UCLA
COMPUTATIONAL AND APPLIED MATHEMATICS

Local Piecewise Hyperbolic Reconstruction of Numerical Fluxes for Nonlinear Conservation Laws

Antonio Marquina

September 1989

CAM Report 89-25

Department of Mathematics
University of California, Los Angeles
Los Angeles, CA. 90024-1555

LOCAL PIECEWISE HYPERBOLIC RECONSTRUCTION OF NUMERICAL FLUXES FOR NONLINEAR CONSERVATION LAWS

ANTONIO MARQUINA

September 8th, 1989

Author's Current Address:
Antonio Marquina
Departamento de Análisis Matemático
Universidad de Valencia 46100-Burjassot (Valencia) SPAIN
e-mail:
marquina%evalvx.decnet.cern%cernvax.bitnet@cunyvm.cuny.edu
fax: (346)3642345

LOCAL PIECEWISE HYPERBOLIC RECONSTRUCTION OF NUMERICAL FLUXES FOR NONLINEAR CONSERVATION LAWS [1]

ANTONIO MARQUINA [2]

Departamento de Análisis Matemático
Universidad de Valencia 46100-Burjassot (Valencia) SPAIN
and
Department of Mathematics
University of California, Los Angeles, CA 90024-1555, U.S.A.

Abstract. In this paper, we construct a local third order accurate shock capturing method for hyperbolic conservation laws, based on numerical fluxes with a total variation diminishing (TVD) Runge-Kutta evolution in time, using the idea recently introduced by C.W. Shu and S.J. Osher for essentially non-oscillatory (ENO) methods. The constructed method is local in the sense that numerical fluxes are reconstructed, without using information about the smoothness in neighboring cells. In order to keep the method local, a new concept of local smoothing is introduced to prevent the increasing of total variation of the solution near discontinuities. The method becomes third order accurate in smooth regions of the solution, except at local extrema where it may degenerate to $O(h^{\frac{3}{2}})$, due to the monotonic character of the reconstruction, thus giving better accuracy than TVD methods at local extrema. Preliminary numerical experiments for scalar conservation laws (1D and 2D), are presented, showing that the method is entropy-satisfying for non-linear fluxes and, for linear discontinuities, the smearing is confined to a very stable neighborhood of the singularity due to the localness of the method, in contrast with ENO and TVD methods. The good resolution of corners (jumps in derivative) is the most remarkable property of our method. The method is efficient since is low-cost and works satisfactorily with high CFL's. Numerical experiments for systems will be presented in future papers.

AMS-MOS Classification - Primary 65M05, Secondary 65M10

Keywords: Nonlinear Conservation Laws, Runge-Kutta, Total Variation Diminishing, Essentially Non-Oscillatory

[1] This paper was written when the author was visiting the Department of Mathematics at U.C.L.A.

[2] Research supported by a Grant from Conselleria de Cultura, Educació i Ciència de la Generalitat Valenciana and Computer time supported by ONR Grant N00014-86-K-0691

1. Introduction

In this paper, we consider numerical approximations to weak solutions of the scalar initial value problem:

$$u_t + f(u)_x = 0 \tag{1.1}$$

$$u(x,0) = u_0(x) \tag{1.2}$$

The initial data $u_0(x)$ are supposed to be piecewise-smooth functions that are either periodic or of compact support.

Let $u_j^n = u_h(x_j, t_n)$ denote a numerical approximation to the exact solution $u(x_j, t_n)$ of (1.1) and (1.2), defined on a computational grid $x_j = jh$, $t_n = n\Delta t$, in conservation form:

$$u_j^{n+1} = u_j^n - \lambda(\hat{f}_{j+\frac{1}{2}} - \hat{f}_{j-\frac{1}{2}}) \tag{1.3}$$

where $\lambda = \frac{\Delta t}{h}$, the numerical flux is a function of $2k$ variables

$$\hat{f}_{j+\frac{1}{2}} = \hat{f}(u_{j-k+1}^n, \ldots, u_{j+k}^n) \tag{1.4}$$

which is consistent with (1.1), i.e.

$$\hat{f}(u, \ldots, u) = f(u) \tag{1.5}$$

The *total variation* of a discrete solution is usually defined by

$$TV(u^n) = \sum_j |u_{j+1}^n - u_j^n| \tag{1.6}$$

This is the total variation with respect to x of the numerical approximation $u_h(x,t)$ in (1.3), considered as a piecewise-constant function defined by

$$u_h(x,t) = u_j^n \tag{1.7}$$

for $x_{j-\frac{1}{2}} < x < x_{j+\frac{1}{2}}$ and $n\Delta t < t < (n+1)\Delta t$ where $x_{j+\frac{1}{2}} = (j+\frac{1}{2})h$.

We say the scheme (1.3) is TVB (total-variation-bounded), if there exists a constant $M > 0$, independent of h for $0 \le t \le T$ (T fixed), such that

$$TV[u_h(\,\cdot\,, t)] \le M \cdot TV(u_0) \tag{1.8}$$

If $M = 1$, the scheme is called TVD (total-variation-diminishing).

The importance of TVB and TVD methods relies on the fact that any refinement sequence $h \to 0$, u_h, has a subsequence h_n, such that u_{h_n} L^1-converges to a weak solution of (1.1) and (1.2). The notion of TVD scheme was introduced by Harten in [2], where he constructed TVD schemes that in the sense of truncation error are high order accurate everywhere except at local extrema where they necessarily degenerate into first order accuracy,(see [1],[2] and [6]).

In order to get high order accuracy in smooth regions, Harten, Osher, Engquist and Chakravarthy constructed essentially non-oscillatory (ENO) schemes which use adaptive stencils, obtaining information from regions of smoothness if discontinuities are present. The analysis and numerical experiments of these methods can be found in [3],[4], and [5]. However, the most efficient implementation of ENO methods has been investigated by C.W. Shu and S.J. Osher in the remarkable papers [8] and [9], where they reconstructed *numerical fluxes* from *point values*, applying the adaptive idea of the ENO interpolation and using a TVD Runge-Kutta type time discretization. Originally, ENO schemes were based on the reconstruction of the *solution* from *cell averages*. The following lemma (taken from [9]) establishes a useful relation between the numerical flux and the flux of a solution to (1.1):

Lemma 1.1 (Shu and Osher) *If a funtion $g(x)$ satifies*

$$f(u(x)) = \frac{1}{h}\int_{x-\frac{h}{2}}^{x+\frac{h}{2}} g(\xi)\,d\xi \tag{1.9}$$

then

$$f(u(x))_x = \frac{g(x+\frac{h}{2}) - g(x-\frac{h}{2})}{h}$$

Lemma 1.1 implies that in order to approximate the numerical flux $\hat{f}_{j+\frac{1}{2}}$ to a high order accuracy it is enough to reconstruct $g(x_{j+\frac{1}{2}})$ up to the same order. The reconstruction "via primitive function" (see [4]), seems to be the most efficient, and, it is used in [9] in an ENO fashion.

In this paper we construct a local third order accurate method by using a piecewise hyperbolic reconstruction of the function g in (1.9), instead of the polynomial ENO interpolation of the primitive function of g as in [9], and the evolution in time is performed by means of the Shu-Osher third order TVD Runge-Kutta method (see [9]), computed by the recurrence formula:

$$u_j^{(i)} = \sum_{k=0}^{i-1}[\alpha_{ik}\cdot u_j^{(k)} + \beta_{ik}\cdot(-\lambda)\cdot(\hat{f}_{j+\frac{1}{2}}^{(k)} - \hat{f}_{j-\frac{1}{2}}^{(k)})], \qquad i = 1,2,3 \tag{1.10}$$

where

$$u_j^{(0)} = u_j^n, \qquad u_j^{(3)} = u_j^{n+1} \tag{1.11}$$

with

$$\lambda = \frac{\Delta t}{h} \leq \lambda_0 \tag{1.12}$$

and λ_0 is inversely proportional to $max|f'(u)|$, as usual, with coefficients of Table 1. The obtained method is *local* in the sense that the numerical flux depends only on four point values, in contrast with the corresponding third order accurate ENO that uses six point values. When discontinuities are present, some smoothing becomes necessary in order to prevent the increasing of the total variation of the solution, and this is achieved, in our method, by means of a "preprocessing of derivatives" carried out in every computational cell, as explained in section 2. The method is constructed using the Roe entropy-fix framework as presented in [9], and it becomes third order accurate in smooth regions except at local extrema where it may degenerate to $O(h^{\frac{3}{2}})$, due to the shifting of the fluxes and the monotonic character of the reconstruction, which may degenerate into second order accuracy along the cell of transition at local extrema (see tables 2 and 3). We present two piecewise hyperbolic reconstructions: The first one, satisfactory for contact discontinuities, and "overcompressive" for non linear fluxes, consisting of natural hyperbolas without "preprocessing" of derivatives; and the second one, satisfactory for all fluxes, with more smoothing for contact discontinuities, consisting of hyperbolas with "preprocessed derivatives". Concerning the method, built from the second reconstruction, the following features were found through numerical experiments (see section 4):

a) Third order accuracy in smooth regions of the solution except at local extrema where it may degenerate to $O(h^{\frac{3}{2}})$.

b) Correct position and speed of discontinuities.

c) Entropy-satisfying discontinuities for non linear fluxes.

d) No spurious oscillations, due to the "denoising" effect of the monotonic character of the reconstruction.

e) Good resolution of linear discontinuities and jumps of derivatives, where the smearing appears to be more local than in ENO methods, with a satisfactory behavior for high CFL (see Figs. 6 and 9).

f) The Artificial Compression method (see [10] and [9]), for linear discontinuities works efficiently with low CFL, (as in ENO methods), (see Figs. 7 and 12).

As the third order ENO method (as it appears in [9], with r=2), our method behaves as a TVB method, but we have not yet been able to prove this property. However, we have found a "local TVB property" of the reconstruction, (too weak to imply TVB), that is a practical criterium in order for a method, based on point values, to prevent the increasing of the total variation of the reconstruction essentially, when discontinuities are present. This property is satisfied not only by our method, but also by the third order ENO, making

our method close to the ENO methods. A maximum principle appears to be necessary in order to prove the TVB property of the scheme.

The paper is organized as follows: section 2 contains the reconstruction step, the complete algorithm is presented in section 3, and section 4 contains numerical experiments, including one-dimensional non-convex Riemann problems, contact discontinuities in 1D and 2D, one-dimensional Burgers' equation with smooth data and two-dimensional Riemann problems for Burgers' equation.

2. Piecewise Hyperbolic Reconstruction

The most important step in our method, as well as in ENO methods, is the reconstruction step. Since we look for the reconstruction of the function g in (1.9) up to third order accuracy in the sense of truncation error, we shall define what we will call "natural data"(i.e. grid data) and we will establish the general reconstruction problem.

Let $g(x)$ be a piecewise smooth function that is either periodic or of compact support. We have defined a computational grid $x_j = jh$, j integer, $h > 0$, where the cells are

$$C_j = \{x : x_{j-\frac{1}{2}} \leq x \leq x_{j+\frac{1}{2}}\} \tag{2.1}$$

where $x_{j+\frac{1}{2}} = x_j + \frac{h}{2}$.

Natural Data :

(1) For every j the mean value of $g(x)$ in C_j, v_j is given, i.e.,

$$v_j = \frac{1}{h}\int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} g(\xi)\, d\xi \tag{2.2}$$

(2) For every j, $d_{j+\frac{1}{2}}$ is given, which is either $g'(x_{j+\frac{1}{2}})$ or

$$d_{j+\frac{1}{2}} = \frac{v_{j+1} - v_j}{h} \tag{2.3}$$

$(d_{j+\frac{1}{2}} = g'(x_{j+\frac{1}{2}}) + O(h^2))$. For our purposes we will suppose (2.3) satisfied all the time.

Let $\Re$ be a class of elementary functions. We shall only concern with third order accurate reconstructions:

General Reconstruction Problem (GRP): *For every j, find r_j in $\Re$, defined on C_j, such that $r_j(x)$ reconstructs $g(x)$ on C_j up to third order accuracy, i.e. for every j such that $g(x)$ is smooth on C_j, then*

$$g(x) - r_j(x) = O(h^3) \qquad \forall x \in C_j \tag{2.4}$$

A method of reconstruction is a well-defined rule to solve GRP.

In order to solve GRP with different classes of functions $\Re$ we will consider the following concept

A *natural solution* to GRP is a third order method of reconstruction $\{r_j\}$ consisting of functions chosen from $\Re$ such that for every j the following conditions are satisfied:

$$v_j = \frac{1}{h}\int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} r_j(\xi)\, d\xi \tag{2.5}$$

$$d_{j+\frac{1}{2}} = r_j'(x_{j+\frac{1}{2}}) \tag{2.6}$$

Standard arguments show that (2.5) and (2.6) imply third order accuracy.

Our discussion will include the following two classes of elementary functions:

(1) The class of parabolas of the form:

$$p_j(x) = a_j + b_j \cdot (x - x_j) + (\frac{c_j}{2}) \cdot (x - x_j)^2 \tag{2.7}$$

defined on C_j. We denote this class by $\Re_p$.

(2) The class of hyperbolas of the form:

$$r_j(x) = a_j + \frac{\lambda_j}{(x - x_j) + c_j} \tag{2.8}$$

defined on C_j. We denote this class by $\Re_h$

We are interested in reconstructions with the following property:

Definition 2.1 We say a method of reconstruction $\{r_j\}$ of $g(x)$ is *local* if for every j the function r_j depends only on v_j, $d_{j-\frac{1}{2}}$ and $d_{j+\frac{1}{2}}$.

The simplest example of a *local* third order method of reconstruction is the "Local Parabolic Reconstruction" (LPR). Indeed, for every j there is a unique parabola of the form (2.5), defined on C_j and determined from v_j, $d_{j-\frac{1}{2}}$ and $d_{j+\frac{1}{2}}$ by

$$c_j = \frac{d_{j+\frac{1}{2}} - d_{j-\frac{1}{2}}}{h} \tag{2.9}$$

$$b_j = \frac{d_{j+\frac{1}{2}} + d_{j-\frac{1}{2}}}{2} \tag{2.10}$$

$$a_j = v_j - c_j \cdot \frac{h^2}{24} \tag{2.11}$$

It is easy to see that the LPR is the *natural solution* to GRP with $\Re = \Re_p$.

We will use the following notations:

$$D_j = d_{j+\frac{1}{2}} - d_{j-\frac{1}{2}} \tag{2.12}$$

$$d_j = \frac{d_{j+\frac{1}{2}} + d_{j-\frac{1}{2}}}{2} \tag{2.13}$$

If d_j is non-zero then we define the adimensional parameter:

$$\alpha_j = \frac{D_j}{2 \cdot d_j} \tag{2.14}$$

Obviously we can also determine the parabola from v_j, d_j and D_j, through (2.9),(2.10) and (2.11).

The reconstruction procedure is repeated at every time step, and, therefore, the change in total variation of the reconstruction must be controlled. The *local total variation* of the reconstruction $\{r_j\}$ is defined by

$$LTV_j = TV(r_j) \tag{2.15}$$

where $TV(r_j)$ means the total variation of the function $r_j(x)$ in the cell C_j. The size of LTV_j determines locally the increasing of the total variation of the reconstruction. Then, we introduce

Definition 2.2 A method of reconstruction is *local total variation bounded* (LTVB), if there exists a constant $M > 0$, independent of h, (depending only on the function $g(x)$ to be reconstructed), such that

$$LTV_j \leq M \cdot h \qquad \forall j \tag{2.16}$$

The local total variation for LPR is

$$TV(p_j) = \frac{h}{2} \cdot \left(|d_{j-\frac{1}{2}}| + |d_{j+\frac{1}{2}}| - c(j) \cdot \frac{2 \cdot |d_{j-\frac{1}{2}}| \cdot |d_{j+\frac{1}{2}}|}{|d_{j+\frac{1}{2}}| + |d_{j-\frac{1}{2}}|} \right) \tag{2.17}$$

where $c(j) = 0$ if $d_{j-\frac{1}{2}} \cdot d_{j+\frac{1}{2}} \geq 0$ and $c(j) = 1$ otherwise. If discontinuities are present in $g(x)$, then, for some j, $d_{j+\frac{1}{2}} = O(h^{-1})$, and therefore, (2.16) is not satisfied for that j. Thus, LPR method is not LTVB. We have seen, through our numerical experiments, that when using the LPR method for Burger's equation with smooth data, with the algorithm of section 3, the total variation of the numerical solution blows up near discontinuities, (when they appear). The LTVB condition appears to be necessary for a method to own enough "smoothing" to deal with discontinuities and to behave as a TVB method.

Thus, we must look for methods of reconstruction satisfying Definition 2.2 in order to prevent the increasing of total variation. Since the total variation of the function $g(x)$ on a cell C_j, where it is smooth, depends essentially (up to $O(h^3)$) on the size of $d_{j+\frac{1}{2}}$ and $d_{j-\frac{1}{2}}$, (this follows from the trapezoidal rule), then, in order to get local total variation

bounded methods of reconstruction, it will be necessary to correct the values of $d_{j+\frac{1}{2}}$ and $d_{j-\frac{1}{2}}$, on every cell, preserving the third order accuracy of the reconstruction. For this purpose we introduce the following

Definition 2.3 A *numerical left derivative*, is a function of $2k+2$ variables, $(k \geq 0)$

$$dl_j = dl(d_{j-k-\frac{1}{2}}, \ldots, d_{j+k+\frac{1}{2}}) \tag{2.18}$$

which is "consistent" in the following sense:

$$dl_j - d_{j-\frac{1}{2}} = O(h^2) \tag{2.19}$$

The concept of numerical right derivative, dr_j is defined analogously and satisfying

$$dr_j - d_{j+\frac{1}{2}} = O(h^2) \tag{2.20}$$

instead of (2.19). The numerical central derivative, d_j^*, is also defined in the same way, by using

$$d_j^* - d_j = O(h^2) \tag{2.21}$$

where d_j is defined by (2.13). The numerical difference, D_j^*, is defined also as a function of the same variables and the following consistency property:

$$D_j^* - D_j = O(h) \tag{2.22}$$

where D_j is defined by (2.12).

A *consistent preprocessing of derivatives* is a set of pairs of lateral numerical derivatives $\{dl_j, dr_j\}$, defined as above, associated to the natural data. A preprocessing is called *local* if the numerical derivatives are functions of only two variables.

A *preprocessed solution* to GPR is a third order method of reconstruction consisting of functions chosen from $\Re$ such that for every j, (2.5),

$$dl_j = r_j'(x_{j-\frac{1}{2}}) \tag{2.23}$$

and

$$dr_j = r_j'(x_{j+\frac{1}{2}}) \tag{2.24}$$

are satisfied. We need consistency properties (2.19),(2.20),(2.21) and (2.22) since they keep preprocessed solutions third order accurate.

Then, we have the following

Theorem 2.4 *The polynomial ENO third order method of reconstruction 'via primitive function' of the function $g(x)$ is a nonlocal preprocessed parabolic solution to* (GRP) *that is local total variation bounded*

Proof: The following algorithm determines the lateral numerical derivatives in the ENO third order method on a cell C_j:

if $|d_{j-\frac{1}{2}}| \leq |d_{j+\frac{1}{2}}|$ **then**

$dl_j = d_{j-\frac{1}{2}}$

if $|D_j| \leq |D_{j-1}|$ **then**

$dr_j = d_{j+\frac{1}{2}}$

else

$dr_j = dl_j + D_{j-1}$

else

$dr_j = d_{j+\frac{1}{2}}$

if $|D_j| \leq |D_{j+1}|$ **then**

$dl_j = d_{j-\frac{1}{2}}$

else

$dl_j = dr_j - D_{j+1}$

Thus, we determine the parabola (2.7) by substituting in (2.9) and (2.10), $d_{j-\frac{1}{2}}$ and $d_{j+\frac{1}{2}}$ by dl_j and dr_j, respectively. We denote that parabola by p_j^*. Since the numerical derivatives are functions of four variables, then, this preprocessing is not local. On the other hand, let us choose a number $h > 0$ such that there is at least two cells between two jumps of $g(x)$. Since $g(x)$ is a piecewise smooth function it is easy to see that there exists a constant $M > 0$, depending only on derivatives of g, in smooth regions, such that for all j, except for a finite number of "isolated" $j's$ (for which $d_{j+\frac{1}{2}} = O(h^{-1})$),

$$|d_{j+\frac{1}{2}}| \leq M$$

Thus, by the definition of the preprocessing we have that for all j,

$$|dl_j| + |dr_j| \leq 4 \cdot M \tag{2.25}$$

Therefore, according to (2.17) applied to p_j^*, we obtain

$$LTV_j = TV(p_j^*) \leq 4 \cdot M \cdot h \tag{2.26}$$

and the ENO reconstruction will be LTVB. ‖

Next, we shall look at local piecewise hyperbolic reconstructions. For the class $\Re_h$ we have no natural solution to GRP if local extrema are present in $g(x)$, because hyperbolas

are monotonic. However, we can find a unique hyperbola r_j of the form (2.8) in every cell C_j, such that $d_{j-\frac{1}{2}} \cdot d_{j+\frac{1}{2}} > 0$, satisfying (2.5) and (2.6). We say a cell C_j is a *transition cell* if $d_{j-\frac{1}{2}} \cdot d_{j+\frac{1}{2}} \leq 0$ Since we look for a hyperbola at every cell, including transition cells, we will study more general conditions for the existence. For our purpose we shall need the following simple lemma,

Lemma 2.5 *We consider the generic cell*

$$C_0 = \{x : |x - x_0| \leq \frac{h}{2}\}$$

Let θ be a real number between -1 and 1. We set $x(\theta) = x_0 + \theta \cdot \frac{h}{2}$. Let v_0 be the mean value of $g(x)$ in C_0. Let d be a nonzero real number. Then

(1) The hyperbola

$$r_0(x) = v_0 + d \cdot h \cdot \frac{1}{\alpha^2} \cdot \left(\log\left(\frac{2 - \alpha(1-\theta)}{2 + \alpha(1+\theta)}\right) - \frac{h}{(x - x_0) - \frac{h}{2} \cdot (\theta + \frac{2}{\alpha})} \right) \tag{2.27}$$

with derivative

$$r_0'(x) = d \cdot \frac{h^2}{\alpha^2 \cdot \left((x - x_0) - \frac{h}{2} \cdot (\theta + \frac{2}{\alpha})\right)^2} \tag{2.28}$$

is well-defined according to the following restrictions on the range of values of the parameter α

(a) If $-1 < \theta < 1$ then $\frac{-2}{1+\theta} < \alpha < \frac{2}{1-\theta}$

(b) If $\theta = 0$ then $-2 < \alpha < 2$

(c) If $\theta = -1$ then $\alpha < 1$

(d) If $\theta = 1$ then $\alpha > -1$

(2) The mean value of $r_0(x)$ in C_0 is v_0, i.e.,

$$v_0 = \frac{1}{h} \int_{x_0 - \frac{h}{2}}^{x_0 + \frac{h}{2}} r_0(\xi)\, d\xi \tag{2.29}$$

(3) The derivative of the hyperbola at $x(\theta)$ is d:

$$r_0'(x(\theta)) = d \tag{2.30}$$

(4) If $d1$ is a nonzero real number such that $d \cdot d1 > 0$, and θ_1 is a real number such that $-1 < \theta_1 < 1$ and $\theta_1 \neq \theta$, then, we can determine the value of α by the formula

$$\alpha = \Big(\frac{2}{\theta_1 - \theta}\Big) \cdot \Big(1 - \sqrt{\frac{d}{d1}}\Big) \tag{2.31}$$

such that

$$r_0'(x(\theta_1)) = d1 \tag{2.32}$$

Thus, the hyperbola is completely determined from v_0, d and $d1$, when $d \cdot d1 > 0$. Moreover, the derivative of $r_0(x)$ at the mid point $x(\frac{\theta+\theta_1}{2})$ is

$$r_0'\Big(x\Big(\frac{\theta+\theta_1}{2}\Big)\Big) = \operatorname{sgn}(d) \cdot \Big(\frac{2 \cdot \sqrt{|d|} \cdot \sqrt{|d1|}}{\sqrt{|d|} + \sqrt{|d1|}}\Big)^2 \tag{2.33}$$

(5) We set $dl_0 = r_0'(x_0 - \frac{h}{2})$ and $dr_0 = r_0'(x_0 + \frac{h}{2})$. Then, the total variation of the hyperbola on C_0 is

$$TV(r_0) = h \cdot \sqrt{dl_0 \cdot dr_0} \tag{2.34}$$

(6) If $\theta = 0$ then the lateral values of r_0 are well defined for $-2 < \alpha < 2$ by the formulas:

$$r_0\big(x_0 + \frac{h}{2}\big) = v_0 + d \cdot h \cdot \eta(\alpha) \tag{2.35}$$

$$r_0\big(x_0 - \frac{h}{2}\big) = v_0 - d \cdot h \cdot \eta(-\alpha) \tag{2.36}$$

where the function η is defined as:

$$\eta(\alpha) = \frac{1}{\alpha^2} \cdot \Big(\log\Big(\frac{2-\alpha}{2+\alpha}\Big) - \frac{2 \cdot \alpha}{2-\alpha}\Big) \tag{2.37}$$

(the function η is positive in $-2 < \alpha < 2$, and it has a removable discontinuity at $\alpha = 0$ by defining $\eta(0) = \frac{1}{2}$).

(7) Let D_0^* be a nonzero preprocessed difference satisfying (2.22). If α defined by

$$\alpha = \frac{D_0^*}{2 \cdot d}$$

satisfies the restrictions in (1), the hyperbola r_0 satisfies (2.30) and $r_0''(x(\theta)) = \frac{D_0^*}{h}$

Proof: Straightforward. ||

First, we will study the "most natural" local hyperbolic reconstruction (LHR). Since our reconstruction is local we restrict our discussion to the cell C_0 and the natural data of the cell: v_0, $d_{-\frac{1}{2}}$ and $d_{\frac{1}{2}}$. In order to fit the hyperbola we establish formulas to obtain d and α with $\theta = 0$. The LHR algorithm computes for the generic cell C_0 the derivative at the point x_0, and the adimensional parameter α from $d_{-\frac{1}{2}}$ and $d_{\frac{1}{2}}$. If C_0 is a non-transition cell, then the value assigned to d is the mean (2.33) and α is computed by means of (2.31), and, therefore, the hyperbola obtained has as derivatives at the end points of the cell $d_{-\frac{1}{2}}$ and $d_{\frac{1}{2}}$. If C_0 is a transition cell we change the derivative with largest absolute value by the other one multiplied by h^2, thus, the reconstruction on this cell may degenerate to second order.

