NASA Technical Memorandum 103789

Accurate Monotone Cubic Interpolation

Hung T. Huynh
Lewis Research Center
Cleveland, Ohio

March 1991

NASA

ACCURATE MONOTONE CUBIC INTERPOLATION

Hung T. Huynh

National Aeronautics and Space Administration

Cleveland, Ohio 44135

Abstract. Monotone piecewise cubic interpolants are simple and effective. They are generally third-order accurate, except near strict local extrema where accuracy degenerates to second-order due to the monotonicity constraint. Algorithms for piecewise cubic interpolants, which preserve monotonicity as well as uniform third and fourth-order accuracy, are presented. The gain of accuracy is obtained by relaxing the monotonicity constraint in a geometric framework in which the median function plays a

and Butland [9]. They constructed limiter functions whose arguments are the slopes of the piecewise linear interpolation. The resulting derivatives satisfy the monotonicity constraint *a priori*, but they are at best second-order accurate. The constraints of De Boor and Swartz and Hyman, on the other hand, can be applied to any high-order approximation of the derivatives.

A major problem of the above methods, including the fourth-order algorithm in [7], is that if the data are no longer monotone, the constraints may cause a loss of accuracy. In fact, near strict local extrema, the interpolants are only second-order accurate, i.e., the error is $O(h^2)$, and thus, no better than linear interpolation. Recently, Hyman's constraint was modified to obtain uniform third-order accuracy [6]. The proofs of stability and monotonicity, however, were not shown.

In a different context, namely numerical simulations of conservation laws, there has been a great deal of effort to preserve monotonicity by constraining the derivatives, see [12], [18], [20]–[22], [24], [27], and the references given there. In an earlier paper [26], Van Leer introduced a monotonicity constraint which corresponds to De Boor and Swartz's constraint for quadratic interpolation. His harmonic mean limiter is also identical to that of Butland [2]. In this context, the study of limiter functions was simplified by reducing them to functions of one variable instead of two. It is well-known that all limiters suffer a loss of accuracy near strict local extrema. Recently, Roe [22] suggested that "Although only very simple ideas are involved, more research is still needed to clarify the properties of limiter functions." In [15], Harten and Osher solved the loss of accuracy problem by combining monotonicity and accuracy into nonoscillatory, piecewise parabolic interpolation. The concept behind their UNO method completely deviates from that of the monotonicity constraint: it defines high-order accurate derivatives which do not cause oscillations, but it does not provide a constraint for other approximations of the derivatives. The ENO schemes [14], which are higher-order extensions of the UNO method, are unstable in the sense that a small change in the data can produce a large change in the interpolant. Although the UNO and ENO schemes are highly accurate, they are somewhat diffusive, e.g., they smear contact discontinuities [13], [28], [30]. Modifications of these schemes to improve the resolution at contact discontinuities can be found in [13], [28]. In [16], the present author took a different approach to obtain uniform high accuracy: the monotonicity constraint was extended in a geometric framework using the UNO method. The resulting SONIC schemes are highly accurate [16], [29], and the UNO scheme itself is a member of the SONIC class.

We follow the author's geometric approach below. First, the concept of monotonicity of the data in an interval is introduced so that monotone parts can be distinguished from strict local extrema. Next, we present a geometric framework in which the median function plays a crucial role for all methods. In this framework, De Boor and Swartz's constraint is essentially as simple as Hyman's. Since the latter may cause oscillations, we choose to extend the former. Our systematic approach results in several uniform third-order as well as fourth-order constraints which are stable and can be applied to any high-order approximation of the derivatives. The geometric framework and the introduction of new concepts also made possible the proofs of monotonicity, accuracy, and stability. We note that the UNO scheme is a member of the class of third-order methods presented here. It produces interpolants which are the "flattest" in this class. On the contrary, the fourth-order ENO scheme, which is unstable, does not belong to the fourth-order class whose members are stable. Since several of our methods employ limiter functions, we present a rather complete analysis of the essential properties of limiters. Finally, for smoother transitions near the boundaries, we introduce nonlinear boundary conditions via the median function. Note that our methods can also be combined with cubic splines in a manner similar to Hyman's method [17]. However, only local methods are presented below.

This paper is essentially self-contained. In section 2, monotonicity of the data in an interval is defined, and well-known results on Hermite cubics are reviewed. The De Boor and Swartz monotonicity constraint and algorithm are presented using our framework in subsection 2.1. Subsection 2.2 deals with limiter functions by first reducing them to functions of one variable. It is shown that the construction of a limiter g can be reduced to that of a function on $[0,1]$ with $g(0) = 0$, and $g(1) = 1$. Furthermore, the order of accuracy of a limiter is determined by the slope $g'(1)$ in the case of a uniform mesh. In subsection 2.3, Hyman's method is described, and examples which show the difficulties of obtaining both accuracy and monotonicity are presented. These difficulties are then resolved in section 3. Subsection 3.1 describes the UNO scheme using the median concept. In subsequent subsections, three different monotonicity-preserving constraints, which are stable and provide third-order accurate interpolants, are presented. It is shown that these constraints reduce to that of de Boor and Swartz for monotone data; however, they can be less restrictive when the data are no longer monotone. In subsection 3.5, besides additional analysis and comparison of third-order methods, we present examples which show that all third-order constraints may not reproduce a cubic, and that near inflection points with zero slopes, they are no higher than third-order accurate. In section 4, fourth-order methods are presented. Nonoscillatory cubics, which are introduced in subsection 4.1, are employed in the definition and the development of fourth-order q-monotonicity-preserving constraints in subsection 4.2. Several nonlinear boundary conditions are presented in section 5 by using the median function. Numerical results are shown in section 6. Finally, the discussion on some applications of the new methods and the conclusions are presented in section 7. For readers who are only interested in high-order monotonicity-preserving constraints, we suggest that they skip subsections 2.2 and 2.3.

