Abstract

In this paper, a novel technique of obtaining high resolution, second order accurate, oscillation free, solution dependent weighted least-squares (SDWLS) reconstruction in finite volume method is explored. A link between the weights of the weighted least-squares based gradient estimation and various existing limiter functions used in variable reconstruction is established for one-dimensional problems for the first time. In this process, a class of solution dependent weights are derived from the link which is capable of producing oscillation free second order accurate solutions for hyperbolic systems of equations without the use of limiter function. The link also helps in unifying various independently proposed limiter functions available in the literature. The way to generate numerous new limiter functions from the link is demonstrated in the paper. An approach to verify TVD criterion of the SDWLS formulation for different choice of weights is explained. The present high resolution scheme is then extended to solve multi-dimensional problems with the interpretation of weights in SDWLS as influence coefficients. A few numerical test examples involving one- and two-dimensional problems are solved using three different new limiter functions in order to demonstrate the utility of the present approach.

Keywords: Higher-order reconstruction; Limiters; TVD; Solution dependent weighted least-squares; SDWLS; Finite volume method; Hyperbolic conservation laws

1. Introduction

Discontinuous solutions are often encountered in the solutions of hyperbolic conservation laws. In order to resolve the discontinuities in the flow field, it is important to use higher order schemes. However, classical higher-order schemes, whilst giving high resolutions to discontinuities of the solutions, exhibits spurious oscillations around such locations. Whereas, any monotone scheme [11], that guarantees non-oscillatory solution, can only be first-order accurate when a linear approach is used. Therefore, 'weak monotone concept' [16], using non-linear limiter schemes, is introduced to limit anti-diffusive terms present in the high accuracy schemes [20]. Many limiters have been proposed in the past, which follow either Harten’s modified flux approach [10] or van Leer’s MUSCL [24] reconstruction to obtain higher order accuracy. Such schemes typically make use of one additional point beyond nearest neighbors in
their difference stencils, which is natural in case of structured meshes. However, in unstructured meshes, consideration of additional points to construct slopes for higher-order (or anti-diffusive) part may cause substantial complexity into the method and additional storage requirement [1]. The unstructured grid computations usually resort to either Green’s theorem approach [4] and its variant [8] or least-squares approach [2] for estimation of the slope within each control volume, and thus higher-order accuracy is obtained without informal inquiries beyond the nearest neighbors [1]. It has been reported by Aftosmis et al. [1] that the least-squares gradient estimation provides significantly more reliable results in most situations.

Several least-squares or weighted least-squares reconstruction techniques for solving hyperbolic conservation laws are reported in the literature [2,15,3,5]. In all these methods, the slope- or flux-limiters are applied directly to the anti-diffusive parts in order to obtain non-oscillatory solutions. The weights used for weighted least-squares methods, as reported in the literatures, are usually based on geometric (distance) informations [2,15]. Although, the possibility of solution-dependent weights are indicated by Barth [2], there is no mention about the way to choose (the expression of) the weights for TVD (non-oscillatory) solutions. To our knowledge, development of solution dependent weights for TVD (non-oscillatory) solution with higher order least-squares based reconstruction has not been reported in literature.

In this paper, a link between the solution dependent weights for weighted least-squares formulation and various independently proposed limiter functions reported in the literature is discovered. This link helps in implementing weighted least-squares reconstruction directly into a higher-order scheme without needing limiter function. The implementation of the present approach thus becomes simple and computationally efficient. It is also shown for the first time, how several independently proposed limiter functions available in the literature belongs to a class of solution dependent weighted least-squares (SDWLS) formulations. The way to generate innumerable limiter functions for non-oscillatory solutions are proposed by extending the same link further. It is also demonstrated in the paper, the means of establishing TVD criterion for the SDWLS formulation in 1D, given a particular type of weight.

The paper is organized in the following manner. After a brief introduction of the finite volume formulation for 1D in Section 2, the conventional reconstruction procedure with the application of limiter functions and weighted least-squares reconstruction procedure are discussed in Section 3 and Section 4, respectively. The link between the solution dependent weighted least-squares and limiter functions has been elaborated [14,19] in Section 5 and Appendix A. The idea of SDWLS formulation has been introduced in the same section. The limiter functions are classified into three categories by parametric representation of the weights in Section 6. The generation of new weights leading to new limiter functions and the means of verifying TVD criterion of the new weights and the corresponding new limiter functions is explained in Section 7. Having established the above link and its related theory for one-dimensional problems, extension of the present formulation to multi-dimensions has been presented in Section 8. Finally, several standard 1D and 2D test problems are solved with the present SDWLS formulation in order to demonstrate the potential of the present approach.

2. Finite volume formulation

Starting with a general form of a scalar conservation law in one-dimension

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0$$

its integral form can be written as

$$\int_{\Omega} \left[ \frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} \right] dx = 0$$

where $u$ is the conserved variable, $f$ is the flux of this conserved quantity through the cell interfaces and $\Omega$ is the control volume (line in a 1D case). The domain can be discretized as shown in Fig. 1.

The cell centered finite volume discretization of the above integral form over the domain $[x_{i+1/2}, x_{i-1/2}]$, as shown in Fig. 1, can be expressed as

$$u_{i}^{n+1} = u_{i}^{n} - \frac{\Delta t}{\Delta X} \left[ f_{i+1/2}^{n} - f_{i-1/2}^{n} \right]$$
where, $n$ is the time level, $i$ is the cell index, $\Delta t$ is the time step and $\Delta X$ is the length of the $i$-th cell as shown in Fig. 1. The computation of $f$ at the interfaces $i + 1/2$ and $i - 1/2$ requires computation of $u$ at these interfaces, which is explained in the following section.

3. Conventional second-order schemes

It is well known that first order upwind schemes being monotone ensure non-oscillatory solutions. However, they are diffusive in nature. Therefore, higher order schemes are developed with the help of variable reconstruction in finite volume method. Writing the most commonly used linear reconstruction expression for the variable $u$ at the cell interface $i + 1/2$ (shown in Fig. 1) as,

$$u_{i+1/2} = u_i + \left( \frac{du}{dx} \right)_i \frac{\Delta x_{i+1/2}}{2}$$

results in a second order scheme (with the first term alone in the right hand side leads to a first order scheme). The quantity $\Delta x_{i+1/2}$ is the distance from the cell center to the interface. The above reconstruction (4) is written only for the positive variable (i.e. variable to be used in the positive flux computation). Without loss of generality, we shall consider only the positive part throughout this paper. The reconstruction of the negative part can be easily derived based on similar lines.

Sweby [20] describes second order schemes as perturbed first order schemes, with the perturbation being interpreted as the anti-diffusive part. Since the anti-diffusive part is responsible for oscillations, its contribution needs to be limited for monotone solutions. The modified expression for the variable reconstruction at the cell interface is thus given by,

$$u_{i+1/2} = u_i + \psi \left( \frac{du}{dx} \right)_i \frac{\Delta x_{i+1/2}}{2}$$

where $\psi$ is the limiter function, typically constructed using adjacent variable differences ($\Delta u_1$ and $\Delta u_2$ for a 3 point stencil as shown in Fig. 1). Thus, the final limited anti-diffusive part is a product of two terms—the gradient and the limiter. It is important to note that the first order part for all upwind schemes is similar (with respect to the choice of stencil). The computation of the gradient, required in the higher order (anti-diffusive) part, and the subsequent limiting procedure, essentially distinguish various schemes from each other. In this paper, we look at weighted least squares approximation (in Section 4) as one of the methods to evaluate the gradient. This approach too usually requires limiting of the higher order part. However, we show in later sections that the weighted least squares formulation with judicious choice of weights (with weights dependent on solution) does not require application of limiters.

