(n_2/n_1)}$, where n_1, n_2 are numbers of subintervals with $n_1 < n_2$ and e_1, e_2 are error corresponding to n_1 and n_2 grid points, respectively.

We apply the full multigrid, the V-cycle multigrid and/or the two-grid methods for CSC to various problems, with either periodic or homogeneous Dirichlet conditions. For all the methods, we use diagonal preconditioning with damping factor 2 as a relaxation scheme and apply two relaxation iterations per level. The coarsest problem size solved is $\min(\frac{n}{2}, 8)$, where n is the number of the discretization points. The initial guess of the solution vector is chosen to be the zero vector for the first step of the CSC and the solution of the first step for the second step of the CSC. The respective error and residual are denoted in the plots as the errors and residual of iteration 0.

We use $\frac{\|r^{(k)}\|}{\|r^{(0)}\|} \leq \varepsilon$ as the stopping criterion for multigrid methods when applied to the CSC linear system, where $r^{(k)}$ is the residual of the linear system solved at the kth iteration, for $k = 1, \cdots, m$ (m is the maximum number of iterations, e.g., 20), $r^{(0)}$ is the residual corresponding to the initial guess of the solution vector, and ε is the tolerance.

With Problems 1 and 2, we apply the full multigrid, the V-cycle multigrid and the two-grid method for CSC to single second order two-point BVPs, to show the efficiency of our methods. We choose tolerance ε to be 10^{-6} and 10^{-4}, respectively, for the first or second step of the optimal CSC. At each iteration, we compute the residual of the linear system and the maximum absolute error of the collocation approximation on global points. The latter can be considered as an approximation to the maximum norm of the error of the collocation approximation.

With Problems 3, 4, 5 and 6, we are considering solving the SWEs (4.1) and (4.2), with different initial conditions. For these problems, we use the semi-Lagrangian semi-implicit integration scheme in time and the Discretization - No Elimination (DNE) approach in space with the standard CSC method, and apply the full multigrid method to solving the resulting CSC linear system at each time step. As for the units, we stick to meters and seconds as appropriate.

In all the tables, u_Δ denotes the approximate solution obtained by our methods.

Problem 1:

Determine $u = u(x)$ such that the second order two-point BVP

$$\mathrm{L}u \equiv u_{xx} - u = g \quad \text{in } \Omega \equiv [0, 1], \tag{5.1}$$

subject to periodic boundary conditions (3.2) is satisfied. The function g is chosen so that the exact solution to the problem is $u(x) = \sin(2\pi x) + \cos(2\pi x)$.

We apply the full multigrid, the V-cycle multigrid and the two-grid methods for CSC to this problem. The maximum absolute error, the respective orders of convergence, and the corresponding number of iterations are presented in Tables 5.1, 5.2 and 5.3. The results show second order convergence for the first step of CSC, and fourth order convergence for the second step.

In order to show how the grid size affects the convergence of the three multigrid methods, we plot the residual for 12 iterations of the full multigrid, the V-cycle multigrid and the two-grid methods, respectively, for 32, 64, 128 and 256 discretization grid points, as in Figures 5.1, 5.3 and 5.5. Figures 5.2, 5.4 and 5.6 show the error for 12 iterations of the full multigrid, the V-cycle multigrid and the two-grid methods, respectively, for 32, 64, 128 and 256 discretization grid points. The experiments show that the full multigrid method converge to within the discretization error in 1 iteration for both steps of CSC. This behaviour of the full multigrid method is typical for model problems. The two-grid method is just a bit behind and needs 2 iterations, while the V-cycle multigrid method requires 3 - 4 iterations. It also shows that both the full multigrid and two-grid methods converge faster for larger grids, while the V-cycle multigrid converges with about equally speed for either large or small grids. We also plot the residual for 12 iterations of the full multigrid, the V-cycle multigrid and the two-grid methods, respectively, for 256 discretization grid points in Figure 5.7, in order to compare their convergence. For the first step of CSC, the two-grid method is the fastest in reducing the residual. The full multigrid method is faster at the beginning, then slows down a little bit. The V-cycle multigrid method is slowest. For the second step of CSC, the full multigrid method is slightly faster than the two-grid method, while the V-cycle is again the slowest.

n	$\|u - u_\Delta\|_\infty$	$\|u' - u'_\Delta\|_{x_i,\tau_i,\infty}$			$\|u'' - u''_\Delta\|_{\lambda_i,\infty}$		no. of iter.
error - optimal CSC first step							
8	6.87e-02	4.51e-01	3.92e-01	4.51e-01	2.62e+00	3.87e+00	6
16	1.76e-02	1.12e-01	1.08e-01	1.12e-01	6.89e-01	1.04e+00	4
32	4.42e-03	2.79e-02	2.76e-02	2.79e-02	1.74e-01	2.63e-01	5
64	1.11e-03	6.96e-03	6.95e-03	6.96e-03	4.37e-02	6.60e-02	3
128	2.77e-04	1.74e-03	1.74e-03	1.74e-03	1.09e-02	1.65e-02	3
256	6.92e-05	4.35e-04	4.35e-04	4.35e-04	2.73e-03	4.10e-03	2
orders of convergence							
16	1.97	2.01	1.86	2.01	1.93	1.90	
32	1.99	2.00	1.97	2.00	1.98	1.98	
64	2.00	2.00	1.99	2.00	2.00	2.00	
128	2.00	2.00	2.00	2.00	2.00	2.00	
256	2.00	2.00	2.00	2.00	2.00	2.01	
error - optimal CSC second step							
8	5.78e-03	4.91e-02	1.98e-02	5.34e-02	4.73e-01	2.70e+00	4
16	2.90e-03	2.81e-03	1.69e-03	6.03e-03	6.30e-02	6.94e-01	3
32	1.41e-04	1.82e-04	1.02e-04	6.13e-04	7.33e-03	1.73e-01	2
64	6.02e-06	1.18e-05	5.69e-06	6.95e-05	8.59e-04	4.30e-02	2
128	2.47e-07	7.56e-07	3.52e-07	8.43e-06	1.07e-04	1.10e-02	2
256	9.58e-09	5.11e-08	2.55e-08	1.05e-06	1.38e-05	2.71e-03	2
orders of convergence							
16	1.00	4.13	3.55	3.15	2.91	1.96	
32	4.36	3.95	4.05	3.30	3.10	2.00	
64	4.55	3.94	4.16	3.14	3.09	2.01	
128	4.61	3.97	4.02	3.04	3.00	1.97	
256	4.69	3.89	3.78	3.01	2.95	2.02	

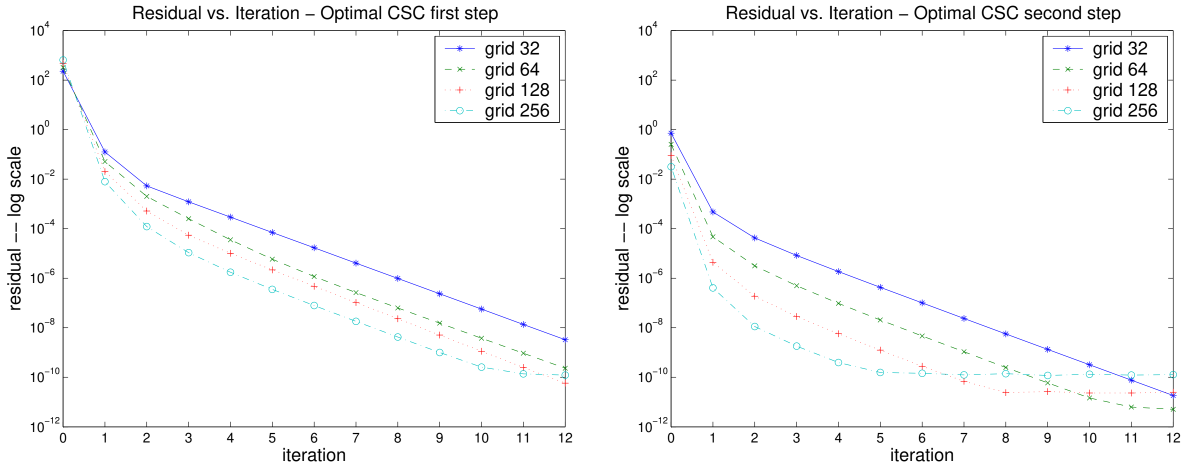
Table 5.1: Observed error and the corresponding orders of convergence of the full multigrid CSC method for Problem 1.

n	$\|u - u_\Delta\|_\infty$	$\|u' - u'_\Delta\|_{x_i,\tau_i,\infty}$			$\|u'' - u''_\Delta\|_{\lambda_i,\infty}$		no. of iter.
error - optimal CSC first step							
8	6.87e-02	4.51e-01	3.92e-01	4.51e-01	2.62e+00	3.87e+00	7
16	1.76e-02	1.12e-01	1.08e-01	1.12e-01	6.89e-01	1.04e+00	5
32	4.42e-03	2.79e-02	2.76e-02	2.79e-02	1.74e-01	2.63e-01	8
64	1.11e-03	6.96e-03	6.95e-03	6.96e-03	4.37e-02	6.60e-02	8
128	2.77e-04	1.74e-03	1.74e-03	1.74e-03	1.09e-02	1.65e-02	8
256	6.92e-05	4.37e-04	4.37e-04	4.37e-04	2.74e-03	4.11e-03	8
orders of convergence							
16	1.97	2.01	1.86	2.01	1.93	1.90	
32	1.99	2.00	1.97	2.00	1.98	1.98	
64	2.00	2.00	1.99	2.00	2.00	1.99	
128	2.00	2.00	2.00	2.00	2.00	2.00	
256	2.00	2.00	2.00	2.00	2.00	2.00	
error - optimal CSC second step							
8	5.78e-03	4.91e-02	1.98e-02	5.34e-02	4.73e-01	2.70e+00	5
16	2.90e-03	2.82e-03	1.69e-03	6.03e-03	6.30e-02	6.94e-01	3
32	1.41e-04	1.82e-04	1.02e-04	6.12e-04	7.33e-03	1.73e-01	5
64	5.88e-06	1.19e-05	5.63e-06	6.95e-05	8.60e-04	4.30e-02	5
128	2.92e-07	8.09e-07	3.99e-07	8.42e-06	1.12e-04	1.10e-02	5
256	2.58e-08	7.08e-08	4.37e-08	1.05e-06	1.66e-05	2.71e-03	5
orders of convergence							
16	1.00	4.12	3.55	3.15	2.91	1.96	
32	4.36	3.95	4.05	3.30	3.10	2.00	
64	4.58	3.94	4.18	3.14	3.09	2.01	
128	4.33	3.88	3.82	3.04	2.94	1.97	
256	3.50	3.51	3.19	3.01	2.76	2.02	

Table 5.2: Observed error and the corresponding orders of convergence of the V-cycle multigrid CSC method for Problem 1.

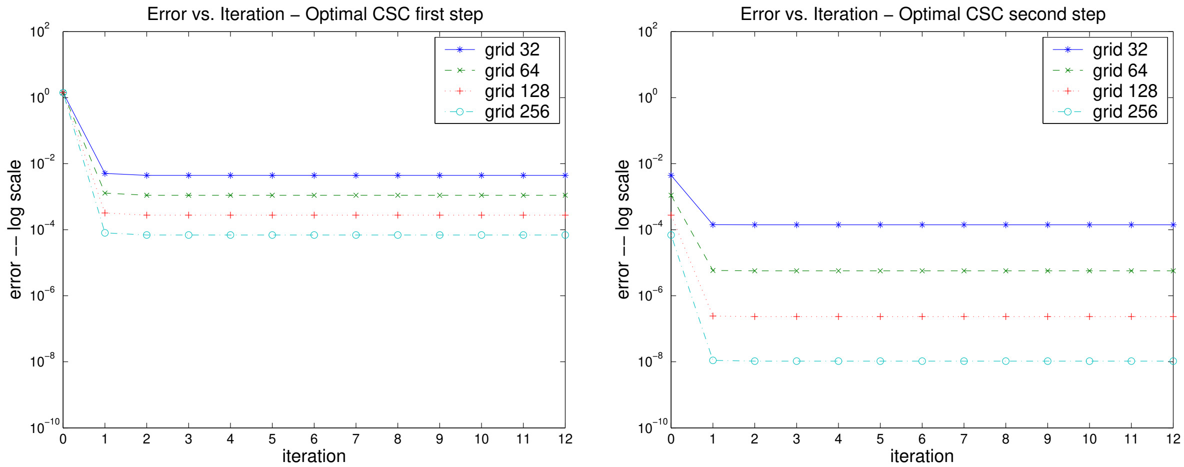
n	$\|u-u_\Delta\|_\infty$	$\|u'-u'_\Delta\|_{x_i,\tau_i,\infty}$			$\|u''-u''_\Delta\|_{\lambda_i,\infty}$		no. of iter.
error - optimal CSC first step							
8	6.87e-02	4.51e-01	3.92e-01	4.51e-01	2.62e+00	3.87e+00	7
16	1.76e-02	1.12e-01	1.08e-01	1.12e-01	6.89e-01	1.04e+00	5
32	4.42e-03	2.79e-02	2.76e-02	2.79e-02	1.74e-01	2.63e-01	4
64	1.11e-03	6.96e-03	6.95e-03	6.96e-03	4.37e-02	6.60e-02	3
128	2.77e-04	1.74e-03	1.74e-03	1.74e-03	1.10e-02	1.65e-02	2
256	6.93e-05	4.35e-04	4.35e-04	4.35e-04	2.74e-03	4.11e-03	2
orders of convergence							
16	1.97	2.01	1.86	2.01	1.93	1.90	
32	1.99	2.00	1.97	2.00	1.98	1.98	
64	2.00	2.00	1.99	2.00	2.00	1.99	
128	2.00	2.00	2.00	2.00	2.00	2.00	
256	2.00	2.00	2.00	2.00	2.00	2.01	
error - optimal CSC second step							
8	5.78e-03	4.91e-02	1.98e-02	5.34e-02	4.73e-01	2.70e+00	5
16	2.90e-03	2.82e-03	1.69e-03	6.03e-03	6.30e-02	6.94e-01	3
32	1.40e-04	1.82e-04	1.01e-04	6.12e-04	7.32e-03	1.73e-01	3
64	5.70e-06	1.18e-05	5.50e-06	6.94e-05	8.53e-04	4.30e-02	2
128	1.02e-07	7.68e-07	3.65e-07	8.44e-06	1.06e-04	1.09e-02	2
256	6.20e-09	4.80e-08	2.28e-08	1.05e-06	1.32e-05	2.71e-03	2
orders of convergence							
16	1.00	4.12	3.55	3.15	2.91	1.96	
32	4.37	3.95	4.06	3.30	3.11	2.00	
64	4.62	3.95	4.21	3.14	3.10	2.01	
128	5.81	3.94	3.91	3.04	3.01	1.97	
256	4.03	4.00	4.00	3.01	3.00	2.02	

Table 5.3: Observed error and the corresponding orders of convergence of the two-grid CSC method for Problem 1.



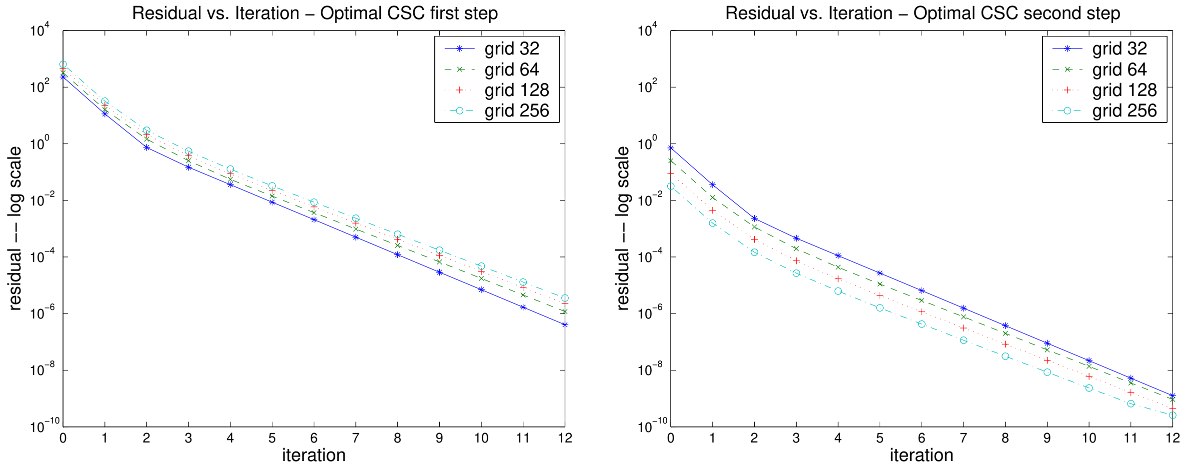
(a) first step (b) second step

Figure 5.1: Plots of the residual (in log scale) versus the iteration index of the full multigrid method applied to Problem 1 with grid sizes 32, 64, 128, 256, for the first step (a) and second step (b) of the optimal CSC method.



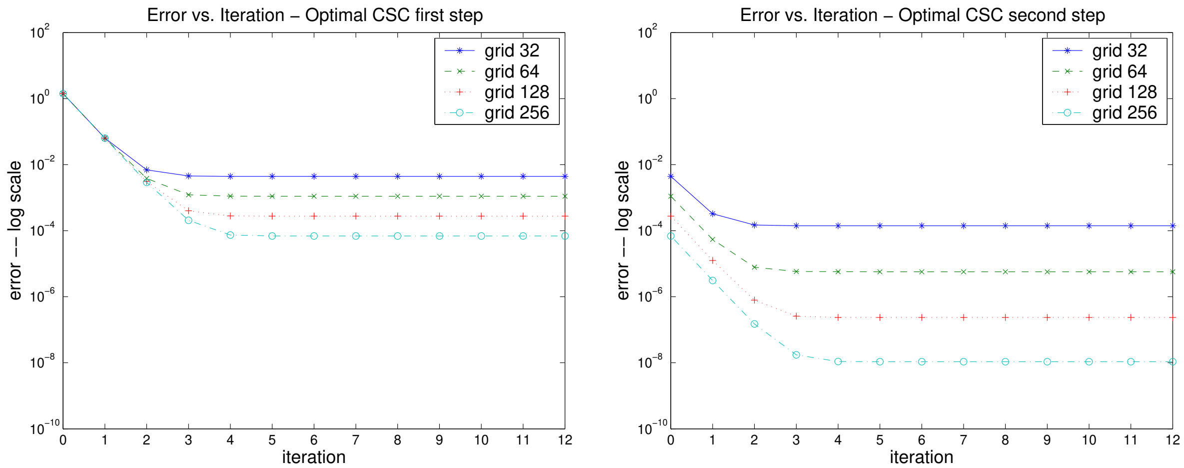
(a) first step (b) second step

Figure 5.2: Plots of the observed error (in log scale) versus the iteration index of the full multigrid method applied to Problem 1 with grid sizes 32, 64, 128, 256, for the first step (a) and second step (b) of the optimal CSC method.