2. Cubic Hermite interpolation. Let the mesh $\{x_i\}_{i=m}^{n}$, $m < n$, be a partition of the interval $[x_m, x_n]$ such that $x_m < x_{m+1} < \ldots < x_{n-1} < x_n$. Let $\{f_i\}$ be the corresponding data which are samples of a piecewise smooth function f, and let $F_i = (x_i, f_i)$. The local mesh spacing is $\Delta x_{i+1/2} = x_{i+1} - x_i$, and the maximum value of all $\Delta x_{i+1/2}$ is denoted by h. The slope of the piecewise linear interpolant is equal to the first divided difference,

$$s_{i+1/2} = f_{[x_i, x_{i+1}]} = \Delta f_{i+1/2} / \Delta x_{i+1/2}. \tag{2.1}$$

The data are *increasing at* x_i if $f_{i-1} \leq f_i \leq f_{i+1}$. They are *increasing in* $[x_i, x_{i+1}]$ if they are increasing at x_i and x_{i+1}, i.e., $f_{i-1} \leq f_i \leq f_{i+1} \leq f_{i+2}$. Similar definitions hold for decreasing data with appropriate sign changes. The data are *monotone at* x_i (or *in* $[x_i, x_{i+1}]$) if they are increasing or decreasing at x_i (or in $[x_i, x_{i+1}]$). Note that our definition of monotonicity of the data in $[x_i, x_{i+1}]$ is rather straightforward; however, it plays an essential role in the high-order extensions of the monotonicity constraints and their proofs. The interpolant Pf is *monotone in* $[x_i, x_{i+1}]$ if $(Pf)(x)$

is monotone for x between x_i and x_{i+1}. The interpolant is of class C^k if $(Pf)(x)$ is continuous and has continuous derivatives for all orders less than or equal to k.

Given the data $\{f_i\}$ and the slopes $\{\dot{f}_i\}$, which respectively approximate the exact values $\{f(x_i)\}$ and $\{f'(x_i)\}$, the cubic Hermite interpolant is defined for $x_m \le x \le x_n$ in terms of f_i, f_{i+1}, $\dot{f}_i$, $\dot{f}_{i+1}$ by (see [4] or any standard text on numerical methods),

$$q(x) = c_3(x - x_i)^3 + c_2(x - x_i)^2 + c_1(x - x_i) + c_0 \tag{2.2}$$

where, for $x_i \le x \le x_{i+1}$,

$$\begin{aligned} c_0 &= f_i, \quad c_1 = \dot{f}_i, \\ c_2 &= (3s_{i+1/2} - 2\dot{f}_i - \dot{f}_{i+1})/(\Delta x_{i+1/2}), \\ c_3 &= (\dot{f}_i + \dot{f}_{i+1} - 2s_{i+1/2})/(\Delta x_{i+1/2})^2. \end{aligned}$$

One can easily verify that $q(x_i) = f_i$, $q'(x_i) = \dot{f}_i$, $q(x_{i+1}) = f_{i+1}$, $q'(x_{i+1}) = \dot{f}_{i+1}$. Consequently, the interpolant (2.2) has a continuous first derivative ($q(x) \in C^1$). It is fourth-order accurate if the derivatives $\{\dot{f}_i\}$ are exact or third-order, and third-order if the derivatives are second-order, etc. These derivatives can be approximated by using polynomial interpolation. If quadratic interpolation is used, one obtains the following formula which is second-order accurate: it approximates the exact derivative to $O(h^2)$,

$$\dot{f}_i = \frac{\Delta x_{i-1/2} s_{i+1/2} + \Delta x_{i+1/2} s_{i-1/2}}{\Delta x_{i-1/2} + \Delta x_{i+1/2}}. \tag{2.3}$$

(Note that the order of accuracy as defined in the literature dealing with conservation laws is one less than that defined here, which follows the definition in the literature dealing with monotone interpolation). The above expression is called the parabolic formula [17] since it gives the slope at x_i of the parabola $P_i(x) = (x, p_i(x))$ through the points F_{i-1}, F_i, and F_{i+1}. Such an approximation has the advantage of being local in the sense that only nearby data are used. The resulting interpolant (2.2) is also local. If the data $\{f_i\}$ are monotone, the interpolant is not necessarily monotone. One needs additional constraints on $\{\dot{f}_i\}$.

2.1. Monotonicity constraints. Let $I[z_1, \ldots, z_k]$ be the smallest closed interval containing $z_1, \ldots, z_k$, i.e.,

$$I[z_1, \ldots, z_k] = [\min(z_1, \ldots, z_k), \max(z_1, \ldots, z_k)].$$

Let the median of three numbers be the one which lies between the other two. Notice that the median lies in the interval defined by any two of the three arguments. Let

$$\operatorname{minmod}(x, y) = \operatorname{median}(x, y, 0),$$

then a convenient formula for the minmod function is

$$\operatorname{minmod}(x, y) = \frac{1}{2}\left[\operatorname{sgn}(x) + \operatorname{sgn}(y)\right] \min(|x|, |y|) \tag{2.4}$$

where $\operatorname{sgn}(x) = 1$ if x is positive, and $\operatorname{sgn}(x) = -1$ if x is negative. Observe that if $x = 0$, it does not matter whether $\operatorname{sgn}(x)$ is defined as -1, 0, or 1 in (2.4) since the resulting minmod is 0. The minmod function is more commonly defined as, see e.g., [12], [15], [24], [26],

$$\operatorname{minmod}(x, y) = \begin{cases} \operatorname{sgn}(x) \min(|x|, |y|) & \text{if } xy > 0, \\ 0 & \text{otherwise.} \end{cases}$$

Conversely, the median function can be expressed in terms of minmod:

$$\begin{aligned} \operatorname{median}(x, y, z) &= x + \operatorname{minmod}(y - x, z - x) \\ &= y + \operatorname{minmod}(x - y, z - y). \end{aligned} \tag{2.5}$$

The following sufficient condition for monotonicity is a generalization of an observation by De Boor and Swartz [5]. It was developed and used by many authors [6], [10], [17], [19].