4. Weighted least squares formulation

For simplicity, we restrict our discussion to conventional weighted least squares formulation in one dimension. Traditionally, estimation of gradients at point $i$ involves solution of a system of equations formed by the truncated Taylor series expansion, which is given as,

$$u_j = u_i + (x_j - x_i) \left( \frac{du}{dx} \right)_i + \frac{1}{2} (x_j - x_i)^2 \left( \frac{d^2 u}{dx^2} \right)_i + \cdots,$$

Fig. 1. 1D Stencil.
where \( j = i + 1 \) or \( i - 1 \) (for a 3 point stencil shown in Fig. 1) or any other neighbouring point. The number of terms to be retained in the above expansion depends on the accuracy and the order of the derivatives required in the differential equation. The system of equations obtained by applying the above expression (keeping only two terms on the right hand side) to the points in the stencil (see Fig. 1) is given as,

\[
\begin{align*}
    u_{i+1} &= u_i + \Delta x_2 \left( \frac{du}{dx} \right)_i, \\
    u_{i-1} &= u_i - \Delta x_1 \left( \frac{du}{dx} \right)_i
\end{align*}
\]

which can be written in a matrix form as,

\[
\begin{bmatrix}
    \Delta u_1 \\
    \Delta u_2
\end{bmatrix} = \begin{bmatrix}
    \Delta x_1 \\
    \Delta x_2
\end{bmatrix} \begin{bmatrix}
    \left( \frac{du}{dx} \right)_i
\end{bmatrix}.
\]

Writing the above equation in vector form,

\[
\Delta \mathbf{u} = \mathbf{S} \mathbf{d} \mathbf{u},
\]

where \( \mathbf{d} \mathbf{u} = \left[ \left( \frac{du}{dx} \right)_i \right] \), \( \Delta \mathbf{u} = \left[ \Delta u_1, \Delta u_2 \right]^T \) and \( \mathbf{S} = \left[ \Delta x_1, \Delta x_2 \right]^T \). The terms \( \Delta u_1, \Delta u_2, \Delta x_1 \) and \( \Delta x_2 \) are explained in Fig. 1. The above equations (9) being an over-determined system, least squares technique [7,2] to approximate the unknown gradient vector is suggested. Here, we have chosen a 3 point stencil for estimating one derivative. In brief, this approach involves minimization of the approximation error, which yields the following equation,

\[
\mathbf{S}^T \Delta \mathbf{u} = \mathbf{S}^T \mathbf{S} \mathbf{d} \mathbf{u}.
\]

The above equation is solved for the unknown gradient vector as given below,

\[
\mathbf{d} \mathbf{u} = (\mathbf{S}^T \mathbf{S})^{-1} \mathbf{S}^T \Delta \mathbf{u}.
\]

In practice, grids are usually non-uniform. The least squares method distributes the total approximation error in derivative computation uniformly among all the points in the stencil, even in a non-uniform mesh. In order to incorporate differences in the influence of each neighbouring point on the gradient (which is usually based on distance from the node), the least squares method needs to be modified. This was achieved by modifying equation (11) with the introduction of a diagonal matrix (called weight matrix) to yield the weighted least squares approximation [7],

\[
\mathbf{d} \mathbf{u} = (\mathbf{S}^T \mathbf{W} \mathbf{S})^{-1} \mathbf{S}^T \mathbf{W} \Delta \mathbf{u}.
\]

The weights (diagonal entries of \( \mathbf{W} \)) are normally geometric quantities, typically of the form \( w_i = 1/(|x_i - x_0|^a) \) with \( a \) taking the values 0, 1, 2 [2] (\( a = 0 \) case corresponds to conventional least squares). However, Barth [2] has proposed the possibility of using also solution dependent weights. To our knowledge, use of solution dependent weights for TVD (non-oscillatory) solutions has not been explored in literature till date. In case of 1D mesh with 3-point stencil (Fig. 1), Eq. (12) can be used to obtain an explicit formula for the required gradient as,

\[
\left( \frac{du}{dx} \right)_i = \frac{w_1 \Delta x_1 \Delta u_1 + w_2 \Delta x_2 \Delta u_2}{(w_1 \Delta x_1^2 + w_2 \Delta x_2^2)}.
\]

In order to facilitate comparison with other schemes proposed in literature, we henceforth consider only uniform grids for analysis. However, it may be noted that this approach naturally extends to non-uniform grids and can handle arbitrary number of data points as well. On the uniform mesh, \( \Delta x_1 = \Delta x_2 = \Delta x \); hence, the expression (13) reduces to,

\[
\left( \frac{du}{dx} \right)_i = \frac{w_1 \Delta u_1 + w_2 \Delta u_2}{(w_1 + w_2) \Delta x}.
\]

The weights in Eq. (14) above can also be interpreted as a means to assign influence coefficients to each neighbouring point in the stencil. This interpretation is possible as it can be seen that the weights \( w_1 \) and \( w_2 \) controlling the contributions of the two differences \( \Delta u_1 \) and \( \Delta u_2 \) in the gradient estimation. The interpretation of weights as influence coefficients will help us extending the present approach to non-uniform grids and multi-dimensional computations.
later. Substituting the expression of this gradient from (14) into the expression (4), the second order reconstructed
value at the interface becomes,

\[ u_{i+1/2} = u_i + \frac{w_1 \Delta u_1 + w_2 \Delta u_2}{(w_1 + w_2) \Delta x} \Delta x_{i+1/2} = u_i + \frac{1}{2} \frac{w_1 \Delta u_1 + w_2 \Delta u_2}{(w_1 + w_2)} \]

as \( \Delta x_{i+1/2} = \Delta x/2 \) for a uniform grid. In subsequent sections, the choice of weights to obtain non-oscillatory solu-
tions is explained.

5. Link between weighted least squares and limiter functions: development of solution dependent weighted least
squares (SDWLS) formulation

In order to choose the expressions for the weights in the least squares formulation, we study the relation between
them and the limiter functions used in TVD schemes. At first, we consider a popularly used limiter function such as
the van Albada limiter [21] for illustrative purpose. The variable at the interface \( i + 1/2 \) with linear reconstruction (5)
and the van Albada limiter on the 3 point stencil (Fig. 1) can be written as (see Appendix A for details)

\[ u_{i+1/2} = u_i + \frac{1}{2} \psi \Delta u_1 = u_i + \frac{1}{2} \frac{(\Delta u^2_1) \Delta u_1 + (\Delta u^2_1) \Delta u_2}{\Delta u_1 (\Delta u^2_1 + \Delta u^2_1)} \Delta u_1. \]

Now, comparing Eqs. (15) and (16), the two reconstruction formulae become identical with the choice of \( w_1 = \Delta u^2_2 \) and \( w_2 = \Delta u^2_1 \). This illustrates that the TVD condition (using van Albada limiter here) would be directly imposed by the
aforementioned choice of weights. Similarly, it also provides a scope to include other limiters in this framework. In
order to extend this link to include other limiters, the expression (14) needs to be modified by the addition of a term \( p \) as,

\[ \left( \frac{du}{dx} \right)_i = p \frac{w_1 \Delta u_1 + w_2 \Delta u_2}{(w_1 + w_2) \Delta x}. \]

The term \( p \) (which acts like Heaviside-like function) in the above expression serves the purpose of switching between
first order and second order expressions. Since, the weights \( (w_1 \text{ and } w_2) \) here in the weighted least squares formulation
is dependent on solution \( u \), the present formulation is named as solution dependent weighted least squares or SDWLS
in short.

