ESAIM: M2AN
M2AN, Vol. 38, N° 3, 2004, pp. 519–540
DOI: 10.1051/m2an:2004027

ESAIM: Mathematical Modelling and Numerical Analysis

THE CORRECT USE OF THE LAX–FRIEDRICHS METHOD*

Michael Breuss[1]

Abstract. We are concerned with the structure of the operator corresponding to the Lax–Friedrichs method. At first, the phenomenae which may arise by the naive use of the Lax–Friedrichs scheme are analyzed. In particular, it turns out that the correct definition of the method has to include the details of the discretization of the initial condition and the computational domain. Based on the results of the discussion, we give a recipe that ensures that the number of extrema within the discretized version of the initial data cannot increase by the application of the scheme. The usefulness of the recipe is confirmed by numerical tests.

Mathematics Subject Classification. 35L65, 65M06, 65M12.

Received: October 8, 2003.

Introduction

We are concerned with the operator corresponding to the explicit Lax–Friedrichs method for the approximation of scalar conservation laws. The term operator denotes the abstract summary of the whole procedure under consideration, *i.e.* in particular including the discretizations of the initial condition, of the computational domain and of the boundary conditions. In the following, this operator is referred to as the LF-Operator.

The well-known Lax–Friedrichs scheme proposed by Lax in 1954 [4] we investigate reads as

$$U_j^{n+1} = \frac{1}{2}\left(U_{j+1}^n + U_{j-1}^n\right) - \frac{1}{2}\lambda\Big[f\left(U_{j+1}^n\right) - f\left(U_{j-1}^n\right)\Big], \tag{1}$$

where j denotes the spatial index at $j\Delta x$, $k \in \{n, n+1\}$ denotes the temporal level $k\Delta t$, $\lambda := \Delta t/\Delta x$ is the abbreviation of the ratio of mesh widths, and where we employ U as a notation for discrete data. For simplicity, we generally assume that the grid is uniform and that λ is a constant. We also assume the validity of a CFL condition in a strict sense, *i.e.* we use that for all the values ϕ in the interval of given data $\mathcal{I}$ holds

$$\lambda f'(\phi) < 1. \tag{2}$$

Keywords and phrases. Conservation laws, numerical methods, finite difference methods, central methods, Lax–Friedrichs method, total variation stability.

* *This work was done during a research stay at Mathématiques Appliquées de Bordeaux, Université Bordeaux 1, 351, Cours de la Libération, 33405 Talence Cedex, France. The author would like to thank the Deutsche Forschungsgemeinschaft (DFG) for supporting this stay under grant No. BR 2245/1-1.*

[1] Technical University Brunswick, Department for Analysis, Pockelsstraße 14, 38106 Brunswick, Germany.
e-mail: michael.breuss@math.u-bordeaux.fr

We use this inequality in a strict form in some proofs in order to sharpen the relevant assertions; however, (2) is a usual situation within computations and poses no particular restriction. Concerning the flux function f, we generally assume that it is Lipschitz continuous and that it grows monotonously on the relevant range of data, *i.e.*

$$f(\phi + \Delta\phi) \geq f(\phi) \quad \forall\, \phi + \Delta\phi,\, \phi \in \mathcal{I} \text{ with } \Delta\phi \geq 0. \tag{3}$$

This assumption includes the most situations encountered in mathematical models in physics and engineering, especially it includes convex as well as non-convex fluxes.

The Lax–Friedrichs method is often used in textbooks to introduce into the subject of numerical schemes for conservation laws. It features many well-established attributes, the most important ones are that it is a consistent, conservative and monotone method, and hence it is also TVD. The scheme is very easy to use since it does not involve the knowledge about the structure of the solution of Riemann problems and because it uses only flux evaluations. Additionally, the staggered grid approach resulting from the odd-even decoupling of nodes within the stencil of the method can be used in the style of Nessyahu and Tadmor [8] to construct higher order schemes. For an overview on the mentioned topics, see *e.g.* the books of LeVeque [6, 7] or Godlewski and Raviart [2] and the references therein. As it is evident, it is of general interest to investigate the Lax–Friedrichs scheme.

The aim of this work is to illuminate the structure of the LF-operator. We verify by the use of a simple linear model equation that the nature of a numerical solution may change depending on the number and choice of discretization points of the initial condition. Furthermore, the choice of the number of grid points may also be important, as well as (as usual) the specification of the boundary conditions. Under conditions analyzed in detail in 1-D, we rigorously prove that the LF-operator changes from being locally variation preserving to locally variation conserving. We give a recipe which completely specifies the structure of the LF-operator as variation preserving.

To clarify the subject of this paper, let us stress that the usual widely accepted description of the Lax–Friedrichs scheme does not completely specify the properties of the method. Especially, depending on the data and on the details of the discretization, the local variation preserving attribute of the LF-Operator may change in a fashion which immediately influences the quality of the numerical approximation. Phenomenae may arise which were to the knowledge of the author up to now generally thought of as "numerical instabilities". We clarify this subject with respect to the Lax–Friedrichs method. Note that although the theoretical investigation concentrates on the 1-D case, our investigation extends to the approximation of scalar conservation laws in one or more spatial dimensions and to the general case of non-linear monotonously growing Lipschitz continuous fluxes.

The reason why the phenomenae we analyze within this paper were not up to now not the subject of research is to our impression threefold. Firstly, during the time when the Lax–Friedrichs method was introduced, the notion of the total variation was presumably not of so much importance as it was later with respect to the construction of (higher order) TVD schemes. Secondly, the way we analyze the occurring phenomenae is based on observations on the discrete level, while many tools for the numerical analysis of such schemes as the Lax–Friedrichs scheme are based on ideas formulated on the differential level or on Fourier analysis, respectively. Furthermore, and in the same line of the latter argumentation, we did not encounter within the literature an analysis employing pairs of cells on the discrete level which is recommended here. At last, many investigations focus on the numerical approximation of Riemann problems incorporating only two states U_{left} and U_{right} where the phenomenae which are investigated here simply do not occur.

The subject of this paper is also linked to the results of investigations concerned with the construction of so-called generalized monotone schemes, see [5] and the discussion therein. The main point of interest about these methods in the context referred to in this paper is that they mimic a property of the analytical solutions of 1-D scalar conservation laws already mentioned, namely that the number of extrema does not increase in time.

Especially, LeFloch and Liu note that the Lax–Friedrichs scheme is not a generalized monotone scheme. As a proof, they discretize the equation $u_t = 0$, using also the initial conditions

$$U_j^n \equiv 0 \text{ for all } j \neq 0 \text{ but } U_0^n > 0.$$

The straightforward application of the method (1) yields

$$U_{-1}^{n+1} = \frac{1}{2}U_0^n \quad \text{and} \quad U_1^{n+1} = \frac{1}{2}U_0^n$$

while all other values are zero, so that the method has produced two new extrema (a local minimum at $j = 0$ and a new local maximum).

Besides that one could argue about the example, let us introduce a change within the setting, namely let us define the grid as consisting only of the three points $\{-1, 0, 1\}$ and introduce periodic boundary conditions. Then exactly the same solution as above has not changed the number of extrema. The point to note is, that a change within the setting even in this trivial example changes a significant property of the Lax–Friedrichs scheme.

In the more general case of a virtually infinite grid which is of course really addressed in [5], the lack of the generalized monotonicity property of the Lax–Friedrichs method originates in the discretization of the underlying initial condition employed in the cited example which defines the LF-Operator as being (globally) variation conserving; thus the creation of new extrema. If the recipe we develop within this paper is followed, we show that such a case cannot occur and the LF-Operator defines a generalized monotone scheme. Let us also note, that the discussion given in [5] is limited to 1-D scalar conservation laws with convex fluxes, while a properly set up Lax–Friedrichs method defines a generalized monotone scheme also in the case of more general fluxes as well as in two or more spatial dimensions.

The content of this paper is organized as follows. In the second section, we analyze the basic phenomenae which may arise by the naive use of the Lax–Friedrichs scheme in 1-D. Here, we employ a virtually infinite grid. As a motivation of the proceeding, we use a simple linear model problem. Within the third section, we discuss the effects of the discretizations of a finite computational domain with boundary conditions on the numerical solution. After that we summarize the observations, and we develop the recipe for the correct application of the Lax–Friedrichs scheme. The usefulness of our approach is validated by the treatment of a non-linear example, and we also show the validity of our investigation in the multi-dimensional case. We finish the paper with some conclusive remarks and acknowledgements.

1. A linear model problem

In this section, we consider the linear advection equation

$$\frac{\partial}{\partial t}u\,(x,\,t) + \frac{\partial}{\partial x}u\,(x,\,t) = 0, \quad x \in \mathbf{R},\, t > 0. \tag{4}$$

The differential equation (4) has to be supplemented by an initial condition $u\,(x,\,0) = u_0\,(x)$, and we employ a virtually infinite computational domain $\mathbf{D}$. We discuss a special choice of the initial condition, namely a square signal

$$u_0\,(x) = \begin{cases} u_h & : \quad x_l \leq x \leq x_r, \\ u_l & : \quad \text{else}, \end{cases}$$

with $x_l < x_r$ and $x_l,\, x_r \in \mathbf{D}$, and where we set without a restriction on the generality of our discussion $u_l < u_h$.

If we do not state something different explicitly, we employ the parameters $\Delta x = 0.1$ and $\Delta t = 0.02$ for our numerical tests. This choice yields the stability of the method.

As computational domain we discretize for the examples within this section the interval $[0,\, 15]$ which is large enough so that no influence of the numerical boundary conditions is visible during the computations.

