UCLA
COMPUTATIONAL AND APPLIED MATHEMATICS

A Treatment for Discontinuities for Finite Difference Methods in Two Dimensional Case

De-kang Mao

December 1990

CAM Report 90-30

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

A Treatment for Discontinuities for Finite Difference Methods in Two dimensional Case

Mao De-kang [1]
Department of Mathematics
University of California
Los Angeles, CA 90024
and
Department of Mathematics
Shanghai University of Science and Technology

Abstract

This paper extends the treatment introduced in [4], [5], and [6] to the two dimensional scalar case. The main idea is still that on each side of a discontinuity the computation draws information from the same side. A numerical method for ordinary differential equations that models the movement of the discontinuity curve is incorporated into the algorithm to compute discontinuity positions. Conservation feature of the treatment is presented for the case of a single discontinuity. Finally, numerical examples are displayed.

Subject Classification: 65M05, 35L65
Key Words: treatment of discontinuity, critical mesh interval, interaction of discontinuities, node mesh.

[1]Research was supported by ONR Grant No. N00014-86-k-0691.

1. Introduction

In this paper we extend the treatment for discontinuities introduced in [4], [5], and [6] to two dimensional scalar conservation laws

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

with initial value condition

$$u(x, y, 0) = u_0(x, y). \tag{1.1b}$$

First we recall the treatment for the one dimensional case in [4]. The treatment is a shock tracking technique, whose main idea is that the computation on each side of a discontinuity draws the information that jumps at the discontinuity only from the same side. An numerical method for ordinary differential equations is incorporated into the underlying scheme to compute positions of the discontinuity. As an example for the performance of the treatment, we describe how it treats a single discontinuity in one dimensional scalar case.

Assume the underlying finite difference scheme is a general conservative scheme:

$$u_j^{n+1} = u_j^n - \lambda(\hat{f}_{j+1/2}^n - \hat{f}_{j-1/2}^n), \tag{1.2}$$

where u_j^n denotes the datum of the numerical solution at a grid point (x_j, t^n),

$$\hat{f}_{j+1/2}^n = \hat{f}(u_{j-k+1}^n, \cdots, u_{j+k}^n) \tag{1.3}$$

is a consistent numerical flux depending on $2k$ variables, $\lambda = \tau/h$ is the mesh ratio, where τ and h are the time and space increments respectively. Suppose that on the level n the numerical solution just has a jump in a cell $[x_{j_1}, x_{j_1+1}]$, on each side of which the numerical solution is supposed to be smooth (as shown in Figure 1.1). Also suppose that the position of the discontinuity within the cell is known as ξ^n. The cell is called as a critical cell, to which the treatment is going to be applied. The treatment performs in the following four steps:

1) Extrapolate the numerical solution from each side of the discontinuity to the other side, and get a set of extrapolated data:

$$u_{j_1-k}^{n,+}, \cdots, u_{j_1}^{n,+}, u_{j_1+1}^{n,-}, \cdots, u_{j_1+k+1}^{n,-}, \tag{1.4}$$

where the data with "$-$" are from the left to the right and the data with "$+$" are from the right to the left.

2) On each side of the discontinuity compute the numerical solution by using the extrapolated data from the same side; i.e., when $x_j \leq x_{j_1}$,

$$u_j^{n+1} = u_j^n - \lambda(\hat{f}_{j+1/2}^{n,-} - \hat{f}_{j-1/2}^{n,-}), \tag{1.5}$$

where

$$\hat{f}_{j+1/2}^{n,-} = \hat{f}(u_{j-k+1}^n, \cdots, u_{j_1}^n, u_{j_1+1}^{n,-}, \cdots, u_{j+k}^{n,-}), \tag{1.6}$$

and when $x_j > x_{j_1}$,

$$u_j^{n+1} = u_j^n - \lambda(\hat{f}_{j+1/2}^{n,+} - \hat{f}_{j-1/2}^{n,+}), \tag{1.7}$$

where

$$\hat{f}_{j+1/2}^{n,+} = \hat{f}(u_{j-k+1}^{n,+} \cdots, u_{j_1}^{n,+}, u_{j_1+1}^n, \cdots, u_{j_1+k}^n), \tag{1.8}$$

3) Compute ξ^{n+1}, the position of the discontinuity on the level $n+1$, by a numerical approximation to the Hugoniot condition; e.g.,

$$\xi^{n+1} = \xi^n + \frac{f(u_{\xi^n}^r) - f(u_{\xi^n}^l)}{u_{\xi^n}^r - u_{\xi^n}^l}\tau, \tag{1.9}$$

where, $u_{\xi^n}^l$ and $u_{\xi^n}^r$ are extrapolated data of the numerical solution from the two sides of the discontinuity at the position of the discontinuity.

4) Determine the critical cell on the level $n+1$ according to the new position of the discontinuity. If ξ^{n+1} is still in $[x_{j_1}, x_{j_1+1}]$, take the same cell as the critical cell on the new level. If ξ^{n+1} moves into the left adjacent cell $[x_{j_1-1}, x_{j_1}]$, take this cell as the critical cell, meanwhile update $u_{j_1}^{n+1}$ by the extrapolated datum of u^{n+1} from the right side. If ξ^{n+1} moves into $[x_{j_1+1}, x_{j_1+2}]$, take this cell as the new critical cell and update $u_{j_1+1}^{n+1}$.

Unlike the traditional shock tracking methods, this treatment does not need a lower dimensional adaptive grid to resolve the discontinuity, nor computes the numerical solution at points of this grid. Thus, the whole computation still proceeds on the regular grid, the corresponding algorithm is quite simple and it is possible to use it to capturing a spontaneous shock.

The numerical solution computed by using the treatment is not conserved; however, [4] proved that it satisfies

$$\sum_j u_j^n h = \sum_j u_j^0 h + O(h), \tag{1.10}$$

which indicates that it is almost conserved.

To build up the treatment for the two dimensional equation (1.1a), we first need to extend the concept of the critical cell to the two dimensional case. Since a two dimensional discontinuity is a curve, we call the mesh intervals the discontinuity crosses as critical mesh intervals (see Figure 2.2). There are two kinds of critical mesh intervals, the horizontal critical mesh intervals that are on horizontal grid lines and the vertical critical mesh intervals that are on vertical grid lines. A discontinuity in the numerical solution is represented by a group of horizontal and vertical critical mesh intervals, each of which contains a position of the discontinuity, i.e., the intersection point of the discontinuity and the critical mesh interval.

An important part of the treatment is to incorporate a numerical method for the Hugoniot condition into the underlying scheme. This is relevantly easy for the one dimensional case since the corresponding condition is an ODE. However, this is somewhat difficult for two dimensional case since the corresponding condition is a partial differential equation involving the normal vector to the discontinuity curve (see [11]). If one discretizes the Hugoniot condition along the normal direction to compute positions of the discontinuity, just as what the real tracking method does, the positions of the discontinuity on the next level will mostly go into grid meshes; therefore the computation can not mantain only on the regular grid.

In this paper we discretize the Hugoniot condition along the horizontal or vertical directions rather than the normal direction. We compute the horizontal or vertical moving speeds of discontinuity positions, and use them to compute the new positions on the next level. Theorem 2.1 in §2. gives the formulae for the calculation, which are derived from the Hugoniot condition. In doing so, the whole computation still proceeds on the regular grid points, and no information from the interior of the grid meshes is needed.

The paper is organized in the following manner: §2. describes the treatment for a single discontinuity. §.3 discusses the conservation feature of the treatment and shows that in two dimensional case the numerical solution is also almost conserved. §4. describes the treatment of interactions of discontinuities. §5. presents some numerical examples of the treatment.

2. Treatment for a Single Discontinuity

The following theorem gives the formulae to calculate the horizontal and vertical moving speeds of discontinuity positions.

Theorem 2.1 On the $(x,y)-$plane, an intersection point of a discontinuity curve of (1.1) with a horizontal line satisfies the following ordinary differential equation

$$\frac{\partial x}{\partial t} = \frac{\alpha[f] + \beta[g]}{\alpha[u]}, \tag{2.1}$$

and an intersection point of the discontinuity curve with a vertical line satisfies

$$\frac{\partial y}{\partial t} = \frac{\alpha[f] + \beta[g]}{\beta[u]}, \tag{2.2}$$

where x and y are the horizontal and vertical coordinates of the intersection point, $[v]$ indicates the jump of a quantity v across the discontinuity, $(\alpha,\ \beta)$ is the normal vector to the discontinuity curve at the intersection point.

Proof. Assume S is a discontinuity surface of (1.1a) in the three dimensional $(x,y,t)-$space, which cuts a horizontal plane at time t by a curve C (see Figure 2.1). At a point p on C the two states connected by the discontinuity satisfy the following Hugoniot condition:

$$\vec{n} \cdot ([u],[f],[g]) = 0, \tag{2.3}$$

where $\vec{n}$ is a normal vector to S at p (see [11]). This means that the vector $\vec{v_1} = ([u],[f],[g])$ is tangential to S. Since the horizontal normal vector to C at p on the (x,y)-plane is (α,β), vector $\vec{v_2} = (0,-\beta,\alpha)$ is also tangential to S. Therefore, the vector

$$\vec{v_1} \times \vec{v_2} = (-\alpha[f] - \beta[g], \alpha[u], \beta[u]) \tag{2.4}$$

is perpendicular to S at p. This indicates that S has the following differential form:

$$(\alpha[f] + \beta[g])dt = \alpha[u]dx + \beta[u]dy. \tag{2.5}$$

By taking x and y to be constants respectively we obtain (2.1) and(2.2). Thus ends the proof.

We assume the underlying scheme is a method of lines approximation to (1.1) (see [7],[8],[9],[10]) with the predictor-corrector time discretization for

u_t. That is

$$\begin{aligned} u_{i,j}^{n+1/2} &= u_{i,j}^n - L_{\hat{f}}(\mathbf{u}^n)_{i,j} - L_{\hat{g}}(\mathbf{u}^n)_{i,j} \qquad \text{(predictor)} \\ \bar{u}_{i,j}^{n+1} &= u_{i,j}^{n+1/2} - L_{\hat{f}}(\mathbf{u}^{n+1/2})_{i,j} - L_{\hat{g}}(\mathbf{u}^{n+1/2})_{i,j} \\ u_{i,j}^{n+1} &= \frac{1}{2}(u_{i,j}^n + \bar{u}_{i,j}^{n+1}), \qquad \text{(corrector)}, \end{aligned} \tag{2.6}$$

where

$$\begin{aligned} L_{\hat{f}}(\mathbf{u}^n)_{i,j} &= \frac{\lambda}{2}(\hat{f}_{i+1/2,j}^n - \hat{f}_{i-1/2,j}^n) \\ L_{\hat{g}}(\mathbf{u}^n)_{i,j} &= \frac{\lambda}{2}(\hat{g}_{i,j+1/2}^n - \hat{g}_{i,j-1/2}^n) \end{aligned} \tag{2.7}$$

are approximations to $\frac{1}{2}\tau f_x$ and $\frac{1}{2}\tau g_x$ at point (i,j) with the fluxes $f_{i+1/2}^n$ and $g_{i,j+1/2}^n$ defined as

$$\begin{aligned} \hat{f}_{i+1/2,j}^n &= \hat{f}(u_{i-k+1,j}^n, \cdots, u_{i+k,j}^n) \\ \hat{g}_{i,j+1/2}^n &= \hat{g}(u_{i,j-k+1}^n, \cdots, u_{i,j+k}^n) \end{aligned} \tag{2.8}$$

consisting with f and g in the sense that

$$\begin{aligned} \hat{f}(u, \cdots, u) &= f(u) \\ \hat{g}(u, \cdots, u) &= g(u). \end{aligned} \tag{2.9}$$

We would like to mention that the discussion in this paper applies to methods of lines approximation with any type of Runge-Kutta time discretizations.

A reason to use this type of schemes is that it is easy to incorporate the Runge-Kutta methods for the Hugoniot condition, which have been proved to be effective numerical methods and already been wildly used to solve ordinary differential equations. Another reason is that we can use fluxes (2.8) with a Runge-Kutta procedure to get a high order underlying scheme, by which we avoid two dimensional extrapolation. This will be shown later in this section.

Since the underlying scheme is essentially a combination of two Euler forward schemes, it is sufficient to build up the treatment for this forward version

$$u_{i,j}^{n+1} = u_{i,j}^n - \frac{\lambda}{2}(\hat{f}_{i+1/2,j}^n - \hat{f}_{i-1/2,j}^n) - \frac{\lambda}{2}(\hat{g}_{i,j+1/2}^n - \hat{g}_{i,j-1/2}^n). \tag{2.10}$$

For a single discontinuity, the treatment generally still performs in four steps as it does in the one dimensional case described in §1, i.e.

1) prepare extrapolated data that will be used in the computation.
2) compute the numerical solution using the extrapolated data from the same side.
3) compute positions of the discontinuity on the next level by (2.1) and (2.2).
4) determine the critical mesh intervals on the next level according to the new positions of the discontinuity and update the numerical solution at some grid points if necessary.

However, due to the geometrical complexity, the two dimensional treatment is more complicated than the one dimensional version, and more details are described in the follows.

Preparation of the extrapolated data and computation on each side of the discontinuity.

Due to the use of fluxes (2.8), step 2) is easy to be done. For example, suppose that we have a discontinuity as shown in Figure 2.2, the two sides connected by which are denoted by "$-$" and "$+$" respectively. The numerical solution at point $P_1(x_{i_1}, y_{j_1})$ is computed as follows:

$$u_{i_1,j_1}^{n+1} = u_{i_1,j_1}^{n} - \frac{\lambda}{2}(\hat{f}_{i_1+1/2,j_1}^{n,-} - \hat{f}_{i_1-1/2,j_1}^{n,-}) - \frac{\lambda}{2}(\hat{g}_{i_1,j_1+1/2}^{n,-} - \hat{g}_{i_1,j_1-1/2}^{n,-}), \quad (2.11)$$

where

$$\hat{f}_{i+1/2,j_1}^{n,-} = \hat{f}(u_{i-k+1,j_1}^{n}, \cdots, u_{i_1,j_1}^{n}, u_{x,i_1+1,j_1}^{n,-}, \cdots, u_{x,i+k,j_1}^{n,-}), \quad (2.12)$$

and

$$\hat{g}_{i_1,j+1/2}^{n,-} = \hat{g}(u_{i_1,j-k+1}^{n}, \cdots, u_{i_1,j_1+1}^{n}, u_{y,i_1,j_1+2}^{n,-}, \cdots, u_{y,i_1,j_1+k}^{n,-}). \quad (2.13)$$

The numerical solution at point $P_2(x_{i_1+1}, y_{j_1})$ is computed as follows:

$$u_{i_1+1,j_1}^{n+1} = u_{i_1+1,j_1}^{n} - \frac{\lambda}{2}(\hat{f}_{i_1+3/2,j_1}^{n,+} - \hat{f}_{i_1+1/2,j_1}^{n,+}) - \frac{\lambda}{2}(\hat{g}_{i_1+1,j_1+1/2}^{n,+} - \hat{g}_{i_1+1,j_1-1/2}^{n,+}), \quad (2.14)$$

where

$$\hat{f}_{i+1/2,j_1}^{n,+} = \hat{f}(u_{x,i-k+1,j_1}^{n,+}, \cdots, u_{x,i_1,j_1}^{n,+}, u_{i_1+1,j_1}^{n}, \cdots u_{i+k,j_1}^{n}) \quad (2.15)$$

and

$$\hat{g}^{n,+}_{i_1+1,j+1/2} = \hat{g}(u^{n,+}_{y,i_1+1,j-k+1}, \cdots, u^{n,+}_{y,i_1+1,j_1-1}, u^{n}_{i_1+1,j_1}, \cdots, u^{n}_{i_1+1,j+k}). \quad (2.16)$$

$\hat{f}$ only uses extrapolated data along the x direction, which are indicated by a lower index x, while $\hat{g}$ only uses extrapolated data along the y direction, which are indicated by a lower index y. This is possible since all the variables in $\hat{f}$ are on a same horizontal grid line and all the variables in $\hat{g}$ are on a same vertical grid line. In doing so, we avoid two dimensional extrapolation, which, otherwise, might cause complication. Since the extrapolated data are prepares in such a dimension-by-dimension way, at a same grid point the two numerical fluxes may use different data that are from different directions. However, the numerical experiments show that this does not cause problems. The reason is, as we claimed in [4], that these extrapolated data are essentially "virtual" since shocks have a characteristic-converging feature and contact discontinuities have a characteristic-paralleling feature, due to which the solution on each side of the discontinuity substantially gets information from the same side.

Calculation of normal vectors.

To compute each position of the discontinuity in step 3), the normal vector (α, β) is needed. This can be numerically computed by interprolating the nearby discontinuity positions. For example, A is a position of the discontinuity in a horizontal critical mesh interval (as shown in Figure 2.2). Choosing (A, B) as the interprolation stencil one can compute (α, β) with first order accuracy,

$$\begin{aligned} \alpha &= -\frac{y_A - y_B}{r_{AB}} \\ \beta &= \frac{x_A - x_B}{r_{AB}}, \end{aligned} \quad (2.17)$$

where (x_A, y_A) and (x_B, y_B) are the coordinates of A and B and r_{AB} is the distance between A and B, i.e.,

$$r_{AB} = ((x_A - x_B)^2 + (y_A - y_B)^2)^{1/2}.$$

Choosing (A, B, C) as the interprolation stencil one can compute (α, β) with

second order accuracy,

$$\alpha = -(r_{AC}\frac{y_A - y_B}{r_{AB}} - r_{AB}\frac{y_A - y_C}{r_{AC}})/(r_{AC} - r_{AB})$$
$$\beta = (r_{AC}\frac{x_A - x_B}{r_{AB}} - r_{AB}\frac{x_A - x_C}{r_{AC}})/(r_{AC} - r_{AB}). \tag{2.18}$$

However, the interprolation stencils are critical to the stability of the computation. Numerical experiments show that an arbitrary selection of the stencils, say, symmetrically picking positions on the left and right hands of A, may produce wiggles for the discontinuity curve and finally spoil the computation. The following discussion gives a stable way to select the interprolation stencils.

According to (2.3), the movement of a discontinuity curve of (1.1) at each point is a combination of two movements, a horizontal movement with a speed $[f]/[u]$ and a vertical movement with a speed $[g]/[u]$. Based on this observation, the interprolation stencils can be selected in an up-wind way. For example, to compute the normal vector at point A, we first evaluate $s_A = (g(u_A^r) - g(u_A^l))/(u_A^r - u_A^l)$, which is an approximation of $[g]/[u]$ at A, where u_A^l and u_A^r are the extrapolated data from the left and right side along the x direction at point A. If $s_A > 0$, which indicates that the discontinuity at this point moves vertically upward, we select $A, B, C, \cdots$ as the interprolation stencil; otherwise, $s_A < 0$, which indicates that the discontinuity at this point moves vertically downward, we select $A, D, E, \cdots$ as the interprolation stencil. The vertical critical mesh intervals can be treated symmetrically.

When a stencil is of two points, it is easy to see how this selection eliminates wiggles to stablize the computation. Assume that $s_A > 0$ and A deviates a little bit leftward from its normal location. This deviation decreases the α and increases the β in (2.17), which increases the x speed in (2.1). Therefore, A moves faster than normal and on the next level the wiggle disappears. On the other hand, one can easily see that selecting (A, D) as the stencil will amplify the wiggle. Other cases also can be verified easily.

Handling of triangles.

The discontinuity curve may intersect a grid mesh obliquely that it cuts out a triangle. An example is displayed in Figure 2.3, in which discontinuity curve C cuts out a triangle ΔOAB from a grid mesh T. Some particular handlings for this triangle are necessary.

First, when a triangle is very small, the discontinuity positions in the two related horizontal and vertical critical mesh intervals, i.e. A and B in Figure 2.3, will be very close to each other. Therefore, interprolation stencils for computing normal vectors might have two very close adjacent points if we successively choose discontinuity positions on the discontinuity curve for the stencils. This type stencils might not be good since these two points together with other points in a stencil might not be in a smooth pattern. To avoid these stencils, a criterion based on observation of distances between adjacent positions is set up in choosing points; thus, when A and B in Figure 2.3 are too close to each other, say, the corresponding distance is less than a constant related to the mesh size, we give up B and chooses C for the stencil.

Second, the movements of the two related horizontal and vertical critical mesh intervals must match each other to keep the continuity of the discontinuity curve. Denote the two related horizontal and vertical critical mesh intervals that contain A and B by CM_A and CM_B respectively. Obviously, only the following two cases are accepted: either both CM_A and CM_B move across grid point O to their adjacent mesh intervals, or none of them does. There should be different ways to accomplish it, and this paper does it in a simple way. We define a direction for the discontinuity curve, which gives an order for the discontinuity position on it. If B is behind A in the order, CM_B's movement must follow CM_A's, even though the new position of B might be a little bit out of its critical mesh interval on the new level. The numerical experiments show that such a little deviation from the corresponding critical mesh intervals for discontinuity positions does not cause problems.

When both A and B in Figure 2.3 cross O, the datum of the numerical solution at O is updated by the mean value of the extrapolated data from the x and y directions.

Cases for small α or β.

When one of the components of normal vectors, i.e. α or β, is very small, which indicates that the discontinuity curve almost parallels horizontal or vertical grid lines, the two dimensional treatment has some geometrical feature the one dimensional version does not have.

First, the moving speeds evaluated by (2.1) and (2.2) might be very big; therefore, positions of the discontinuity might cross more than one mesh intervals along grid lines in one timestep, no metter how the mesh ratio is restricted. Figure 2.4-a gives such an example, in which horizontal critical mesh

interval CM_A moves two mesh intervals to the right when the discontinuity curve moves from C^n to C^{n+1} in one timestep. The direction of the discontinuity curve determines whether CM_A and CM_B cross O to their adjacent mesh intervals. However, since α is very small, A's new position might have a rather big error, it is improper that the direction still determines whether CM_A and CM_C cross O_1 and whether CM_A and CM_D cross O_2, and so on. We stipulate that CM_A's movement follow CM_C's and CM_D's movements so that the later two ones dominate the situation. If CM_A crosses O_1 and O_2, the data at these points will be updated by the extrapolated data along the y direction. Also an adjustment of A is necessary if it deviates too much (say, more than one mesh interval) from its critical mesh interval, since this may cause problems in computing normal vectors. A simple way is to cut off the part out of its critical mesh interval.

Second, the discontinuity might lose critical mesh intervals when it moves. Figure 2.4-b gives such an example, in which critical mesh intervals CM_A and CM_B are lost when the discontinuity curve moves from C^n to C^{n+1}.

Third, the discontinuity might get new critical mesh intervals when it moves. Figure 2.4-c gives such an example, in which critical mesh intervals CM_A and CM_B are got when the discontinuity curve moves from C^n to C^{n+1}. There are different ways to calculate discontinuity positions for the new-generated critical mesh intervals; however, in this paper we simply choose their middle points as the positions of discontinuity.

The adjustment of discontinuity positions and the selection of discontinuity positions for the new-generated critical mesh intervals are not very accurate and, therefore, the discontinuity curve might have small wiggles there. However, the numerical experiments show that the stability mechanism implemented by proper selection of interprolation stencils described before will smooth the curve soon after in computation.

