

A comparative study of the performance of high resolution advection schemes in the context of data assimilation

S. Akella and I. M. Navon^{†,‡}

*School of Computational Science and Department of Mathematics,
Florida State University, Tallahassee, FL 32306, U.S.A.*

SUMMARY

High resolution advection schemes have been developed and studied to model propagation of flows involving sharp fronts and shocks. So far the impact of these schemes in the framework of inverse problem solution has been studied only in the context of linear models. A detailed study of the impact of various slope limiters and the piecewise parabolic method (PPM) on data assimilation is the subject of this work, using the nonlinear viscous Burgers equation as the mathematical model paradigm. The monotonicity preserving limiter of van Leer is shown to be the most efficient in terms of both faster minimization convergence and ability to recover the initial conditions. The results obtained in this work may point out to suitability of these advection schemes for data assimilation in more complex higher dimensional models.

[†]Correspondence to: Prof. I. M. Navon, School of Computational Science, Florida State University,

Tallahassee, FL 32306, U.S.A.

[‡]E-mail: navon@csit.fsu.edu

Contract/grant sponsor: NSF; contract/grant number: ATM-9731472

Received

Revised

Copyright © 2000 John Wiley & Sons, Ltd.

KEY WORDS: finite volume methods; high resolution schemes; adjoint model; data assimilation; Burgers equation

1. INTRODUCTION

Spatial discretization methods for solving partial differential equations (PDEs) can be broadly classified as finite difference (FD) [1, 2], finite volume (FV) [3], finite element (FE) [4, 5, 6] and spectral methods [7]. All of these methods combined with explicit or implicit time integration schemes can be effectively applied to solve parabolic and elliptic PDEs. Whereas for solutions of hyperbolic PDEs which are used to model fluid flows, acoustic waves etc., only the FD, FV and spectral methods have been shown to be most effective in higher dimensions for complicated geometries (however, generation of grids for finite difference methods in complex geometries is a non-trivial task).

For numerical solutions of conservation laws, such the Euler equations in gas dynamics [8] which describe evolution and propagation of flows involving sharp fronts and shocks, several methods have been suggested in the FD, FV and spectral methods literature. Some of the most popular methods in the FV context are Lax- Wendroff, Lax- Friedrichs, Roe' s, flux corrected transport methods of Boris- Book and Zalesak, slope limited methods of van Leer, piecewise parabolic method (PPM) of Colella and Woodward essentially non-oscillatory schemes of Harten-Shu-Osher (see [3, 9, 10] for details of these methods), to name a few.

In geophysical fluid dynamics problems, discontinuities usually do not develop from smooth

initial conditions; except in cases such as the formation of hydraulic jumps that evolve in the shallow-water flows from smooth initial data. For instance in mid- latitudes, fronts can be formed in low- pressure systems, yet these fronts are not entirely discontinuities. Atmospheric fronts (also substances such as chemical pollutants) are transported from one location to another, described very well by a tracer advection model. Due to the deformation (stretching and shearing) of the velocity field that advects the front, discontinuities can be formed on the resolution scale of the (computational) model, see section 5.3 of [10] for details. As a result of finiteness of clouds, variables such as moisture (density) and temperature are discontinuous (once again, on the scale of the model resolution) across the interface of the cloud [11]. Therefore, from a purely computational stand point, there is a need to apply numerical schemes devised for numerical solutions of conservation laws which support discontinuous solutions, in the geophysical fluid flows. Rood [12] provided a detailed analysis and comparison of various advection schemes on simple a linear atmospheric transport model. Lin et. al [13] have analysed the effect of varying the slope limiters using an atmospheric general circulation model. Lin and Rood [14] have compared the first order upwind, central difference, PPM (modified monotonic and positive definite) and monotonic van Leer schemes, and conclude that their monotonic version [15] of PPM yields most accurate results. Towards the development of a fully operational atmospheric general circulation model based on FV discretization [16], Lin and Rood [17] have implemented slope limited van Leer schemes and the PPM scheme on a shallow water equations model using a semi- Lagrangian semi- implicit time integration scheme. For a discussion and applications of other popular schemes such as MPDATA of Smolarkiewicz [18] and QUICK of Leonard [19, 20], see [21].

Fusing models with measurements (observations) and finding response of a system to

(external) disturbances, all require solving inverse problems and as such the minimization of a goal functional, whose gradient with respect to control variables is efficiently provided via adjoint methods, see [22, 23, 24]. In aerodynamics applications, such as minimization of drag, maximization of lift (as target functionals) are often performed by considering the geometry of the immersed body in the fluid as the control variable, which is called shape optimization [25], for other applications and details, see [22]. In weather forecasting, we desire to fit collected observations with model output by data assimilation, see [24]. The control variables in such a setting, include the initial conditions used for integrating the model forward in time.

The impact of different discretization techniques for advection term(s) in the framework of inverse problems and problems related to data assimilation have not been extensively tested, except for work by Vukićević et al.[21] and Thuburn and Haine [26]. In [21] the authors performed data assimilation experiments to reveal the relationships between their properties with respect to data assimilation with three different (central difference: LEAPFROG, MPDATA, QUICK) schemes for the advection of a passive tracer in two dimensions using a linear 2-D transport equation. Their results indicate that more accurate advection schemes need to be used to solve both, forward and backwards in time to achieve higher accuracy regarding recovery of initial conditions for data assimilation; also the same discretization scheme should be applied consistently both for forward and adjoint model integrations. Thuburn and Haine [26] recall Godunov's theorem (which states that any linear monotonic advection scheme is cannot provide more than first order accuracy), hence they studied the affects on adjoint sensitivity computations using a nonlinear, nonoscillatory (QUICK) scheme on a one dimensional linear advection equation model. They also suggest modifications to advection schemes so as to obtain adjoint sensitivity results that are meaningful (in the

particular physical setting considered by them). In this context, a total variation diminishing (TVD) scheme (it is worth mentioning that the TVD property, first introduced by Harten [27] is more relaxed than the monotonicity preserving condition of Godunov) based on a slope limiter, though the advection scheme does not satisfy the linearity property of the governing advection equation model, for obtaining well-behaved sensitivity results has been suggested. Since the mathematical models used to study fluid flows and weather prediction are highly nonlinear, as a step towards understanding the affects of using high order advection schmes, we study in this work the impact of using FV methods that are slope limited using van Leer type and PPM for spatial discretization to perform data assimilation with a nonlinear viscous Burgers equation model (which serves as a proxy for more complex models) in a one dimensional spatial setting. We show that for a particular smooth initial condition, we obtain a smooth solution for this model problem (in the context of smoothness property of geophysical flows as discussed above), and implement the adjoint method to successfully recover initial conditions in 4-D VAR data assimilation set-up.

The plan of the paper is as follows. In section two we present the forward model as well as describe numerical solution of the nonlinear Burgers equation using FV discretization. Section 3 describes the test case considered along with results obtained with several slope limited TVD schemes as described in section 2. Section 4 describes the derivation of the adjoint and tangent linear models (which are used for data assimilation) and verification of these discrete models. Section 5 provides a brief description of the minimization algorithm used. The performance of the various slope limited and PPM schemes for the minimization of a certain cost functional, in other words, in data assimilation experiments is presented in section 6. Finally in the section of summary and conclusions we discuss the impact of the different advection schemes in the

framework of our numerical results. Some future research work will involve assessing impact of these schemes in the context of 4-D VAR and ensemble Kalman filtering, using the Lin-Rood finite volume shallow water model [17] in spherical coordinates.

2. DESCRIPTION OF THE MATHEMATICAL AND NUMERICAL MODELS

Let us consider the following 1-D (nonlinear) scalar conservation law ($\phi(x, t) \in \mathbb{C}^2$, the space of continuous functions that are at-least twice differentiable)

$$\frac{\partial \phi}{\partial t} + \frac{\partial f}{\partial x} = \frac{\partial S}{\partial x}, \quad (1)$$

where the f is a convex flux function given by $\frac{\phi^2}{2}$ and S represents the source term(s). Equation (1) is the well known Burgers equation [28] which is a very important fluid dynamical model useful for conceptual understanding of nonlinear waves, shock formation [8, 29] and turbulence [30]. Various numerical schemes (see Fletcher [31] for a detailed numerical analysis) have been suggested and tested on this model equation to efficiently capture shocks.

We will now describe and test a variety of finite volume methods [3] to solve the above equation, all differing in the way which we reconstruct the solution, ϕ , in each cell using different slope limiters. We will closely follow the approach taken by Monotone Upstream-centered Schemes for Conservation Laws (MUSCL), see [32, 33, 34, 15, 35].

Let us start by writing the integral form of (1) within the i -th. cell, \mathcal{C}_i ,

$$\frac{\partial}{\partial t} \int_{\mathcal{C}_i} \phi(x, t) dx = f[\phi(x_{i-\frac{1}{2}}, t)] - f[\phi(x_{i+\frac{1}{2}}, t)] + S(x_{i+\frac{1}{2}}, t) - S(x_{i-\frac{1}{2}}, t) \quad (2)$$

$$\mathcal{C}_i : x \in [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}].$$

We define i -th. cell average at time interval t_n ($t \in [t_0, t_{final}]$) has been discretized into a

number of time steps $[t_0, t_1, \dots, t_n]$ as,

$$\Phi_i^n \approx \frac{1}{\Delta x_i} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \phi(x, t_n) dx, \quad (3)$$

where $\Delta x_i = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}$ is the length of the i -th. cell.

Integrating equation (2) from t_n to t_{n+1} yields,

$$\begin{aligned} \int_{C_i} \phi(x, t_{n+1}) dx - \int_{C_i} \phi(x, t_n) dx &= \int_{t_n}^{t_{n+1}} f[\phi(x_{i-\frac{1}{2}}, t)] dt - \int_{t_n}^{t_{n+1}} f[\phi(x_{i+\frac{1}{2}}, t)] dt \\ &+ \int_{t_n}^{t_{n+1}} [S(x_{i+\frac{1}{2}}, t) - S(x_{i-\frac{1}{2}}, t)] dt, \end{aligned}$$

dividing by Δx_i and rearranging,

$$\begin{aligned} \frac{1}{\Delta x_i} \int_{C_i} \phi(x, t_{n+1}) dx &= \frac{1}{\Delta x_i} \int_{C_i} \phi(x, t_n) dx \\ &- \frac{1}{\Delta x_i} \int_{t_n}^{t_{n+1}} \{f[\phi(x_{i+\frac{1}{2}}, t)] - f[\phi(x_{i-\frac{1}{2}}, t)]\} dt + \\ &+ \frac{1}{\Delta x_i} \int_{t_n}^{t_{n+1}} [S(x_{i+\frac{1}{2}}, t) - S(x_{i-\frac{1}{2}}, t)] dt. \end{aligned}$$

Assuming a viscous dissipative source $S = \nu \phi_x$ (ν is the kinematic viscosity) and using equation (3) we obtain,

$$\begin{aligned} \Phi_i^{n+1} &= \Phi_i^n - \frac{\Delta t}{\Delta x_i} [(Flux)_{i+\frac{1}{2}}^n - (Flux)_{i-\frac{1}{2}}^n] + \\ &\nu \frac{\Delta t}{\Delta x_i} [\phi_x(x_{i+\frac{1}{2}}, t) - \phi_x(x_{i-\frac{1}{2}}, t)], \end{aligned} \quad (4)$$

where $(Flux)_{i+\frac{1}{2}}^n \approx \frac{1}{\Delta t} \int_{t_n}^{t_{n+1}} f[\phi(x_{i+\frac{1}{2}}, t)] dt$ is some approximation of the average flux (described later in this section) along the cell interface at $x_{i+\frac{1}{2}}$, see figure (2) for an illustration of the grid cells .

2.1. MUSCL limiters

Within each cell if we consider a piecewise constant approximation to the solution (i.e., slope of the reconstruction is equal to zero), then we obtain a first order method; however if we use a piecewise linear approximation within each cell, \mathcal{C}_i ,

$$\phi(x \in [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]) = \Phi_i + \Delta\Phi_i(x - x_i)$$

where Φ_i is given by equation (3), x_i is the coordinate of the i -th. cell center and $\Delta\Phi_i$ is equal to the difference between the values of the state at the right and left cell interfaces (it denotes the slope of reconstructed solution in each cell), we obtain a family of second order approximate schemes.

Conservation laws such as the Euler equations in gas dynamics [9] and the simple Burgers equation (1) support solutions that have discontinuities (or, shocks), expansion fans, contact discontinuities. Apart from ensuring satisfaction of the CFL (Courant-Friedrichs-Lewy) condition [1], unless special treatment is taken, the numerical solutions will lead to excessive dissipation, incorrect phase speeds, spurious oscillations; see Laney [9] for an extensive comparison of many numerical methods applied to solve simple linear and nonlinear advection and Euler equations.

One way to prevent such spurious oscillations and preserve TVD [27, 36, 3] property is by limiting the values of the slopes ($\Delta\Phi_i$). Lin et al. [13] listed a number of consistent ways of deriving the limited slopes in various forms and compared their impact on the solution of linear advection equation. We will follow their approach for arriving at the various formulations of the slope (from now onwards we will assume an uniform grid, i.e, $\Delta x_i = \Delta x \forall i$).

1. Limitier 1 (first order scheme):

$$\Delta\Phi_i^n \equiv 0, \forall i \quad (5)$$

gives us a first order accurate scheme.

2. Limitier 2 (simplest second order scheme):

$$[\Delta\Phi_i^n]_{avg} = \frac{1}{\Delta x} \frac{\delta\Phi_{i-\frac{1}{2}}^n + \delta\Phi_{i+\frac{1}{2}}^n}{2}, \quad (6)$$

where $\delta\Phi_{i+\frac{1}{2}}^n = \Phi_{i+1}^n - \Phi_i^n$ and "avg" means the averaging operator in the above equation.

This provides us a second order accurate scheme, but the values of the slopes are not limited, in other words, no limiter has yet been applied.