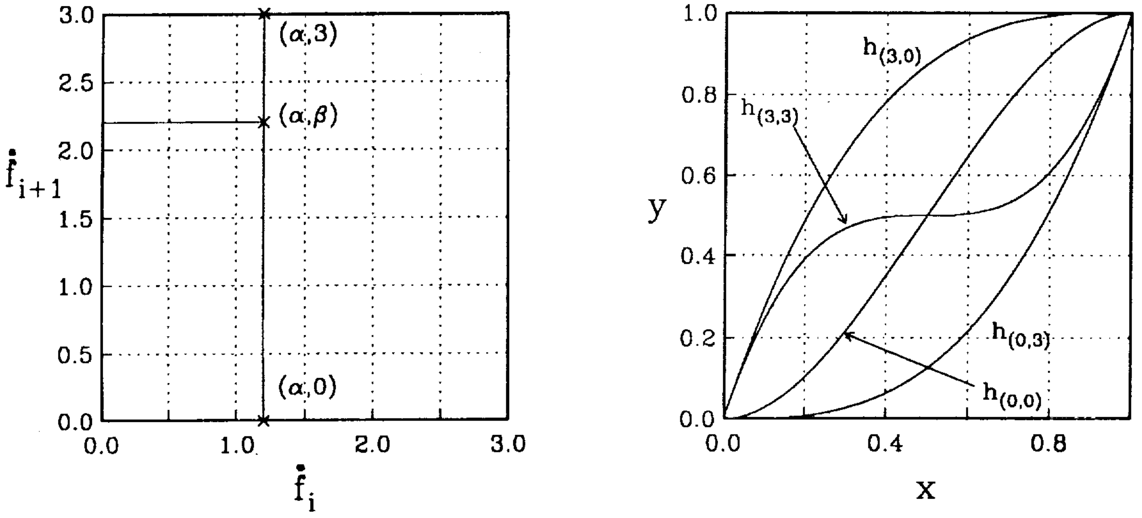
Lemma 1. *If*

$$\dot{f}_i, \dot{f}_{i+1} \in I\,[0, 3s_{i+1/2}], \tag{2.6}$$

then the resulting interpolant (2.2) is monotone in $[x_i, x_{i+1}]$.

The constructive proof below is simpler than those of the necessary and sufficient condition [6], [10]. The case of $f_i = f_{i+1}$ is trivial since the interpolant is a constant function. If $f_i \neq f_{i+1}$, consider the linear change of coordinates $\tilde{x} = (x - x_i)/\Delta x_{i+1/2}$, $\tilde{y} = (y - f_i)/\Delta f_{i+1/2}$, where (x, y) is the original coordinate system. Clearly, the cubic $y = q(x)$ defined by (2.2) is transformed to a cubic $\tilde{y} = \tilde{q}(\tilde{x})$. By using the chain rule, the monotonicity of $q(x)$ for $x_i \leq x \leq x_{i+1}$ is equivalent to the monotonicity of $\tilde{q}(\tilde{x})$ for $0 \leq \tilde{x} \leq 1$. Without loss of generality, therefore, one may assume $x_i = 0$, $x_{i+1} = 1$, and $f_i = 0$, $f_{i+1} = 1$. As a result, each pair $(\dot{f}_i, \dot{f}_{i+1})$ determines an interpolant. The four pairs $(\dot{f}_i, \dot{f}_{i+1}) = (0,0)$, $(3,0)$, $(0,3)$, and (3,3) form the corners of a square called the De Boor and Swartz square shown in Fig. 1a. Denote the four corresponding interpolants by $h_{(0,0)}$, $h_{(3,0)}$, $h_{(0,3)}$, and $h_{(3,3)}$. Using definition (2.2), it follows that

$$h_{(0,0)}(x) = -2x^3 + 3x^2, \qquad h_{(3,0)}(x) = x^3 - 3x^2 + 3x,$$
$$h_{(0,3)}(x) = x^3, \qquad h_{(3,3)}(x) = 4x^3 - 6x^2 + 3x.$$



(a) *De Boor and Swartz's square* (b) *The interpolants of its four corners*

Figs. 1. *De Boor and Swartz's square and the interpolants of its four corners.*

One can verify that the above cubics are increasing for $0 \le x \le 1$, see Fig. 1b. Any linear combination of them with positive coefficients is also increasing for $0 \le x \le 1$. Denote $(\dot{f}_i, \dot{f}_{i+1})$ by (α, β), then condition (2.6) is equivalent to $0 \le \alpha/3 \le 1$, and $0 \le \beta/3 \le 1$. Consequently, the following cubic Hermite interpolants are increasing for $0 \le x \le 1$:

$$h_{(\alpha,0)} = \frac{\alpha}{3} h_{(3,0)} + (1 - \frac{\alpha}{3}) h_{(0,0)},$$
$$h_{(\alpha,3)} = \frac{\alpha}{3} h_{(3,3)} + (1 - \frac{\alpha}{3}) h_{(0,3)},$$
$$h_{(\alpha,\beta)} = \frac{\beta}{3} h_{(\alpha,3)} + (1 - \frac{\beta}{3}) h_{(\alpha,0)}.$$

Note that the slopes $(\dot{f}_i, \dot{f}_{i+1})$ of the above cubics are respectively $(\alpha, 0)$, $(\alpha, 3)$, and (α, β). This completes the proof.