It is easy to see that the two dimensional treatment, just as its one dimensional version, does not need a lower dimensional grid for the discontinuity, nor adaptive schemes to compute the numerical solution near the discontinuity. The whole algorithm still proceeds on the regular grid. And also, if the underlying scheme is high order accurate, by using high order extrapolation and interpolation in the computation of the extrapolated data and the normal vectors one can get high order spatial accuracy for the overall algorithm.

We follow the way described in §6 in [4] to incorporate the treatment

for (2.10) into the two step predictor-corrector method, which substantially applies to any Runge-Kutta methods.

1) The treatment should perform in each of the predictor and corrector steps; however, only the predictor step moves the critical mesh intervals according to their new discontinuity positions.

2) For a critical mesh interval, the predictor step gives a discontinuity position $\xi^{n+1/2}$, and the corrector step gives a position $\bar{\xi}^{n+1}$. The discontinuity position on the new level is finally given by

$$\xi^{n+1} = \frac{1}{2}(\xi^n + \bar{\xi}^{n+1}). \tag{2.19}$$

3) When a critical mesh interval moves to its adjacent mesh interval in predictor step, at the point at which the numerical solution is updated, $u^n_{i,j}$ in the second formula in the corrector step should be replaced accordingly by an extrapolated datum.

3. Conservation Feature of the Treatment

We first recall the conservation feature of the treatment for the one dimensional case. The treatment can be written into a conservation-like form by introducing some artificial terms; i.e.,

$$u^{n+1}_j = u^n_j - \lambda(\hat{f}^n_{j+1/2} - \hat{f}^n_{j-1/2}) + p^n_{j+1/2} - p^n_{j-1/2} + q^{n+1}_j - q^n_j, \tag{3.1}$$

where the artificial terms p^n and q^n are nonzero only in vicinity of critical cells. q^n is called the local conservation error since $\sum_j (u^n_j - q^n_j)$ is conserved. [4] proved that

$$q^n_{j_1} = \frac{(\xi^n - x_{j_1+1/2})(u^n_{j_1+1} - u^n_{j_1})}{h} + O(1) \tag{3.2}$$

for the first order version of the treatment and

$$q^n_{j_1} = \frac{(u^n_{j_1} - u^{n,+}_{j_1})(x_{j_1+1} - \xi^n)^2 + (u^n_{j_1+1} - u^{n,-}_{j_1+1})(\xi^n - x_{j_1})^2}{2h^2} + O(h) \tag{3.3}$$

for the second order version of the treatment, where $[x_{j_1}, x_{j_1+1}]$ is the critical cell with a discontinuity position ξ^n, $x_{j_1+1/2} = \frac{1}{2}(x_{j_1} + x_{j_1+1})$. It is easy to see

from (3.2) and (3.3) that $q^n_{j_1}$ is uniformly bounded if the numerical solution is uniformly bounded and ξ^n has at least first order accuracy. As a result,

$$\sum_j u^n_j h = \sum_j u^0_j h + O(h), \tag{3.4}$$

which means that the numerical solution is almost conserved. A geometrical understanding of (3.2) is as follows: Usually, the numerical solution is considered to be a piecewise constant function

$$U^n(x) = \begin{cases} u^n_j & \text{when } x_j < x \le x_{j+1/2} \\ u^n_{j+1} & \text{when } x_{j+1/2} < x \le x_{j+1} \end{cases} \tag{3.5}$$

in a cell $[x_j, x_{j+1}]$. If the numerical solution in the critical cell $[x_{j_1}, x_{j_1+1}]$ is considered to be

$$\hat{U}^n(x) = \begin{cases} u^n_{j_1} & \text{when } x_{j_1} < x \le \xi^n \\ u^n_{j_1+1} & \text{when } \xi^n < x \le x_{j_1+1} \end{cases} \tag{3.6}$$

since the real discontinuity is located at ξ^n, the principle part on the right side of (3.2) multiplied by h is just the portion that $\hat{U}^n(x)$ deviates from $U^n(x)$ (as shown in Figure 3.1). (3.3) can be understood in the same way if we consider the numerical solution to be a piecewise linear function.

Now let's turn to the two dimensional case. Since the underlying scheme (2.6) is a combination of two Euler forward versions, it is sufficient to discuss the treatment's conservation feature with (2.10) as the underlying scheme. The two dimensional treatment also can be written in a conservation-like form if there is no losing and obtaining critical mesh size, and no adjustment of discontinuity positions. First, we denote

$$L_{\hat{f},\hat{g}}(\mathbf{u}^n)_{i,j} = u^n_{i,j} - L_{\hat{f}}(\mathbf{u}^n)_{i,j} - L_{\hat{g}}(\mathbf{u}^n)_{i,j}. \tag{3.7}$$

Then

$$\begin{aligned} u^{n+1}_{i,j} = \; & L_{\tilde{f},\tilde{g}}(\mathbf{u}^n)_{i,j} \\ & + p^n_{x,i+1/2,j} - p^n_{x,i-1/2,j} + q^{n+1}_{x,i,j} - q^n_{x,i,j} \\ & + p^n_{y,i,j+1/2} - p^n_{y,i,j-1/2} + q^{n+1}_{y,i,j} - q^n_{y,i,j} \end{aligned} . \tag{3.8}$$

$\tilde{f}$, p_x, and q_x are defined as follows: When $[x_{i_1}, x_{i_1+1}]$ is a horizontal critical mesh interval on the grid line $y = y_j$,

$$\tilde{f}^n_{i+1/2,j} = \begin{cases} \hat{f}^{n,l}_{i+1/2,j} & \text{when } i < i_1 \\ \hat{f}^{n,r}_{i+1/2,j} & \text{when } i \ge i_1 \end{cases}, \tag{3.9}$$

where $\hat{f}^{n,l}_{i+1/2,j}$ and $\hat{f}^{n,r}_{i+1/2,j}$ are the numerical fluxes using extrapolated data from the left and right sides respectively. When the critical mesh interval moves one mesh interval to the left,

$$\begin{aligned} p^n_{x,i_1-1/2,j} = \; & -q^n_{x,i_1,j} + (\alpha^n_{x,i_1,j})^2(u^n_{x,i_1,j} - u^{n,r}_{x,i_1,j}) + \frac{\lambda}{2}(\hat{f}^{n,l}_{i_1-1/2,j} - \hat{f}^{n,r}_{i_1-1/2,j}) \\ & -(\alpha^n_{x,i_1,j})^2(u^{n+1}_{i_1,j} - L_{\hat{f},\hat{g}}(\mathbf{u}^{n,r})_{i_1,j}) \end{aligned} \quad ;$$

$$q^{n+1}_{x,i_1-1,j} = -p^n_{x,i_1-1/2,j} \tag{3.10}$$

when the critical mesh interval remains in the same mesh interval,

$$q^{n+1}_{x,i_1,j} = q^n_{x,i_1,j} + \frac{\lambda}{2}(\hat{f}^{n,r}_{i_1+1/2,j} - \hat{f}^{n,l}_{i_1+1/2,j}); \tag{3.11}$$

and when the critical mesh interval moves one mesh interval to the right,

$$p^n_{x,i_1+1/2,j} = q^n_{x,i_1,j} + \frac{\lambda}{2}(\hat{f}^{n,r}_{i_1+1/2,j} - \hat{f}^{n,l}_{i_1+1/2,j})$$

$$\begin{aligned} q^{n+1}_{x,i_1+1,j} = \; & q^n_{x,i_1,j} + (\alpha^n_{x,i_1,j})^2(u^{n,l}_{x,i_1+1,j} - u^n_{x,i_1+1,j}) + \frac{\lambda}{2}(\hat{f}^{n,r}_{i_1+3/2,j} - \hat{f}^{n,l}_{i_1+3/2,j}) \\ & +(\alpha^n_{x,i_1,j})^2(u^{n+1}_{i_1+1,j} - L_{\hat{f},\hat{g}}(\mathbf{u}^{n,l})_{i_1+1,j}) \end{aligned} \tag{3.12}$$

Here, $u^{n,l}_{x,i,j}$ and $u^{n,r}_{x,i,j}$ are extrapolated data from the left and right sides respectively, $\alpha^n_{x,i,j}$ is the horizontal component of the normal vector corresponding to the critical mesh interval. The two dimensional treatment has a large-step behavior described in the previous section; i.e., critical mesh intervals may move across several mesh intervals. Corresponding formulas also can be set up for these cases. For example, when a horizontal critical mesh interval located at $[x_{i_1}, x_{i_1+1}]$ moves across several mesh intervals to left to

$[x_{i_2}, x_{i_2+1}]$,

$$
\begin{aligned}
p^n_{x,i_1-1/2,j} = \; & -q^n_{x,i_1,j} - (\alpha^n_{x,i_1,j})^2(u^n_{i_1,j} - u^{n,r}_{i_1,j}) + \frac{\lambda}{2}(\hat{f}^{n,l}_{i_1-1/2,j} - \hat{f}^{n,r}_{i_1-1/2,j}) \\
& -(\alpha^n_{x,i_1,j})^2(u^{n+1}_{i_1,j} - L_{\hat{f},\hat{g}}(\mathbf{u}^{n,r})_{i_1,j}) \\
p^n_{x,i_1-3/2,j} = \; & -q^n_{x,i_1,j} - (\alpha^n_{x,i_1,j})^2(u^n_{i_1,j} - u^{n,r}_{i_1,j}) + (\alpha^n_{x,i_1,j})^2(u^n_{i_1-1,j} - u^{n,r}_{i_1-1,j}) \\
& +\frac{\lambda}{2}(\hat{f}^{n,l}_{i_1-3/2,j} - \hat{f}^{n,r}_{i_1-3/2,j}) - (\alpha^n_{x,i_1,j})^2(u^{n+1}_{i_1,j} - L_{\hat{f},\hat{g}}(\mathbf{u}^{n,r})_{i_1,j}) \\
& -(\alpha^n_{x,i_1,j})^2(u^{n+1}_{i_1-1,j} - L_{\hat{f},\hat{g}}(\mathbf{u}^{n,r})_{i_1-1,j}) \\
& \cdots\cdots\cdots \\
p^n_{x,i_2+1/2,j} = \; & -q^n_{x,i_1,j_1} - \sum_{i=i_1}^{i_2+1}(\alpha^n_{x,i_1,j})^2(u^n_{i,j} - u^{n,r}_{i,j}) + \frac{\lambda}{2}(\hat{f}^{n,l}_{i_2+1/2,j} - \hat{f}^{n,r}_{i_2+1/2,j}) \\
& - \sum_{i=i_1,}^{i_2+1}(\alpha^n_{x,i_1,j})^2(u^{n+1}_{i,j} - L_{\hat{f},\hat{g}}(\mathbf{u}^{n,r})_{i,j}) \\
q^{n+1}_{x,i_2,j} = \; & -p^n_{i_2+1/2,j}
\end{aligned}
\tag{3.13}
$$

Similar formulas also can be set up for the case that critical mesh intervals move across several mesh intervals to the right.

$\tilde{g}$, p_y, and q_y are defined almost symmetrically, except that we use $(1 - (\alpha^n_{x,i_*,j_*})^2)$ rather than $(\beta^n_{y,i,j})^2$. That is, assume $[y_{j_1}, y_{j_1+1}]$ is a vertical critical mesh interval on the grid line $x = x_i$, then

$$
\tilde{g} = \begin{cases} \hat{g}^{n,b}_{i,j+1/2} & \text{when } j < j_1 \\ \hat{g}^{n,t}_{i,j+1/2} & \text{when } j \geq j_1 \end{cases}, \tag{3.14}
$$

where $\hat{g}^{n,b}_{i,j+1/2}$ and $\hat{g}^{n,t}_{i,j+1/2}$ are the numerical fluxes using extrapolated data from the top and bottom respectively. When the critical mesh interval moves

one mesh interval to the bottom,

$$\begin{aligned} p^n_{y,i,j_1-1/2} = \; & -q^n_{y,i,j_1} + (1-(\alpha^n_{x,i_*,j_*})^2)(u^n_{y,i,j_1} - u^{n,t}_{y,i,j_1}) \\ & + \frac{\lambda}{2}(\hat{g}^{n,b}_{i,j_1-1/2} - \hat{g}^{n,t}_{i,j_1-1/2}) \\ & -(1-(\alpha^n_{x,i_*,j_*})^2)(u^{n+1}_{i,j_1} - L_{\hat{f},\hat{g}}(\mathbf{u}^{n,t})_{i,j_1}) \quad ; \end{aligned} \tag{3.15}$$

$$q^{n+1}_{y,i,j_1-1} = -p^n_{y,i,j_1-1/2}$$

and when the critical mesh interval remains in the same mesh interval,

$$q^{n+1}_{y,i,j_1} = q^n_{y,i,j_1} + \frac{\lambda}{2}(\hat{f}^{n,t}_{i,j_1+1/2} - \hat{f}^{n,b}_{i,j_1+1/2}); \tag{3.16}$$

and so on. The index i_* and j_* are as follows: $[x_{i_*}, x_{i_*+1}]$ is the horizontal critical mesh interval on the grid line $y = y_{j_*}$ that also crosses the grid point the considered vertical critical mesh interval crosses. For example, if CM_C in Figure 2.4-a is the critical mesh interval $[y_{j_1}, y_{j_1+1}]$ on the grid line $x = x_i$, the critical mesh interval $[x_{i_*}, x_{i_*+1}]$ on the grid line $y = y_{j_*}$ should be CM_A. So that the coefficients of $u^{n+1}_{i,j}$'s in the right side of (3.8) are 1.

There are terms like $(u^{n+1}_{i_1,j} - L_{\hat{f},\hat{g}}(\mathbf{u}^{n,l})_{i_1,j})$ in (3.10), (3.12), (3.13) and (3.15), which are of $O(h)$, since when critical mesh intervals cross grid points the numerical solution is updated by the extrapolated data of $\mathbf{u}^{n+1}$ at them.

To show the conservation feature of the treatment, we separate the local conservation error q^n into two parts; i.e.,

$$q^n = \bar{q}^n + \bar{\bar{q}}^n. \tag{3.16}$$

In the case corresponding to (3.10),

$$\begin{aligned} \bar{\bar{q}}^{n+1}_{x,i_1-1,j} = \; & \bar{\bar{q}}^n_{x,i_1,j} + \frac{\lambda}{2}((\beta^n_{x,i_1,j})^2(\hat{f}^{n,r}_{i_1-1/2,j} - \hat{f}^{n,l}_{i_1-1/2,j}) \quad ; \\ & -\alpha^n_{x,i_1,j}\beta^n_{x,i_1,j}(\hat{g}^{n,r}_{i_1-1/2,j} - \hat{g}^{n,l}_{i_1-1/2,j})) \end{aligned} \tag{3.17}$$

in the case corresponding to (3.11),

$$\begin{aligned} \bar{\bar{q}}^{n+1}_{x,i_1,j} = \; & \bar{\bar{q}}^n_{x,i_1,j} + \frac{\lambda}{2}((\beta^n_{x,i_1,j})^2(\hat{f}^{n,r}_{i_1+1/2,j} - \hat{f}^{n,l}_{i_1+1/2,j}) \quad ; \\ & -\alpha^n_{x,i_1,j}\beta^n_{x,i_1,j}(\hat{g}^{n,r}_{i_1+1/2,j} - \hat{g}^{n,l}_{i_1+1/2,j})) \end{aligned} \tag{3.18}$$

in the case coresponding to (3.12),

$$\begin{aligned}\bar{\bar{q}}^{n+1}_{x,i_1+1,j} = \; & \bar{\bar{q}}^{n}_{x,i_1,j} + \frac{\lambda}{2}((\beta^n_{x,i_1,j})^2(\hat{f}^{n,r}_{i_1+3/2,j} - \hat{f}^{n,l}_{i_1+3/2,j}) \\ & -\alpha^n_{x,i_1,j}\beta^n_{x,i_1,j}(\hat{g}^{n,l}_{i_1+3/2,j} - \hat{g}^{n,l}_{i_1+3/2,j})\end{aligned} \; ; \tag{3.19}$$

and in the case corresponding to (3.13),

$$\begin{aligned}\bar{\bar{q}}^{n+1}_{x,i_2,j} = \; & \bar{\bar{q}}^{n}_{x,i_1,j} + \frac{\lambda}{2}((\beta^n_{x,i_1,j})^2(\hat{f}^{n,r}_{i_2+1/2,j} - \hat{f}^{n,l}_{i_2+1/2,j}) \\ & -\alpha^n_{x,i_1,j}\beta_{x,i_1,j}(\hat{g}^{n,r}_{i_2+1/2,j} - \hat{g}^{n,l}_{i_2+1/2,j})\end{aligned} \; ; \tag{3.20}$$

and so on, where

$$\hat{g}^n_{i+1/2,j} = \hat{g}(u^n_{i-k+1,j}, \cdots, u^n_{i+k,j}).$$

$\bar{q}^n_{x,i,j}$ is just

$$\bar{q}^n_{x,i,j} = q^n_{x,i,j} - \bar{\bar{q}}^n_{x,i,j}$$

q^n_y is separated in a symmetrical way as the q^n_x is. The following theorem concerns the conservation feature of the treatment.

Theorem 3.1 *Assume that the solution to (1.1) is a piecewise smooth function with a smooth discontinuity curve. If both the underlying scheme and the treatment are at least first order accurate (which implies that the numerical solution and positions of the discontinuity are at least first order accurate too), and*

$$\min(|\alpha^n|, |\beta|^n) \geq c_0 > 0, \tag{3.21}$$

then

$$\begin{aligned}\bar{q}^n_{x,i_1,j} &= \frac{(\alpha^n_{x,i_1,j})^2(\xi^n_x - x_{i_1+1/2})(u^n_{i_1+1,j} - u^n_{i_1,j})}{h} + O(1) \\ \bar{q}^n_{y,i,j_1} &= \frac{(\beta^n_{y,i,j_1})^2(\xi^n_y - y_{j_1+1/2})(u^n_{i,j_1+1} - u^n_{i,j_1})}{h} + O(1)\end{aligned} \; , \tag{3.22}$$

where ξ^n_x and ξ^n_y are discontinuity positions in a horizontal critical mesh interval $[x_{i_1}, x_{i_1+1}]$ on the grid line $y = y_j$ and a vertical critical mesh interval $[y_{j_1}, y_{j_1+1}]$ on the grid line $x = x_i$ respectively, and $\bar{\bar{q}}^n_{x,i,j} + \bar{\bar{q}}^n_{y,i,j}$ is of $O(h)$ in a weak sense that for any domain D in (x, y)-plane,

$$\sum_{(x_i,y_j)\in D} (\bar{\bar{q}}^n_{x,i,j} + \bar{\bar{q}}^n_{y,i,j})h^2 = O(h). \tag{3.23}$$

The condition (3.21) implies that the discontinuity curve intersects the grid obliquely, so that a horizontal and a vertical mesh intervals that cross a same grid point (e.g. the critical mesh sizes CM_A and CM_C in Figure 2.4-a) only have a distance in a range of $O(h)$. We believe that such a restriction is not essential but technical, and by more careful (therefore, more complicated) estimation we can get rid of it.

The theorem is only a first order result corresponding to (3.2) in the one dimensional case. A second order result corresponding to (3.3) could be obtained; however, it needs to consider (2.1) as the underlying scheme and the discussion might be very complicated. Technically, Theorem 3.1 is sufficient to display the treatment's conservation feature since, obviously, it implies the uniform boundedness of $\bar{q}_x^n$ and $\bar{q}_y^n$, and therefore,

$$\sum_i \sum_j u_{i,j}^{n+1} h^2 = \sum_i \sum_j u_{i,j}^0 h^2 + O(h). \tag{3.24}$$

Proof. The proof of (3.22) is almost the same as that in [4] for one dimensional case. It is sufficient only to deal with the first equation in (3.22) since the second one can be handled almost symmetrically by noticing the condition (3.21). Denote

$$S_x^n = (\alpha_{x,i_1,j}^n)^2(\xi_x^n - x_{i_1+1/2})(u_{i_1+1,j}^n - u_{i_1,j}^n) - h\bar{q}_{x,i_1,j}^n, \tag{3.25}$$

and we are going to show that

$$S_x^{n+1} - S_x^n = O(h^2), \tag{3.26}$$

from which (3.22) follows. Following the same derivation of (3.17) and (3.19) in [4], we have

$$\begin{aligned} S_x^{n+1} - S_x^n = \; & (\alpha_{x,i_1,j}^n)^2 d(u_{i_1+1/2,j}^{n+1/2} - u_{i_1-1/2,j}^{n+1/2}) \\ & -\tau(((\alpha_{x,i_1,j}^n)^n)^2(\hat{f}_{i_1-1/2,j}^{n,r} - \hat{f}_{i_1-1/2,j}^{n,l}) \\ & +\alpha_{x,i_1,j}^n \beta_{x,i_1,j}^n(\hat{g}_{i_1-1/2,j}^{n,r} - \hat{g}_{i_1-1/2,j}^{n,l})) + O(h^2) \end{aligned} \tag{3.27}$$

when the critical mesh interval moves one mesh interval to the left,

$$\begin{aligned} S_x^{n+1} - S_x^n = \; & (\alpha_{x,i_1,j}^n)^2 d(u_{i_1+1,j}^{n+1/2} - u_{i_1,j}^{n+1/2}) \\ & -\tau(((\alpha_{x,i_1,j}^n)^2(\hat{f}_{i_1+1/2,j}^{n,r} - \hat{f}_{i_1+1/2,j}^{n,l}) \\ & +\alpha_{x,i_1,j}^n \beta_{x,i_1,j}^n(\hat{g}_{i_1+1/2,j}^{n,r} - \hat{g}_{i_1+1/2,j}^{n,l})) + O(h^2) \end{aligned} \tag{3.28}$$

when the critical mesh interval does not move, and

$$\begin{aligned} S_x^{n+1} - S_x^n = \; & (\alpha^n_{x,i_1,j})^2 d(u^{n+1/2}_{i+3/2,j} - u^{n+1/2}_{i_1+1/2,j}) \\ & -\tau(((\alpha^n_{x,i_1,j})^2(\hat{f}^{n,r}_{i_1+3/2,j} - \hat{f}^{n,l}_{i_1+3/2,j}) \\ & +\alpha^n_{x,i_1,j}\beta^n_{x,i_1,j}(\hat{g}^{n,r}_{i_1+3/2,j} - \hat{g}^{n,l}_{i_1+3/2,j})) + O(h^2) \end{aligned} \tag{3.29}$$

when the critical mesh interval moves one mesh interval to the right. Here $d = \xi_x^{n+1} - \xi_x^n$, $u_{i,j}^{n+1/2} = \frac{1}{2}(u_{i,j}^{n+1} + u_{i,j}^n)$, and $u_{i+1/2,j}^{n+1/2} = \frac{1}{2}(u_{i+1,j}^{n+1} + u_{i,j}^n)$. Since the principle terms on the right of (3.27), (3.28) and (3.29) are first order approximations to (2.1), (3.26) follows easily. The case when the critical mesh interval moves several mesh intervals to the left or right can be handled similarly by noticing condition (3.21).