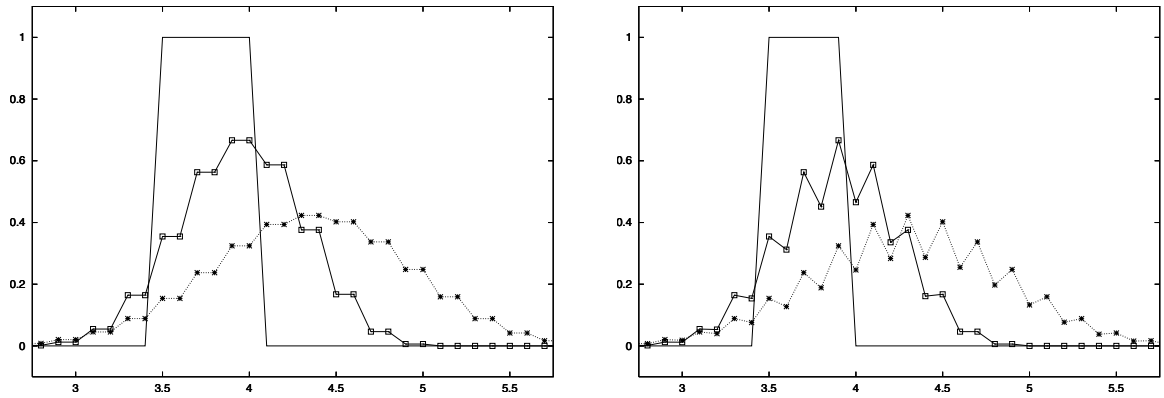

FIGURE 1. This figure shows (a) on the left hand side numerical results based on the initial condition (5), and (b) on the right hand side numerical results based on the initial condition (6). In both cases, the initial condition together with the numerical solutions after 10 (lines with boxes) and 30 (dotted lines with stars) time steps are displayed, respectively.

Within our first example, we discretize the initial condition

$$\begin{aligned} & u_0(x) = \begin{cases} 1 & : \quad 3.5 \le x \le 4, \\ 0 & : \quad \text{else}, \end{cases} \\ \text{setting} \qquad & \begin{array}{ll} U_j^0 := 1 & \text{for} \quad j \in \{35, \ldots, 40\}, \\ U_j^0 := 0 & \text{else}. \end{array} \end{aligned} \tag{5}$$

That is, the number of discretization points for the square signal is 6, so it is even.

Let us note here, that we do not seek to discuss the effects of taking point values as data. Rather than that, we want to discuss the effects of specific structures of data values. For example, the same initial data as in (5) could have been obtained by taking cell averages for the initial condition

$$\tilde{u}_0(x) = \begin{cases} 1 & : \quad 3.5 - \Delta x \le x \le 4 + \Delta x, \\ 0 & : \quad \text{else}, \end{cases}$$

when centering the cells at $j\Delta x$, $j \in \mathbf{Z}$.

The initial condition (5) together with snapshots of its temporal evolution is shown in Figure 1a. We observe that the numerical solution evolves as it could have been expected by the application of the Lax–Friedrichs method.

Now we modify the initial condition. We discretize

$$\begin{aligned} & u_0(x) = \begin{cases} 1 & : \quad 3.5 \le x \le 3.9, \\ 0 & : \quad \text{else}, \end{cases} \\ \text{setting} \qquad & \begin{array}{ll} U_j^0 := 1 & \text{for} \quad j \in \{35, \ldots, 39\}, \\ U_j^0 := 0 & \text{else}. \end{array} \end{aligned} \tag{6}$$

The number of discretization points for the square signal is now 5, thus it is odd.

As for the first example we display the initial condition together with snapshots of its temporal evolution, see Figure 1b. The oscillations appearing in the numerical solution are clearly observable.

This is a very surprising phenomenon because the Lax–Friedrichs scheme incorporates excessive numerical viscosity, and thus one expects that the solution always looks like an approximation of a smooth solution of an

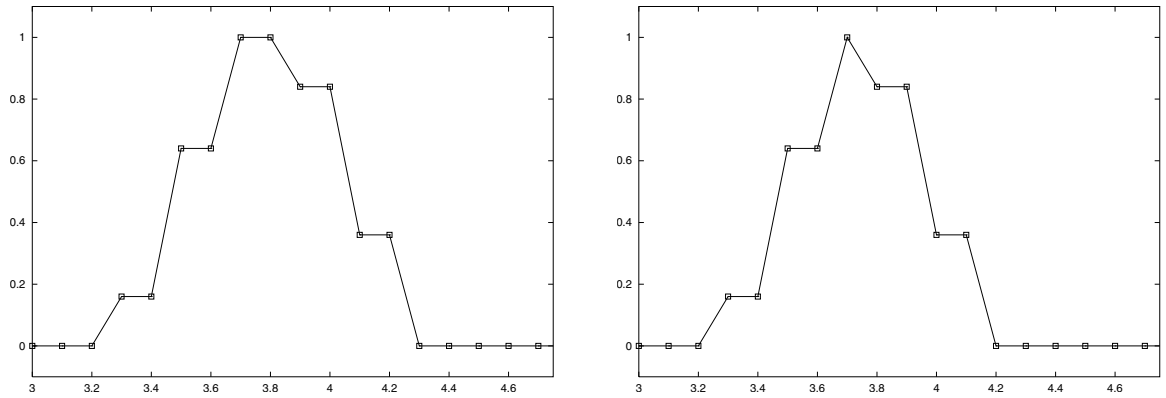

FIGURE 2. The figure shows the numerical solutions after 2 timesteps, (a) on the left hand side using the initial condition (5), and (b) on the right hand side using the initial condition (6).

advection-diffusion equation. Especially, the occurring oscillations are very unusual. One could also ask *e.g.* for the monotonicity property; however, the validity of this property is guaranteed by well-established theoretical assertions. Since the monotonicity establishes a comparison principle

$$V^n \geq W^n \quad \Rightarrow \quad V^{n+1} \geq W^{n+1}$$

for data sets V^k, W^k, $k \in \{n, n+1\}$, it is easily seen that such a comparison principle also holds in the oscillatory case depicted in Figure 1b.

If we write for instance the Lax–Friedrichs scheme (1) for the linear case in the form

$$U_j^{n+1} = \frac{1}{2}\left(1 - \lambda a\right) U_{j+1}^n + \frac{1}{2}\left(1 + \lambda a\right) U_{j-1}^n,$$

it becomes evident that we observe in Figure 1b a specific feature of the interpolation procedure inherently involved with the scheme. Note, that in the case of non-linear fluxes the situation is qualitatively the same.

Let us discuss what had happened in the course of the computations of the solutions depicted in Figures 1a and 1b in some detail.

We applied the Lax–Friedrichs scheme (1) on a discrete Riemann-problem, *i.e.* on a sequence of data

$$\begin{gathered} \ldots, U_{ll}, U_l, U_r, U_{rr}, \ldots \\ \text{with} \quad \ldots = U_{ll} = U_l \ \text{ and } \ U_r = U_{rr} = \ldots \end{gathered}$$

At the two cells at the jump from U_l to U_r, the method (1) gives

$$U_l^{\text{new}} = U_r^{\text{new}} = \frac{1}{2}\left(U_r + U_l\right) - \frac{\lambda}{2}\left(f\left(U_r\right) - f\left(U_r\right)\right),$$

forming a step consisting of two cells as given data for the next time step. A structure of stairs arises with two cells in every step.

Since data sequences consisting of such steps will be very useful within our discussion, we denote two adjacent cells featuring the same data value as a *2-cell*, while a single cell with a different value with respect to its neighbours is denoted as a *1-cell*.

Because the Lax–Friedrichs method thins the plateaus of the initial square signals (5) and (6) by one point each time step in the described fashion, we arrive after 2 time steps at the situations depicted in Figure 2. The discussed 2-cell structures are clearly visible. However, we observe a difference which originates in the use

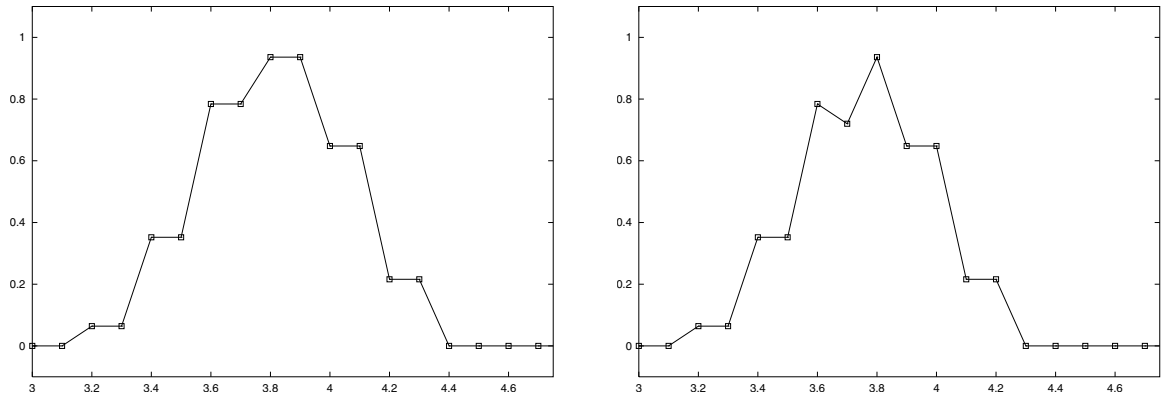

FIGURE 3. Analogously to Figure 2, this figure shows the numerical solutions after 3 timesteps.

of an even number and an odd number of cells for the discretization of the initial square signal, respectively. As depicted in Figure 2a, an even number results in a maximum consisting of a 2-cell, while as displayed in Figure 2b, an odd number implies that a 1-cell is left from the original plateau.