Referring to Eq. (16), expression for the limiter function too can be extracted in terms of weights as follows,

\[ \psi = p \frac{w_1 \Delta u_1 + w_2 \Delta u_2}{(w_1 + w_2) \Delta u_1}. \]

Now, the various limiters given in literature can be expressed in terms of weights as shown in Table 1. It may be noted
that the term \( p \) is also helpful in casting limiter functions such as Minmod, MC etc. in the weighted least squares form,
in which case \( p \) takes values other than 0 and 1 (see Appendix A). Detailed derivations of the weights corresponding
to independently proposed limiter functions in the literature [21,25,22,12,17,23,4] are shown in Appendix A. The
derivations are very similar to the van Albada case (explained above) and hence are not discussed here for the sake of
brevity.

For illustration of use of the table, let us choose the Venkatakrishnan limiter, which has the most complicated
form. For the same limiter function, different forms are identified by assigning a subscript to them, for example \( \psi_1 \)
and \( \psi_2 \) in case of the Venkatakrishnan limiter [25]. Here, based on a certain switching logic (see Appendix A), one
out of two possible forms can be chosen. For \( \psi_1 \) the choice of weights are \( w_1 = 5 \Delta u^2_2 + \ell_1 \Delta u^2_1 + \Delta u_1 \Delta u_2 \) and
\( w_2 = 6 \Delta u^2_2 + (1 + \ell_2) \Delta u^2_1 + 3 \Delta u_1 \Delta u_2 \). Similarly, the various forms of other limiter functions too are represented in
the table.

6. Parametrization of weights and interpretation

In this section, a framework to unify all weights is developed based on the link, between the weights and the limiter
functions, established earlier. The advantage of this framework is that it can be used to derive weights (influence
coefficients) for points in stencils having more than 3 points and also for multi-dimensional stencils as will be shown in Section 8. The expression in this framework, also referred to as the divided form, is given as,

\[ w_\alpha = \sum_{m=1}^{N} a_m \prod_{\beta=1}^{N_v} |\Delta u_\beta|^{\gamma_\alpha,\beta}, \]  

(19)

where \( \alpha \) denotes the index of the neighbouring stencil point, for which the weight (influence coefficient) is \( w_\alpha \), \( N \) is the number of terms appearing in the expression for the weight, \( N_v \) is the number of neighbouring points (i.e. points in the stencil other than the one at which the gradient is being estimated), \( a_m \) and \( \gamma_\alpha,\beta \) are numeric constants.

The expression (19) can be written in a simpler form for the three point stencil shown in Fig. 1 as,

\[ w_\alpha = \sum_{m=1}^{N} \frac{a_m}{|\Delta u_1|^{\gamma_\alpha,1} |\Delta u_2|^{\gamma_\alpha,2}}, \]  

(20)

where \( \alpha = 1 \) or 2 (corresponding to \( i-1 \) or \( i+1 \) grid points respectively) for the 3-point stencil. Considering an example of one of the limiters, say van Albada limiter, we get \( w_1 = \frac{1}{\Delta u_2^2} \) and \( w_2 = \frac{1}{\Delta u_1^2} \), as the divided forms of the earlier weights (in Table 1). Referring to Eq. (20), the different parameters are \( N = 1 \), \( a_1 = 1 \), \( \gamma_1,1 = 2 \), \( \gamma_1,2 = 0 \), \( a_2 = 1 \), \( \gamma_2,1 = 0 \) and \( \gamma_2,2 = 2 \). All the weights given in Table 1 in the previous section can now be expressed as a choice of the three parameters in the divided form as illustrated in Table 2 (see detailed derivations in Appendix A).

The divided form also provides a new means to categorize limiter functions based on the parameters \( a \) and \( \gamma \). Observing the weights for various limiter functions from Tables 1 and 2, we arrive at three distinct categories of limiter functions, namely

---

### Table 1

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Limiter</th>
<th>Form</th>
<th>( w_1 )</th>
<th>( w_2 )</th>
<th>( p )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>van Albada</td>
<td>( \psi )</td>
<td>( \Delta u_2^2 )</td>
<td>( \Delta u_1^2 )</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>Venkatakrishnan</td>
<td>( \psi_1 )</td>
<td>( (\varepsilon_1/2) \Delta u_2^2 + 5 \Delta u_2^2 + \Delta u_1 \Delta u_2 )</td>
<td>( (1 + \varepsilon_1/2) \Delta u_1^2 + 6 \Delta u_2^2 + 3 \Delta u_1 \Delta u_2 )</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \psi_2 )</td>
<td>( (6 + \varepsilon_1/2) \Delta u_2^2 + \Delta u_2^2 + 3 \Delta u_1 \Delta u_2 )</td>
<td>( (5 + \varepsilon_1/2) \Delta u_1^2 + \Delta u_1 \Delta u_2 )</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>van Leer</td>
<td>( \psi )</td>
<td>(</td>
<td>\Delta u_2</td>
<td>)</td>
</tr>
<tr>
<td>4</td>
<td>Minmod</td>
<td>( \psi_1 )</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \psi_2 )</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>Superbee</td>
<td>( \psi_1 )</td>
<td>1</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \psi_2 )</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \psi_3 )</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \psi_4 )</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>MC</td>
<td>( \psi_1 )</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \psi_2 )</td>
<td>1</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \psi_3 )</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>Barth and Jesperson, Central</td>
<td>( \psi_1 )</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \psi_2 )</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \psi_3 )</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \psi_4 )</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>Barth and Jesperson, Upwind</td>
<td>( \psi_1 )</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \psi_2 )</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>Barth and Jesperson, Downwind</td>
<td>( \psi_1 )</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>( \psi_2 )</td>
<td>1</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>
Table 2
Modified expression for weights in divided form

<table>
<thead>
<tr>
<th>S. Lim.</th>
<th>Form</th>
<th>$p$</th>
<th>$w_1$</th>
<th>$w_2$</th>
<th>$m = 1$</th>
<th>$m = 2$</th>
<th>$m = 3$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$\alpha$</td>
<td>$\gamma_1$</td>
<td>$\gamma_2$</td>
</tr>
<tr>
<td>1</td>
<td>vA</td>
<td>$\psi$</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>VK</td>
<td>$\psi_1$</td>
<td>1</td>
<td>5</td>
<td>6</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>0</td>
<td>1 + $\epsilon_1/2$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$\psi_2$</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>6 + $\epsilon_1/2$</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>vL</td>
<td>$\psi$</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>Mm</td>
<td>$\psi_1$</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$\psi_2$</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>Sb</td>
<td>$\psi_1$</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$\psi_2$</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$\psi_3$</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$\psi_4$</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>MC</td>
<td>$\psi_1$</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$\psi_2$</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$\psi_3$</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>BJ_C</td>
<td>$\psi_1$</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$\psi_2$</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$\psi_3$</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$\psi_4$</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>BJ_U</td>
<td>$\psi_1$</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$\psi_2$</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>BJ_D</td>
<td>$\psi_1$</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$\psi_2$</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Abbreviations: vA = van Albada, VK = Venkatakrishnan, vL = van Leer, Mm = Minmod, Sb = Superbee, BJ_C = Barth and Jesperson, Central, BJ_U = Barth and Jesperson, Upwind and BJ_D = Barth and Jesperson, Downwind.