3. Limitier 3 (simple positive definite scheme):

$$[\Delta\Phi_i^n] = \frac{1}{\Delta x} \text{SIGN}([\Delta\Phi_i^n]_{avg}) \cdot \text{MIN}[|[\Delta\Phi_i^n]_{avg}|, 2\text{DIM}(\Phi_i^n, \Phi^{min})], \quad (7)$$

the value of the slope has been limited using the least value (over all of x_i) of Φ_i^n and $[\Delta\Phi_i^n]_{avg}$. $\text{DIM}(p, q)$ is defined as the positive difference between p and q ,

$$\text{DIM}(p, q) = \begin{cases} p - q, & \text{if } p > q \\ 0, & \text{otherwise.} \end{cases}$$

4. Limitier 4 (monotonicity preserving scheme):

Another form of slope which ensures monotonicity, suggested by van Leer [15] is the following,

$$[\Delta\Phi_i^n] = \begin{cases} \frac{1}{\Delta x} [\delta\Phi_{i-\frac{1}{2}}^n \cdot \delta\Phi_{i+\frac{1}{2}}^n] / [\Delta\Phi_i^n]_{avg}, & \text{if } \text{SIGN}(\delta\Phi_{i-\frac{1}{2}}^n) = \text{SIGN}(\delta\Phi_{i+\frac{1}{2}}^n), \\ 0, & \text{otherwise.} \end{cases} \quad (8)$$

5. *Limiter 5* (local min./ max. slope limited scheme):

We can determine locally defined minimum and maximum values of the solution as,

$$\begin{aligned}\Phi_i^{min} &= MIN[\Phi_{i-1}^n, \Phi_i^n, \Phi_{i+1}^n] \\ \Phi_i^{max} &= MAX[\Phi_{i-1}^n, \Phi_i^n, \Phi_{i+1}^n]\end{aligned}\tag{9}$$

and use them to limit the value of the slope as following,

$$[\Delta\Phi_i^n] = \frac{1}{\Delta x} SIGN([\Delta\Phi_i^n]_{avg}) \cdot MIN[|[\Delta\Phi_i^n]_{avg}|, 2DIM(\Phi_i^n, \Phi_i^{min}), 2DIM(\Phi_i^{max}, \Phi_i^n)].\tag{10}$$

6. *Limiter 6* (global min./ max. slope limited scheme):

In the above formulation of the limiter, we used the locally computed minimum and maximum values of the solution. Instead if the global minimum and maximum values of Φ_i^n are set to be equal to Φ_{global}^{min} and Φ_{global}^{max} respectively, and replacing these in above limiter formulation, we obtain:

$$[\Delta\Phi_i^n] = \frac{1}{\Delta x} SIGN([\Delta\Phi_i^n]_{avg}) \cdot MIN[|[\Delta\Phi_i^n]_{avg}|, 2DIM(\Phi_i^n, \Phi_{global}^{min}), 2DIM(\Phi_{global}^{max}, \Phi_i^n)].\tag{11}$$

We will now use these values of slopes and follow the approach of Essentially Non Oscillatory (ENO) schemes to arrive at an expression for the flux at the cell interfaces.

2.2. *ENO flux*

To calculate the flux at the right cell face $x_{i+\frac{1}{2}}$, we used the ENO [37, 38, 39, 9] flux formulation.

Using the i -th. and $i + 1$ cell reconstructed values evaluated at $x_{i+\frac{1}{2}}$ (see Laney [9] chapter 23

for details), we obtain

$$(Flux)_{i+\frac{1}{2}}^n = f^G[\{\Phi_i^n + \frac{\Delta\Phi_i^n \Delta x}{2}(1 - \frac{\Delta t}{\Delta x} \Phi_i^n)\}, \{\Phi_{i+1}^n - \frac{\Delta\Phi_{i+1}^n \Delta x}{2}(1 + \frac{\Delta t}{\Delta x} \Phi_{i+1}^n)\}], \quad (12)$$

where

$$f^G[\Phi_i^n, \Phi_{i+1}^n] = \begin{cases} MIN[f(\Phi_i^n), f(\Phi_{i+1}^n), f(\Phi_*)] & \text{if } \Phi_i^n \leq \Phi_{i+1}^n, \\ MAX[f(\Phi_i^n), f(\Phi_{i+1}^n), f(\Phi_*)] & \text{if } \Phi_i^n > \Phi_{i+1}^n. \end{cases} \quad (13)$$

where Φ_* is such that the flow speed given by, $\frac{\partial f}{\partial \Phi} = \frac{\partial \frac{\Phi^2}{2}}{\partial \Phi} = \Phi = \Phi_* = 0$.

Remark: If the slope in each cell is equal to zero, as in the equation (5), then the above ENO flux form reduces to Godunov flux form [40].

Instead of using a piecewise linear reconstruction within each cell, we can as well apply the piecewise parabolic reconstruction approach of Colella and Woodward [41, 42] within each cell.

2.3. PPM reconstruction

We have applied the PPM to reconstruct the state within each cell and to obtain the values of the state at left and right cell interfaces.

$$\phi(x \in [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]) = \Phi_{L,i} + x[\Delta\Phi_i + \Phi_{6,i}(1-x)].$$

$\Phi_{L,i}$ and $\Phi_{R,i}$ are approximations of the state at the left and right cell interface, as in MUSCL piecewise linear extrapolation, $\Delta\Phi_i = \Phi_{R,i} - \Phi_{L,i}$ and $\Phi_{6,i} = 6(\Phi_i - \frac{1}{2}(\Phi_{L,i} + \Phi_{R,i}))$ for details of the above reconstruction procedure, see [41].

The fluxes at the interfaces have been directly evaluated using the calculated values, $\Phi_{L,i}$

and $\Phi_{R,i}$ for every i -th. cell. We have used a second order Runge-Kutta (R-K) explicit scheme to integrate in time, described below.

2.4. Integration in time using a second order optimal TVD R-K method

Using equations (12) and (13) or the PPM scheme for calculating the flux and forward differencing for the diffusion term, we can write the following simple forward Euler update formula for Φ_i^{n+1} ,

$$\begin{aligned} \Phi_i^{n+1} &= \Phi_i^{n+1} - \frac{\Delta t}{\Delta x} [(Flux)_{i+\frac{1}{2}}^n - (Flux)_{i-\frac{1}{2}}^n] + \\ &\quad \nu \frac{\Delta t}{\Delta x^2} [\Phi_{i+1}^n - 2\Phi_i^n + \Phi_{i-1}^n]. \end{aligned} \quad (14)$$

The above numerical scheme is at-least second order accurate (MUSCL schemes: (6)- (9) second order, whereas PPM being third order accurate) in space for sufficiently smooth ϕ ($\phi \in \mathbb{C}^2$), but it is only first order accurate in time, also it does not preserve the TVD property for time integration. In order to overcome these drawbacks, we used a second order (accurate in time) optimal TVD R-K scheme [43, 44], given by Gottlieb and Shu [45]. Following their notation, let

$$L(\Phi_i^n) = -\frac{1}{\Delta x} [(Flux)_{i+\frac{1}{2}}^n - (Flux)_{i-\frac{1}{2}}^n] + \nu \frac{1}{\Delta x^2} [\Phi_{i+1}^n - 2\Phi_i^n + \Phi_{i-1}^n],$$

then the following sequence of two steps gives us,

$$\Phi_i^{(1)} = \Phi_i^n + \Delta t L(\Phi_i^n),$$

$$\Phi_i^{n+1} = \frac{1}{2}\Phi_i^n + \frac{1}{2}\Phi_i^{(1)} + \frac{1}{2}\Delta t L(\Phi_i^{(1)}).$$

This completes the description of discretization in space and time. We have tested these various finite volume methods using the aforementioned advection schemes. Comparison of the numerical results with the exact solution is provided for the following test case.

3. TEST CASE AND RESULTS

We will consider in this section the following Burgers equations,

$$\frac{\partial \phi}{\partial t} + \frac{\partial}{\partial x} \left(\frac{\phi^2}{2} \right) = \nu \left(\frac{\partial^2 \phi}{\partial x^2} \right), \quad (15)$$

for $x \in (-\pi, \pi)$ and $t > 0$.

Periodic boundary conditions have been prescribed, $\phi = 0$ at $x = \pm \pi$, or, equivalently one can say that boundary conditions have been assigned such that $\phi(x = \pm \pi, t) = 0$.

Benton and Platzman[46] provide an exact solution for the above equation (15), with initial condition given by,

$$\phi(x, 0) = \phi(x, t = 0) = -R \sin(x), \quad (16)$$

where R is the Reynolds number. It is related to the viscosity via the relationship, $R = \frac{UL}{\nu}$, here the values of (velocity scale) U and (length scale) L have been prescribed to be equal to unity. Then the exact solution assumes the form:

$$\phi_{exact}(x, t) = \frac{4 \sum_{n=1}^{\infty} n a_n e^{-n^2 t} \sin(nx)}{a_0 + 2 \sum_{n=1}^{\infty} a_n e^{-n^2 t} \cos(nx)}, \quad (17)$$

where $a_n = (-1)^n I_n(\frac{1}{2}R)$, I_n is the Bessel function of second kind. For small values of R , viscous dissipation dominates over advection and the solution decays uniformly as time, t

increases, as depicted in figure (1) (which has been generated by setting $R = 1$).

In figures (3)- (6) we have plotted the exact solution along with solutions obtained using the various numerical schemes (MUSCL schemes, with limiters 1- 6, and the PPM) at $t = 1$. Since the Bessel functions of second kind are exponentially decreasing functions, to compute ϕ_{exact} we have used $n = 10$ for the summation in equation (17). The numerical solutions have been computed using a resolution of $N_x = 40$ grid cells ($\Delta x = \frac{2\pi}{N_x}$) and a time step (Δt) given by the CFL criteria, $CFL = \frac{U\Delta t}{\Delta x}$, where the CFL number was assigned a value of 0.01.

The first order scheme (limiter 1) and limiter 3 (simple positive definite scheme, which was based on limiting the slope based on the least value of Φ_i^n and $[\Delta\Phi_i^n]_{avg}$) are dissipative, whereas all the other schemes compare very well with the exact solution. It is to be noted that all the numerical solutions have the correct phase speed. In the case of the global min./max. limiter (4), we prescribed $\Phi^{min} = -1$ and $\Phi^{max} = 1$.

In table (I) we show that the numerical solutions converge to the exact solution in both L_2 and L_∞ norms, at $t = 1$. As expected the first order scheme (limiter 1) has the largest error compared to all other schemes. The monotonicity preserving slope limiter of van Leer, limiter 4 yields most accuracy for this test case (a nonlinear viscous Burgers equation model). Lin et. al [13] compared limiters 3, 4, 5, 6 on a linear advection problem using a rectangular pulse and conclude that limiter 4 provides the largest implicit diffusion among all the limiters.

To further investigate the performance of these limiters on a model problem with no viscosity, we tested them using the following inviscid nonlinear Burgers equation,

$$\frac{\partial\phi}{\partial t} + \frac{\partial}{\partial x}\left(\frac{\phi^2}{2}\right) = 0.$$

With the following initial condition (on the whole real line) [47],

$$\phi(x, 0) = \begin{cases} 0, & x < -1 \\ \frac{1}{2}, & -1 < x < 0 \\ 0, & x > 0. \end{cases} \quad (18)$$

The solution develops into a shock and an expansion fan (for details of the solution, see [47]), analytically given by (for $t \leq 4$, i.e., before the expansion fan meets the shock),

$$\phi(x, t) = \begin{cases} 0, & x < -1 \\ \frac{x+1}{t}, & -1 < x < \frac{t}{2} - 1 \\ \frac{1}{2}, & \frac{t}{2} - 1 < x < \frac{t}{4} \\ 0, & x > \frac{t}{4}. \end{cases} \quad (19)$$

In figures (7)- (10) we have plotted the exact along with the numerical solutions obtained using the various numerical schemes (MUSCL schemes, with limiters 1- 6, and the PPM) at $t = 2$. The numerical solutions have been computed using a resolution of $N_x = 80$ grid cells ($\Delta x = \frac{4}{N_x}$) and a time step (Δt) given by the CFL criteria, $CFL = \frac{U\Delta t}{\Delta x}$, where the CFL number was assigned a value of 0.1 and $U = 0.5$.

Once again, the first order accurate scheme (limiter 1) is diffusive. The solutions obtained by using limiter 2 (simplest second order scheme) and limiter 3 (simple positive definite scheme), both over shoot, indicating that there is a lack of (implicit) viscosity, though the solution obtained by using limiter 4 (monotonicity preserving scheme) does not suffer from such problems, it is diffusive, when compared to the computed solutions using limiters 5, 6 (local and global min./ max. slope limited schemes respectively) and the PPM scheme.

We thus conclude that the slope limiters 5, 6 and the PPM scheme generate satisfactory solutions for the inviscid Burgers equation (recall that the limiter 6 needs specification of the global min. and max. of the solution, for more complicated models and fluid flows, these global min./ max. values may not be available a priori).

4. DERIVATION AND VERIFICATION OF THE ADJOINT AND TANGENT LINEAR MODELS

This section details the derivation of the development of the adjoint method, used to obtain the gradient of the cost functional with respect to the control parameters efficiently, closely following [53].

The following form of the cost functional is considered,

$$\mathcal{J}(\mathbf{x}) = \frac{1}{2} \sum_{k=0}^n (\mathbf{x}(t_k) - \mathbf{x}^{obs}(t_k))^T W(t_k) (\mathbf{x}(t_k) - \mathbf{x}^{obs}(t_k)), \quad (20)$$

where $t \in [t_0, t_n]$ is the (data) assimilation time window comprised of n time steps, $W(t_k)$ is a diagonal weighting matrix, $\mathbf{x}(t_k)$ is the evolving state vector and $\mathbf{x}^{obs}(t_k)$ is another (evolving) vector, which is made up of the observations that are distributed in space and time.