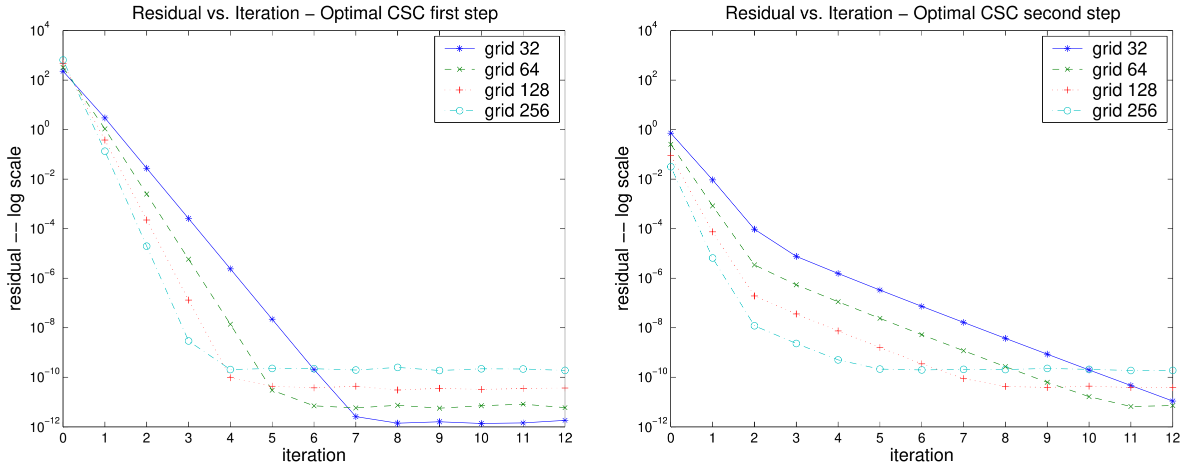
(a) first step (b) second step

Figure 5.3: Plots of the residual (in log scale) versus the iteration index of the V-cycle multigrid method applied to Problem 1 with grid sizes 32, 64, 128, 256, for the first step (a) and second step (b) of the optimal CSC method.



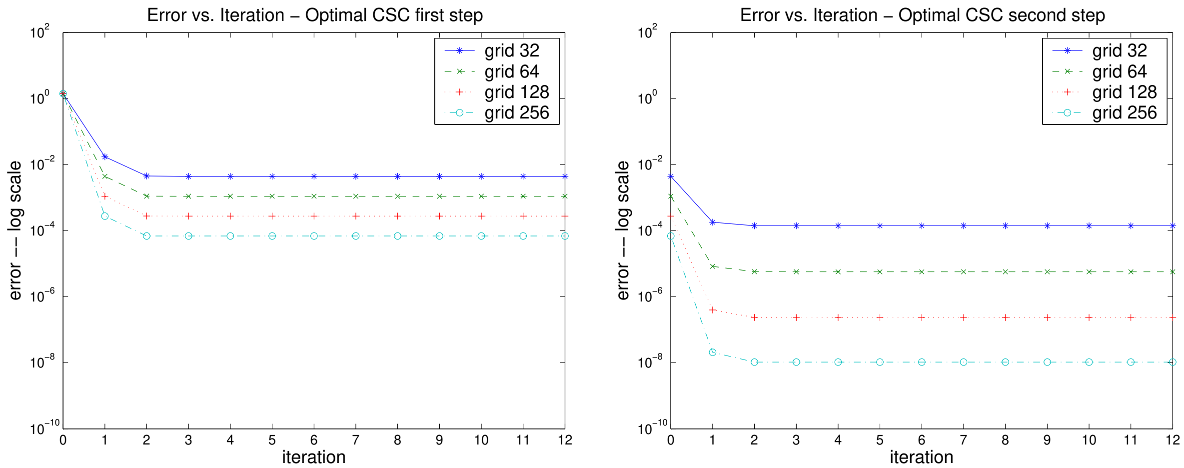
(a) first step (b) second step

Figure 5.4: Plots of the observed error (in log scale) versus the iteration index of the V-cycle multigrid method applied to Problem 1 with grid sizes 32, 64, 128, 256, for the first step (a) and second step (b) of the optimal CSC method.



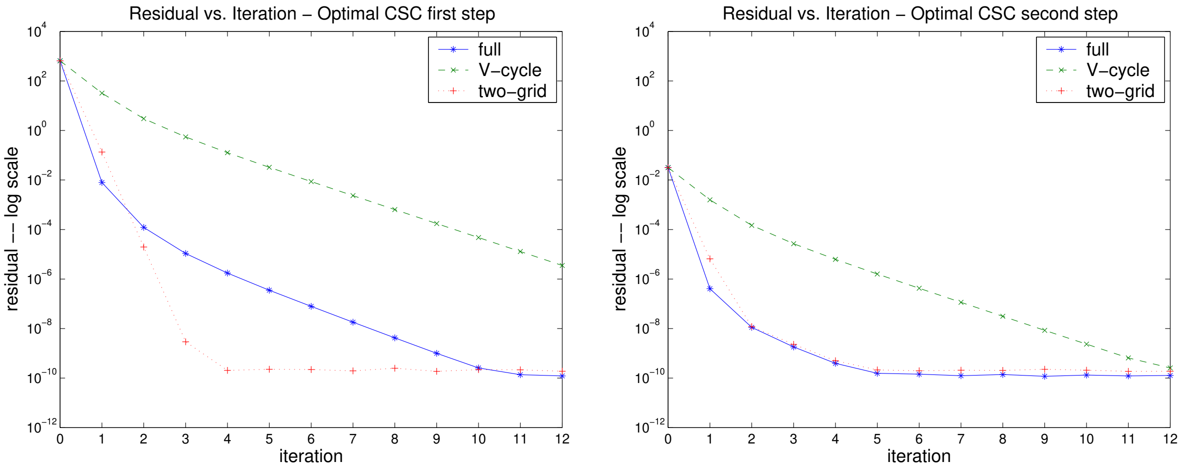
(a) first step (b) second step

Figure 5.5: Plots of the residual (in log scale) versus the iteration index of the two-grid multigrid method applied to Problem 1 with grid sizes 32, 64, 128, 256, for the first step (a) and second step (b) of the optimal CSC method.



(a) first step (b) second step

Figure 5.6: Plots of the observed error (in log scale) versus the iteration index of the two-grid multigrid method applied to Problem 1 with grid sizes 32, 64, 128, 256, for the first step (a) and second step (b) of the optimal CSC method.



(a) first step

(b) second step

Figure 5.7: Plots of the residual (in log scale) versus the iteration index of the full, V-cycle and two-grid multigrid methods applied to Problem 1 with grid size 256, for the first step (a) and second step (b) of the optimal CSC method.

Problem 2:

Determine $u = u(x)$ to satisfy the second order two-point BVP

$$\mathrm{L}u \equiv u_{xx} - u = g \quad \text{in } \Omega \equiv [0,1], \tag{5.2}$$

subject to homogeneous Dirichlet boundary conditions (2.5). The function g is chosen so that the exact solution to the problem is $u(x) = x^2(x-1)^2e^x$.

We apply the full multigrid, the V-cycle multigrid and the two-grid methods for CSC to this problem and show results similar to those obtained from Problem 1. The maximum absolute error, the respective observed orders of convergence, and the corresponding number of iterations are presented in Tables 5.4, 5.5 and 5.6. The results show second order convergence for the first step of CSC, and fourth order convergence for the second step.

In order to show how the grid size affects the convergence of the three multigrid methods, we plot the residual for 12 iterations of the full multigrid, the V-cycle multigrid and the two-grid methods, respectively, for 32, 64, 128 and 256 discretization grid points, in Figures 5.8, 5.10 and 5.12. Figures 5.9, 5.11 and 5.13 show the error for 12 iterations of the full multigrid, the V-cycle multigrid and the two-grid methods, respectively, for 32, 64, 128 and 256 discretization grid points. The experiments show that the full multigrid method reaches the discretization error in 1 iteration for both steps of CSC independently of grid size. The two-grid method is just a bit behind and needs 2 iterations independently of grid size. The V-cycle multigrid method requires 2, 3, 4, and 5 iterations for the first step, for 32, 64, 128 and 256 discretization grid points respectively, and 3, 4, 5, and 6 iterations for the first step, for 32, 64, 128 and 256 discretization grid points respectively. The results also show that the full multigrid converges faster for larger grids, especially at the initial few iterations. Both the two-grid method and the V-cycle multigrid converge with about the same speed for either large or small grids.

We also plot the residual for 12 iterations of the full multigrid, the V-cycle multigrid and the two-grid methods, respectively, for 256 discretization grid points in Figure 5.14, in order to compare their convergence. For the both steps of CSC, the multigrid method is the fastest in reducing the residual. The V-cycle multigrid method is the slowest.

n	$\|u - u_\Delta\|_\infty$	$\|u' - u'_\Delta\|_{x_i,\tau_i,\infty}$			$\|u'' - u''_\Delta\|_{\lambda_i,\infty}$		no. of iter.
error - optimal CSC first step							
8	7.41e-03	5.62e-02	4.06e-02	5.62e-02	2.53e-01	3.65e-01	9
16	1.83e-03	1.41e-02	1.20e-02	1.41e-02	6.86e-02	1.01e-01	7
32	4.56e-04	3.54e-03	3.26e-03	3.54e-03	1.79e-02	2.65e-02	6
64	1.14e-04	8.85e-04	8.50e-04	8.85e-04	4.56e-03	6.80e-03	5
128	2.85e-05	2.21e-04	2.17e-04	2.21e-04	1.15e-03	1.73e-03	4
256	7.11e-06	5.54e-05	5.49e-05	5.53e-05	3.03e-04	4.43e-04	3
orders of convergence							
16	2.02	1.99	1.76	1.99	1.88	1.86	
32	2.00	2.00	1.88	2.00	1.94	1.93	
64	2.00	2.00	1.94	2.00	1.97	1.96	
128	2.00	2.00	1.97	2.00	1.98	1.98	
256	2.00	2.00	1.98	2.00	1.93	1.96	
error - optimal CSC second step							
8	1.43e-04	1.42e-03	3.93e-04	3.60e-03	1.92e-02	2.61e-01	3
16	1.08e-05	1.06e-04	2.87e-05	4.57e-04	2.59e-03	6.59e-02	3
32	7.06e-07	7.35e-06	2.30e-06	5.69e-05	3.37e-04	1.59e-02	2
64	4.46e-08	4.82e-07	1.71e-07	6.95e-06	4.52e-05	4.17e-03	2
128	2.77e-09	3.63e-08	1.58e-08	8.56e-07	8.23e-06	1.07e-03	3
256	1.67e-10	1.62e-08	7.36e-09	1.17e-07	4.66e-06	2.67e-04	4
orders of convergence							
16	3.74	3.73	3.77	2.98	2.89	1.98	
32	3.93	3.86	3.64	3.01	2.94	2.05	
64	3.98	3.93	3.75	3.03	2.90	1.93	
128	4.01	3.73	3.44	3.02	2.46	1.97	
256	4.05	1.16	1.10	2.87	0.82	1.99	

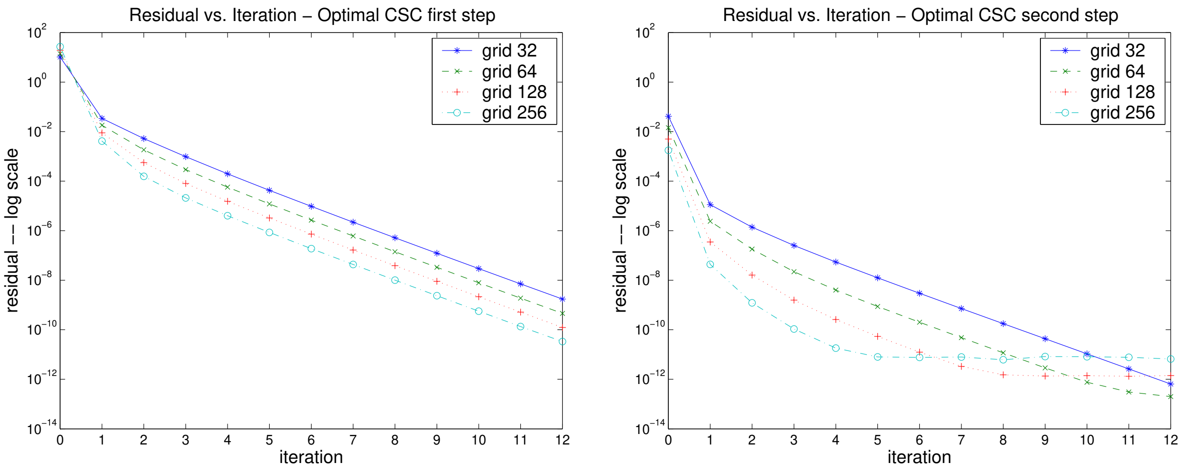
Table 5.4: Observed error and the corresponding orders of convergence of the full multigrid CSC method for Problem 2.

n	$\lVert u - u_\Delta \rVert_\infty$	$\lVert u' - u'_\Delta \rVert_{x_i,\tau_i,\infty}$			$\lVert u'' - u''_\Delta \rVert_{\lambda_i,\infty}$		no. of iter.
error - optimal CSC first step							
8	7.41e-03	5.62e-02	4.06e-02	5.62e-02	2.53e-01	3.65e-01	8
16	1.83e-03	1.41e-02	1.20e-02	1.41e-02	6.86e-02	1.01e-01	9
32	4.56e-04	3.54e-03	3.26e-03	3.54e-03	1.78e-02	2.65e-02	10
64	1.14e-04	8.85e-04	8.50e-04	8.84e-04	4.55e-03	6.79e-03	10
128	2.85e-05	2.21e-04	2.17e-04	2.21e-04	1.15e-03	1.72e-03	11
256	7.11e-06	5.52e-05	5.47e-05	5.51e-05	2.88e-04	4.30e-04	11
orders of convergence							
16	2.02	1.99	1.76	1.99	1.88	1.86	
32	2.00	2.00	1.88	2.00	1.94	1.93	
64	2.00	2.00	1.94	2.00	1.97	1.96	
128	2.00	2.00	1.97	2.00	1.99	1.98	
256	2.00	2.00	1.99	2.00	2.00	2.00	
error - optimal CSC second step							
8	1.43e-04	1.42e-03	3.93e-04	3.60e-03	1.92e-02	2.61e-01	6
16	1.08e-05	1.06e-04	2.87e-05	4.57e-04	2.59e-03	6.59e-02	6
32	7.06e-07	7.28e-06	2.24e-06	5.68e-05	3.37e-04	1.59e-02	6
64	4.47e-08	4.76e-07	1.56e-07	6.95e-06	4.27e-05	4.17e-03	7
128	2.80e-09	3.14e-08	9.37e-09	8.61e-07	5.79e-06	1.07e-03	7
256	1.76e-10	4.65e-09	1.52e-09	1.08e-07	1.64e-06	2.71e-04	7
orders of convergence							
16	3.74	3.73	3.77	2.98	2.89	1.98	
32	3.93	3.87	3.68	3.01	2.94	2.05	
64	3.98	3.94	3.84	3.03	2.98	1.93	
128	4.00	3.92	4.06	3.01	2.88	1.97	
256	3.99	2.76	2.62	3.00	1.82	1.98	

Table 5.5: Observed error and the corresponding orders of convergence of the V-cycle multigrid CSC method for Problem 2.

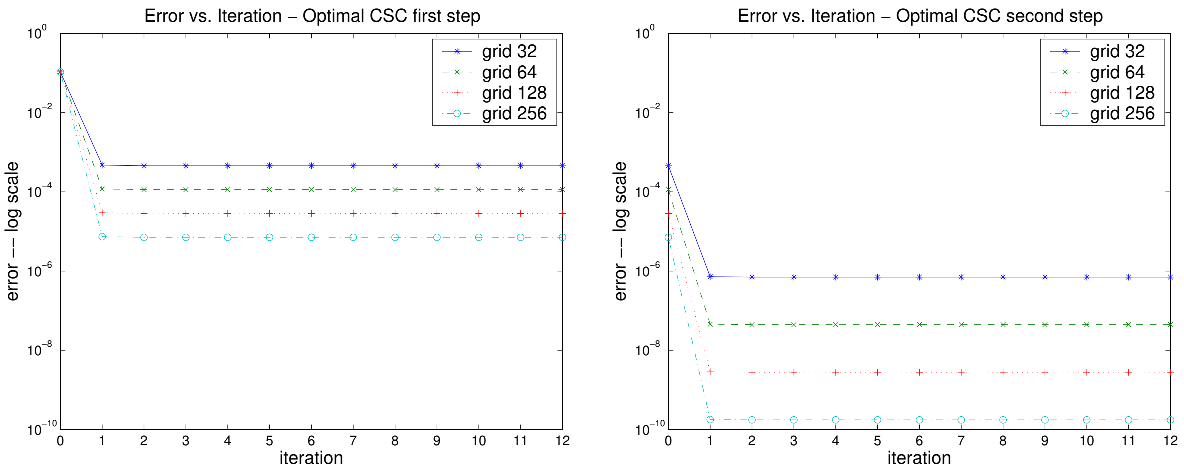
n	$\|u - u_\Delta\|_\infty$	$\|u' - u'_\Delta\|_{x_i,\tau_i,\infty}$			$\|u'' - u''_\Delta\|_{\lambda_i,\infty}$		no. of iter.
error - optimal CSC first step							
8	7.41e-03	5.62e-02	4.06e-02	5.62e-02	2.53e-01	3.65e-01	8
16	1.83e-03	1.41e-02	1.20e-02	1.41e-02	6.86e-02	1.01e-01	9
32	4.56e-04	3.54e-03	3.26e-03	3.54e-03	1.78e-02	2.65e-02	9
64	1.14e-04	8.85e-04	8.50e-04	8.84e-04	4.55e-03	6.79e-03	9
128	2.85e-05	2.21e-04	2.17e-04	2.21e-04	1.15e-03	1.71e-03	8
256	7.11e-06	5.53e-05	5.48e-05	5.51e-05	2.88e-04	4.27e-04	8
orders of convergence							
16	2.02	1.99	1.76	1.99	1.88	1.86	
32	2.00	2.00	1.88	2.00	1.94	1.93	
64	2.00	2.00	1.94	2.00	1.97	1.96	
128	2.00	2.00	1.97	2.00	1.99	1.99	
256	2.00	2.00	1.99	2.00	2.00	2.01	
error - optimal CSC second step							
8	1.43e-04	1.42e-03	3.93e-04	3.60e-03	1.92e-02	2.61e-01	6
16	1.08e-05	1.06e-04	2.87e-05	4.57e-04	2.59e-03	6.59e-02	6
32	7.06e-07	7.32e-06	2.28e-06	5.69e-05	3.37e-04	1.59e-02	5
64	4.47e-08	4.81e-07	1.60e-07	6.95e-06	4.27e-05	4.17e-03	5
128	2.83e-09	5.20e-08	1.92e-08	8.66e-07	8.59e-06	1.07e-03	5
256	1.82e-10	1.34e-08	5.36e-09	1.10e-07	4.24e-06	2.72e-04	5
orders of convergence							
16	3.74	3.73	3.77	2.98	2.89	1.98	
32	3.93	3.86	3.65	3.01	2.94	2.05	
64	3.98	3.93	3.84	3.03	2.98	1.93	
128	3.98	3.21	3.06	3.01	2.31	1.96	
256	3.96	1.95	1.84	2.98	1.02	1.98	

Table 5.6: Observed error and the corresponding orders of convergence of the two-grid multi-grid CSC method for Problem 2.