1. **Logical Type**: From Table 1, we find that weights corresponding to certain limiter functions (such as Minmod), do not depend on the value of the variable being reconstructed. In terms of the coefficients given in expression (20), $\gamma = 0$ for these limiter functions. Further, it can be seen that the weights corresponding to such limiter functions are either 0 or 1. Hence, this category is named as the Logical type. Examples of other limiter functions belonging to the Logical type are Superbee, MC etc.
(2) **Monomial Type:** The weights corresponding to some limiter functions such as the van Albada and the van Leer limiters depend only on (the value of the variable being reconstructed at) the central point and the neighbouring stencil point for which the weight is being evaluated, which can be inferred from Table 2. Also, for these limiter functions, the polynomial weight expression (20) reduces to a monomial (i.e. $N = 1$) and the denominator too effectively has a single contributing term in the product (other terms reducing to 1, i.e. $\gamma_{m,\alpha} = 0$ if $\alpha \neq \beta$). Since, the entire expression (20) reduces to a single term, with the denominator too reducing to a single term, this category is named as the Monomial type.

(3) **Polynomial Type:** This final category of limiter functions encompasses all limiter functions that do not fall into the Logical and Monomial categories. Since the weights corresponding to this category of limiter functions, when cast in the form (20) have at least two contributing terms, either in the summation, or in the product in the denominator, this category is termed the Polynomial type.

7. New weight combinations and generation of new limiter functions

In this section, the utility of the SDWLS approach in generating new limiter functions is demonstrated by constructing three new limiter functions, one from each category (described in Section 6). The new limiter functions are required to lie in the Sweby region in order to be TVD [20]. The Sweby region plot for the corresponding three limiter functions is verified (see Fig. 2).

The limiter functions corresponding to each of the weight choices in Table 3 are derived by using the expressions (18), (20) for the link developed in Section 5, as,

<table>
<thead>
<tr>
<th>S</th>
<th>Form</th>
<th>$u_{\alpha}$</th>
<th>$m = 1$</th>
<th>$m = 2$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>$a_{\alpha}$</td>
<td>$\gamma_{m,1}$</td>
</tr>
<tr>
<td>1</td>
<td>$\psi_1$</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$w_1$</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$w_2$</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$\psi_2$</td>
<td>1.2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$w_1$</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$w_2$</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$\psi_3$</td>
<td>1.2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$w_1$</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$w_2$</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>$\psi$</td>
<td>1</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>$w_1$</td>
<td>1</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>$w_2$</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>$\psi$</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>$w_1$</td>
<td>2</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>$w_2$</td>
<td>2</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>

*Note: $p = 0$ whenever adjacent differences in flow variable have opposite sign, that is, whenever $r < 0$ for all limiter functions.*
New Limiter-1: \( \psi(r) = \min \text{mod}(1 + r, 1.2, 1.2r) \) (logical type),
New Limiter-2: \( \psi(r) = \frac{r^{2} + r}{r^{2} + 1} \) (monomial type),
New Limiter-3: \( \psi(r) = \frac{r^{2} + 3r}{r^{2} + r + 2} \) (polynomial type).

Innumerable new limiter functions can thus be generated by different choices of weights.

8. Extension of SDWLS to two dimensions

In this section, the extension of the SDWLS scheme to two dimensions is presented. Fig. 3 shows a typical 2D stencil involving triangular grids, commonly used in two-dimensional unstructured grid computations. Since, in cell centered finite volume formulation, the values of the field variable are available only at the cell centers, the evaluation of numerical fluxes at the interfaces requires solution reconstruction. In an upwind approach, higher order reconstruction of any variable \( u \) at the left and right of cell interface of cells \( i \) and \( j \) is given by,

\[
\begin{align*}
\hat{u}_{ij}^{L} &= u_{i} + \left( \frac{\partial u}{\partial x} \right)_{i} \Delta x^{L} + \left( \frac{\partial u}{\partial y} \right)_{i} \Delta y^{L}, \\
\hat{u}_{ij}^{R} &= u_{j} + \left( \frac{\partial u}{\partial x} \right)_{j} \Delta x^{R} + \left( \frac{\partial u}{\partial y} \right)_{j} \Delta y^{R}
\end{align*}
\]

where, \( \Delta x^{L}, \Delta y^{L} \) are the differences in the \( x \) and \( y \) coordinates between the cell interface midpoint and the left cell center \( i \) (refer to Fig. 3). Similarly, \( \Delta x^{R}, \Delta y^{R} \) are the differences in the \( x \) and \( y \) coordinates between the cell interface midpoint and the right cell center \( j \). The estimation of gradients (appearing in the reconstruction expressions above) at the cell centroid, \( i \), using least squares formulation utilizes the overdetermined system,

\[
\begin{bmatrix}
\Delta u_{1} \\
\Delta u_{2} \\
\Delta u_{3}
\end{bmatrix} = \begin{bmatrix}
\Delta x_{1} & \Delta y_{1} \\
\Delta x_{2} & \Delta y_{2} \\
\Delta x_{3} & \Delta y_{3}
\end{bmatrix} \begin{bmatrix}
\frac{\partial u}{\partial x} \\
\frac{\partial u}{\partial y}
\end{bmatrix}_{i}
\]

where, \( \Delta u_{1} = u_{j} - u_{i}, \Delta u_{2} = u_{k} - u_{i}, \Delta u_{3} = u_{l} - u_{i}, \Delta x_{1} = x_{j} - x_{i}, \Delta x_{2} = x_{k} - x_{i}, \Delta x_{3} = x_{l} - x_{i}, \Delta y_{1} = y_{j} - y_{i}, \Delta y_{2} = y_{k} - y_{i} \) and \( \Delta y_{3} = y_{l} - y_{i} \). An identical approach can be used for the gradient estimation at the cell center \( j \); hence not elaborated separately.

Similar to the approach followed in Section 4 for 1D, the unknown gradients can then be obtained from the above system (23) using weighted least squares formulation [7] as

\[
\begin{align*}
\mathbf{d}u &= (S^{T}WS)^{-1}S^{T}W\Delta u.
\end{align*}
\]
It may be noted that any conventional higher order reconstruction requires the antidiffusive terms in (21), (22) to be multiplied by limiter function \( \psi \) in order to achieve non-oscillatory solution. Since, such a TVD scheme in multi-dimensions are at most first order accurate [9], the extension of TVD schemes to multi-dimensions are usually carried out in a dimension-by-dimension manner [6]. However, such dimensional splitting is not possible for unstructured grids. Consider, for instance, triangular cells in an unstructured grid. Each cell has three nearest face neighbours, but the usual definition of limiter functions can accommodate only two of them at a time. There has been no clarity till date on the issue of how to use all the data points involving the neighbours to arrive at a single limiter function. The above problems in TVD theory, therefore make limiting a problematic step in unstructured grid computations.