The influence of this difference is immediately seen after the computation of the next time step; the results are displayed in Figure 3. In Figure 3a, we see that the 2-cell structure is preserved and the plateau is lowered as it is natural by the application of (1). By Figure 3b, we see that the other solution is oscillatory.

In order to state clearly what we mean by the latter assertion, let us give the following

Definition 1.1. Let a data sequence $\{U_l, U_m, U_r\}$ be given. We define the state U_m^{new} obtained by application of the Lax–Friedrichs scheme as an oscillation if one of the two assertions holds:

$$\begin{aligned} U_l \le U_m \ge U_r \quad &\text{and} \quad U_l^{\text{new}} > U_m^{\text{new}} < U_r^{\text{new}} \text{ is true,} \\ \text{or } U_l \ge U_m \le U_r \quad &\text{and} \quad U_l^{\text{new}} < U_m^{\text{new}} > U_r^{\text{new}} \text{ is true.} \end{aligned}$$

For example, in the situation investigated by Figures 2b and 3b, there is an oscillation exactly at $x = 3.7$, while the local maxima at $x = 3.6$ and 3.8 displayed in Figure 3b are not oscillations.

Next, we want to investigate rigorously, if oscillations generally arise in the case of monotonously growing Lipschitz continuous fluxes for data as encountered in the discussed example. For the proceeding, it is useful to give the technical

Lemma 1.2. *Let two numbers a and b be given with $b > a$ and a,b in $\mathcal{I}$. Then the following inequalities hold:*

$$b - \lambda f(b) > a - \lambda f(a) \tag{7}$$

$$\textit{and} \qquad b + \lambda f(b) > a + \lambda f(a). \tag{8}$$

Proof. The inequality (7) holds if and only if

$$(b - a) - \lambda \left[f(b) - f(a) \right] > 0$$

is true. Division by $b - a$ yields, that this inequality is equivalent to

$$1 - \lambda \frac{f(b) - f(a)}{b - a} > 0.$$

Since f is Lipschitz continuous, there exists a state a^* in the open interval $(a, b) \subset \mathcal{I}$ such that the latter inequality reads as

$$1 - \lambda f'(a^*) > 0$$

which is exactly the CFL condition (2). Since we assumed that the strict CFL condition holds, (7) is verified. The validity of (8) follows by $b > a$ and because of the assumption (3). □

Then we prove as indicated the following

Lemma 1.3. *A local extremum consisting of a 1-cell which has only 2-cells as neighbours always yields an oscillation.*

Proof. Let us locate a local maximum at U_m. It is sufficient to investigate the data sequence

$$\{U_{ll}, U_l, U_m, U_r, U_{rr}\}$$

within one time step with

$$U_{ll} = U_l,\ U_l < U_m > U_r \text{ and } U_r = U_{rr}.$$

We now compute the values U_l^{new}, U_m^{new} and U_r^{new} using (1). Under the described setting, these are

$$\begin{aligned} U_l^{\text{new}} &= \frac{1}{2}(U_l + U_m) - \frac{\lambda}{2}(f(U_m) - f(U_l)), \\ U_m^{\text{new}} &= \frac{1}{2}(U_l + U_r) - \frac{\lambda}{2}(f(U_r) - f(U_l)) \\ \text{and } U_r^{\text{new}} &= \frac{1}{2}(U_m + U_r) - \frac{\lambda}{2}(f(U_r) - f(U_m)). \end{aligned}$$

We can then compute the differences $U_l^{\text{new}} - U_m^{\text{new}}$ and $U_r^{\text{new}} - U_m^{\text{new}}$ in a straightforward fashion. We have

$$U_l^{\text{new}} - U_m^{\text{new}} = \frac{1}{2}(U_m - U_r) - \frac{\lambda}{2}(f(U_m) - f(U_r)).$$

By (7) we thus obtain $U_l^{\text{new}} - U_m^{\text{new}} > 0$. Analogously, we find

$$U_r^{\text{new}} - U_m^{\text{new}} = \frac{1}{2}(U_m - U_l) + \frac{\lambda}{2}(f(U_m) - f(U_l)).$$

Application of (8) on this equation yields $U_r^{\text{new}} - U_m^{\text{new}} > 0$. The result for a local minimum follows analogously. □

In the light of the comments given up to now, the meaning of Lemma 1.3 is that the interpolation procedure incorporated in the scheme always gives an oscillation for the indicated structure of data, no matter if the flux is linear or non-linear (but monotonously growing and Lipschitz continuous).

Moreover, we prove the following

Lemma 1.4. *An oscillation arising in the way described by Lemma 1.3 conserves the local variation of the data exactly.*

Proof. For the proof we compute in a straightforward fashion the variation of the data in the indicated situation. We only do this for the case of a local maximum, the assertion for a local minimum follows analogously.

With Figure 4 we illustrate the procedure of the computation of the variation of the data directly at the local maximum, just before an oscillation takes place. The dashed line indicates the height of the local maximum.

This dashed line is taken over into Figure 5 which shows the situation as the oscillation appears after the next time step when using data as displayed in Figure 4. The idea behind the computations within the proof is illustrated: we compute the quantities indicated by δ_i, $i = 1, 2$, within Figure 5 and show their equality, respectively.

Therefore, we again investigate a data sequence

$$\{U_{ll}, U_l, U_m, U_r, U_{rr}\}$$

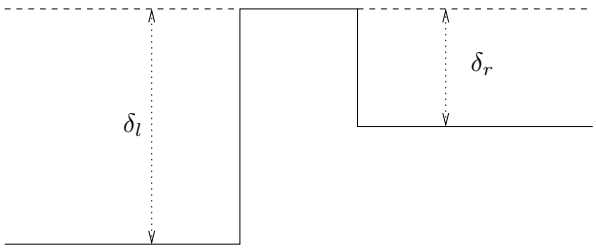

FIGURE 4. The figure shows a local maximum consisting of a 1-cell surrounded by 2-cells. The level of the maximum is represented by a dashed line. The computation of the total variation especially incorporates summing up the data differences – denoted here δ_l and δ_r – left and right of a local extremum.

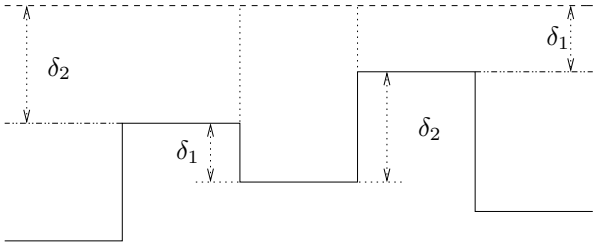

FIGURE 5. The figure sketches the principle behind the proof of the conservation of the total variation. We observe which lengths can be identified when considering the differences of the new data with respect to the level of the former local maximum indicated by the dashed line.

with $U_{ll} = U_l$, $U_l < U_m > U_r$ and $U_{rr} = U_r$. As in the proof of Lemma 1.3 we can compute the values

$$\begin{aligned} U_l^{\text{new}} &= \frac{1}{2}\left(U_m + U_l\right) - \frac{\lambda}{2}\left(f\left(U_m\right) - f\left(U_l\right)\right), \\ U_r^{\text{new}} &= \frac{1}{2}\left(U_r + U_m\right) - \frac{\lambda}{2}\left(f\left(U_r\right) - f\left(U_m\right)\right). \end{aligned}$$

Note that these are exactly the values we would obtain if U_m would be located in a 2-cell. We already know from Lemma 1.3 that

$$U_l^{\text{new}} > U_m^{\text{new}} \quad \text{and} \quad U_r^{\text{new}} > U_m^{\text{new}}$$

holds. Thus, we have on the one hand a *loss* in variation by

$$U_m - U_l^{\text{new}} \quad \text{and} \quad U_m - U_r^{\text{new}},$$

while we have on the other hand a *gain* in variation by

$$U_l^{\text{new}} - U_m^{\text{new}} \quad \text{and} \quad U_r^{\text{new}} - U_m^{\text{new}}.$$

Note that we did not establish rigorously up to now, that for general fluxes f the values U_l^{new} and U_r^{new} left and right of an oscillation in U_m^{new} are bounded by the value of the former extremum U_m. Thus, the *loss* indicated

above could be negative (which means, a gain) up to now. We compute the terms contributing to the loss.

$$\begin{aligned} U_m - U_l^{\text{new}} &= \frac{1}{2}(U_m - U_l) + \frac{\lambda}{2}(f(U_m) - f(U_l)), \\ U_m - U_r^{\text{new}} &= \frac{1}{2}(U_m - U_r) - \frac{\lambda}{2}(f(U_m) - f(U_r)). \end{aligned}$$

Note that $U_m - U_l^{\text{new}}$ is positive since $U_m > U_l$ and (3), while $U_m - U_r^{\text{new}}$ is positive because of Lemma 1.2. Thus, there is a real (*i.e.* positive) loss in variation. The computation of the terms contributing to the gain in variation yield

$$\begin{aligned} U_l^{\text{new}} - U_m^{\text{new}} &= \frac{1}{2}(U_m - U_r) - \frac{\lambda}{2}(f(U_m) - f(U_r)), \\ U_r^{\text{new}} - U_m^{\text{new}} &= \frac{1}{2}(U_m - U_l) + \frac{\lambda}{2}(f(U_m) - f(U_l)). \end{aligned}$$

We obtain

$$U_m - U_l^{\text{new}} = U_r^{\text{new}} - U_m^{\text{new}} \quad \text{and} \quad U_m - U_r^{\text{new}} = U_l^{\text{new}} - U_m^{\text{new}},$$

which means that the *loss* in variation is equal to the *gain* in variation, *i.e.*, the local variation is exactly conserved. □

It is natural to ask for the case when cells in the vicinity of an extremum do not have a 2-cell structure. Therefore, let us briefly study the steps within the proof of Lemma 1.3. Let us again fix a local extremum at U_m with $U_m > U_l$ and $U_m > U_r$.