Denote by T^n the left side of (3.23). It is not difficult to see that $T^{n+1} - T^n$ is a first order approximation to the following integral:

$$\tau \int_C \{[f]((\beta)^2 dx + \alpha\beta dy) + [g](\alpha\beta dx + (\alpha)^2 dy)\} = 0, \tag{3.30}$$

where C is the portion of the discontinuity curve inside Domain D and (α, β) is the normal vector of C; hence (3.23) follows. Thus ends the proof.

4. Treatment for Interactions of Discontinuities

For simplicity, we assume that f and g are convex. When two straight line discontinuities L_1 and L_2 meet at a point O, with three constant states U_2, U_3, and U_1 between them, a third straight line discontinuity L_3 may be formed from O connecting U_2 and U_1 (as shown in Figure 4.1). A problem is when given L_1 and L_2, and U_2, U_3 and U_1, what the third discontinuity L_3, (if there is one) should be. Our treatment for interactions of discontinuities is based on the observation of this problem. Without losing generality we suppose that O is the origin of the (x, y)-plane. Since L_1 connects U_2 and U_3, its normal propagating speed is

$$S_1 = \frac{\alpha_1[f]_{2,3} + \beta_1[g]_{2,3}}{[U]_{2,3}}, \tag{4.1}$$

where (α_1, β_1) is the unit normal vector of L_1, $[f]_{2,3}$ and $[g]_{2,3}$ are the jumps of the two fluxes across L_1. and $[u]_{2,3}$ is the jump of the solution across L_1. Hence, the equation for L_1 is

$$\alpha_1 x + \beta_1 y = S_1 t. \tag{4.2}$$

Also the equation for L_2 is

$$\alpha_2 x + \beta_2 y = S_2 t, \tag{4.3}$$

where

$$S_2 = \frac{\alpha_2[f]_{3,1} + \beta_2[g]_{3,1}}{[U]_{3,1}}. \tag{4.3}$$

Since L_3 connects U_2 and U_1, its normal propagating speed is

$$S_3 = \frac{\alpha_3[f]_{1,2} + \beta_3[g]_{1,2}}{[U]_{1,2}}; \tag{4.4}$$

therefore, L_3 should be

$$\alpha_3 x + \beta_3 y = S_3 t, \tag{4.5}$$

where (α_3, β_3) is the unit normal vector of L_3 and is to be solved. As the straight lines L_1, L_2, and L_3 meet at the same point O, the determinant

$$\begin{vmatrix} \alpha_1 & \beta_1 & S_1 \\ \alpha_2 & \beta_2 & S_2 \\ \alpha_3 & \beta_3 & S_3 \end{vmatrix} = 0. \tag{4.6}$$

from which and the restriction ${\alpha_3}^2 + {\beta_3}^2 = 1$, α_3 and β_3 can be solved. Particularly, when (α_1, β_1) and (α_2, β_2) are not parallel each other, there are k_1 and k_2 so that

$$\begin{aligned} k_1\alpha_1 + k_2\alpha_2 &= \alpha_3 \\ k_1\beta_1 + k_2\beta_2 &= \beta_3 \\ k_1 S_1 + k_2 S_2 &= S_3 \end{aligned} . \tag{4.7}$$

Now we are going to build up the treatment for interactions of discontinuities. We call the mesh that is supposed to contain a interaction point a node mesh. A node mesh should have three critical mesh intervals corresponding to three discontinuities, though two of them might overlap each other (as shown in Figure 4.2). In the vicinity of the node mesh, critical mesh intervals of different discontinuities might be very close to each other so that at some grid points the stencils of the numerical fluxes might cover all of them, just as the situation in the one dimensional case (see [4]). Therefore, fluxes with the form (4.1) in [4] might be used, and the order of the extrapolation might be lowered since we do not have enough grid points in between to keep

the same order. Also when two critical mesh intervals overlap each other on an mesh interval, they are treated separately as they are independent to each other with a hiding middle state between them before the two discontinuities positions cross each other, just as what we did in the one dimensional case (see [4]). For simplicity, we stipulate that each timestep only one of the three critical mesh intervals could move away from its original mesh interval, even though some discontinuity positions might deviate a little bit from the corresponding critical mesh intervals.

The node mesh will move, which is caused by the interaction of two of the three discontinuities. The critical mesh intervals corresponding to the two interacting discontinuities could either be the same type or different types. Figure 4.3-a and Figure 4.3-b give two typical cases. In Figure 4.3-a, horizontal critical mesh intervals CM_1 and CM_2 belonging to discontinuity L_1 and L_2 merge since their positions of discontinuities cross each other, and generate a new critical mesh interval, which belongs to L_3. During that time the node mesh moves from mesh M_1 to M_2. In Figure 4.3-b a vertical critical mesh interval CM_1 belonging to L_1 moves one mesh interval to the top so that the node mesh moves from mesh M_1 to M_2. During that time the discontinuity L_2 loses critical mesh interval CM_2, which becomes a new critical mesh interval of L_3. More general cases always can be reduced to one of them.

Just the same as in the one dimensional case, the important thing is still the calculation of the discontinuity position for the new-generated critical mesh interval of L_3. We calculate the new discontinuity position by assuming that L_1 and L_2 are straight lines and the new position is on the straight line of the discontinuity $\tilde{L}_3$, with which L_1 and L_2 form a triple point described before. $\tilde{L}_3$ might not be the real third discontinuity L_3. What we expect is a formula similar to (4.2) in [4], which calculates the new position of the discontinuity by the two old ones. Obviously, our treatment only has first order accuracy for we suppose the discontinuities to be straight lines near the triple point. However, since interactions of discontinuities only happen at a rate of $O(h^2)$ if the solution is piecewise smooth, the overall algorithm still can have second order accuracy. High order treatment could be built up by the same idea, however, will be rather complicated.

Now we begin with the case presented in Figure 4.3-a. Denote by $\xi^n_{x,1}$ and $\xi^n_{x,2}$ the discontinuity positions of the critical mesh intervals CM_1 and CM_2,

which cross each other. Also denote by $\xi^n_{x,3}$ the discontinuity position of the new-generated critical mesh interval. Denote by O the triple point formed by L_1, L_2 and L_3 on the level n with the coordinates (x_O, y_O). Since $\xi^n_{x,1}$ is on L_1, which passes O and has the normal vector (α_1, β_1), $\xi^n_{x,1}$ satisfies

$$\alpha_1(\xi^n_{x,1} - x_O) + \beta_1(y_{j_1} - y_O) = 0, \tag{4.8}$$

where $P(x_{i_1}, y_{j_1})$ is the left end point of the critical mesh interval. By the same argument,

$$\alpha_2(\xi^n_{x,2} - x_O) + \beta_2(y_{j_1} - y_O) = 0, \tag{4.9}$$

and

$$\alpha_3(\xi^n_{x,3} - x_O) + \beta_3(y_{j_1} - y_O) = 0. \tag{4.10}$$

Multiply (4.8) and (4.9) by k_1 and k_2 defined immediately after (4.6), sum them, subtract (4.10) and use (4.7), then

$$\xi^n_{x,3} = \frac{k_1\alpha_1\xi^n_{x,1} + k_2\alpha_2\xi^n_{x,2}}{\alpha_3}, \tag{4.11}$$

which is what we expect.

For the case presented in Figure 4.3-b, we adopt all the notations in Figure 4.3-a, except we denote by $\xi^n_{y,1}$ the discontinuity position of CM_1, since it is a vertical critical mesh interval. By the similar argument we have

$$\alpha_1(x_{i_1} - x_O) + \beta_1(\xi^n_{y,1} - y_O) = 0, \tag{4.12}$$

and (4.9) and (4.10), from which

$$\xi^n_{x,3} = \frac{k_1(\alpha_1 x_{i_1} + \beta_1(\xi^n_{y,1} - y_{j_1})) + k_2\alpha_2\xi^n_{x,2}}{\alpha_3}. \tag{4.13}$$

The treatment also has a conservation feature in some weak sense; however, the discussion will be quite complicated and a separated paper will mainly focus on it.

5. Numerical Experiments

In this section we shall display three numerical examples in which we use the treatment to track discontinuities. The numerical fluxes used in the underlying scheme (2.6) for both f and g are a second order TVD flux described in [7].

EXAMPLE 1. This is an example for linear case. The partial differential equation is

$$u_t + u_x + u_y = 0 \qquad |x| \le 1, |y| \le 1, \tag{5.1}$$

with initial data

$$u(x,y,0) = u_0(x,y) = \begin{cases} 0.75\cos((x+y)\pi)\cos((x-y)\pi) & x^2+y^2 \le 0.6 \\ 0 & \text{otherwise} \end{cases}, \tag{5.2}$$

and a periodic boundary condition. The exact solution of this problem is

$$u(x,y,t) = u_0(x-t, y-t), \tag{5.3}$$

which has a discontinuity circle with a radius of $\sqrt{6}$. The λ in (2.7), i.e. the mesh ratio, is chosen to be 0.5 and $\Delta x = \Delta y = 0.05$ Figure 5.1-a shows the numerical solution at $t = 2$ (160 timesteps); Figure 5.1-b shows the discontinuity circle of the numerical solution, which is obtained by connecting the discontinuity positions by straight lines; and Figure 5.1-c and Figure 5.1-d show the intersection surfaces of the numerical solution at $x = 0$ and $y = 0.45$ respectively, where the circles present the numerical solution and the solid lines presents the exact solution. Figure 5.2 shows the same stuff as Figure 5.1 at time $t = 16$ (1280 timesteps). We can see in the figures that both the numerical solution and the discontinuity positions approximate the exact ones quite well.

EXAMPLE 2. The partial differential for this example is

$$u_t + u_x + u_y = -\mu u(u-1)(u-\frac{1}{2}), \tag{5.4}$$

which is a linear advection equation with a source term that is stiff for large μ. This equation is interested since it models the reacting flow problem, and many difference methods for it would produce wrong propagating speed if its solution has a propagating discontinuity (see [3]). Chang (see [1]) have used the subcell resolution proposed by Harten (see [2]) to the one dimensional version of this problem and got a very good numerical result. Here, we use our

treatment, which is similar in some sense to the Harten's subcell technique (see [5]), to track the two dimensional propagating discontinuity.

The initial data is still (5.2), which indicates that the solution has the same propagating discontinuity curve as in the previous example. λ, Δx and Δy are chosen as before. A Strang type splitting method suggested in [3] is used to solve this problem, in which the same numerical solution operator with our treatment solves the convection equation without the source term, and a implicit predictor-corrector method for ordinary differential equations models the chemistry. Figure 5.3-a shows the numerical solution for $\mu = 0.15$(non-stiff) at time $t = 1$, and Figure 5.3-b shows the discontinuity circle at the same time. Figure 5.4 shows the same stuff as Figure 5.3 for $\mu = 150$ (stiff). In the figures we see that numerical solutions for both stiff and non-stiff cases have correct propagating speeds.

EXAMPLE 3. This is an example for nonlinear case. The partial differential equation is

$$u_t + (\frac{1}{2}u^2)_x + (\frac{1}{2}u^2)_y = 0, \tag{5.5}$$

with Riemann type initial data

$$u(x,y,0) = u_0(x,y) = \begin{cases} u_1 & x > 0, y > 0 \\ u_2 & x \le 0, y > 0 \\ u_3 & x \le 0, y \le 0 \\ u_4 & x > 0, y \le 0 \end{cases}. \tag{5.6}$$

The following three cases are tested:
case 1.

$$(u_1, u_2, u_3, u_4) = (-0.2, -1.0, 0.5, 0.8)$$

case 2.

$$(u_1, u_2, u_3, u_4) = (-1.0, 0.5, -0.2, 0.8)$$

case 3.

$$(u_1, u_2, u_3, u_4) = (-1.0, -0.2, 0.5, 0.8)$$

λ is still chosen to be 0.5 and Δx and Δy are chosen to be 0.025. Figure 5.5, 5.6, and 5.7 show the numerical results at time $t = 1$ for the case 1., case 2. and case 3. respectively. All the pictures with "a" show the numerical solutions, all the pictures with "b" show the discontinuity positions of the numerical solutions, in which the circles present the numerical positions and

the solid lines present the exact discontinuity curves; and all the pictures with "c" and "d" show the intersection surfaces of the numerical solution at $x = 0$ and $y = 0$ respectively. In the figures we can see that both the numerical solutions and the discontinuity positions approximate the exact ones very well.

The discontinuity curve in the Case 2. has a very sharp corner; however, Figure 5.6-b shows that the computation of the discontinuity positions around this corner is stable and still have a reasonable resolution. Case 3. has a triple point formed in interactions of discontinuities. The treatment for interactions is applied and the result is quite good.

6. Conclusions

We have built up a treatment for discontinuities for the two dimensional scalar conservation laws, which is the extension of the treatment for one dimensional case introduced in [4]. We also have presented the conservation feature of the treatment for a single discontinuity, for which we wrote the overall algorithm into a conservation-like form involving local conservation errors and proved that the local conservation errors are uniformly bounded. Treatment for interactions of discontinuities also has been set up; however, the corresponding conservation-feature is still expected. Numerical experiments show that the treatment is very effective in tracking discontinuities.

Acknowelegements

The author thanks professor Ami Harten, professor Stanley Osher and Dr. Rosa Donat for helpful discussions and/or correcting English.

References

[1] Shin-Hung Chang, *On the application of subcell resolution to conservation laws with stiff source terms.* NASA TM-102384 (1989).

[2] A. Harten, J. Comput. Phys. **83** 148(1989).

[3] R. J. LeVeque and H. C. Yee, *A study of numerical methods for hyperbolic conservation laws with stiff source terms.* NASA TM-100075 (1988).

[4] D. Mao, *A treatment of discontinuities for finite difference methods.* UCLA CAM Report 90-19 (1990).

[5] D. Mao, *A treatment of discontinuities in shock-capturing finite difference methods.* to appear in J. Comput. Phys.

[6] D. Mao, J. Comput. Math. **No. 3**, 256(1985) (in Chinese).

[7] S. Osher and S. Chakravarthy, *Very high order accurate TVD schemes.* ICASE Report 84.44, 1984, IMA Volume in Mathematics and its Applications, —bf Vol. 2, Springer-Verlag, 229(1986).

[8] S. Osher and S. Chakravarthy, SIAM Numer. Anal. **21**, 955(1984).

[9] S. Osher, SIAM Numer. Anal. **21**, 217(1984).

[10] Chi-Wang Shu, Math. Comp. **49**, 105(1987).

[11] D. H. Wagner, SIAM. J. Math Anal. **14** , 3 (1983).

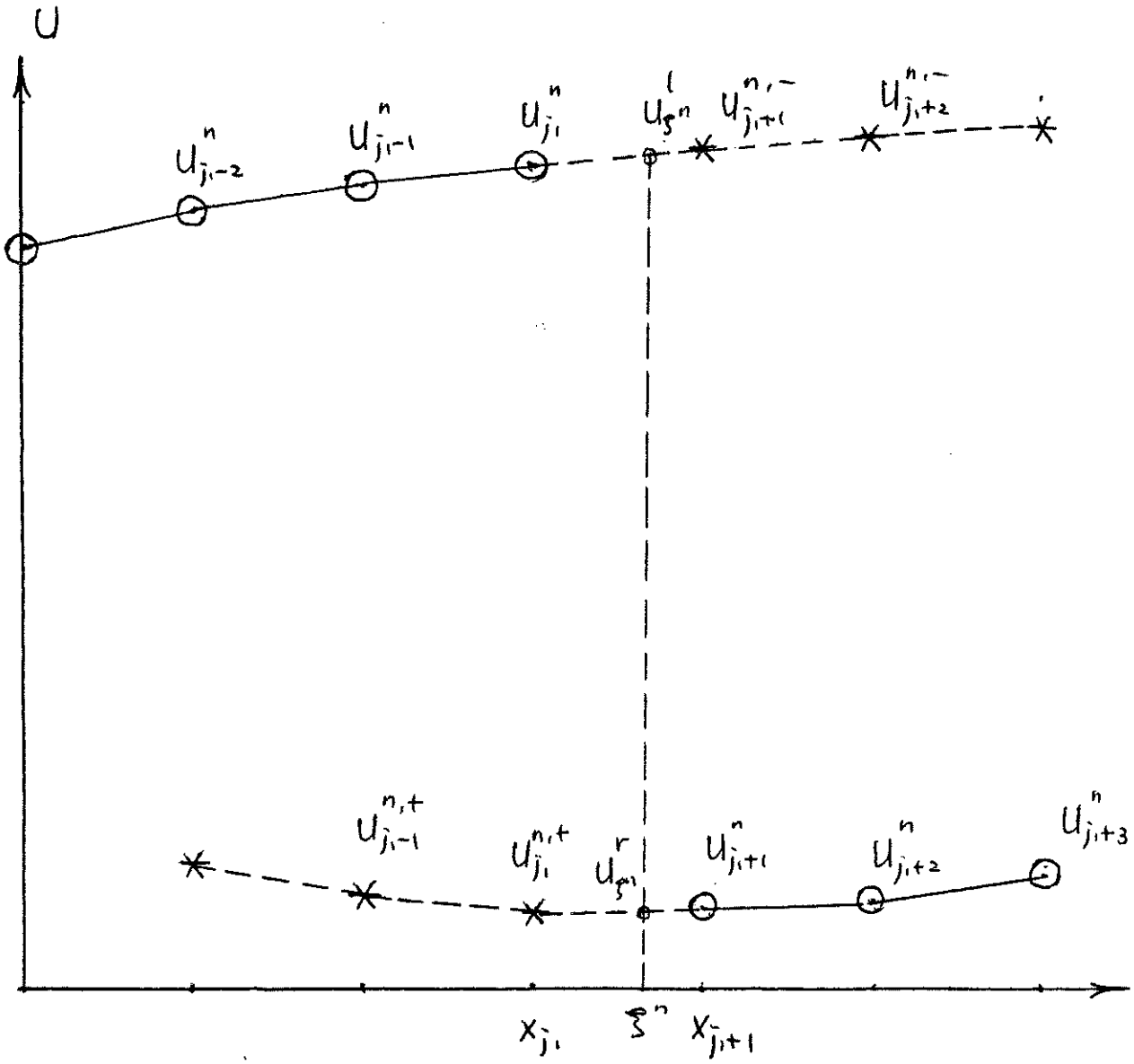


FIGURE 1.1

Numerical solution on the level n just has a jump in the critical cell $[x_{j_1}, x_{j_1+1}]$, on each side of which the numerical solution is supposed to be smooth. ξ^n is the discontinuity position in the critical cell.

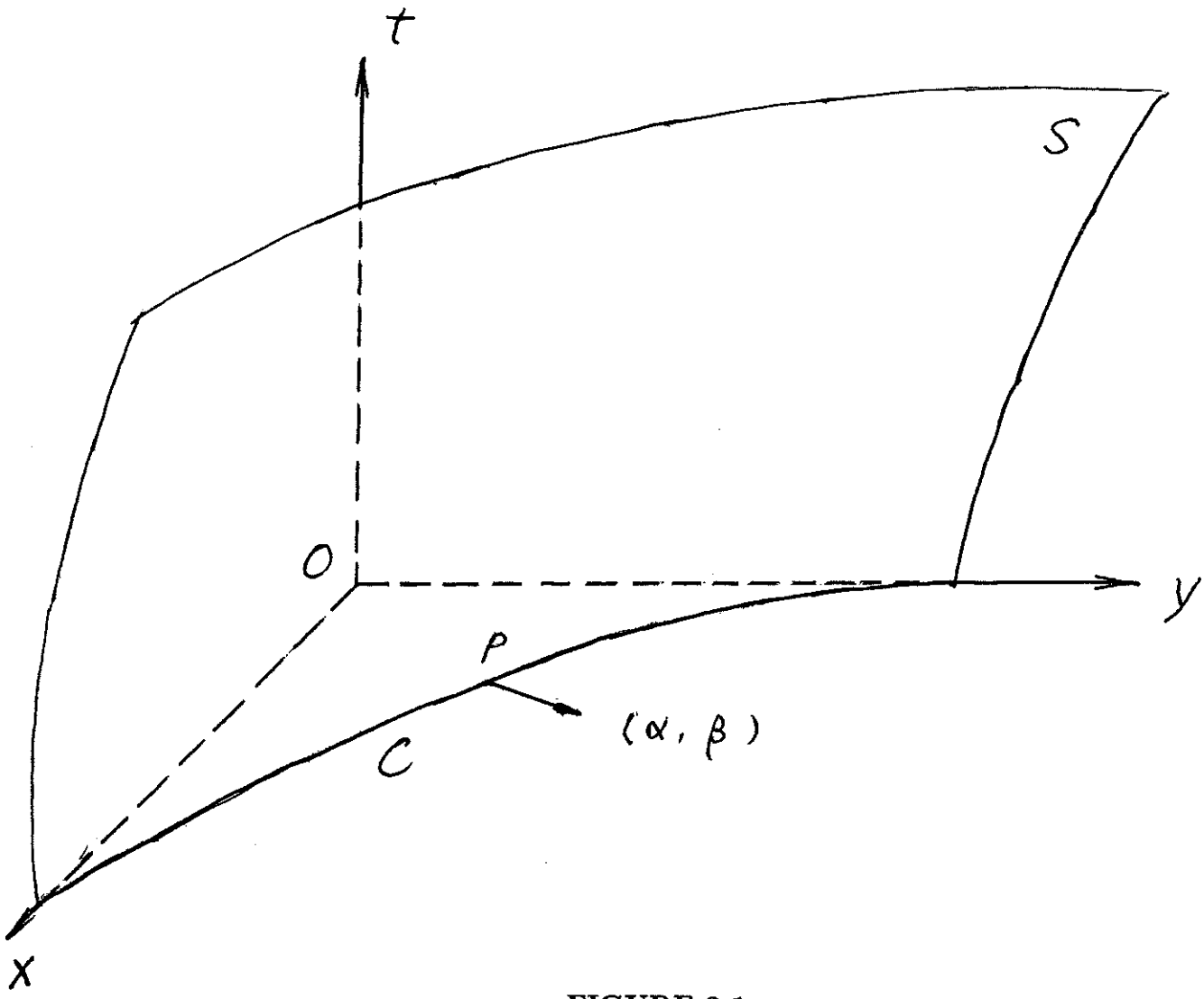
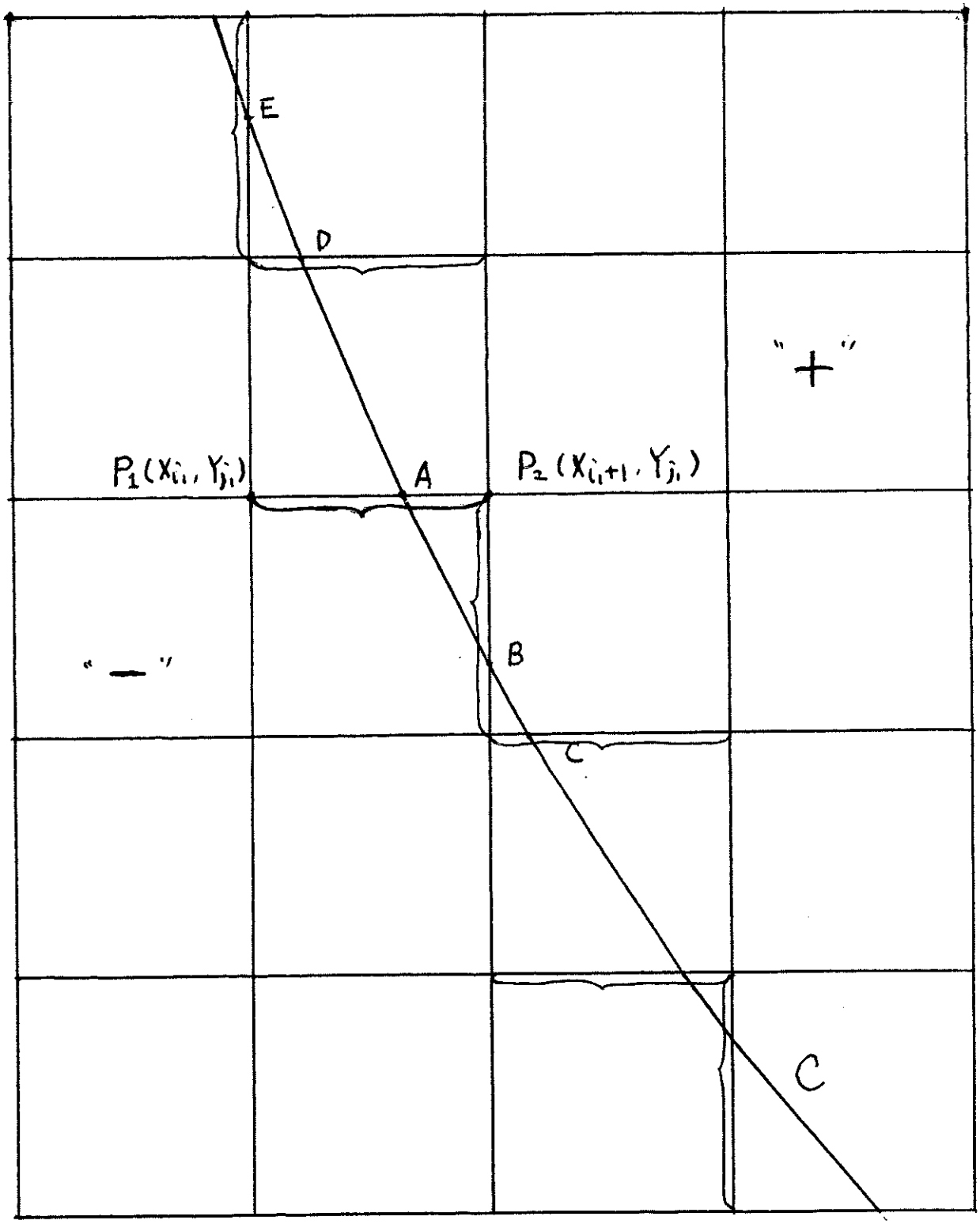


FIGURE 2.1

Discontinuity surface S cuts a horizontal plane by a curve C, (α, β) is a horizontal normal vector to S at p.