In the present formulation, we use solution dependent weights to obtain non-oscillatory solutions, without the use of limiter functions. The linear reconstruction expressions (21), (22) are used directly without any modification. We obtain the expression of weights in 2D using the interpretation of weights as influence coefficients as explained in Section 6. The link between weighted least squares formulation and limiter function thus provides an alternative means of obtaining non-oscillatory solutions. Though, these influence coefficients are shown to make the scheme TVD in 1D, a formal proof for proving the same in 2D is not attempted for the lack of TVD theory in multi-dimensions.

Following the expression (19), the generic form of the weights for a typical triangular stencil (as in Fig. 3) can be written as,

\[
w_{\alpha} = \sum_{m=1}^{N} \frac{\alpha_{\alpha}^{m} |\Delta u_{j}| \gamma_{\mu_{1}}^{m} |\Delta u_{2}| \gamma_{\mu_{2}}^{m} |\Delta u_{3}| \gamma_{\mu_{3}}^{m}}{11_{l_{12}|\Delta u_{j}| \gamma_{\mu_{1}}^{m} |\Delta u_{2}| \gamma_{\mu_{2}}^{m} |\Delta u_{3}| \gamma_{\mu_{3}}^{m}}},
\]

where \( \alpha = 1 \) or 2 or 3 (corresponding to \( j \) or \( k \) or \( l \)) as there are three adjacent differences in flow variable (as compared to two in the one-dimensional case with a 3-point stencil). The weights can be chosen as \( w_{\alpha} = \frac{1}{11_{l_{12}|\Delta u_{j}| \gamma_{\mu_{1}}^{m} |\Delta u_{2}| \gamma_{\mu_{2}}^{m} |\Delta u_{3}| \gamma_{\mu_{3}}^{m}} \) (Monomial type), corresponding to the van Albada limiter in the 1D case. The new expressions for the two partial derivatives can then be written as,

\[
\left( \frac{\partial u}{\partial x} \right)_{j} = p_{1}(l_{22}r_{1} - l_{21}r_{2})/G,
\]

\[
\left( \frac{\partial u}{\partial y} \right)_{j} = p_{2}(l_{11}r_{2} - l_{12}r_{1})/G,
\]

where \( l_{11} = \sum_{q} w_{q} \Delta x_{q}^{2}, l_{22} = \sum_{q} w_{q} \Delta y_{q}^{2}, l_{12} = \sum_{q} w_{q} \Delta x_{q} \Delta y_{q}, r_{1} = \sum_{q} w_{q} \Delta x_{q} \Delta u_{q}, r_{2} = \sum_{q} w_{q} \Delta y_{q} \Delta u_{q} \) and \( G = l_{11}l_{22} - l_{12}^{2} \), with \( q = 1, 2, 3 \) (corresponding to the cell centers \( j, k, l \) in Fig. 3). In this case too, the addition of a multiplier is required for the same reasons as explained in the 1D case. As there are two derivatives here, two multipliers \( p_{1} \) and \( p_{2} \) are included.

Algorithm for SDWLS Reconstruction. The algorithm for the derivative computation and the variable reconstruction can be briefly summarized as:

1. Depending on the grid type and stencil selected, choose any weight as given in expression (19) (e.g. expression (20) for 3-point stencil in Fig. 1, and expression (25) for triangular stencil in Fig. 3) that would satisfy TVD property. However, the final form of the weight recommended while computation is

\[
w_{\alpha} = \sum_{m=1}^{N} \frac{\alpha_{\alpha}^{m}}{\prod_{\beta=1}^{N} |\Delta u_{\beta}| \gamma_{\beta}^{m}} + \epsilon.
\]

Here, \( \epsilon \), a small number, is added to the denominator in order to avoid division by zero.

2. Compute the gradient based on the WLS expressions (13) for 1D, and (26) and (27) for 2D.

3. In 2D, assign values to the multipliers \( p_{1} \) and \( p_{2} \), based on the sign of adjacent slopes. The quantities \( p_{1} \) and \( p_{2} \) take a non-zero value only when all adjacent slopes are of the same sign. For the 1D case, only one multiplier \( p \) is chosen based on the signs of adjacent slopes.

4. Reconstruct the variable at the cell interface using Eq. (15) for 1D and (21) and (22) for 2D.

In this paper, 2D extension is demonstrated only to illustrate the capability of extending the concept of weights perceived as influence coefficients to obtain oscillation free solutions. Hence, to avoid technical difficulties and for
ease of implementation, we demonstrate the 2D SDWLS formulation on a structured grid. However, the data is handled in an unstructured manner here.

9. Numerical test examples

Six test problems have been chosen to demonstrate the numerical feasibility of the newly developed limiters. The complexity of the test examples follows a logical course from the simplest hyperbolic equation—the one dimensional linear advection equation, to its non-linear counterpart, the inviscid Burger’s equation in both 1D and 2D cases. To demonstrate multi-dimensional feasibility, two 2D examples are illustrated, one being a linear convection equation and the other being a non-linear convection equation [18]. These examples are computed on a structured grid for simplicity. However, the higher order linear reconstruction is done using SDWLS formulation in an unstructured manner. The performance of the three new weight combinations (as in Table 3) and the corresponding new limiter functions (developed in Section 7) is demonstrated using the 1D examples given in Sections 9.1–9.3 below. In case of 2D, only a single weight choice is demonstrated to illustrate the possibility of extension of SDWLS approach to multi-dimensions.

9.1. Linear advection equation—square wave

The equation being solved here is the linear advection equation,
\[
\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = 0.
\]
This test problem is an unsteady one, given by Sweby [20], with a square wave initial condition. The domain, which has a unit length, is discretized into 100 evenly spaced points for the computation. The width and height of the initial square wave are taken to be 0.2 and 1.0 units respectively. The time for computation is chosen in such a way that the discontinuity never hits the boundaries and hence the boundaries are left free. The results with the three new weight combinations (given in Table 3) along with the exact solution and the first order solution are shown in Fig. 4. All the three new limiters (given in Section 7) corresponding to the above weights produce substantial improvement in accuracy over the first order solution. However, there is no significant difference between the results obtained with these three new weight combinations (and limiters).

9.2. Linear advection equation—sine-squared wave

This problem is similar to the previous test case, again given by Sweby [20]. The only difference is that a sine-squared wave is used as the initial condition. The domain is of unit length as in the previous example and is discretized

Fig. 4. Test case 1.
into 100 evenly spaced points. Initially, the sine-squared wave has a width and height of 0.2 units. This problem differs from the previous problem with respect to nature of the resolution required in the solution. Here, the solution is smooth (as shown in Fig. 5), whereas in the previous case, there are two discontinuities. It is observed from Fig. 5 that in this case too the above three new weight combinations (or limiter functions) yield substantially more accurate solutions compared to the first order solution. As in the first example, there is not much difference in performance between the new three limiters.

9.3. Inviscid Burger’s equation

The inviscid Burger’s equation in conservative form can be written as,

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

In this test case, which is taken from the book by Laney [13], 40 evenly spaced grid points are used to discretize the domain having a length of 2 units. The initial condition is a square wave having a width of $2/3$ units and a height of 1 unit. At the left boundary, Dirichlet boundary conditions are used, whereas the right boundary is free. The solution after 0.4 seconds, consists of a shock and an expansion fan. In this case, the exact solution never crosses the sonic point ($u = 0$); however, a numerical method may exhibit spurious oscillations. Hence, this serves as a good test case. The results with the three new weight combinations chosen as earlier, the first order results and the exact solution are shown in the Fig. 6. It can be seen that the solutions obtained with the new limiters are far more accurate relative to the first order solution. The performances of all the three limiters are very similar to each other in this example as in the case of the earlier two examples.