Let tol be a constant such that $tol = O(h^2)$, (e.g. define $tol = h^2$).

*** Local Hyperbolic Reconstruction**

if $(|d_{-\frac{1}{2}}| \leq tol)$ **and** $(|d_{\frac{1}{2}}| \leq tol)$ **then**

$d = 0$ **and** $\alpha = 0$

else

if $(|d_{-\frac{1}{2}}| \leq tol)$ **or** (C_0 is a transition cell with $|d_{-\frac{1}{2}}| \leq |d_{\frac{1}{2}}|$) **then**

$$d = 4 \cdot d_{\frac{1}{2}} \cdot \left(\frac{h}{1+h}\right)^2$$

$$\alpha = 2\frac{1-h}{1+h}$$

else

if $(|d_{\frac{1}{2}}| \leq tol)$ **or** (C_0 is a transition cell with $|d_{\frac{1}{2}}| \leq |d_{-\frac{1}{2}}|$) **then**

$$d = 4 \cdot d_{-\frac{1}{2}} \cdot \left(\frac{h}{1+h}\right)^2$$

$$\alpha = -2\frac{1-h}{1+h}$$

else

$$d = \operatorname{sgn}(d_{-\frac{1}{2}}) \cdot \frac{4 \cdot |d_{-\frac{1}{2}}| \cdot |d_{\frac{1}{2}}|}{|d_{-\frac{1}{2}}| + |d_{\frac{1}{2}}| + 2 \cdot \sqrt{d_{-\frac{1}{2}} \cdot d_{\frac{1}{2}}}}$$

if $|d_{-\frac{1}{2}}| \leq |d_{\frac{1}{2}}|$ **then**

$$\alpha = 2 \cdot \left(\sqrt{\frac{d}{d_{-\frac{1}{2}}}} - 1\right)$$

else

$$\alpha = 2 \cdot \left(1 - \sqrt{\frac{d}{d_{\frac{1}{2}}}}\right)$$

if $(|d_{\frac{1}{2}}| \leq tol)$ **or** (C_0 is a transition cell with $|d_{\frac{1}{2}}| \leq |d_{-\frac{1}{2}}|$) **then**

$$d = 2 \cdot d_{-\frac{1}{2}} \cdot \left(\frac{h^2}{1+h^2}\right)$$

$$\alpha = -2 \cdot \left(\sqrt{\frac{2}{1+h^2}} - 1\right)$$

else

$$d = \frac{2 \cdot d_{-\frac{1}{2}} \cdot d_{\frac{1}{2}}}{d_{-\frac{1}{2}} + d_{\frac{1}{2}}}$$

if $|d_{-\frac{1}{2}}| \leq |d_{\frac{1}{2}}|$ **then**

$$\alpha = 2 \cdot \left(\sqrt{\frac{d}{d_{-\frac{1}{2}}}} - 1\right)$$

else

$$\alpha = 2 \cdot \left(1 - \sqrt{\frac{d}{d_{\frac{1}{2}}}}\right)$$

Transition cells are treated analogously to the LHR method, but using the harmonic mean instead of (2.33). In Table 3, we have the numerical order computed for LHHR method, for the same function as in Table 2, under the same conditions.

Now, we will show that LHHR method is LTVB. We shall need the following

Lemma 2.6 *The range of values of the adimensional parameter α for the LHHR method is*

$$-2(\sqrt{2}-1) \leq x \leq 2(\sqrt{2}-1) \tag{2.39}$$

Proof: For transition cells the proof is trivial. If C_0 is a non-transition cell then we will suppose that $|d_{-\frac{1}{2}}| \leq |d_{\frac{1}{2}}|$, (the other case is symmetric). Then, the algorithm defines

$$\alpha = 2 \cdot \left(\sqrt{\frac{d}{d_{-\frac{1}{2}}}} - 1\right)$$

Thus, it is enough to prove that $\frac{d}{d_{-\frac{1}{2}}} \leq 2$ and this follows from $d_{-\frac{1}{2}} \cdot d_{\frac{1}{2}} > 0$. ||

Theorem 2.7 *The (LHHR) method of reconstruction of the function $g(x)$ is a local preprocessed hyperbolic solution to* GRP *that is local total variation bounded*

Proof: Following the same argument as in Theorem 2.4, we can find a constant $M > 0$ such that for all j, except for a finite number of 'isolated' j's (for which $d_{j+\frac{1}{2}} = O(h^{-1})$), $|d_{j+\frac{1}{2}}| \leq M$. From (2.28) if C_j is a non-transition cell and $|d_{j-\frac{1}{2}}| \leq |d_{j+\frac{1}{2}}|$, then the preprocessed derivatives at the end point of the cells are the following:

$$dl_j = d_{j-\frac{1}{2}} \tag{2.40}$$

$$dr_j = d_{j-\frac{1}{2}} \cdot \left(\frac{2}{2-\alpha}\right)^2 \tag{2.41}$$

From (2.34) and Lemma 2.6 it follows that

$$LTV_j = TV(r_j) \leq 2 \cdot M \cdot h \tag{2.42}$$

The argument is similar for transition cells. ||.

In theory, it is possible to choose other means between the harmonic mean and (2.33) giving methods of reconstruction which are LTVB, (and more compressive!), but we have not found any else as computationally convenient as LHHR.

Other reconstructions, based on hyperbolas, are possible and methods of reconstruction that recover accuracy at local extrema are under investigation. For contact discontinuities we have constructed methods based on Lemma 2.5 (7), more compressive than (LHR) method, and they will be treated in future papers. As a sample, we show in Figure 8 an experiment involving jumps in the derivative, where we have compression and resolution with high CFL (0.8).

3. Piecewise Hyperbolic Methods

In this section, we describe the algorithm based on the first order Roe scheme ([7]), with the entropy-fix correction, due to Shu and Osher ([9]), for local piecewise hyperbolic reconstructions, giving us what we will call Piecewise Hyperbolic Methods (PHM). Since the evolution in time is performed by means of the third order TVD Runge-Kutta method (1.10) with positive coefficients (see Table 1), we have that the numerical solution at every time step is a convex combination of Euler forward time substeps (1.3). Thus, we restrict our description of the algorithm to the computation of numerical fluxes $\hat{f}_{j+\frac{1}{2}}$ in (1.3).

Roughly speaking, the reconstruction procedure is integrated in (1.3), taking into account the dynamics of the differential equation (1.1). Indeed, the numerical fluxes are reconstructed from the *upwind side*, (i.e., according to the direction of the *wind*), except if *wind* changes direction at the cell, (i.e., there is a "sonic point" at the cell), then, a local Lax-Friedrichs Flux Decomposition is performed. Thus, we have two alternative phases: Upwindness or Flux Decomposition. Flux Decomposition is only used in cells containing "sonic points", and, since that points are isolated, the cost of the algorithm depends on the "upwindness phase". The "upwind side" is determined according to the *local* sign of $f'(u)$ at $x_{j+\frac{1}{2}}$. We use in our case the "Roe" speed

$$\overline{a}_{j+\frac{1}{2}} = \frac{f(u^n_{j+1}) - f(u^n_j)}{u^n_{j+1} - u^n_j} \tag{3.1}$$

to determine the sign of $f'(u_{j+\frac{1}{2}})$. For a detailed explanation on the Local Lax-Friedrichs Flux Decomposition we refer to [9]. We then have the following algorithm with the LHHR method:

Algorithm PHM-REF

Step 1: *Computation of Natural Data*

From u^n_j we compute the natural data by means of:

$$v_j = f(u^n_j) \tag{3.2}$$

$$d_{j+\frac{1}{2}} = \frac{v_{j+1} - v_j}{h} \tag{3.3}$$

for all j.

Step 2: *Local Preprocessing of Derivatives*

Computation of $d(j)$ and α_j using LHHR for all j.

Step 3: **for every** j **do**

begin

if $f'(u)$ does not change sign between u_j^n and u_{j+1}^n **then**

Upwindness Phase

(UP1) $\overline{a}_{j+\frac{1}{2}} = \frac{v_{j+1}-v_j}{u_{j+1}^n-u_j^n}$ (Roe speed)

if $\overline{a}_{j+\frac{1}{2}} \geq 0$ **then**

(UP21) $\hat{f}_{j+\frac{1}{2}} = v_j + d(j) \cdot h \cdot \eta(\alpha_j)$ (using (2.35))

else

(UP22) $\hat{f}_{j+\frac{1}{2}} = v_{j+1} - d(j+1) \cdot h \cdot \eta(-\alpha_{j+1})$ (using (2.36))

else

Flux Decomposition Phase

(FD1) $M_{j+\frac{1}{2}} = \max_{u_j^n < u < u_{j+1}^n} |f'(u)|$

(FD2) $v_k^+ = \frac{1}{2} \cdot (v_k + M_{j+\frac{1}{2}} \cdot u_k^n), \qquad k = j-1, j, j+1$

(FD3) $d_{k-\frac{1}{2}}^+ = \frac{v_k^+ - v_{k-1}^+}{h}, \qquad k = j, j+1$

(FD4) Computation of $d^+(j)$ and α_j^+ using LHHR from d^+'s.

(FD5) $f^+ = v_j^+ + d^+(j) \cdot h \cdot \eta(\alpha_j^+)$

(FD6) $v_k^- = \frac{1}{2} \cdot (v_k - M_{j+\frac{1}{2}} \cdot u_k^n), \qquad k = j, j+1, j+2$

(FD8) $d_{k+\frac{1}{2}}^- = \frac{v_{k+1}^- - v_k^-}{h}, \qquad k = j, j+1$

(FD9) Computation of $d^-(j)$ and α_j^- using LHHR from d^-'s.

(FD10) $f^- = v_{j+1}^- - d^-(j+1) \cdot h \cdot \eta(-\alpha_{j+1}^-)$

(FD11) $\hat{f}_{j+\frac{1}{2}} = f^+ + f^-$

end

If we use LHR method instead of LHHR we obtain a compressive PHM method that we will refer by (CPHM-REF).

Some remarks are in order:

(1) From the algorithm it follows that numerical fluxes are functions of four variables. This is also true if we use any local reconstruction method instead of LHHR. However, the third order ENO reconstruction in this case (due to Shu and Osher, [9]) gives numerical fluxes which are functions of six variables. That point is crucial in order to distinguish the

behavior between local and nonlocal methods. As we will see in numerical experiments the spreading of "noise" coming from the singularities of the solution is clearly more reduced in local methods.

(2) Since "local smoothing" ignores the information about the smoothness of second differences, then the "corners" of the solution, (jumps in first derivative), are well preserved during the evolution, as numerical experiments show, (see Figs. 3 and 4).

(3) The "pseudocode" that describes the algorithm (PHM-REF) was made for better understanding and it is not optimal in the sense of computational cost. In spite of obvious improvements, (division by h can be avoided), the cost of the algorithm is a bit higher than third order ENO. The cost depends on the computation of the first differences and the evaluation of the parameters d, α and the function η. In order to evaluate η at values close to 0, it may be computationally convenient to use a Padé approximant near 0, because of the removable discontinuity at 0. On the other hand, local methods constructed in this way are stable and accurate for high CFL's (Courant-Friedrichs-Lewy constant), and that reduces computational cost.

(4) In order to sharpen contact discontinuities we have used the Yang's Artificial Compression Method, (see [10]), that is applicable to the (PHM-REF) algorithm, in the version given by Shu and Osher in [9]. We have found that this method works efficiently for "small" CFL's. The interested reader can find details about that method in [10].

(5) Numerically we observe that (PHM-REF) always yields the correct entropy solution even for nonconvex fluxes. See section 4 for some examples.

(6) Scalar multidimensional initial value problems of the form:

$$u_t + \sum_{i=1}^{d} f_i(u)_{x_i} = 0 \tag{3.4}$$

$$u(x,0) = u_0(x) \tag{3.5}$$

are approximated by applying the one-dimensional procedure to each of the terms $f_i(u)_{x_i}$ in (3.4), keeping all other variables fixed. Then the Runge-Kutta method (1.10) is used with CFL numbers shrunk by a factor d^{-1}. Indeed, a classical CFL restriction

$$\frac{\Delta t}{h} \cdot \max_u |f'(u)| \leq \lambda_0$$

will be replaced by

$$\Delta t \cdot \max_u \sum_{i=1}^{d} \frac{1}{h_i} |f_i'(u)| \leq \lambda_0$$

4. Numerical Experiments

We denote by PHM-REF, CPHM-REF and ENO3-REF the algorithm described in section 3, with the LHHR method, the LHR method and the Shu-Osher third order ENO reconstruction method, respectively. If we add the label AC, that means that Yang's Artificial Compression Method, in the version of Shu and Osher (see [9]), has been applied. We have run most examples for different CFL's and time levels, but here we only include what we consider as representatives. Concerning the figures, circles are numerical solutions and solid lines are piecewise-linear functions that interpolate exact solutions computed either at grid points or at points of a finer grid. Three-dimensional figures represent level curves of the numerical solution.

Example 1. We solve the scalar linear conservation law:

$$u_t + u_x = 0 \qquad 0 \le x < 1 \tag{4.1a}$$

with the 1-periodic smooth initial data

$$u(x,0) = \frac{1}{2}\left(\frac{1}{2} + \sin(2\pi x)\right) \tag{4.1b}$$

In Figures 1 and 2 we show the solution of (4.1a),(4.1b) at $t = 1$ with $CFL = 0.8$ and $N = 80$ grid points, for the PHM-REF and CPHM-REF methods, respectively. Let us observe the compressive character of CPHM at local extrema. In Table 4 we list the L_∞-error and L_1-error for a refinement sequence with $N = 20, 40, 80, 160$, for both methods, comparing with the exact solution. We observe that both methods are $O(h^{\frac{3}{2}})$ accurate in the L_∞-norm, because of the loss of accuracy at local extrema. In Table 5, numerical orders, r_{20} and r_{40}, at 20 points are shown for the method CPHM-REF, computed by Richardson extrapolation, on 20 and 40 grid points, taken as starting grid, respectively. Figures for PHM-REF are similar. Numerically, both methods become $O(h^{\frac{3}{2}+\frac{1}{p}})$ accurate in L_p. Let us observe that the accuracy is recovered outside a neighbourhood of local extrema. Thus, the behavior is clearly better than TVD methods.

Example 2. We consider the following periodic initial value problem:

$$u_t + u_x = 0 \qquad -1 \le x < 1 \tag{4.2a}$$

$$u(x,0) = \begin{cases} \sin\left(\pi \cdot \frac{x+0.3}{0.6}\right), & \text{if } -0.3 \le x \le 0.3; \\ 0, & \text{otherwise.} \end{cases} \tag{4.2b}$$

We solve (4.2a),(4.2b) at t=2 and $CFL = 0.8$. In Table 6 we list the L_∞-error and L_1-error for a refinement sequence with $N = 20, 40, 80, 160$, for PHM-REF and CPHM-REF methods, comparing with the exact solution. We observe that both methods retain first order accuracy in the L_∞-norm, in spite of the presence of two jumps in the derivative of the solution, showing the good resolution of corners in both methods. In Table 7 we show numerical orders, r_{20} and r_{40}, computed at 20 points as in Example 1. Numerical orders marked with * correspond to transition points of jumps of the first derivative of the solution and they are meaningless. We have tested our schemes with longer time levels, showing a good behavior for high CFL's and the loss of accuracy at local extrema is confined to a stable neighborhood of singularities. We show in Figure 3 the behavior of PHM-REF for $CFL = 0.8$ at t=4 and it is compared with the Shu-Osher third order ENO shown in Figure 4. Figures 5 and 6 show solutions of the same problem at $t = 4$ with $CFL = 0.2$, and Figure 7 shows the PHM-REF method with Yang's artificial compression, (see [9],[10]). In Figure 8 we show the solution by CPHM-REF with "natural compression", and $CFL = 0.8$, based on the substitution of the LHR reconstruction with more compressive hyperbolas, near local extrema, which will be treated in future papers.

Example 3. We consider the following periodic initial value problem:

$$u_t + u_x = 0 \qquad -1 \le x < 1 \tag{4.3a}$$

$$u(x,0) = \begin{cases} 1, & \text{if } -0.2 \le x \le 0.2; \\ 0, & \text{otherwise.} \end{cases} \tag{4.3b}$$

We solve (4.3a),(4.3b) at t=4. Figures 9 and 10 show the solution with $CFL = 0.2$ for the CPHM-REF and PHM-REF, respectively. In Figure 11 the behavior of PHM-REF with $CFL = 0.8$ is shown. The artificial compression method is applied toghether with PHM-REF (with $CFL = 0.2$), to solve the same problem, and the result is shown in Figure 12.

Example 4. The PHM-REF scheme is used to solve the nonlinear Burgers' equation with periodic smooth initial conditions:

$$u_t + \left(\frac{u^2}{2}\right)_x = 0 \qquad -1 \le x < 1 \tag{4.4a}$$

$$u(x,0) = \frac{1}{2}\left(\frac{1}{2} + \sin(\pi x)\right) \tag{4.4b}$$

The exact solution is smooth up to $t = \frac{2}{\pi}$, then it develops a moving shock, interacting with the rarefaction waves. We compute the exact solution by using a Newton iteration,

(see [5]). We show in Figures 13,14,15 and 16 the evolution at $t = 0.3$, at t=0.64 (near shock transition), at $t = 0.9$ and at $= 1.1$, when the reaction between the shock and the rarefaction waves is over.

Example 5. The PHM-REF scheme is tested with the nonconvex flux of the well known Buckley-Leverett example:

$$u_t + (f(u))_x = 0 \qquad -1 \leq x < 1 \tag{4.5a}$$

where

$$f(u) = \frac{4u^2}{4u^2 + (1-u)^2} \tag{4.5b}$$

with periodic initial data

$$u(x,0) = 1, \text{in } \left[-\frac{1}{2}, 0\right], \text{and } u = 0, \text{elsewhere} \tag{4.5c}$$

Our scheme resolves the correct solution well, with less viscosity than ENO3-REF. Numerical experiments with PHM-REF and ENO3-REF at $t = 0.2$ and $t = 0.4$ are shown in Figures 17, 18, 19 and 20, $(CFL = 0.1)$.

Example 6. We test the PHM-REF scheme with two Riemann problems for the following nonconvex flux example:

$$u_t + (f(u))_x = 0 \tag{4.6a}$$

where

$$f(u) = \frac{1}{4}(u^2 - 1)(u^2 - 4) \tag{4.6b}$$

with the following initial data:

$$u(x,0) = -3, \text{ if } x \leq 0 \text{ and } u = 3 \text{ if } x > 0, \tag{4.6c}$$

$$u(x,0) = 2, \text{ if } x \leq 0 \text{ and } u = -2 \text{ if } x > 0, \tag{4.6d}$$

Our scheme gives also the correct solution well, with less viscosity than ENO3-REF. Numerical experiments with PHM-REF and ENO3-REF for the Riemann problem (4.6c)

at $t = 0.04$ are shown in Figures 21 and 22. The Riemann problem (4.6d) is solved for both methods at $t = 0.2$, and Figures 23 and 24 show the numerical solutions.

Example 7. We consider the 2D linear equation

$$u_t + u_x + u_y = 0 \qquad -1 \le x, y \le 1 \tag{4.7a}$$

where u is periodic in x and y with periods 2. In order to study the behavior of PHM-REF scheme with contact discontinuities and corners, we will test the following three initial data:

$$u(x, y, 0) = \begin{cases} 1, & \text{if } (x, y) \in S; \\ 0, & \text{otherwise.} \end{cases} \tag{4.7b}$$

where

$$S = \{(x, y) : |x - y| < \frac{1}{\sqrt{2}},\ |x + y| < \frac{1}{\sqrt{2}}\ \}$$

is a unit square rotated by an angle of $\frac{\pi}{4}$, (due to Harten, see [8], Example 3).

$$u(x, y, 0) = \begin{cases} 1 - w, & \text{with } w = 2(x^2 + y^2), \text{ if } w \le 1; \\ 0, & \text{otherwise.} \end{cases} \tag{4.7c}$$

$$u(x, y, 0) = \begin{cases} 1 - w, & \text{with } w = 2(x^2 + 2y^2), \text{ if } w \le 1; \\ 0, & \text{otherwise.} \end{cases} \tag{4.7d}$$

We have used for the three examples $\Delta x = \Delta y = \frac{1}{20}$. We have tested the example (4.7b) with PHM-REF and CPHM-REF schemes with $CFL = 0.4$ at $t = 2$ in order to study the smearing and stability of our methods. In Figures 25, 26, 27, 28 and 31 numerical results are displayed, including cross sections $y = 0$ and $y = -0.4$. Figures 29, 30 and 32 show the effect of Yang's artificial compression.

The example (4.7c) is a circular paraboloid truncated by the (x, y)-plane. We have tested CPHM-REF scheme with $CFL = 0.4$ at $t = 2$ and Figures 33, 34, 35 and 36 show the good resolution of corners for different cross sections. We also test PHM-REF with Yang's artificial compression at the same time with $CFL = 0.1$, and the corresponding numerical results are shown in Figures 37, 38, 39 and 40.

The third initial data (4.7d) is an elliptic paraboloid truncated by the (x, y)-plane. We use this example in order to study the smearing of corners, (jumps in first partial derivatives), in different directions. Figures 41, 42, 43 and 44 represent the numerical solution for CPHM-REF method at $t = 2$ with $CFL = 0.2$ corresponding to the following cross sections: $y = 0$, $y = -0.4$, $x = 0$ and $x = -0.4$. We observe that our method does smear differently in the x and y directions due to the asymetry of the initial function.

Example 8. We solve a Riemann problem for the 2D Burgers' equation

$$u_t + \left(\frac{u^2}{2}\right)_x + \left(\frac{u^2}{2}\right)_y = 0 \tag{4.8a}$$

with the following Riemann data:

$$u(x,y,0) = \begin{cases} u_1, & \text{if } x > 0, y > 0; \\ u_2, & \text{if } x < 0, y > 0; \\ u_3, & \text{if } x < 0, y < 0; \\ u_4, & \text{if } x > 0, y < 0; \end{cases} \tag{4.8b}$$

We represent a 2D Riemann data by a four component vector (u_1, u_2, u_3, u_4), defining a step function by means of (4.8b). There are eight essentially different solution types, depending on the orders of the numbers u_i's, (see [11] for details). We have tested PHM-REF scheme and we have observed that it converges to the entropy solution with good resolution in all eight cases. Some numerical results are displayed in Figures 45-52. All experiments were performed for 50×50 grid points and using $\Delta x = \Delta y = \frac{1}{20}$. Figures 47, 50, 51 and 52 are viewed from the side of the negative x-axis, (instead of the usual orientation), for better observation of the resolution of the method.

We have also tested PHM-REF for (4.8a) with the periodic smooth data:

$$u(x,y,0) = \frac{1}{4} + \frac{1}{2}\sin\pi\left(\frac{x+y}{2}\right) \qquad -2 \le x, y \le 2$$

We have observed shock transition 'pictures' very similar to the one-dimensional case, hence we omit them.

5. Concluding Remarks

PHM schemes based on fluxes and the Shu-Osher third order TVD Runge-Kutta method seem to work very well in our preliminary numerical tests. The most remarkable property of our method is the good resolution of discontinuities in derivatives. In spite of the loss of accuracy of our schemes at local extrema where they may degenerate to $O(h^{\frac{3}{2}})$, the behavior in presence of discontinuities is stable and the viscosity appears to be lower than ENO3-REF. On the other hand, we have found that some local reconstructions, as LHR, work satisfactorily with contact discontinuities and recover accuracy for high CFL's, showing that for that kind of discontinuities it is not necessary as much smoothing as for shocks in order to prevent the increasing of total variation of the solution.

Acknowledgments

I have learned most on nonlinear PDE and numerical methods from Stan Osher and I have progressed thanks to helpful discussions with Rosa Donat.