FIGHUR 2.2

Discontinuity curve C is represented by a group of critical mesh intervals, which are indicated by horizontal and vertical braces. $A, B, C, \cdots$ are discontinuity positions in the critical mesh intervals. The two sides of the numerical solution connected by C are indicated by "$-$" and "$+$" respectively.

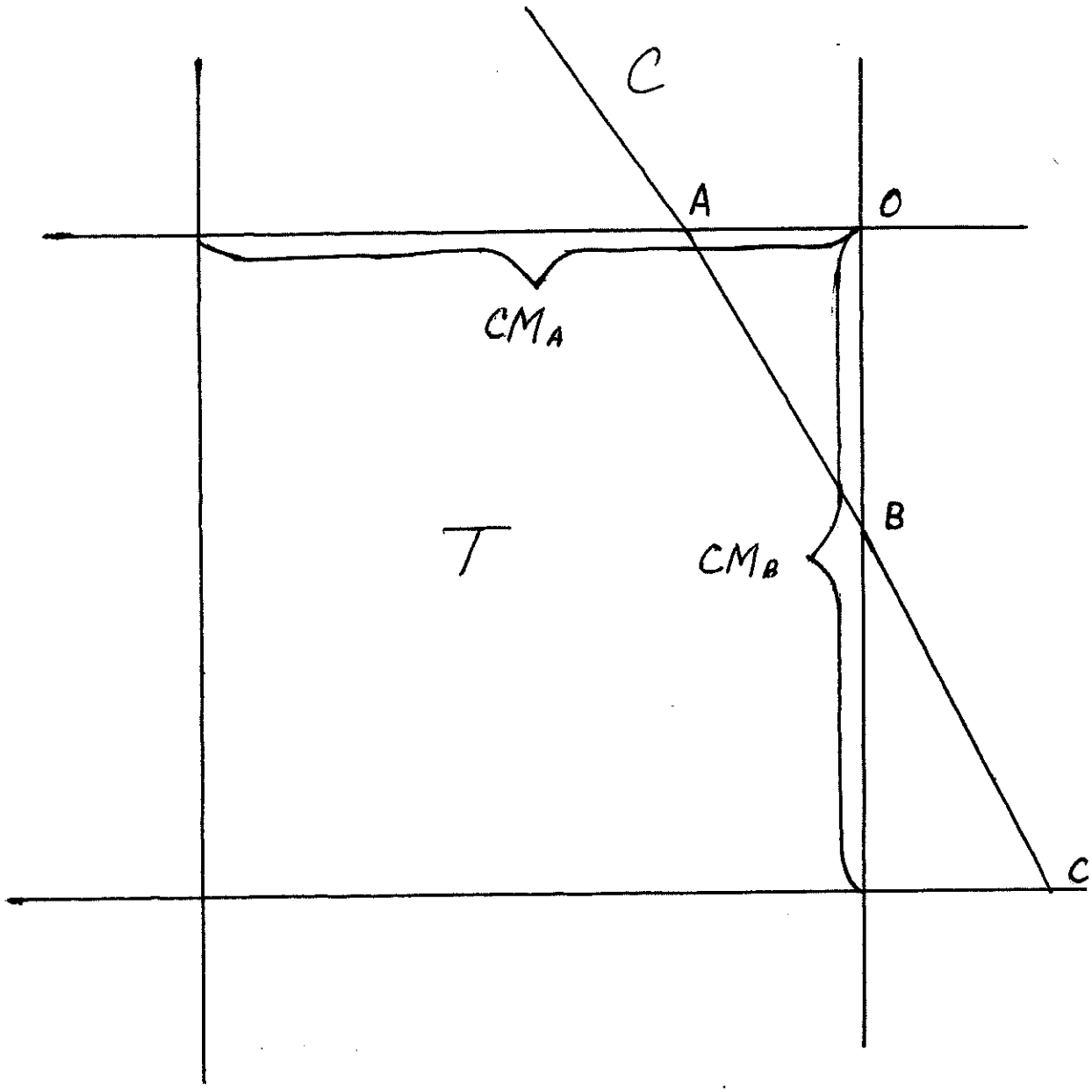
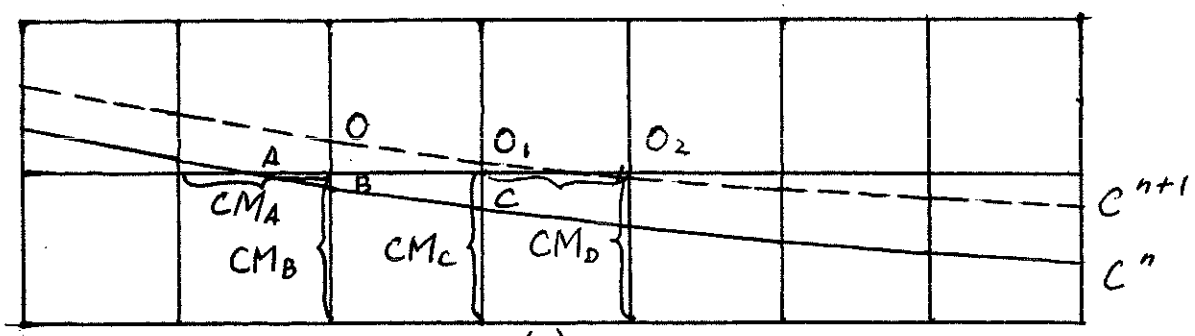


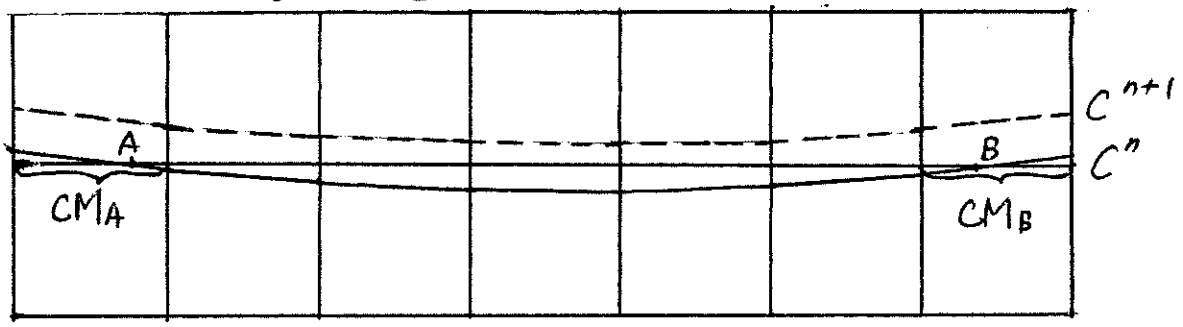
FIGURE 2.3

Discontinuity curve C intersects the grid mesh T obliquely that it cuts out a triangle OAB. CM_A and CM_B are the horizontal and vertical critical mesh intervals whose movements should match each other.



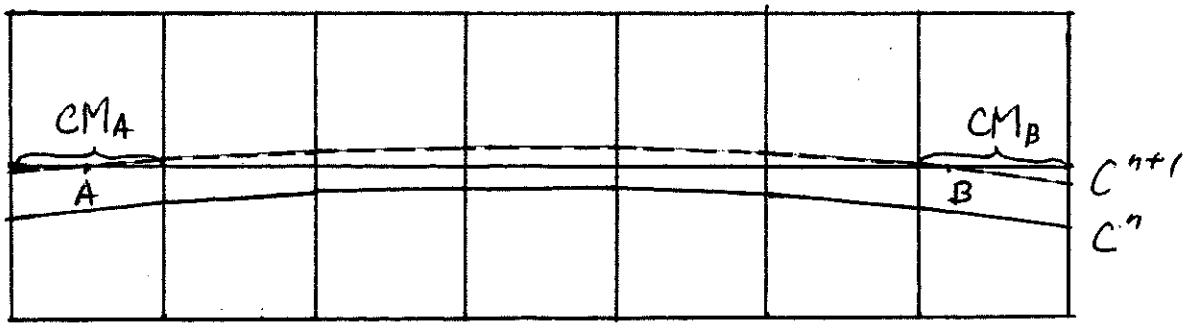
(a)

Critical mesh interval CM_A moves two mesh intervals to the right when the discontinuity curve moves from C^n to C^{n+1}. The direction of the discontinuity curve determines whether CM_A and CM_B cross O. However, the movements of CM_C and CM_D determine whether they cross O_1 and O_2.



(b)

Critical mesh intervals CM_A and CM_B are lost when the discontinuity curve moves from C^n from C^{n+1}.



(c)

Critical mesh intervals CM_A and CM_B are obtained when the discontinuity curve moves from C^n to C^{n+1}. The middle points are chosen to be their discontinuity positions.

FIGURE 2.4

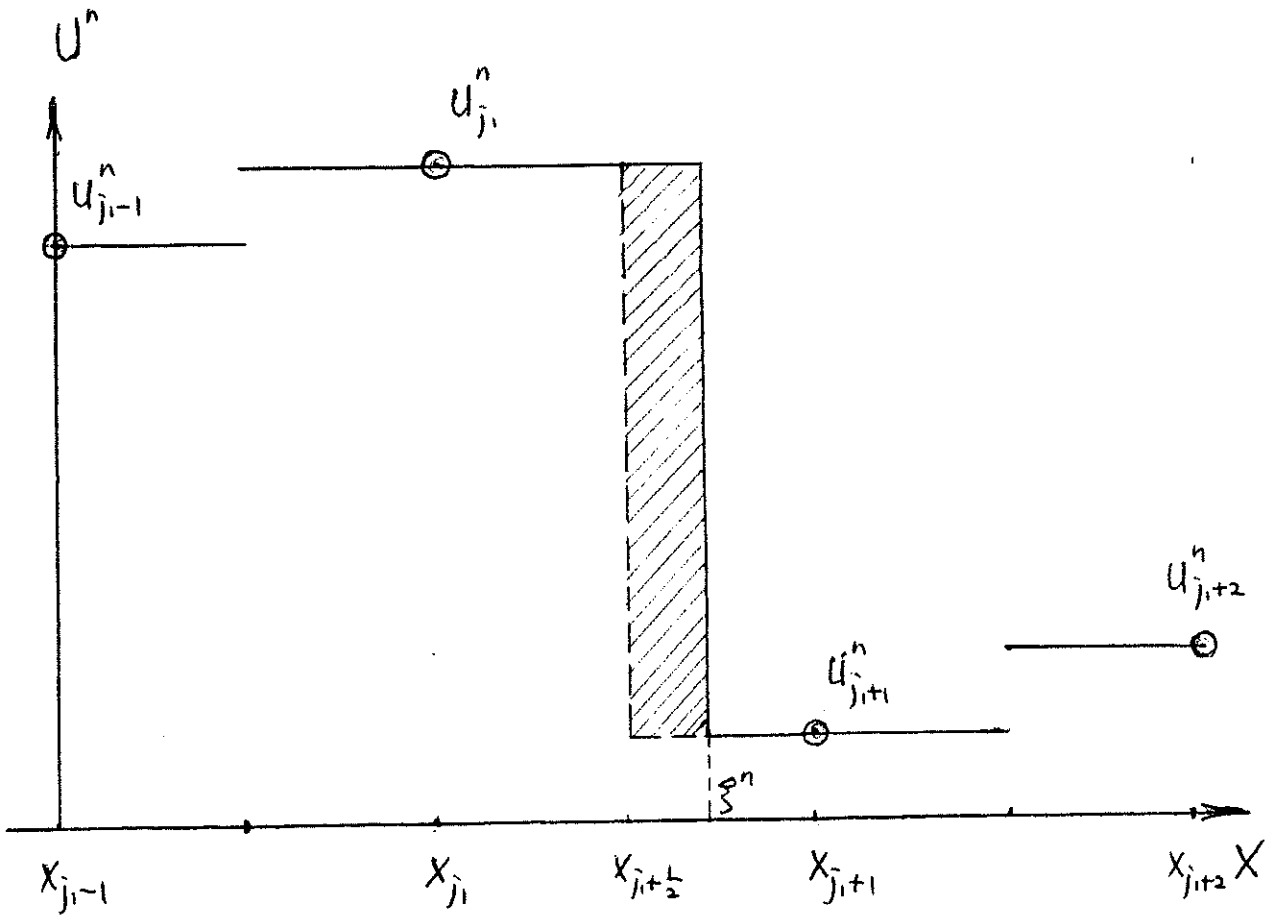


FIGURE 3.1

The dash line is $U^n(x)$ defined by (3.5) and the solid line is $\hat{U}^n(x)$ defined by (3.6). The shadow is the difference between $U^n(x)$ and $\hat{U}^n(x)$, which is just the principle part of (3.2) multiplied by h.

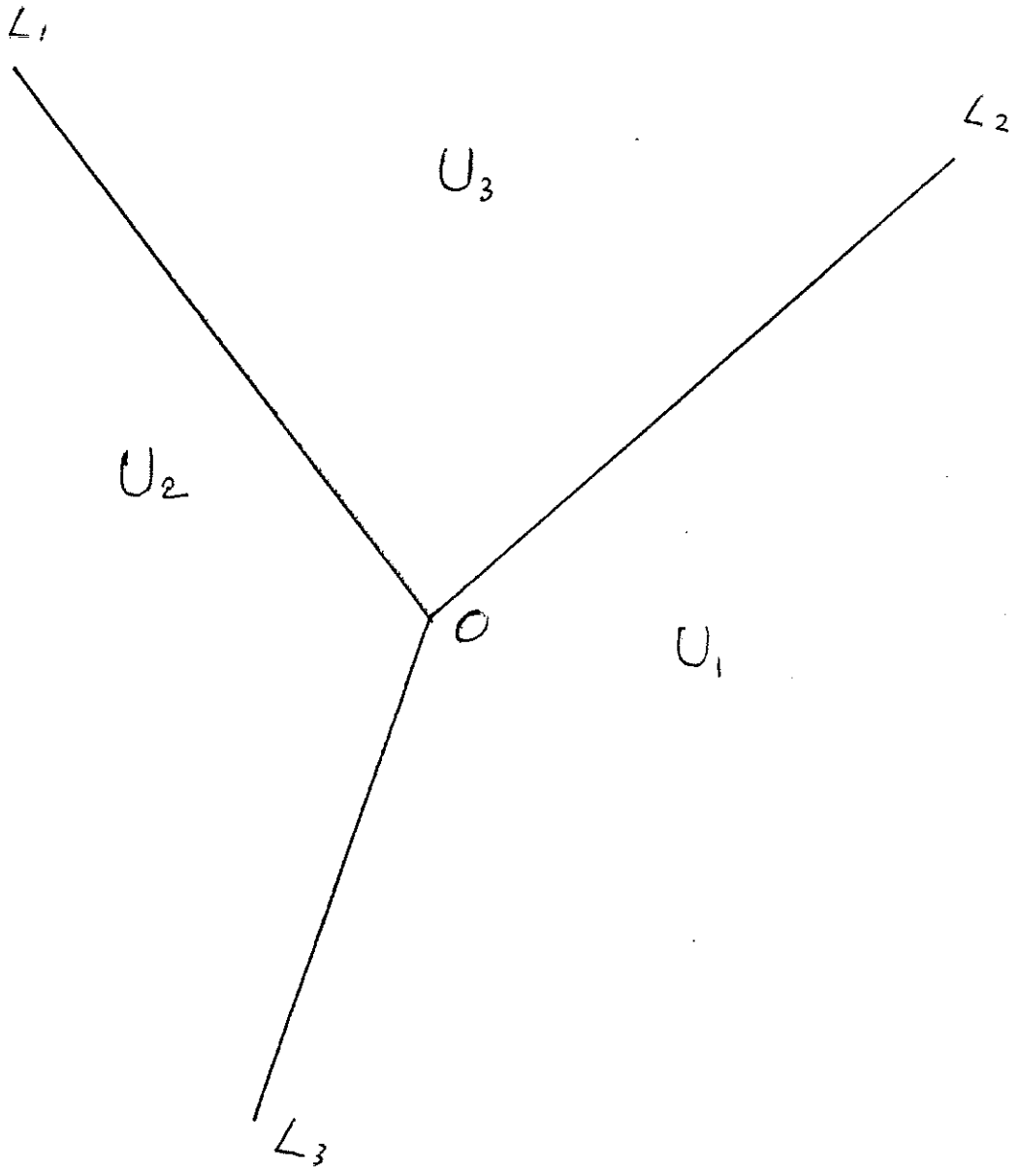


FIGURE 4.1

Two straight line discontinuities L_1 and L_2 meet at a point O, connecting three constant states U_2, U_3 and U_1, a third straight line discontinuity L_3 forms from O connecting U_2 and U_1.

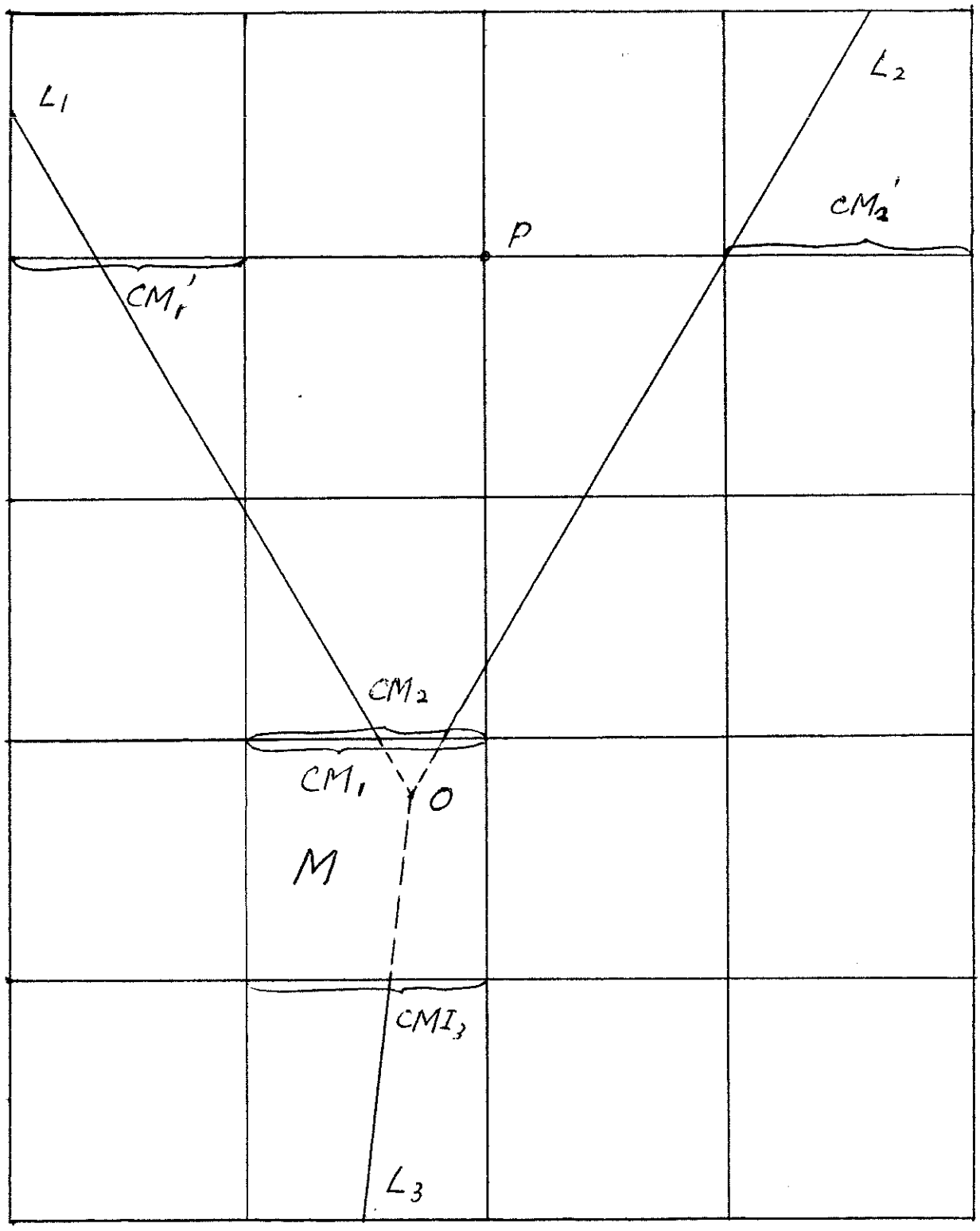


FIGURE 4.2

A node mesh M has three critical mesh intervals CM_1, CM_2 and CM_3, two of them are overlapped. In the vicinity of M, two critical mesh intervals CM_1' and CM_2' belonging to L_1 and L_2 respectively are close to each other so that stencils of the numerical fluxes near point P might cover both of them.