9.4. 2D linear convection problem 1

This example and the two following ones are solved using a single weight choice ($w_\alpha = 1/(\Delta u_\alpha^2 + \epsilon)$) to show the possibility of extension of the 1D SDWLS theory to multi-dimensions. Here, the problem is a linear one with a square domain $[0, 1] \times [0, 1]$, as given by Spekreijse [18].

The governing equation is:

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} + b \frac{\partial u}{\partial y} = 0,$$

where $a = \cos(\phi)$, $b = \sin(\phi)$, $\phi \in (0, \pi/2)$.

The boundary conditions are:
\[ u(0, y) = 1, \quad 0 < y < 1, \]
\[ u(x, 0) = 0, \quad 0 < x < 1. \]

A uniform \(32 \times 32\) grid was used for the computations and the two contour plots in Fig. 7 shows the first order result and the second order result from SDWLS scheme. The contour plot for the SDWLS scheme, given in Fig. 7(b) shows a much improved result over the first order one shown in Fig. 7(a), thus indicating the feasibility and attractiveness of the present approach. The convergence history for the SDWLS computation is shown in Fig. 7(c).

9.5. 2D linear convection problem 2

This is also a test problem solved by Spekreijse [18]. The problem is a linear convection problem on a rectangular domain \([-1, 1] \times [0, 1]\).

Here, the governing equation is:
\[
\frac{\partial u}{\partial t} + y \frac{\partial u}{\partial x} - x \frac{\partial u}{\partial y} = 0.
\]

The boundary conditions are:
\[ u(x, 0) = 0, \quad \text{if } x < -0.65, \]
\[ u(x, 0) = 1, \quad \text{if } -0.65 < x < -0.35, \]
\[ u(x, 0) = 0, \quad \text{if } -0.35 < x < 0, \]
\[ u(-1, y) = 0, \quad 0 < y < 1, \]
\[ u(x, 1) = 0, \quad 0 < x < 1. \]

The results shown in Fig. 8 were computed using a uniform \(64 \times 32\) grid. Both the first and second order (using SDWLS scheme) results compare well with the results of Spekreijse [18]. The results obtained using the SDWLS scheme (in Fig. 8(b)) show a substantial improvement in accuracy over the first order scheme (in Fig. 8(a)). The improvement in accuracy is also verified by the plot of the solution at the boundary shown in Fig. 8(d). The convergence history is shown in Fig. 8(c).

9.6. 2D non-linear convection problem

This test problem also solved by Spekreijse [18], is a non-linear one with a shock at the center of the square domain having dimensions \([0, 1] \times [0, 1]\).
The governing equation is:
\[
\frac{\partial u}{\partial t} + \frac{\partial}{\partial x} \left( \frac{u^2}{2} \right) - \frac{\partial u}{\partial y} = 0.
\]

The boundary conditions are:
\[
\begin{align*}
    u(0, y) &= 1, & 0 < y < 1, \\
    u(1, y) &= -1, & 0 < y < 1, \\
    u(x, 0) &= 1 - 2x, & 0 < x < 1.
\end{align*}
\]

In the steady state this equation reduces to the inviscid Burger’s equation. The exact solution is also given by Spekreijse [18]. The solutions obtained using the first order and the second order SDWLS schemes are shown in Fig. 9(a) and 9(b) respectively. The central shock and the three distinct regions in the steady state solution are clearly resolved with the SDWLS approach (as shown in Fig. 9(b)) which compares well with the exact solution, thus validating the SDWLS scheme. Fig. 9(c) shows the convergence history in this case.

10. Concluding remarks

In this paper, we investigated a weighted least squares based second-order accurate variable reconstruction technique for high resolution finite volume method. A link between the weights of least squares gradient estimation and various independently proposed limiter functions is discovered. Details about the above link is elaborated. Based on the link, a class of solution dependent weights for least squares reconstruction formulation is derived for the first time.
The expression of the solution dependent weights is cast in different forms and interpreted as an influence coefficient in order to be able to extend it to accommodate larger 1D stencils and multi-dimensional stencils. A generalized form of weights is derived that helps in exploring the possibility of generating innumerable new weights and corresponding new limiter functions. The weights and the limiter functions are categorized into different classes. In order to demonstrate the utility in generating new weights and limiter functions using the present formulation, one each from different categories of weights and their corresponding limiter functions are explored. It is also demonstrated as to how Sweby graph can be used in arriving at an acceptable new weight and its corresponding limiter function. Several test examples involving one and two dimensions are solved to demonstrate the capability of the present formulation to provide high resolution solutions.

All the results provided here are computed on structured grids involving scalar hyperbolic equations. However, the grids in case of two dimensions are treated in an unstructured manner. The extension of the present formulation to a system of equations and computations on truly unstructured grids are straightforward. The efforts are on to solve coupled non-linear systems of equations with the present formulation for future publications.

Appendix A

In the following sections derivations of weights corresponding to different limiter functions reported in literature and the coefficients for these weights are described. Here, all the major steps in the derivations are presented. Further details can be found in the reference [14]. In several limiter functions proposed in literature, a switch between two or more alternate forms is required based on the variable values at points on the stencil. The task of representing limiter functions in weighted least squares format would be complete, if we cast each of these forms of the limiter functions in terms of weights and use the same switching logic to select corresponding weights. Throughout the appendix, the limiter functions are denoted by $\psi$ and the various forms of the limiter functions are denoted by $\psi_i$, $i = 1, 2, 3, 4$, where the subscript $i$ denotes the particular form of a particular limiter function. As an example, let us consider the Minmod limiter $[12]$. Here, $\psi = \frac{\Delta u_1}{\Delta x}$ if $|\Delta u_1| < |\Delta u_2|$ else, $\psi = \frac{\Delta u_2}{\Delta x}$. In our terminology, the first form for $\psi$ would...
be denoted as $\psi_1$ and the second one as $\psi_2$. In the following sub-sections, the corresponding weights for each limiter function listed in Tables 1 and 2 are derived.

It is important to note that for all the limiter functions, the multiplier $p$ introduced in Eqs. (18), (26), (27) takes the value 0, if the adjacent slopes have opposite sign. This step also forms a part of the switching condition and hence is not mentioned explicitly in any of the following sub-sections. However, $p$ is also an integral part of the limiter function in cases of limiters such as MC limiter, and in those cases, the value of $p$ is explicitly mentioned.

It is useful to note that in the case of the Logical type limiter functions (from Section A.4 onwards), the weights do not depend on the values of the variable being reconstructed. Hence, the divided form can be obtained easily by inspection alone, and so the derivation of the divided form is avoided in the corresponding sections.