Ferguson and Miller [8], and Fritsch and Carlson [10] independently found a necessary and sufficient condition which is an extension of (2.6), see also [6]. Because of its simplicity, however, condition (2.6) is more commonly used. Note that in the context of conservation laws, the factor 3 in (2.6) is replaced by 2. The reason for this difference is that the solutions, as functions of time steps, correspond to quadratic rather than cubic interpolations, see [16].

Expression (2.6) with index i replaced by $i-1$ together with (2.6) imply

$$\dot{f}_i \in I\,[0, 3s_{i-1/2}] \quad \text{and} \quad \dot{f}_i \in I\,[0, 3s_{i+1/2}]. \tag{2.6'}$$

Consequently, $\dot{f}_i$ belongs to the intersection of $I\,[0, 3s_{i-1/2}]$ and $I\,[0, 3s_{i+1/2}]$, which is $I\,[0, \text{minmod}\,(3s_{i-1/2}, 3s_{i+1/2})]$. Thus, if

$$\dot{f}_i \in I\,[0, 3s_i] \tag{2.7}$$

where

$$s_i = \text{minmod}\,(s_{i-1/2}, s_{i+1/2}) \tag{2.8}$$

for all i, $m < i < n$, then the interpolant (2.2) is monotone in $[x_i, x_{i+1}]$, $m < i < n-1$.

To bring $\dot{f}_i$ into the interval $I\,[0, 3s_i]$, it suffices to replace $\dot{f}_i$ by the median of $\dot{f}_i$, 0, and $3s_i$. Using the definition of the minmod function, one obtains the monotonicity algorithm

$$\dot{f}_i \leftarrow \text{minmod}\,(\dot{f}_i, 3s_i). \tag{2.9}$$

Notice that (2.9) can also be expressed as

$$\dot{f}_i \leftarrow \text{minmod}\,[\text{minmod}\,(\dot{f}_i, 3s_{i-1/2}), 3s_{i+1/2}], \tag{2.9'}$$

i.e., the constraints are enforced on the left, and then on the right. Both (2.9) and (2.9′) are equivalent to the following commonly used formula:

$$\dot{f}_i \leftarrow \begin{cases} 0 & \text{if } s_{i-1/2}s_{i+1/2} \le 0, \\ \min\,[\max\,(0, \dot{f}_i), 3\min\,(s_{i-1/2}, s_{i+1/2})] & \text{if } s_{i-1/2} \text{ and } s_{i+1/2} > 0, \\ \max\,[\min\,(0, \dot{f}_i), 3\max\,(s_{i-1/2}, s_{i+1/2})] & \text{if } s_{i-1/2} \text{ and } s_{i+1/2} < 0. \end{cases}$$

Expressions (2.9), (2.7), and the interval in (2.7) are called the MP (monotonicity-preserving) algorithm, the MP constraint, and the MP interval, respectively. The combination of (2.3) and (2.9) is called the MP-parabolic algorithm.

Next, we show that if $f \in C^2$, $\{f_i\}$ are second or higher-order accurate, and the original $\{\dot{f}_i\}$ are at least first-order, then at the part where the exact derivative $f'(x) \neq 0$, the MP algorithm has no effect provided the mesh is fine enough. Indeed, let x_i be in a small neighborhood of x such that $f'(x_i) \neq 0$. Since both $s_{i-1/2}$ and $s_{i+1/2}$ approximate $f'(x_i) \neq 0$ to $O(h)$, they are both strictly positive, or strictly negative for small enough h; consequently, the MP interval contains $I[0, 3f'(x_i)]$ up to an error of the order $O(h)$. Because the original $\dot{f}_i$ approximates $f'(x_i)$ to at least $O(h)$, it belongs to the MP interval, and thus is not altered by the MP algorithm.

To analyze the MP constraint near strict local extrema, assume that $f \in C^3$, $\{f_i\}$ are third or higher-order accurate, and the original $\{\dot{f}_i\}$ are at least second-order. Near a strict local extremum at $x = x^*$, suppose $f''(x^*) \neq 0$. The following arguments hold in a small neighborhood of x^* where higher-order terms can be neglected. Again, by using a linear change of coordinates as shown in the proof of Lemma 1, we may assume $x^* = 0$, and $f(x) = x^2 +$ higher-order terms. We claim that if $0 \leq x_{i-1} < x_i < x_{i+1}$ (similarly for $x_{i-1} < x_i < x_{i+1} \leq 0$), then the MP algorithm does not alter $\dot{f}_i$ provided h is small enough. Indeed, by calculating $s_{i-1/2}$, $s_{i+1/2}$ from the above expression for $f(x)$, it follows that the MP interval contains $[0, 3x_i]$. Since $f'(x_i) = 2x_i + O(x_i^2)$, and $|\dot{f}_i - f'(x_i)| = O(x_i^2)$, one concludes that $\dot{f}_i$ lies in the MP interval if h is small enough, and the claim follows. Observe that there are at most two data points adjacent to the local extremum which satisfy neither $x^* \leq x_{i-1} < x_i < x_{i+1}$ nor $x_{i-1} < x_i < x_{i+1} \leq x^*$. At these points F_i, $x_{i-1} < x^* < x_{i+1}$, and the MP algorithm may change the original $\dot{f}_i$, causing a loss of accuracy.

Note that in the MP-parabolic algorithm, both (2.3) and (2.9) are defined in terms of $s_{i-1/2}$ and $s_{i+1/2}$. Therefore, one may attempt to derive formulas for $\dot{f}_i$ which satisfy the MP constraint (2.7) *a priori*. Such an approach is presented below.