The above convex cost functional is minimized using a robust unconstrained minimization method described in section 4. The directional derivative of the above cost functional, in the direction of $\delta\mathbf{x}$ is given by

$$\lim_{\|\delta\mathbf{x}\| \rightarrow 0} \delta\mathcal{J} = (\nabla_{\mathbf{x}}\mathcal{J})^T \delta\mathbf{x}. \quad (21)$$

From equation (20),

$$\delta\mathcal{J}(\mathbf{x}) = \sum_{k=0}^n (W(t_k) (\mathbf{x}(t_k) - \mathbf{x}^{obs}(t_k)))^T \delta\mathbf{x}(t_k), \quad (22)$$

where $\delta\mathbf{x}(t_k)$ is the perturbation of the state vector obtained from the perturbation of the model parameters, \mathbf{x} . Using the above two equations,

$$(\nabla_{\mathbf{x}}\mathcal{J})^T\delta\mathbf{x} = \sum_{k=0}^n (W(t_k)(\mathbf{x}(t_k) - \mathbf{x}^{obs}(t_k)))^T\delta\mathbf{x}(t_k). \quad (23)$$

The evolution of the state vector using the nonlinear model can be symbolically written as

$$\mathbf{x}(t_{k+1}) = F(\mathbf{x}(t_k)).$$

Linearizing the model about the current model solution, we obtain the equation for the evolution of perturbations,

$$\delta\mathbf{x}(t_{k+1}) = \frac{\partial F(\mathbf{x}(t_k))}{\partial\mathbf{x}}\delta\mathbf{x}. \quad (24)$$

Let $L(t)$ represent the Jacobian, $\frac{\partial F(\mathbf{x}(t_k))}{\partial\mathbf{x}}$, then we can rewrite the above equation as,

$$\begin{aligned} \delta\mathbf{x}(t_k) &= L(t_k - \Delta t)\delta\mathbf{x}(t_k - \Delta t) \\ &= L(t_k - \Delta t)L(t_k - 2\Delta t)\delta\mathbf{x}(t_k - 2\Delta t) \\ &= L(t_k - \Delta t)L(t_k - 2\Delta t)L(t_k - 3\Delta t)\delta\mathbf{x}(t_k - 3\Delta t) \\ &= \dots \\ &= M_k\delta\mathbf{x}, \end{aligned} \quad (25)$$

Thus $\delta\mathbf{x}(t_k) = M_k\delta\mathbf{x}$, where M_k represents the application of all the linear operators to obtain $\delta\mathbf{x}(t_k)$.

Using equations (23) and (25) the gradient of the cost functional with respect to the control parameters, \mathbf{x} is given by

$$\nabla_{\mathbf{x}}\mathcal{J} = \sum_{k=0}^n M_k^T W(t_k)(\mathbf{x}(t_k) - \mathbf{x}^{obs}(t_k)). \quad (26)$$

Using $W(t_k)(\mathbf{x}(t_k) - \mathbf{x}^{obs}(t_k))$ as the (initial) values of the adjoint variables $\mathbf{x}^*(t_k)$ at time

t_k , the adjoint equations

$$\mathbf{x}^*(t_0) = M_k^T(\mathbf{x}^*(t_k)), \quad (27)$$

are integrated backwards in time to obtain the values of the adjoint variables at initial time, t_0 . Therefore,

$$\nabla_{\mathbf{x}} \mathcal{J} = \sum_{k=n}^0 \mathbf{x}^*(t_k). \quad (28)$$

Now we will briefly describe the method of programming the adjoint model, in other words, equations (25) and (27) and implementation of (28).

4.1. Coding the Adjoint model

We follow the approach of first discretize and then differentiate, see [54] and [22] for details. Discrete numerical operations in the nonlinear forward model having unique corresponding operations in the (backward evolving) adjoint model. The linear equation (24) is now onwards referred to as the Tangent Linear Model (TLM). The TLM code is programmed by linearizing line by line, the nonlinear forward model code. Following equation (25), the TLM can be formally viewed as a result of multiplying linear operators: $M_k = L_1, L_2, \dots, L_k$, where each of the L_k is either a DO-loop or a subroutine. Then the adjoint model, M_k^T is a product of the (adjoint) linear operators, $L_1^T, L_2^T, \dots, L_k^T$. Hence the adjoint model is the *transpose* of the TLM. This relationship is used to write the adjoint model code, using the TLM code (see [55] and [24] for details), and to verify the same for the transposition property (all our subroutines satisfactorily passed this test). We have used TAMC [56] (an automatic differentiation software) to help us derive the TLM and adjoint model codes; however, we would like to emphasize that sufficient caution must be taken while differentiating functions such as the ABS (absolute value function), SIGN (signum function), DIM (dimension function), MIN

and MAX (minimum and maximum functions respectively), these functions frequently arise due to the nature of the formulation of the various slope limiters, such as limiters 3, 5 and 6 (section 2.1). In appendix A, we provide a segment of our FORTRAN code which illustrates the differentiation of the MIN function.

To adjoint model is integrated backwards in time to obtain the gradient of the cost functional, $\nabla_{\mathbf{x}}\mathcal{J}$ in the following sequence of three steps,

1. Integrate the adjoint model backwards in time, from time step t_k to t_0 with zero final conditions for the adjoint variables \mathbf{x}^* .
2. The *forcing term* $W(t_k)(\mathbf{x}(t_k) - \mathbf{x}^{obs}(t_k))$ is added to the value of adjoint variables whenever time t_k ($k = 1, 2, \dots, n$) is reached.
3. Finally at t_0 the value of adjoint variables equals the gradient of the cost functional with respect to the control variables.

Using the Taylor series expansion of the cost functional, upto first order,

$$\mathcal{J}(\mathbf{x} + \eta\nabla\mathcal{J}) = \mathcal{J}(\mathbf{x}) + \eta(\nabla\mathcal{J})^T\nabla\mathcal{J} + O(\eta^2), \quad (29)$$

where η is a scalar and the gradient, $\nabla\mathcal{J} = \nabla_{\mathbf{x}}\mathcal{J}$, is obtained by using the adjoint model. We can rewrite the above equation as in [54],

$$\Psi(\eta) = \frac{\mathcal{J}(\mathbf{x} + \eta\nabla\mathcal{J}) - \mathcal{J}(\mathbf{x})}{\eta\nabla\mathcal{J}^T\nabla\mathcal{J}} = 1 + O(\eta) \quad (30)$$

Therefore, the gradient provided by the adjoint model is assumed to be accurate up-to machine accuracy if $\lim_{\eta \rightarrow 0} |\Psi(\eta)| = 1.0$. The truncation errors dominate for $\eta > 10^{-3}$, whereas for η near machine epsilon, roundoff errors accumulate. In figure (11) we have plotted $\Psi(\eta)$ versus η to show the satisfactory performance of our adjoint model for the PPM advection scheme case (the results with other advection schemes are similar).

5. MINIMIZATION

We used an unconstrained limited memory Quasi-Newton (L-BFGS) minimization algorithm [48, 49] (available for download at www.netlib.org/opt/lbfgs_um.shar) for minimization of the cost functional $\mathcal{J} = \mathcal{J}(\mathbf{x}_k)$, where \mathbf{x}_k is the n component (control) vector at the k^{th} iteration. $\mathbf{g}_k = \mathbf{g}(\mathbf{x}_k) = \nabla \mathcal{J}_k$ is the gradient vector of size n , and $H_k = \nabla^2 \mathcal{J}_k$ is the $n \times n$ symmetric Hessian matrix of the second partial derivatives of \mathcal{J} with respect to the control vector. The new iterate is given by,

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k, \quad (31)$$

where \mathbf{p}_k is the descent direction (for instance, $\mathbf{p}_k = -\mathbf{g}_k$ for the steepest descent method and $\mathbf{p}_k = -H_k^{-1} \mathbf{g}_k$ for the quasi-Newton methods), and α_k is the step length.

Iterations are terminated when (using the L_2 norm)

$$\|\mathbf{g}_k\| < EPS \cdot MAX(1, \|\mathbf{x}_k\|).$$

Here we specified $EPS = 10^{-5}$ as our termination criteria.

Given a sequence of two successive iterates, \mathbf{x}_{k+1} and \mathbf{x}_k , $\mathbf{g}_k = \nabla \mathcal{J}_k$ and $\mathbf{g}_{k+1} = \nabla \mathcal{J}_{k+1}$. Then $\mathbf{g}_{k+1} - \mathbf{g}_k = H_k \mathbf{p}_k$ which can be rewritten as $\mathbf{q}_k = H_k \mathbf{p}_k$. If the Hessian is constant, then $\mathbf{q}_k = H \mathbf{p}_k$, and we can write the following quasi-Newton condition for $0 \leq i \leq k$,

$$H_{k+1}^{-1} \mathbf{q}_i = \mathbf{p}_i$$

In general, the evaluation of the Hessian matrix is impractical and costly. Quasi-Newton methods use an approximation of the inverse Hessian matrix. We start with an identity matrix and then iteratively, a better approximation to the inverse Hessian matrix is built up, in such a way that H_k preserves positive definiteness and symmetry.

The Broyden-Fletcher-Goldfarb-Shanno (BFGS) update formula for the B_{k+1} (i.e, H_{k+1}^{-1}) is given by,

$$B_{k+1} = B_k + \frac{(1 + \mathbf{q}_k^T B_k \mathbf{q}_k) \mathbf{p}_k \mathbf{p}_k^T}{\mathbf{q}_k^T \mathbf{p}_k} - \frac{\mathbf{p}_k \mathbf{q}_k^T B_k + B_k \mathbf{q}_k \mathbf{p}_k^T}{\mathbf{q}_k^T \mathbf{p}_k}, \quad (32)$$

this is a symmetric rank two update, constructed using the vectors \mathbf{p}_k and $B_k \mathbf{q}_k$. Thus each minimization iteration proceeds by first checking for termination criteria, finding the direction of descent: \mathbf{p}_k (using the approximation to the inverse Hessian matrix), find an optimal step length (α_k) in the direction of \mathbf{p}_k , and finally using equation (31) find the next \mathbf{x}_{k+1} . The limited memory version, L-BFGS is an adaptation of the BFGS algorithm to large problems, achieved by changing the above Hessian update formula, see for details [48, 49], [50] and [51, 52] for applications.

6. DATA ASSIMILATION EXPERIMENTS

This section describes results obtained using the adjoint model described in the previous section in order to conduct data assimilation for retrieval of initial conditions as control variables. Following work of Vukićević et al. [21], we have consistently used the same advection scheme both in the nonlinear forward and adjoint models. In a twin experiment framework, our goal is to minimize the cost functional given in equation (20), namely

$$\mathcal{J}(\mathbf{x}) = \frac{1}{2} \sum_{k=0}^n (\mathbf{x}(t_k) - \mathbf{x}^{obs}(t_k))^T W(t_k) (\mathbf{x}(t_k) - \mathbf{x}^{obs}(t_k)),$$

with respect to the initial state $\mathbf{x}(t_0)$ as the control parameter and we have prescribed $W(t_k) \equiv I$, i.e. the identity matrix. In our twin experiments, we used the initial condition given in (16), run the forward model up-to time step t_k to obtain the observations, $\mathbf{x}^{obs}(t_k)$.

The initial condition (16) is then randomly perturbed,

$$\phi^{pert}(x, 0) = \phi(x, 0) + \epsilon \cdot RAND \cdot \phi(x, 0), \quad (33)$$

where ϵ has been assigned a value of 0.01 and $RAND$ is a pseudo random number, such that $RAND \in [-0.5, 0.5]$.

The above perturbed initial condition is used as a first guess to minimize the cost functional \mathcal{J} , and to integrate the nonlinear model to t_k , which yields $\mathbf{x}(t_k)$. Thus the goal is to recover the unperturbed initial condition for ϕ (from now onwards denoted by ϕ^{rec}), which is close to $\phi(x, 0)$ at the conclusion of the minimization process (backward integration of the adjoint model yields the gradient of the cost functional with respect to the initial conditions). An assimilation time window of $[0, 2.0]$ has been used. The same discretization, in space and time, which was used in section 3 to test and compare the different schemes for the smooth test case (with exact solution (17)), is used here as well.

Figures (12) - (18) show the variation of the cost functional and gradient norm (L_2 norm) versus the number of iterations and in table (II) we compare the values of ϕ^{rec} with $\phi(x, 0)$ for different advection schemes (limiters 1- 6 and the PPM). The cost functional has been successfully reduced by about nine orders of magnitude, whereas the gradient norm was reduced by about five orders of magnitude (in section 4, we described the termination criteria for the minimization process) for all the cases, except when limiter 3 (simple positive definite scheme) was used. The fact that all of these schemes achieve the same convergence criteria for successful termination in about 45-50 minimization iterations (limiter 1: 49 iterations, limiter 4: 47, limiter 5: 52, limiter 6: 46), except for limiter 2 (simplest second order scheme) and PPM scheme, which take 58 and 65 iterations respectively indicates that the approximation to the Hessian matrix that is constructed by the L-BFGS minimization algorithm does not differ

from one advection scheme to the other (the spectrum of the eigenvalues of the Hessian matrix influences the minimization process [50]). It is to be noted that though the PPM scheme is well known to be a very accurate scheme (third order accurate), it requires more CPU time when compared to that required by other schemes (both in forward and adjoint modes, since the adjoint model performs forward computations as well, this problem becomes compounded). As indicated by table (II), after 65 minimization iterations (highest when compared to all other MUSCL limiters), the ϕ^{rec} obtained is not as good as the one obtained, for example by using the monotonicity preserving van Leer limiter (limiter 4).

In order to run the adjoint model for the case of limiter 3, we require to specify the value of the adjoint variable corresponding to Φ^{min} in equation (7). In the forward model, we assigned $\Phi^{min} = 0$, we used the same value of zero as a reasonable guess even the adjoint mode. Following a similiary argument, for the case with limiter 6 (global min./ max. limiter), we assigned a value of zero to the adjoint variable corresponding to Φ^{min} , and as a guess value for the adjoint variable corresponding to Φ^{max} we assigned value of unity (once again we would like to emphasize that assigned such values to the adjoint variables a priori is not trivial). Based on the results obtained, the minimization process partially failed for the case of limiter 3, yet it succeeded for limiter 6. The numerical solution obtained using the forward model for limiter 3 (figure (4)) did not capture the peak values of the solution correctly, since this scheme has more implicit diffusion than the first order scheme, it is not due to the lack of implicit diffusion that such an undershoot occurred. We conclude that the strategy of limiting the slopes based on only the minimum value of the distribution (which is not known a priori), or in other words, guaranteeing only positive-definiteness is not a sufficient criteria for success (both for forward and adjoint models).