Since we now want to give U_{ll} and U_{rr} some degree of freedom, let us introduce unspecified quantities δ_l and δ_r with

$$U_{ll} \equiv U_l + \delta_l \quad \text{and} \quad U_{rr} \equiv U_r + \delta_r.$$

Then we cannot simplify the computation of the differences $U_l^{\text{new}} - U_m^{\text{new}} > 0$ and $U_r^{\text{new}} - U_m^{\text{new}} > 0$ as within the proof of Lemma 1.3.

The re-computation of $U_l^{\text{new}} - U_m^{\text{new}}$ gives

$$\begin{aligned} U_l^{\text{new}} - U_m^{\text{new}} &= \frac{1}{2}(U_l + \delta_l + U_m) - \frac{\lambda}{2}(f(U_m) - f(U_l + \delta_l)) \\ &\quad -\frac{1}{2}(U_l + U_r) + \frac{\lambda}{2}(f(U_r) - f(U_l)) \\ &= \frac{1}{2}(U_m - U_r) - \frac{\lambda}{2}(f(U_m) - f(U_r)) \\ &\quad +\frac{1}{2}\delta_l + \frac{\lambda}{2}(f(U_l + \delta_l) - f(U_l)). \end{aligned} \tag{9}$$

Analogously, we compute for $U_r^{\text{new}} - U_m^{\text{new}}$

$$\begin{aligned} U_r^{\text{new}} - U_m^{\text{new}} &= \frac{1}{2}(U_r + \delta_r + U_m) - \frac{\lambda}{2}(f(U_r + \delta_r) - f(U_m)) \\ &\quad -\frac{1}{2}(U_l + U_r) + \frac{\lambda}{2}(f(U_r) - f(U_l)) \\ &= \frac{1}{2}(U_m - U_l) + \frac{\lambda}{2}(f(U_m) - f(U_l)) \\ &\quad +\frac{1}{2}\delta_r - \frac{\lambda}{2}(f(U_r + \delta_r) - f(U_r)). \end{aligned} \tag{10}$$

Comparing to the situation within the proof of Lemma 1.3, extra terms have arisen. Concerning the extra terms in (9), *i.e.*

$$\frac{1}{2}\delta_l + \frac{\lambda}{2}\left(f\left(U_l + \delta_l\right) - f\left(U_l\right)\right),$$

one can notice that they show a positive or negative value for a positive or negative sign of δ_l, respectively, because of the assumed monotonicity of f. The analysis of the extra term in (10), *i.e.*

$$\frac{1}{2}\delta_r - \frac{\lambda}{2}\left(f\left(U_r + \delta_r\right) - f\left(U_r\right)\right),$$

requires a few more steps. For positive δ_r, division by $1/2\,\delta_r$ and the use of (7) shows that the value of the extra term is positive. If $\delta_r < 0$ holds, the value of the extra term is only positive if and only if

$$\delta_r > \lambda\left(f\left(U_r + \delta_r\right) - f\left(U_r\right)\right)$$

is satisfied. Multiplication of the latter inequality with (-1) and the use of the abbreviation $\tilde{\delta}_r := -\delta_r$ yields its equivalence with

$$\tilde{\delta}_r < \lambda\left(f\left(U_r - \tilde{\delta}_r\right) - f\left(U_r\right)\right).$$

Division by $\tilde{\delta}_r$ leads to the condition

$$\lambda\frac{f\left(U_r - \tilde{\delta}_r\right) - f\left(U_r\right)}{\tilde{\delta}_r} > 1.$$

This condition is equivalent to

$$\lambda f'\left(\tilde{U}_r\right) > 1$$

for a value $\tilde{U}_r \in (U_r - \tilde{\delta}_r, U_r)$, which is in direct contradiction to the fulfilled CFL condition (2). Thus, the extra term in (10) is negative for negative δ_r.

However, in general these terms have to be balanced against the other terms in (9) and (10), respectively. As a consequence of the given discussion, we write down the following proposition, which includes the case of minima which can be treated in an identical fashion.

Proposition 1.5. *If locally given data*

$$\{U_{ll}, U_l, U_m, U_r, U_{rr}\}$$

features either the structure

$$\begin{aligned} & U_{ll} \geq U_l,\ U_l < U_m > U_r \text{ and } U_r \leq U_{rr} \\ \text{or} \quad & U_{ll} \leq U_l,\ U_l > U_m < U_r \text{ and } U_r \geq U_{rr}, \end{aligned}$$

then an oscillation will arise at U_m^{new}.

This proposition may appear relatively innocent at first glance. However, one might think that any oscillation in given data (*e.g.* in U_m^{new} for $U_{ll} > U_l < U_m > U_r < U_{rr}$) might die out regardless of the data simply by application of the scheme (1) due to the large amount of numerical viscosity of the method. This proposition tells us that exactly this is not the case. Moreover, the above proposition shows, that in general any arising oscillation might induce further oscillations in the surrounding cells. The reason is that an arising oscillation may leave the surrounding cells as local extrema, and then the whole proceeding illuminated up to now applies again on these extrema.

Having discussed this, we ask for a generalization of Lemma 1.4. An answer is given by

Theorem 1.6. *Arising oscillations conserve the local variation of the data of the preceding time step exactly.*

Proof. It will be sufficient to consider a slight generalization of the proceeding leading to Proposition 1.5.

We investigate an arbitrarily chosen but fixed sequence of data

$$\{\dots, U_{ll}, U_l, U_m, U_r, U_{rr}, \dots\}$$

featuring the situation

$$U_{ll} \geq U_l < U_m > U_r \leq U_{rr}.$$

Thereby, we do not assume any particular relationship to hold between U_{ll}, U_m and U_{rr}.

As within the derivation of Proposition 1.5 we introduce quantities Δ_l and Δ_r with

$$U_{ll} \equiv U_l + \Delta_l \quad \text{and} \quad U_{rr} \equiv U_r + \Delta_r,$$

however, this time we define $\Delta_l \geq 0$ and $\Delta_r \geq 0$. This specifies the investigated situation as featuring a local maximum U_m located in a 1-cell where we observe the effects of oscillatory perturbations of the 2-cells left and right to U_m.

The terms U_l^{new}, U_m^{new} and U_r^{new} can again be computed using the described setting, yielding as in (9) and (10) the differences

$$\begin{aligned} U_l^{\text{new}} - U_m^{\text{new}} &= \frac{1}{2}(U_m - U_r) - \frac{\lambda}{2}(f(U_m) - f(U_r)) \\ &\quad + \frac{1}{2}\Delta_l + \frac{\lambda}{2}(f(U_l + \Delta_l) - f(U_l)), \qquad (11) \\ U_r^{\text{new}} - U_m^{\text{new}} &= \frac{1}{2}(U_m - U_l) + \frac{\lambda}{2}(f(U_m) - f(U_l)) \\ &\quad + \frac{1}{2}\Delta_r - \frac{\lambda}{2}(f(U_r + \Delta_r) - f(U_r)). \qquad (12) \end{aligned}$$

Let us decompose the right hand side of (11) into the parts

$$\begin{aligned} \theta &:= \frac{1}{2}(U_m - U_r) - \frac{\lambda}{2}(f(U_m) - f(U_r)) \\ \text{and} \quad \eta &:= \frac{1}{2}\Delta_l + \frac{\lambda}{2}(f(U_l + \Delta_l) - f(U_l)). \end{aligned}$$

As Lemma 1.4 showed, the term θ stands for the equalizing of the loss in the local variation due to the oscillation in U_m^{new} if there are 2-cells left and right from U_m.

Moreover, if we locate the value $U_{ll} \equiv \tilde{U}_m$ at the center of another oscillation, *i.e.* in $U_{ll}^{\text{new}} \equiv \tilde{U}_m^{\text{new}}$, the term η turns out to be exactly the difference between

$$\tilde{U}_r^{\text{new}} - \tilde{U}_m^{\text{new}} \quad \text{for} \quad U_l \equiv \tilde{U}_r,\ U_{llll} \equiv \tilde{U}_l \equiv U_{lll} \text{ and } U_m \equiv \tilde{U}_{rr}.$$

Thus, these terms are given such that they equalize the loss of variation due to an oscillation in U_{ll}^{new}. Analogously, it can be argued that the extra terms in (12) equalize the loss of variation due to an oscillation in U_{rr}^{new}.

Since the stencil of the Lax–Friedrichs scheme does not involve more points than the discussed ones, all possible interactions are covered. □

Let us observe the important fact that the proceeding within the proof of Theorem 1.6 relies completely on the consideration of $(2-1-2)$-cell sequences of data. Note also that the proposed decomposition works for general non-linear fluxes while the terms contributing to an oscillation simply add up.