Table 1

Third Order TVD Runge-Kutta Scheme (1.10)

α_{1k}	α_{2k}	α_{3k}	β_{1k}	β_{2k}	β_{3k}
1	0	0	1	0	0
3/4	1/4	0	0	1/4	0
1/3	0	2/3	0	0	2/3

Table 2: *Numerical Order of LHR method*

abscisa	right value	left value
0.025	3.027	2.987
0.050	3.045	2.969
0.075	3.063	2.951
0.100	3.081	2.934
0.125	3.103	2.918
0.150	3.133	2.898
0.175	3.187	2.871
0.200	3.318	2.823
0.225	4.892	2.714
0.250	1.985	1.985
0.275	2.714	4.892
0.300	2.823	3.318
0.325	2.871	3.187
0.350	2.898	3.133
0.375	2.918	3.103
0.400	2.934	3.081
0.425	2.951	3.063
0.450	2.969	3.045
0.475	2.987	3.027
0.500	3.007	3.007
0.525	3.027	2.987
0.550	3.045	2.969
0.575	3.063	2.951
0.600	3.081	2.934
0.625	3.103	2.918
0.650	3.133	2.898
0.675	3.187	2.871
0.700	3.318	2.823
0.725	4.892	2.714
0.750	1.985	1.985
0.775	2.714	4.892
0.800	2.823	3.318
0.825	2.871	3.187
0.850	2.898	3.133
0.875	2.918	3.103
0.900	2.934	3.081
0.925	2.951	3.063
0.950	2.969	3.045
0.975	2.987	3.027

Table 3: *Numerical Order of LHHR*

abscisa	right value	left value
0.025	3.051	2.982
0.050	3.080	2.955
0.075	3.103	2.933
0.100	3.122	2.916
0.125	3.143	2.901
0.150	3.174	2.884
0.175	3.231	2.860
0.200	3.372	2.815
0.225	4.131	2.703
0.250	1.994	1.994
0.275	2.703	4.131
0.300	2.815	3.372
0.325	2.860	3.231
0.350	2.884	3.174
0.375	2.901	3.143
0.400	2.916	3.122
0.425	2.933	3.103
0.450	2.955	3.080
0.475	2.982	3.051
0.500	3.015	3.015
0.525	3.051	2.982
0.550	3.080	2.955
0.575	3.103	2.933
0.600	3.122	2.916
0.625	3.143	2.901
0.650	3.174	2.884
0.675	3.231	2.860
0.700	3.372	2.815
0.725	4.131	2.703
0.750	1.994	1.994
0.775	2.703	4.131
0.800	2.815	3.372
0.825	2.860	3.231
0.850	2.884	3.174
0.875	2.901	3.143
0.900	2.916	3.122
0.925	2.933	3.103
0.950	2.955	3.080
0.975	2.982	3.051

Table 4

Numerical Solution of $u_t + u_x = 0,\ 0 \le x < 1$
$u(x,0) = \frac{1}{2}\left(\frac{1}{2} + \sin(2\pi x)\right)$ *at t=1 and* $CFL = 0.8$

	L_∞-error		L_1-error	
N	PHM-REF	CPHM-REF	PHM-REF	CPHM-REF
20	$6.45 \cdot 10^{-2}$	$5.14 \cdot 10^{-2}$	$2.53 \cdot 10^{-2}$	$1.80 \cdot 10^{-2}$
40	$2.36 \cdot 10^{-2}$	$1.79 \cdot 10^{-2}$	$6.90 \cdot 10^{-3}$	$4.50 \cdot 10^{-3}$
80	$8.60 \cdot 10^{-3}$	$6.30 \cdot 10^{-3}$	$1.70 \cdot 10^{-3}$	$1.10 \cdot 10^{-3}$
160	$3.10 \cdot 10^{-3}$	$2.20 \cdot 10^{-3}$	$5.32 \cdot 10^{-4}$	$3.39 \cdot 10^{-4}$

Table 5

Numerical Orders r_{20} and r_{40}, for the CPHM-REF method with the periodic smooth data (4.1b) at $t = 1$ and $CFL = 0.8$

abscisa	20	40	80	160	r_{20}	r_{40}
0	0.2434	0.2501	0.2500	0.2500	5.9465	3.8010
0.0500	0.4109	0.4041	0.4045	0.4045	3.8372	4.4834
0.1000	0.5550	0.5459	0.5438	0.5439	2.1664	4.9464
0.1500	0.6523	0.6599	0.6549	0.6545	0.6077	3.7543
0.2000	0.6935	0.7222	0.7273	0.7260	2.4857	1.9948
0.2500	0.6986	0.7321	0.7437	0.7478	1.5306	1.5175
0.3000	0.6936	0.7206	0.7273	0.7262	1.9913	2.5859
0.3500	0.6462	0.6607	0.6555	0.6545	1.4772	2.4440
0.4000	0.5589	0.5480	0.5438	0.5439	1.3822	5.2062
0.4500	0.4194	0.4049	0.4045	0.4045	5.4191	3.6015
0.5000	0.2566	0.2499	0.2500	0.2500	5.9362	4.1168
0.5500	0.0891	0.0960	0.0955	0.0955	3.8397	4.3775
0.6000	-0.0550	-0.0459	-0.0438	-0.0439	2.1664	4.9771
0.6500	-0.1523	-0.1599	-0.1549	-0.1545	0.6069	3.7488
0.7000	-0.1935	-0.2222	-0.2273	-0.2260	2.4857	1.9934
0.7500	-0.1986	-0.2321	-0.2437	-0.2478	1.5307	1.5180
0.8000	-0.1936	-0.2206	-0.2273	-0.2262	1.9916	2.5836
0.8500	-0.1462	-0.1607	-0.1555	-0.1545	1.4773	2.4412
0.9000	-0.0589	-0.0480	-0.0438	-0.0439	1.3817	5.2256
0.9500	0.0806	0.0951	0.0955	0.0955	5.4153	3.5294
1.0000	0.2434	0.2501	0.2500	0.2500	5.9515	3.7419

Table 6

Numerical Solution of $u_t + u_x = 0$, $-1 \leq x < 1$
$u(x,0) = \sin(\pi \frac{x+0.3}{0.6})$ *for* $-0.3 \leq x \leq 0.3$ *and* $u(x,0) = 0$ *otherwise,*
at t=2 and $CFL = 0.8$

	L_∞-error		L_1-error	
N	PHM-REF	CPHM-REF	PHM-REF	CPHM-REF
20	$4.03 \cdot 10^{-1}$	$3.68 \cdot 10^{-1}$	$2.01 \cdot 10^{-1}$	$1.82 \cdot 10^{-1}$
40	$1.62 \cdot 10^{-1}$	$1.51 \cdot 10^{-1}$	$6.84 \cdot 10^{-2}$	$5.92 \cdot 10^{-2}$
80	$8.40 \cdot 10^{-2}$	$8.06 \cdot 10^{-2}$	$2.96 \cdot 10^{-2}$	$2.69 \cdot 10^{-2}$
160	$4.96 \cdot 10^{-2}$	$4.72 \cdot 10^{-2}$	$1.34 \cdot 10^{-2}$	$1.19 \cdot 10^{-2}$

Table 7

Numerical Orders r_{20} and r_{40}, for the PHM-REF method with the periodic data (4.2b) *at* $t = 2$ *and* $CFL = 0.8$

abscisa	20	40	80	160	r_{20}	r_{40}
-1.0000	0.0002	0.0000	0.0000	0.0000	4.1642	1.1328
-0.9000	-0.0010	0.0001	0.0000	0.0000	3.7317	5.2511
-0.8000	-0.0013	0.0000	0.0000	0.0000	5.8374	2.0182
-0.7000	-0.0017	-0.0002	0.0000	0.0000	3.2136	3.6189
-0.6000	0.0055	-0.0004	0.0000	0.0000	4.0620	3.5202
-0.5000	0.0439	0.0001	-0.0001	0.0000	7.8317	1.5502
-0.4000	0.1363	0.0237	0.0022	0.0000	2.3883	3.2835
-0.3000	0.3083	0.1584	0.0840	0.0496	1.0105	1.1154
-0.2000	0.4920	0.4912	0.4630	0.4938	-5.1506*	-0.1268*
-0.1000	0.5887	0.7995	0.8803	0.8706	1.3835	3.0594
0.0000	0.5965	0.8463	0.9531	0.9826	1.2249	1.8546
0.1000	0.5789	0.7835	0.8764	0.8728	1.1374	4.6636
0.2000	0.4709	0.5058	0.4769	0.4965	0.2686*	0.5628*
0.3000	0.3170	0.1622	0.0765	0.0455	0.8522	1.4685
0.4000	0.1418	0.0199	0.0012	0.0000	2.7052	3.9823
0.5000	0.0427	0.0010	0.0000	0.0000	5.4843	4.4638
0.6000	0.0080	0.0002	0.0000	0.0000	5.3541	4.3739
0.7000	0.0024	0.0001	0.0000	0.0000	4.2692	9.1782
0.8000	0.0019	0.0000	0.0000	0.0000	5.3310	3.8352
0.9000	0.0011	0.0000	0.0000	0.0000	4.5504	3.6096
1.0000	0.0002	0.0000	0.0000	0.0000	4.1642	1.1328

REFERENCES

[1]. A. Harten, *High Resolution Schemes for Hyperbolic Conservation Laws,* J. Comput. Phys., v. 49, (1983) pp. 357-393.

[2]. A. Harten, *On a class of High Resolution Total Variation Stable Finite Difference Schemes,* SIAM J. Numer. Anal., v. 21, (1984) pp. 1-23.

[3]. A. Harten and S. Osher, *Uniformly High Order Accurate Nonoscillatory Schemes I,* SIAM J. Numer. Anal., v. 24, No.2,(1987) pp. 279-309.

[4]. A. Harten, S. Osher, B. Engquist and S. Chakravarthy, *Some Results on Uniformly High-Order Accurate Essentially Non-oscillatory Schemes ,* Applied Numer. Math., v. 2,(1987) pp. 347-377.

[5]. A. Harten, B. Engquist, S. Osher and S. Chakravarthy, *Uniformly High Order Accurate Essentially Non-oscillatory Schemes III,* J. Comput. Phys., v. 71 No. 2, 1987.

[6]. S. Osher and S.R. Chakravarthy, *High Resolution Schemes and the entropy condition,* SIAM J. Numer. Anal., v. 21,(1984) pp. 955-984.

[7]. P.L. Roe, *Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes ,* J. Comput. Phys., v. 43, (1981) pp. 357-372.

[8]. C. W. Shu and S. J. Osher, *Efficient Implementation of Essentially Non-Oscillatory Shock Capturing Schemes,* J. Comput. Phys., v. 77, (1988) pp. 439-471.

[9]. C. W. Shu and S. J. Osher, *Efficient Implementation of Essentially Non-Oscillatory Shock Capturing Schemes II,* J. Comput. Physics, Vol. 83, (1988) pp. 32-78.

[10]. H. Yang, *An Artificial Compression Method for ENO Schemes. The Slope Modification Method.* Preprint.

[11]. D. Wagner, *The Riemann Problem in Two Space Dimensions for a Single Conservation Law,* SIAM J. Math. Anal., v.14, (1983) pp. 534-559.