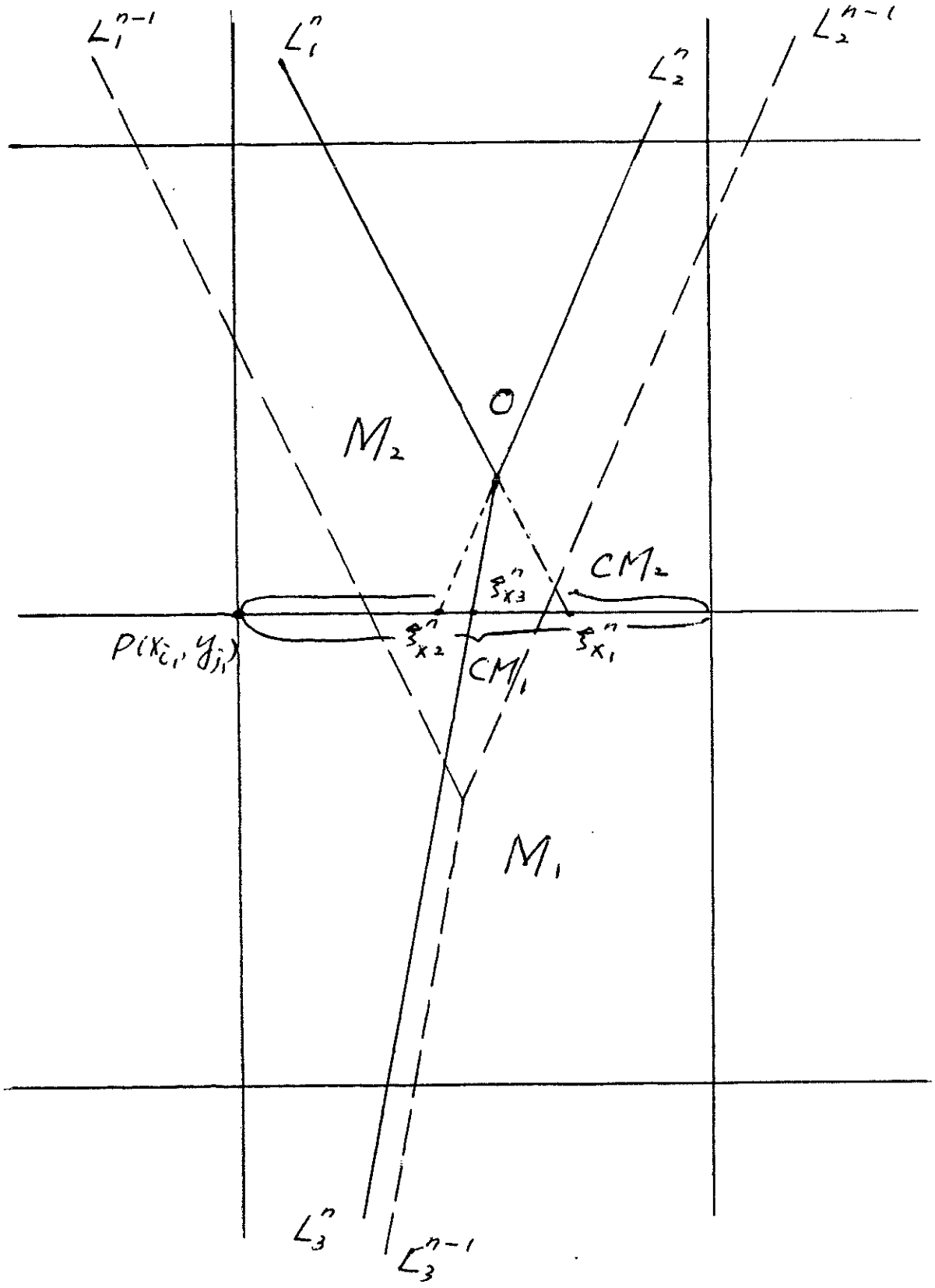


FIGURE 4.3-a

Two horizontal critical mesh intervals CM_1 and CM_2 of L_1 and L_2 merge to generate a new critical mesh interval for L_3, meanwhile the node mesh moves from M_1 to M_2.

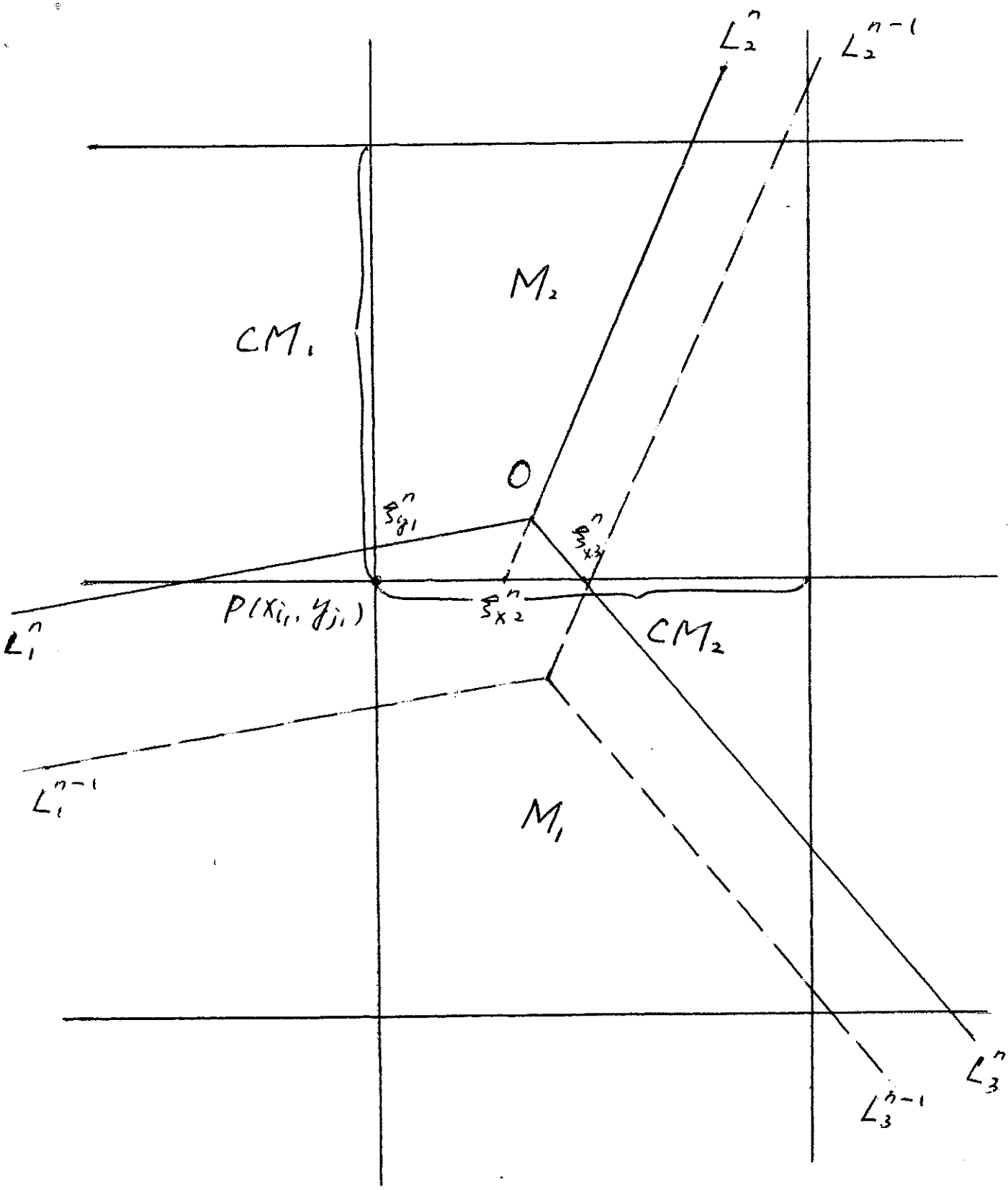
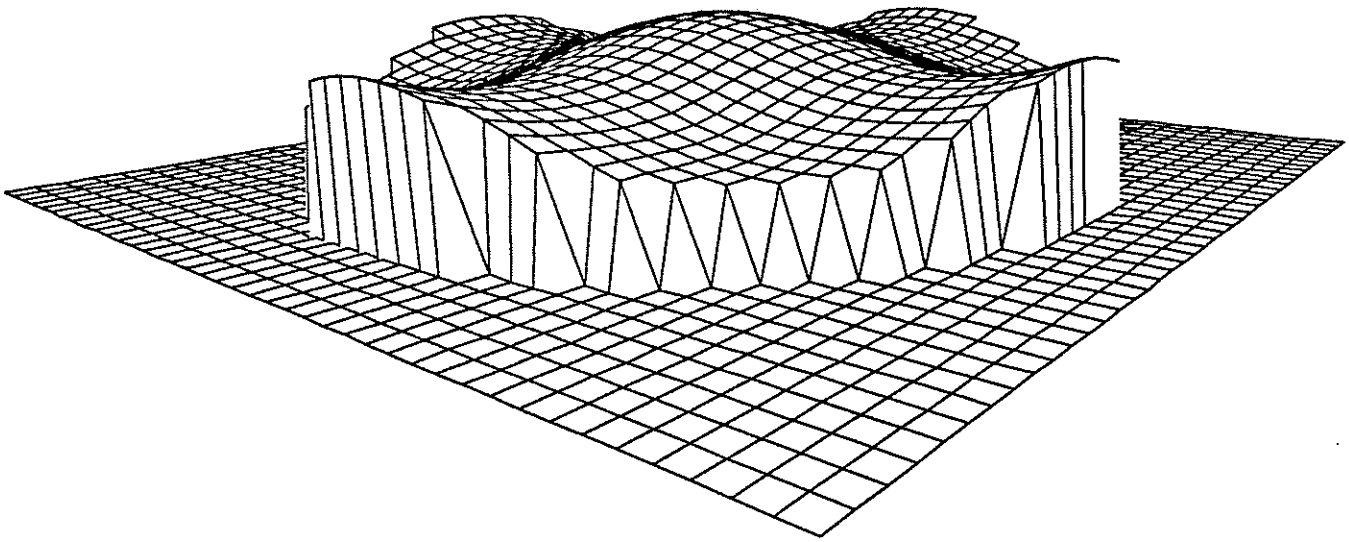


FIGURE 4.3-b

Horizontal critical mesh interval CM_1 of L_1 moves one mesh interval to the top so that L_2 loses critical mesh interval CM_2, which becomes a critical mesh interval of L_3, meanwhile the node mesh moves from M_1 to M_2.



surface, t=2. (160 steps)

FIGURE 5.1-a

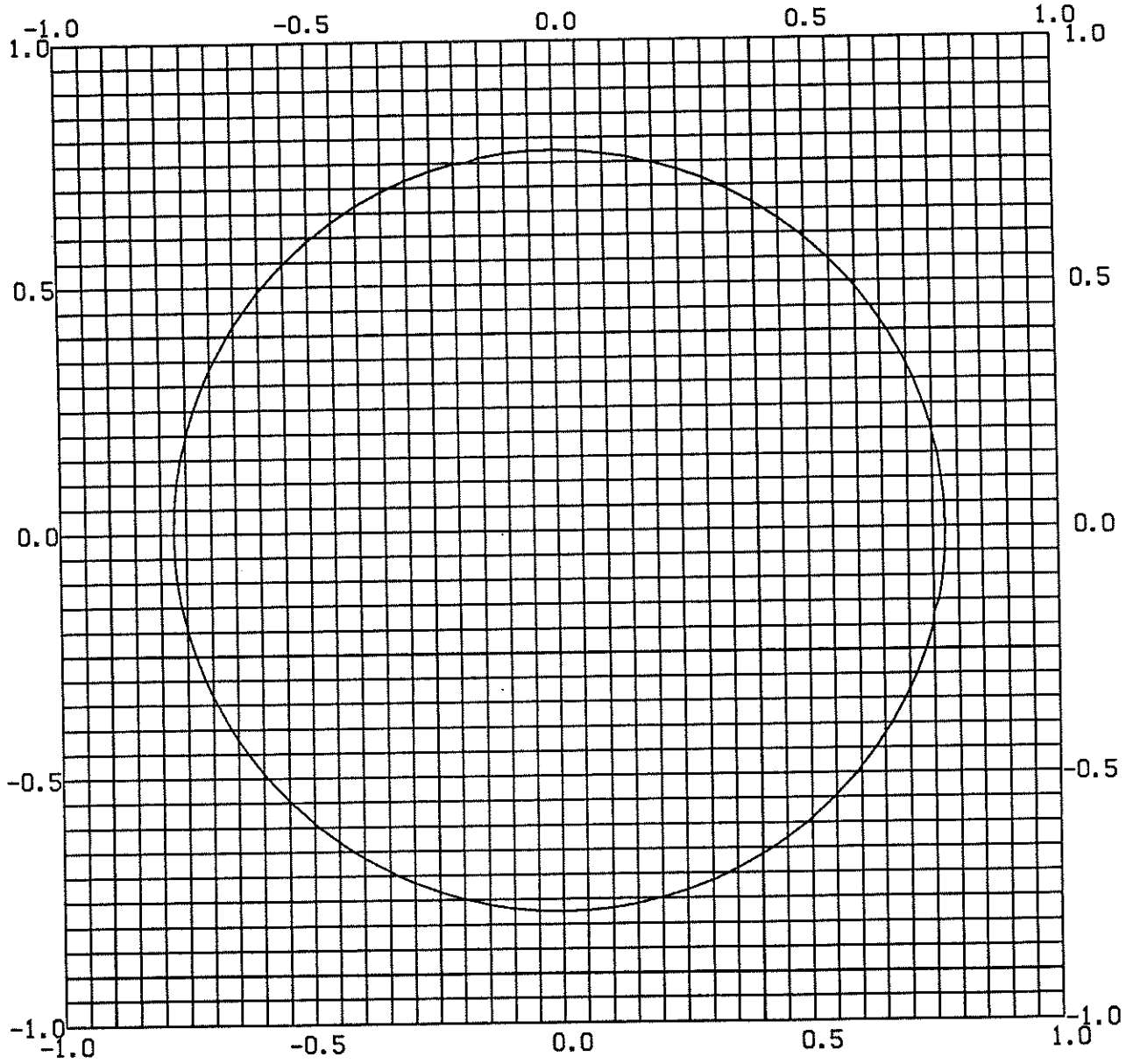

-1.0
-0.5
0.0
0.5
1.0
1.0
0.5
0.0
-0.5
-1.0
location of discontinuity, t=2. (160 steps)

FIGURE 5.1-b

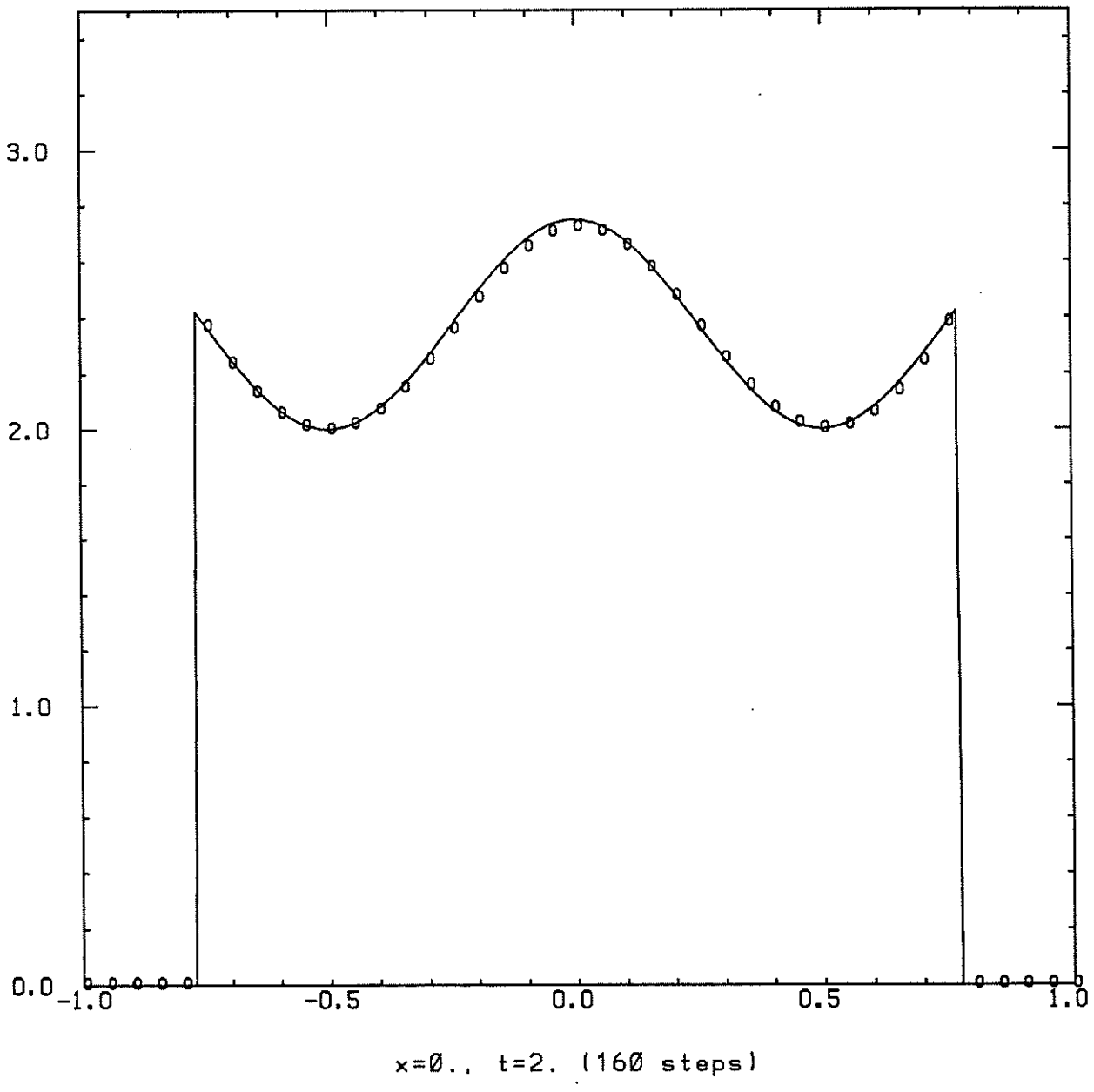


FIGURE 5.1-c

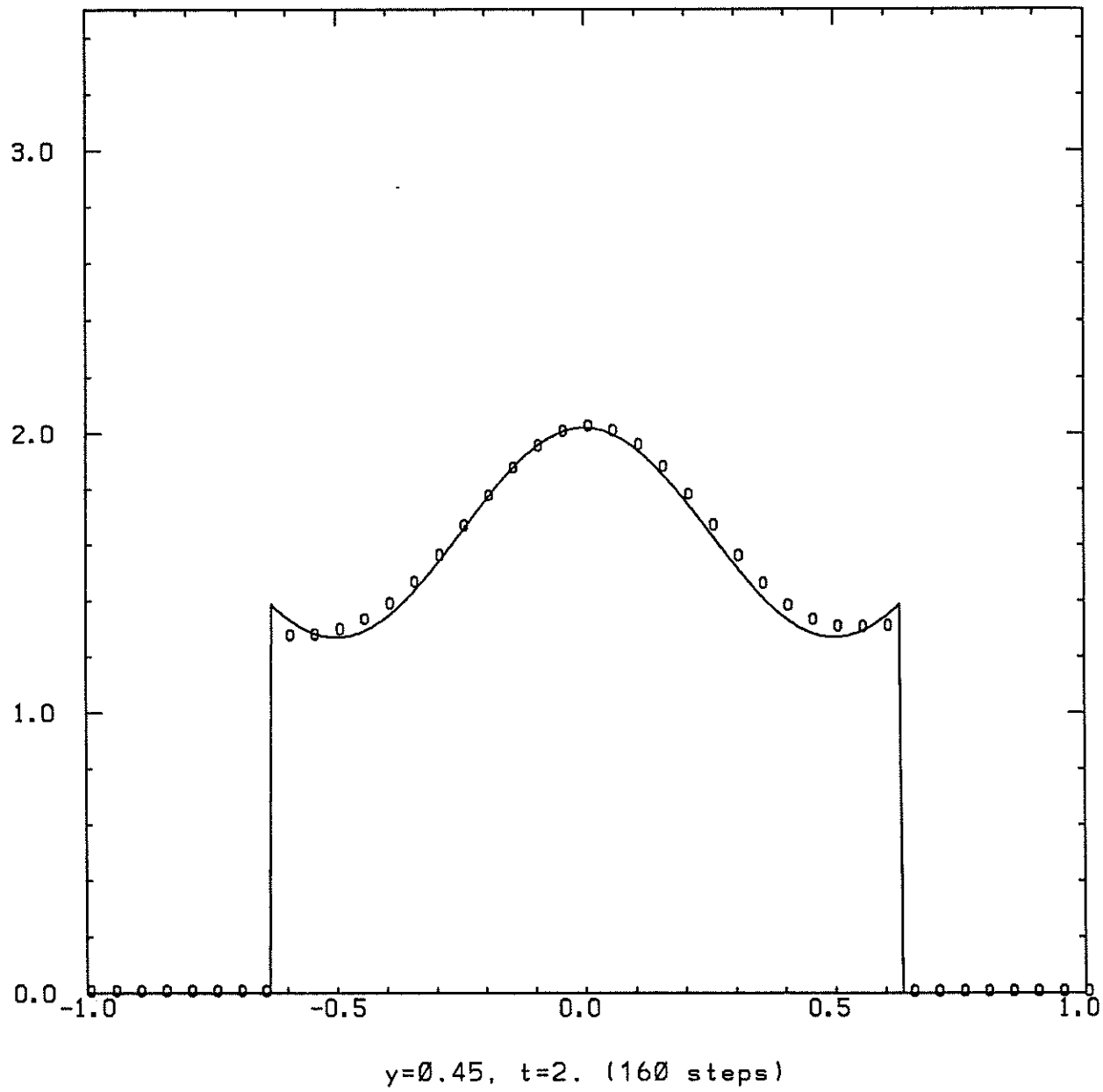
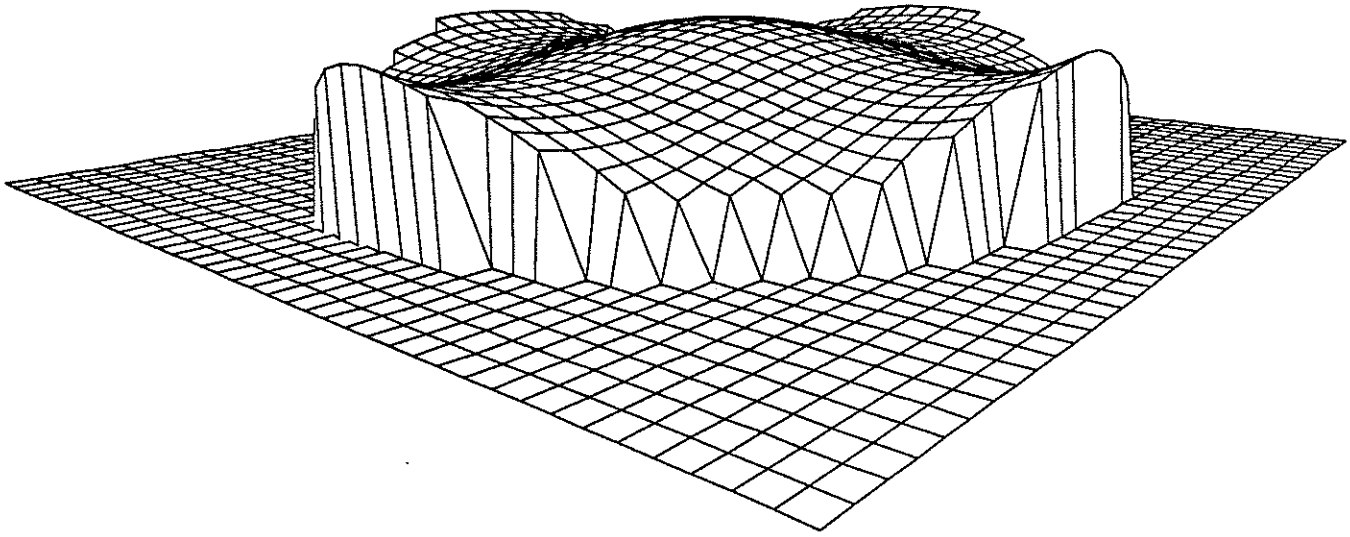