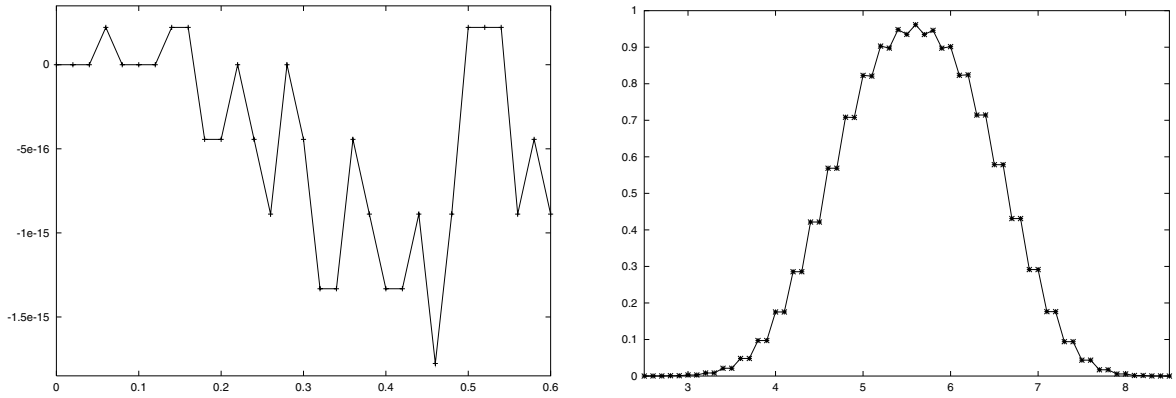

FIGURE 6. The figure shows (a) on the left hand side the evolution of the error within the total variation within the first 30 timesteps, and (b) on the right hand side the corresponding numerical solution after 30 timesteps.

As a numerical verification for the content of this theorem, we consider the same setting as before but with the initial data

$$\text{with} \qquad \begin{array}{l} u_0(x) = \left\{ \begin{array}{lcl} 1 & : & 4.0 \leq x \leq 6.0, \\ 0 & : & \text{else}, \end{array} \right. \\ U_j^0 := 1 \quad \text{for} \quad j \in \{40, \ldots, 60\}, \\ U_j^0 := 0 \qquad\qquad \text{else}. \end{array} \tag{13}$$

That is, the number of discretization points for the square signal is 21, so it is odd and oscillations will appear. We investigate the error within the evolution of the total variation during the first 30 time steps. We choose the initial condition (13) because it enables us to observe the total variation also in some detail during the first 10 time steps, *i.e.* before an oscillation can take place.

Figure 6 displays the mentioned error, *i.e.* it depicts the size of the term

$$2.0 \; - \; \langle \text{ computed total variation } \rangle ,$$

together with the numerical solution after 30 timesteps. We see that the computed total variation stays almost identical to the original total variation. The slight differences computed here also occur especially within the first ten time steps in the same order of magnitude.

It remains to discuss, at which situations within an arbitrary computation an oscillation may arise. In general, it is clear that this depends on the balance of terms derived in (9) and (10). However, we can state the following

Theorem 1.7. *During any one time step, an oscillation may only arise at a local extremum consisting of a 1-cell.*

Proof. Since the method is monotone, it is also monotonicity preserving. Thus, no oscillation can arise originating from locally monotone profiles with the exception of local extrema.

The discussion of extrema consisting of more than two cells reduces to the discussion of 2-cell extrema since these form their border at the transition to the non-extremal region.

We now have to distinguish between extrema consisting of 1-cells and of 2-cells.

For extrema consisting of 1-cells, it was already shown that an oscillation may arise. Thus, we investigate now the case of extrema consisting of 2-cells. Thereby, we exclude extrema consisting of 1-cells in the discussion.

Without restriction on the generality, we only discuss a local maximum at U_m, *i.e.*, we investigate a situation

$$\ldots, U_{ll}, U_l, U_m, U_m, U_r, U_{rr}, \ldots$$

with $U_m > U_l$ and $U_m > U_r$. By consideration of (9) and (10), we exclude the cases $U_{ll} > U_l$ and $U_{rr} > U_r$ since otherwise we would have local extrema at 1-cells, namely at U_l and U_r. Thus, $U_{ll} \leq U_l$ and $U_{rr} \leq U_r$ holds for our discussion.

At the jump from U_l to U_m we obtain

$$\begin{aligned} U_m^{\text{new}} - U_l^{\text{new}} &= \frac{1}{2}\left(U_l - U_{ll}\right) + \frac{\lambda}{2}\left(f\left(U_l\right) - f\left(U_{ll}\right)\right) \\ &\geq 0 \quad \text{by (8)}, \end{aligned}$$

and at the jump from U_m to U_r we compute

$$\begin{aligned} U_m^{\text{new}} - U_r^{\text{new}} &= \frac{1}{2}\left(U_r - U_{rr}\right) - \frac{\lambda}{2}\left(f\left(U_r\right) - f\left(U_{rr}\right)\right) \\ &\geq 0 \quad \text{by (7)}. \end{aligned}$$

Thus, no oscillation can occur in the case of a 2-cell extremum. □

Note, however, that the assertion of Theorem 1.7 is limited to exactly one time step. For example, a 2-cell extremum may be reduced to a 1-cell extremum during exactly the discussed time step, and after that an oscillation may arise at this newly formed 1-cell extremum. Thus, it would be desirable that also monotone profiles feature a 2-cell structure where this cannot happen.

Let us also note, that Theorem 1.7 gives not an assertion concerning an automatic mechanism, *i.e.*, of course, not each time a local extremum appears an oscillation is created. Our numerical tests show, that initial data featuring transition steps between zones of constant data values (as they are most of the time created by employing cell averages for the discretization of initial data) are more suited than sharp transitions in order to avoid phenomenae as discussed here.

2. Other discretization parameters

Within this section, we show how the number of discretization points of a finite computational domain together with the numerical boundary conditions influence the structure of the LF-Operator.

We employ again the set up of the linear test problems from the section before, and unless stated otherwise we use periodic boundary conditions for the discussion. We will see, that the influence of the boundary conditions acts as a natural extension concerning the properties of the LF-Operator discussed before.

To illuminate this assertion, let us use the initial condition

$$\begin{aligned} & u_0\left(x\right) = \begin{cases} 1 & : \quad 1.0 \leq x \leq 1.9, \\ 0 & : \quad \text{else}, \end{cases} \\ \text{setting} \qquad & \begin{aligned} U_j^0 &:= 1 \quad \text{for} \quad j \in \{10, \ldots, 19\}, \\ U_j^0 &:= 0 \qquad\qquad \text{else}. \end{aligned} \end{aligned} \tag{14}$$

That is, we employ an even number of points for the discretization of the initial square signal. Our grid consists of the discretized domain $[0.0, 2.9]$, which means that we have 30 discretization points for it.

An even number of points for both discretizations means that everything will be as expected by the application of the method since the 2-cell structure which will arise by the numerical solution of the problem will not be influenced. Note that this will also be the case if employing *e.g.* von Neumann boundary conditions here, since in that case a natural 2-cell structure arises at the boundary. To complete the discussion of this test case,

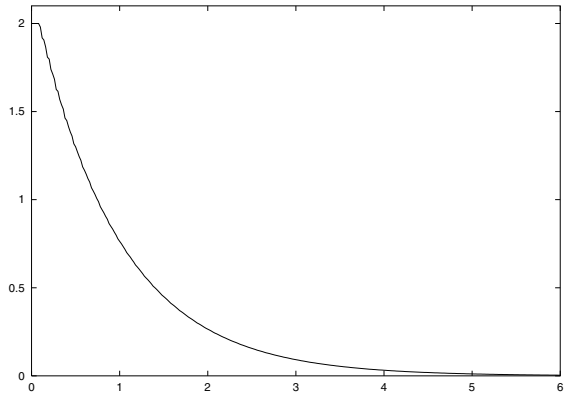

FIGURE 7. The figure shows the evolution of the total variation within the first 300 timesteps.

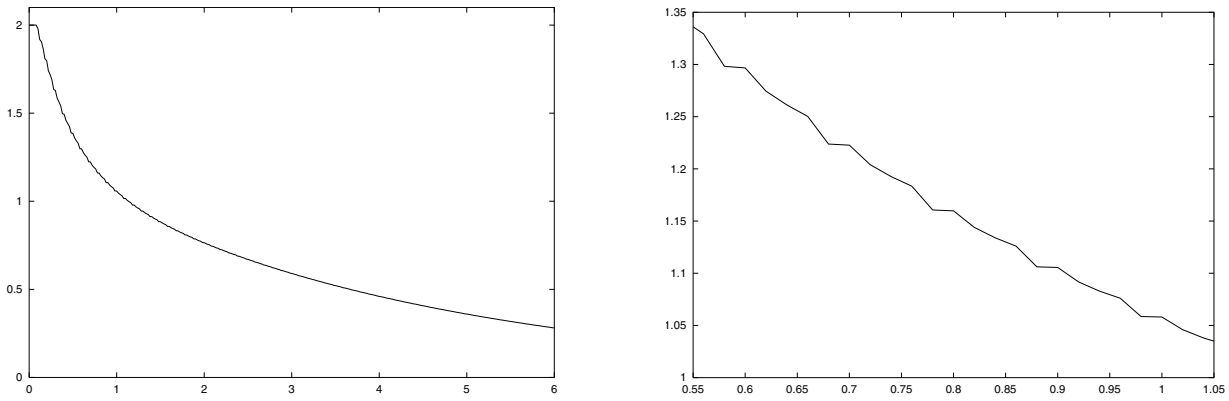

FIGURE 8. The figure shows the evolution of the total variation (a) on the left hand side up to $t = 3$, and (b) on the right hand side from $t = 0.55$ to $t = 1.05$.

we give a plot of the evolution of the total variation during the first 300 time steps, *i.e.* until $t = 6$, see Figure 7. We see that the total variation decreases in a strictly monotone fashion to 0, which could have been expected since the Lax–Friedrichs method is very diffusive.