Figure legends

Figure 1: PHM-REF, (4.1) with 80 grid points, $CFL = 0.8$

Figure 2: CPHM-REF, (4.1) with 80 grid points, $CFL = 0.8$

Figure 3: PHM-REF, (4.2), 100 points, $t = 4.0$, $CFL = 0.8$

Figure 4: ENO3-REF, (4.2), 100 points, $t = 4.0$, $CFL = 0.8$

Figure 5: PHM-REF, (4.2), 100 points, $t = 4.0$, $CFL = 0.2$

Figure 6: CPHM-REF, (4.2), 100 points, $t = 4.0$, $CFL = 0.2$

Figure 7: PHM-REF-AC, (4.2), 100 points, $t = 4.0$, $CFL = 0.2$

Figure 8: CPHM-REF-NC, (4.2), 100 points, $t = 4.0$, $CFL = 0.8$

Figure 9: CPHM-REF, (4.3), 100 points, $t = 4.0$, $CFL = 0.2$

Figure 10: PHM-REF, (4.3), 100 points, $t = 4.0$, $CFL = 0.2$

Figure 11: PHM-REF, (4.3), 100 points, $t = 4.0$, $CFL = 0.8$

Figure 12: PHM-REF-AC, (4.3), 100 points, $t = 4.0$, $CFL = 0.2$

Figure 13: PHM-REF, (4.4), 100 points at $t = 0.3$, $CFL = 0.8$

Figure 14: PHM-REF, (4.4), 100 points at $t = 0.64$, $CFL = 0.8$

Figure 15: PHM-REF, (4.4), 100 points at $t = 0.9$, $CFL = 0.8$

Figure 16: PHM-REF, (4.4), 100 points at $t = 1.1$, $CFL = 0.8$

Figure 17: PHM-REF, (4.5), 100 points at $t = 0.2$

Figure 18: PHM-REF, (4.5), 100 points at $t = 0.4$

Figure 19: ENO3-REF, (4.5), 100 points at $t = 0.2$

Figure 20: ENO3-REF, (4.5), 100 points at $t = 0.4$

Figure 21: PHM-REF, (4.6c), 100 points at $t = 0.04$, $CFL = 0.05$

Figure 45: PHM-REF, (4.8) with $(-1, -0.2, 0.5, 0.8)$, 50×50 points, $t = 1$

Figure 46: PHM-REF, (4.8) with $(-0.2, -1, 0.5, 0.8)$, 50×50 points, $t = 1$