2.2. Limiter functions. For convenience of notation, in the rest of this section, denote $s_{i-1/2}$ and $s_{i+1/2}$ by s and t, respectively. Let

$$\dot{f}_i = G(s, t)$$

where G is a nonlinear averaging function called a *limiter function*. Fritsch and Butland [9] analyzed G as a function of two variables. We follow the analysis commonly used in conservation laws by first reducing G to a function of one variable, see e.g., [18], [20]–[22], [24], [26], [27]. The analysis below is more complete and leads to many new limiters. In order for G to give a result independent of units of measurement, it is necessary that G is homogeneous of degree one, that is, for any real number k,

$$G(ks, kt) = kG(s, t).$$

Substitute $k = 1/t$ into the above equation, one obtains

$$G(s,t) = tG(s/t, 1) = tG(r, 1) = tg(r) \tag{2.10}$$

where $r = s/t$ and $g(r) = G(r, 1)$. Expression (2.10) shows that G is completely determined by g; therefore, g is also called a limiter. Since G is an averaging function, the following properties for G are desirable (see also [9]):

a. G is symmetric, i.e., $G(s,t) = G(t,s)$. From (2.10), this is equivalent to

$$tg(r) = sg(1/r), \quad \text{or} \quad g(1/r) = (1/r)g(r). \tag{2.11}$$

b. $G(s,t)$ lies between s and t. Equivalently,

$$g(r) \in I\,[1, r], \tag{2.12}$$

i.e., the graph of g lies in the shaded region of Fig. 2a, which is called the second-order region (more precisely, sufficient for second-order accuracy region) for the following reason: since both s and t approximate $f'(x_i)$ to $O(h)$, the above condition implies the same is true for $\dot{f}_i$; consequently, the interpolant (2.2) is at least second-order accurate. Note that (2.12) implies $g(1) = 1$.

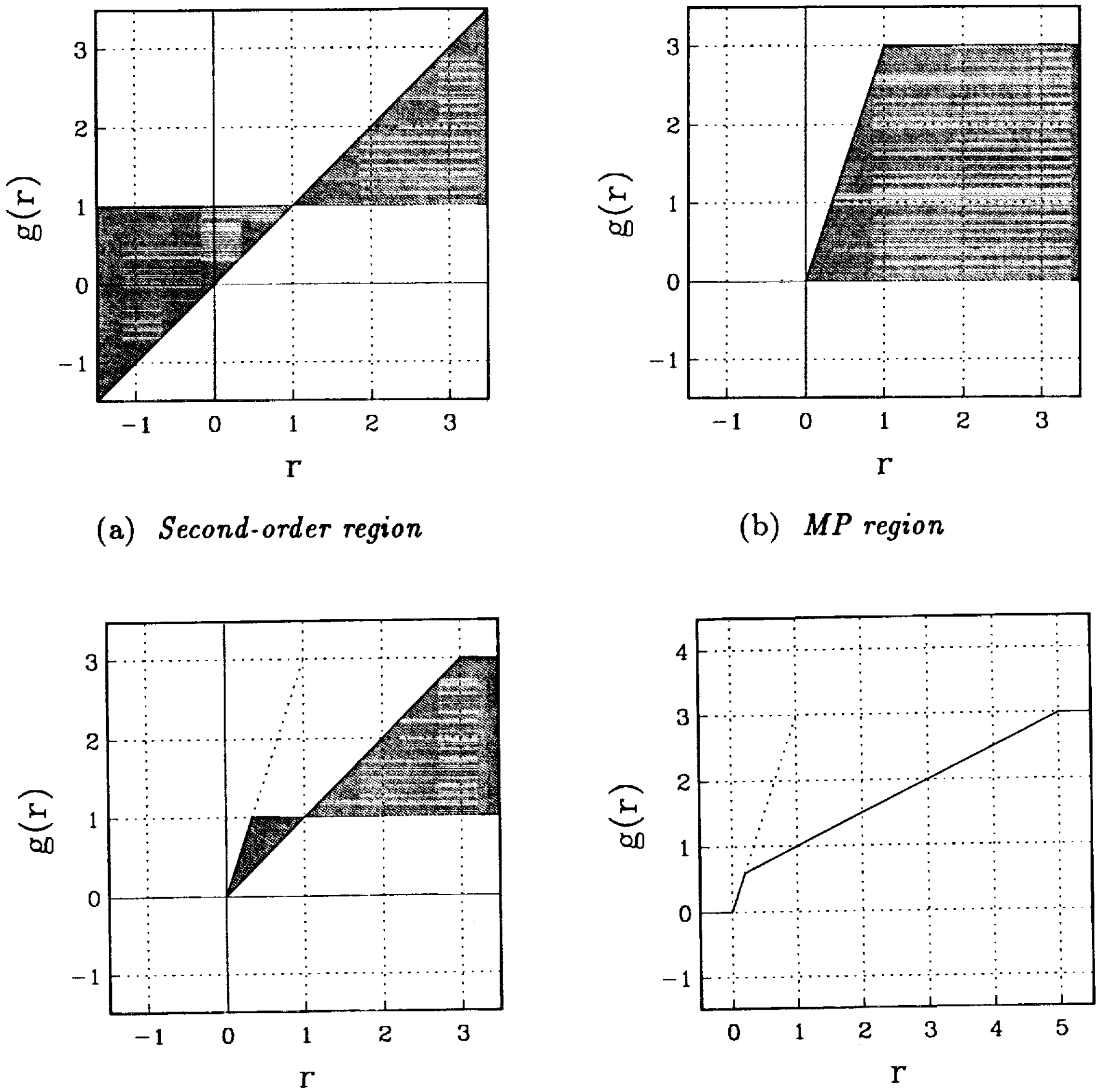
The next two conditions correspond to stability and monotonicity:

c. G or g is continuous,

d. $G(s,t)$ satisfies the MP constraint (2.7), i.e.,

$$g(r) \in I\,[0, 3\,\text{minmod}\,(1, r)]. \tag{2.13}$$

Observe that the above condition implies that if $r \le 0$, then $g(r) = 0$. For $r > 0$, it means that the graph of g lies in the shaded region of Fig. 2b. This region, together with the negative r-axis, is called the MP region.