The first order scheme (limiter 1) yields the closest ϕ^{rec} (to $\phi(x, 0)$), yet based on number of iterations taken during the minimization process, as well as the forward model results (section 3), we conclude that for this viscous nonlinear Burgers equation data assimilation model problem, the monotonicity preserving van Leer limiter performs the best, when compared to all other limiters and PPM schemes.

We would like to mention that limiter 3 (simple positive definite scheme), the local and global min./ max. (limiters 5 and 6 respectively) slope limited and PPM schemes all have *switches*, in other words, involve computation of min. and (or) max. of certain variables to evaluate the slope limiter (see equations (7), (9) and (11)). Programming these switches in the adjoint model proves to be a very tedious and time consuming task. We believe that due to presence of these switches (which introduce *artificial* discontinuities) in the PPM scheme, the optimal step length calculation by the minimization algorithm fails for gradient norm less than 10^{-4} in magnitude, which explains why the initial condition could not be very well recovered by this scheme. Following the work of Thuburn and Haine [26], for a non-smooth solution case (unlike a smooth solution case considered here), serious modifications to these schemes should be required for their satisfactory performance in adjoint models.

7. SUMMARY AND CONCLUSIONS

We have studied the impact of various high resolution TVD, FV (which use MUSCL slope limiters and PPM) schemes on data assimilation for a nonlinear model problem, namely the viscous Burgers equation, which has a smooth solution (section 3). To the best of our knowledge, the PPM scheme has not been used for data assimilation in adjoint model in any previous research work, thus-far. Based on the data assimilation experiments, we suggest that

limiter 4 (monotonicity preserving van Leer limiter) yields better results, when compared to all other schemes.

Following these results obtained here, we want to further investigate validity of the above findings for a higher dimensional system. In particular, investigate the performance of these schemes to assimilate data (in a twin experiment, i.e. model generated data and gathered real data) using variational data assimilation and ensemble Kalman filtering using the finite volume shallow water model of Lin and Rood [17]. The adjoint model required for this purpose has been developed by the authors, see the *Documentation of the TLM and adjoint models of the Lin-Rood spherical shallow water finite volume model* (<http://www.csit.fsu.edu/navon/publ.html>).

ACKNOWLEDGEMENTS

The authors acknowledge the support from the NSF grant number ATM-9731472 managed by Dr. Linda Peng whom we would like to thank, for her support.

8. APPENDIX A

In this appendix, we illustrate the differentiation of functions which require special care, such as the ABS, SIGN, DIM, MIN, MAX functions etc. Following is an example which shows a section from the forward code to obtain

```
phi_local_min = MIN(phi_old(i-1),phi_old(i),phi_old(i+1))
```

is rewritten as:

```
IF(phi_old(i-1) .LE. phi_old(i))THEN
  IF(phi_old(i-1) .LE. phi_old(i+1))THEN
    phi_local_min = phi_old(i-1)
  ELSE
    phi_local_min = phi_old(i+1)
  END IF
ELSE IF(phi_old(i) .LE. phi_old(i+1))THEN
  phi_local_min = phi_old(i)
ELSE
  phi_local_min = phi_old(i+1)
END IF
```

The linearization of the above segment is give by,

```
if (phi_old(i-1) .le. phi_old(i)) then
  if (phi_old(i-1) .le. phi_old(i+1)) then
    g_phi_local_min = g_phi_old(i-1)
    phi_local_min = phi_old(i-1)
  else
    g_phi_local_min = g_phi_old(i+1)
    phi_local_min = phi_old(i+1)
```

```

endif
else if (phi_old(i) .le. phi_old(i+1)) then
  g_phi_local_min = g_phi_old(i)
  phi_local_min = phi_old(i)
else
  g_phi_local_min = g_phi_old(i+1)
  phi_local_min = phi_old(i+1)
endif

```

the corresponding adjoint statements are as following,

```

if (phi_old(i-1) .le. phi_old(i)) then
  if (phi_old(i-1) .le. phi_old(i+1)) then
    adphi_old(i-1) = adphi_old(i-1)+adphi_local_min
    adphi_local_min = 0.d0
  else
    adphi_old(i+1) = adphi_old(i+1)+adphi_local_min
    adphi_local_min = 0.d0
  endif
else if (phi_old(i) .le. phi_old(i+1)) then
  adphi_old(i) = adphi_old(i)+adphi_local_min
  adphi_local_min = 0.d0
else
  adphi_old(i+1) = adphi_old(i+1)+adphi_local_min
  adphi_local_min = 0.d0
endif

```

It is to be noted that in order to compute the adjoint variables in the backward direction, we require that the forward states be available (as evident from the above piece of adjoint code) in memory or recompute them, see research on checkpointing [57, 58] for discussion on the trade-off between storing

in memory and recomputation.

REFERENCES

1. Thomas JW. *Numerical Partial Differential Equations: Finite Difference Methods*. Springer, 1998.
2. Thomas JW. *Numerical Partial Differential Equations : Conservation Laws and Elliptic Equations* Springer, 1999.
3. Le Veque R.J. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, 2002.
4. Zienkiewicz OC, Taylor RL. *Finite Element Method: Volume 1, The Basis*. (5th edn) Butterworth-Heinemann, 2000.
5. Zienkiewicz OC, Taylor RL. *Finite Element Method: Volume 2, Solid Mechanics*. (5th edn) John Wiley & Sons, 2000.
6. Zienkiewicz OC, Taylor RL. *Finite Element Method: Volume 3, Fluid Mechanics*. (5th edn) Butterworth-Heinemann, 2000.
7. Canuto C, Hussaini MY, Quarteroni A, Zang TA. *Spectral Methods in Fluid Dynamics*. (3rd edn) Springer, 1991.
8. Le Veque R.J. *Numerical Methods for Conservation Laws*. Birkhäuser, 1990.
9. Laney CB. *Computational Gasdynamics*. Cambridge University Press, 1998.
10. Durran DR. *Numerical Methods for Wave Equations in Geophysical Fluid Dynamics*. Springer-Verlag: New-York, 1999.
11. Zhang S, Zou X, Ahlquist J, Navon IM. Use of differentiable and nondifferentiable optimization algorithms for variational data assimilation with discontinuous cost functions. *Monthly Weather Review* 2000; **128**(12): 4031–4044.
12. Rood RB. Numerical advection algorithms and their role in atmospheric transport and chemistry models. *Reviews of Geophysics* 1987; **25**(1): 71–100.
13. Lin S-J, Chao WC, Sud YC, Walker GK. A class of the van Leer transport schemes and its applications to the moisture transport in a general circulation model. *Monthly Weather Review* 1994; **122**:1575–1593.
14. Lin S-J, Rood RB. Multidimensional flux-form semi-Lagrangian transport schemes,. *Monthly Weather Review* 1996; **124**: 2046–2070.
15. van Leer B. Towards the ultimate conservative difference scheme IV. A new approach to numerical convection. *Journal of Computational Physics* 1977; **23**:276–299.
16. Lin S-J. A vertically Lagrangian finite-volume dynamical core for global models. *Monthly Weather Review* 2004; **132**: 2293–2307.

17. Lin S-J, Rood RB. An explicit flux-form semi- Lagrangian shallow-water model on the sphere. *Q. J. R. Meteorological Society* 1997; **123**: 2477–2498.
18. Smolarkiewicz PK. A fully multidimensional positive definite advection algorithm with small implicit diffusion. *Journal of Computational Physics* 1984; **54**: 325–362.
19. Leonard BP. A stable and accurate convective modeling procedure based on quadratic upstream interpolation. *Computer Methods in Applied Mechanics and Engineering* 1979; **19**: 59–98.
20. Leonard BP. The ULTIMATE conservative difference scheme applied to unsteady one-dimensional advection. *Computer Methods in Applied Mechanics and Engineering* 1991; **88**(1): 17–74.
21. Vukićević T, Steyskal M, Hecht M. Properties of advection algorithms in the context of variational data assimilation. *Monthly Weather Review* 2001; **129**(5): 1221–1231.
22. Gunzburger MD. *Perspectives in flow control and optimization*. SIAM: Philadelphia, 2003.
23. Cacuci DG *Sensitivity and uncertainty analysis*. Chapman & Hall/CRC, 2003.
24. Kalnay E *Atmospheric Modeling, Data Assimilation and Predictability*. Cambridge University Press, 2002.
25. Mohammadi B, Pironneau O. *Applied Shape Optimization for Fluids*. Oxford University Press, 2001.
26. Thuburn J, Haine TWN. Adjoint of nonoscillatory advection schemes. *Journal of Computational Physics* 2001; **171**: 616–631.
27. Harten A. High resolution schemes for hyperbolic conservations laws. *Journal of Computational Physics* 1983; **49**:357–393.
28. Burgers JM. A mathematical model illustrating the theory of turbulence. In *Advances in Applied Mechanics*, Academic Press: New York, 1948; **1**:171–199.
29. Whitham G. *Linear and Nonlinear Waves*. Wiley-Interscience, 1974.
30. Zhang DS, Wei GW, Kouri DJ, Hoffman DK. Burgers' equations with high Reynolds number. *Physics of Fluids* 1997; **9**(6):1853–1855.
31. Fletcher CAJ *Computational techniques for fluid dynamics* Springer-Verlag: Berlin, 1988.
32. van Leer B. Towards the ultimate conservative difference scheme I. The quest for monotonicity. In *Springer Lecture Notes in Physics*, Springer, 1973; **18**:163–168.
33. van Leer B. Towards the ultimate conservative difference scheme II. Monotonicity and conservation combined in a second order scheme. *Journal of Computational Physics* 1974; **14**:361–370.
34. van Leer B. Towards the ultimate conservative difference scheme III. Upstream-centered finite difference schemes for ideal compressible flow. *Journal of Computational Physics* 1977; **23**:263–275.
35. Osher S. Convergence of generalized MUSCL schemes. *SIAM Journal of Numerical Analysis* 1985; **22**:

- 947–961.
36. Harten A. On a class of high resolution total variation stable finite difference schemes. *SIAM Journal of Numerical Analysis* 1984; **21**: 1–23.
 37. Harten A, Osher S. Uniformly high-order accurate nonoscillatory schemes. I. *SIAM Journal of Numerical Analysis* 1987; **24**: 279–309.
 38. Harten A, Engquist B, Osher S, Chakravarthy SR. Uniformly high-order accurate essentially non-oscillatory schemes, III. *Journal of Computational Physics* 1987; **71**: 231–303.
 39. Harten A. ENO schemes with subcell resolution. *Journal of Computational Physics* 1987; **83**: 148–184.
 40. Harten A, Lax PD, van Leer B. On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *SIAM Review* 1983; **25**(1): 35–61.
 41. Colella P, Woodward P. The piecewise-parabolic method (PPM) for gas-dynamical simulations. *Journal of Computational Physics* 1984; **54**: 174–201.
 42. Woodward PR, Colella P. The numerical simulation of two-dimensional fluid flow with strong shocks. *Journal of Computational Physics* 1984; **54**: 115–173.
 43. Shu C-W. Total-variation-diminishing time discretizations. *SIAM Journal of Scientific Statistical Computation* 1988; **9**: 1073–1084.
 44. Shu C-W, Osher S. Efficient implementation of essentially non-oscillatory shock capturing schemes. *Journal of Computational Physics* 1988; **77**: 439–471.
 45. Gottlieb S, Shu C-W. Total variation diminishing Runge-Kutta schemes. *Mathematics of Computation* 1998; **67**(221): 73–85.
 46. Benton ER, Platzman GW. A table of solutions of the one-dimensional Burgers equation. *Quarterly of Applied Mathematics* 1972; **30**: 195–212.
 47. Numerical schemes applied to the Burgers and Buckley-Leverett equations.
http://www.extra.rdg.ac.uk/Maths/Research/Publications/Msc_dissertations/rakhib_ahmed.pdf
 [30 March 2005].
 48. Liu DC, Nocedal J. On the limited memory BFGS method for large scale minimization. *Mathematical Programming* 1989; **45**: 503–528.
 49. Nash SG, Nocedal J. A numerical study of the limited memory BFGS method and the truncated-Newton method for large scale optimization. *SIAM Journal on Optimization* 1991; **1**: 358–372
 50. Nocedal J, Wright SJ. *Numerical Optimization*. Springer: New-York, 1999.
 51. Daescu D, Navon IM. An analysis of a hybrid optimization method for variational data assimilation.

- International Journal of Computational Fluid Dynamics* 2003; **17**(4): 231–233.
52. Zou X, Navon IM, Berger M, Phua MK, Schlick T, LeDimet FX. Numerical experience with limited-memory, quasi-Newton methods for large-scale unconstrained nonlinear minimization. *SIAM Journal on Optimization* 1993; **3**(3): 582–608.
53. Homescu C, Navon IM, Li Z. Suppression of vortex shedding for flow around a circular cylinder using optimal control. *International Journal for Numerical Methods in Fluids* 2002; **38**: 43–69.
54. Navon IM, Zou X, Derber J, Sela J. Variational data assimilation with an adiabatic version of the NMC spectral model. *Monthly Weather Review* 1992; **120**: 1433–1446.
55. Yang W, Navon IM. Documentation of the tangent linear model and its adjoint of the adiabatic version of the NASA GEOS-1 C-grid GCM (Version 5.2). NASA Technical Memorandum 104606. Edited by Max Suarez. *NASA Technical Memorandum Report Series on Global Modeling and Data Assimilation* 1996.
56. Tangent and Adjoint Model Compiler (TAMC).
<http://www.autodiff.com/tamc>.
[3 April 2005].
57. Restrepo JM, Leaf GK, Griewank A. Circumventing storage limitations in variational data assimilation studies. *SIAM Journal on Scientific Computing* 1998; **19**(5): 1586–1605.
58. Griewank A, Walther A. An implementation of checkpointing for the reverse or adjoint mode of computational differentiation. *ACM Transactions on Mathematical Software* 2000; **26**(1): 19–45.