Figure 47: PHM-REF, (4.8) with $(0.5, -1, -0.2, 0.8)$, 50×50 points, $t = 1$

Figure 48: PHM-REF, (4.8) with $(-1, 0.5, -0.2, 0.8)$, 50×50 points, $t = 1$

Figure 49: PHM-REF, (4.8) with $(-1, -0.2, 0.8, 0.5)$, 50×50 points, $t = 1$

Figure 50: PHM-REF, (4.8) with $(0.8, -1, -0.2, 0.5)$, 50×50 points, $t = 1$

Figure 51: PHM-REF, (4.8) with $(0.8, -1, 0.5, -0.2)$, 50×50 points, $t = 1$

Figure 52: PHM-REF, (4.8) with $(0.8, -0.2, -1, 0.5)$, 50×50 points, $t = 1$

Piecewise Hyperbolic Methods

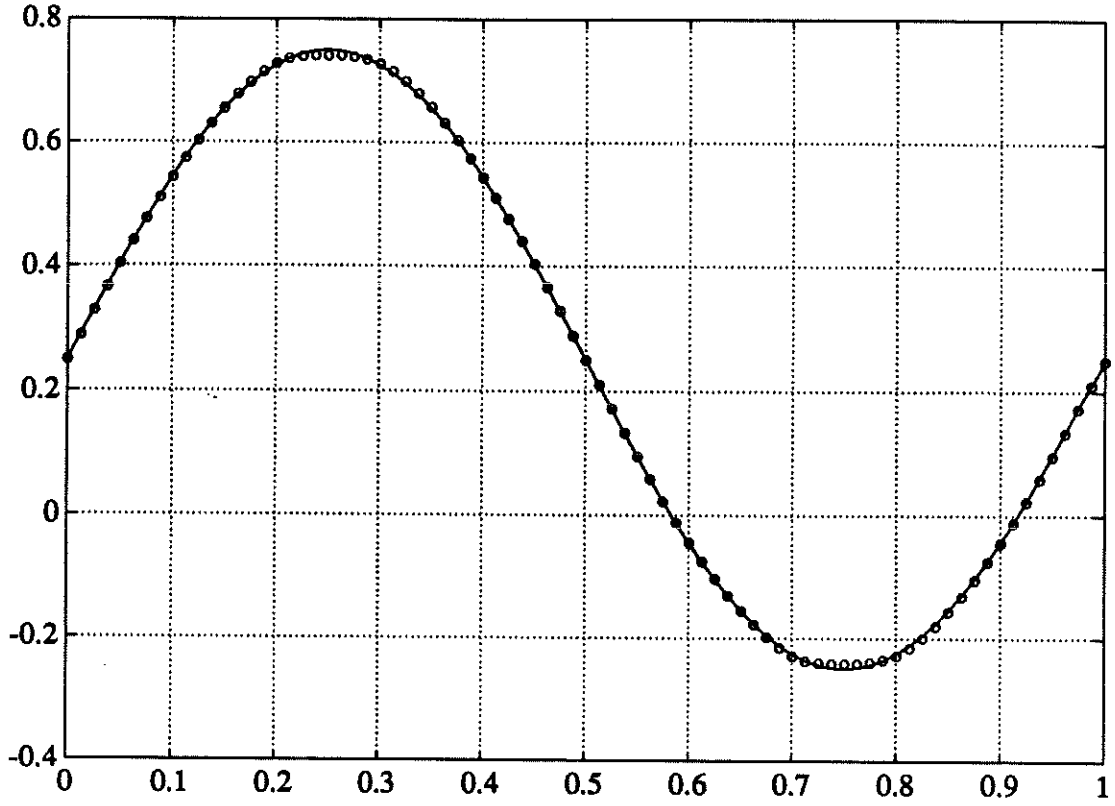

Figure 1: PHM-REF, (4.1) with 80 grid points, CFL=0.8

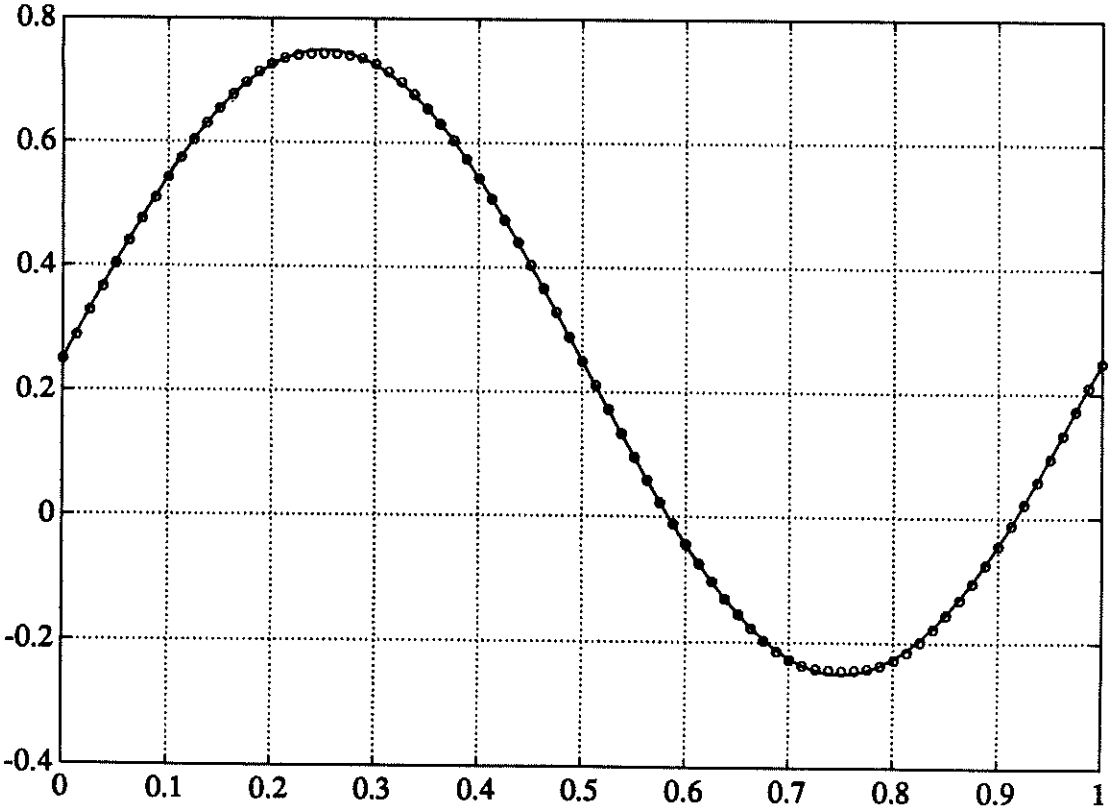

Figure 2: CPHM-REF, (4.1) with 80 grid points, CFL=0.8

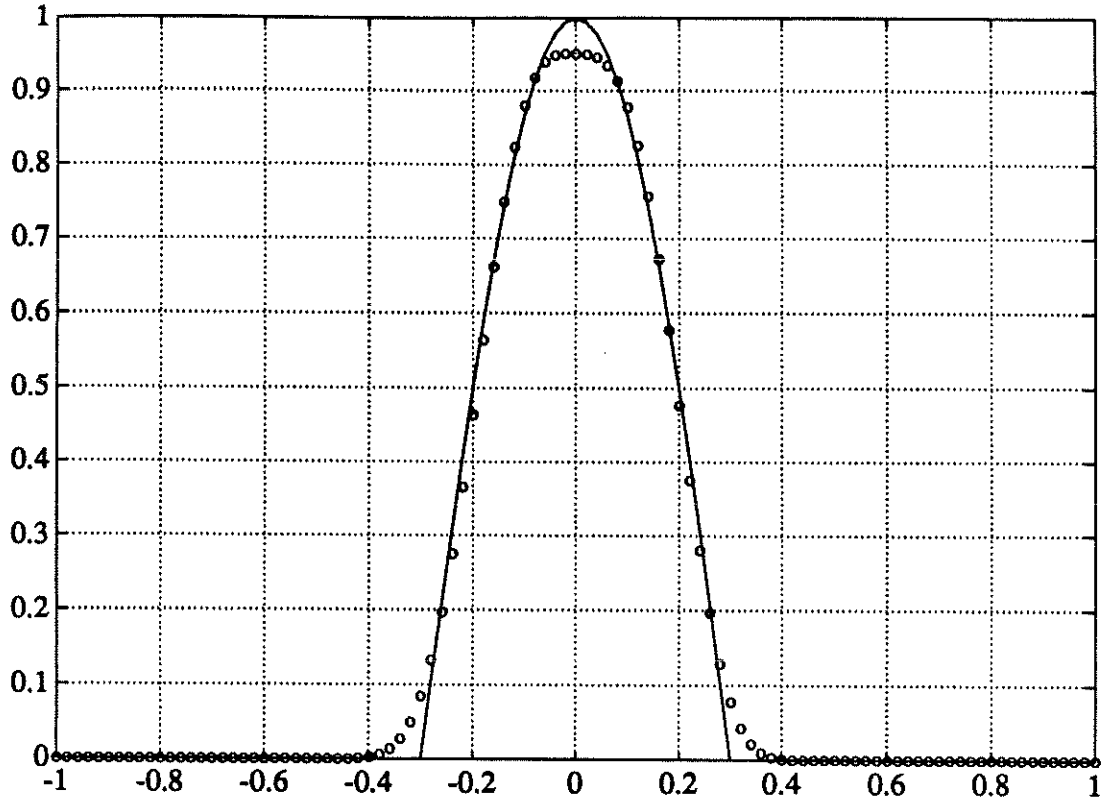

Figure 3: PHM-REF, (4.2) 100 points, t=4.0, CFL=0.8

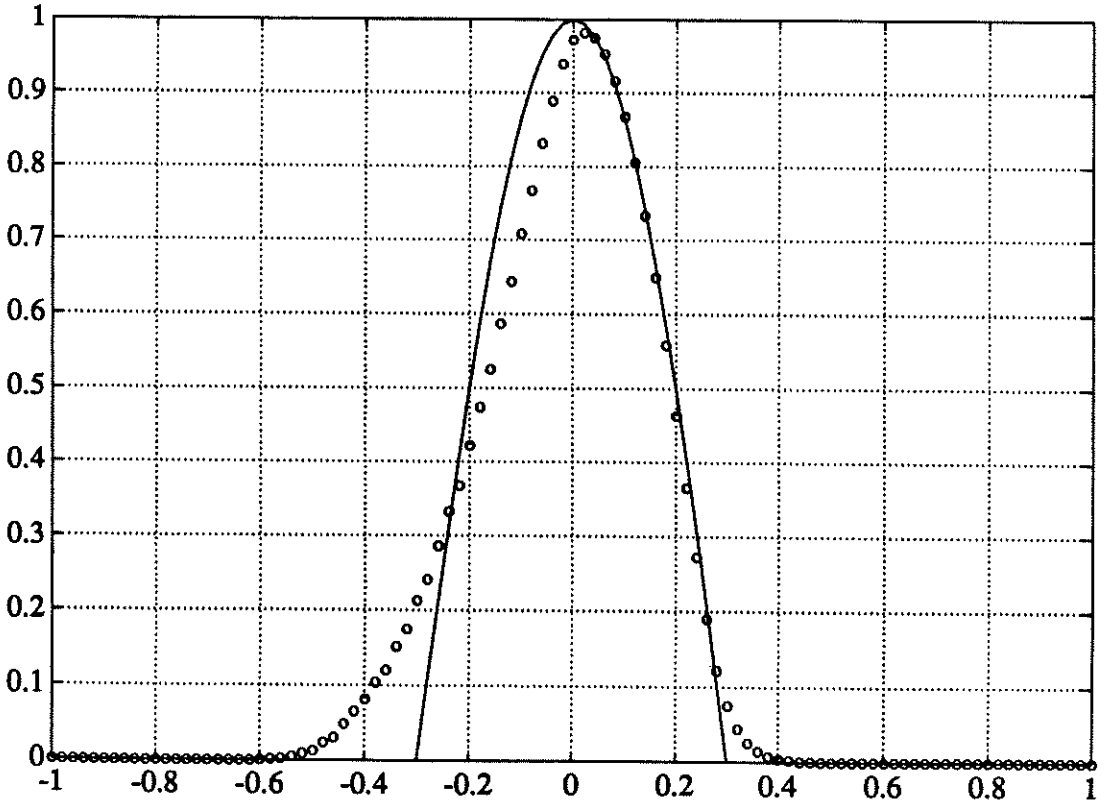

Figure 4: ENO3-REF, (4.2) 100 points, t=4.0, CFL=0.8

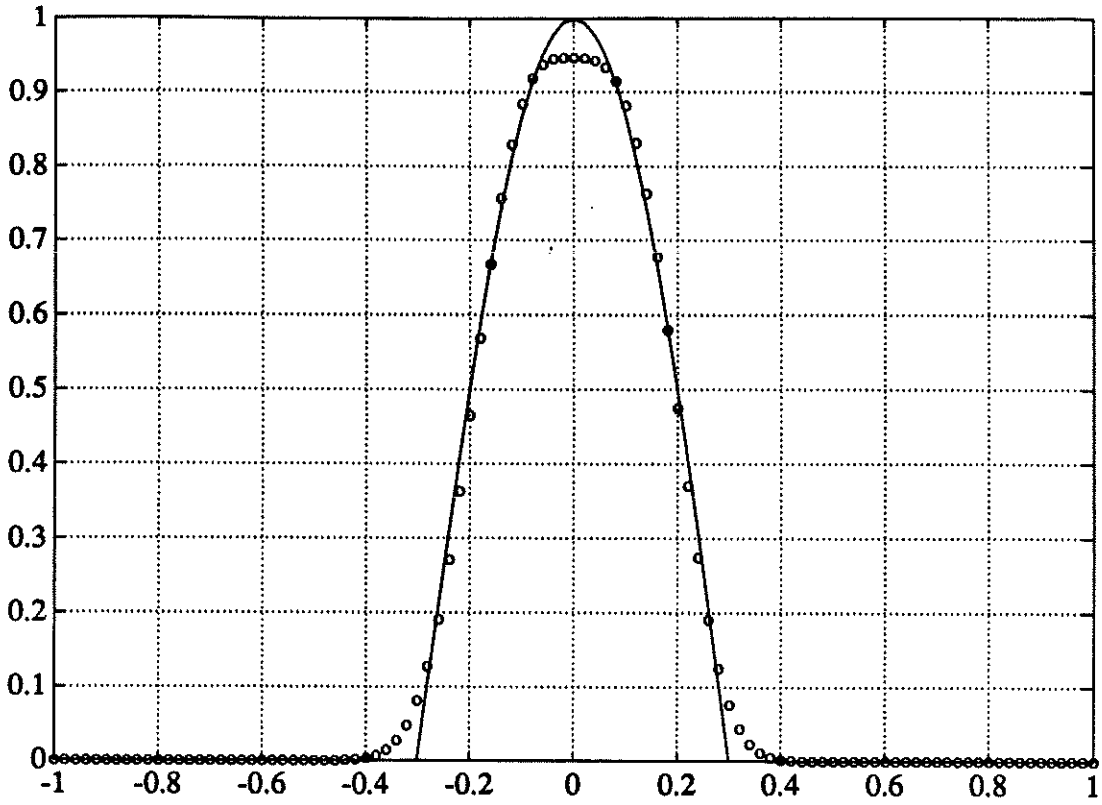

Figure 5: PHM-REF, (4.2) 100 points, t=4.0, CFL=0.2

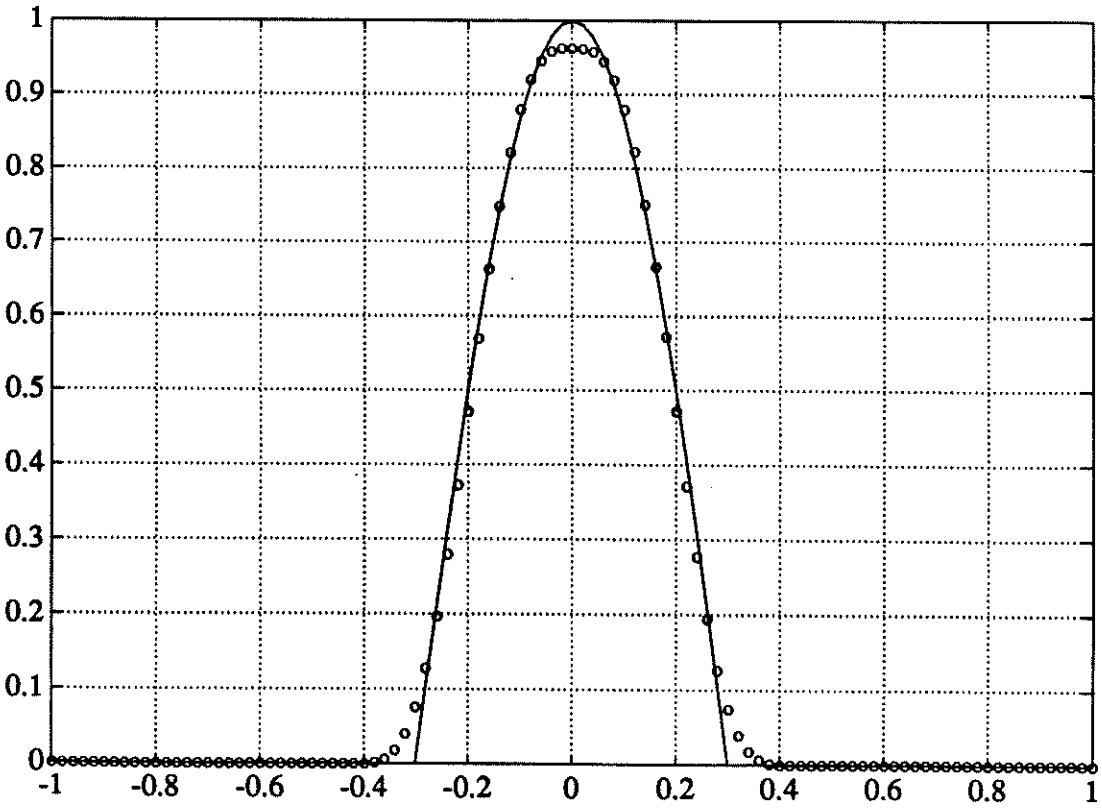

Figure 6: CPHM-REF, (4.2) 100 points, t=4.0, CFL=0.2

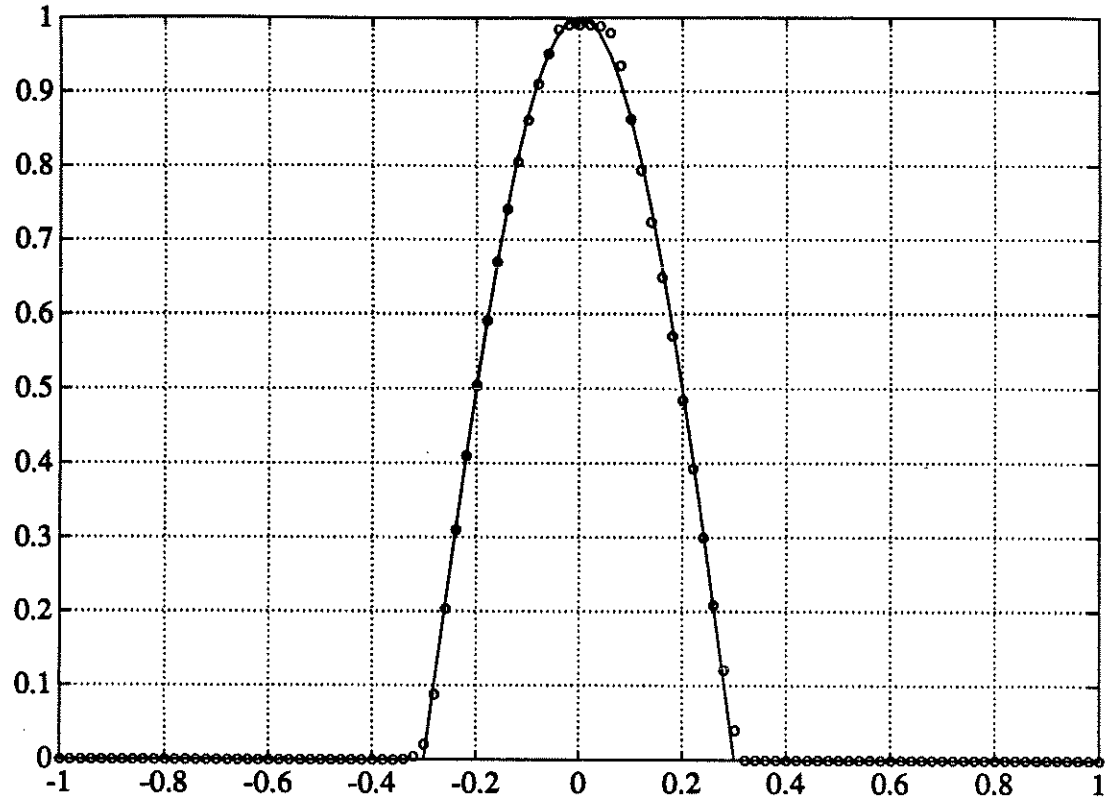

Figure 7: PHM-REF-AC, (4.2), 100 points, t=4.0, CFL=0.2

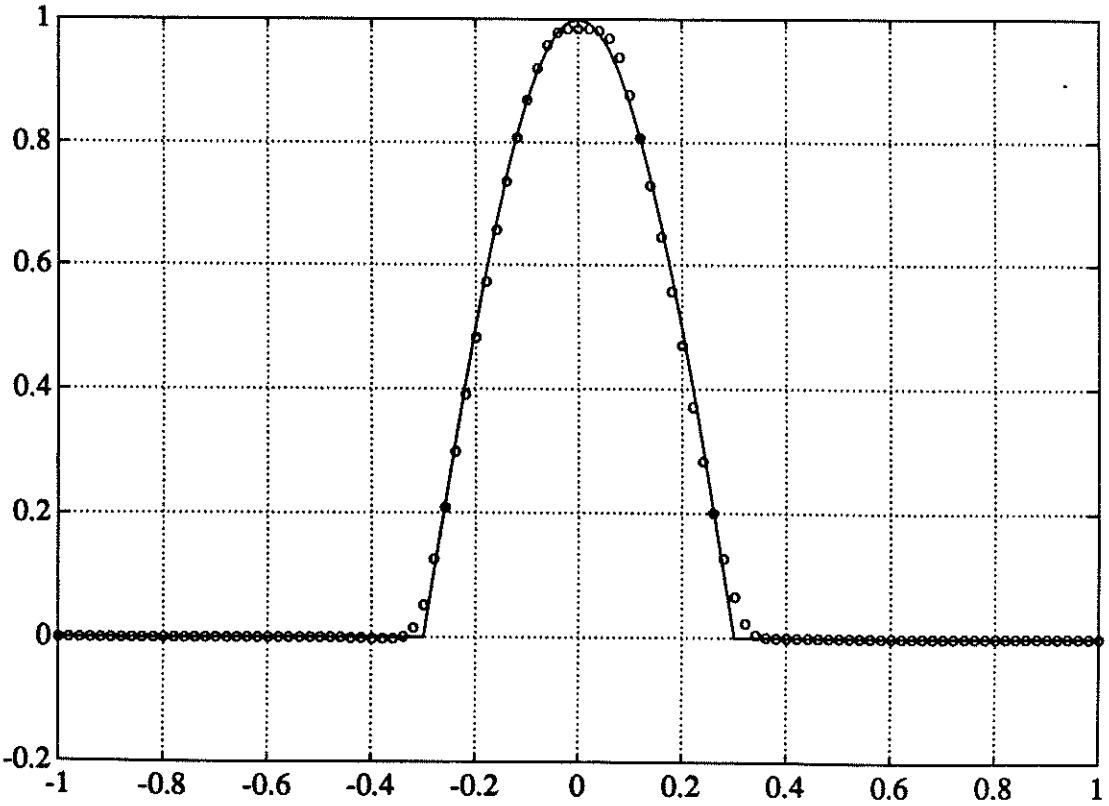

Figure 8: CPHM-REF-NC, (4.2), 100 points, t=4.0, CFL=0.8

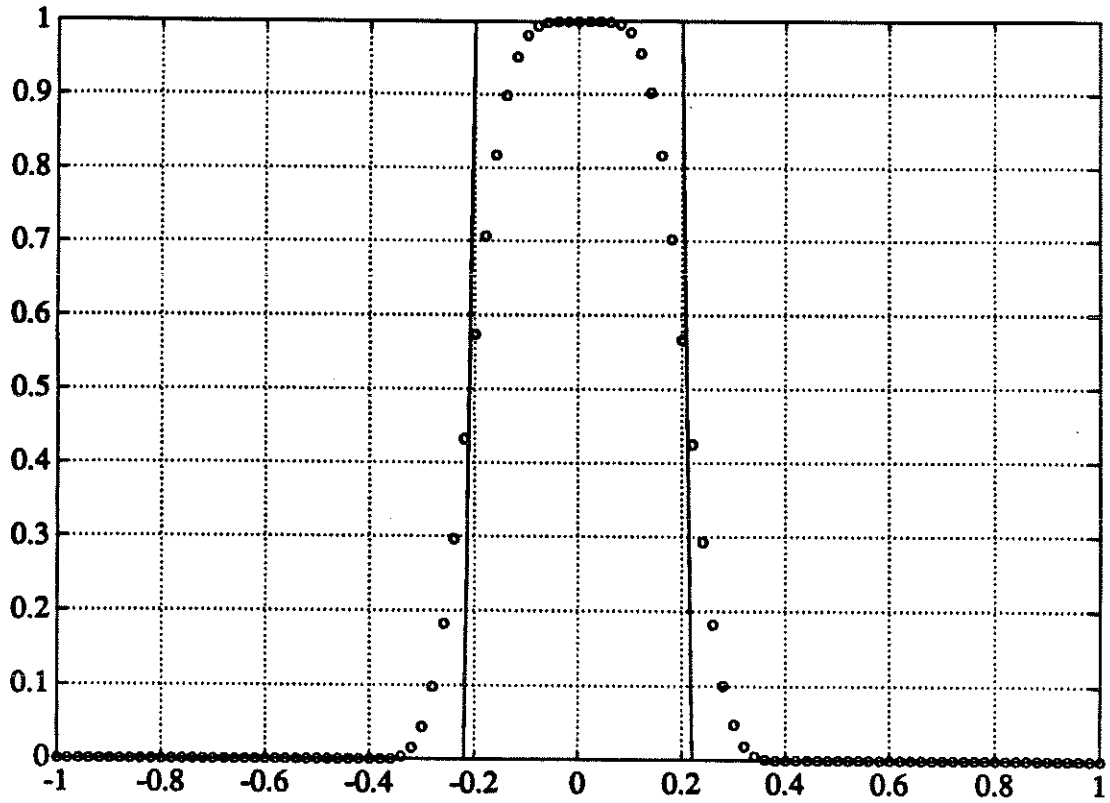

Figure 9: CPHM-REF, (4.3) 100 points, t=4.0, CFL=0.2

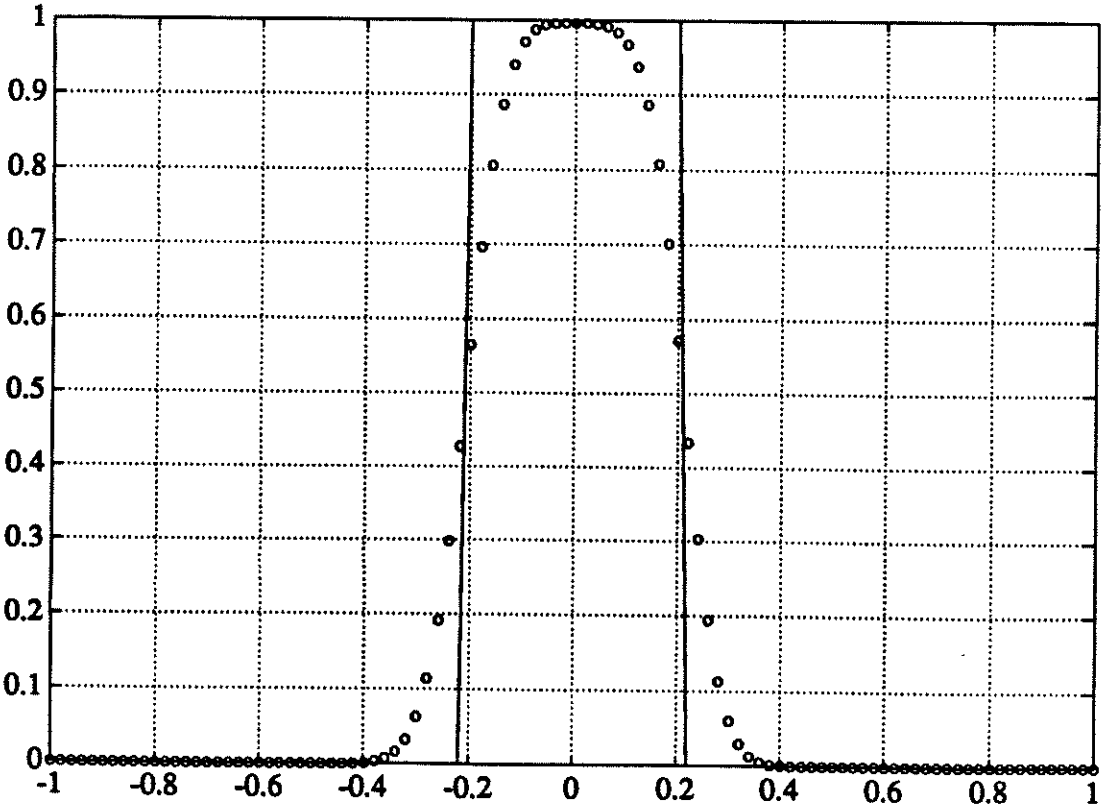