(a) *Second-order region*

(b) *MP region*

(c) *MP2 region*

(d) *Average limiter*

Figs. 2. *Limiters.*

In this paper, unless otherwise stated, all limiters satisfy the first three of the above four conditions, and limiters in this subsection also satisfy the fourth. Note that in the context of conservation laws, limiters need not satisfy all of the above conditions, see e.g., [18], [23], [24].

For $r > 0$, condition (2.12) together with (2.13) are equivalent to the fact that the graph of g lies in the intersection of the shaded regions of Figs. 2a and 2b, which is shown in Fig. 2c. This region, together with the negative r-axis, is called the MP2 region. (The MP2 region is similar to the TVD2 region defined by Sweby [24]; the difference is that the constant 2 is replaced by 3.) Furthermore, for $r > 1$, $g(r)$ can be defined in terms of $g(r)$ where $0 < r < 1$ thanks to the symmetric property (2.11). Thus, the task of defining the limiter function G reduces to that of defining a continuous function $g(r)$ for $0 \leq r \leq 1$ such that $g(0) = 0$, $g(1) = 1$, and the graph of g lies in the region bounded by the three lines $\phi(r) = r$, $\phi(r) = 1$, and $\phi(r) = 3r$. Some popular limiter functions are listed below.

The minmod limiter [12], which corresponds to the lower boundary of the MP2 region, is defined by

$$g(r) = \text{minmod}\,(1, r), \quad \text{or} \quad G(s,t) = \text{minmod}\,(s,t). \tag{2.14}$$

To obtain the harmonic mean limiter, which was discovered independently by Van Leer [vL] and Butland [2], one represents g by a rational function of the form $ar/(br+1)$ where a and b are constants. If we assume $g'(0^+) = 2$, then $a = 2$; since $g(1) = 1$, $b = 1$. Thus,

$$g(r) = \frac{r + |r|}{1 + |r|}, \tag{2.15a}$$

or

$$G(s,t) = \begin{cases} 0 & \text{if } st \leq 0, \\ \dfrac{2st}{s+t} & \text{if } st > 0. \end{cases} \tag{2.15b}$$

Fritsch and Butland [9] observed that the harmonic mean limiter produces curves that are "too flat" because $g'(0^+) = 2$. They proposed the following limiter function with $g'(0^+) = 3$,

$$g(r) = \begin{cases} 0 & \text{if } r \leq 0, \\ \dfrac{3r}{1+2r} & \text{if } 0 < r \leq 1, \\ \dfrac{3r}{2+r} & \text{if } r > 1, \end{cases} \tag{2.16}$$

or

$$G(s,t) = \begin{cases} 0 & \text{if } st \leq 0, \\ \dfrac{3st}{2s+t} & \text{if } st > 0 \text{ and } |s| \leq |t|, \\ \dfrac{3st}{s+2t} & \text{if } st > 0 \text{ and } |s| > |t|. \end{cases}$$

The limiter which corresponds to the upper boundary of the MP2 region is similar to Roe's "Superbee" limiter [20], [24]:

$$g(r) = \min\{\max(1,r), 3\,\mathrm{minmod}\,(1,r)\}. \tag{2.17a}$$

The corresponding expression for $G(s,t)$ is obtained by first defining the maxmod function,

$$\begin{aligned} \mathrm{maxmod}\,(s,t) &= \frac{1}{2}\,[\mathrm{sgn}\,(s) + \mathrm{sgn}\,(t)]\,\max\,(|s|,|t|), \\ G(s,t) &= \mathrm{minmod}\,[\mathrm{maxmod}\,(s,t), 3\mathrm{minmod}\,(s,t)] \\ &= \frac{1}{2}\,[\mathrm{sgn}\,(s) + \mathrm{sgn}\,(t)]\min\,[\max(|s|,|t|), 3\min(|s|,|t|)]. \end{aligned} \tag{2.17b}$$

The above definition of the maxmod function is unstable when $s = 0$ or $t = 0$; however, since the limiter returns 0 in these cases, it is stable. Note that our definition of the maxmod function simplifies the expression of the "Superbee" limiter; furthermore, (2.17b) extends to higher-order cases in the next sections.

The minmod, the Fritsch-Butland, and the "Superbee" limiters are only first-order accurate because the slopes of these limiters are discontinuous at $r = 1$ (more on this later). Assume that g is piecewise C^1, then g' is continuous at $r = 1$ if and only if $g'(1^-) = 1/2$, or $g'(1^+) = 1/2$, or

$$g'(1) = 1/2. \tag{2.18}$$

Indeed, by differentiating (2.11),

$$g'(r) + \frac{1}{r}g'\left(\frac{1}{r}\right) = g\left(\frac{1}{r}\right),$$

and the above claim follows, since $g(1) = 1$. Furthermore, if g is piecewise C^2, then again by differentiating the above equation,

$$g''(r) = \frac{1}{r^3}g''\left(\frac{1}{r}\right).$$

Consequently, $g''(1^-) = g''(1^+)$, i.e., g'' is continuous at $r = 1$ *a priori*. We present below three limiter functions with $g'(1) = 1/2$ and $g'(0^+) = 3$.

The average limiter is defined by enforcing the constraint on the average, (see [26], [27] for the average limiter with $g'(0^+) = 2$),

$$g(r) = \mathrm{minmod}\left\{\frac{1+r}{2}, 3\,\mathrm{minmod}\,(1,r)\right\}, \tag{2.19}$$

or

$$G(s,t) = \text{minmod}\left\{\frac{s+t}{2}, 3\,\text{minmod}\,(s,t)\right\}.$$

The above limiter is the combination of Fromm's scheme [11] and the MP algorithm. Its graph is shown in Fig. 2d.