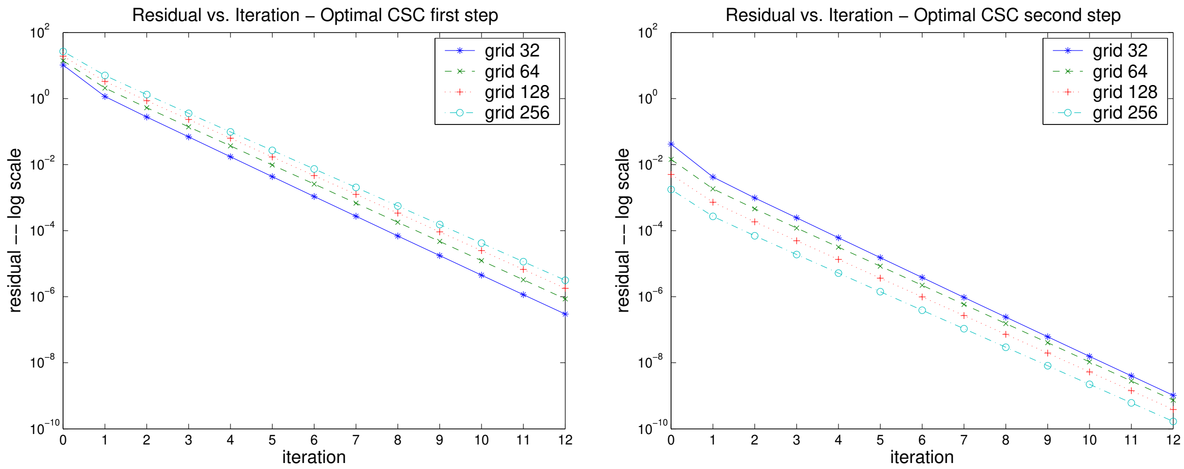
(a) first step (b) second step

Figure 5.8: Plots of the residual (in log scale) versus the iteration index of the full multigrid method applied to Problem 2 with grid sizes 32, 64, 128, 256, for the first step (a) and second step (b) of the optimal CSC method.



(a) first step (b) second step

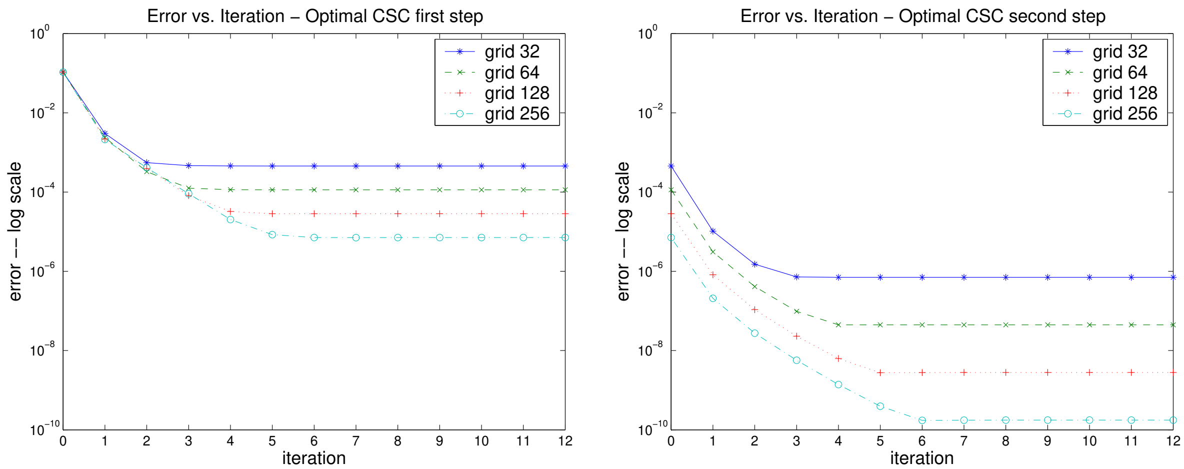
Figure 5.9: Plots of the observed error (in log scale) versus the iteration index of the full multigrid method applied to Problem 2 with grid sizes 32, 64, 128, 256, for the first step (a) and second step (b) of the optimal CSC method.



(a) first step

(b) second step

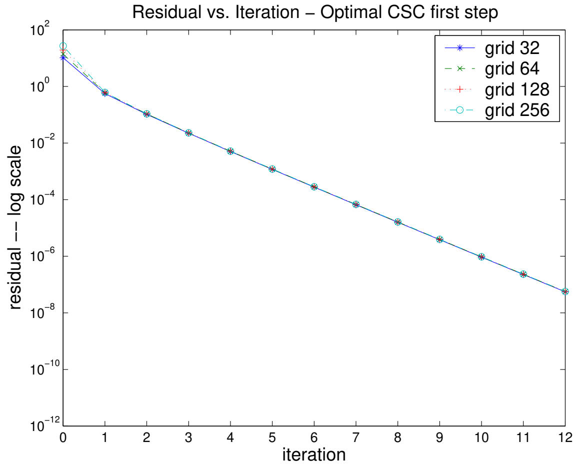
Figure 5.10: Plots of the residual (in log scale) versus the iteration index of the V-cycle multigrid method applied to Problem 2 with grid sizes 32, 64, 128, 256, for the first step (a) and second step (b) of the optimal CSC method.



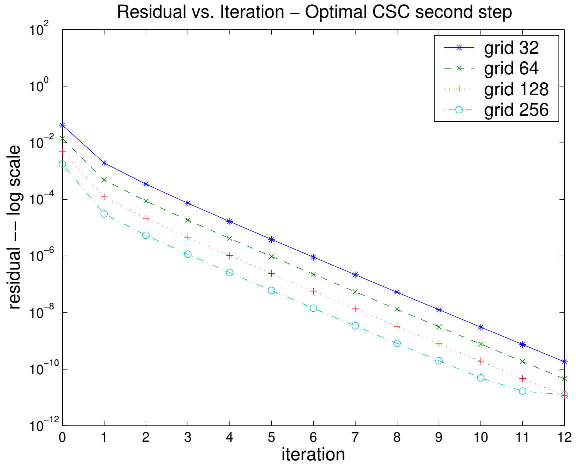
(a) first step

(b) second step

Figure 5.11: Plots of the observed error (in log scale) versus the iteration index of the V-cycle multigrid method applied to Problem 2 with grid sizes 32, 64, 128, 256, for the first step (a) and second step (b) of the optimal CSC method.

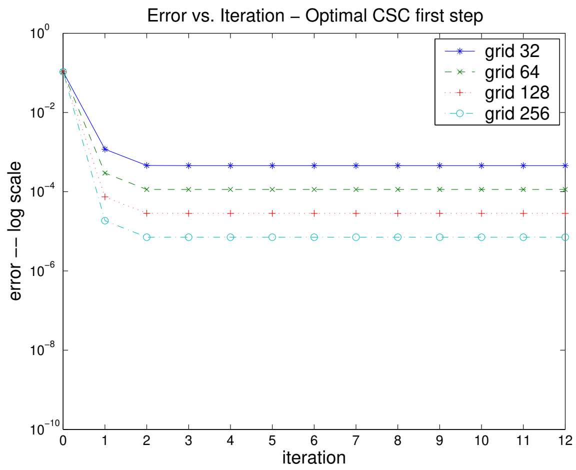


(a) first step

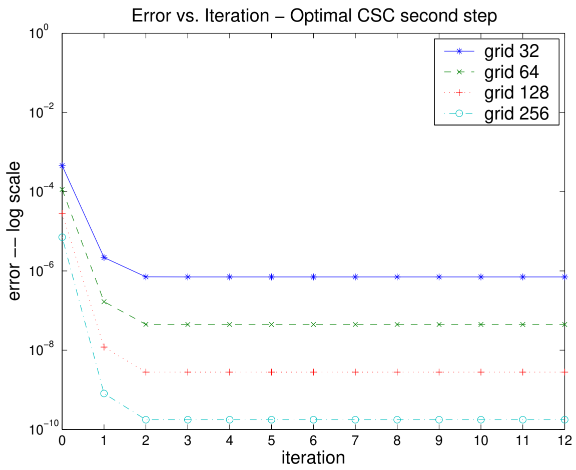


(b) second step

Figure 5.12: Plots of the residual (in log scale) versus the iteration index of the two-grid multigrid method applied to Problem 2 with grid sizes 32, 64, 128, 256, for the first step (a) and second step (b) of the optimal CSC method.

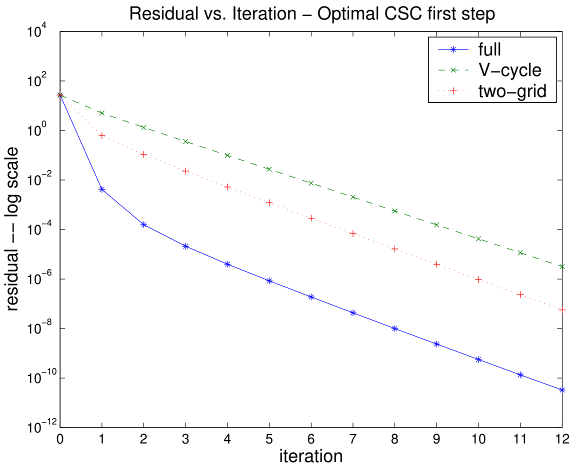


(a) first step

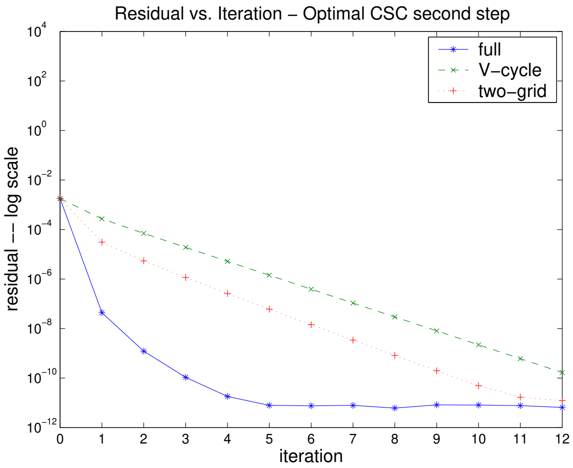


(b) second step

Figure 5.13: Plots of the observed error (in log scale) versus the iteration index of the two-grid multigrid method applied to Problem 2 with grid sizes 32, 64, 128, 256, for the first step (a) and second step (b) of the optimal CSC method.



(a) first step



(b) second step

Figure 5.14: Plots of the residual (in log scale) versus the iteration index of the full, V-cycle and two-grid multigrid methods applied to Problem 2 with grid size 256, for the first step (a) and second step (b) of the optimal CSC method.

Problem 3

Consider solving the SWEs (4.1) and (4.2) with periodic boundary conditions, $x \in [0, L]$ and zero right hand side, i.e.,

$$\frac{du}{dt} + \phi_x = 0, \tag{5.3}$$

$$\frac{d\phi}{dt} + \phi^* \cdot u_x = 0. \tag{5.4}$$

We integrate the above equations with initial conditions

$$u(x,0) = \frac{1}{3} e^{-a(\frac{x-L/2}{L})^2}, \tag{5.5}$$

$$\phi(x,0) = \frac{95}{3} e^{-a(\frac{x-L/2}{L})^2}, \tag{5.6}$$

where $L = 1.05 \times 10^7$ m, $a = 30$, and $\phi^* = 9184$ m^2/s^2.

This is a one-dimensional version of the nonlinear equatorial Kelvin wave, investigated by [2] and [20]. The above initial conditions generate gravity waves traveling toward the right, and wrapping around when reaching the boundary. The shape of the wave remains the same during the movement.

We apply the full multigrid method for CSC to this problem. We use the semi-Lagrangian semi-implicit integration scheme in time and DNE approach in space with the standard CSC method, and apply the full multigrid method to solving the resulting CSC linear system at each time step.

Simulations were run with time step-size 50 seconds on 128 discretization grid points. We chose tolerance $\varepsilon = 10^{-6}$ and set 2 as the maximum number of iterations for the full multigrid method. Snapshots of the computed approximation to the solution at various time steps are shown in Figures 5.15 and 5.16. In each snapshot, we also plot the solution at the initial state, in order to show how the wave moves to the right and retains its shape. The average number of iterations that the full multigrid method took was 2. The results agree very well with the theory. The shape of the traveling wave is preserved without dispersion or loss of magnitude. See Figure 5.17 for the computed solution after 5 or 10 periods. The results also confirm that our full multigrid method for CSC is working properly and efficiently.

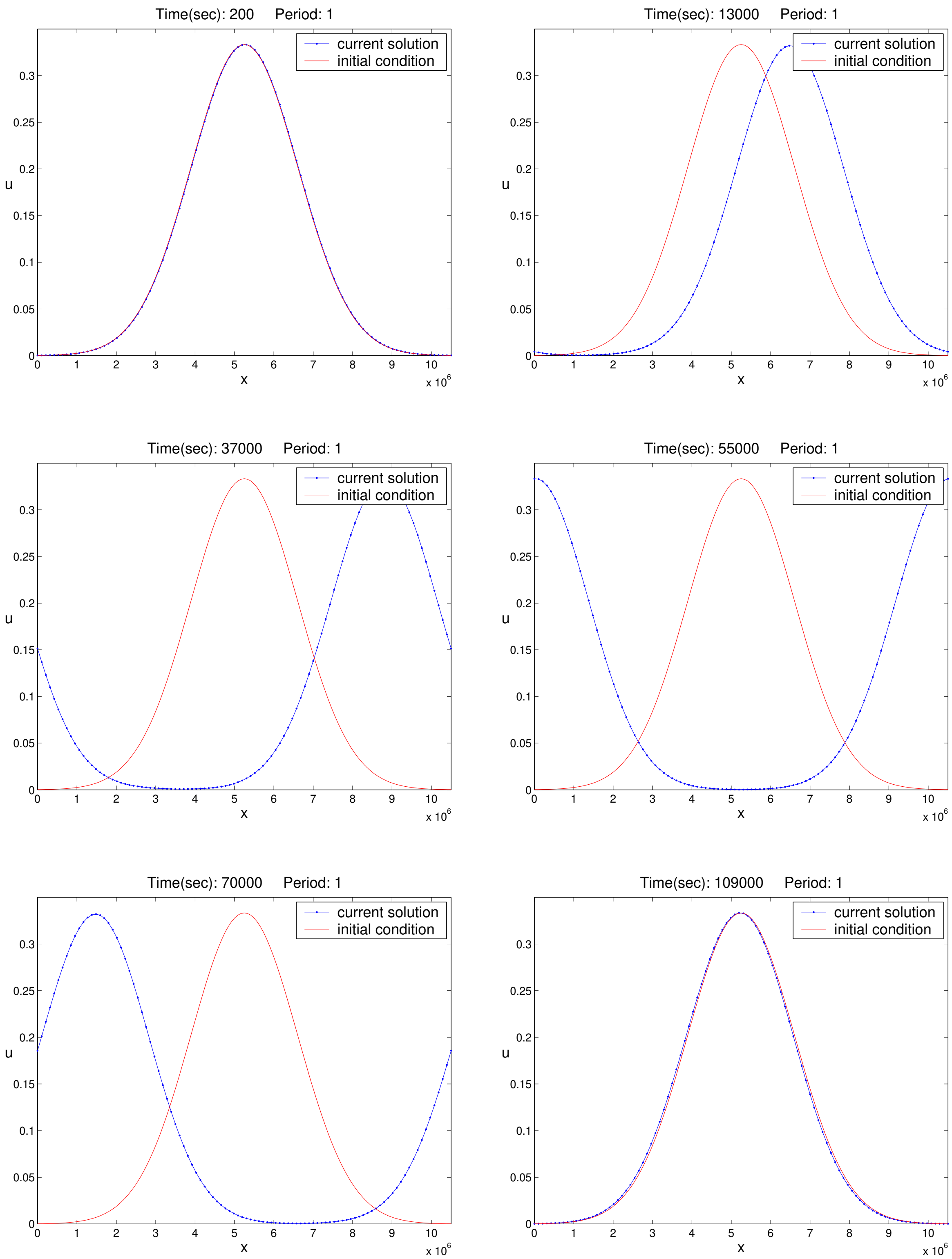


Figure 5.15: Snapshots of the computed approximation to the solution u of Problem 3 integrated with time step-size 50 seconds on 128 discretization grid points. The full multigrid method is used to solve the resulting CSC linear system at each time step.