Figure 10: PHM-REF, (4.3) 100 points, t=4.0, CFL=0.2

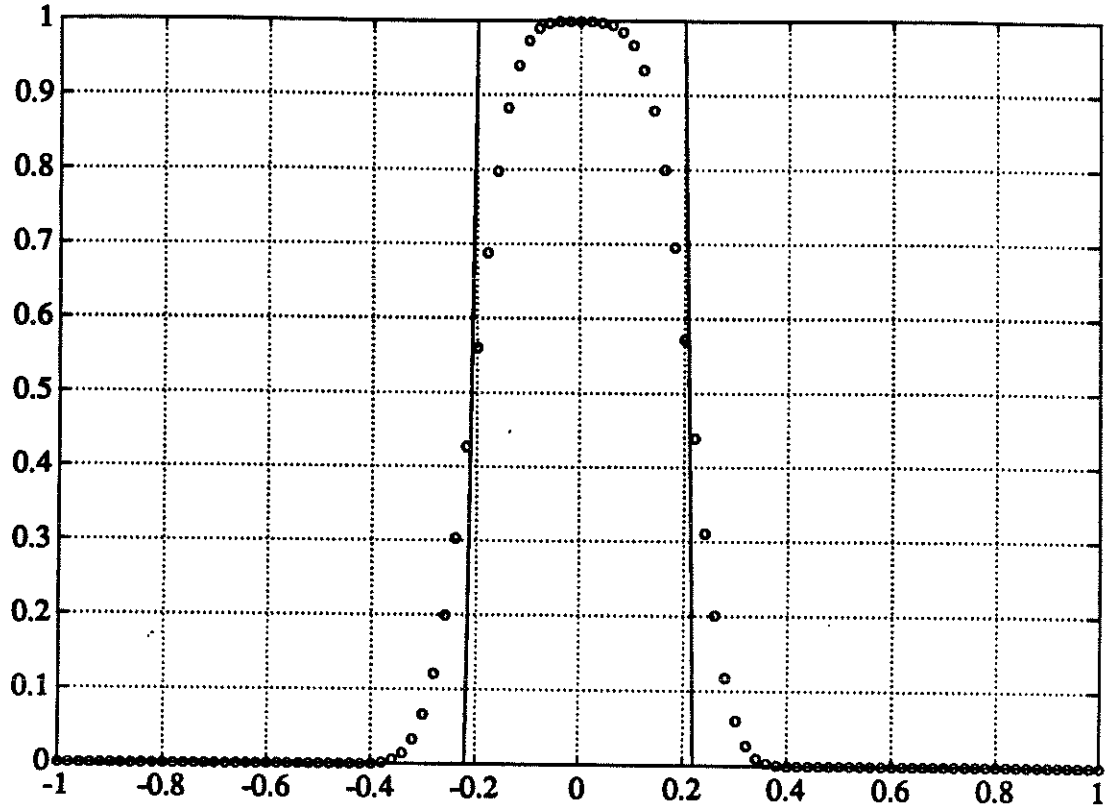

Figure 11: PHM-REF, (4.3) 100 points, t=4.0, CFL=0.8

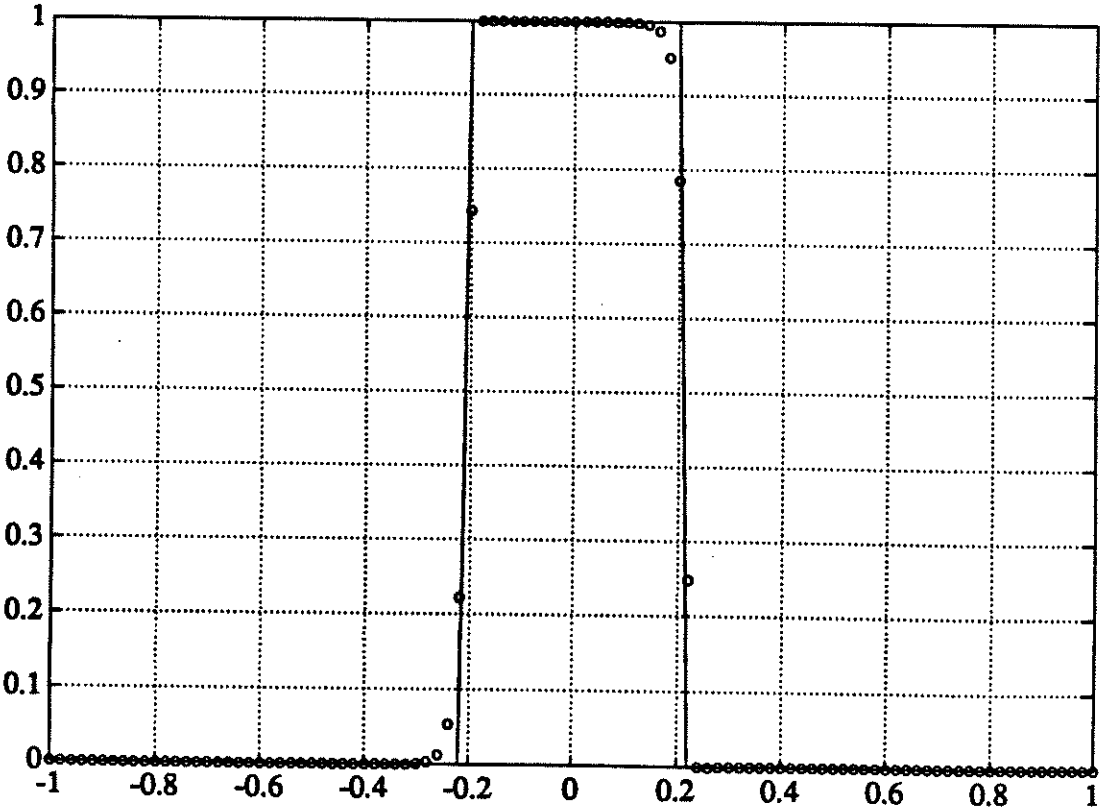

Figure 12: PHM-REF-AC, (4.3) 100 points, t=4.0, CFL=0.2

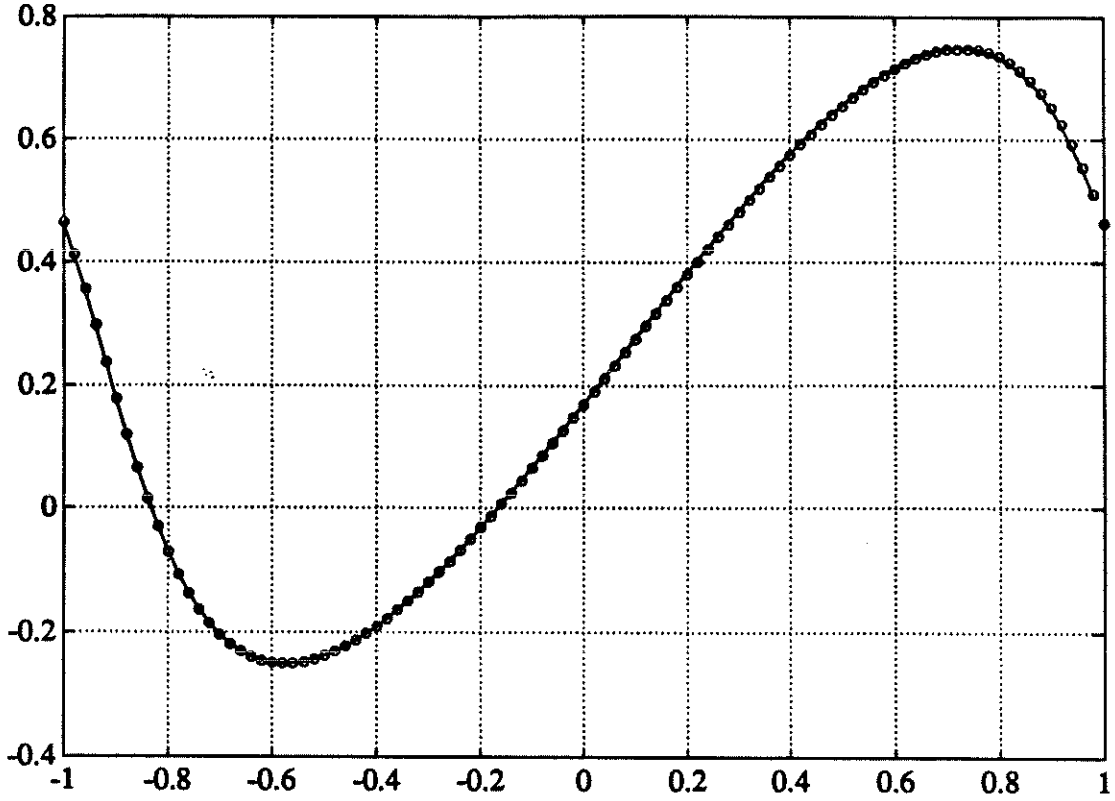

Figure 13: PHM-REF, (4.4), 100 points at t=0.3, CFL=0.8

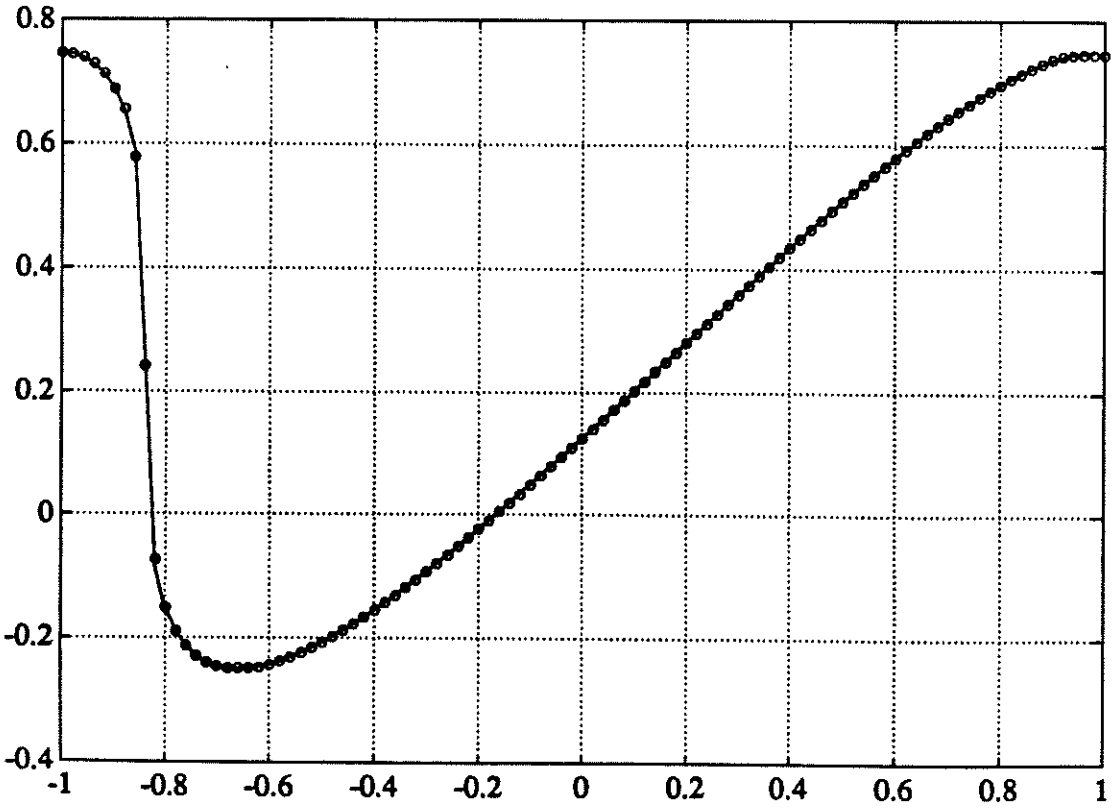

Figure 14: PHM-REF, (4.4), 100 points at t=0.64, CFL=0.8

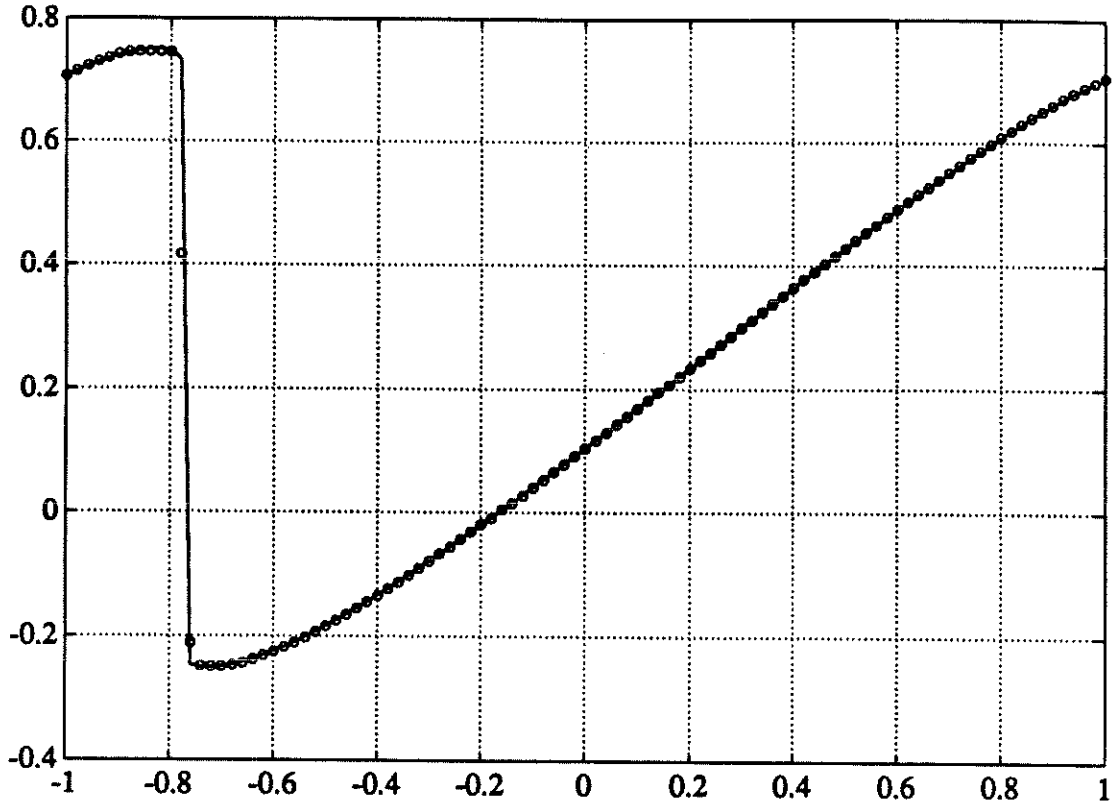

Figure 15: PHM-REF, (4.4), 100 points at t=0.9, CFL=0.8

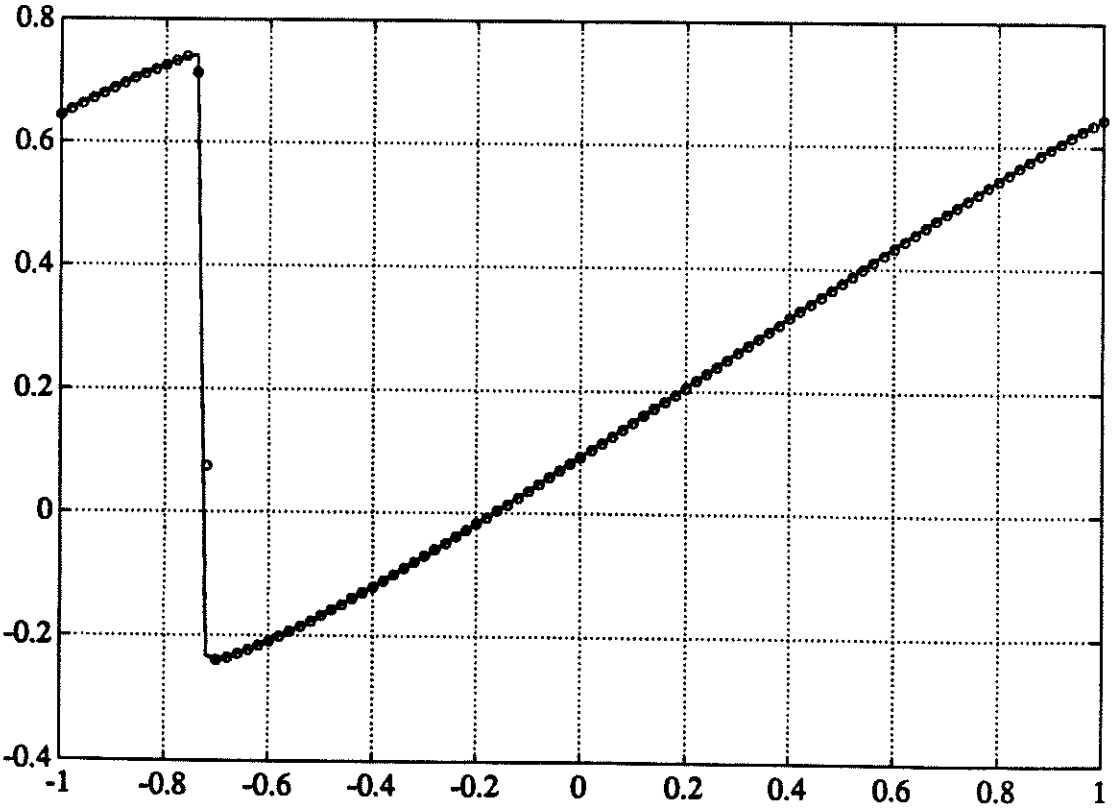

Figure 16: PHM-REF, (4.4), 100 points at t=1.1, CFL=0.8

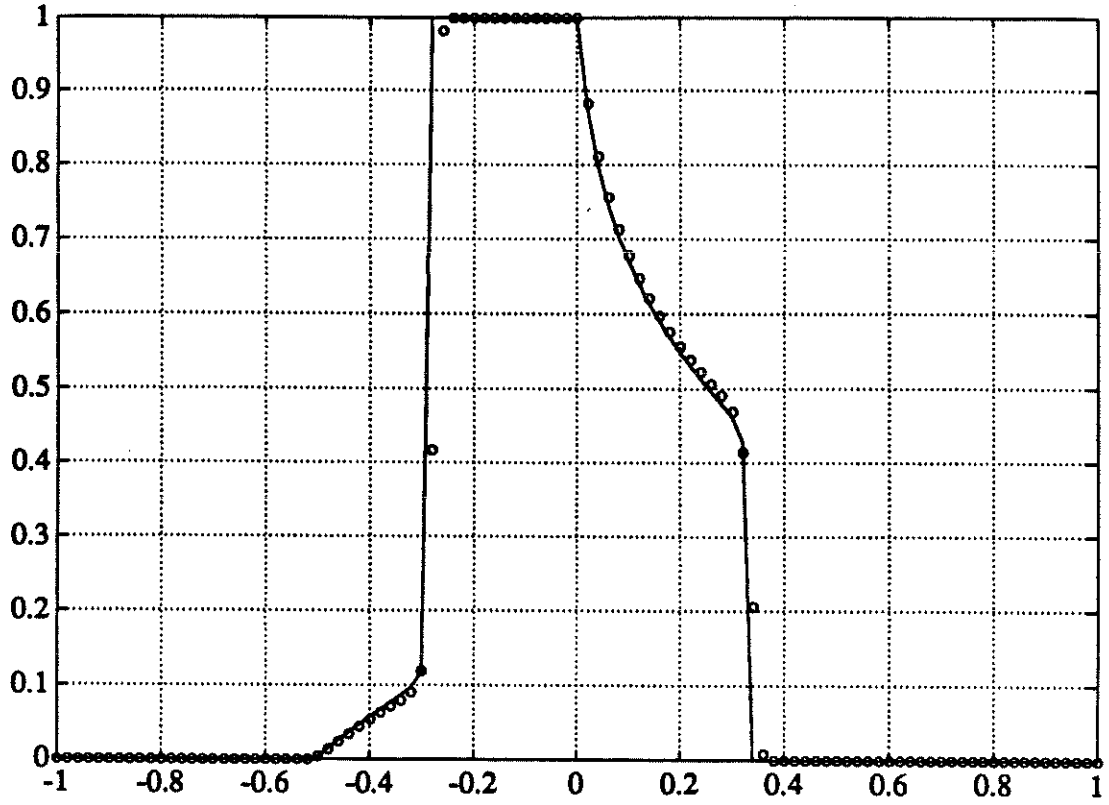

Figure 17: PHM-REF, (4.5), 100 points at t=0.2

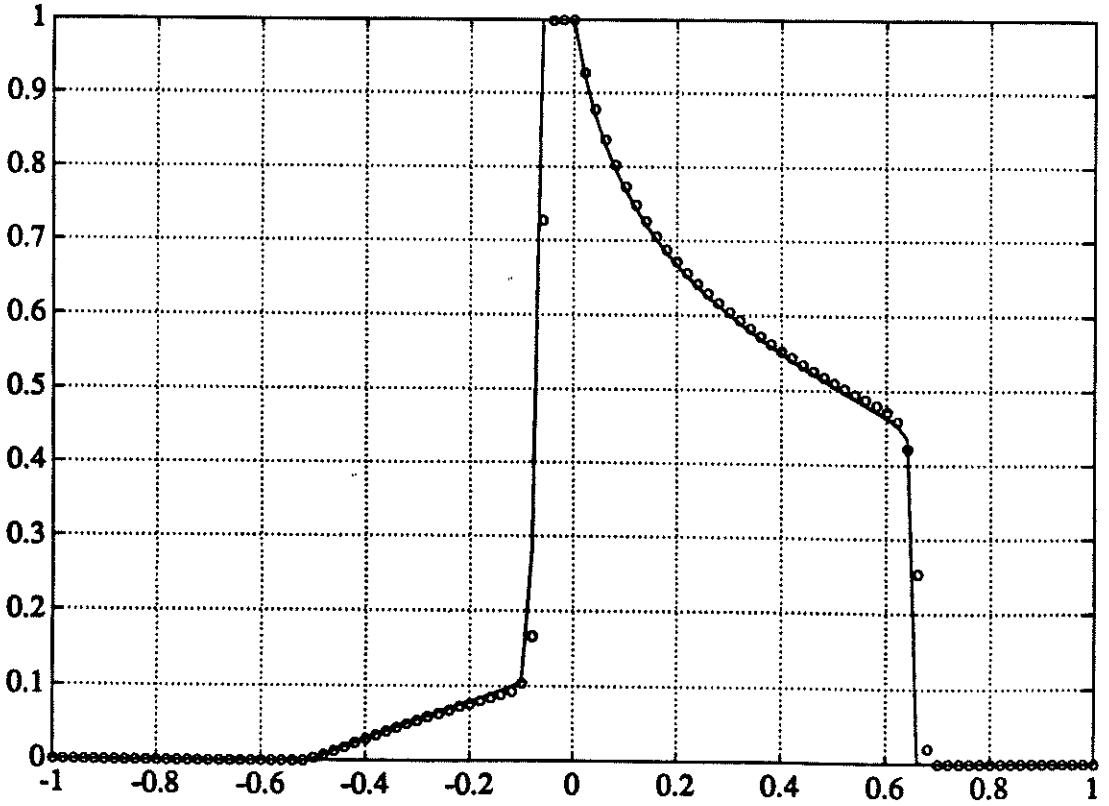

Figure 18: PHM-REF, (4.5), 100 points at t=0.4

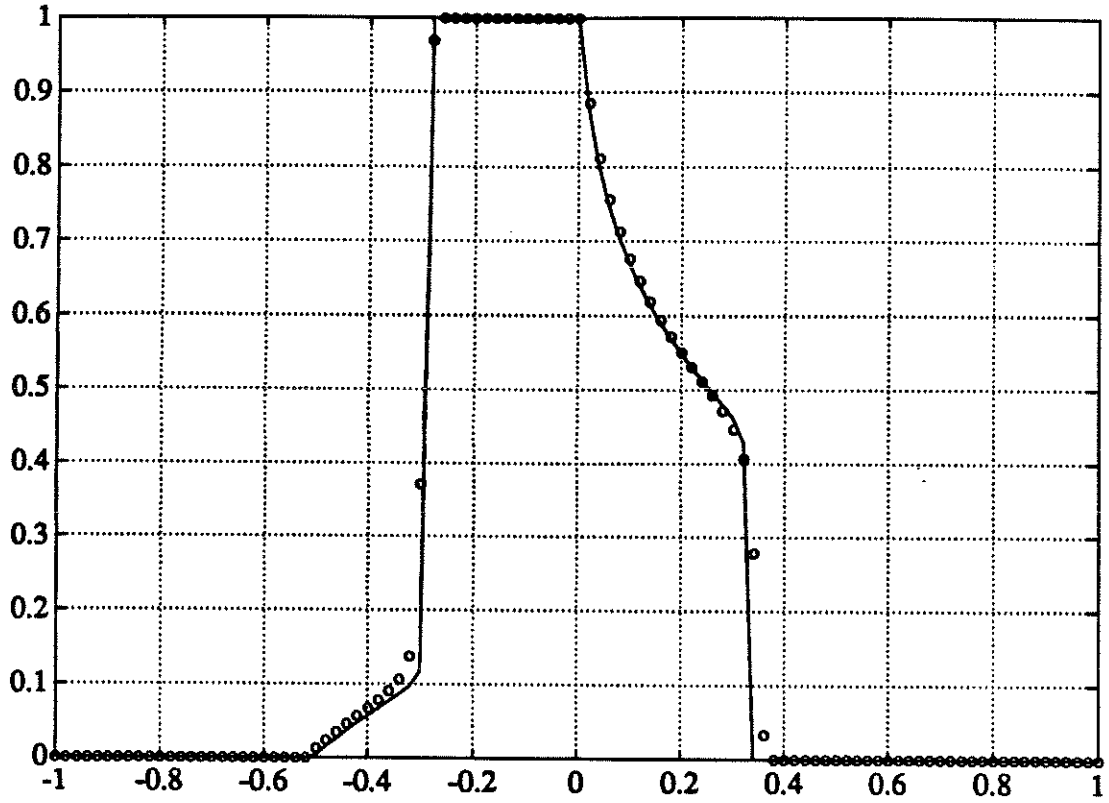

Figure 19: ENO3-REF, (4.5), 100 points at t=0.2

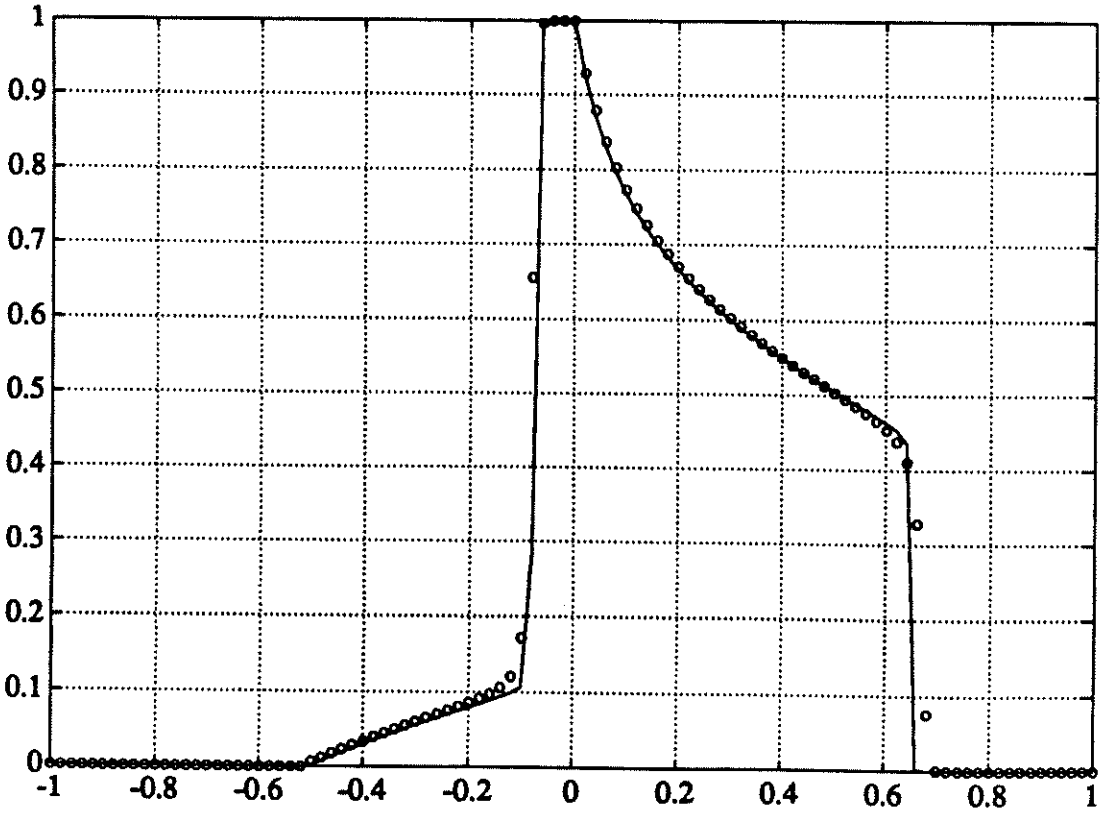

Figure 20: ENO3-REF, (4.5), 100 points at t=0.4

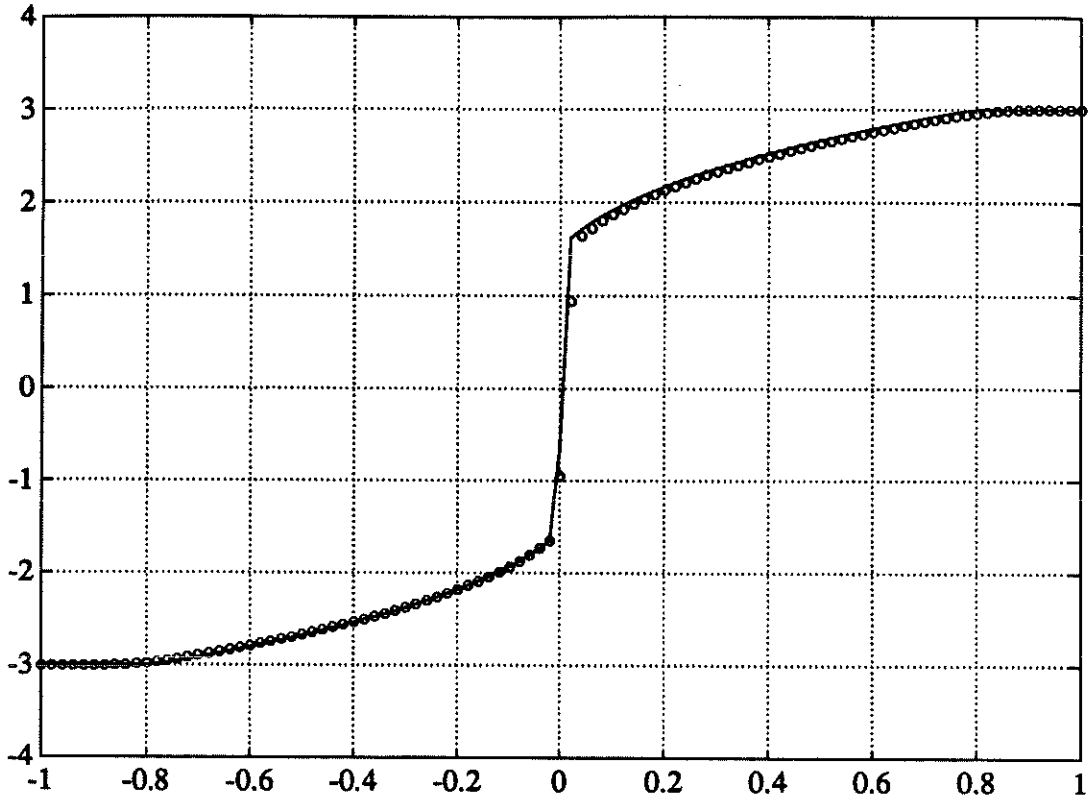

Figure 21: PHM-REF, (4.6c), 100 points at t=0.04, CFL=0.05

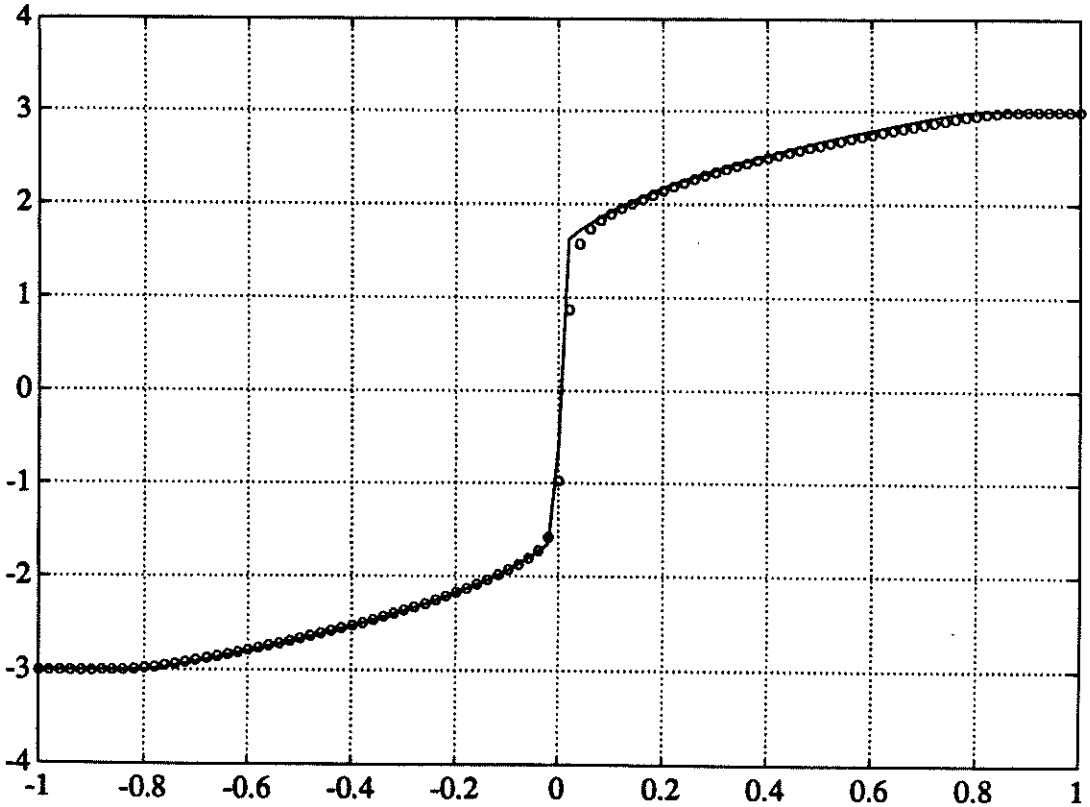

Figure 22: ENO3-REF, (4.6c), 100 points at t=0.04, CFL=0.05

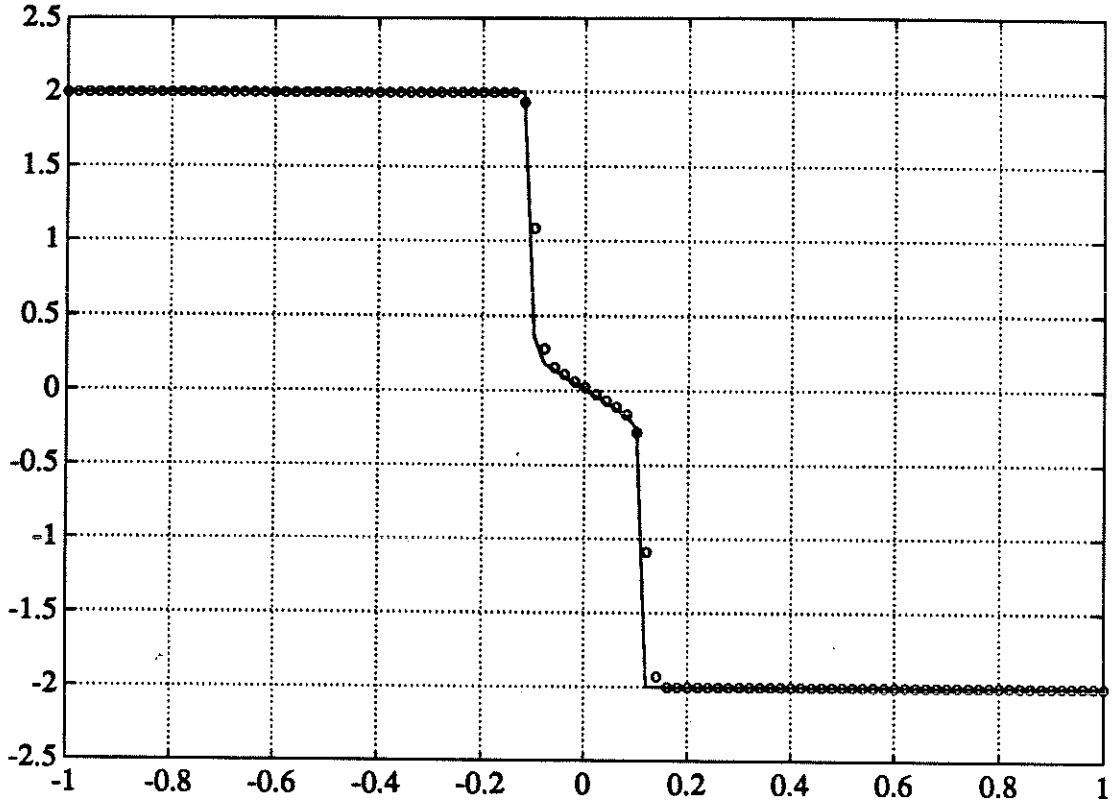

Figure 23: PHM-REF, (4.6d), 100 points at t=0.2, CFL=0.05

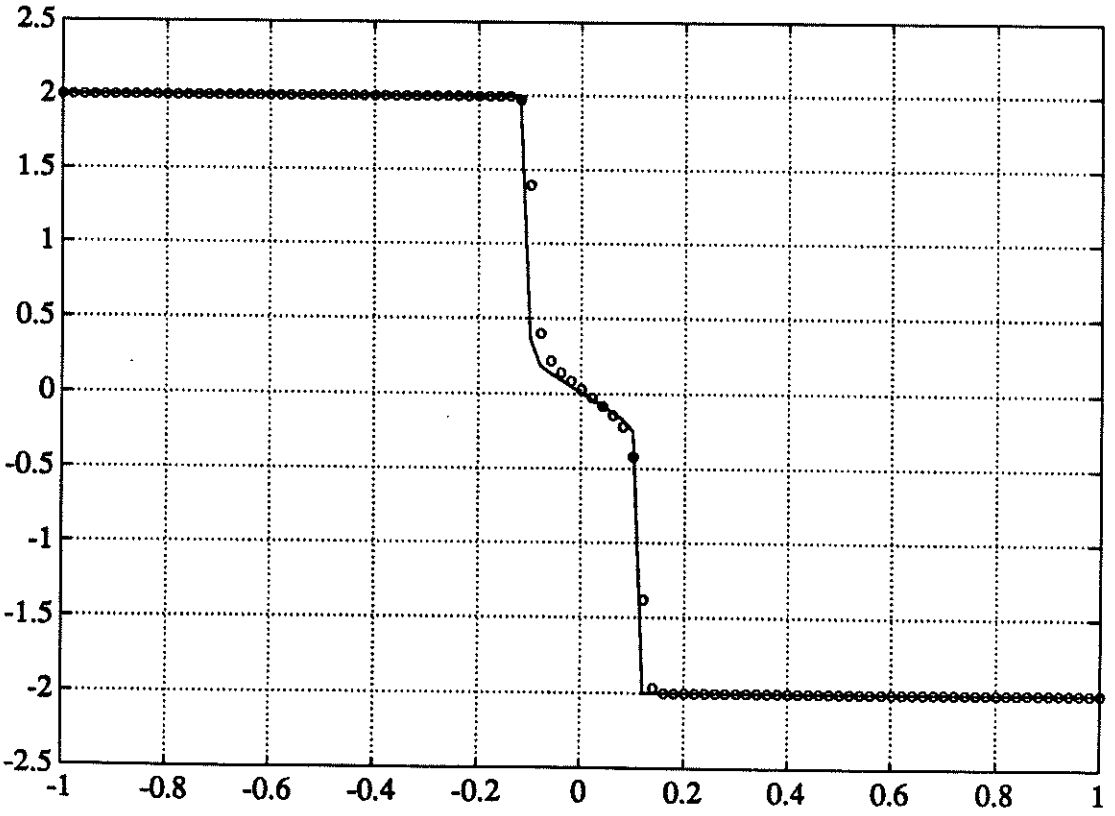

Figure 24: ENO3-REF, (4.6d), 100 points at t=0.2, CFL=0.05

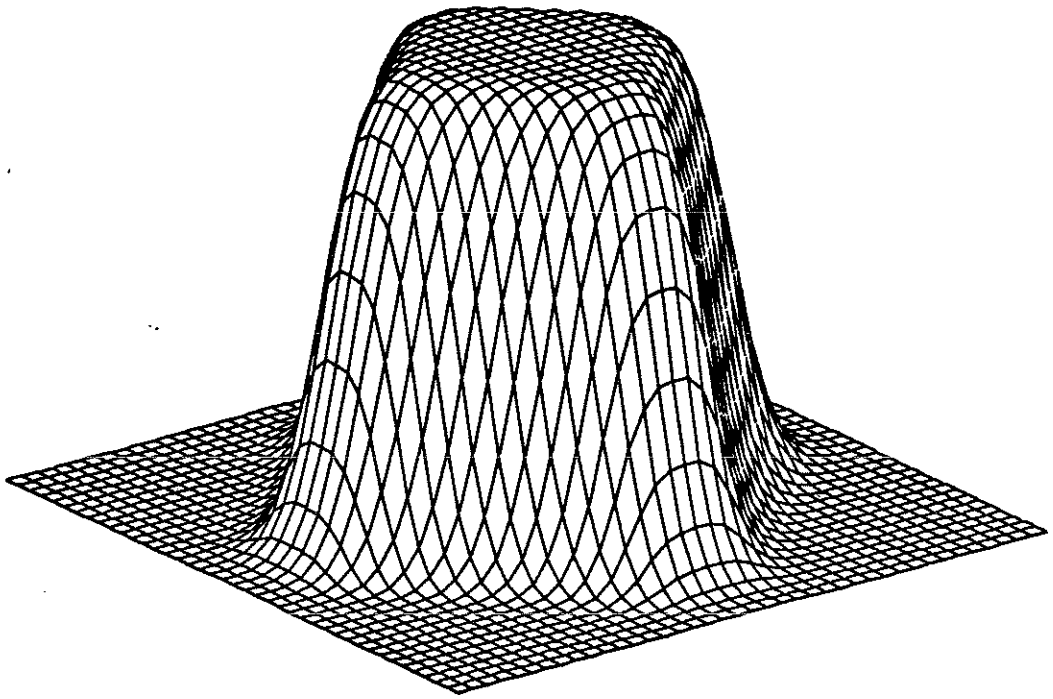