FIGURE 5.1-d



surface, t=16. (1280 steps)

FIGURE 5.2-a

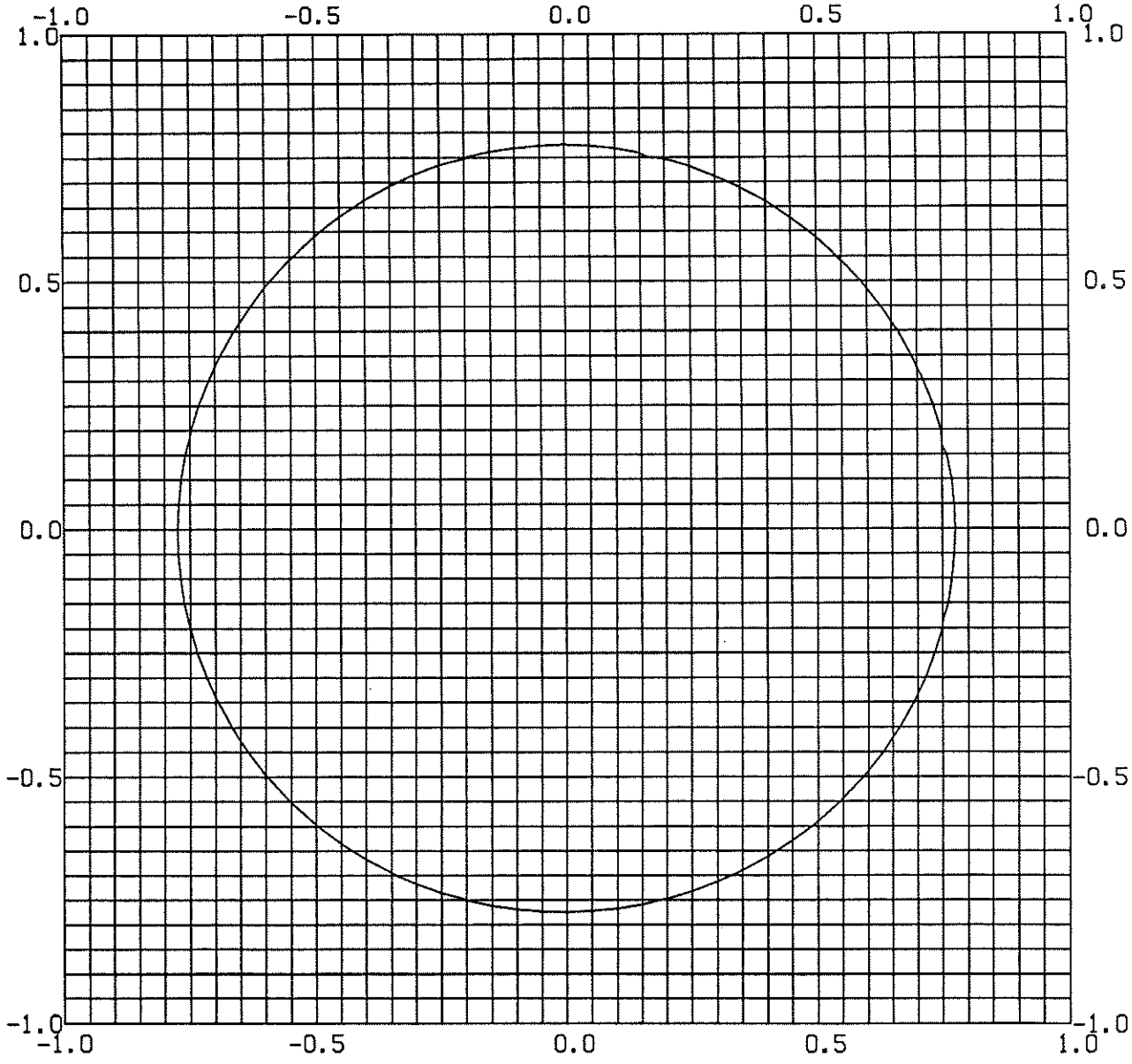

-1.0
-0.5
0.0
0.5
1.0
1.0
0.5
0.0
-0.5
-1.0
location of discontinuity, t=16. (1280 steps)

FIGURE 5.2-b

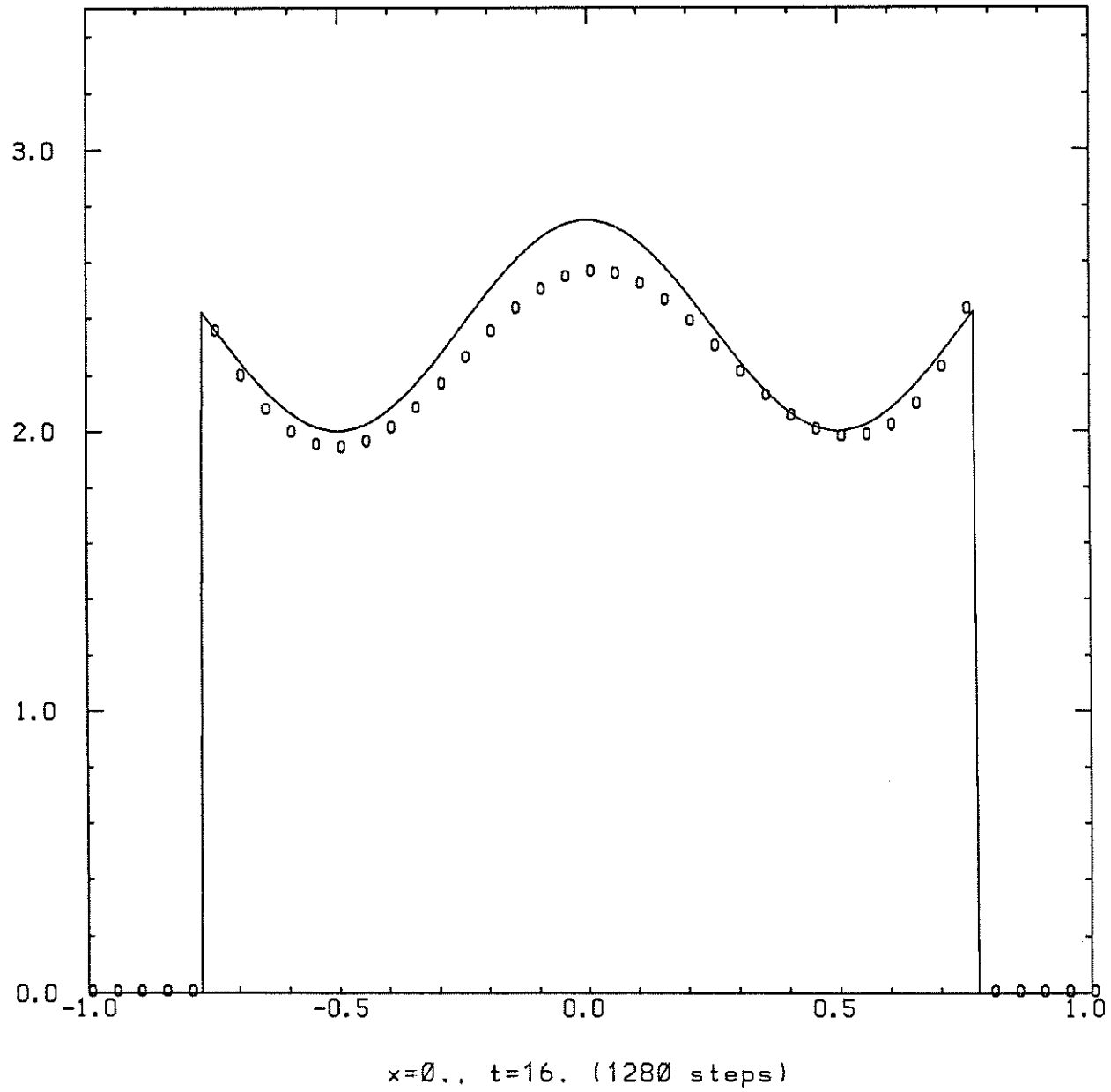


FIGURE 5.2-c

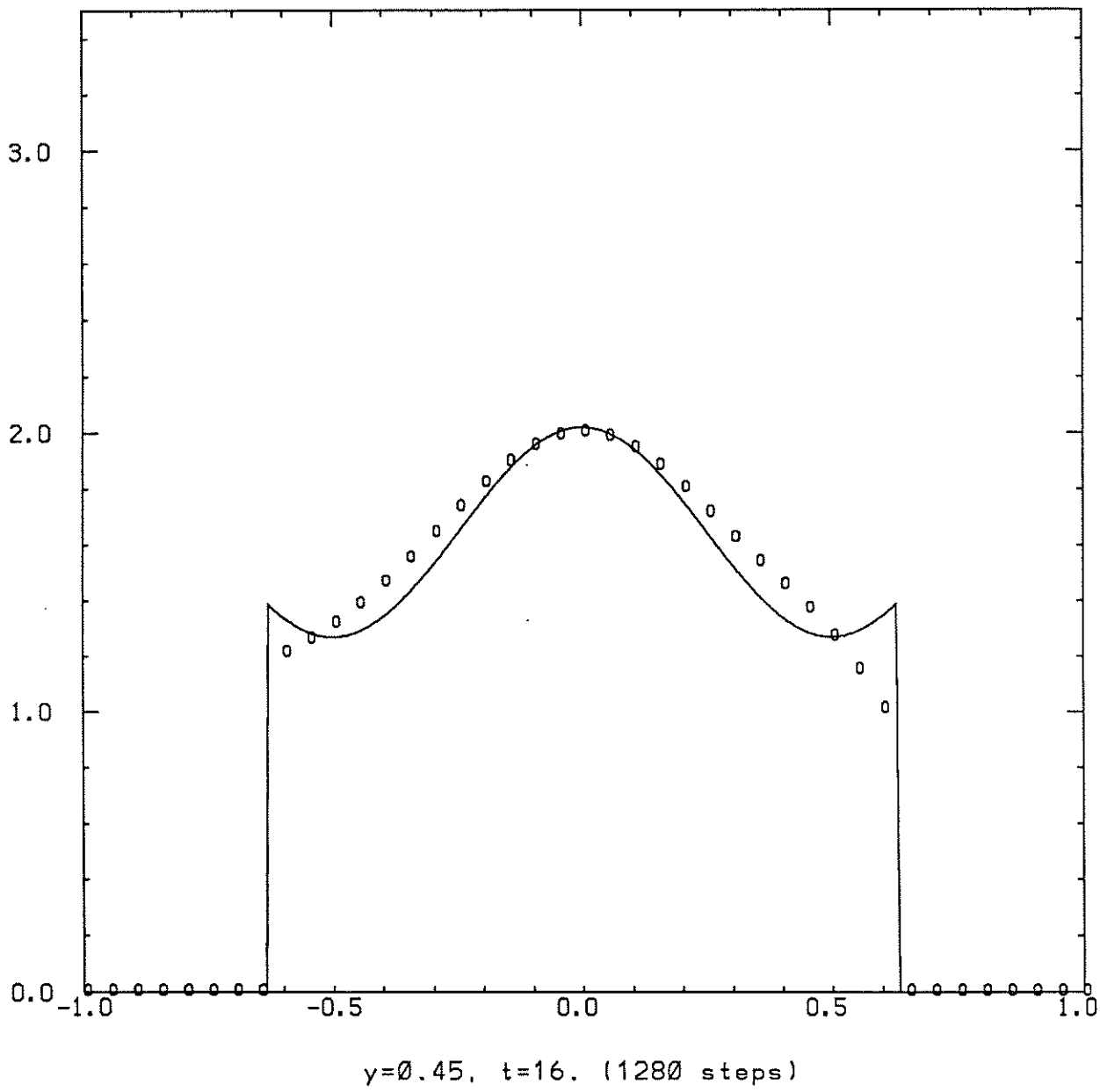
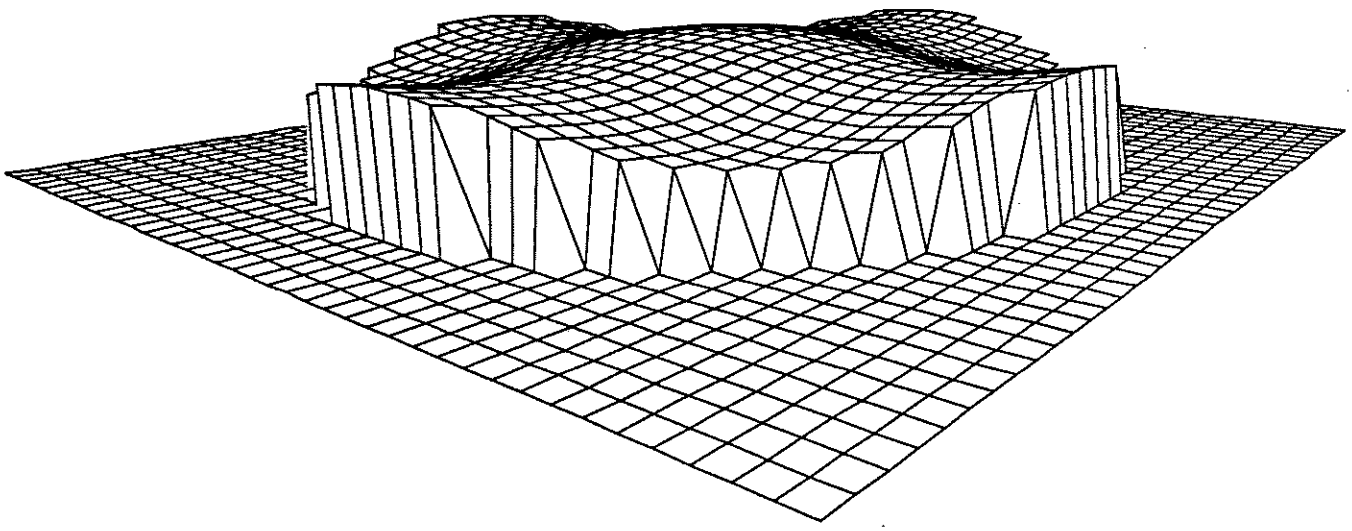


FIGURE 5.2-d



surface, t=1. (μ = 0.15)

FIGURE 5.3-a

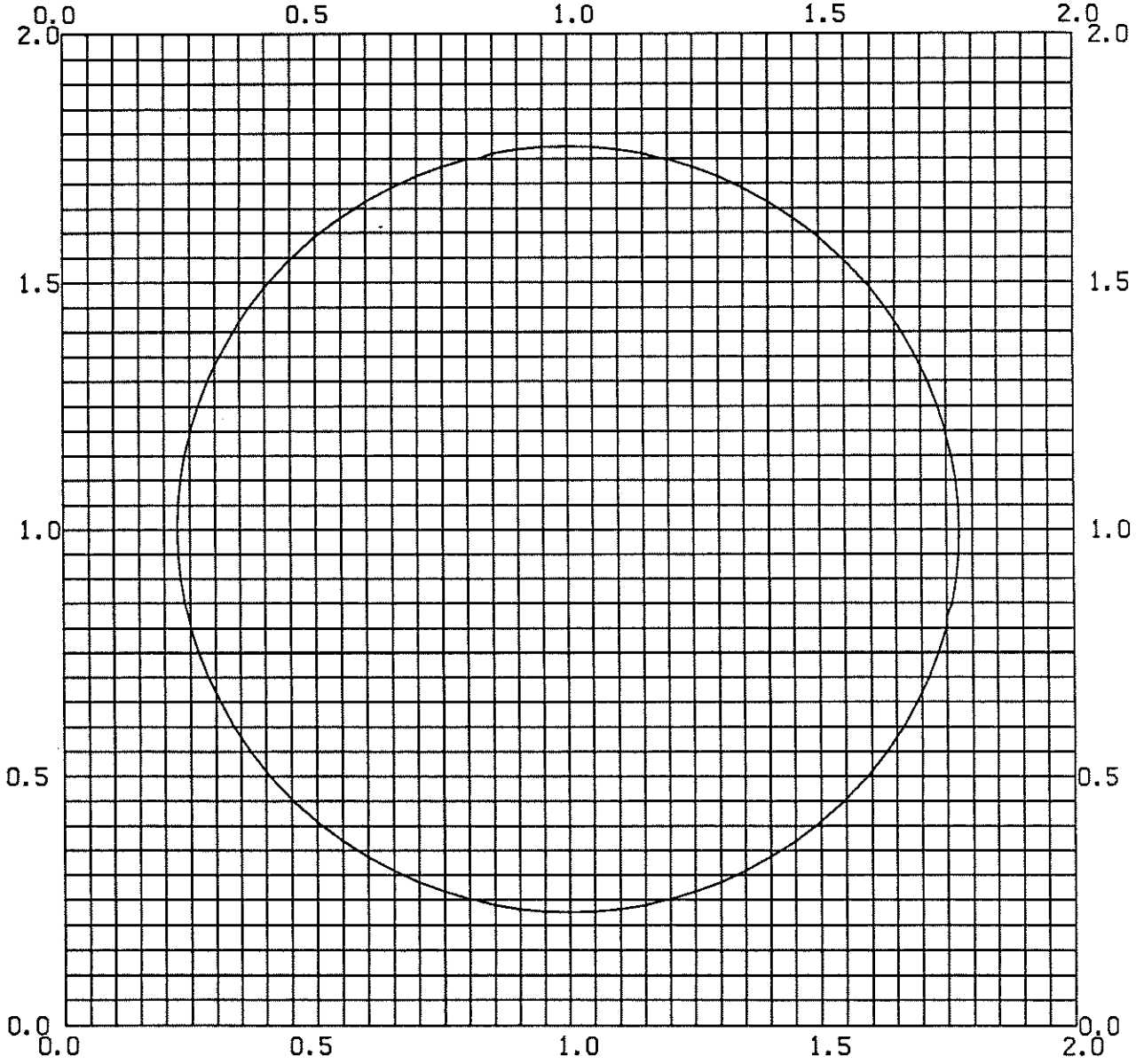
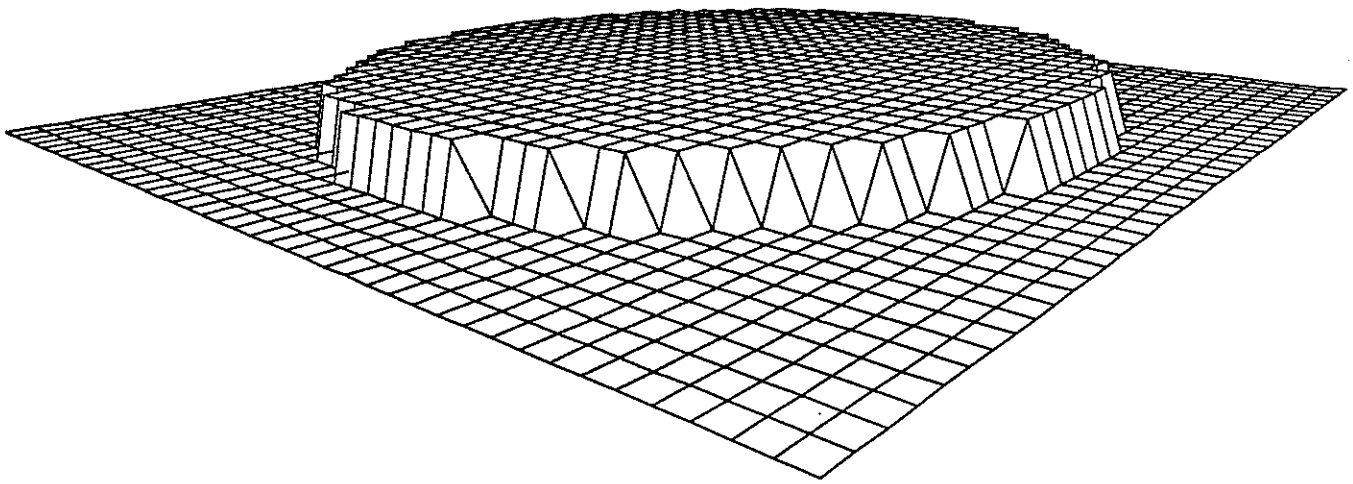


FIGURE 5.3-b



surface, t=1. (μ = 150.)