Table I. Errors in L_2 and L_∞ norms for different numerical schemes in forward mode, with $\Delta t =$

$$1.5708 \times 10^{-4} \text{ at } t = 1$$

 L_2 - Error

N_x	limiter 1	limiter 2	limiter 3	limiter 4	limiter 5	limiter 6	PPM
40	3.0359×10^{-2}	1.6845×10^{-3}	1.6845×10^{-3}	1.4700×10^{-3}	1.6296×10^{-3}	1.6844×10^{-3}	1.9321×10^{-3}
80	2.2498×10^{-2}	6.4579×10^{-4}	6.4578×10^{-4}	6.0083×10^{-4}	6.3581×10^{-4}	6.4578×10^{-4}	7.1212×10^{-4}
160	1.6221×10^{-2}	2.7148×10^{-4}	2.7148×10^{-4}	2.6128×10^{-4}	2.6851×10^{-4}	2.7148×10^{-4}	3.1819×10^{-4}

 L_∞ - Error

N_x	limiter 1	limiter 2	limiter 3	limiter 4	limiter 5	limiter 6	PPM
40	7.8244×10^{-3}	5.1272×10^{-4}	5.1272×10^{-4}	4.6731×10^{-4}	4.9658×10^{-4}	5.1271×10^{-4}	5.7239×10^{-4}
80	4.1003×10^{-3}	1.3852×10^{-4}	1.3852×10^{-4}	1.3341×10^{-4}	1.3645×10^{-4}	1.3852×10^{-4}	1.4500×10^{-4}
160	2.0900×10^{-3}	4.0477×10^{-5}	4.0477×10^{-5}	3.9767×10^{-5}	4.0172×10^{-5}	4.0477×10^{-5}	4.2992×10^{-5}

Table II. Comparison of the ϕ^{rec} for different advection schemes based on data assimilation experiments, $\|\phi^{pert}(x, 0) - \phi(x, 0)\|_2 = 1.3004 \times 10^{-2}$ for all the schemes.

Advection Scheme	$\ \phi^{rec} - \phi(x, 0)\ _2$
Limiter 1	3.0662×10^{-6}
Limiter 2	4.5988×10^{-6}
Limiter 3	5.0651×10^{-3}
Limiter 4	3.2596×10^{-6}
Limiter 5	4.1360×10^{-6}
Limiter 6	3.3876×10^{-6}
PPM	1.3140×10^{-5}

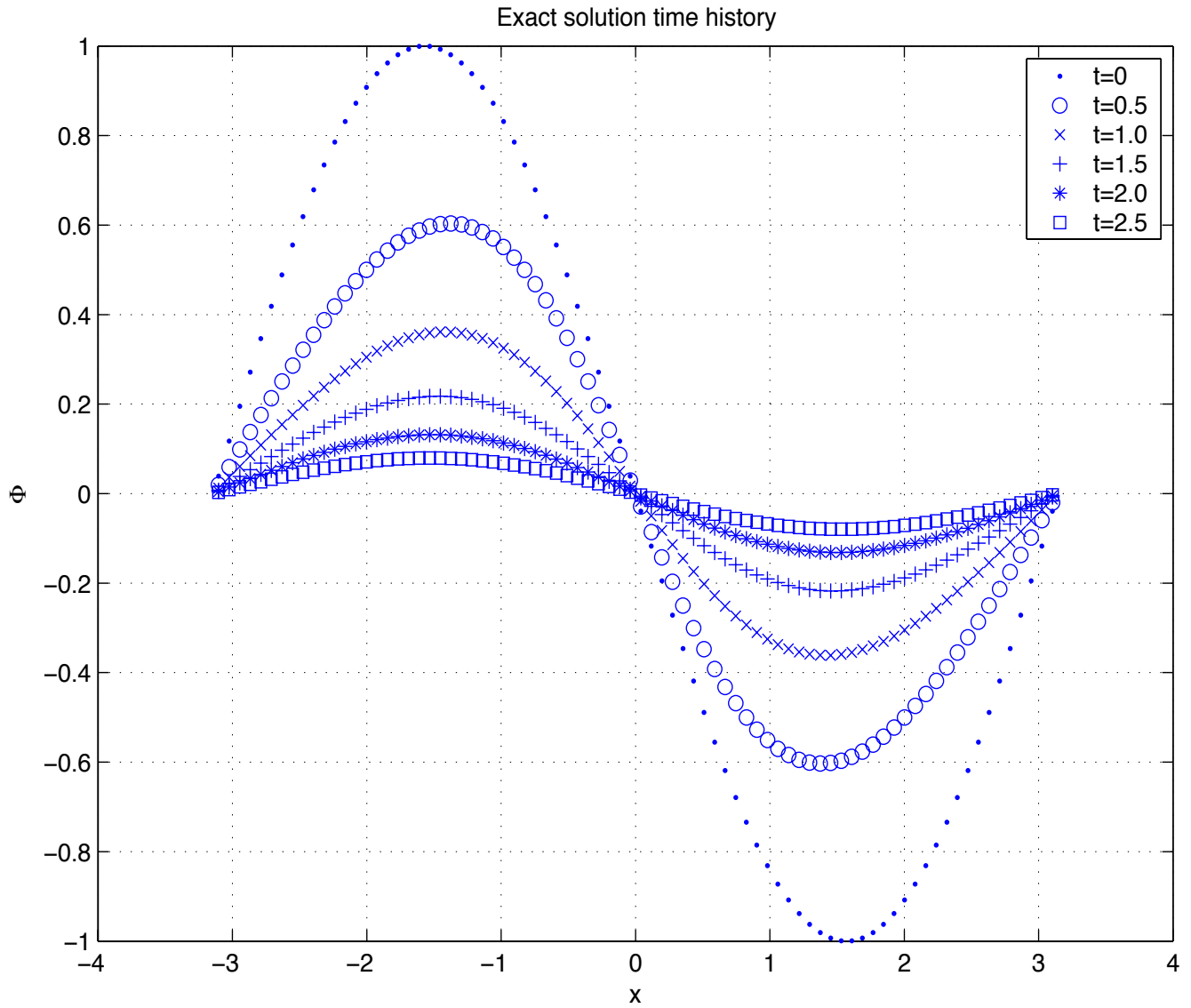


Figure 1. Exact solution at $t = 0, 0.5, 1.0, 1.5, 2.0, 2.5$

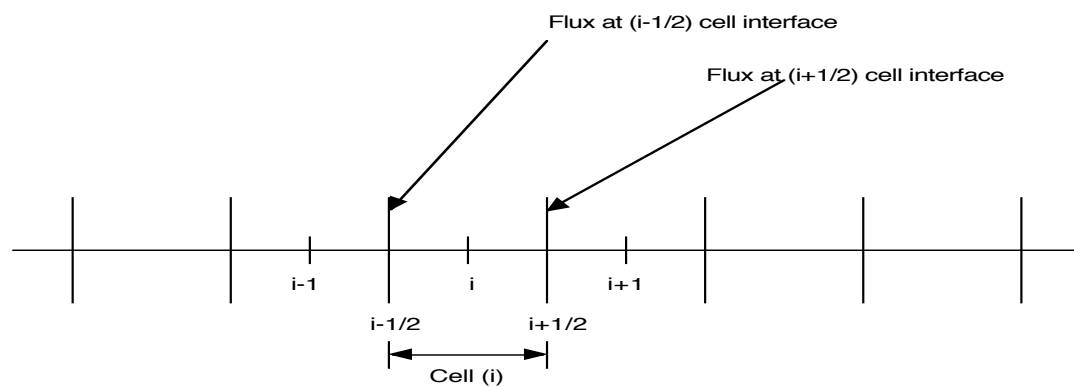


Figure 2. Finite volume discretization

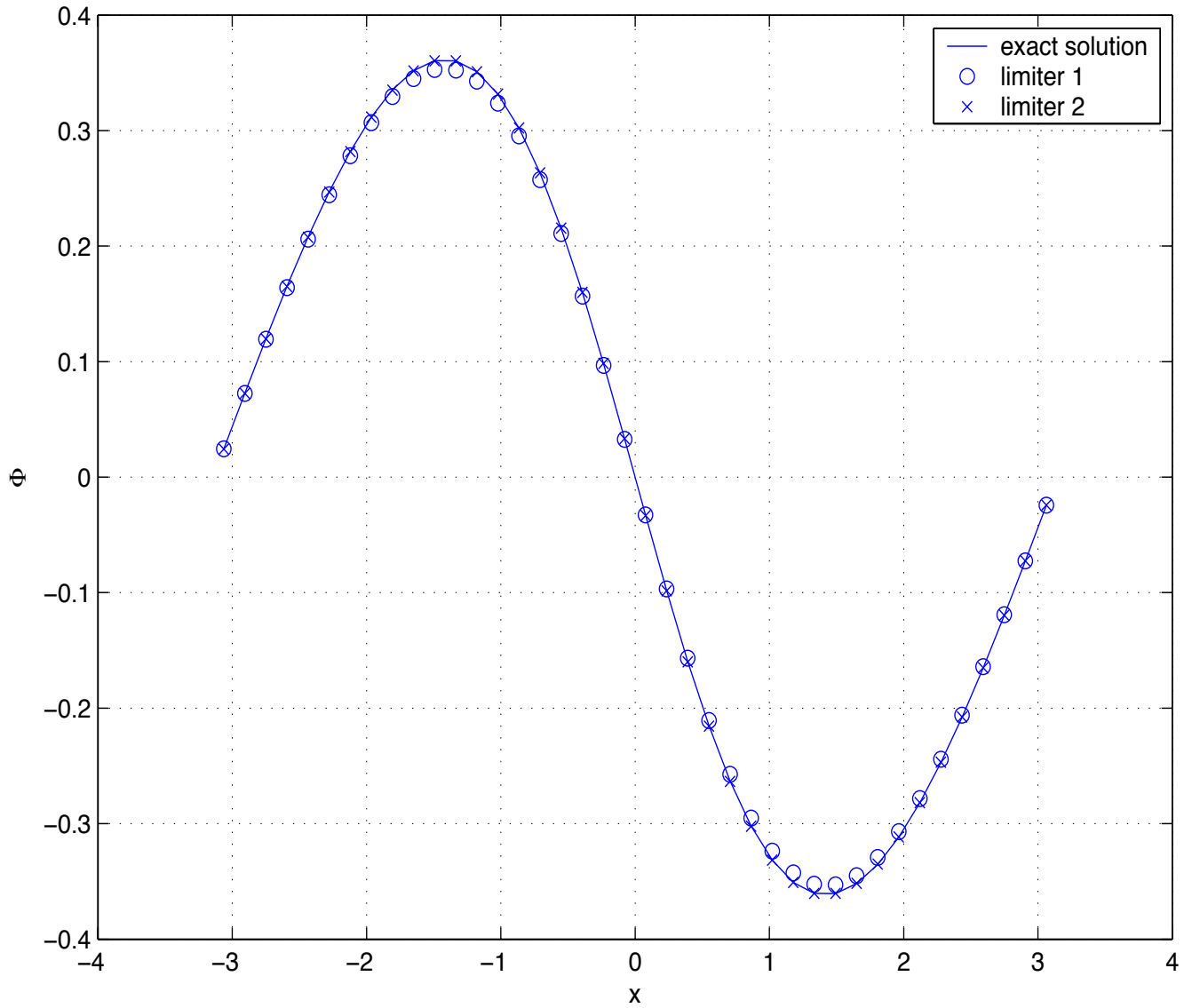


Figure 3. Exact and numerical solutions (in forward mode) of the 1-D nonlinear viscous Burgers equation with slope limiters 1 and 2 at $t = 1$.

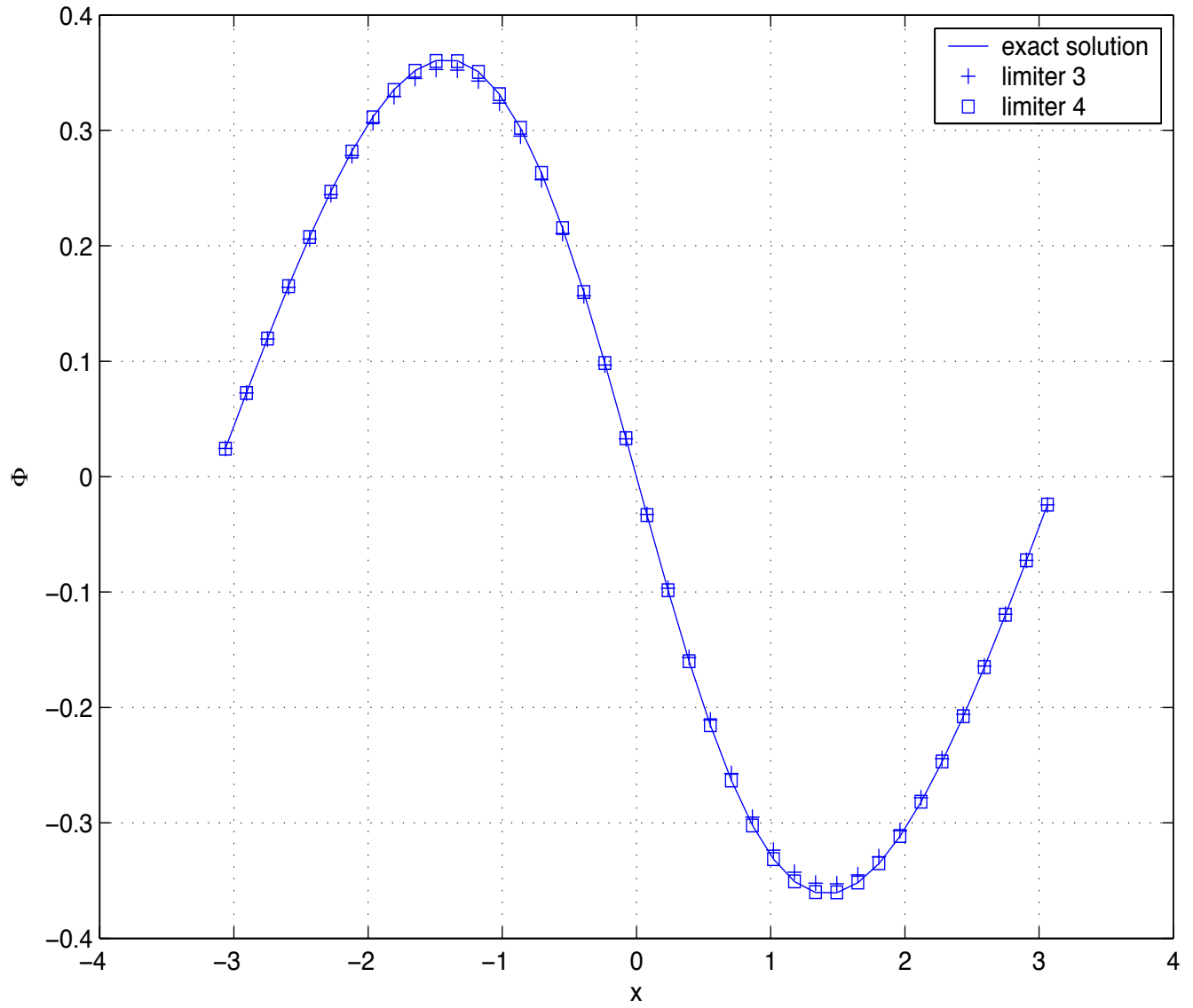


Figure 4. Same as in figure (3), but with limiters 3 and 4.