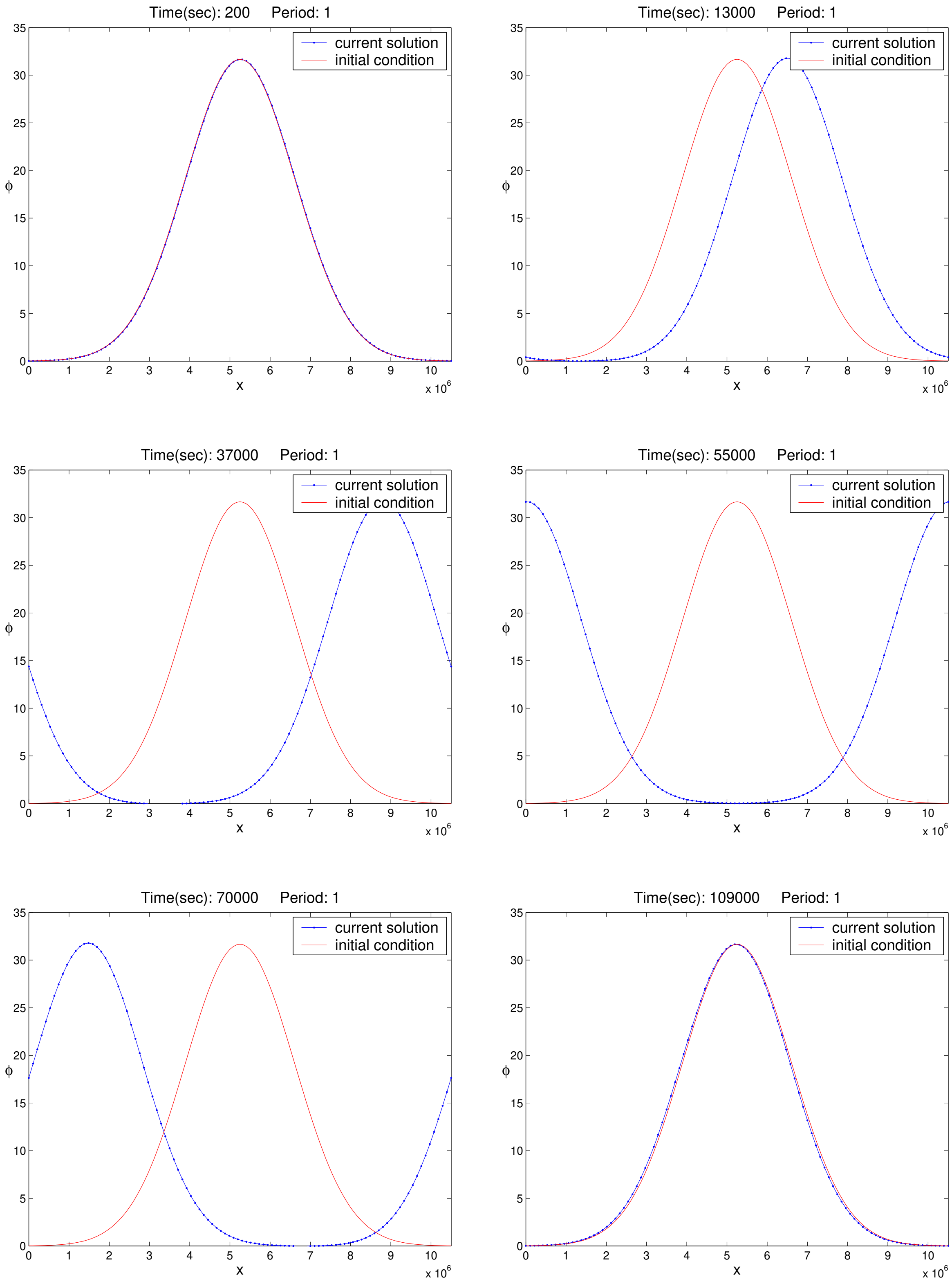
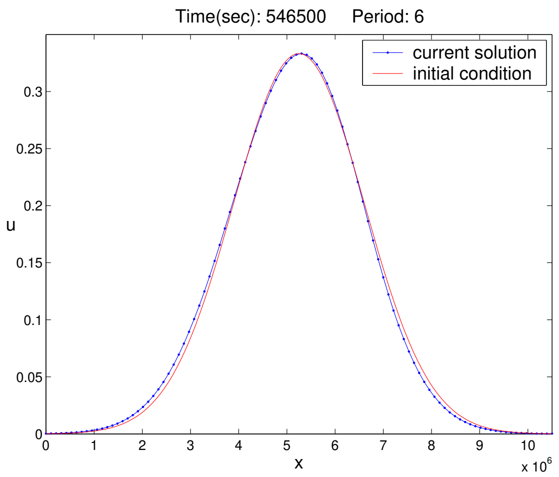
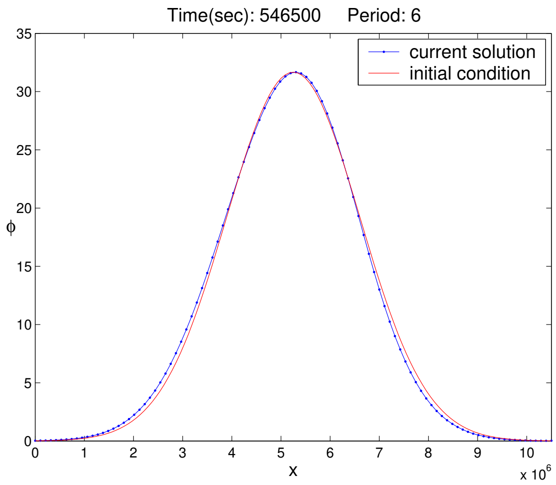


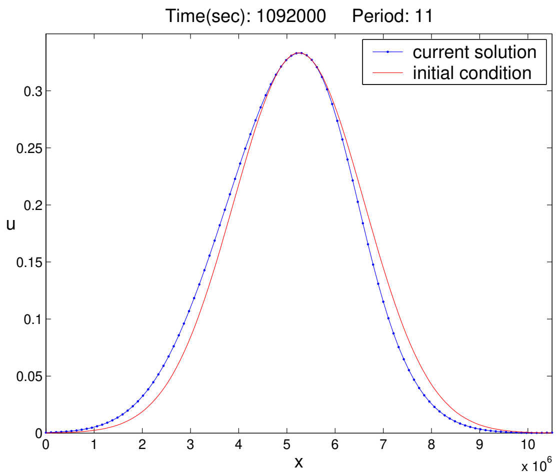
Figure 5.16: Snapshots of the computed approximation to the solution ϕ of Problem 3 integrated with time step-size 50 seconds on 128 discretization grid points. The full multigrid method is used to solve the resulting CSC linear system at each time step.



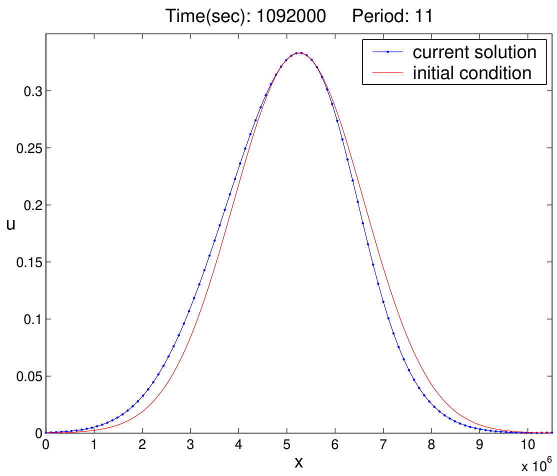
(a) Approximation to u after 5 periods



(b) Approximation to ϕ after 5 periods



(c) Approximation to u after 10 periods



(d) Approximation to ϕ after 10 periods

Figure 5.17: Snapshots of the computed approximation to the solutions of Problem 3, after 5 and 10 periods, integrated with time step-size 50 seconds on 128 discretization grid points. The full multigrid method is used to solve the resulting CSC linear system at each time step.

Problem 4

Consider solving the same system as in Problem 3, i.e., SWEs (4.1) and (4.2) with zero right hand side and periodic boundary conditions. However, we choose a new set of initial conditions

$$
\begin{aligned}
u(x,0) &= 0, && (5.7)\\
\phi(x,0) &= \frac{5g}{3}e^{-a(\frac{x-L/2}{L})^2}, && (5.8)
\end{aligned}
$$

where $L = 1.05 \times 10^7$ m, $a = 30$, $g = 9.80616$ m^2/s and $\phi^* = 50g$ m^2/s^2.

We apply the full multigrid method for CSC to this problem. We use the semi-Lagrangian semi-implicit integration scheme in time and DNE approach in space with the standard CSC method, and apply the full multigrid method to solving the resulting CSC linear system at each time step.

Simulations were run with time step-size 100 seconds on 32 discretization grid points. We chose tolerance $\varepsilon = 10^{-6}$ and set 2 as the maximum number of iterations for full multigrid method. These initial conditions generate periodic oscillating waves without any damping [20], i.e., the wave goes up and down forever. Snapshots of the computed approximation to the solution at various time step are shown in Figures 5.18 and 5.19, which agree well with the theory. No dispersion or loss of magnitude is observed. The average number of iterations that the full multigrid method took was 2. The results again confirm that our full multigrid method for CSC is working properly and efficiently.

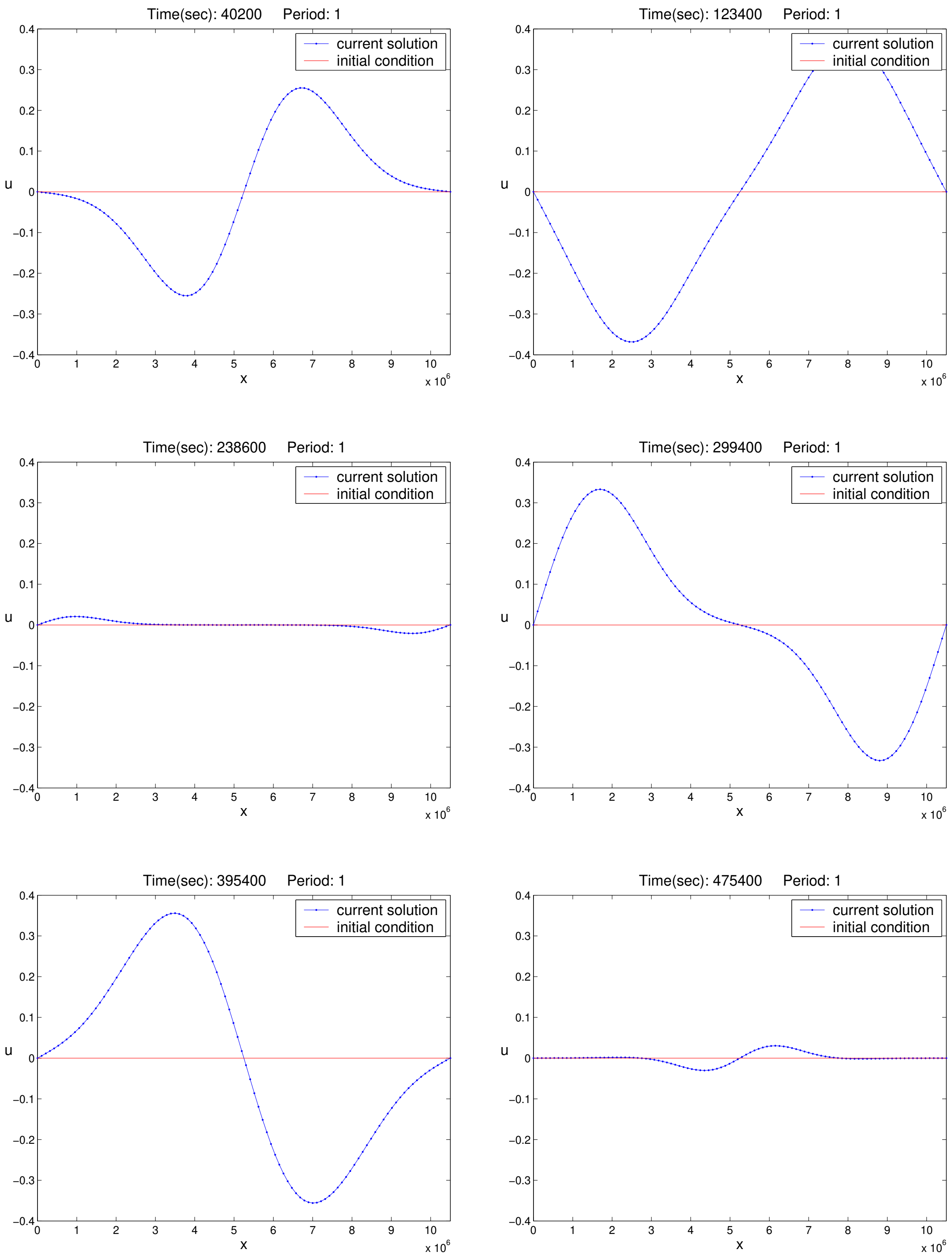


Figure 5.18: Snapshots of the computed approximation to the solution u of Problem 4 integrated with time step-size 100 seconds on 32 discretization grid points. The full multigrid method is used to solve the resulting CSC linear system at each time step.

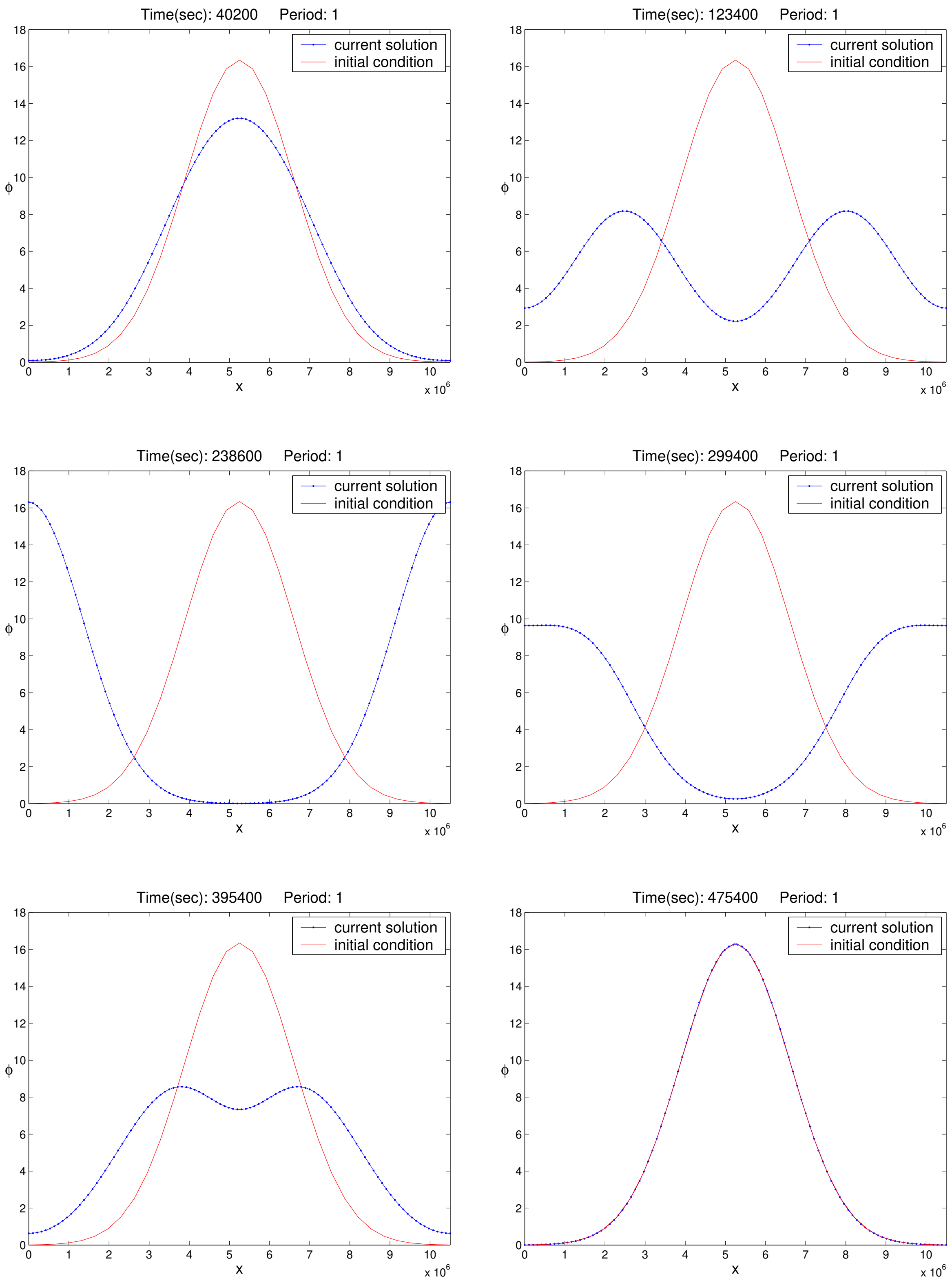


Figure 5.19: Snapshots of the computed approximation to the solution ϕ of Problem 4 integrated with time step-size 100 seconds on 32 discretization grid points. The full multigrid method is used to solve the resulting CSC linear system at each time step.

Problem 5

Consider solving the one-dimensional SWEs (4.1) and (4.2) with periodic boundary conditions and $x \in [0, L]$. The right hand side is chosen so that the exact solution to the problem is

$$u(x,t) = \frac{U_0}{R}(1 - e^{-t/T_0})e^{-a_3(\frac{x-L/2-d}{L})^2}, \tag{5.9}$$

$$\phi(x,t) = 50ge^{-a_1(\frac{x-L/2}{L})^2} - \log{(1 + t/T_0)}e^{-a_2(\frac{x-L/2}{L})^2}. \tag{5.10}$$

The solution to this problem at the early stage resembles the solution to the SWEs with zero right hand side and initial conditions

$$u(x,0) = 0, \tag{5.11}$$

$$\phi(x,0) = 50ge^{-a_1(\frac{x-L/2}{L})^2}, \tag{5.12}$$

as shown in [20].

In our experiments, we choose $U_0 = 1.5 \times 10^6$ m, $R = 6378137$ m, $L = 1.05 \times 10^7$ m, $T_0 = 5000$ s, $a_3 = 150$, $d = L/20$, $g = 9.80616$ m^2/s, $a_1 = 70$, $a_2 = 420$ and $\phi^* = 50g$ m^2/s^2.

We apply the full multigrid method for CSC to this problem. We use the semi-Lagrangian semi-implicit integration scheme in time and DNE approach in space with the standard CSC method, and apply the full multigrid method to solving the resulting CSC linear system at each time step. Two relaxation iterations are applied at each level.

Simulations were carried out with time step-size 50 seconds on 32, 64, 128 and 256 discretization grid points. We chose tolerance $\varepsilon = 10^{-6}$ and $m = 10$ as the maximum number of iterations for the full multigrid method. The full multigrid method took an average of 8, $6 – 7$, $4 – 5$, and 3 iterations, respectively, for grid size 32, 64, 128, and 256 to reach the tolerance.

Snapshots of the true solution of the system and the computed approximation to it are shown for 128 grid points in Figures 5.20 and 5.21. The absolute error on the nodal points and the corresponding orders of convergence are shown in Table 5.7. The results indicate that the computed approximation agrees very well with the true solution and our method achieves average fourth order convergence.