FIGURE 5.4-a

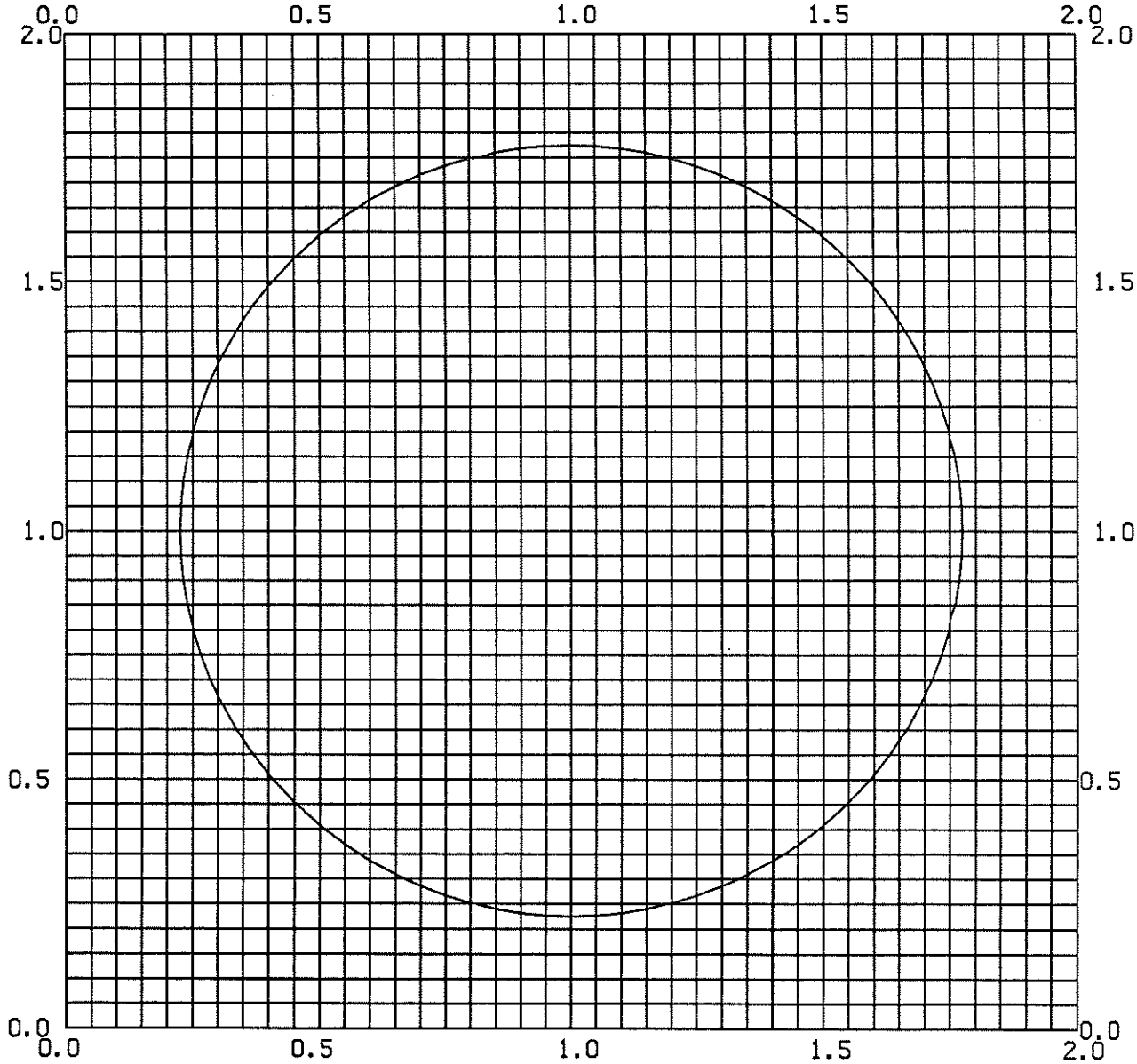
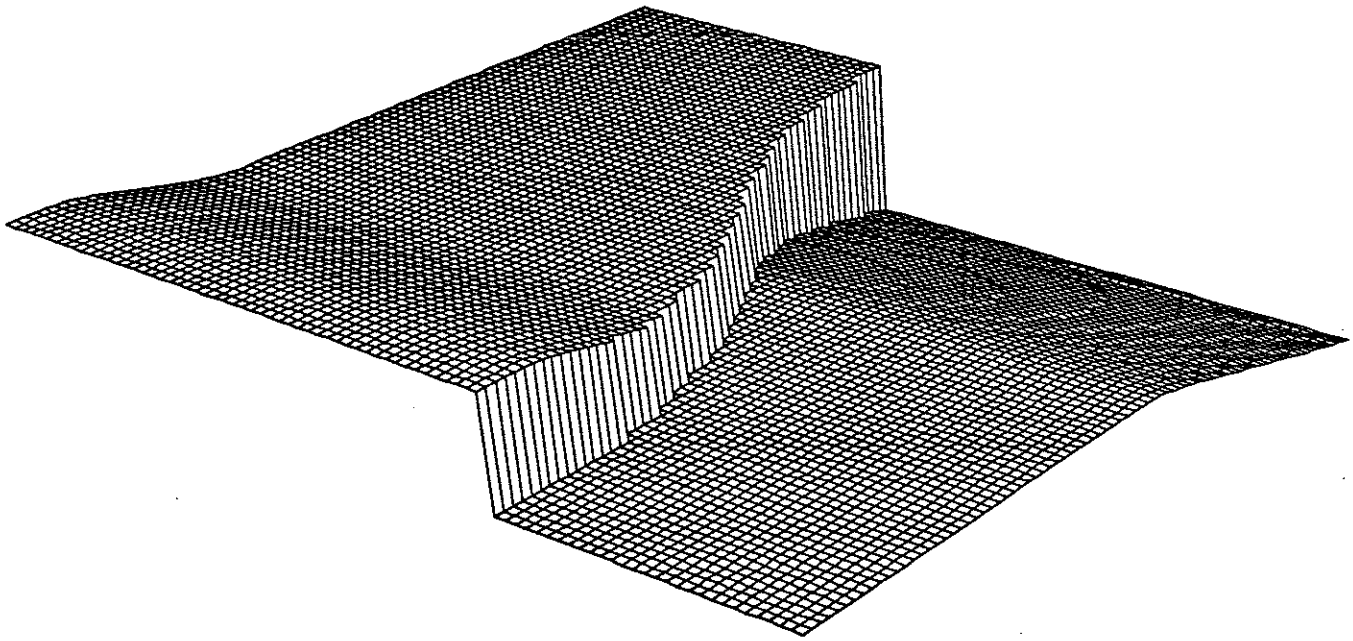


FIGURE 5.4-b



surface, t=1. (160 steps)

FIGURE 5.5-a

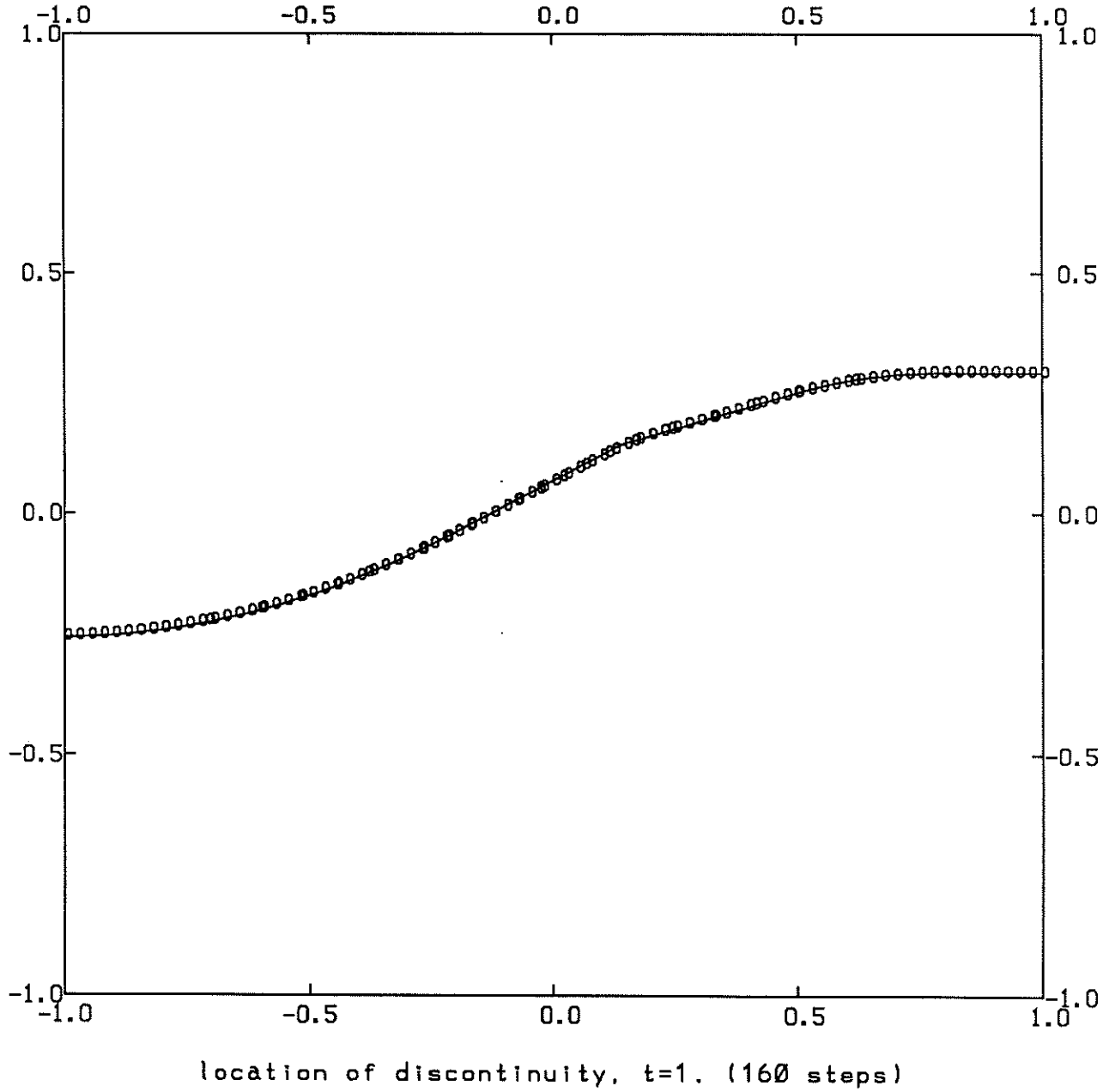

-1.0
-0.5
0.0
0.5
1.0
1.0
0.5
0.0
-0.5
-1.0
location of discontinuity, t=1. (160 steps)

FIGURE 5.5-b

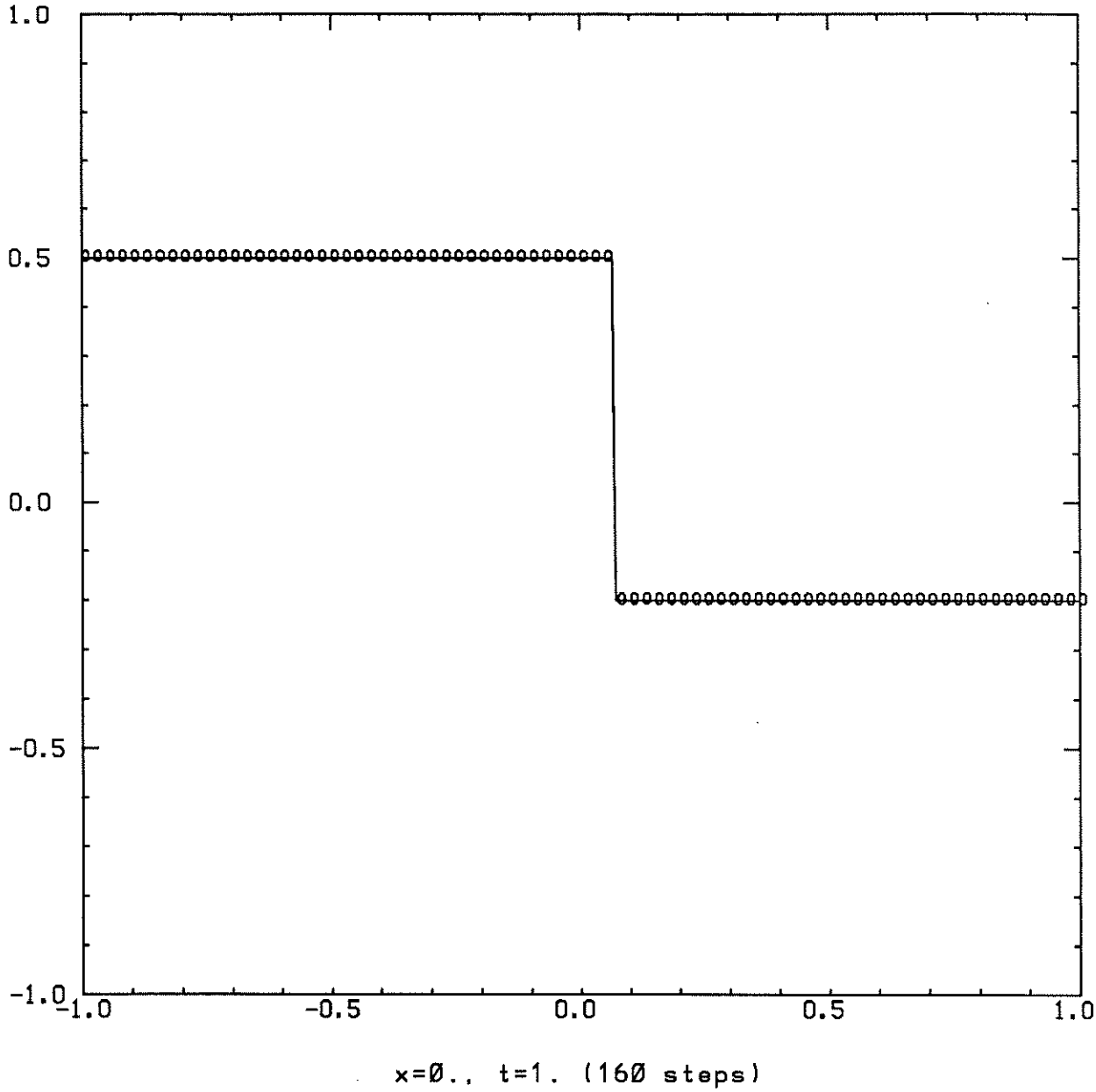


FIGURE 5.5-c

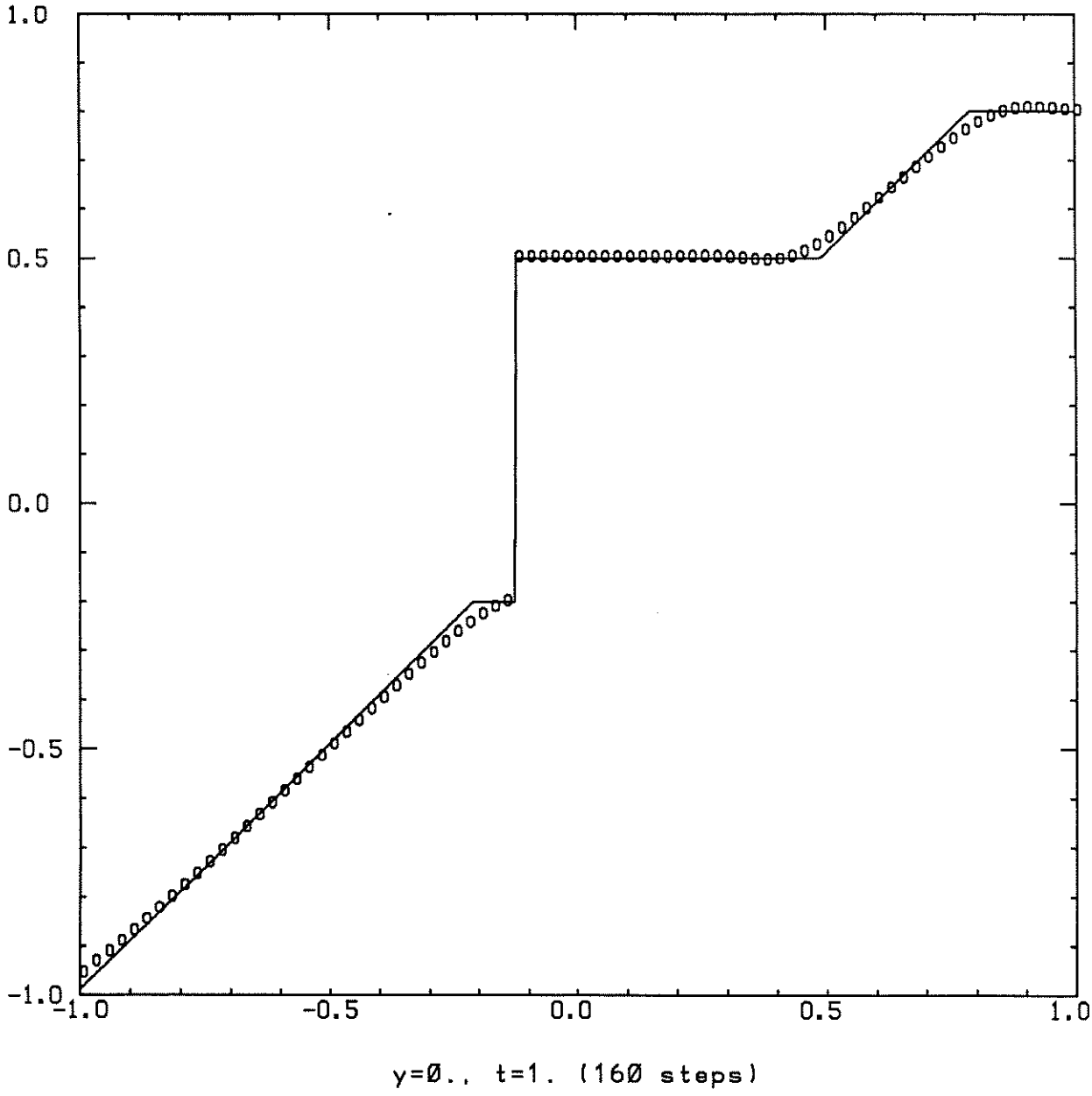
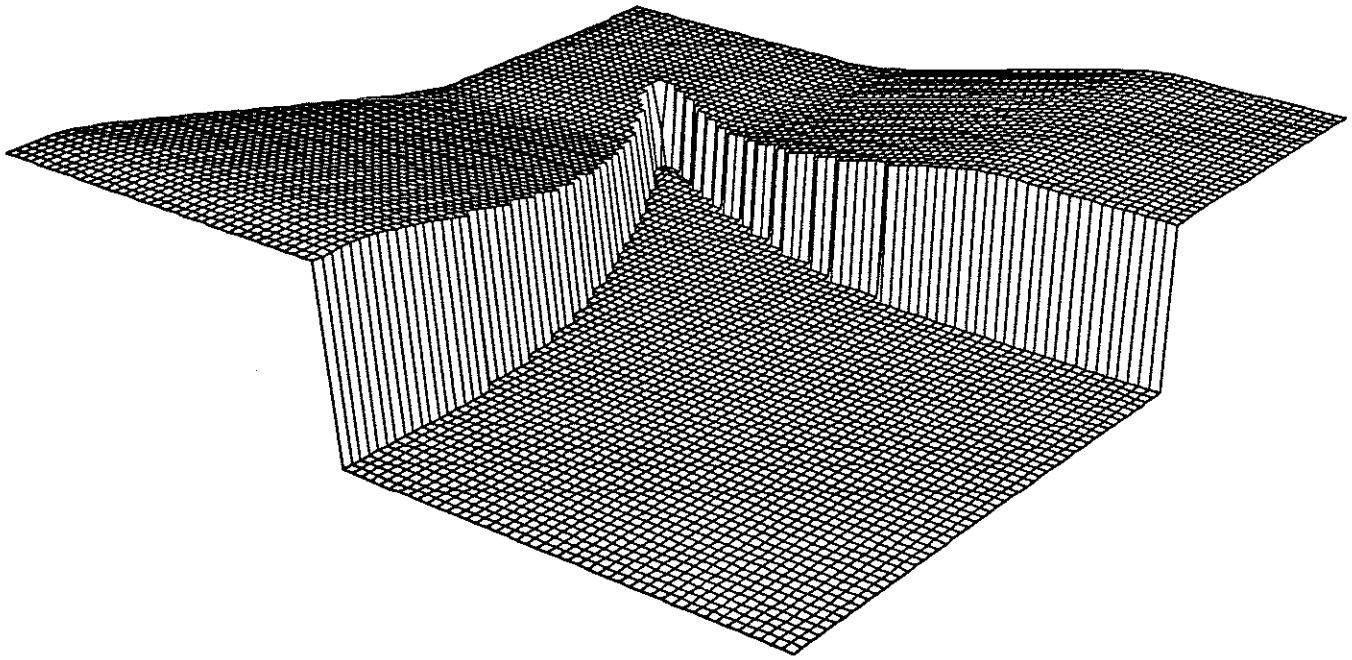


FIGURE 5.5-d



surface, t=1. (160 steps)

FIGURE 5.6-a

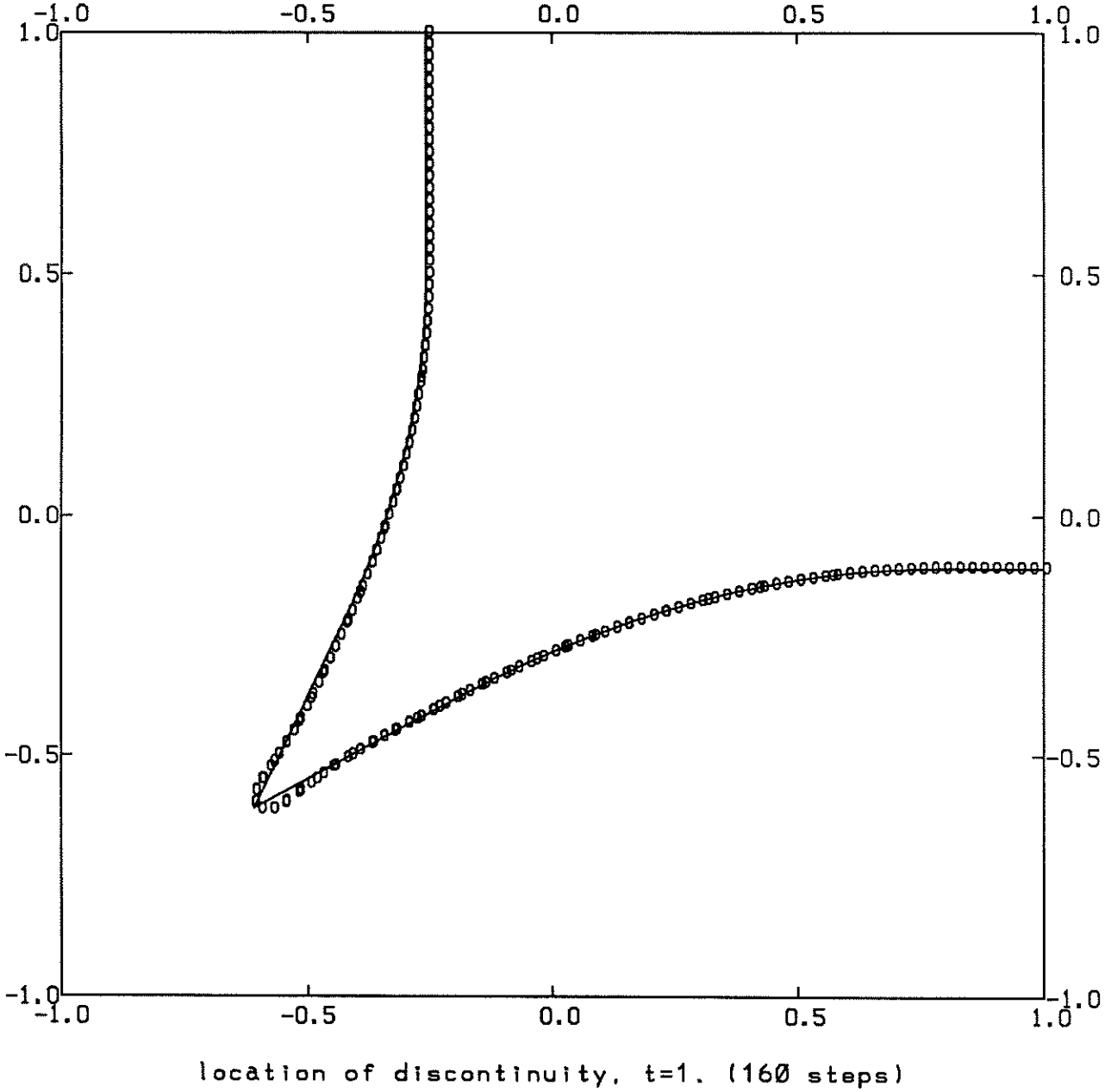

-1.0
-0.5
0.0
0.5
1.0
1.0
0.5
0.0
-0.5
-1.0
location of discontinuity, t=1. (160 steps)