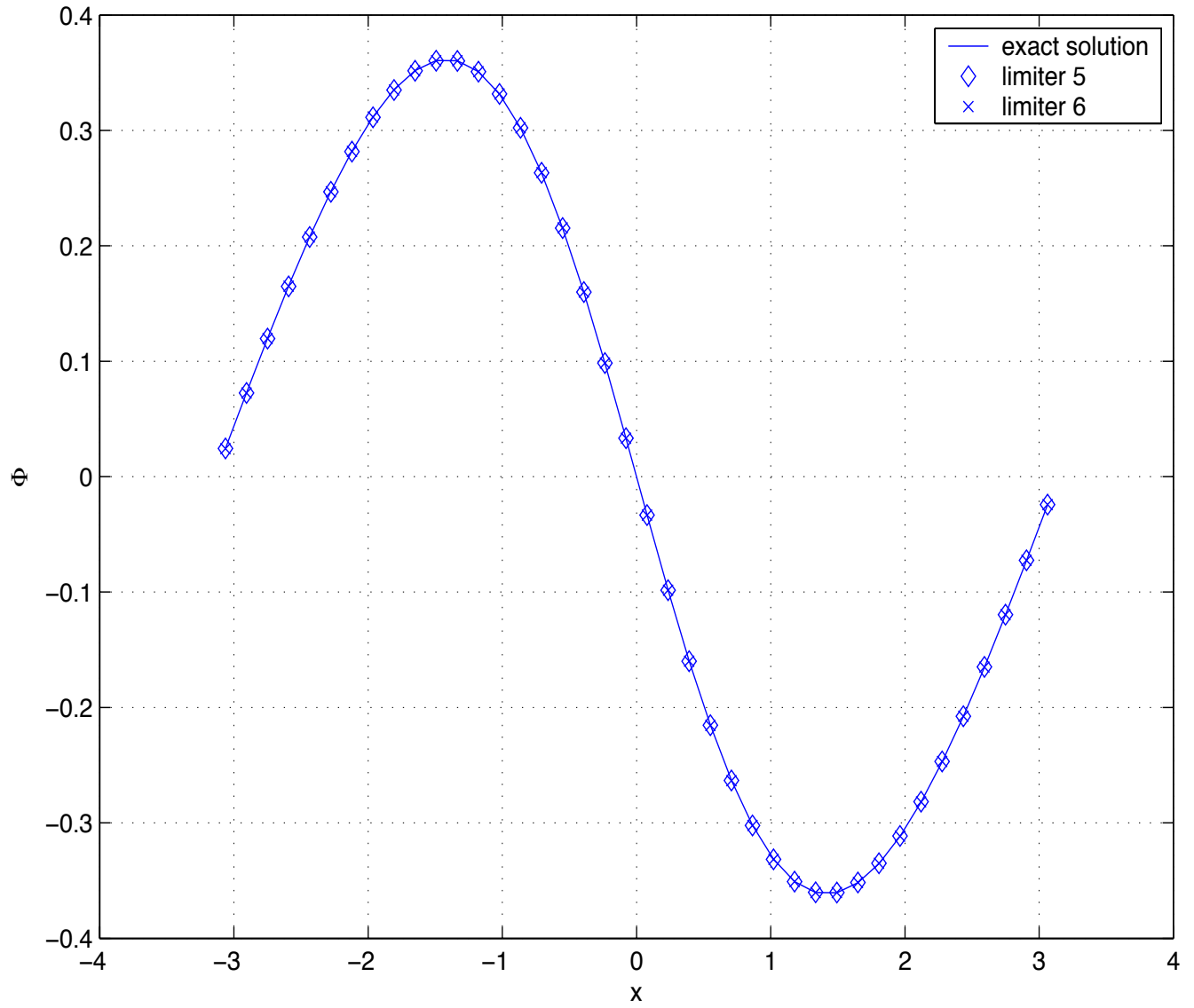


Figure 5. Same as in figure (3), but with limiters 5 and 6.

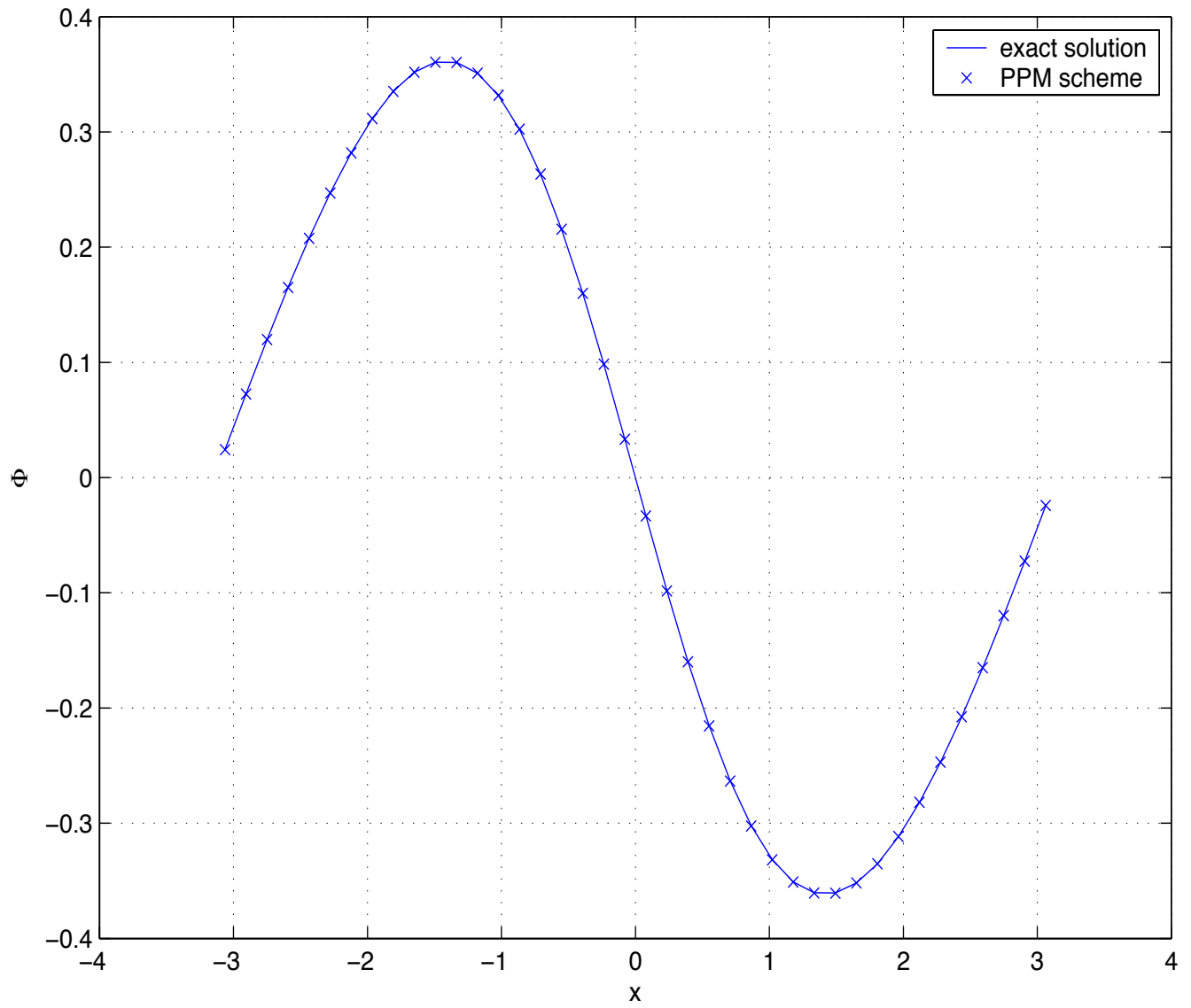


Figure 6. Same as in figure (3), but with the PPM scheme at $t = 1$.

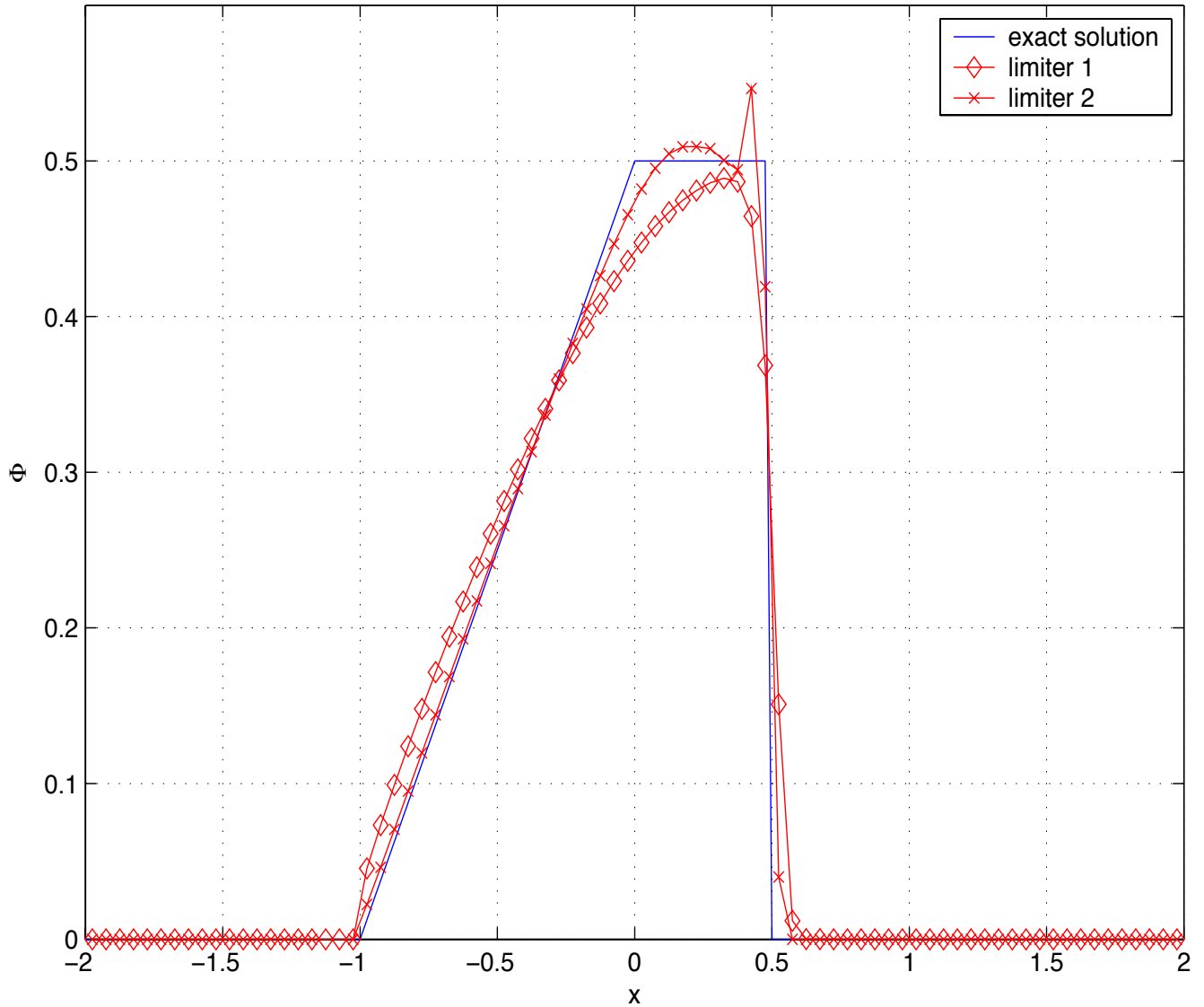


Figure 7. Exact and numerical solution (in forward mode) of the 1-D nonlinear inviscid Burgers equation with slope limiters 1 and 2 at $t = 2$.