If we use a rational function of the form $(a_2r^2 + a_1r)/(r^2 + br + 1)$ to calculate g, then, in order that $g'(0) = 3$, $a_1 = 3$. Since g tends to 3 as r tends to ∞, $a_2 = 3$. Finally, $g(1) = 1$ implies $b = 4$,

$$g(r) = \begin{cases} 0 & \text{if } r \leq 0, \\ \dfrac{3r^2 + 3r}{r^2 + 4r + 1} & \text{if } r > 0, \end{cases} \tag{2.20}$$

or

$$G(s,t) = \begin{cases} 0 & \text{if } st \leq 0, \\ \dfrac{3st(s+t)}{s^2 + 4st + t^2} & \text{if } st > 0. \end{cases}$$

If a cubic of the form (2.2) is used, the result is

$$g(r) = \begin{cases} 0 & \text{if } r \leq 0, \\ \dfrac{3}{2}r^3 - \dfrac{7}{2}r^2 + 3r & \text{if } 0 < r \leq 1, \\ \dfrac{6r^2 - 7r + 3}{2r^2} & \text{if } r > 1. \end{cases} \tag{2.21}$$

Since the corresponding expression for $G(s,t)$ can be obtained in a similar manner as the above limiters, it is omitted.

Next, we show that the harmonic mean and the above three limiters are second-order accurate.

Lemma 2. *Suppose $f \in C^3$, and the data $\{f_i\}$ are third or higher-order accurate; the mesh is uniform; the limiter function g is continuous, piecewise C^2, and satisfies (2.11) and (2.12); then at the part where $f'(x) \neq 0$, $\dot{f}_i$ defined by g approximates the true derivative $f'(x_i)$ to second-order accuracy for small enough h if and only if $g'(1) = 1/2$.*

First, suppose $g'(1) = 1/2$. We will show that $\dot{f}_i$ is second-order. Since the mesh is uniform, the limiter $g_\alpha(r) = (r+1)/2$ yields identical $\dot{f}_i$ as the parabolic formula (2.3); consequently, it is second-order. Because both s and t approximate $f'(x_i) \neq 0$

to $O(h)$, if h is small enough, t is bounded away from 0, and $r - 1 = O(h)$. Since g_α and g have the same slopes at $r = 1$,

$$g(r) - g_\alpha(r) = O((r-1)^2) = O(h^2);$$

therefore, $\dot{f}_i$ defined by g is second-order.

Next, suppose $g'(1^-) = \beta \neq 1/2$. We will provide an example where the resulting $\dot{f}_i$ is only first-order accurate (the case $g'(1^+) \neq 1/2$ is similar). Let f_i be defined by the quadratic $f(x) = x^2 + x$, and let $x_i = 0$, $x_{i-1} = -h$, and $x_{i+1} = h$. Then $s = 1 - h$, $t = 1 + h$, $r = 1 - 2h + O(h^2)$. From (2.10),

$$\dot{f}_i = (1+h)[1 + \beta(r-1) + O(h^2)] = 1 + (1-2\beta)h + O(h^2).$$

Since $f'(x_i) = 1$ and $\beta \neq 1/2$, the above $\dot{f}_i$ is first-order. This completes the proof.

For an irregular mesh, all limiters are first-order accurate, while the MP-parabolic algorithm is still second-order away from strict local extrema. Near strict local extrema, however, the monotonicity constraint causes a loss of accuracy as shown below.

2.3. Accuracy and monotonicity. If the data are not monotone in $[x_i, x_{i+1}]$, the MP algorithm (2.9) still produces a monotone interpolant, which causes a loss of accuracy as in the following well-known example, see e.g., [6]. Let the data points be on a parabola such that $f_i = f_{i+1}$ as shown in Fig. 3; then the MP constraint (2.7) implies that $\dot{f}_i = \dot{f}_{i+1} = 0$. Therefore, $\dot{f}_i$ and $\dot{f}_{i+1}$ are only first-order accurate, and the cubic interpolant reduces to linear interpolation between f_i and f_{i+1}. Observe that the interpolant is still monotone in $[x_i, x_{i+1}]$ even though the data are not. One may attempt to turn off the monotonicity constraint near strict local extrema to avoid the above "clipping" phenomenon; however, this makes the method unstable in the sense that a small change in the data may cause a large change in the interpolant.

In [17], Hyman extended the MP constraint to

$$\dot{f}_i \in I\,[-3\min(|s|, |t|), 3\min(|s|, |t|)]. \qquad (2.22a)$$

Clearly, the above interval contains the MP interval and unless s or t equals 0, it is strictly larger. In terms of the ratio r, expression (2.22a) takes the form

$$g(r) \in I\,[-3\min(1, |r|), 3\min(1, |r|)], \qquad (2.22b)$$

i.e., the graph of g lies in the shaded region of Fig. 4a.