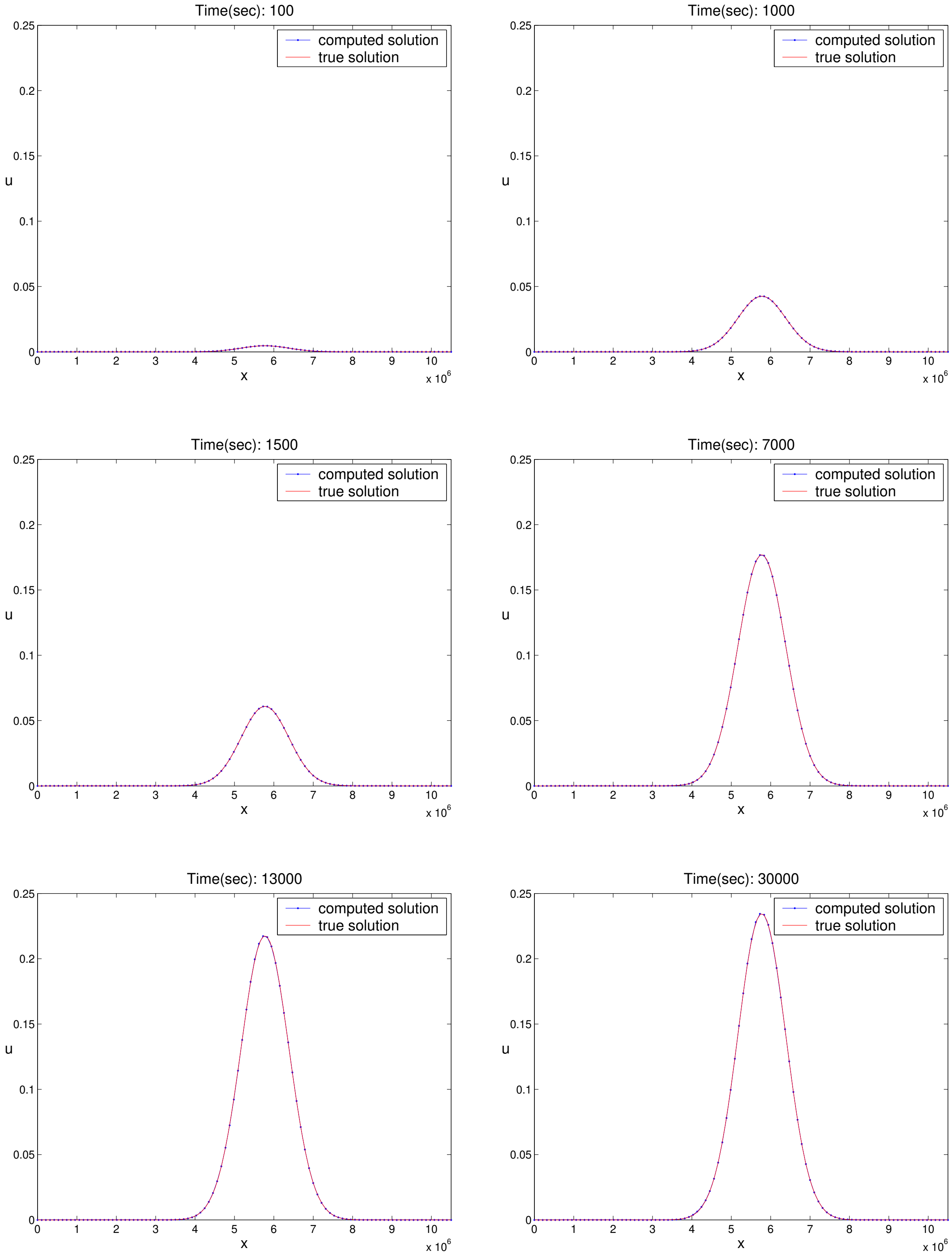


Figure 5.20: Snapshots of the computed approximation to the solution u of Problem 5 integrated with time step-size 50 seconds on 128 discretization grid points. The full multigrid method is used to solve the resulting CSC linear system at each time step.

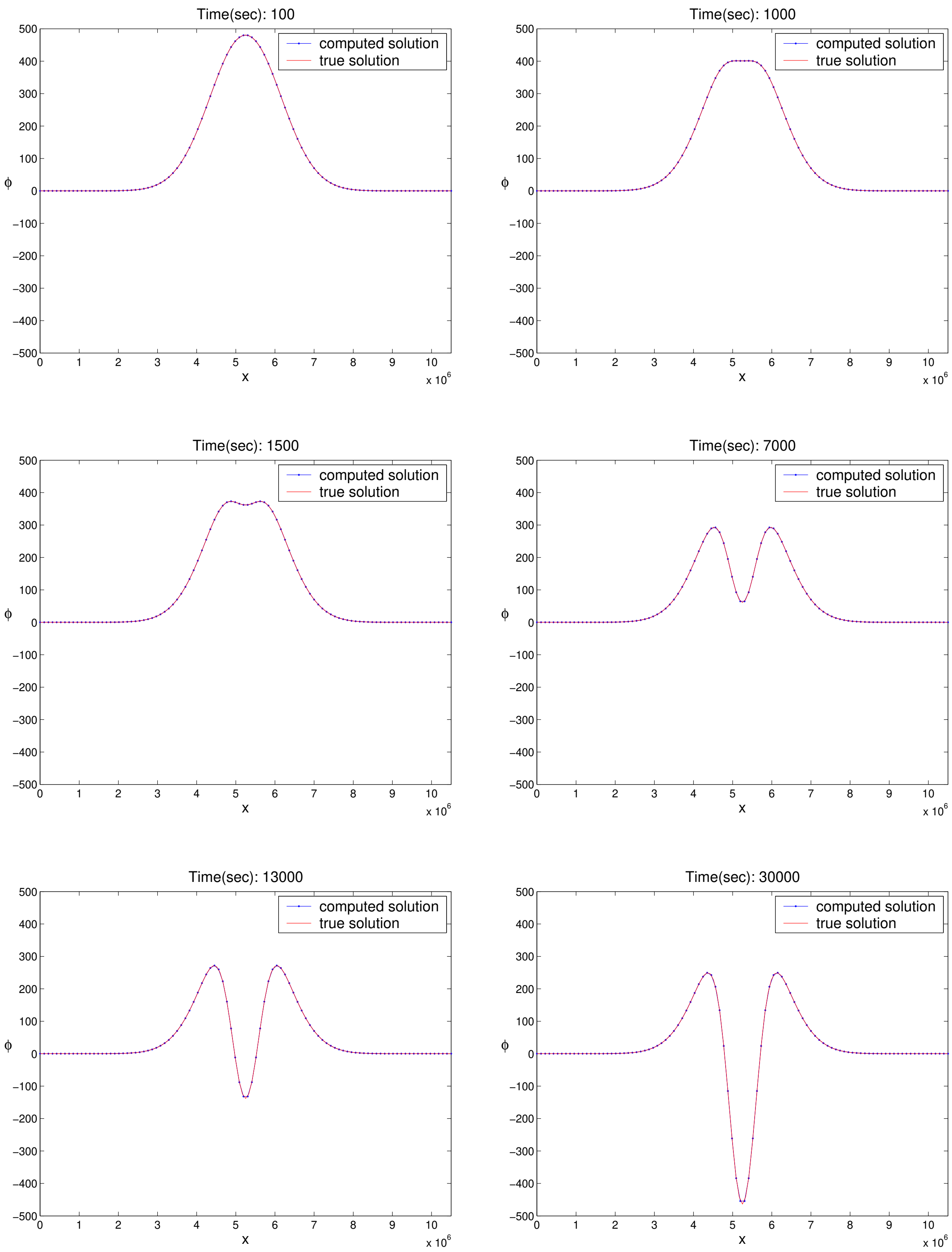


Figure 5.21: Snapshots of the computed approximation to the solution ϕ of Problem 3 integrated with time step-size 50 seconds on 128 discretization grid points. The full multigrid method is used to solve the resulting CSC linear system at each time step.

	$n = 32$	$n = 64$		$n = 128$		$n = 256$	
$time$	$\|\|u - u_\Delta\|\|_{x_i}$	$\|\|u - u_\Delta\|\|_{x_i}$	Convg.	$\|\|u - u_\Delta\|\|_{x_i}$	Convg.	$\|\|u - u_\Delta\|\|_{x_i}$	Convg.
100	3.44e-05	1.42e-06	4.60	5.13e-07	1.47	5.02e-07	0.03
3450	3.26e-02	1.88e-03	4.12	1.32e-04	3.83	2.98e-05	2.15
6800	1.08e-01	5.98e-03	4.18	3.98e-04	3.91	5.78e-05	2.78
10150	2.04e-01	1.07e-02	4.26	7.13e-04	3.91	8.62e-05	3.05
13500	3.02e-01	1.47e-02	4.36	1.00e-03	3.87	1.11e-04	3.17
16850	3.85e-01	1.71e-02	4.50	1.21e-03	3.82	1.31e-04	3.21
20200	4.40e-01	1.74e-02	4.66	1.31e-03	3.73	1.41e-04	3.21
23550	4.60e-01	1.56e-02	4.88	1.28e-03	3.61	1.42e-04	3.17
26900	4.41e-01	1.28e-02	5.11	1.16e-03	3.46	1.34e-04	3.11
30000	3.89e-01	9.96e-03	5.29	9.90e-04	3.33	1.22e-04	3.03
	$n = 32$	$n = 64$		$n = 128$		$n = 256$	
$time$	$\|\|\phi - \phi_\Delta\|\|_{x_i}$	$\|\|\phi - \phi_\Delta\|\|_{x_i}$	Convg.	$\|\|\phi - \phi_\Delta\|\|_{x_i}$	Convg.	$\|\|\phi - \phi_\Delta\|\|_{x_i}$	Convg.
100	3.77e-05	3.90e-05	-.05	3.91e-05	-.01	3.91e-05	-.00
3450	1.04e-01	5.95e-03	4.13	1.10e-03	2.43	1.18e-03	-.11
6800	7.24e-01	4.77e-02	3.93	2.73e-03	4.13	1.25e-03	1.13
10150	2.16e+00	1.41e-01	3.93	6.03e-03	4.55	1.17e-03	2.36
13500	4.51e+00	2.86e-01	3.98	1.31e-02	4.45	1.41e-03	3.22
16850	7.73e+00	4.69e-01	4.04	2.31e-02	4.35	1.67e-03	3.79
20200	1.16e+01	6.66e-01	4.13	3.43e-02	4.28	1.92e-03	4.16
23550	1.60e+01	8.52e-01	4.23	4.53e-02	4.23	2.68e-03	4.08
26900	2.04e+01	1.00e+00	4.34	5.47e-02	4.20	3.62e-03	3.92
30000	2.42e+01	1.10e+00	4.46	6.12e-02	4.17	4.34e-03	3.82

Table 5.7: Observed absolute error and the corresponding orders of convergence of u and ϕ for Problem 5.

Problem 6

Consider solving the one-dimensional SWEs (4.1) and (4.2), with periodic boundary conditions and $x \in [0, L]$.

First, we consider running the experiments with f_u and f_ϕ formed so that the solution to the system is

$$u(x,t) = U_0 \sin(2\pi r x L)\cos(2\pi r c t L), \tag{5.13}$$
$$\phi(x,t) = \Phi^*(1 - \frac{U_0}{c}\cos(2\pi r x L)\sin(2\pi r c t L)). \tag{5.14}$$

As shown in [22], this solution is also the exact solution to the linearized one-dimensional Shallow Water Equations with zero right hand side, i.e.,

$$\frac{\partial u}{\partial t} + \phi_x = 0, \tag{5.15}$$
$$\frac{\partial \phi}{\partial t} + \phi^* \cdot u_x = 0. \tag{5.16}$$

The parameters are chosen as following: The amplitude $U_0 = 27.3$m, $L = 1.05 \times 10^7$ m, $\phi^* = \Phi^* = 91840$ m^2/s^2, $c = \sqrt{\Phi^*}$, $r = 2$.

Simulations were carried out with time step-size 10 seconds on 16, 32, 64 and 128 discretization grid points. The snapshots of the solution to the system with the nonzero right hand side on 128 grid points are shown in Figures 5.22 and 5.23. Since the true solution to the system, equations (5.13) and (5.14), are periodic functions with period $\frac{L}{rc} \approx 1.7324 * 10^4$ seconds, we integrated the system for 25 periods, which corresponds to about $25 \times 1.7324 * 10^4 = 4.331 * 10^5$ s ≈ 5 days. The snapshots show that the shape of the solution is well preserved even after 25 periods. Table 5.8 shows the relative error on the nodal points at the beginning of each period and the corresponding orders of convergence.

For comparison, we carried out the same experiments but using the Finite Difference (FD) as the discretization method at each time step, and the LU decomposition with backward and forward substitution, which we call the direct method, to solve the resulting linear system from FD. The relative error on the nodal points at the beginning of each period and the corresponding orders of convergence are shown in Table 5.9.

The results show that our multigrid method for CSC achieves much more accuracy and higher order of convergence.

Secondly, we consider integrating the same system but with zero right hand side, i.e., $f_u = 0$ and $f_\phi = 0$. Assume that the initial conditions are given by

$$u(x,0) = U_0 \sin(2\pi r x L), \tag{5.17}$$

$$\phi(x,0) = \Phi^*. \tag{5.18}$$

The parameters are the same as above. We still use the semi-Lagrangian semi-implicit integration scheme in time and the DNE approach in space. But in order to solve the resulting CSC linear system, we consider using two different methods. One is the full multigrid method, the other is the direct method. For comparison, we also consider the same experiments but with FD as the discretization method and the direct method for the solution of the resulting linear system.

Simulations were carried out with time step-size 10 seconds on 32 or 64 discretization grid points.

Snapshots of the approximation to the solution computed by the multigrid method for CSC on 32 grid points are shown in Figures 5.24 and 5.25 and those on 64 grid points are shown in Figures 5.26 and 5.27. The results show that our approach is stable in the sense that the magnitude of both u and ϕ remain the same. There is no blow-up or great loss in magnitude. However, we also observed that the computed approximations have oscillations. The oscillations increase as time increases. The oscillations also increase as we increase the number of discretization grid points. Actually, the number of bumps in the computed approximation nearly doubles as the discretization grid points doubled.

Through comparing the results produced by the two linear system solution methods for CSC, i.e., the multigrid and direct methods, we can see that the multigrid method doesn't bring any oscillation into the solution. Snapshots of the computed approximation to the solution by the direct method are shown in Figure 5.28 and 5.29.

Since we don't know the true solution to this problem, we can't exclude the possibility that the oscillations we observed are actually inherently part of the solution. If this is not the case, we would argue that our results look oscillatory because of the high order discretization method we use. To support our arguments, we could take a look at the results from the finite difference method, which is a lower order discretization method than CSC, as shown in Figures 5.30 and 5.31 for 32 grid points and Figures 5.32 and 5.33 for 64 grid points. Though the oscillations still increase as the number of grid points increases, the finite difference method produces much less oscillatory results. Of course, they are also less accurate.

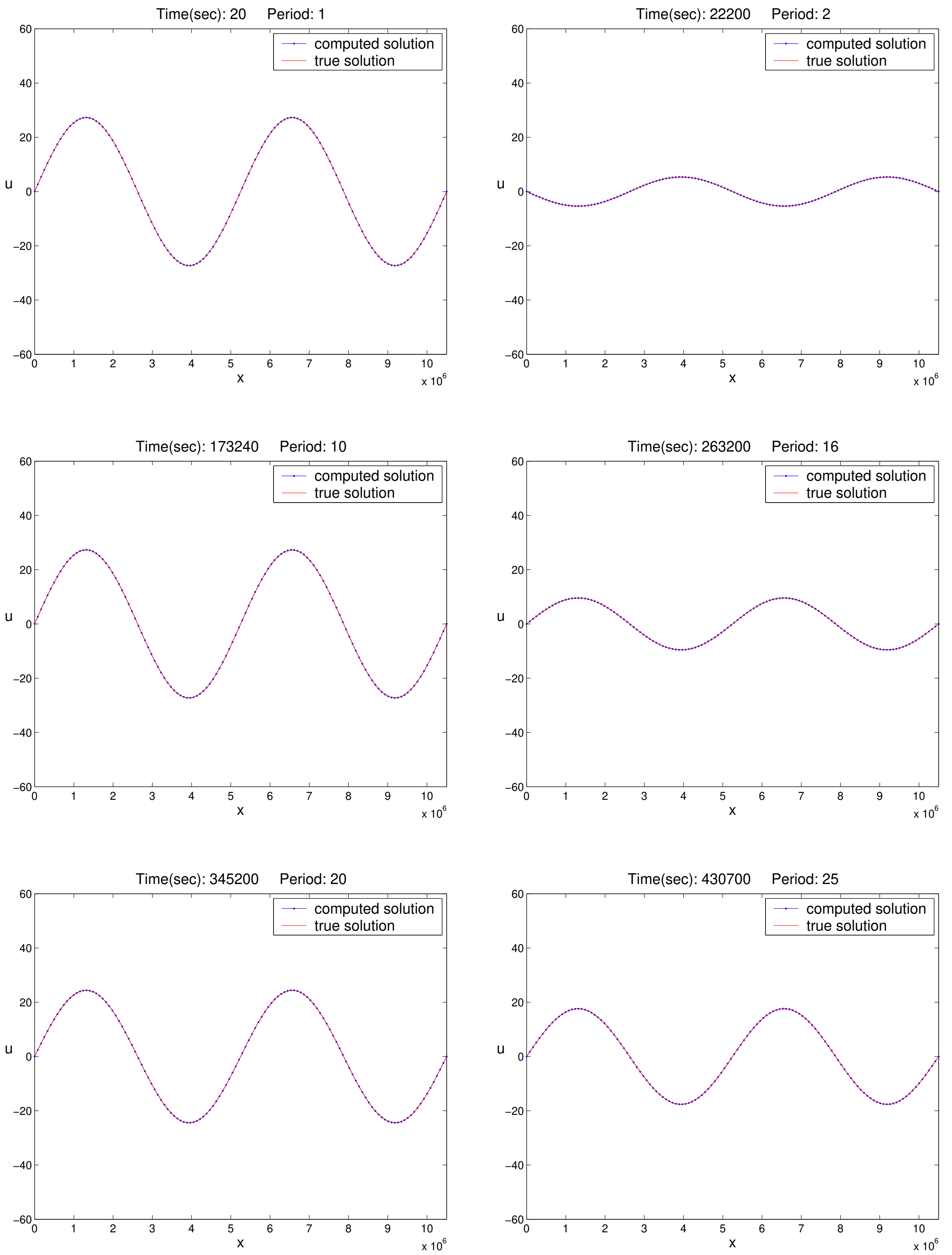


Figure 5.22: Snapshots of the computed approximation to the solution u of Problem 6 with nonzero right hand side, integrated with time step-size 10 seconds on 128 discretization grid points. The full multigrid method is used to solve the CSC linear system at each time step.