Now, we slightly change this setting and employ an odd number of points for the computational domain instead of an even one, *i.e.*, the grid consists of the discretized domain $[0.0, 3.0]$ with 31 discretization points. Figure 8 illustrates the result obtained for the evolution of the total variation exactly analogous to Figure 7, together with a zoom into the evolution from time $t = 0.55$ to $t = 1.05$ (where the occurring phenomenon can be observed very sharply). We see, that the total variation decreases in a different fashion as in the example before.

The reason is the interaction of the smeared parts of the solution. There, automatically extrema consisting of 1-cells and local variation conserving oscillations arise since the grid features an odd number of points. Thus, the total variation decreases monotonously, but not in the same fashion as before. However, the size of these oscillations is small compared to the size of the main part of the smeared square signal, and so the oscillations will be assimilated as parts of the monotone profile of that main part, and they will eventually die out in this example.

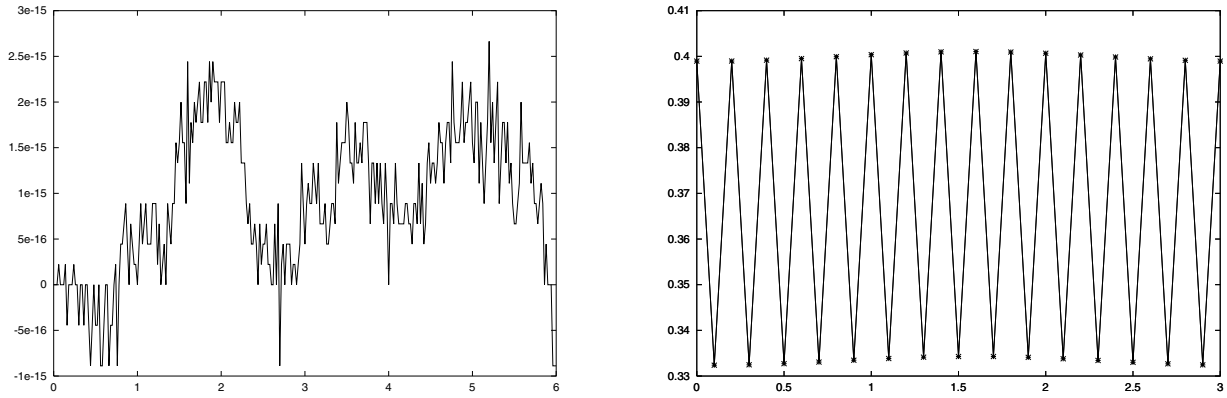

FIGURE 9. The figure shows (a) on the left hand side the error within the evolution of the total variation up to $t = 3$, and (b) on the right hand side the numerical solution at $t = 3$.

Let us now consider an odd number of points for the square signal

$$u_0(x) = \begin{cases} 1 & : \quad 1.0 \le x \le 2.0, \\ 0 & : \quad \text{else}, \end{cases}$$

$$\text{namely} \qquad \begin{array}{ll} U_j^0 := 1 & \text{for} \quad j \in \{10, \ldots, 20\}, \\ U_j^0 := 0 & \text{else}, \end{array} \tag{15}$$

together with en even number of grid points, *i.e.*, again for $[0.0, 2.9]$.

As indicated in the section before, oscillations will arise. Since this means a maximum followed by a minimum – which is again a structure consisting of two cells – an even number of grid points means that the oscillations may not interact, and exactly this happens. Figure 9 shows the error within the evolution of the total variation up to $t = 3$ together with the state at $t = 3$. As expected, the error within the evolution of the total variation stays within reasonable bounds around zero, while the numerical solution distributes the initial total variation over 30 mesh points, so that it is very oscillatory. Thus, in this case the LF-Operator is total variation conserving.

If we change the setting just discussed by one point of the computational domain, the two-cell-structures "minimum following maximum" shown in Figure 9b have to interact because of the odd number of mesh points. After the discussion within the last section it is no surprise that they may cancel out in the long run: We document this by showing Figure 10 which displays the evolution of the total variation up to $t = 25$ together with the state at $t = 25$. This state is of course not a steady state, but it is a good example of an intermediate state which is only slightly oscillatory. The total variation goes in the manner of an exponential function to zero; however, in the details one could observe similar phenomenae as shown in Figure 8b.

We conclude this section with the obvious result of our investigation, summarized *via*

Proposition 2.1. *In order to specify the properties of the LF-Operator completely, the following parts of the setting in addition to the scheme (1) have to be clarified:*

- *The discretization of the initial condition has to be described in detail.*
- *The discretization of the computational domain has to be described in detail.*
- *The boundary condition and its implementation has to be described in detail.*

3. The specification of the LF-operator

It is evident, that the meaning of Proposition 2.1 yields a much too complicated procedure for setting up the Lax–Friedrichs method. Moreover, even if the whole setting is described, it will not generally be evident

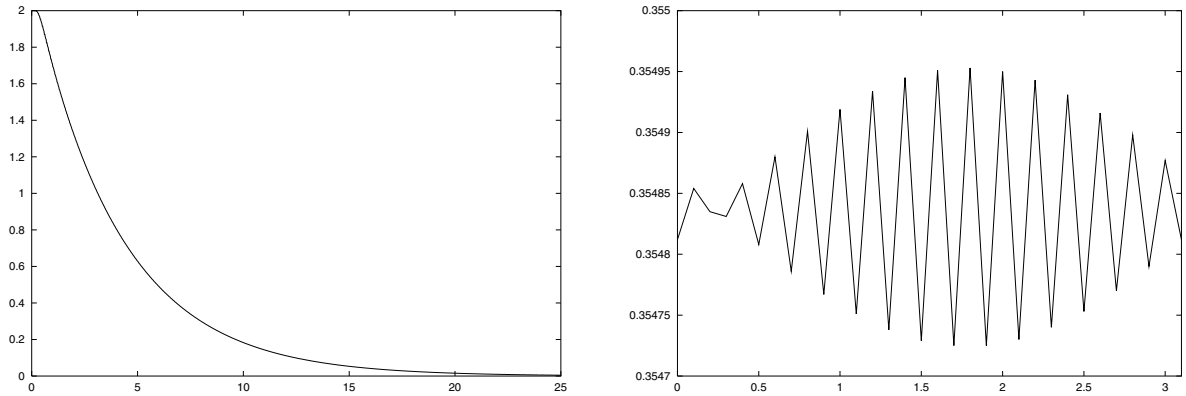

FIGURE 10. The figure shows (a) on the left hand side the evolution of the total variation up to $t = 25$, and (b) on the right hand side the numerical solution at $t = 25$.

how exactly the solution will behave although many clues are given within the last two sections. Thus, we seek a general procedure which makes it easy to apply the Lax–Friedrichs method even in cases of rather complicated fluxes, initial conditions, etc. Exactly this general procedure is given *via* the following

Algorithm 3.1. *Herewith, we define the following procedure:*

(1) *Choose the computational domain and give a discretization with uniform cells.*
(2) *Discretize your initial condition in the normal fashion (e.g. via cell averages) using this mesh.*
(3) *Divide every cell of your computational domain in two cells. These two cells inherit the cell average of the cell they originate from.*
(4) *Discretize the boundary conditions in a way which is in accordance with the 2-cell structure given by step 3 of the algorithm.*

The Algorithm 3.1 yields a discretization where we have only 2-cell structures for the values together with a computational domain with an even number of cells as well as appropriate numerical boundary conditions. Obviously, the 2-cell structure of values will not be destroyed during any time step. It is easy to prove the following

Theorem 3.2. *The Algorithm 3.1 completely specifies the LF-Operator as local (and total) variation preserving. If the Lax–Friedrichs method is set up in this fashion, the number of extrema will not increase during a computation.*

Proof. Both assertions of the theorem are obviously equivalent to the statement that no oscillation will arise during a computation.

For the proof of this statement, it is sufficient to consider an arbitrarily chosen but fixed Riemann problem. Let us choose the values U_l and U_r, $U_l \neq U_r$, and let us consider the evolution of the data at the jump. Since we have a 2-cell structure by Algorithm 3.1, the values left and right of any jump are given by

$$U_{ll}, U_l \text{ and } U_r, U_{rr} \quad \text{with} \quad U_{ll} = U_l \text{ and } U_r = U_{rr}.$$

This yields

$$U_l^{\text{new}} = \frac{1}{2}(U_r + U_l) - \frac{\lambda}{2}(f(U_r) - f(U_l)) = U_r^{\text{new}},$$

so that it is not possible to obtain an oscillation at any jump since the values at any jump are always equalized. □

By Theorem 3.2 and Algorithm 3.1, the Lax–Friedrichs method is a generalized monotone method in the sense of LeFloch and Liu [5].

4. A non-linear test case

In order to show the usefulness of the proceeding developed within the previous section, we apply Algorithm 3.1 on a non-linear problem. We also discuss the usual way to apply the Lax–Friedrichs scheme on this problem in some detail.

Our test problem in this section is concerned with Burgers equation

$$u_t + \left(\frac{1}{2}u^2\right)_x = 0$$

supplemented by the initial condition

$$u_0(x) = \begin{cases} \frac{x-10}{2} & : \quad 10 \le x \le 12 \\ 0 & : \quad \text{else} \end{cases} .$$

The exact solution is the special N-wave

$$N(x,t) = \begin{cases} \frac{x-10}{t+2} & : \quad 10 \le x \le 10 + \sqrt{2(t+2)} \\ 0 & : \quad \text{else} \end{cases} .$$

Since every solution of Burgers equation decays to an N-wave, this example has a profound meaning, see [1] for a discussion of this and related examples.