A.1. Van Albada limiter

The van Albada limiter proposed by van Albada et al. [21] is,

$$
\psi(r) = \frac{r^2 + r}{r^2 + 1}.
$$

Converting the variable $r$ in terms of the ratio of the adjacent slopes, $r = \frac{\Delta u_2}{\Delta u_1}$ and multiplying the numerator and denominator of the resultant right hand side expression yields,

$$
\psi = \frac{(\Delta u_2^2)\Delta u_1 + (\Delta u_1^2)\Delta u_2}{\Delta u_1(\Delta u_2^2 + \Delta u_1^2)}.
$$

(A.1)
Now, we look at the similarities between the derivatives obtained by the limiter based method and the weighted least squares method. Comparing the above expression for $\psi$ in Eq. (A.1) with Eq. (18), where the limiter function is written in terms of weights, we identify the following choices of weights, $w_1 = \frac{1}{\Delta u_1^2}$ and $w_2 = \frac{1}{\Delta u_2^2}$. It may be noted that the value of $p = 1$ in (18) is considered when higher order part is activated.

We now express these weights in terms of the coefficients in expression (20). One can observe that division of the numerator and denominator of the resulting expression for $\psi$ obtained using these two terms would be denoted as $\psi_1$ and $\psi_2$ respectively. Using Eqs. (A.3) and (A.4), multiplying the numerator and denominator of the resulting expression for $\phi_1$ and simplifying further, we get further

$$\phi_1 = \frac{4r(3r + 1) + \frac{\delta}{2}(r + 1)^2}{11r^2 + 4r + 1 + \frac{\delta}{2}(r + 1)^2}.$$
which can be written as,

\[
\phi_1 = \frac{4r(3r + 1) + \epsilon_1}{11r^2 + 4r + 1 + \epsilon_1}
\]  

(A.6)

by introducing \(\epsilon_1 = \frac{r}{2}(r + 1)^2\).

Similar to the operations performed to yield \(\phi_1\) from Eqs. (A.3) and (A.4), we get \(\phi_2\) as,

\[
\phi_2 = \frac{4(r + 3) + \epsilon_1}{r^2 + 4r + 11 + \epsilon_1}.
\]  

(A.7)

Substituting expressions for \(\phi_1\) and \(\phi_2\) in Eq. (A.3) we can get \(\psi_1\) and \(\psi_2\) as explained below. We shall retain the same logic as given by Venkatakrishnan in choosing between \(\psi_1\) and \(\psi_2\) as the final limiter based on the minimum of \(\phi_1\) and \(\phi_2\). Now,

\[
\psi_1 = \frac{1}{2}(r + 1)\phi_1
\]

which can be written after substituting \(\phi_1\) from (A.6) and simplification using the definition of \(r\) as,

\[
\psi_1 = \frac{2\Delta u_2(3\frac{\Delta u_2^2}{\Delta u_1^2} + \frac{\Delta u_3}{\Delta u_1} + 3\frac{\Delta u_2}{\Delta u_1} + 1) + \epsilon_1(\frac{\Delta u_2}{\Delta u_1} + 1)\Delta u_1}{\Delta u_1(11\frac{\Delta u_2^2}{\Delta u_1^2} + 4\frac{\Delta u_2}{\Delta u_1} + 1 + \epsilon_1)}
\]

(A.8)

Multiplying Numerator and Denominator by \(\Delta u_1^2\) and simplifying we get,

\[
\psi_1 = \frac{(5\Delta u_2^2 + \Delta u_1\Delta u_2 + \frac{1}{2}\epsilon_1\Delta u_1^3)\Delta u_1 + (6\Delta u_2^2 + \Delta u_1^2 + 3\Delta u_1\Delta u_2 + \frac{1}{2}\epsilon_1\Delta u_1^2)\Delta u_2}{\Delta u_1((5\Delta u_2^2 + \Delta u_1\Delta u_2 + \frac{1}{2}\epsilon_1\Delta u_1^2) + (6\Delta u_2^2 + \Delta u_1^2 + 3\Delta u_1\Delta u_2 + \frac{1}{2}\epsilon_1\Delta u_1^2))}
\]

(A.8)

Comparing the above expression for \(\psi\) in (A.8) with Eq. (18) (with \(p = 1\) for higher order case), where the limiter function is written in terms of weights, we identify the following choices of weights,

\[
w_1 = \frac{\epsilon_1}{2}\Delta u_1^2 + 5\Delta u_2^2 + \Delta u_1\Delta u_2 \quad \text{and} \quad w_2 = \left(1 + \frac{\epsilon_1}{2}\right)\Delta u_1^2 + 6\Delta u_2^2 + 3\Delta u_1\Delta u_2.
\]

Similarly, \(\psi_2\) can be obtained as,

\[
\psi_2 = \frac{(6\Delta u_1^2 + \Delta u_2^2 + 3\Delta u_1\Delta u_2 + \frac{1}{2}\epsilon_1\Delta u_1^3)\Delta u_1 + (5\Delta u_1^2 + \Delta u_1\Delta u_2 + \frac{1}{2}\epsilon_1\Delta u_1^2)\Delta u_2}{\Delta u_1((6\Delta u_1^2 + \Delta u_2^2 + 3\Delta u_1\Delta u_2 + \frac{1}{2}\epsilon_1\Delta u_1^2) + (5\Delta u_1^2 + \Delta u_1\Delta u_2 + \frac{1}{2}\epsilon_1\Delta u_1^2))}
\]

(A.9)

Comparing Eq. (A.9) and Eq. (18) we get,

\[
w_1 = \left(6 + \frac{1}{2}\epsilon_1\right)\Delta u_1^2 + \Delta u_2^2 + 3\Delta u_1\Delta u_2 \quad \text{and} \quad w_2 = \left(5 + \frac{1}{2}\epsilon_1\right)\Delta u_1^2 + \Delta u_1\Delta u_2
\]

as the choice of weights corresponding to the second form of the Venkatakrishnan limiter function.

We now express these weights in the divided form (i.e. in terms of the coefficients in expression (20)). There are four weights corresponding to the two limiter function forms. Let us consider the weights corresponding to \(\psi_1\). Consider dividing \(w_1\) and \(w_2\) obtained earlier for \(\psi_1\) by \(\Delta u_1^2\). Due to this operation \(w_1\) becomes,

\[
w_1 = \frac{5}{\Delta u_1^2} + \frac{\epsilon_1}{2\Delta u_2^2} + \frac{1}{\Delta u_1\Delta u_2}
\]

which implies \(N = 3\) in Eq. (20), and the coefficients of the three terms in expression (20) are, \(a_1^1 = 5, \gamma_{1,1}^1 = 2, \gamma_{1,2}^1 = 0, a_2^1 = 1, \gamma_{1,1}^2 = 1, \gamma_{1,2}^2 = 1\) and \(a_1^3 = \frac{3}{2}, \gamma_{1,1}^3 = 0, \gamma_{1,2}^3 = 2\). Now, \(w_2\), becomes,

\[
w_2 = \frac{6}{\Delta u_1^2} + \frac{(1 + \frac{\epsilon_1}{2})}{\Delta u_2^2} + \frac{3}{\Delta u_1\Delta u_2}
\]
which again implies $N = 3$ in Eq. (20) for $w_2$ also, and the coefficients of the three terms in expression (20) are,

\[ a_1^3 = 6, \gamma_{1,1}^3 = 2, \gamma_{1,2}^3 = 0, a_2^3 = 1 + 1/2, \gamma_{2,1}^3 = 0, \gamma_{2,2}^3 = 2, a_3^3 = 3, \gamma_{3,1}^3 = 1 \text{ and } \gamma_{3,2}^3 = 1. \]