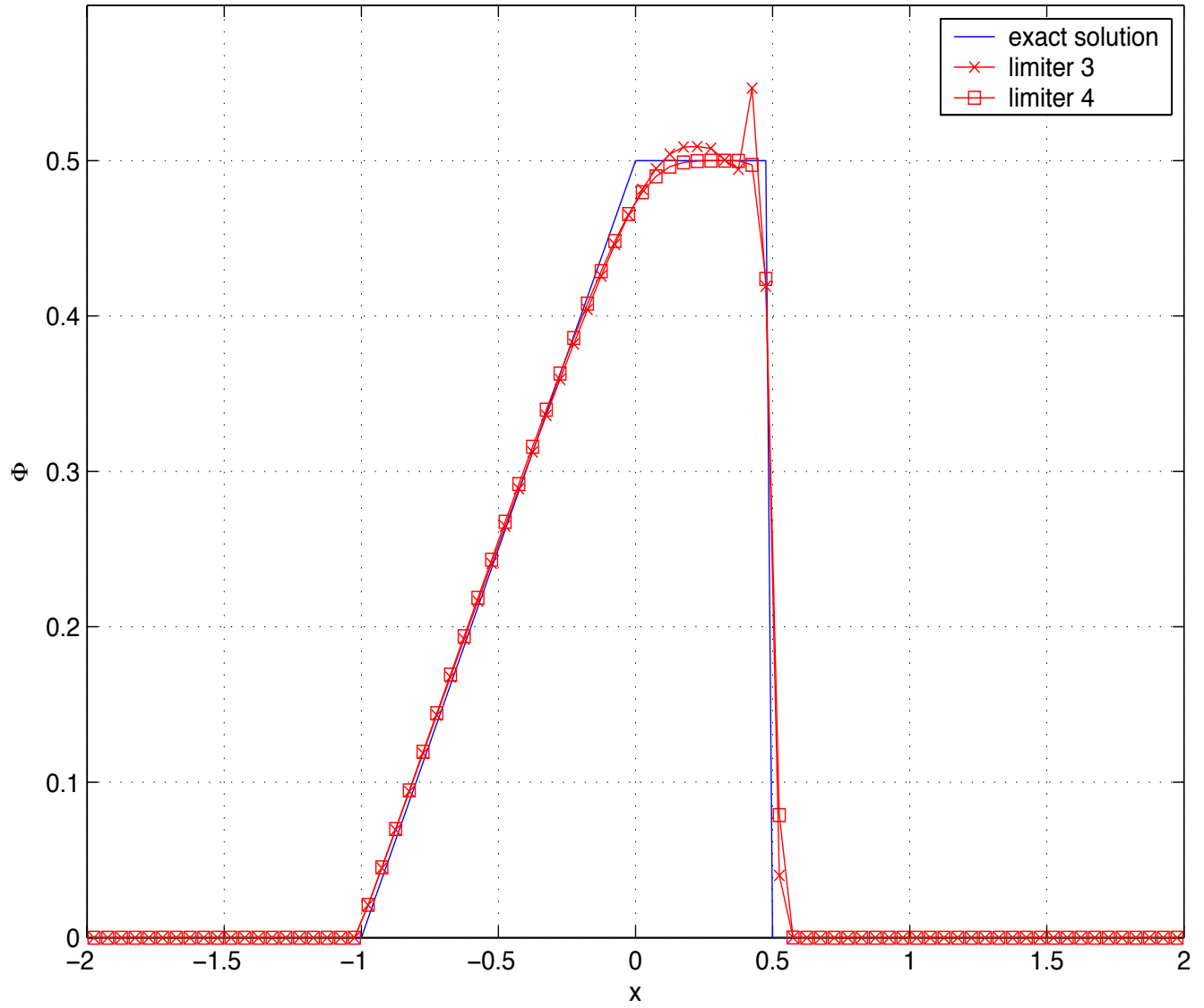


Figure 8. Same as in figure (7), but with limiters 3 and 4.

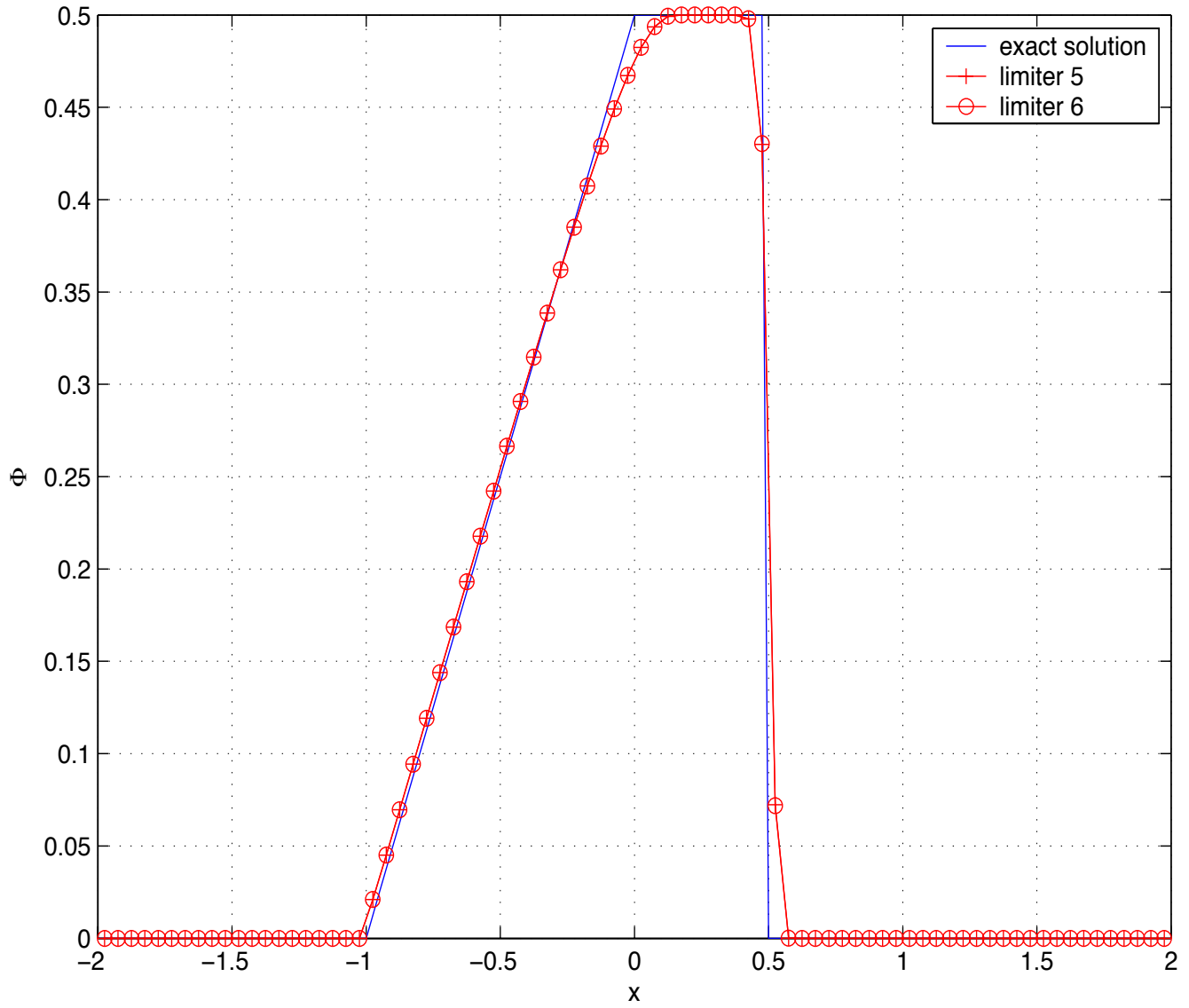


Figure 9. Same as in figure (7), but with limiters 5 and 6.

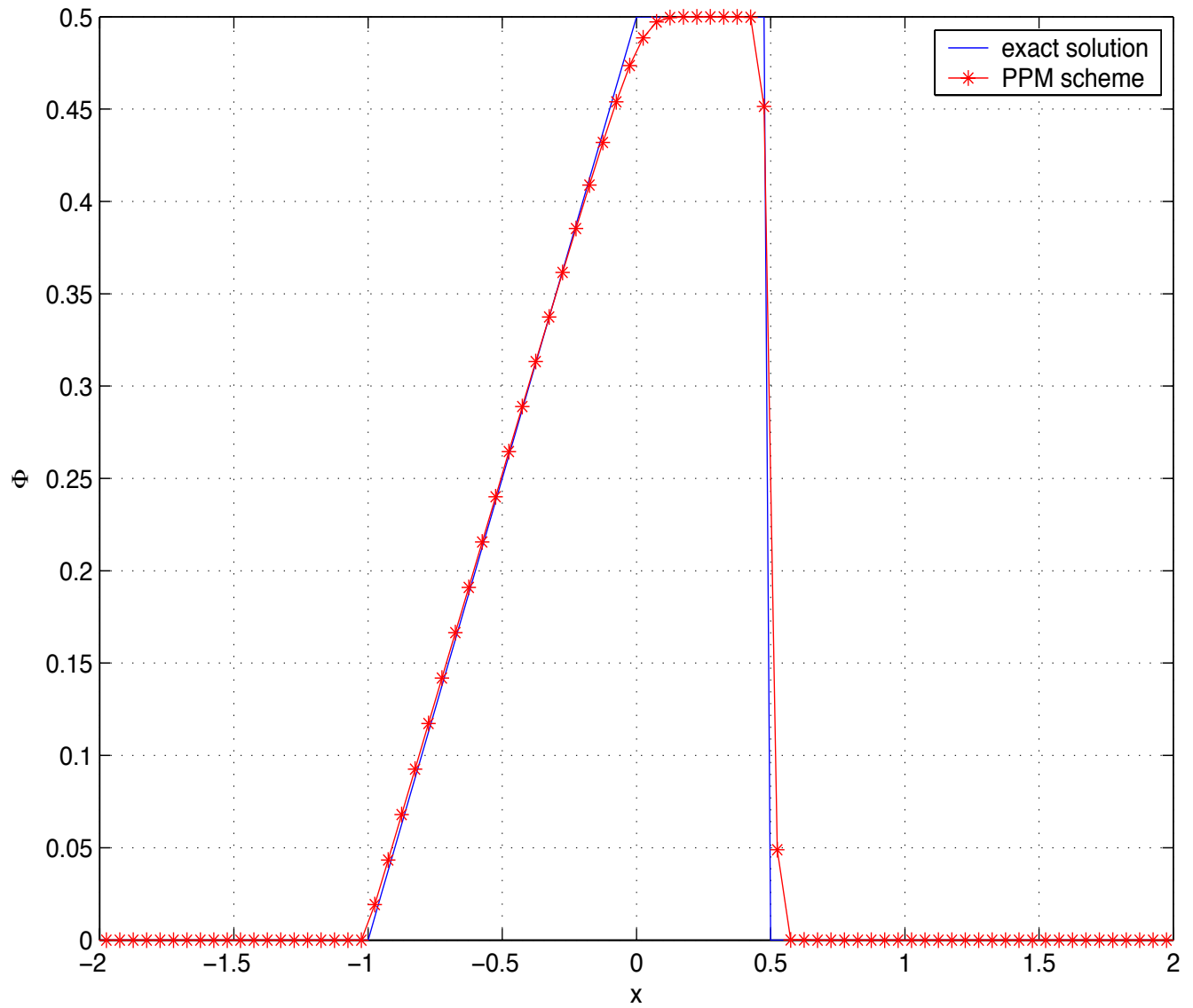


Figure 10. Same as in figure (7), but with limiters 3 and 4.

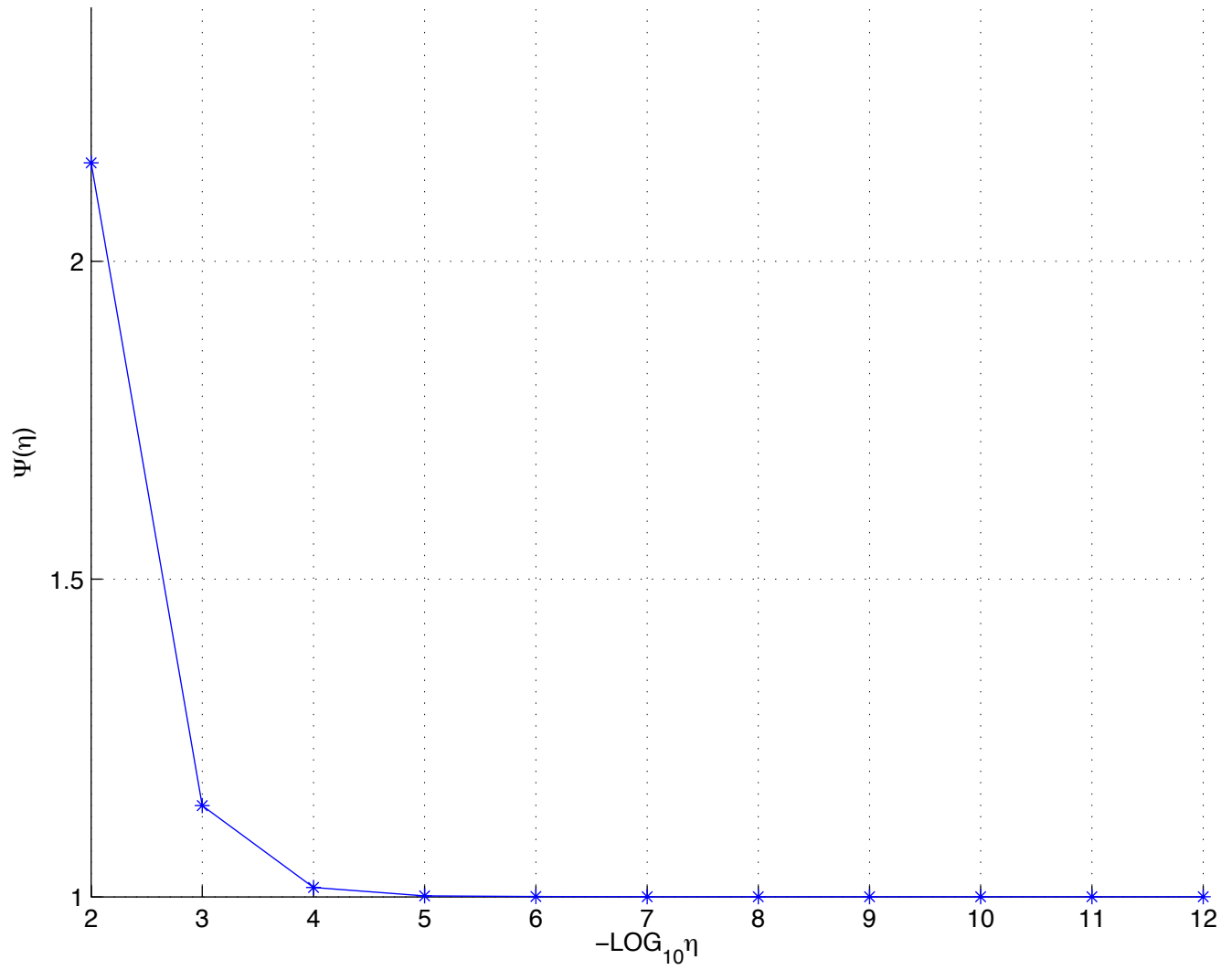


Figure 11. Gradient check ratio using the PPM scheme in the adjoint mode.