Figure 25: PHM-REF, (4.7b), 40x40 grid points, t=2, CFL=0.4

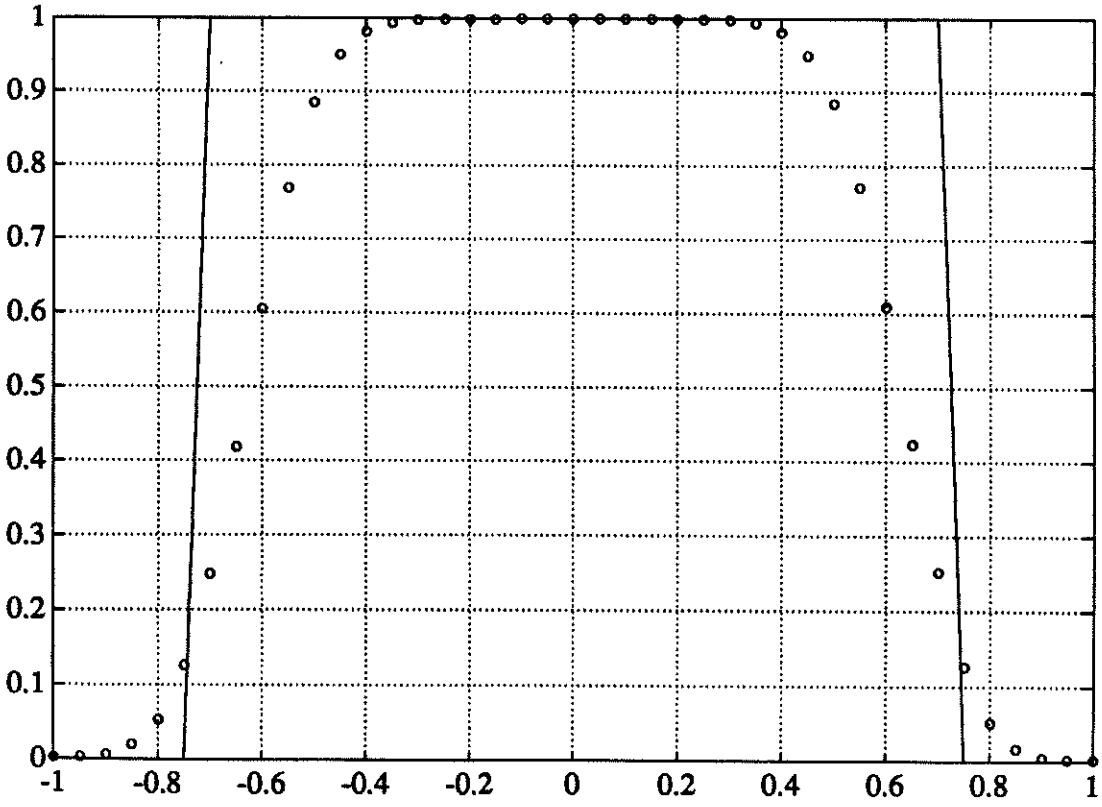

Figure 26: PHM-REF, (4.7b), Section y=0, t=2, CFL=0.4

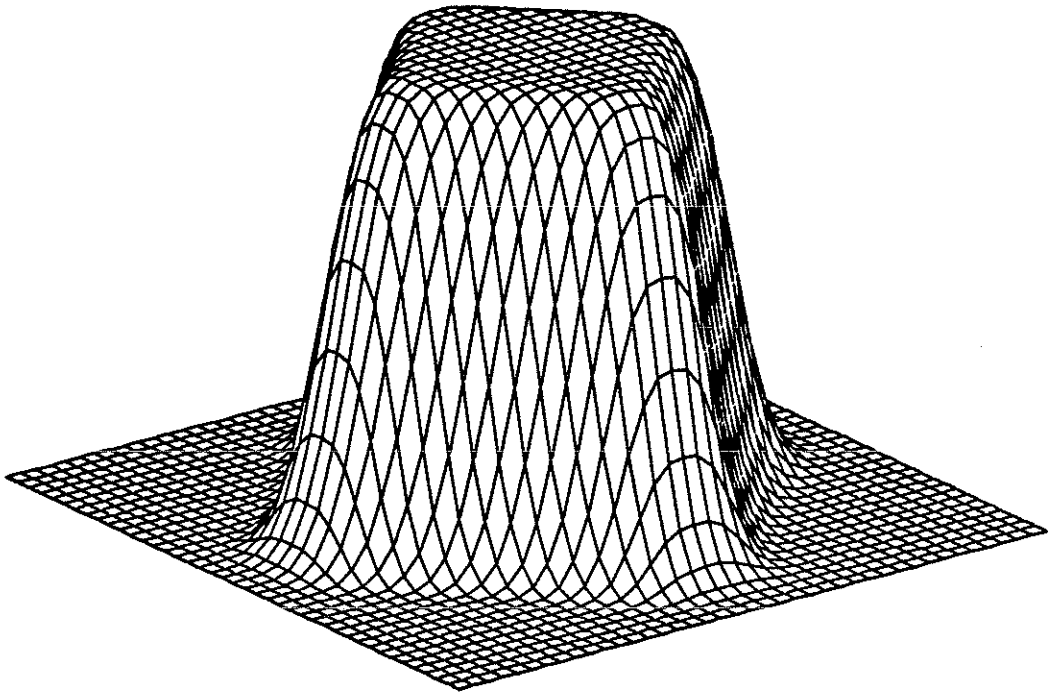

Figure 27: CPHM-REF, (4.7b), 40x40 grid points, t=2, CFL=0.4

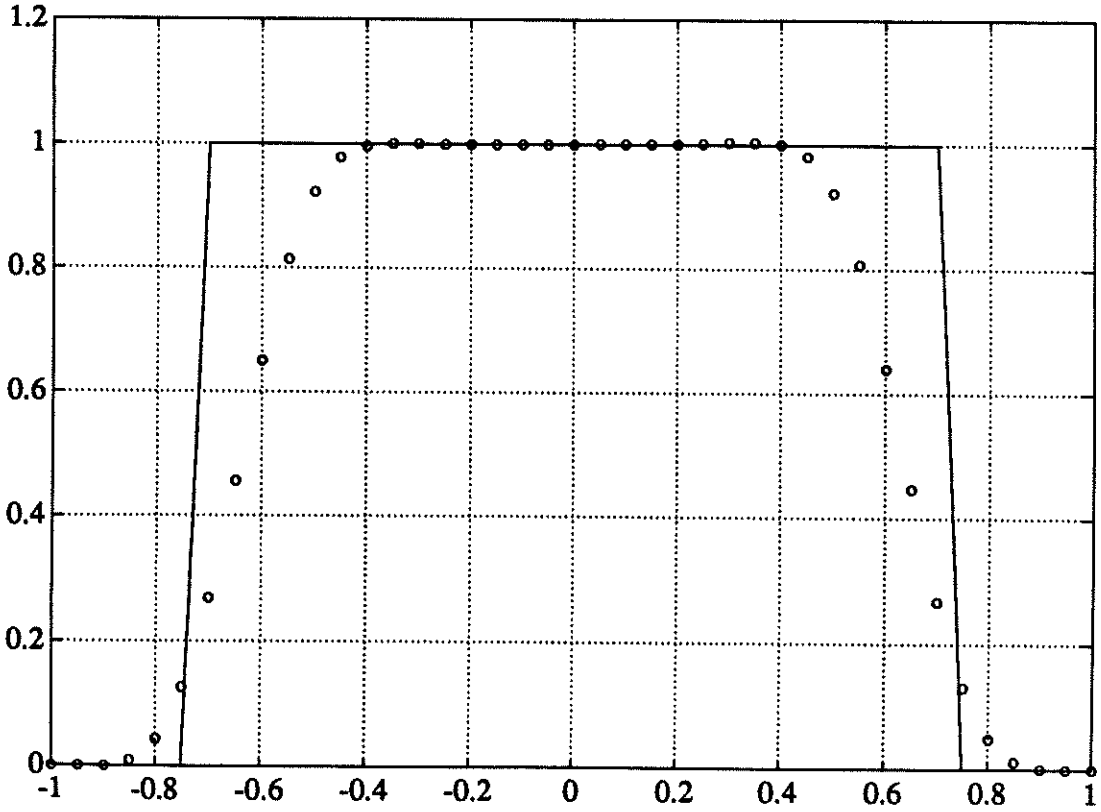

Figure 28: CPHM-REF, (4.7b), Section y=0, t=2, CFL=0.4

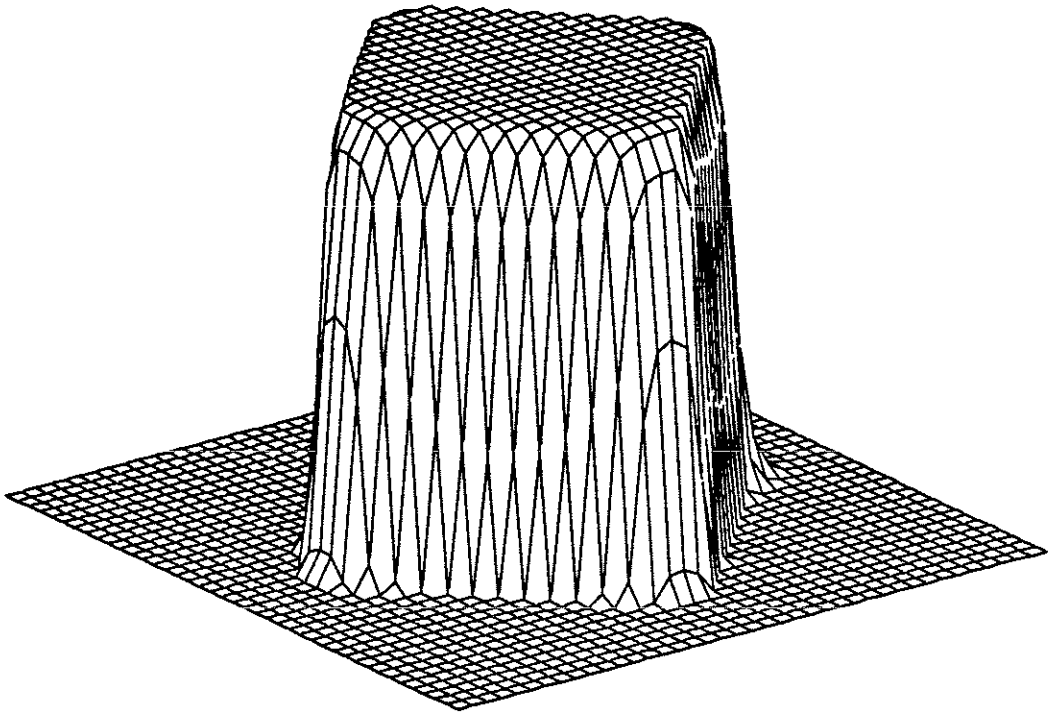

Figure 29: PHM-REF-AC, (4.7b), 40x40 grid points, t=2, CFL=0.1

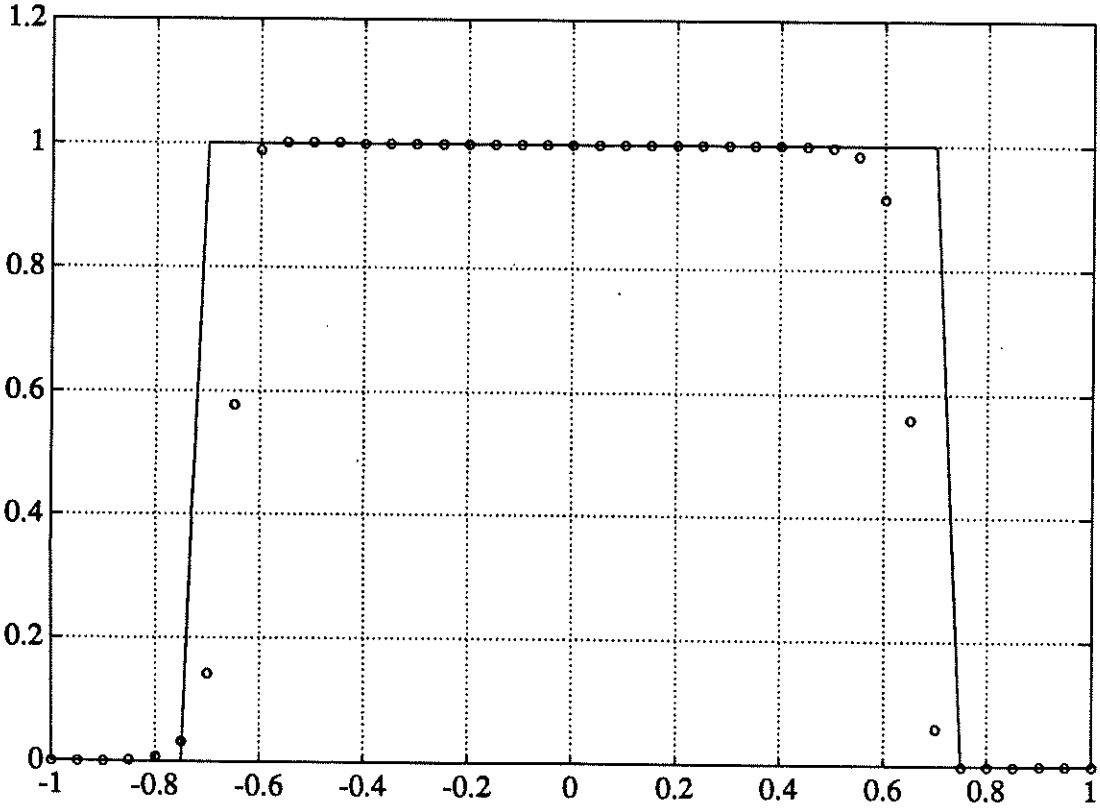

Figure 30: PHM-REF-AC, (4.7b), Section y=0, t=2, CFL=0.1

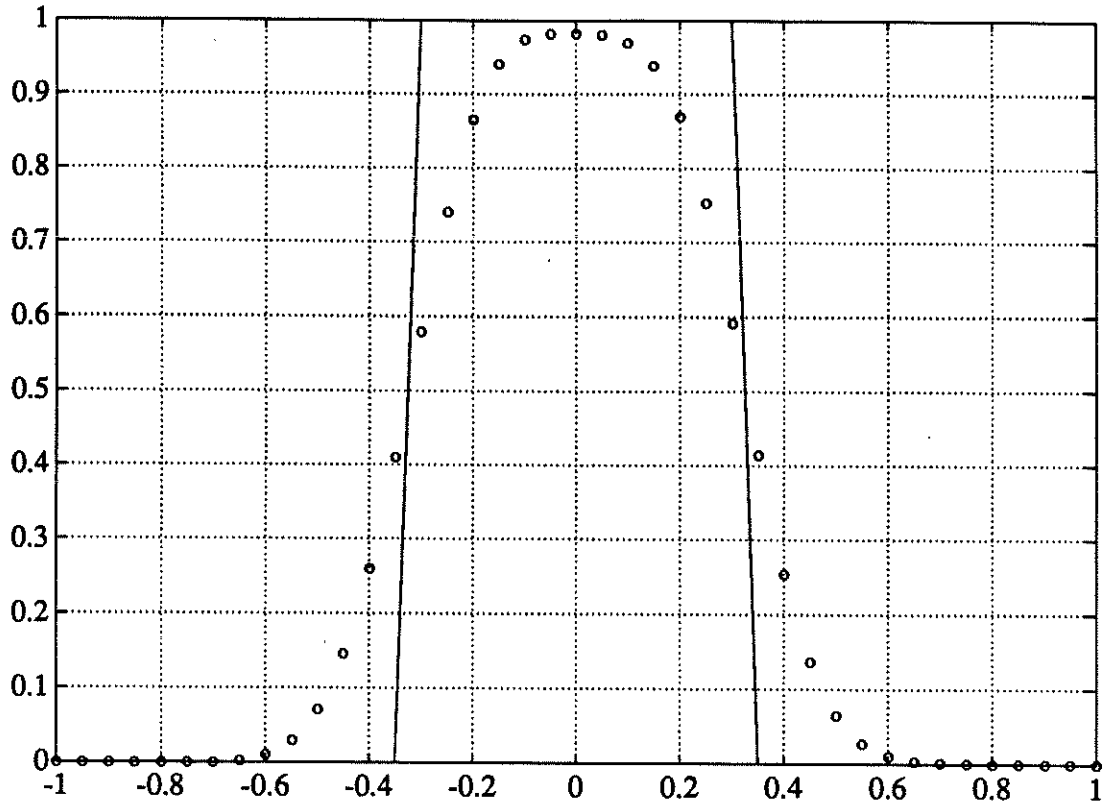

Figure 31: PHM-REF, (4.7b), Section y=-0.4, t=2, CFL=0.4

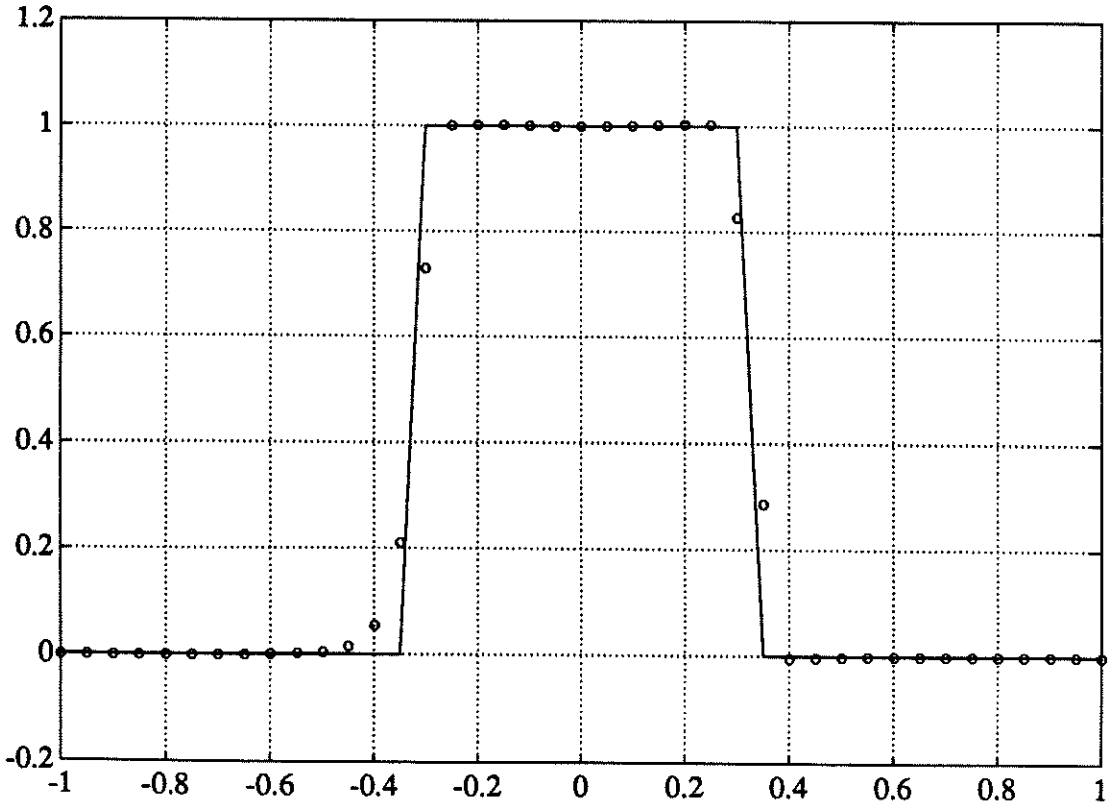

Figure 32: PHM-REF-AC, (4.7b), Section y=-0.4, t=2, CFL=0.1

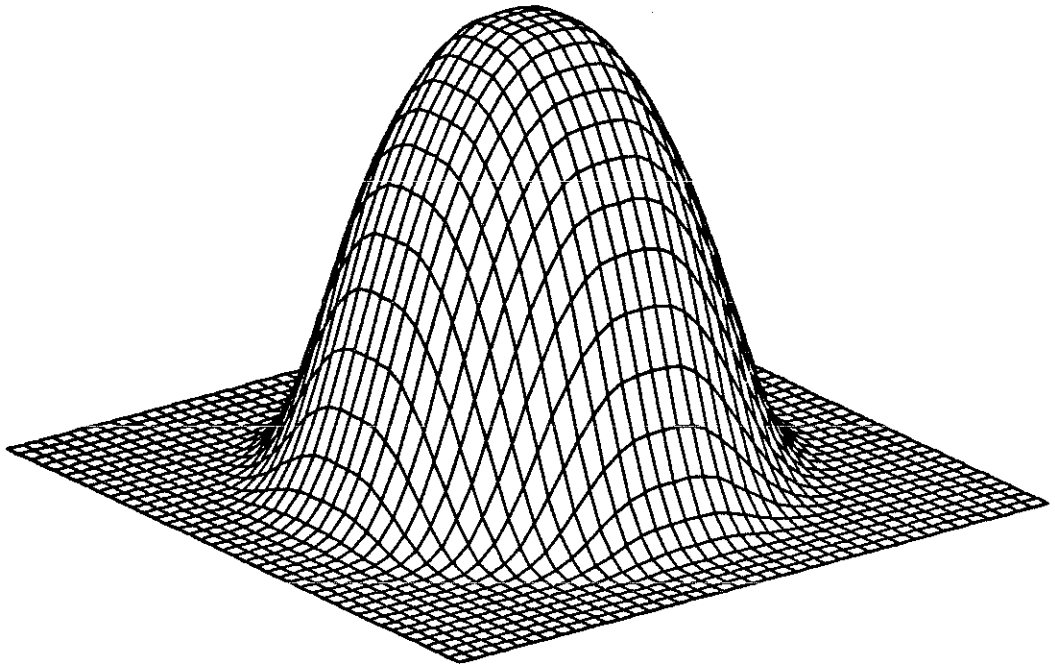

Figure 33: CPHM-REF, (4.7c), 40x40 grid points, t=2, CFL=0.4

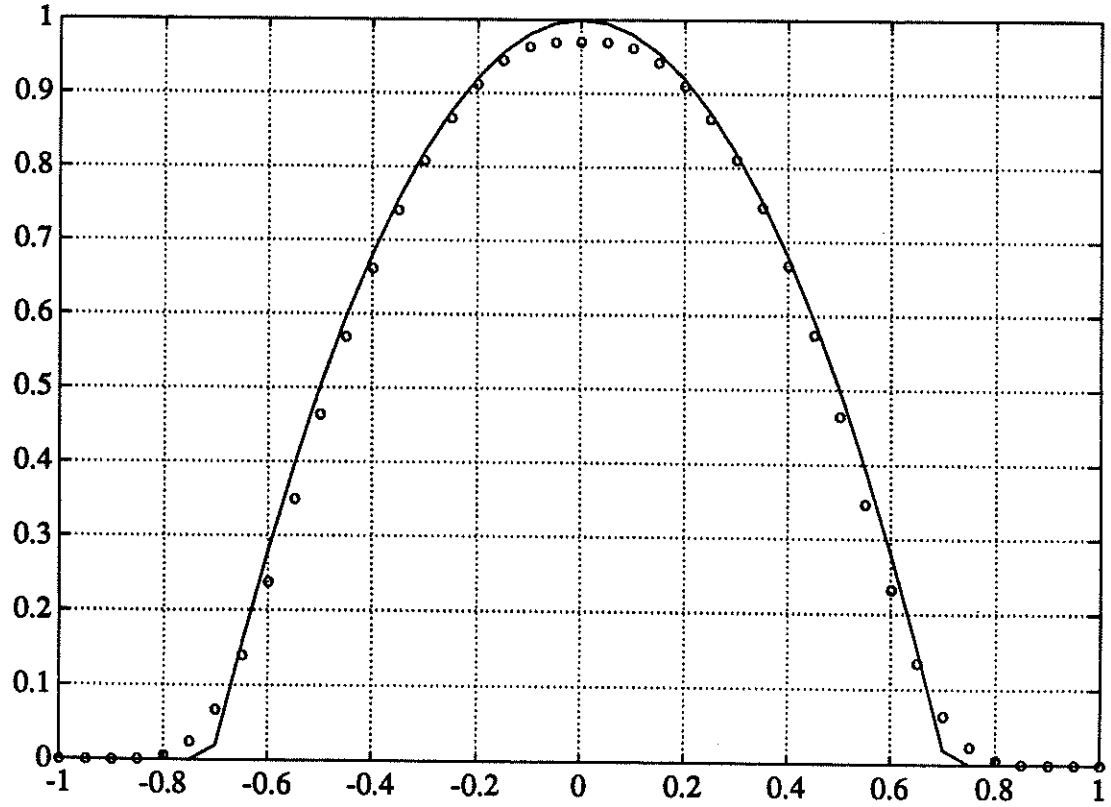

Figure 34: CPHM-REF, (4.7c), Section y=0, t=2, CFL=0.4

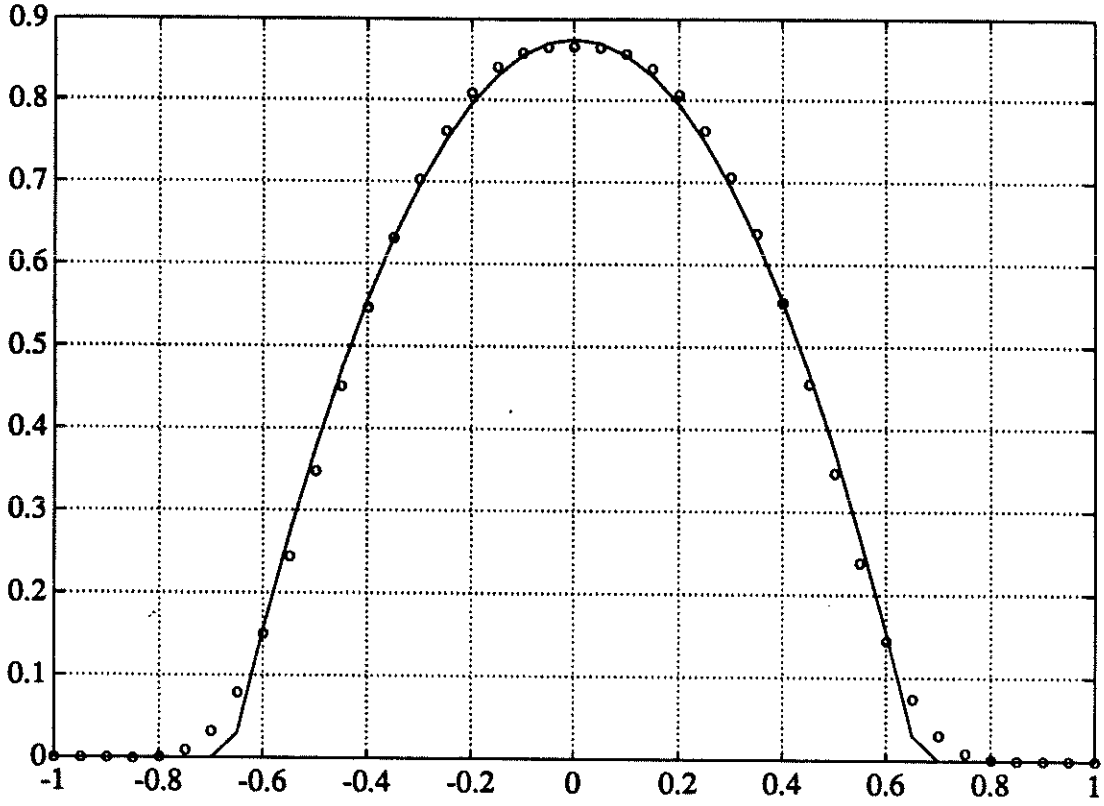

Figure 35: CPHM-REF, (4.7c), Section y=-0.25, t=2, CFL=0.4

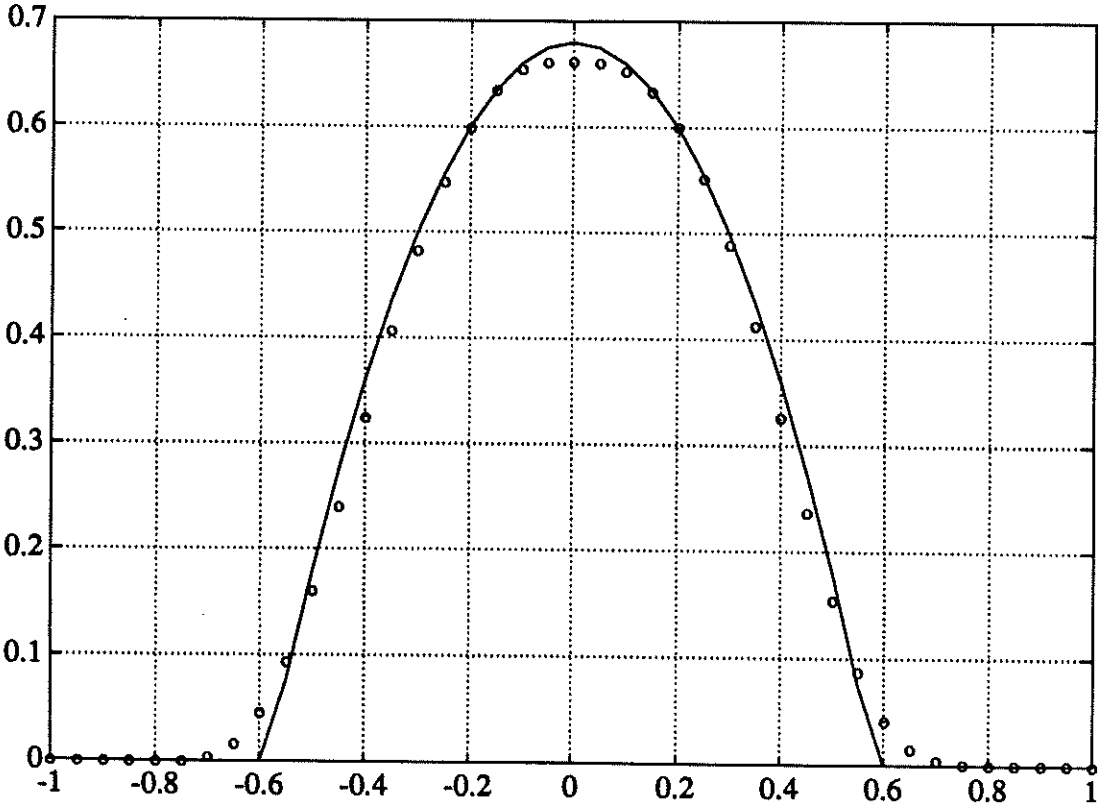

Figure 36: CPHM-REF, (4.7c), Section y=-0.4, t=2, CFL=0.4

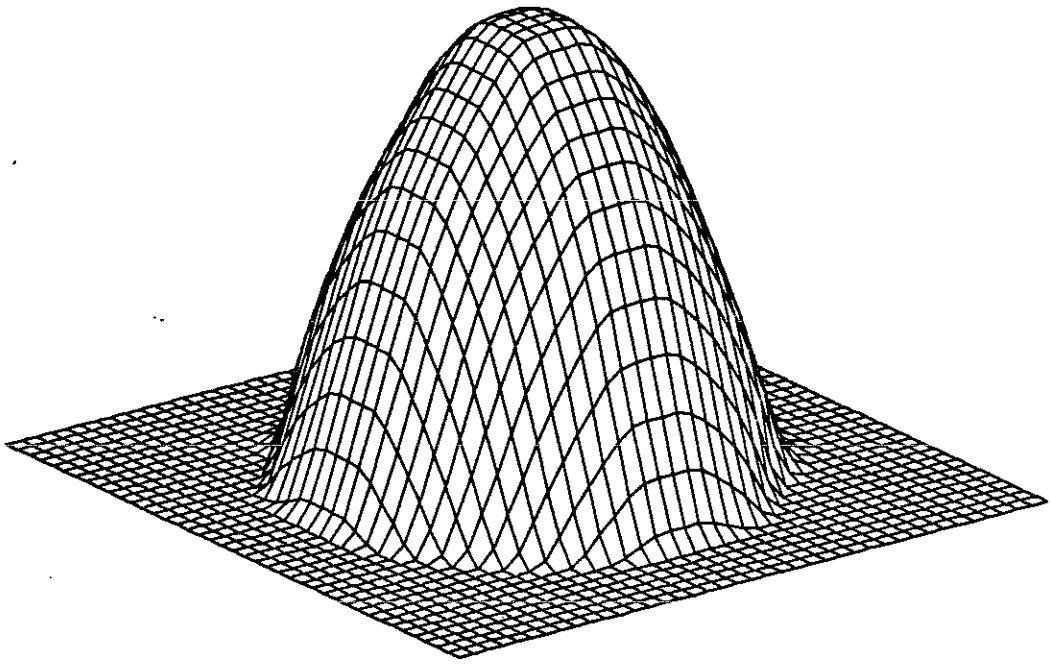

Figure 37: PHM-REF-AC, (4.7c), 40x40 grid points, t=2, CFL=0.1

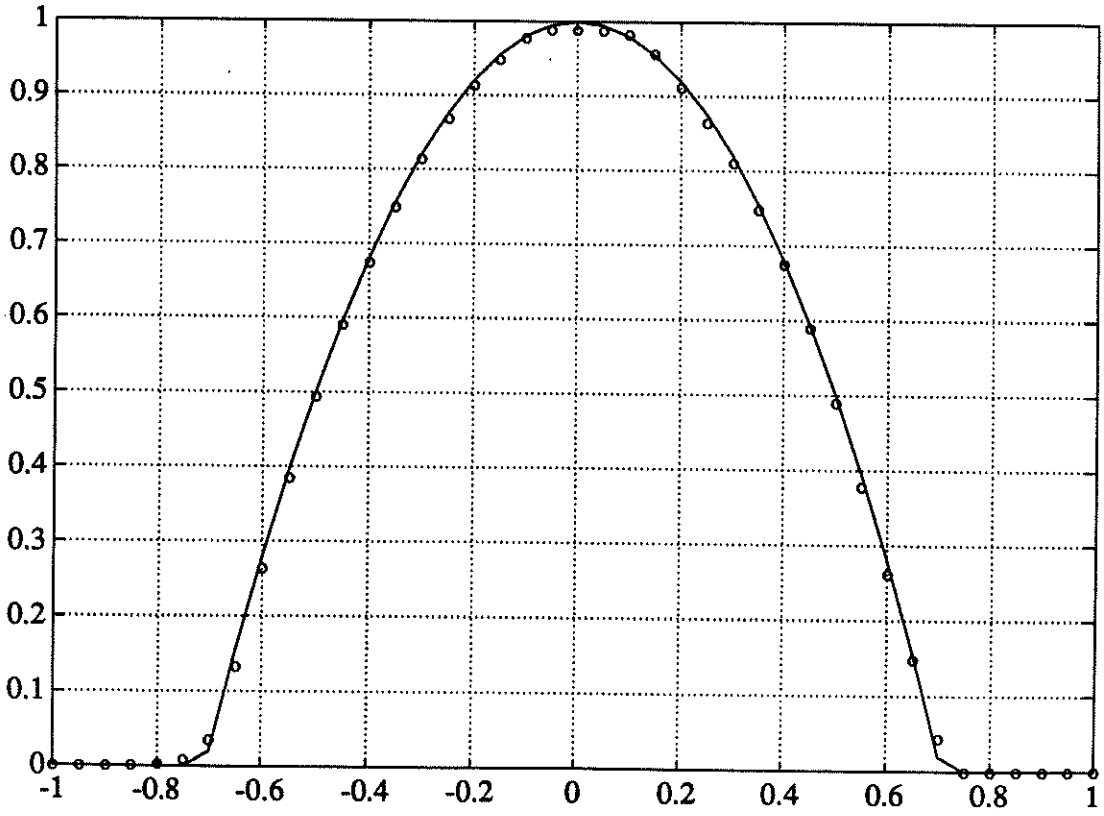

Figure 38: PHM-REF-AC, (4.7c), Section y=0, t=2, CFL=0.1

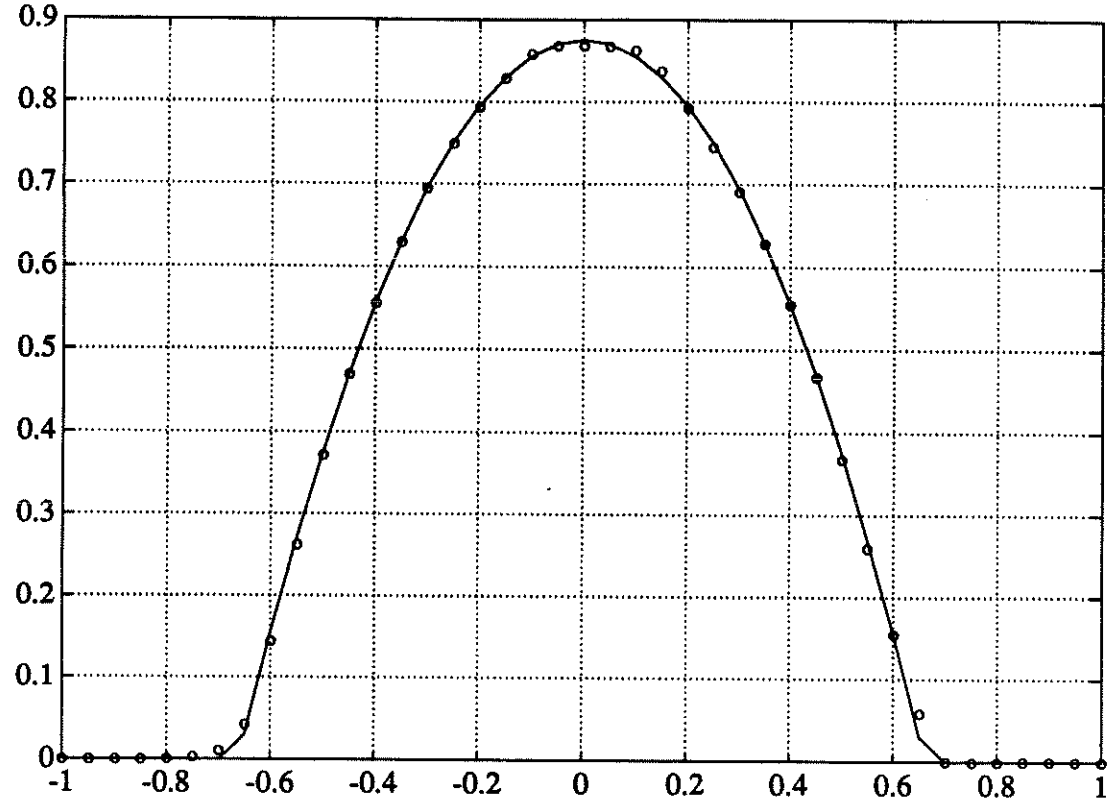