Adopting a very similar approach, the two weights corresponding to $\psi_2$ are modified as,

\[
\begin{align*}
    w_1 &= \frac{1}{\Delta u_1^2} + \frac{(6 + \frac{1}{2}\epsilon_1)}{\Delta u_2^2} + \frac{3}{\Delta u_1 \Delta u_2} \quad \text{and} \quad w_2 = \frac{(5 + \frac{1}{2}\epsilon_1)}{\Delta u_2^2} + \frac{1}{\Delta u_1 \Delta u_2}
\end{align*}
\]

which yield the coefficients in expression (20) for $w_1$ as $a_1^1 = 1, \gamma_{1,1}^1 = 2, \gamma_{1,2}^1 = 0, a_1^2 = 6 + \frac{1}{2}, \gamma_{1,1}^2 = 0, \gamma_{1,2}^2 = 2, a_1^3 = 3, \gamma_{1,1}^3 = 1 \text{ and } \gamma_{1,2}^3 = 1, \text{ and for } w_2 \text{ as, } a_2^1 = 1, \gamma_{2,1}^1 = 1, \gamma_{2,2}^1 = 1, a_2^2 = 5 + \frac{1}{2}, \gamma_{2,1}^2 = 0 \text{ and } \gamma_{2,2}^2 = 2 \text{ (the other coefficients being zero since there are only two terms in } w_2 \text{ corresponding to } \psi_2).$

### A.3. Van Leer limiter

The van Leer limiter is given as [22],

\[
\psi(r) = \frac{r + |r|}{r + 1}
\]

Realizing that the above form is identical to

\[
\psi(r) = \frac{r + |r|}{|r| + 1}
\]

the expression for $\psi$, after using the definition of $r$, multiplying the numerator and denominator of the right hand side of the resulting expression by $\Delta u_1$, can be written as,

\[
\psi(r) = \frac{((\frac{\Delta u_2}{\Delta u_1})\Delta u_1 + (1)\Delta u_2)}{\Delta u_1((\frac{\Delta u_2}{\Delta u_1}) + 1)}.
\]

Comparing Eq. (A.10) and Eq. (18) (with $p = 1$), we get the following expressions for the weights as $w_1 = |\Delta u_2|$ and $w_2 = |\Delta u_1|$. The divided form of the weights (shown in Table 2) can be easily obtained by dividing $w_1$ and $w_2$ by $|\Delta u_1| |\Delta u_2|$.  

### A.4. Minmod limiter

The Minmod limiter is written as [12], $\psi = \frac{\Delta u_1}{\Delta x}$ if $|\Delta u_1| < |\Delta u_2|$ else, $\psi = \frac{\Delta u_2}{\Delta x}$. Therefore, in our terminology $\psi_1 = 1$ and $\psi_2 = |\Delta u_2|/\Delta u_1$. Comparing above expressions for $\psi_1$ and $\psi_2$ and (18), we get the corresponding weights as $w_1 = 1, w_2 = 0$ for $\psi_1$ and $w_1 = 0, w_2 = 1$ for $\psi_2$. In the case of this limiter function, as well as all subsequent limiter functions, the weights do not depend on the values of the variable being reconstructed. Hence, the divided form can be obtained easily by inspection alone, and so the derivation of the divided form is avoided in this and further sections.

### A.5. Superbee limiter

The Superbee limiter is given as [17],

\[
\frac{du}{dx} = \maxmod\left(\frac{du_1}{dx} \cdot \frac{du_2}{dx}\right)
\]

where

\[
\frac{du_1}{dx} = \minmod\left(\frac{\Delta u_2}{\Delta x}, 2\left(\frac{\Delta u_1}{\Delta x}\right)\right), \quad \frac{du_2}{dx} = \minmod\left(2\left(\frac{\Delta u_2}{\Delta x}\right), \frac{\Delta u_1}{\Delta x}\right).
\]

This function results in four alternative forms of the limiter function $\psi$. These four forms and their corresponding weights and multiplier $p$ obtained by comparison with Eq. (18) are,
\[ \psi_1 = \Delta u_2 / \Delta u_1 \quad \text{for which } p = 1, w_1 = 0, w_2 = 1; \]
\[ \psi_2 = 2 \quad \text{for which } p = 2, w_1 = 1, w_2 = 0; \]
\[ \psi_3 = 2\Delta u_2 / \Delta u_1 \quad \text{for which } p = 2, w_1 = 0, w_2 = 1; \]
\[ \psi_4 = 1 \quad \text{for which } p = 1, w_1 = 1, w_2 = 0. \]

### A.6. MC limiter

The monotonized central-difference limiter, popularly known as the MC limiter is given as [23],
\[
\frac{du}{dx} = \minmod\left( \frac{\Delta u_1 + \Delta u_2}{2\Delta x}, \frac{2\Delta u_1}{\Delta x}, \frac{2\Delta u_2}{\Delta x} \right).
\]
The above function, results in three distinct limiter forms as,
\[ \psi_1 = (\Delta u_1 + \Delta u_2) / \Delta u_1 \quad \text{for which } p = 1, w_1 = 1, w_2 = 1; \]
\[ \psi_2 = 2 \quad \text{for which } p = 2, w_1 = 1, w_2 = 0; \]
\[ \psi_3 = 2\Delta u_2 / \Delta u_1 \quad \text{for which } p = 2, w_1 = 0, w_2 = 1. \]

### A.7. Barth and Jesperson central limiter

This limiter is given as [4],
\[ \psi(r) = \frac{1}{2} (r + 1) \min\left( 1, \frac{4r}{r + 1} \right), \min\left( 1, \frac{4}{r + 1} \right) \right]. \]
The above expression has four different limiter function forms. These forms and the weights obtained by substituting the expression for \( r \) in these forms, subsequent simplification and comparison with Eq. (18) are given as,
\[ \psi_1 = (\Delta u_1 + \Delta u_2) / 2\Delta u_1 \quad \text{for which } p = 1, w_1 = 1, w_2 = 1; \]
\[ \psi_2 = 2\Delta u_2 / \Delta u_1 \quad \text{for which } p = 2, w_1 = 0, w_2 = 1; \]
\[ \psi_3 = (\Delta u_1 + \Delta u_2) / 2\Delta u_1 \quad \text{for which } p = 1, w_1 = 1, w_2 = 1; \]
\[ \psi_4 = 2 \quad \text{for which } p = 2, w_1 = 1, w_2 = 0. \]

### A.8. Barth and Jesperson upwind limiter

This limiter is given as [4], \( \psi(r) = \min(1, 2r) \) and has two alternate forms. The simplified expressions for these forms and the corresponding weights obtained using the link given in Eq. (18) can be written as
\[ \psi_1 = 1 \quad \text{for which } p = 1, w_1 = 1, w_2 = 0; \]
\[ \psi_2 = 2\Delta u_2 / \Delta u_1 \quad \text{for which } p = 2, w_1 = 0, w_2 = 1. \]

### A.9. Barth and Jesperson downwind limiter

This limiter again has two alternate forms and is given as [4], \( \psi(r) = \min(r, 2) \) The simplified expressions for these two forms and the corresponding weights obtained using the link given in Eq. (18) can be written as
\[ \psi_1 = \Delta u_2 / \Delta u_1 \quad \text{for which } p = 1, w_1 = 0, w_2 = 1; \]
\[ \psi_2 = 2 \quad \text{for which } p = 2, w_1 = 1, w_2 = 0. \]
References