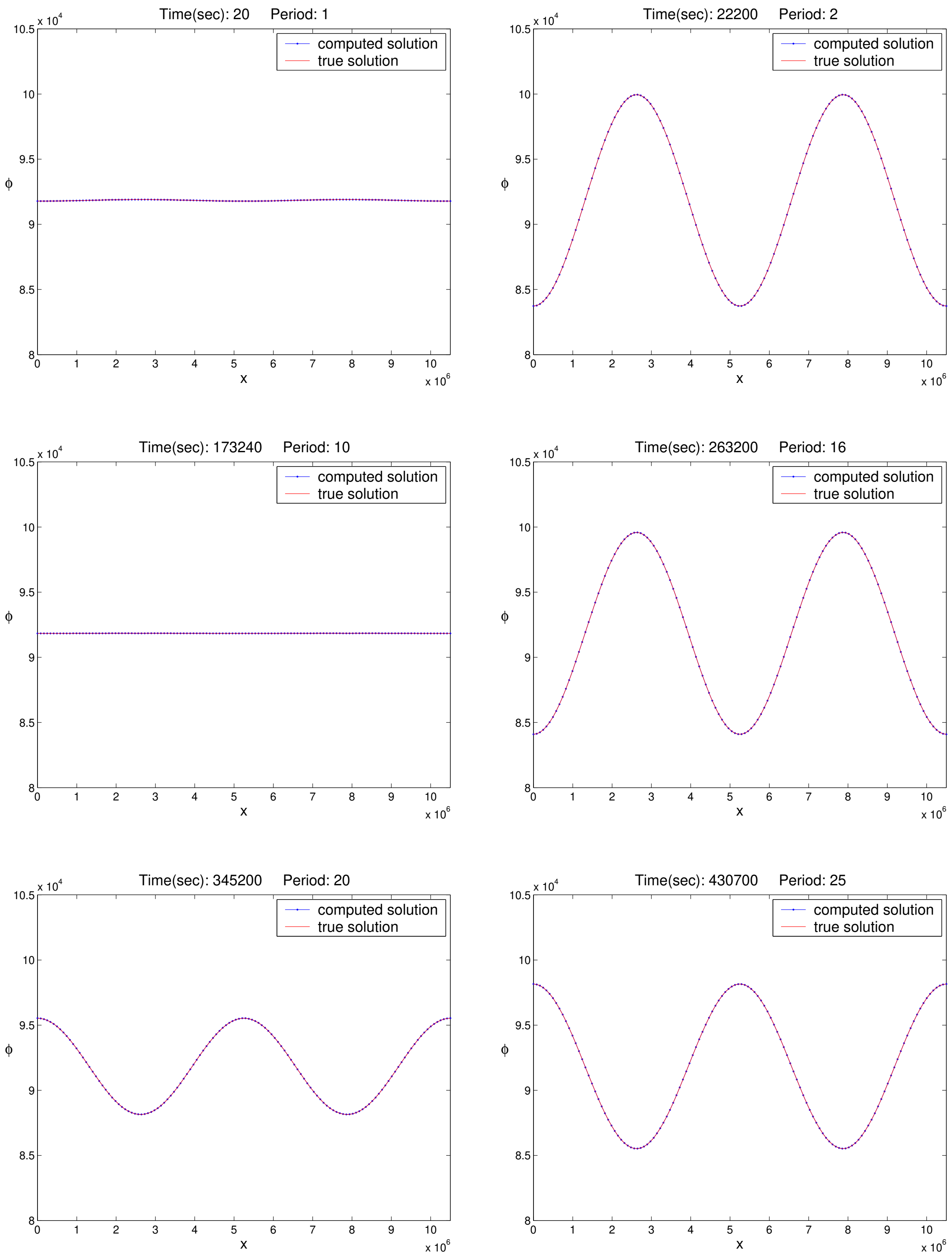


Figure 5.23: Snapshots of the computed approximation to the solution ϕ of Problem 6 with nonzero right hand side, integrated with time step-size 10 seconds on 128 discretization grid points. The full multigrid method is used to solve the CSC linear system at each time step.

	$n = 16$	$n = 32$		$n = 64$		$n = 128$	
$period$	$\frac{\|\|u-u_\Delta\|\|_{x_i}}{\|\|u\|\|_{x_i}}$	$\frac{\|\|u-u_\Delta\|\|_{x_i}}{\|\|u\|\|_{x_i}}$	Convg.	$\frac{\|\|u-u_\Delta\|\|_{x_i}}{\|\|u\|\|_{x_i}}$	Convg.	$\frac{\|\|u-u_\Delta\|\|_{x_i}}{\|\|u\|\|_{x_i}}$	Convg.
1	1.18e-05	1.60e-06	2.88	2.04e-07	2.97	2.57e-08	2.99
3	2.12e-02	2.80e-03	2.92	3.65e-04	2.94	4.52e-05	3.01
5	5.08e-02	6.66e-03	2.93	8.06e-04	3.05	9.87e-05	3.03
9	9.57e-02	1.57e-02	2.61	2.62e-03	2.58	3.17e-04	3.05
11	1.08e-01	1.95e-02	2.47	3.64e-03	2.42	5.56e-04	2.71
13	1.22e-01	2.21e-02	2.46	4.42e-03	2.32	7.99e-04	2.47
15	1.39e-01	2.36e-02	2.56	4.95e-03	2.25	9.87e-04	2.32
21	1.76e-01	2.77e-02	2.67	5.64e-03	2.30	1.24e-03	2.18
23	1.85e-01	3.07e-02	2.59	5.77e-03	2.41	1.27e-03	2.18
25	1.95e-01	3.40e-02	2.52	5.82e-03	2.55	1.28e-03	2.18

	$n = 16$	$n = 32$		$n = 64$		$n = 128$	
$period$	$\frac{\|\|\phi-\phi_\Delta\|\|_{x_i}}{\|\|\phi\|\|_{x_i}}$	$\frac{\|\|\phi-\phi_\Delta\|\|_{x_i}}{\|\|\phi\|\|_{x_i}}$	Convg.	$\frac{\|\|\phi-\phi_\Delta\|\|_{x_i}}{\|\|\phi\|\|_{x_i}}$	Convg.	$\frac{\|\|\phi-\phi_\Delta\|\|_{x_i}}{\|\|\phi\|\|_{x_i}}$	Convg.
1	7.45e-07	4.44e-08	4.07	3.07e-09	3.85	5.27e-10	2.54
3	2.84e-03	2.12e-04	3.75	1.54e-05	3.78	2.59e-06	2.57
5	5.29e-03	4.66e-04	3.51	4.10e-05	3.51	7.14e-06	2.52
9	1.06e-02	7.93e-04	3.73	8.56e-05	3.21	2.40e-05	1.83
11	1.34e-02	9.15e-04	3.87	1.03e-04	3.15	3.05e-05	1.76
13	1.57e-02	1.01e-03	3.97	1.31e-04	2.94	3.39e-05	1.95
15	1.79e-02	1.06e-03	4.07	1.52e-04	2.81	3.62e-05	2.07
21	2.48e-02	1.15e-03	4.42	1.85e-04	2.64	4.50e-05	2.04
23	2.66e-02	1.16e-03	4.51	2.04e-04	2.51	4.60e-05	2.15
25	2.82e-02	1.20e-03	4.55	2.18e-04	2.46	4.66e-05	2.23

Table 5.8: Observed relative relative error and the corresponding orders of convergence of u and ϕ for Problem 6 with nonzero right hand side. The full multigrid method is used to solve the resulting CSC linear system at each time step.

	$n = 16$	$n = 32$		$n = 64$		$n = 128$	
period	$\frac{\|\|u-u_\Delta\|\|_{x_i}}{\|\|u\|\|_{x_i}}$	$\frac{\|\|u-u_\Delta\|\|_{x_i}}{\|\|u\|\|_{x_i}}$	Convg.	$\frac{\|\|u-u_\Delta\|\|_{x_i}}{\|\|u\|\|_{x_i}}$	Convg.	$\frac{\|\|u-u_\Delta\|\|_{x_i}}{\|\|u\|\|_{x_i}}$	Convg.
1	1.03e-05	1.19e-06	3.11	9.92e-08	3.58	7.27e-09	3.77
3	6.69e-01	1.01e-01	2.73	1.05e-02	3.26	5.54e-04	4.24
5	1.74e+00	2.82e-01	2.63	6.81e-02	2.05	9.29e-03	2.87
9	7.72e-01	6.34e-01	0.28	2.17e-01	1.55	6.39e-02	1.77
11	9.91e-02	8.28e-01	-3.06	2.91e-01	1.51	8.98e-02	1.70
13	6.70e-01	1.07e+00	-0.67	3.62e-01	1.56	1.07e-01	1.76
15	1.67e+00	1.32e+00	0.34	4.44e-01	1.57	1.19e-01	1.90
21	1.68e-01	1.88e+00	-3.48	4.92e-01	1.93	1.96e-01	1.33
23	6.90e-01	1.95e+00	-1.50	4.98e-01	1.97	2.06e-01	1.28
25	1.62e+00	1.93e+00	-0.26	5.00e-01	1.95	2.08e-01	1.27
	$n = 16$	$n = 32$		$n = 64$		$n = 128$	
period	$\frac{\|\|\phi-\phi_\Delta\|\|_{x_i}}{\|\|\phi\|\|_{x_i}}$	$\frac{\|\|\phi-\phi_\Delta\|\|_{x_i}}{\|\|\phi\|\|_{x_i}}$	Convg.	$\frac{\|\|\phi-\phi_\Delta\|\|_{x_i}}{\|\|\phi\|\|_{x_i}}$	Convg.	$\frac{\|\|\phi-\phi_\Delta\|\|_{x_i}}{\|\|\phi\|\|_{x_i}}$	Convg.
1	1.66e-05	4.19e-06	1.99	1.05e-06	2.00	2.63e-07	2.00
3	8.72e-02	3.12e-02	1.48	9.36e-03	1.74	2.44e-03	1.94
5	5.20e-02	5.66e-02	-0.12	1.94e-02	1.55	6.15e-03	1.66
9	8.32e-02	1.06e-01	-0.35	2.96e-02	1.84	1.24e-02	1.25
11	4.33e-03	1.25e-01	-4.85	3.27e-02	1.93	1.38e-02	1.25
13	7.91e-02	1.21e-01	-0.61	3.74e-02	1.69	1.43e-02	1.39
15	4.61e-02	1.14e-01	-1.31	3.91e-02	1.55	1.46e-02	1.43
21	6.21e-03	3.64e-02	-2.55	3.48e-02	0.06	1.43e-02	1.28
23	7.08e-02	1.21e-02	2.55	3.62e-02	-1.58	1.44e-02	1.33
25	4.26e-02	2.02e-02	1.08	3.90e-02	-0.95	1.44e-02	1.44

Table 5.9: Observed relative relative error and the corresponding orders of convergence of u and ϕ for Problem 6 with nonzero right hand side. Finite difference is used as the discretization method at each time step.

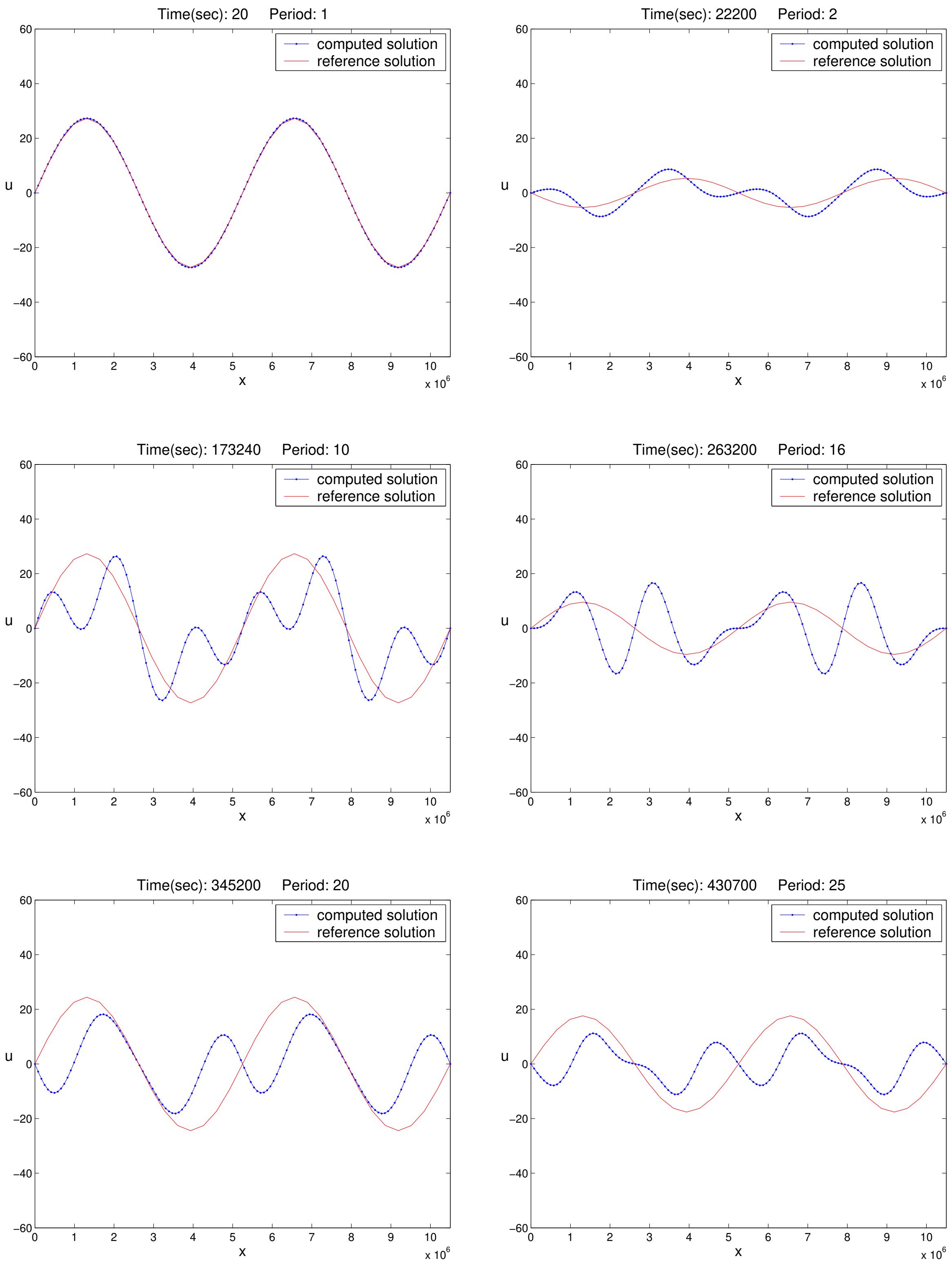


Figure 5.24: Snapshots of the computed approximation to the solution u of Problem 6 with zero right hand side, integrated with time step-size 10 seconds on 32 discretization grid points. The full multigrid method is used to solve the resulting CSC linear system at each time step.

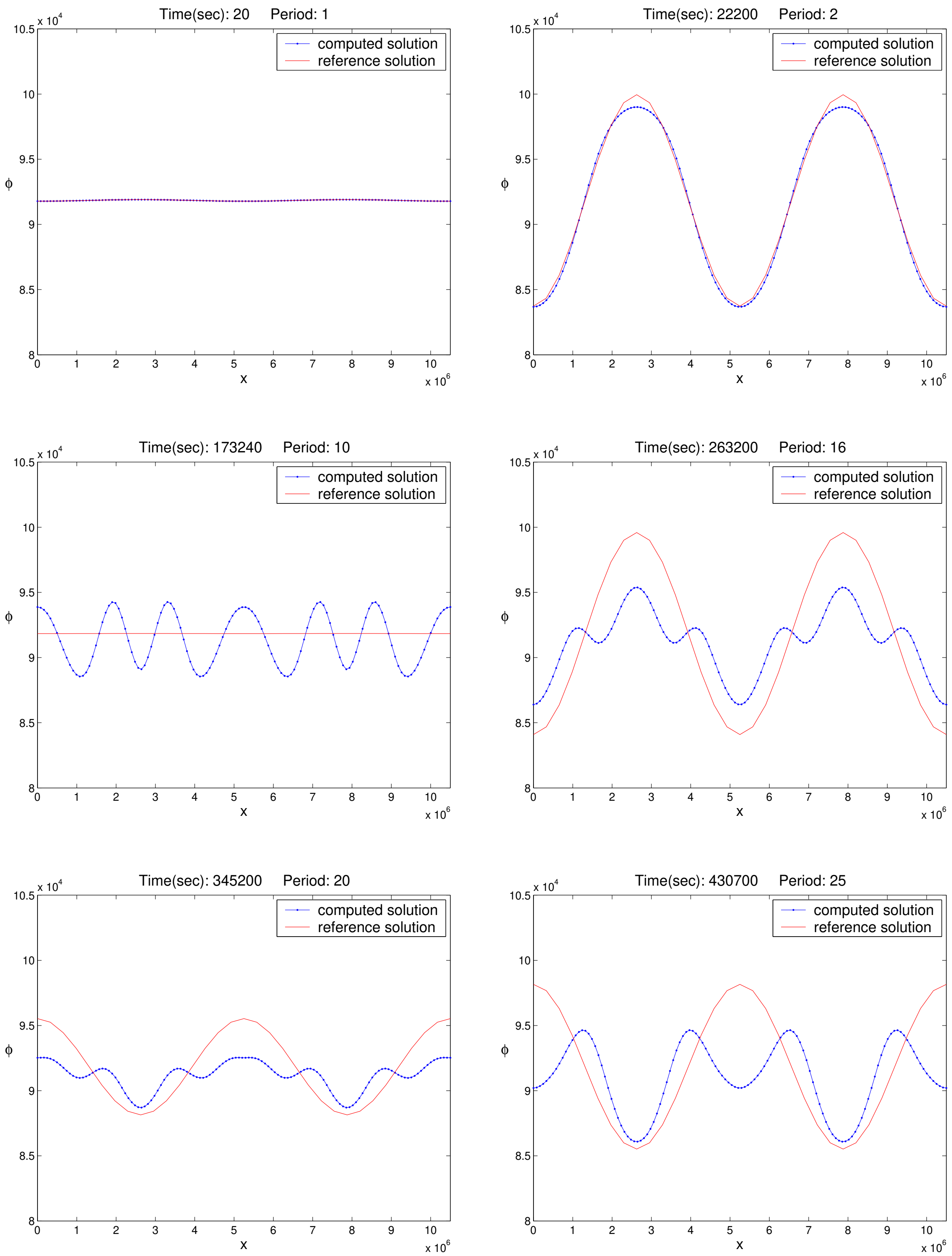


Figure 5.25: Snapshots of the computed approximation to the solution ϕ of Problem 6 with zero right hand side, integrated with time step-size 10 seconds on 32 discretization grid points. The full multigrid method is used to solve the resulting CSC linear system at each time step.