For the numerical tests, we employ a computational domain large enough so that the numerical boundary conditions are meaningless.

At first, we discuss the usual means of applying the Lax–Friedrichs scheme. We do this by means of Figure 11.

We observe an oscillation after the first time step due to the local extremum at $x = 12$. The occurrence of this oscillation comes as no surprise by the discussion within the previous sections, as well as it is no surprise that the oscillations spread over the numerical solution.

In contrast, we apply the developed recipe and show analogously to Figure 11 the relevant results within Figure 12. We observe the 2-cell structure already within the initial condition which is conserved during all time steps. It is obvious and no surprise that no oscillations arise.

5. Two or more spatial dimensions

We now investigate the properties of the LF-Operator in higher spatial dimensions. Since the basic phenomenae of interest are clarified after the investigations before, it will be sufficient to discuss an example in the two-dimensional case.

As the underlying equation for our numerical investigations serves the 2-D Burgers equation, *i.e.* the conservation law

$$u_t + f(u)_x + g(u)_y \;=\; 0$$

with $f(u) = \frac{1}{2}u^2 = g(u)$. For the relevant parameters for the numerical tests we always chose $\Delta x = \Delta y = 0.1$ and $\Delta t = 0.04$. It will turn out to be interesting to consider the two-dimensional version of the Lax–Friedrichs scheme as well as a dimensional splitting approach.

At first, we discuss the two-dimensional version of the Lax–Friedrichs scheme for the approximation of two-dimensional conservation laws

$$u_t + f(u)_x + g(u)_y \;=\; 0$$

as it can be found *e.g.* in [3], *i.e.*,

$$\begin{aligned} U_{j,k}^{n+1} \;=\; & U_{j,k}^n + \frac{1}{4}\left(U_{j+1,k}^n - 2U_{j,k}^n + U_{j-1,k}^n\right) - \frac{\lambda_x}{2}\left[f\left(U_{j+1,k}^n\right) - f\left(U_{j-1,k}^n\right)\right] \\ & + \frac{1}{4}\left(U_{j,k+1}^n - 2U_{j,k}^n + U_{j,k-1}^n\right) - \frac{\lambda_y}{2}\left[g\left(U_{j,k+1}^n\right) - g\left(U_{j,k-1}^n\right)\right]. \end{aligned} \tag{16}$$

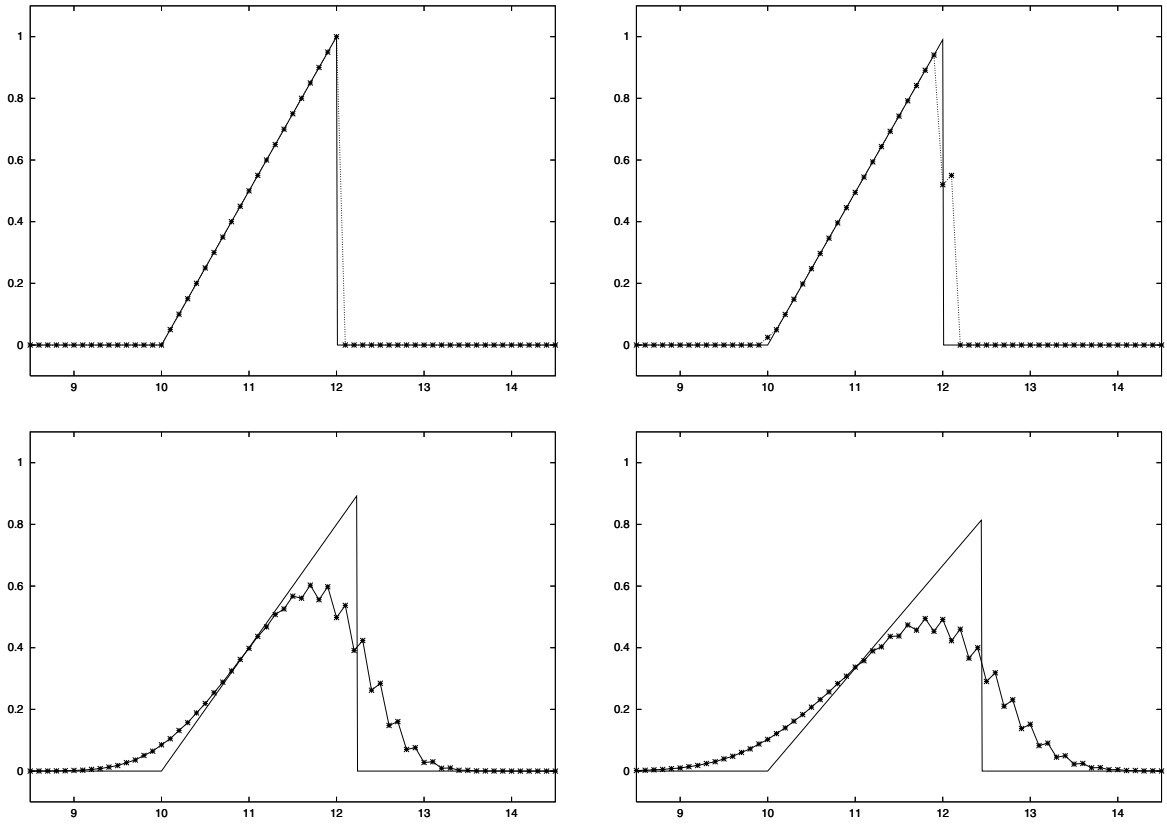

FIGURE 11. This figure shows from left to right and top to bottom: (a) the exact initial condition and its discretization, then the exact solution and the numerical solution after (b) one, (c) 25 and (d) 50 time steps. The snapshots of the exact solutions are always drawn with lines, the numerical solutions are depicted by dots or dotted lines, respectively.

As the most simple test, let us consider the situation analogously to the one defined in [5], *i.e.*,

$$u^0_{j_0,k_0} := 1 \quad \text{and} \quad u^0_{j,k} := 0 \ \text{ for all } \ (j,k) \neq (j_0,k_0).$$

Evidently, for general fluxes f and g with $f(0) \equiv g(0) \equiv 0$ holds $u^1_{j_0,k_0} = 0$ by (16), while the quantity u is distributed over the four cells adjacent to the cell (j_0,k_0).

Thus, the appearance of oscillations (or, more generally, of new local extrema) found in the 1-D case carries over to (16) and obviously also to any higher-dimensional analog on of (16).

As a more sophisticated numerical test example for this behaviour, we consider as indicated the 2-D Burgers equation. As initial condition, we first take a block signal of 3×3 cells with the value 1 while every other value is 0, see Figure 13.

Two typical numerical results of two subsequent time steps are displayed in Figure 14. One can especially observe the two-dimensional structure of the oscillations at the peak as well as within the levels of the numerical approximation running like rings around the peak.

One may think that it is possible to employ a two-dimensional version of Algorithm 3.1 in order to construct a solution featuring a structure of 2×2-cells, and thus to avoid such oscillations.

However, a quick investigation of the scheme (16) when considering the most simple test employing a block signal of 2×2 cells shows that this is not the case, see Figure 15. We observe that the 2×2-cell structure of the

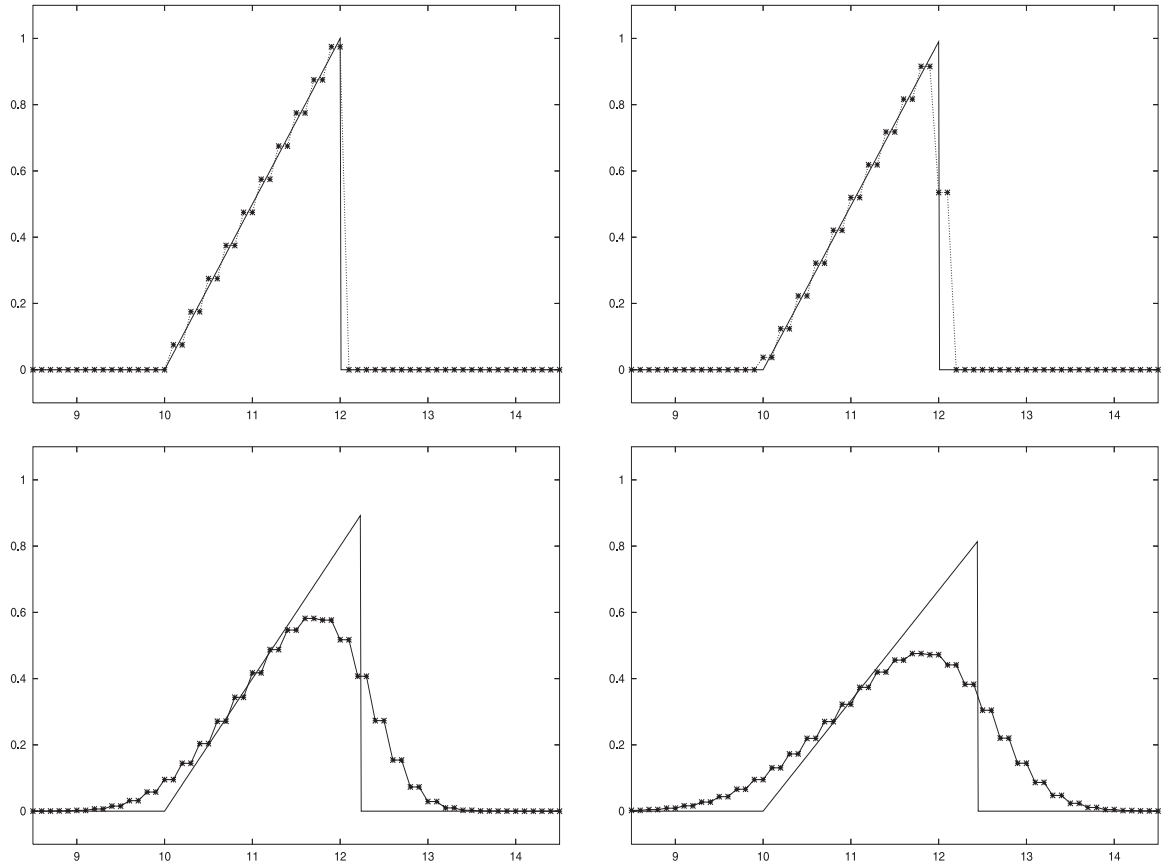

FIGURE 12. This figure shows from left to right and top to bottom: (a) the exact initial condition and its discretization, then the exact solution and the numerical approximation obtained using our recipe after (b) one, (c) 25 and (d) 50 time steps. The snapshots of the exact solutions are always drawn with lines, the numerical solutions are depicted by dots or dotted lines, respectively.