FIGURE 5.6-b

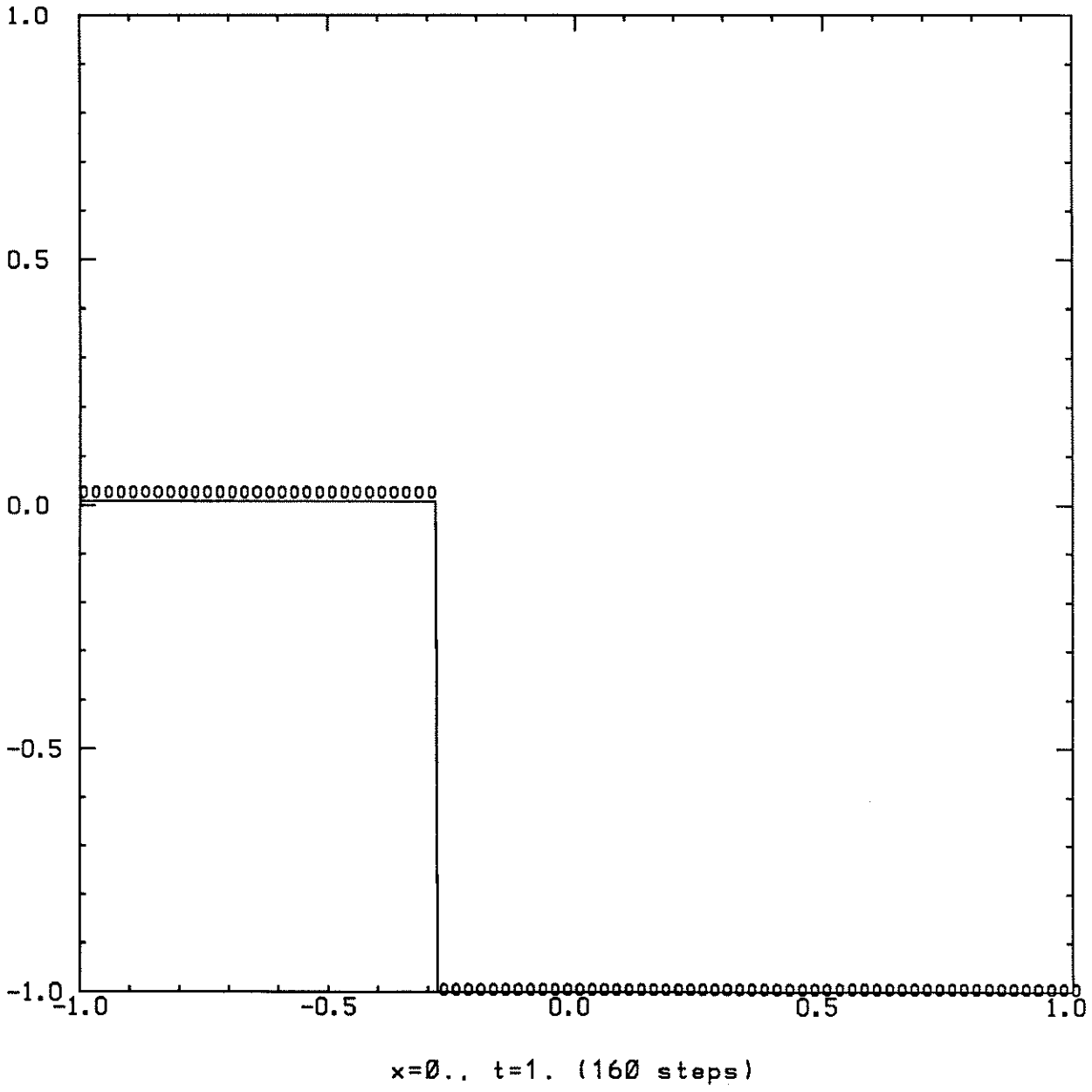


FIGURE 5.6-c

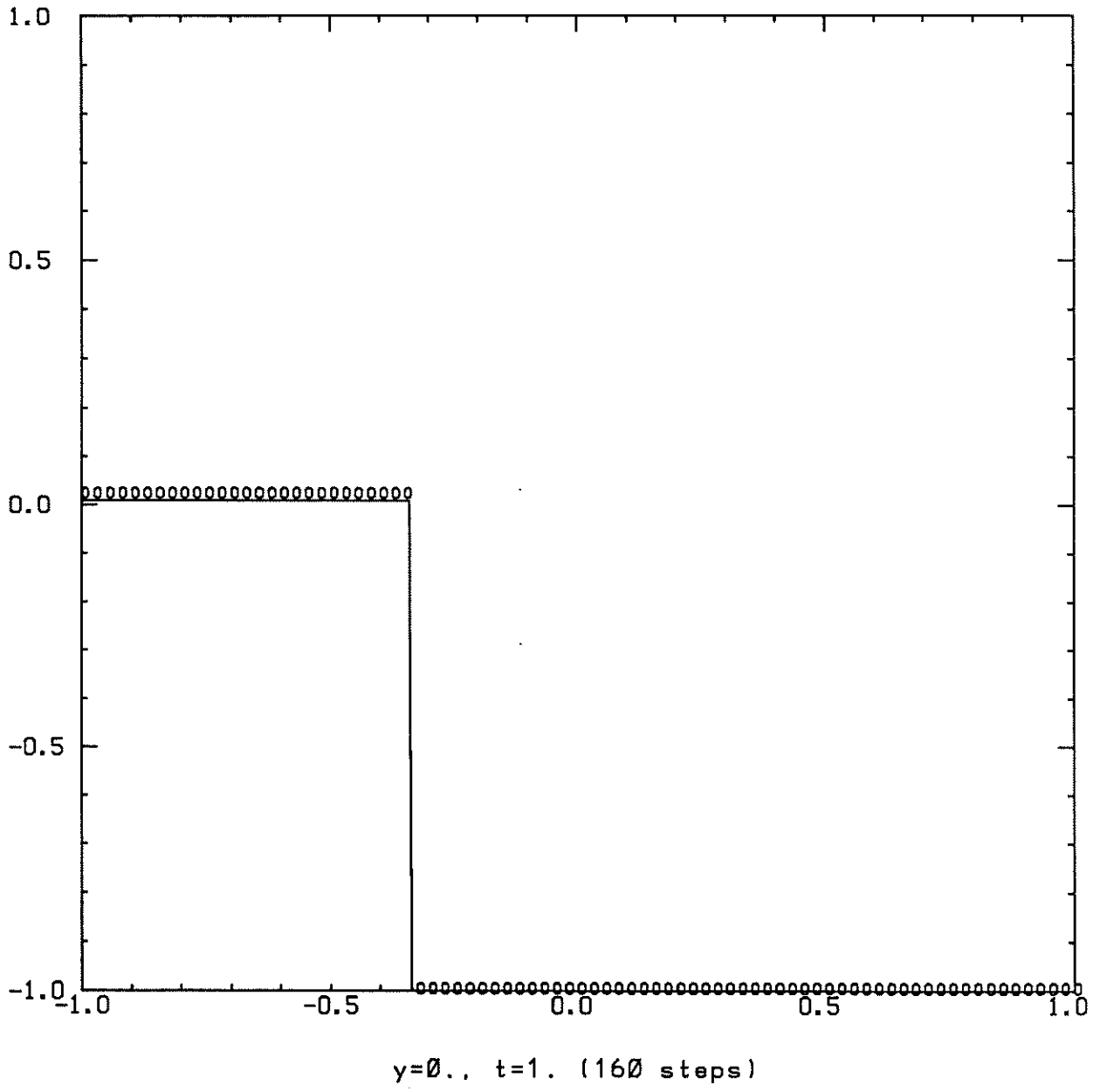
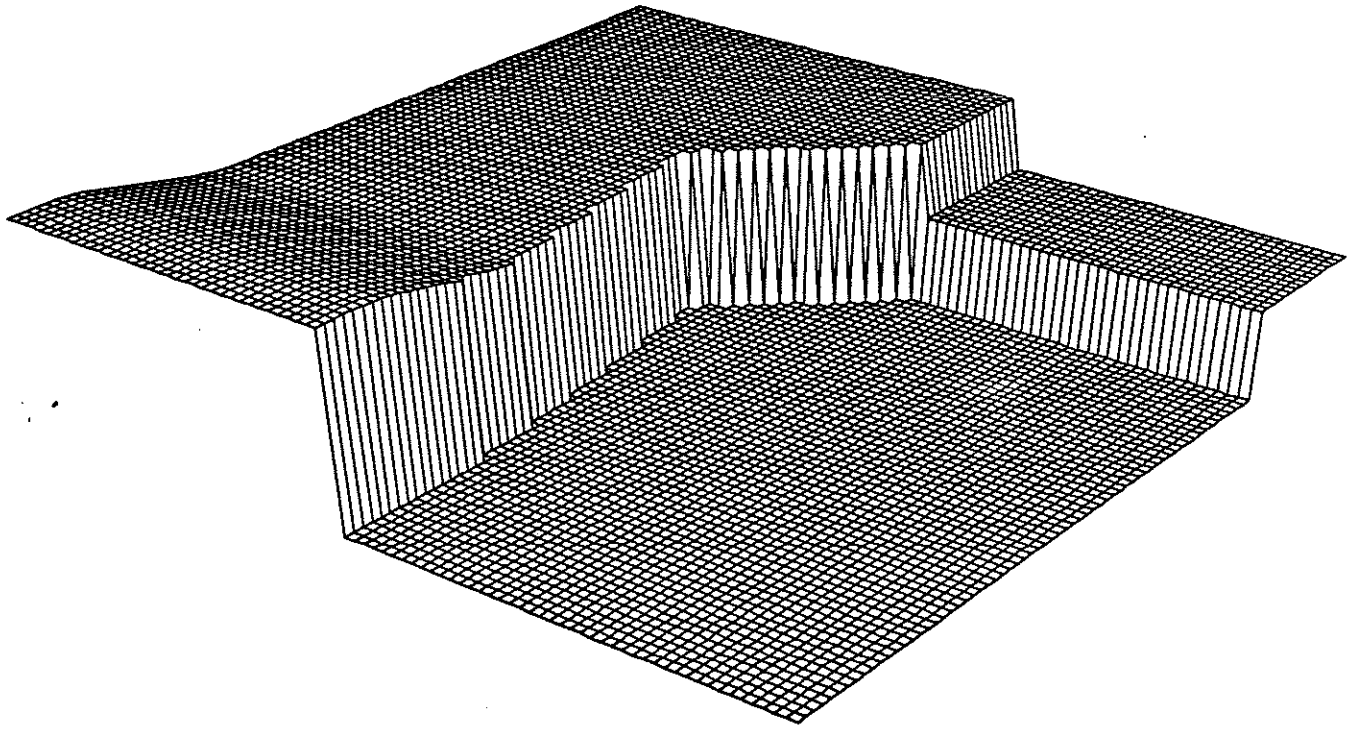


FIGURE 5.6-d



surface, t=1. (160 steps)

FIGURE 5.7-a

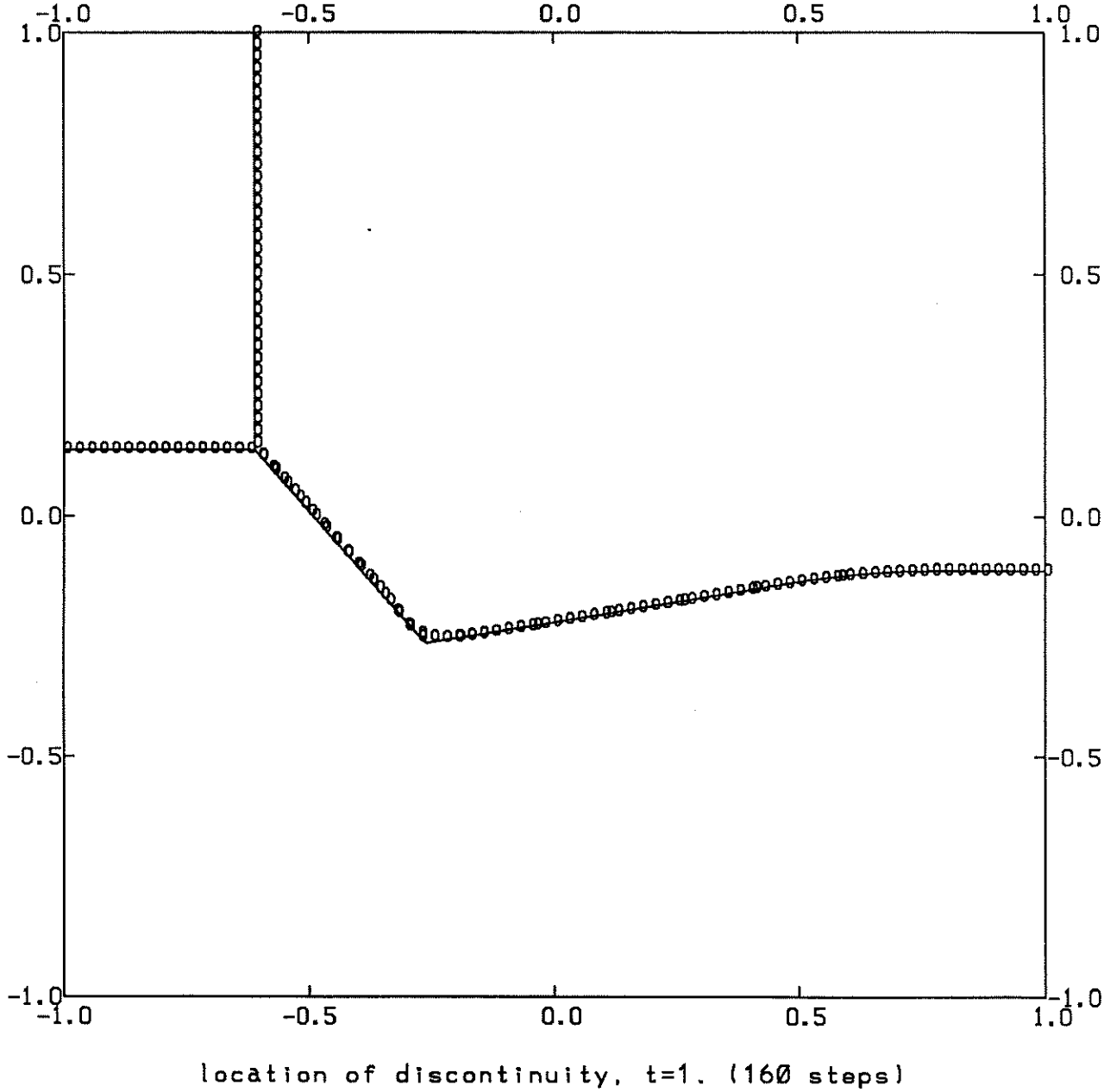

-1.0
-0.5
0.0
0.5
1.0
1.0
0.5
0.0
-0.5
-1.0
location of discontinuity, t=1. (160 steps)

FIGURE 5.7-b

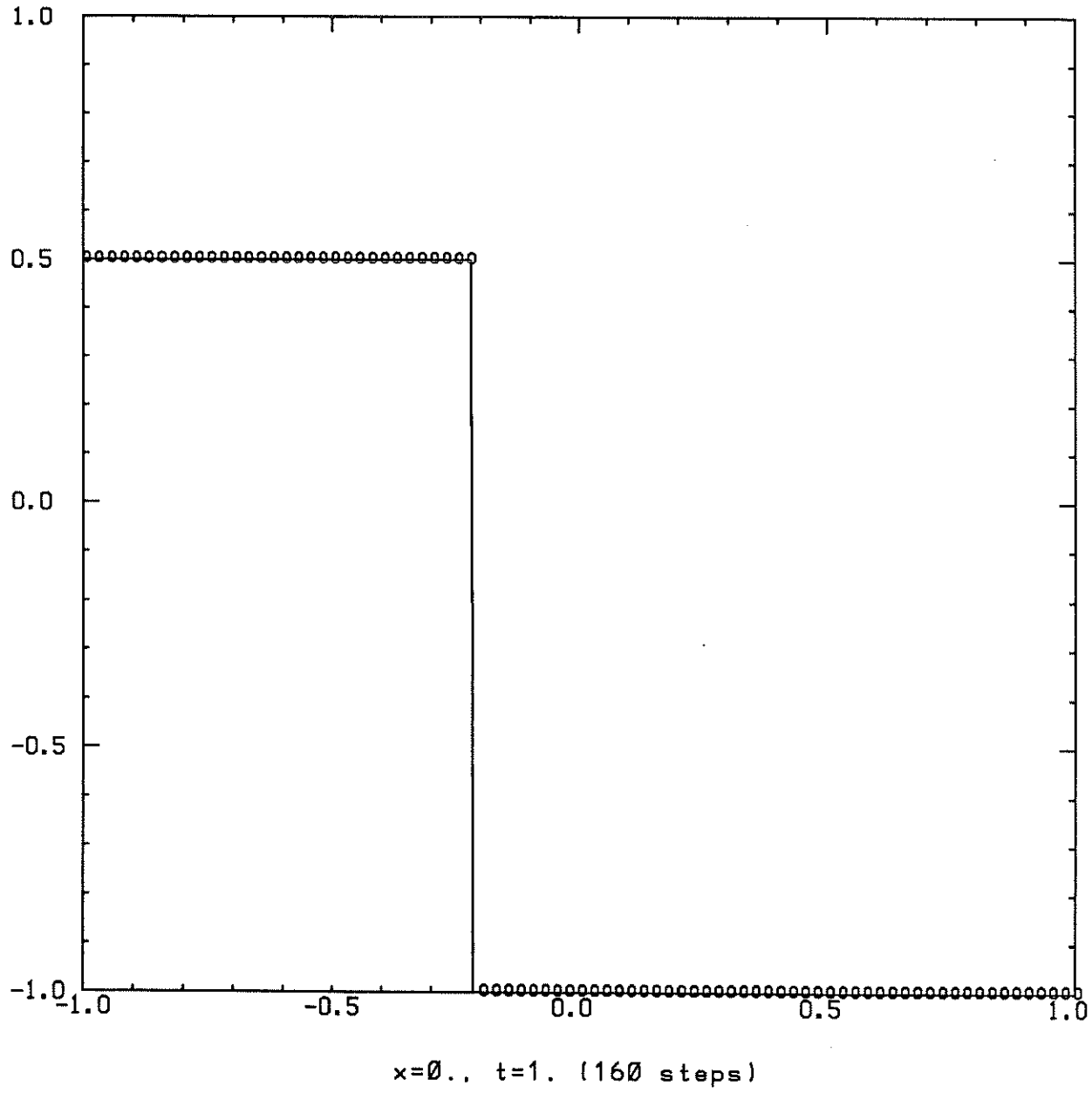


FIGURE 5.7-c

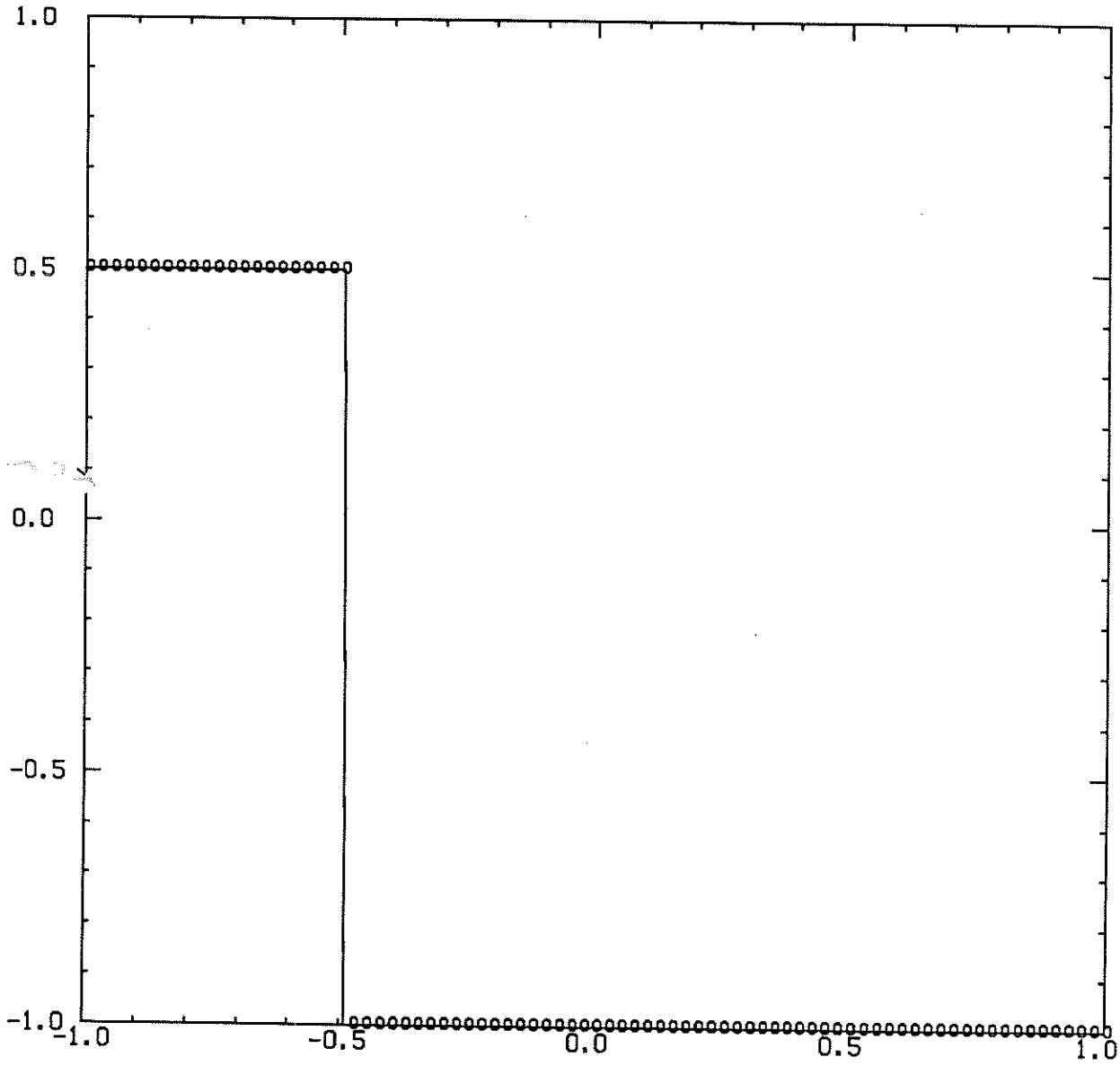


FIGURE 5.7-d