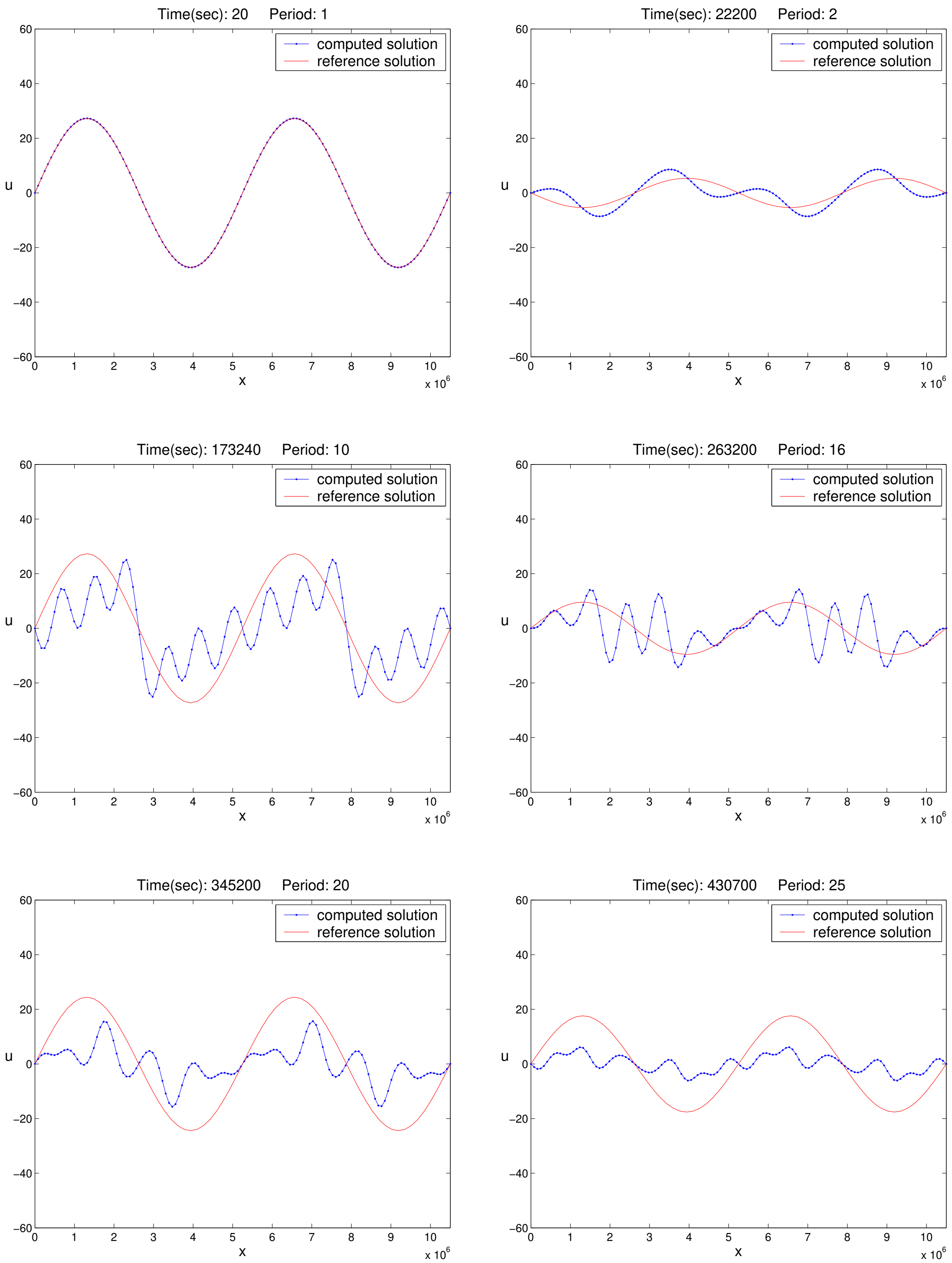


Figure 5.26: Snapshots of the computed approximation to the solution u of Problem 6 with zero right hand side, integrated with time step-size 10 seconds on 64 discretization grid points. The full multigrid method is used to solve the resulting CSC linear system at each time step.

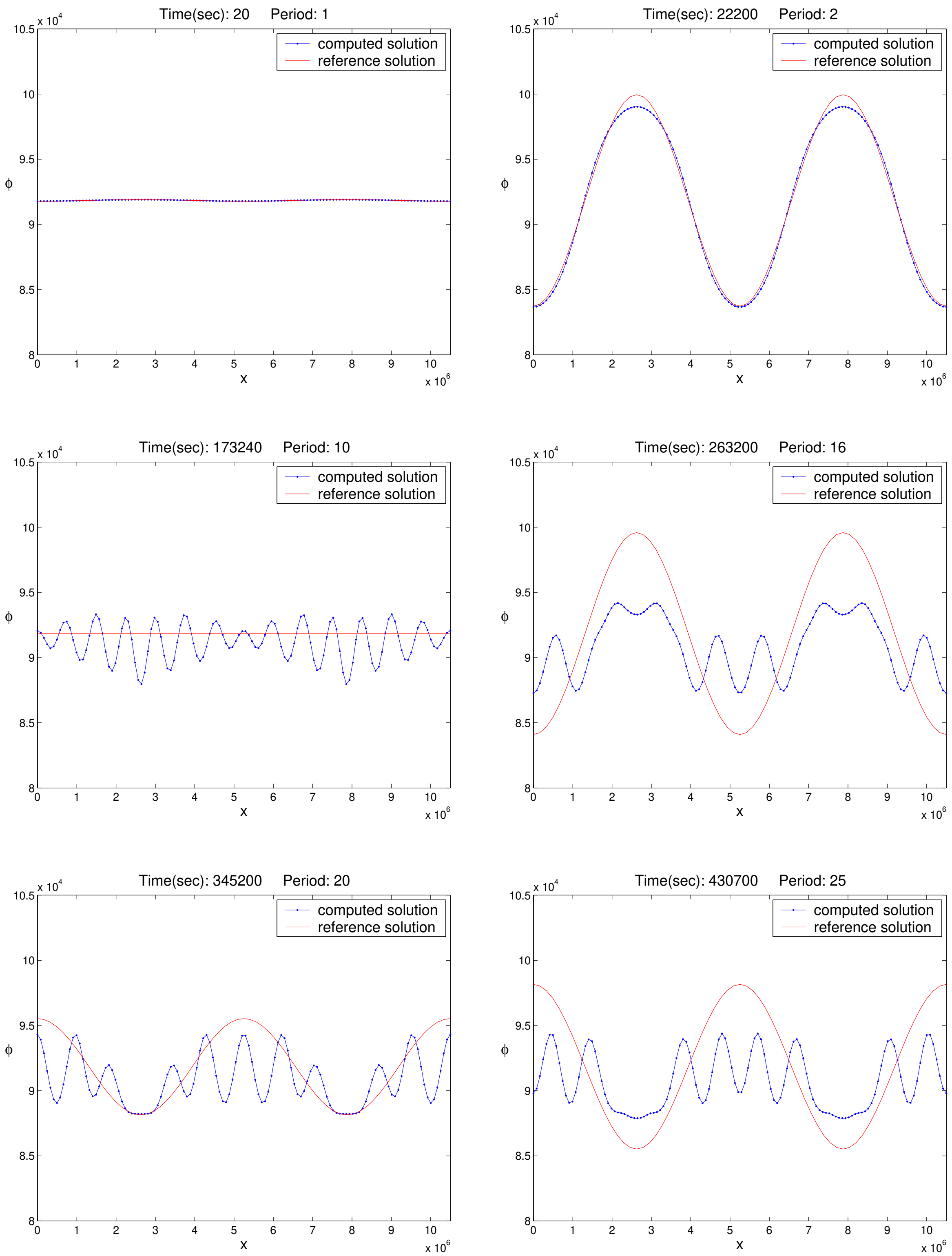


Figure 5.27: Snapshots of the computed approximation to the solution ϕ of Problem 6 with zero right hand side, integrated with time step-size 10 seconds on 64 discretization grid points. The full multigrid method is used to solve the resulting CSC linear system at each time step.

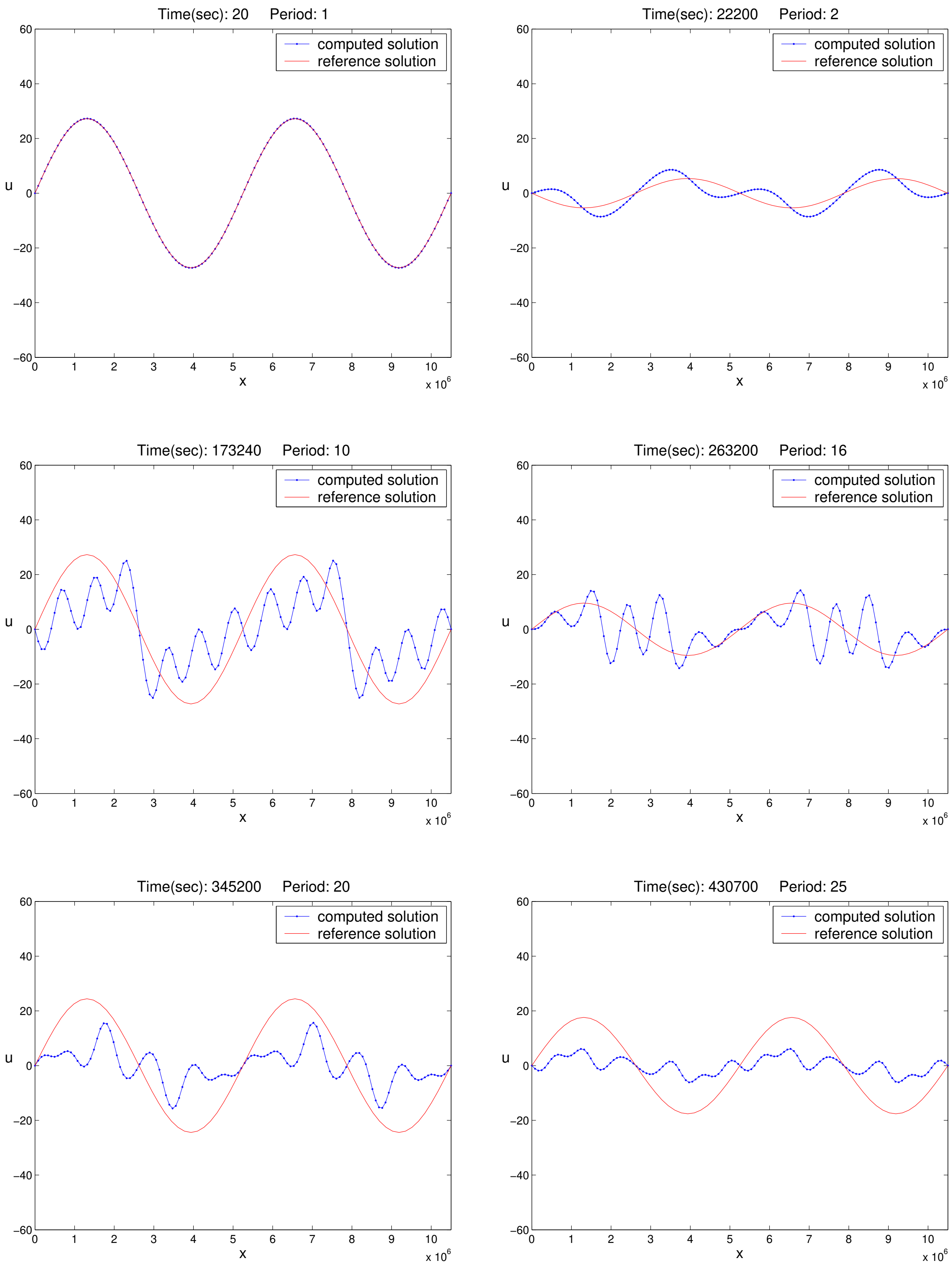


Figure 5.28: Snapshots of the computed approximation to the solution u of Problem 6 with zero right hand side, integrated with time step-size 10 seconds on 64 discretization grid points. The direct method is used to solve the resulting CSC linear system at each time step.

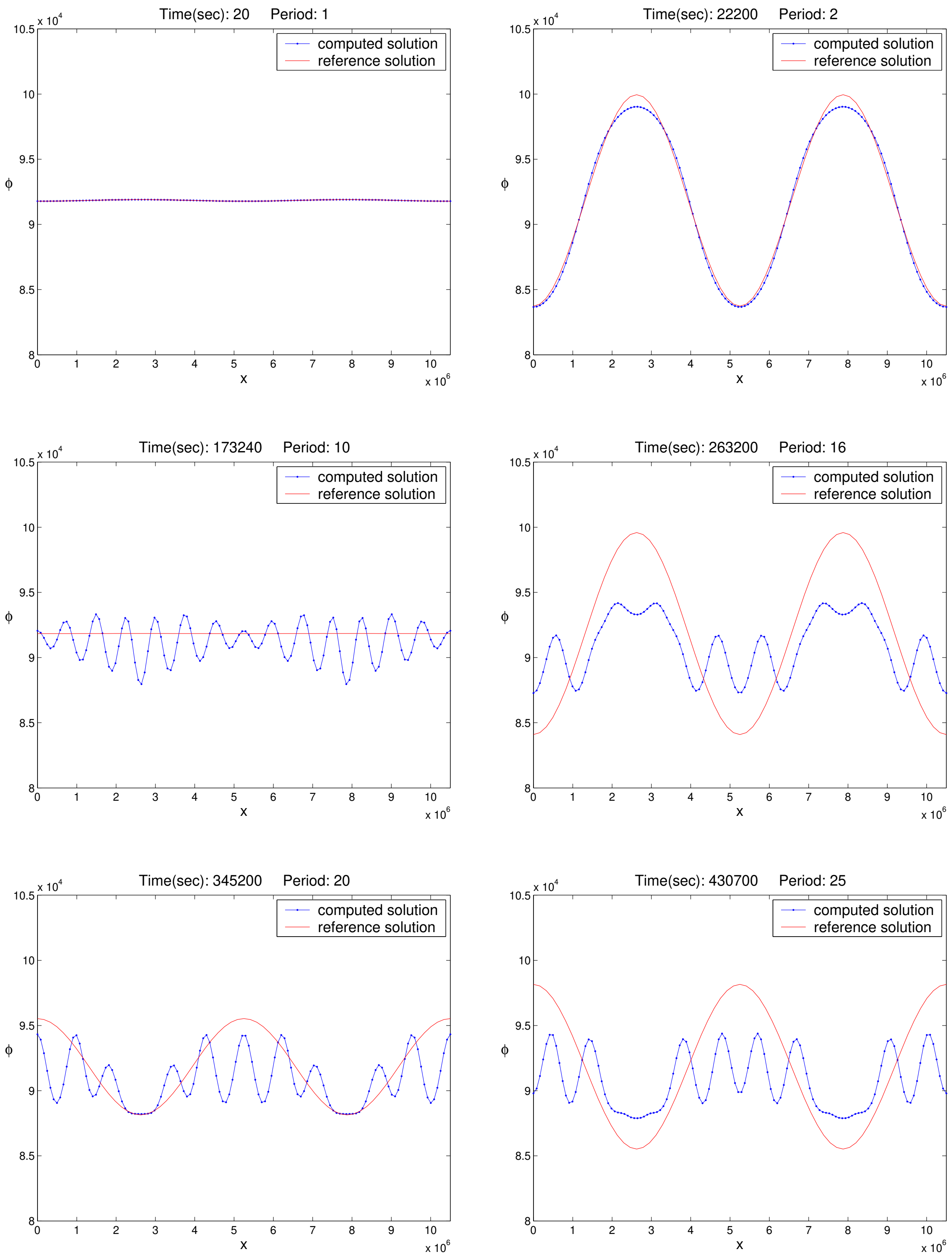


Figure 5.29: Snapshots of the computed approximation to the solution ϕ of Problem 6 with zero right hand side, integrated with time step-size 10 seconds on 64 discretization grid points. The direct method is used to solve the resulting CSC linear system at each time step.

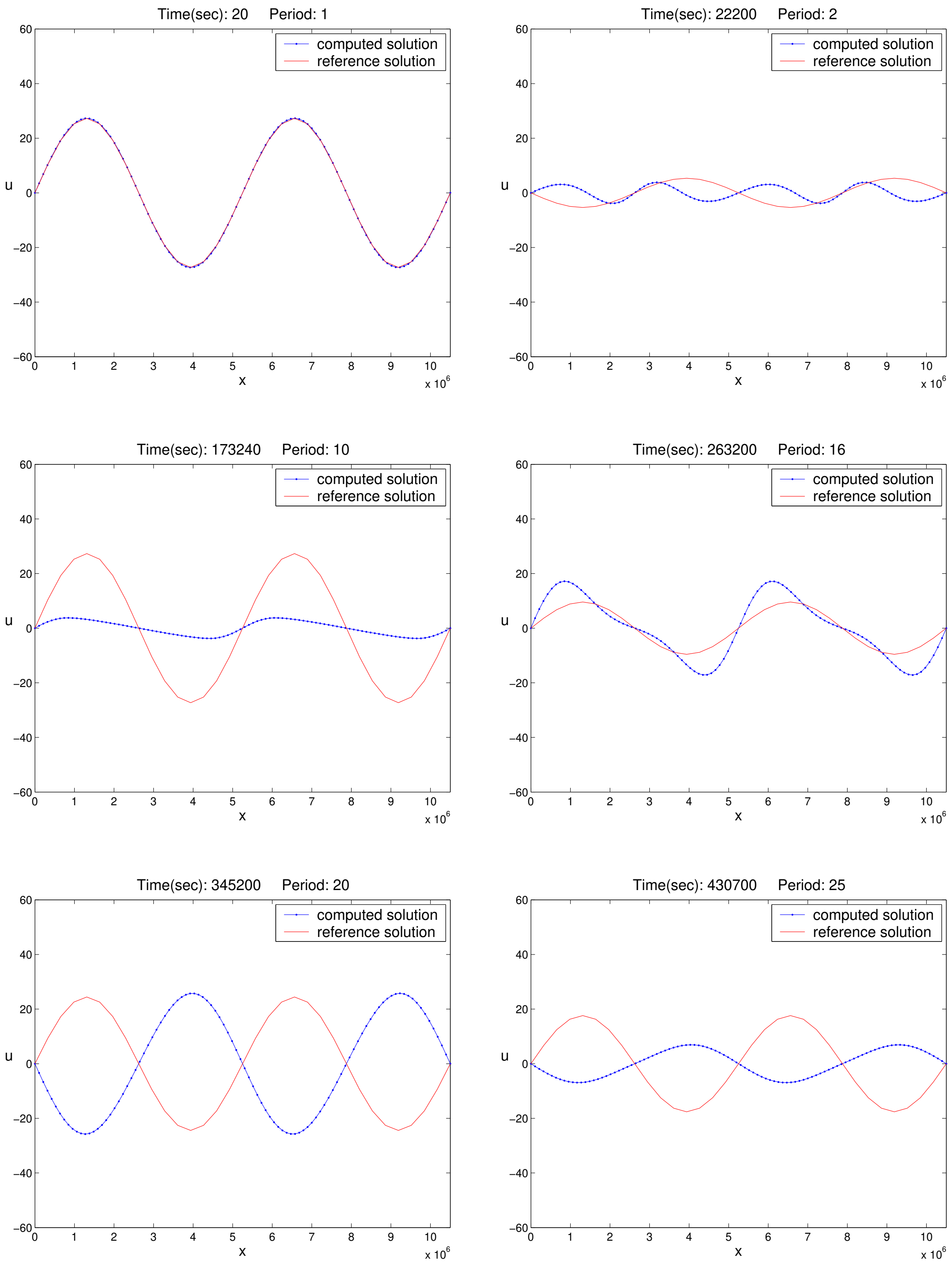


Figure 5.30: Snapshots of the computed approximation to the solution u of Problem 6 with zero right hand side, integrated with time step-size 10 seconds on 32 discretization grid points. Finite difference is used as the discretization method at each time step.