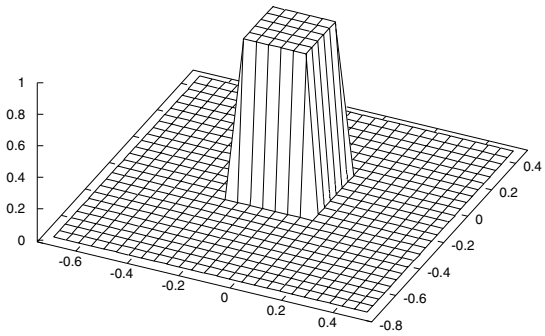

FIGURE 13. The figure shows the initial condition of our 2-D test case.

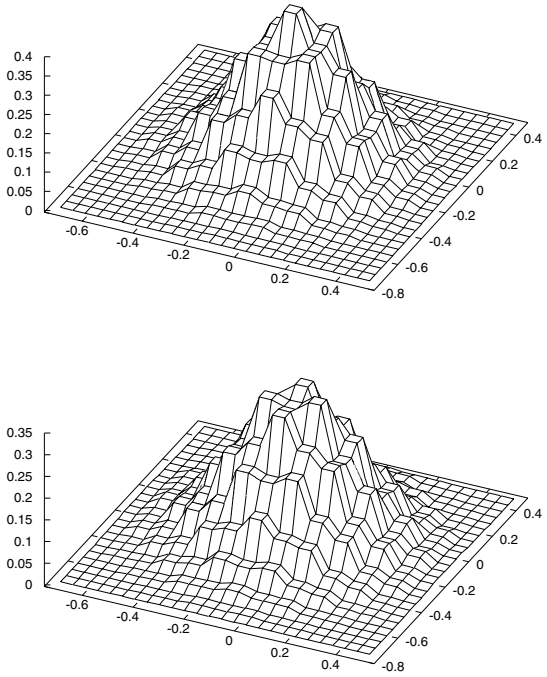

FIGURE 14. The figure shows two typical numerical solutions of two subsequent time steps. The oscillations are clearly observable.

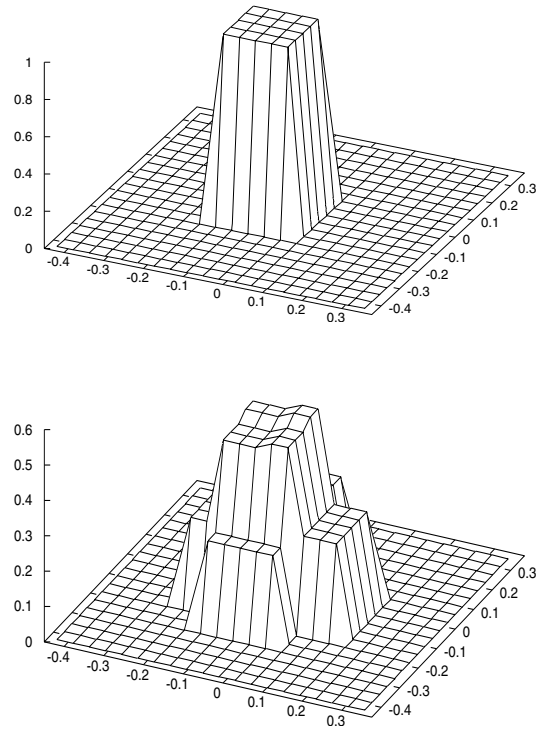

FIGURE 15. The figure shows (a) at the top the 2×2 block signal serving as initial condition as well as (b) at the bottom the numerical solution after one time step.

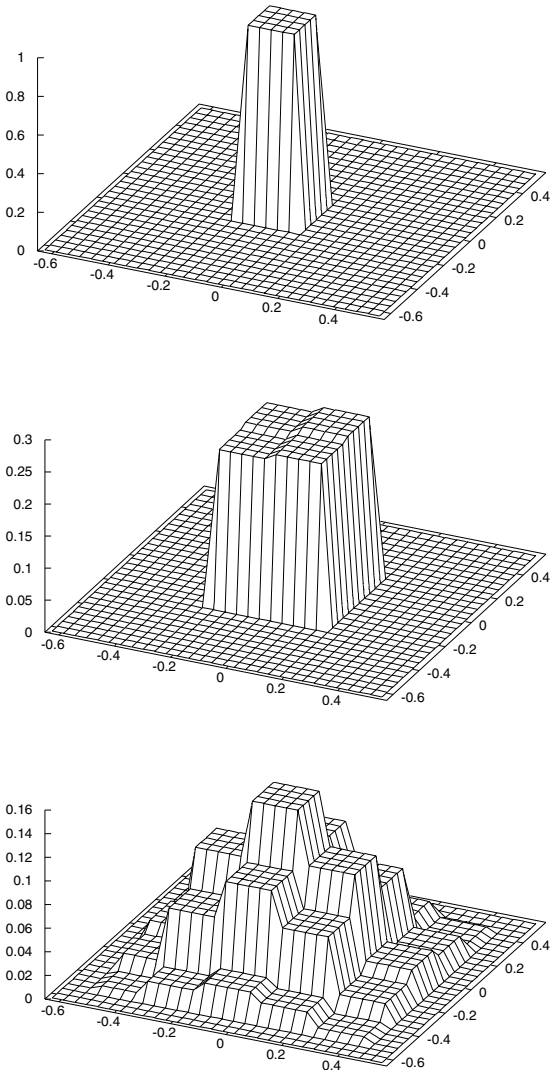

FIGURE 16. The figure shows (a) at the top the 2×2 block signal serving as initial condition, (b) in the middle the numerical solution after one time step and (c) at the bottom the numerical solution after four time steps.

initial condition is not carried over; however, we also do not observe an oscillation in the course of the further computations concerning this particular example.

Our tests conjecture that an initial discretization using such 2×2-cells in the two-dimensional case yields solutions without oscillations; however, up to now this assertion could not be verified rigorously.

If we employ the usual dimensional splitting approach as described *e.g.* in [7], the situation is different: Since all directions are treated separately in the fashion of a one-dimensional problem, the 2-cell structure we employed in the preceding sections is recovered in every step of the splitting. Thus, initial conditions featuring a 2×2-cell structure in 2-D yield solutions showing again this structure. Numerically, we validate this statement for the example just employed before, see Figure 16. The 2×2-cell structures are clearly observable.

For the dimensional splitting approach it is obvious that the Algorithm 3.1 makes sense in the form corresponding to the number of spatial dimensions under consideration. Note also, that the splitting error is often not very large in practical computations, see *e.g.* [7] for a discussion.

6. Conclusive remarks

We have investigated the data dependence of the structure of the LF-Operator in detail. All illuminated properties of the LF-Operator were validated numerically.

The investigation lead to the development of Algorithm 3.1 which allows the application of the Lax–Friedrichs method with no irritations. This was confirmed by the use of several linear and non-linear tests in one and two spatial dimensions.

The basic phenomenae encountered when using the Lax–Friedrichs method are clarified. It remains *e.g.* to investigate the effects of using a non-uniform grid on the structure of the corresponding LF-Operator. It is also an important open question if the observed properties of the LF-Operator are inherited by other schemes which were constructed using the staggered grid approach, as *e.g.* the Nessyahu-Tadmor scheme or other more modern central schemes.

Acknowledgements. The author would also like to thank Professor Thomas Sonar for his encouragement.

The author has not corrected the proofs of the article.

References

[1] L. Evans, *Partial Differential Equations.* American Mathematical Society (1998).
[2] E. Godlewski and P.-A. Raviart, *Hyperbolic systems of conservation laws.* Ellipses, Edition Marketing (1991).
[3] E. Godlewski and P.-A. Raviart, *Numerical Approximation of Hyperbolic Systems of Conservation Laws.* Springer-Verlag, New York (1996).
[4] P.D. Lax, Weak solutions of nonlinear hyperbolic equations and their numerical approximation. *Comm. Pure Appl. Math.* **7** (1954) 159–193.
[5] P.G. LeFloch and J.-G. Liu, Generalized monotone schemes, discrete paths of extrema, and discrete entropy conditions. *Math. Comp.* **68** (1999) 1025–1055.
[6] R.J. LeVeque, *Numerical Methods for Conservation Laws.* Birkhäuser Verlag, 2nd edn. (1992).
[7] R.J. LeVeque, *Finite Volume Methods for Hyperbolic Problems.* Cambridge University Press (2002).
[8] H. Nessyahu and E. Tadmor, Non-oscillatory central differencing for hyperbolic conservation laws. *J. Comput. Phys.* **87** (1990) 408–436.