Figure 39: PHM-REF-AC, (4.7c), Section y=-0.25, t=2, CFL=0.1

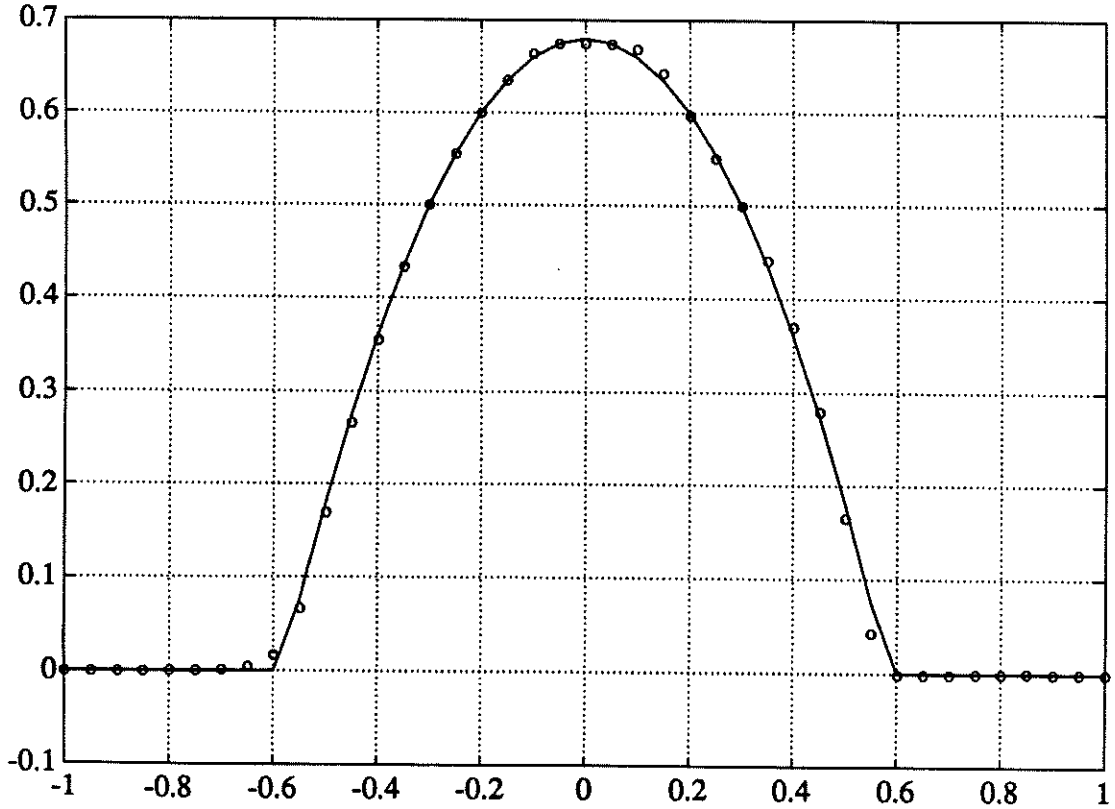

Figure 40: PHM-REF-AC, (4.7c), Section y=-0.4, t=2, CFL=0.1

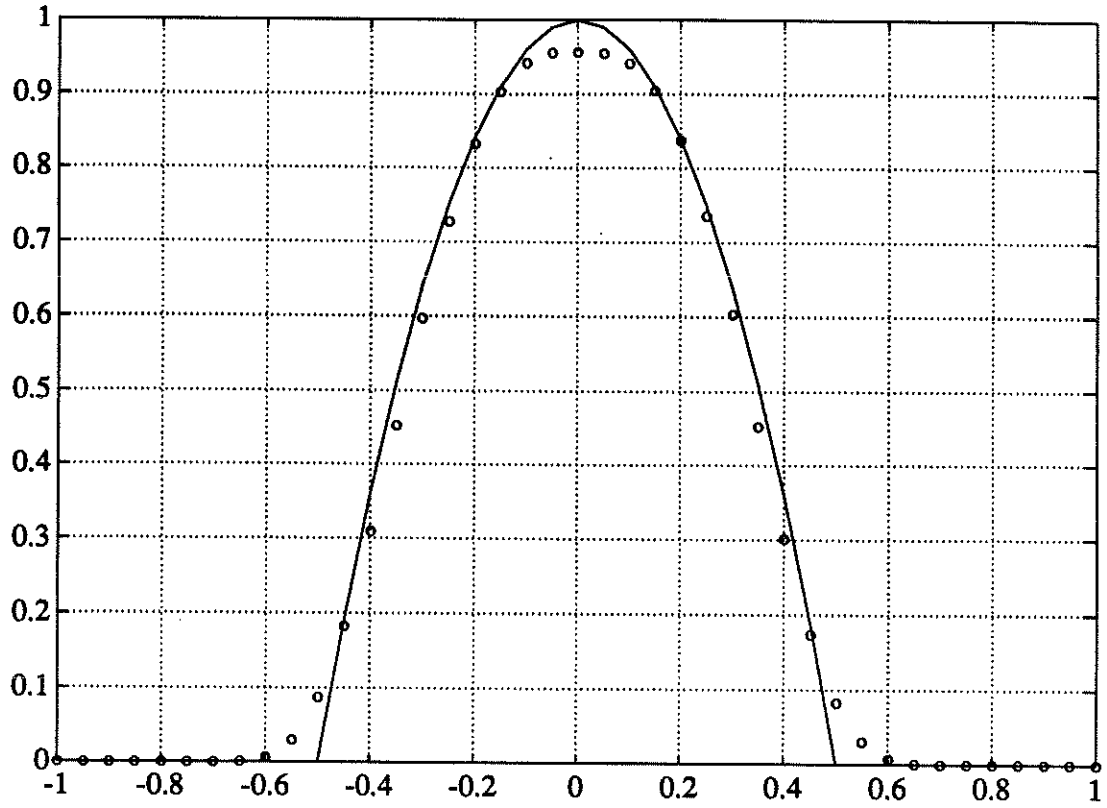

Figure 41: CPHM-REF, (4.7d), Section y=0, t=2, CFL=0.2

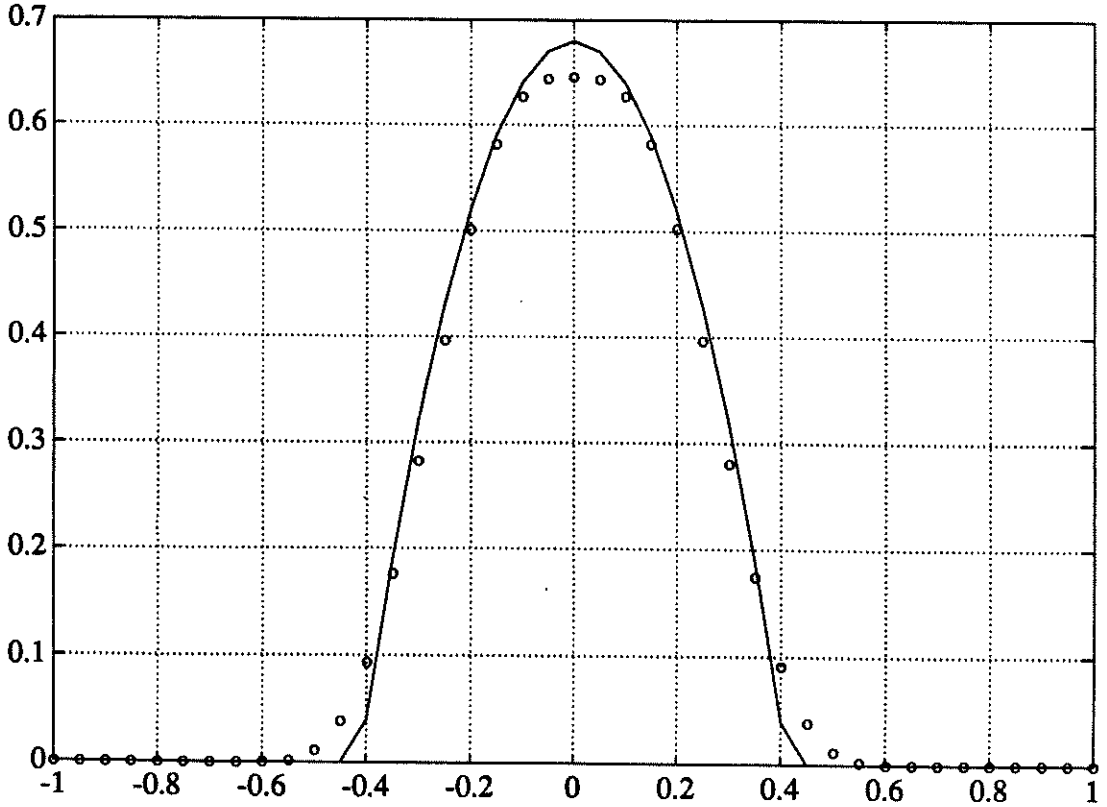

Figure 42: CPHM-REF, (4.7d), Section y=-0.4, t=2, CFL=0.2

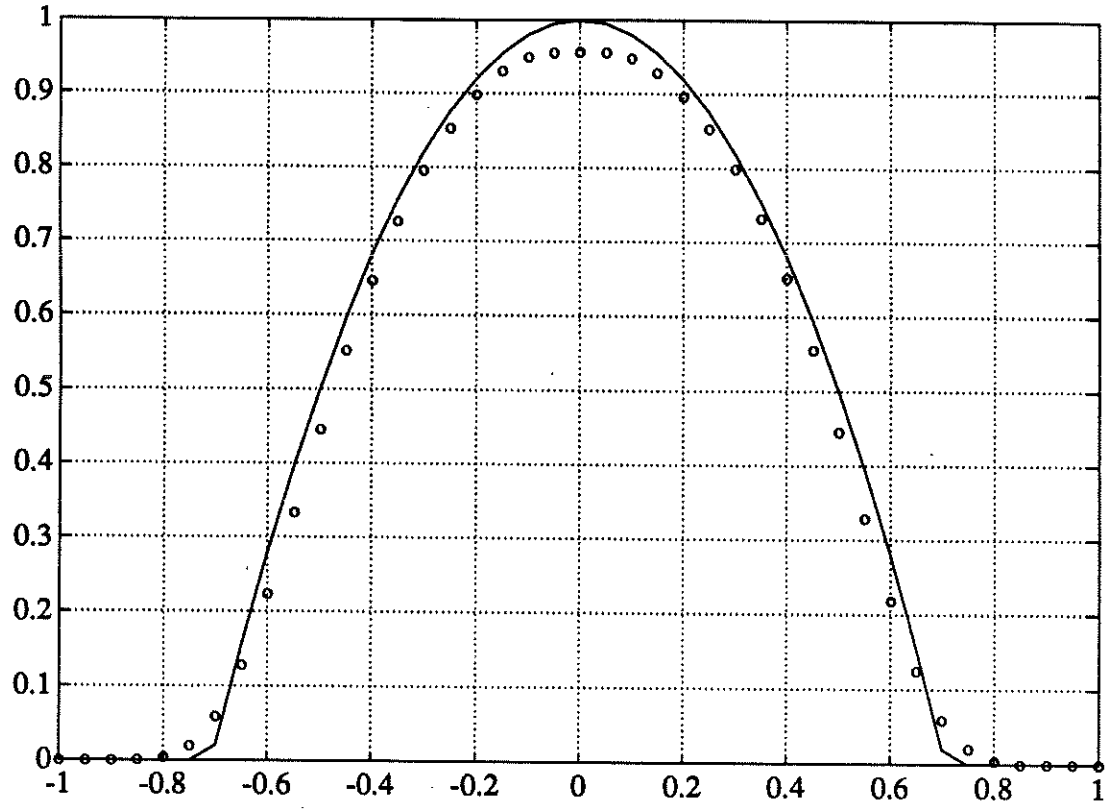

Figure 43: CPHM-REF, (4.7d), Section x=0, t=2, CFL=0.2

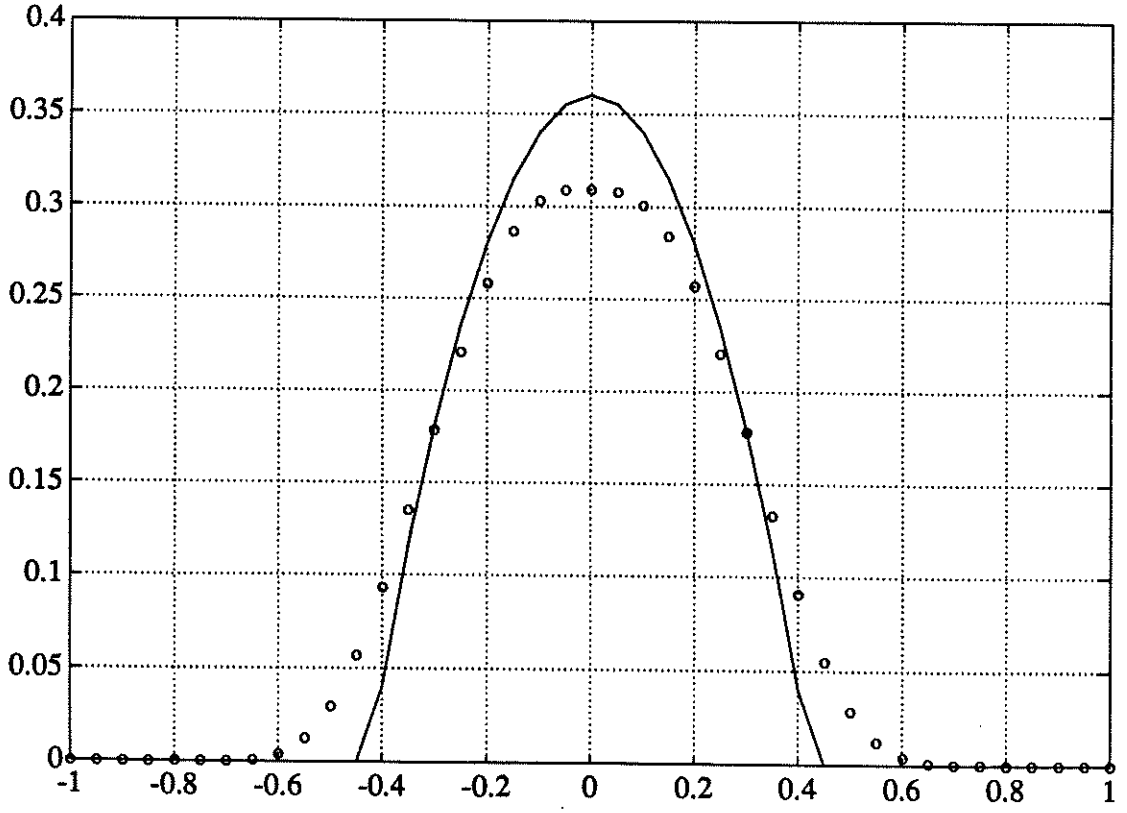

Figure 44: CPHM-REF, (4.7d), Section x=-0.4, t=2, CFL=0.2

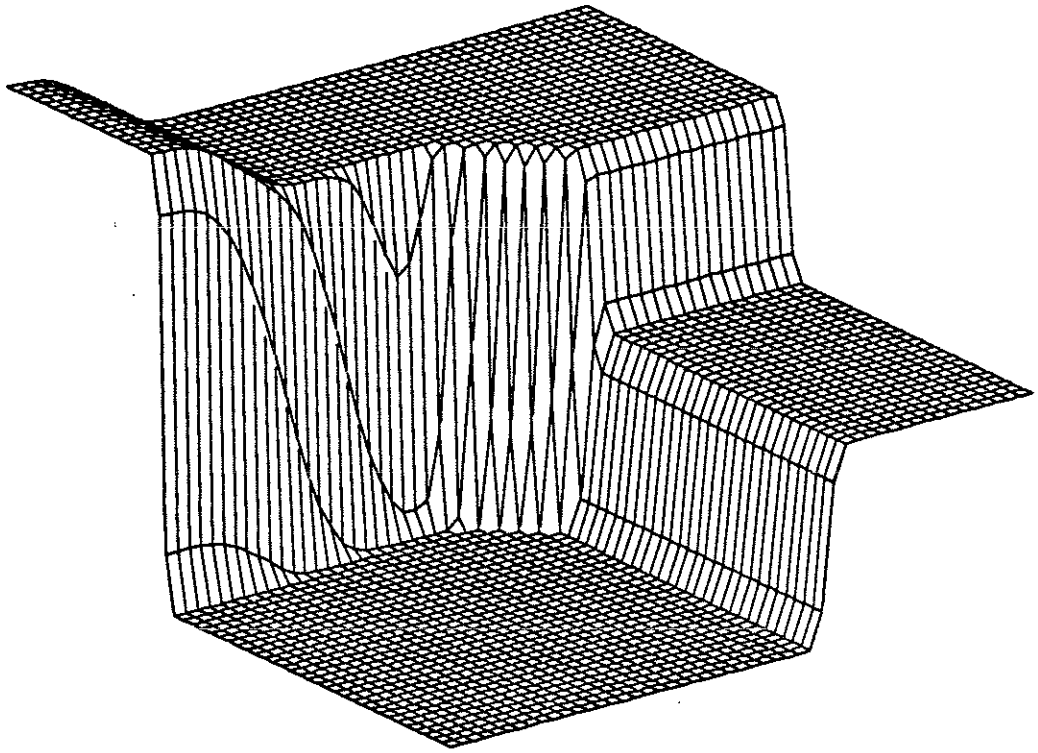

Figure 45: PHM-REF (4.8) with (-1,-0.2,0.5,0.8), 50x50 points, t=1

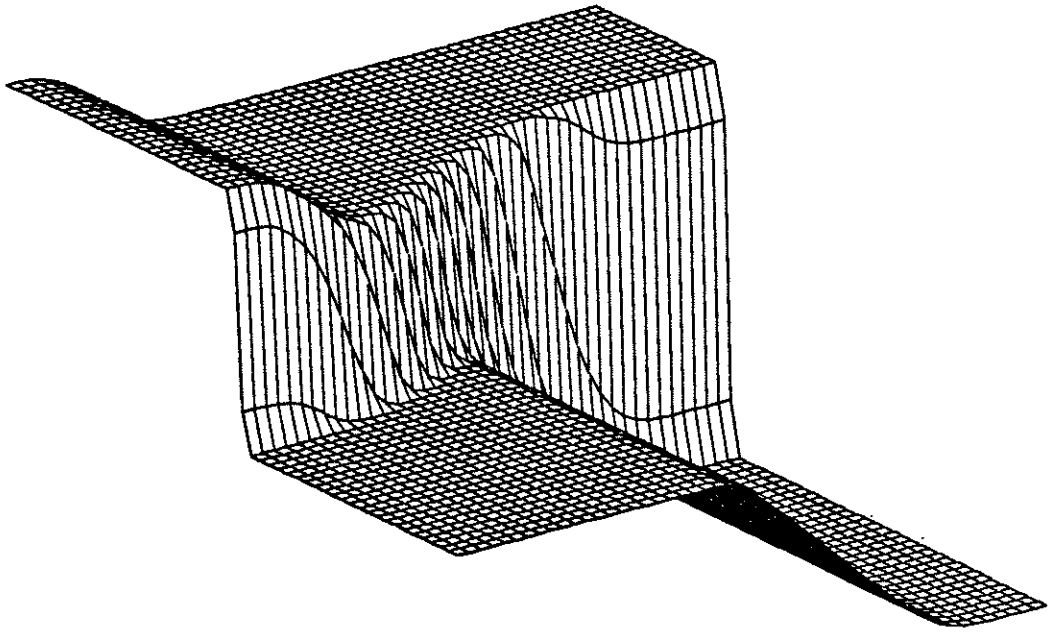

Figure 46: PHM-REF (4.8) with (-0.2,-1,0.5,0.8), 50x50 points, t=1

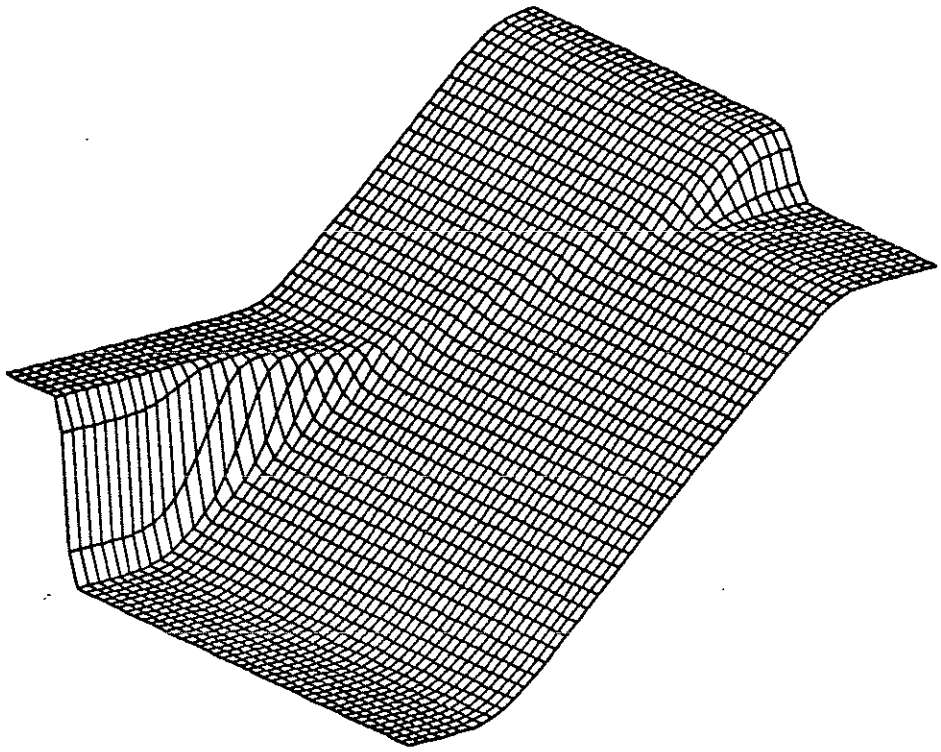

Figure 47: PHM-REF (4.8) with (0.5,-1,-0.2,0.8), 50x50 points, t=1

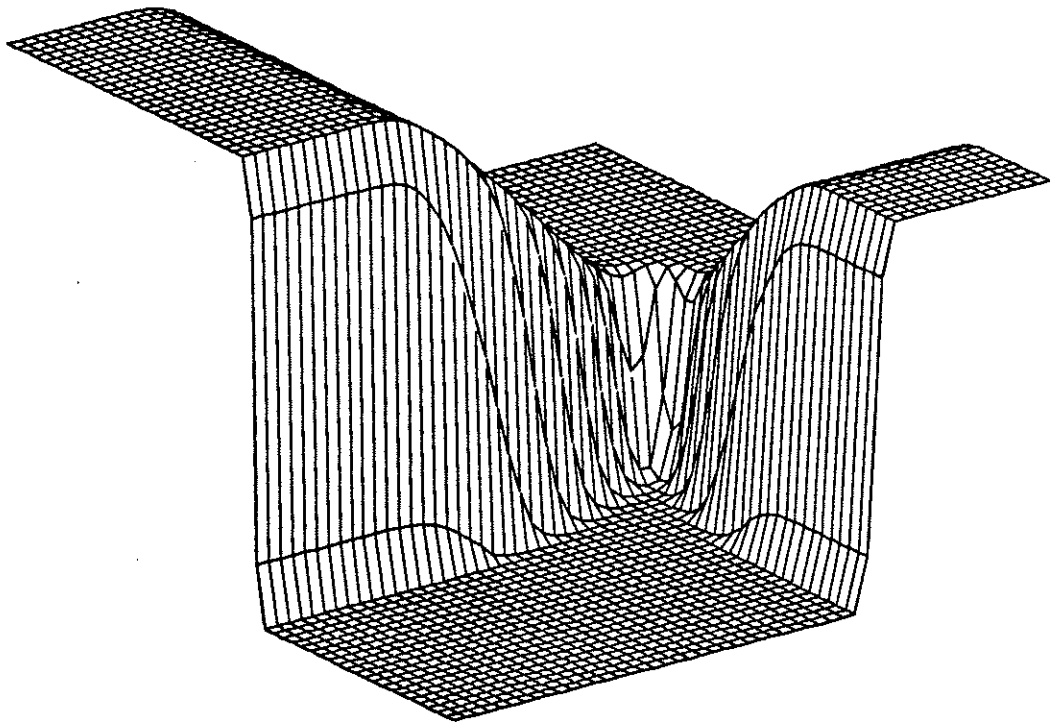

Figure 48: PHM-REF (4.8) with (-1,0.5,-0.2,0.8), 50x50 points, t=1

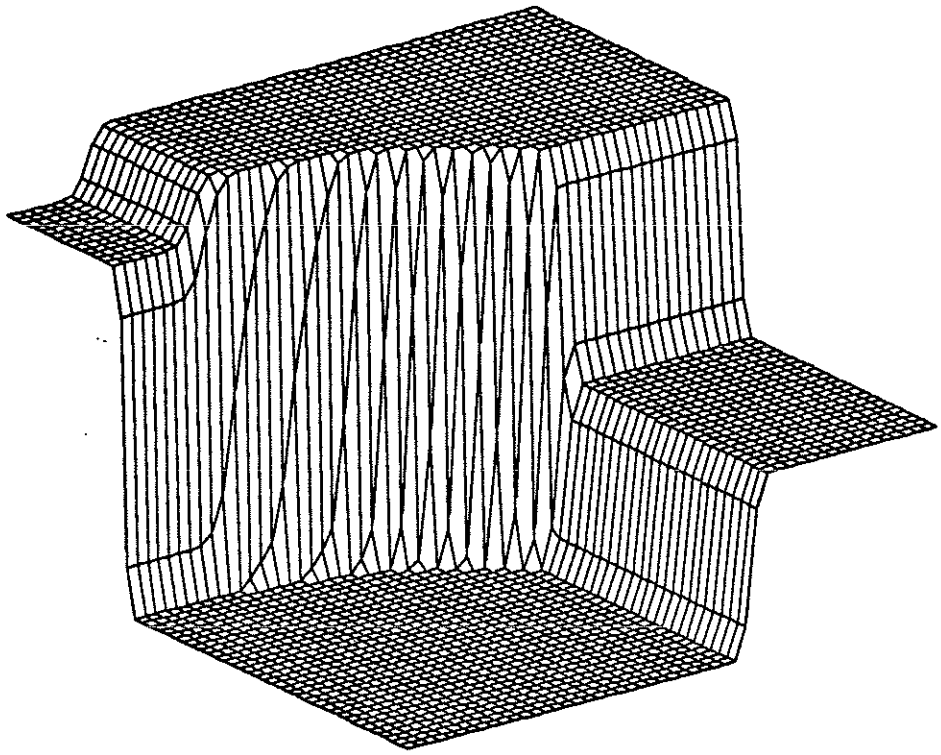

Figure 49: PHM-REF (4.8) with (-1,-0.2,0.8,0.5), 50x50 points, t=1

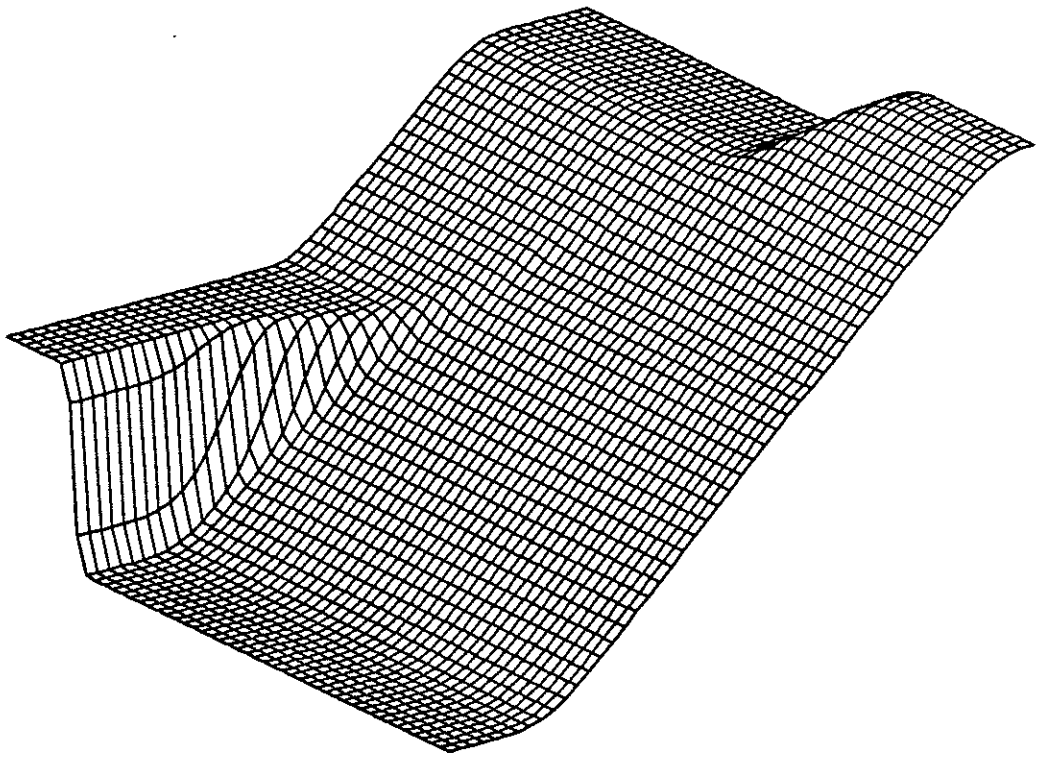

Figure 50: PHM-REF (4.8) with (0.8,-1,-0.2,0.5), 50x50 points, t=1

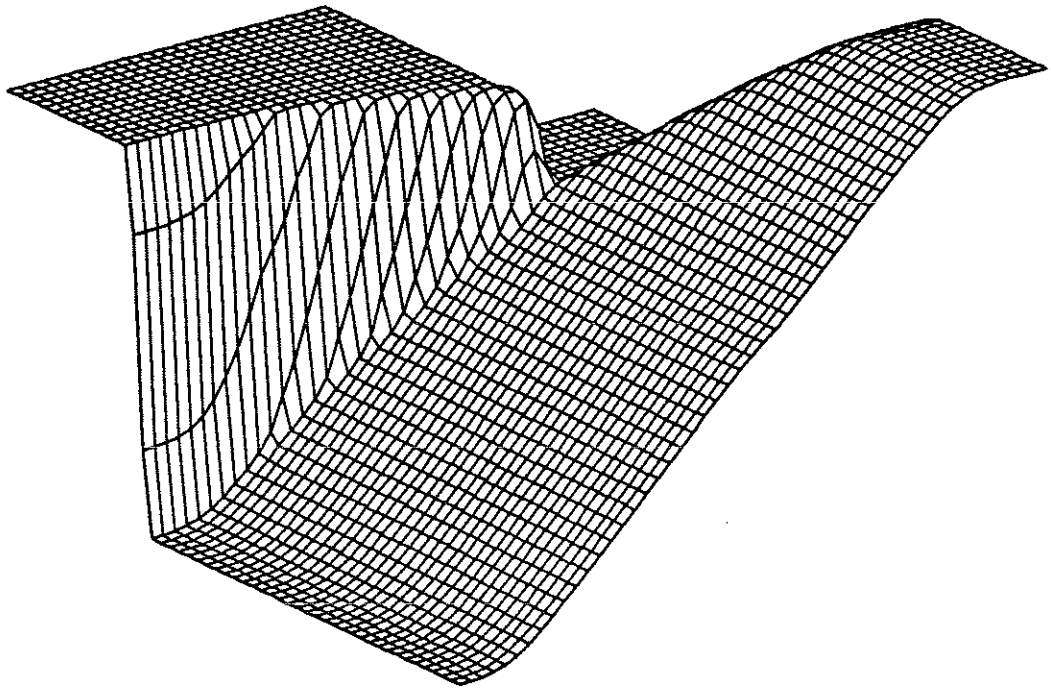

Figure 51: PHM-REF (4.8) with (0.8,-1,0.5,-0.2), 50x50 points, t=1

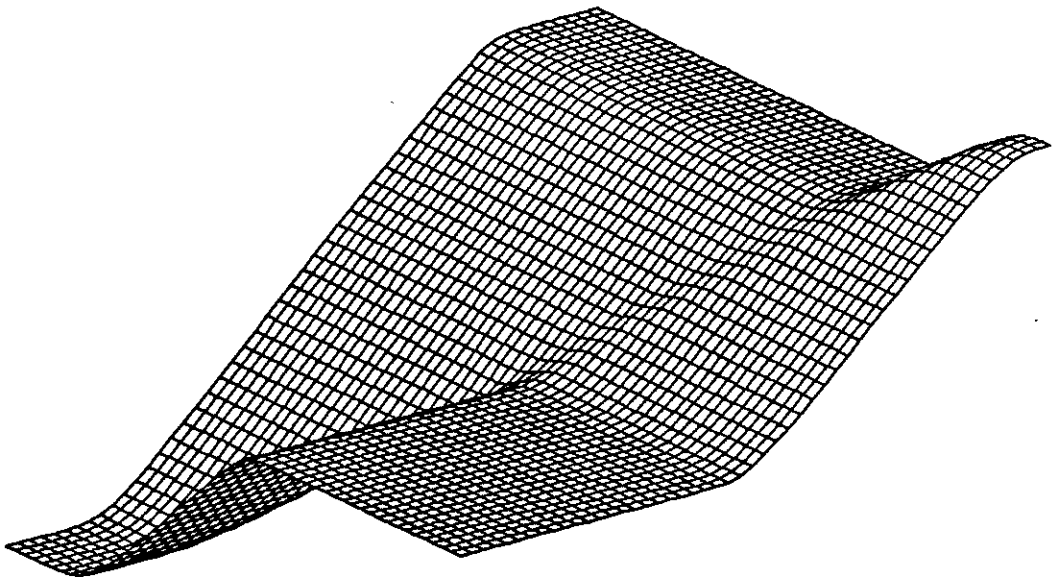

Figure 52: PHM-REF (4.8) with (0.8,-0.2,-1,0.5), 50x50 points, t=1