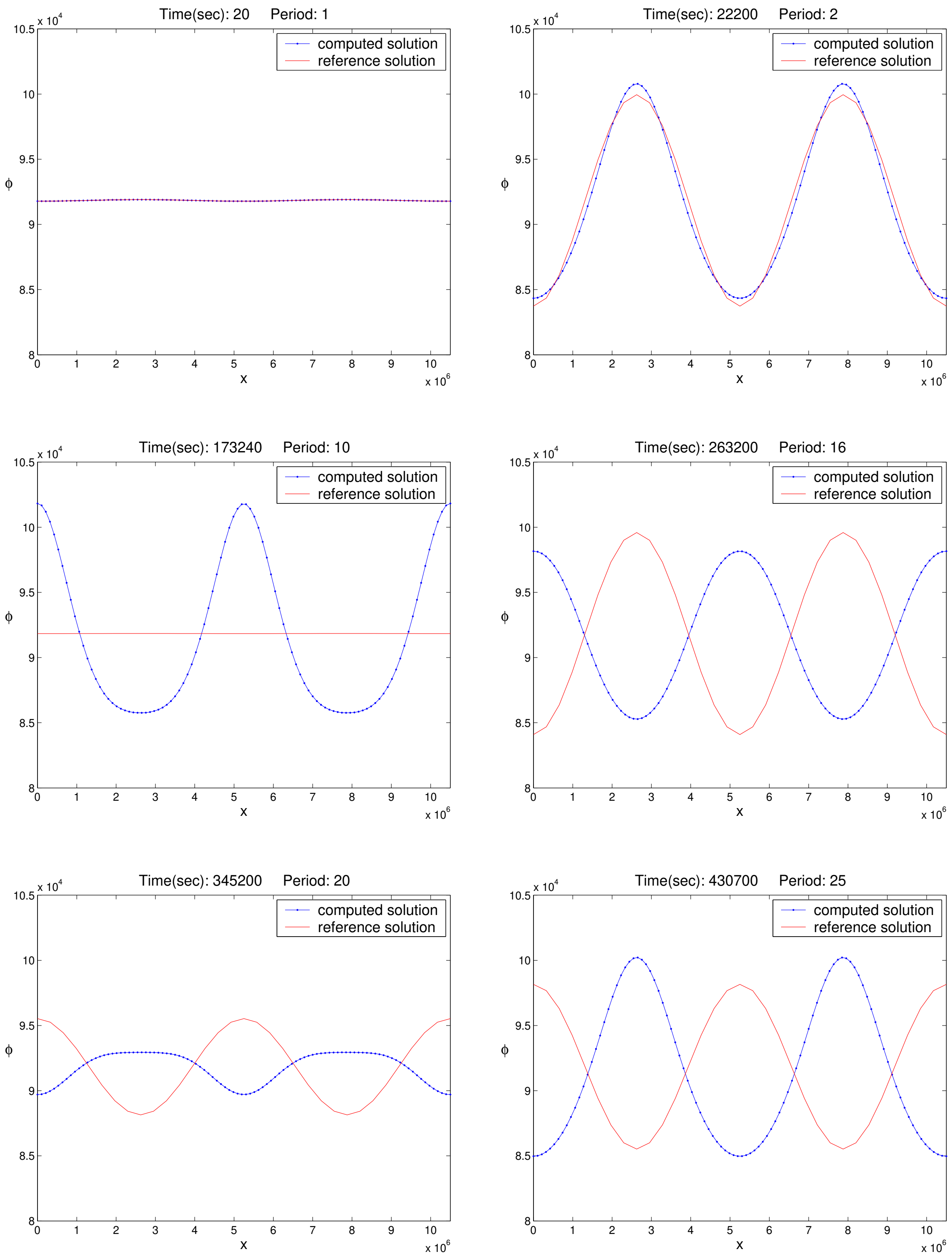


Figure 5.31: Snapshots of the computed approximation to the solution ϕ of Problem 6 with zero right hand side, integrated with time step-size 10 seconds on 32 discretization grid points. Finite difference is used as the discretization method at each time step.

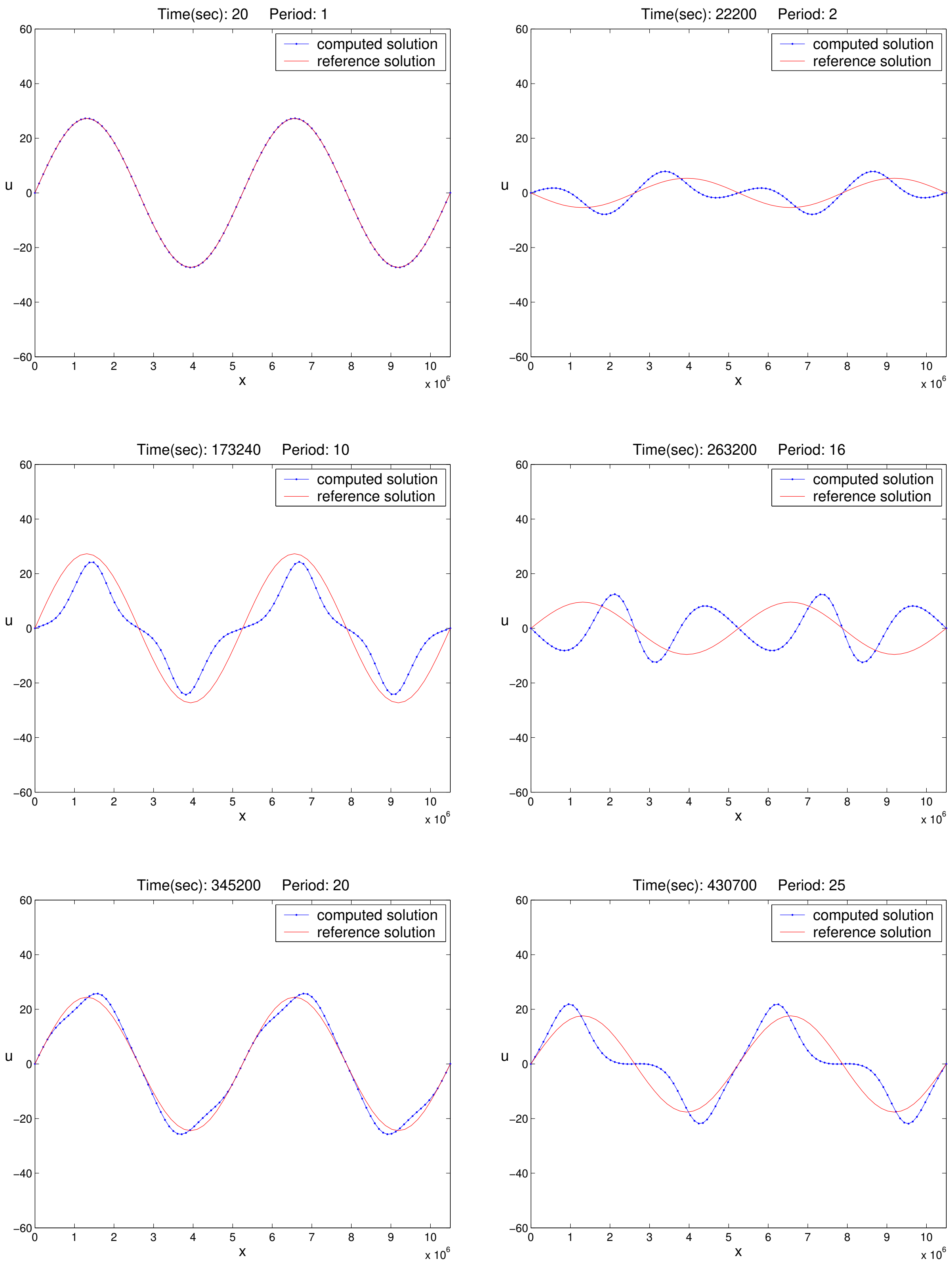


Figure 5.32: Snapshots of the computed approximation to the solution u of Problem 6 with zero right hand side, integrated with time step-size 10 seconds on 64 discretization grid points. Finite difference is used as the discretization method at each time step.

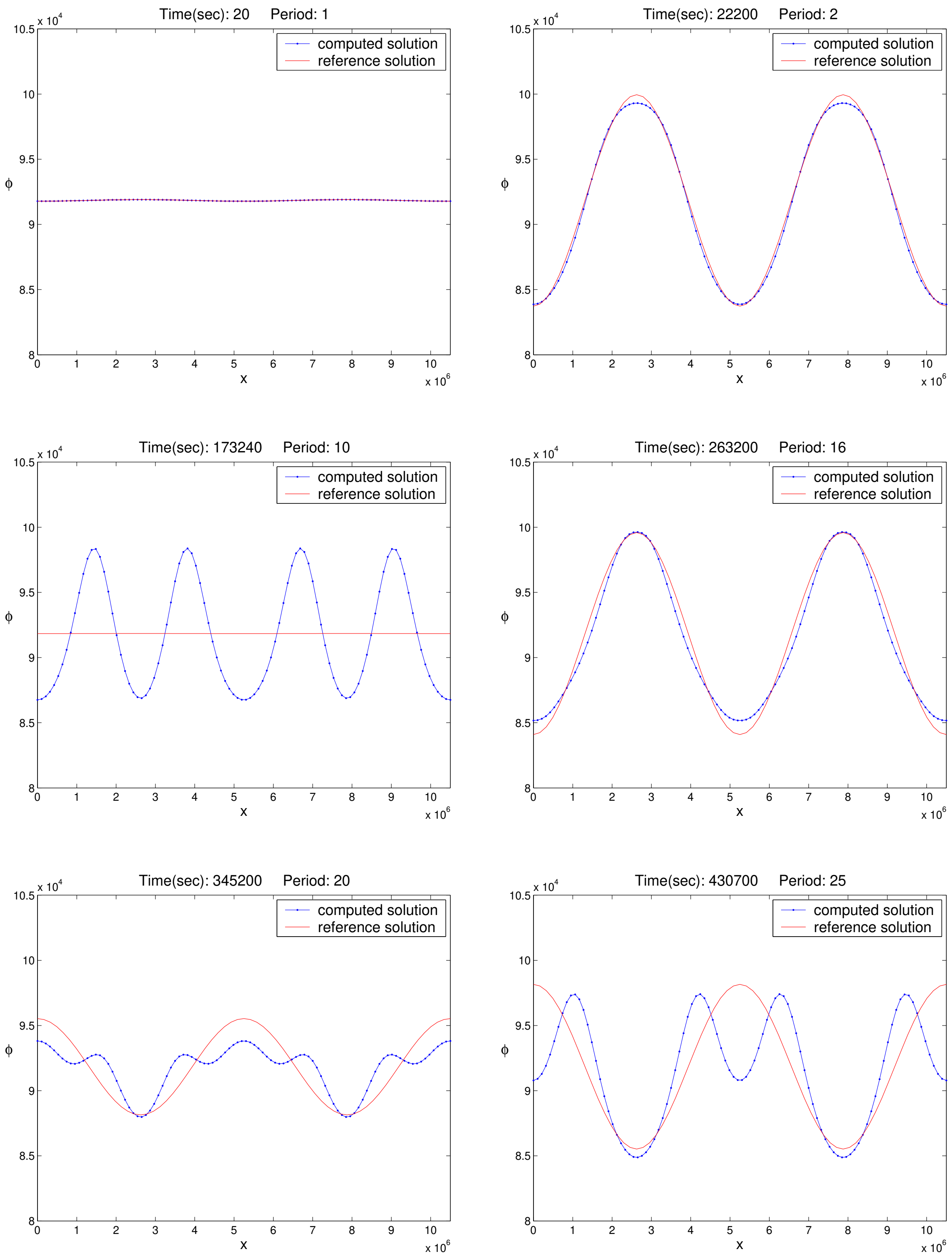


Figure 5.33: Snapshots of the computed approximation to the solution ϕ of Problem 6 with zero right hand side, integrated with time step-size 10 seconds on 64 discretization grid points. Finite difference is used as the discretization method at each time step.

Chapter 6

Conclusion

In this thesis, numerical methods based on the multigrid and CSC methodologies are developed for the solution of advection equations in one-dimensional space.

We first develop multigrid methods for cubic splines by defining appropriate restriction and extension operators for cubic splines that satisfy various BCs, i.e., Dirichlet, Neumann, periodic and general BCs. We apply the multigrid methods to one-dimensional two-point second order BVPs with periodic BCs and prove mathematically that the two-grid method integrated with a damped Richardson relaxation scheme as smoother converges with a contraction factor less than or equal to $1/2$ in the Euclidean norm, independently of the discretization step-size.

Then, we extend the multigrid methods for cubic splines to the solution of one-dimensional SWEs. The SWEs are discretized in time with a semi-Lagrangian semi-implicit scheme, and in space with CSC. There are three different approaches to perform the CSC discretization at each time step: analytic elimination – discretization (AED), discretization – analytic elimination (DAE), and discretization – no elimination (DNE). The AED approach produces a linear system with nice properties, such as symmetry and diagonal dominance, but it does not preserve some stability properties of SWEs. The DAE approach does not suffer from the instability problem, but it generates a nonstandard linear system which is computationally expensive to solve. The DNE approach has the same nice stability properties, but results in a linear system twice as big as those of the first two approaches. Fortunately, with multigrid methods, we could bear the burden of solving the increased linear system for the reason of stability.

We have developed new numerical methods based on multigrid and CSC for solving the SWEs discretized with either the AED or the DNE approaches. We have compared our methods with Jacobi's iterative method applied to the same problems. We have carried out a convergence analysis and discussed advantages and disadvantages of each technique. We have demonstrated that the multigrid methods proposed are convergent and efficient even with the DNE approach. In particular, the multigrid methods do not impose any restriction on the time step-size when applied to the AED system, while they impose a less stringent restriction than Jacobi, when applied to the DNE system.

We have tested our methods by applying them to various problems. Numerical results have confirmed our analysis and showed that our multigrid methods for CSC are convergent and efficient for solving advection equations. Notice that the multigrid solver produces the same results as the direct solver. This indicates that our methods are accurate and thus are preferable when applied to large systems.

Even though only one-dimensional problems are considered in this thesis, our methods can be extended to higher dimensions. Future work includes applying the multigrid CSC methods to two-dimensional SWEs, in particular, the SWEs on the sphere in spherical coordinates. Research in different restriction and extension operators for the multigrid methods is also of great interest.

Bibliography

[1] D. Archer. An $O(h^4)$ cubic spline collocation method for quasilinear parabolic equations. *SIAM J. Numer. Anal.*, 14(4):620–637, 1977.

[2] J. P. Boyd. The nolinear equatorial Kelvin wave. *Journal of Physical Oceanography*, 10:1–11, 1980.

[3] A. Brandt and I. Yavneh. Accelerated multigrid convergence and high-Reynolds recirculating flows. *SIAM J. Sci. Comput.*, 14(3):607–626, 1993.

[4] C. C. Christara and B. Smith. Multigrid and multilevel methods for quadratic spline collocation. *BIT*, 37:4:781–803, 1997.

[5] J. Côté and A. Staniforth. A two-time-level semi-Lagrangian semi-implicit scheme for spectral models. *Mon. Weather Rev.*, 116:2003–2012, 1988.

[6] J. Côté and A. Staniforth. An accurate and efficient finite-element global model of the shallow-water equations. *Mon. Weather Rev.*, 118:2707–2717, 1990.

[7] J. W. Daniel and B. K. Swartz. Extrapolated collocation for two-point boundary-value problems using cubic splines. *J. Inst. Maths Applics*, 16:161–174, 1975.

[8] J. E. Dendy. Black box multigrid for nonsymmetric problems. *Applied Mathematics and Comuptation*, 13:261:283, 1983.

[9] D. J. Fyfe. The use of cubic splines in the solution of two-point boundary value problems. *Comput. J.*, 12:188–192, 1969.

[10] J. Gary. The multigrid iteration applied to the collocation method. *SIAM J. Numer. Anal.*, 18:2:211–224, 1975.

[11] G. J. Haltiner and R. T. Williams. *Numerical Prediction and Dynamics Meteorology*. Wiley, 1980.

[12] A. T. Layton. Cubic spline collocation method for the shallow water equations on the sphere. *Journal of Computational Physics*, pages 578–592, 2004.

[13] A. T. Layton, C. C. Christara, and K. R. Jackson. Quadratic spline collocation methods for the shallow water equations on the sphere: Collocation. *Mathematics and Computers in Simulation*, 2004.

[14] A. T. Layton, C. C. Christara, and K. R. Jackson. Quadratic spline collocation methods for the shallow water equations on the sphere: Galerkin. *Mathematics and Computers in Simulation*, 2004.

[15] K. S. Ng. Quadratic spline collocation methods for systems of elliptic PDEs, 2000.

[16] P. M. Prenter. *Splines and Variational Methods*. John Wiley & Sons, 1975.

[17] A. Robert. A semi-Lagrangian and semi-implicit numerical integration scheme for the primitive meteorological equations. *Jpn. Meteor. Soc.*, 146:319–325, 1982.

[18] R. Sadourny. The dynamics of finite-difference models of the shallow-water equations. *Journal of Atmospheric Sciences*, 32(4), 1975.

[19] A. Staniforth and H. Mitchell. A semi-implicit finite-element barotropic model. *Mon. Weather Rev.*, 105:154–169, 1977.

[20] A. W. Tam. *High-Order Spatial Discretization Methods for the Shallow Water Equations*. PhD thesis, Department of Computer Science, University of Toronto, Canada, 2001.

[21] U. Trottenberg, C. W. Oosterlee, and A. Schüller. *Multigrid*. Academic Press, 2001.

[22] H. Wang, P. Halpern, J. Douglas, and T. Dupont. Numerical solution of the one-dimensional primitive equations using Galerkin approximations with localized basis functions. *Mon. Weather Rev.*, 100:738–746, 1972.

[23] T. Weiyan. *Shallow Water Hydrodynamics: Mathematical Theory and Numerical Solution for a Two-dimensional System of Shallow Water Equations*. Elsevier, Amsterdam, 1992.

[24] J. Xu and L. Zikatanov. A monotone finite element scheme for convection-diffusion equations. *Math. Comp.*, 68(228):1429–1446, 1999.

[25] I. Yavneh, C. H. Venner, and A. Brandt. Fast multigrid solution of the advection problem with closed charateristics. *SIAM J. Sci. Comput.*, 19(1):111–125, 1998.