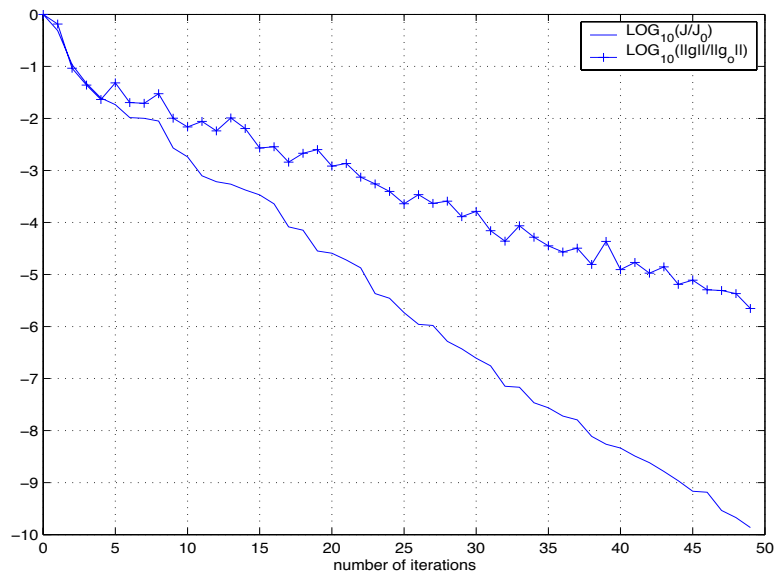


Figure 12. Variations of the normalized cost function $\frac{J}{J_0}$

and normalized gradient $\frac{\|g\|}{\|g_0\|}$ with the number of iterations using slope limiter 1, in forward and adjoint models.

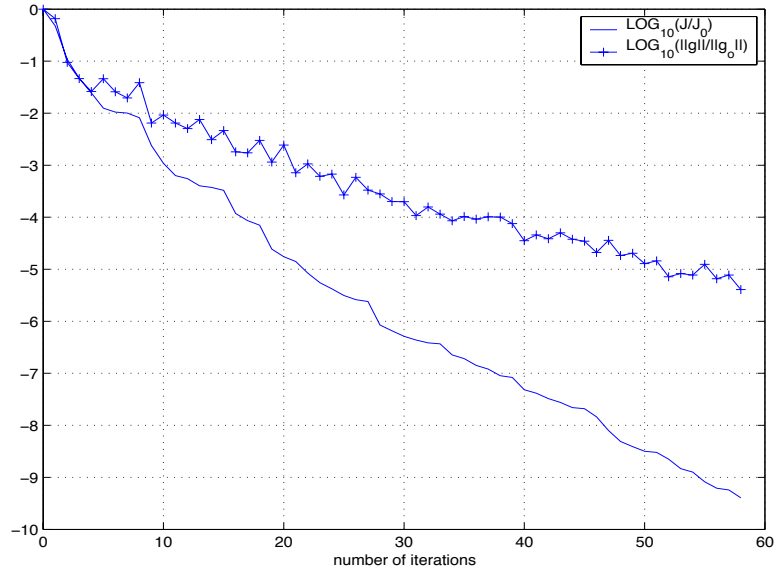


Figure 13. Same as in figure (12), but with limiter 2.

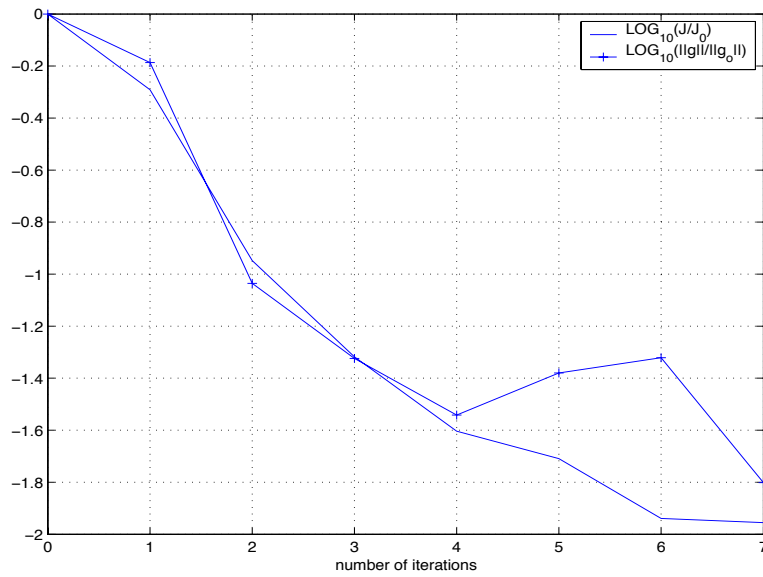


Figure 14. Same as in figure (12), but with limiter 3.

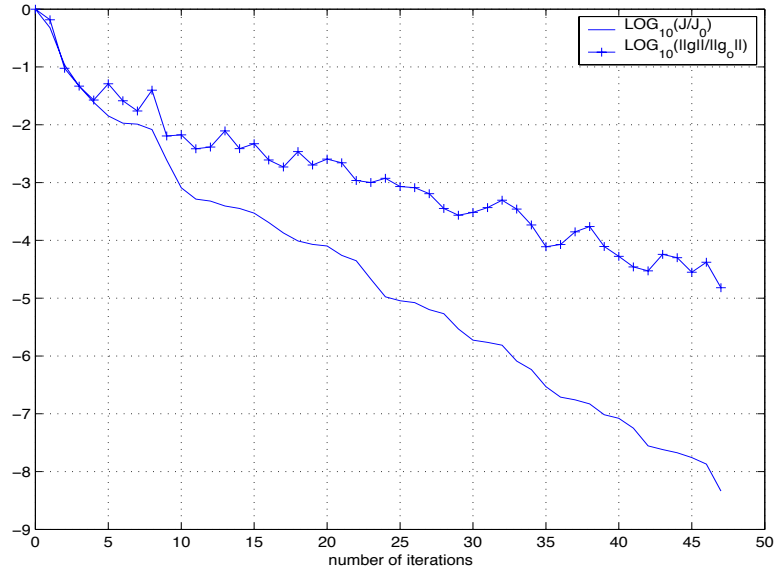


Figure 15. Same as in figure (12), but with limiter 4.

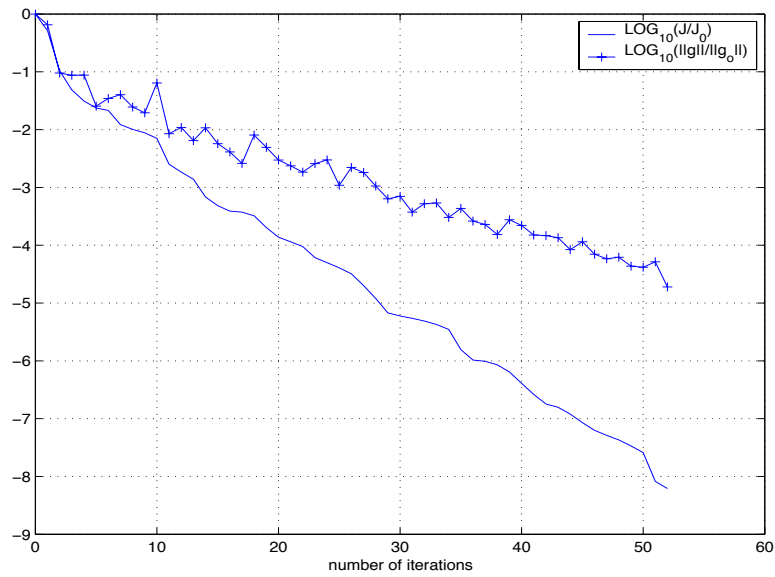


Figure 16. Same as in figure (12), but with limiter 5.

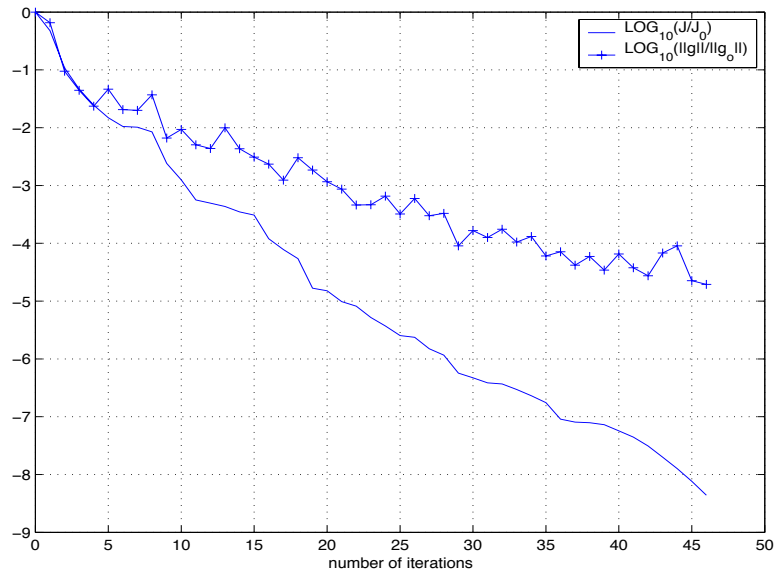


Figure 17. Same as in figure (12), but with limiter 6.

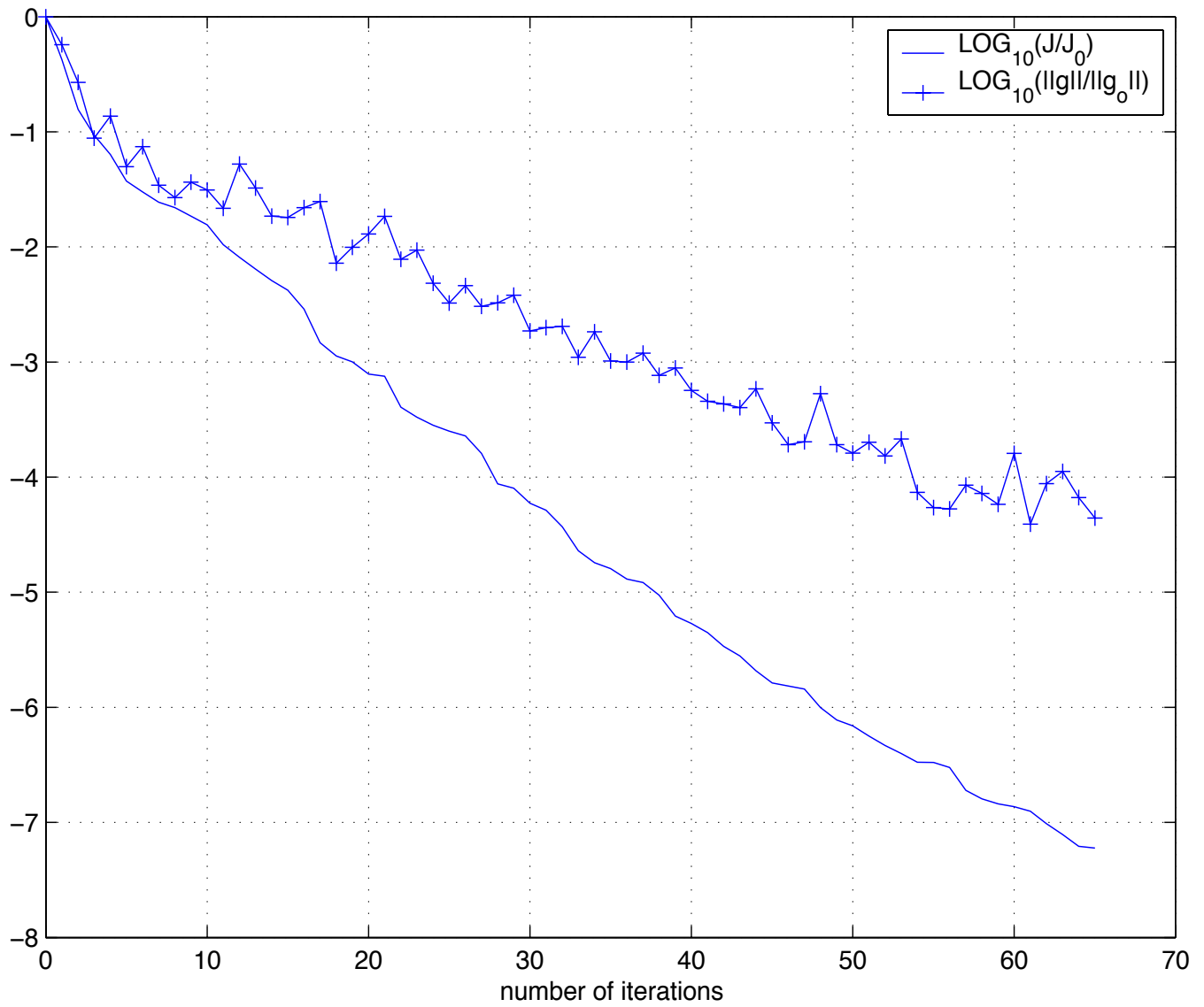


Figure 18. Same as in figure (12), but with the PPM scheme.