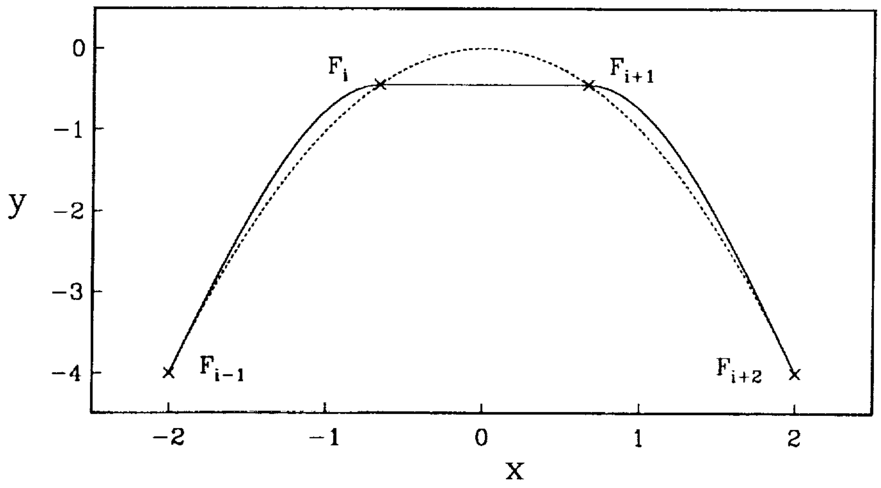
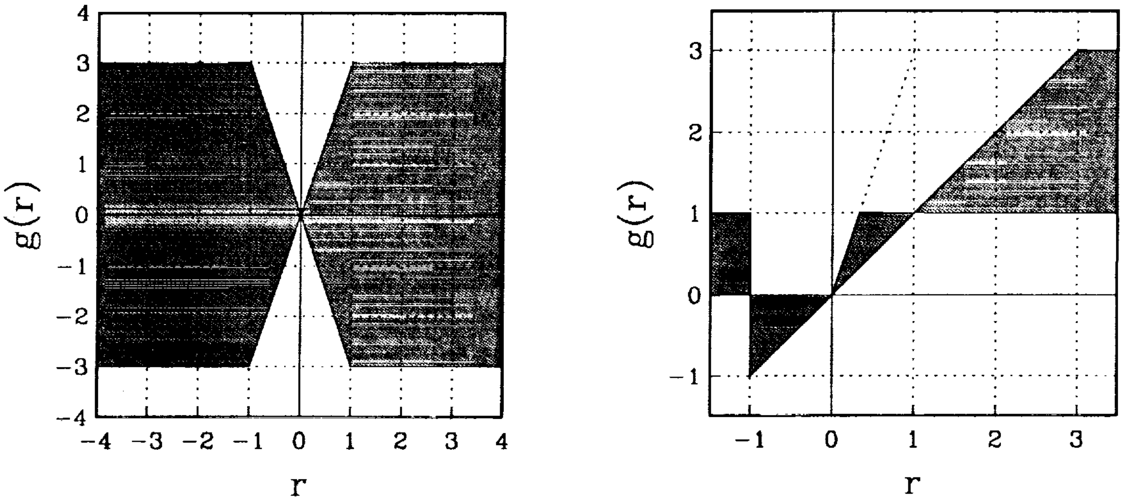


Fig. 3. *Loss of accuracy near strict local extrema.*



(a) *Hyman's extension*

(b) *M2 region*

Figs. 4. *Extensions of the monotonicity region.*

The following limiter function, which was introduced earlier by Van Albada, Van Leer, and Roberts [25], satisfies condition (2.22):

$$g(r) = \frac{r^2 + r}{r^2 + 1}, \tag{2.23a}$$

or

$$G(s,t) = \frac{s^2 t + s t^2}{s^2 + t^2}. \tag{2.23b}$$

Observe that as r tends to ∞, $g(r)$ tends to 1. It can be shown that for $r \geq 0$, the above limiter lies between the minmod and the harmonic mean limiters. Therefore, it shares the property of producing curves that are "too flat". On the positive side, adjacent to discontinuities, this limiter approximates $f'(x_i)$ better than any limiter mentioned in the previous subsection, as shown in the following example.

Let $x_i = i + 1/2$ for $i = -3, \ldots, 2$, and

$$f_i = -5 - x_i \text{ for } x_i < 0, \quad f_i = 5 - x_i \text{ for } x_i > 0; \tag{2.24}$$

i.e., the data lie on the lines $y = -5 - x$ for $x < 0$, and $y = 5 - x$ for $x > 0$. Any limiter function in the previous subsection gives $\dot{f}_{-1} = \dot{f}_0 = 0$. The exact solutions are $f'(-1/2) = f'(1/2) = -1$. Using the above limiter, one obtains $\dot{f}_{-1} = \dot{f}_0 = -72/82$ which are much closer to the exact solutions. In the context of conservation laws, however, this limiter, similar to the minmod, smears discontinuities.

The algorithm corresponding to (2.22a) is obtained by replacing $\dot{f}_i$ by the median of $\dot{f}_i$, $-3\min(|s|, |t|)$, and $3\min(|s|, |t|)$. The result is

$$\dot{f}_i \leftarrow \operatorname{sgn}(\dot{f}_i)\min(|\dot{f}_i|, 3|s|, 3|t|). \tag{2.25}$$

The above algorithm is very practical due to its simplicity and a slight gain of accuracy, see [6], [17], [19]. Theoretically, however, it is only as accurate as the MP algorithm: in the example at the beginning of this subsection where the data lie on a parabola, it gives the same result, i.e., it still produces only first-order accurate $\dot{f}_i$ near strict local extrema. Moreover, it may create oscillations as shown in the next examples.

Let $x_i = i$ for $i = 0, \ldots, 4$, and $f_i = i$ for $i = 0, \ldots, 3$. Let $f_4 = k \geq 4$, then the data are increasing. The quartic through these points and its slope at x_2 are (see also Hyman's fourth-order finite difference formula in [17])

$$p(x) = x + \frac{k-4}{24}x(x-1)(x-2)(x-3), \qquad \dot{f}_2 = \frac{8-k}{12}. \tag{2.26}$$

If $k = 20$, then $\dot{f}_2 = -1$, and Hyman's algorithm (2.25) does not change $\dot{f}_2$. We have a strictly negative slope in spite of the fact that the data are strictly increasing. In fact, even the MP algorithm (2.9) gives an "unrealistic" albeit monotone result of $\dot{f}_2 = 0$.

Note that in the above example, if one uses the parabolic formula (2.3), then $\dot{f}_2 = 1$, and both the MP and the Hyman algorithms work well in this case. In general, the combination of the parabolic formula and Hyman's algorithm may still create unwanted oscillations, as in example (2.24) where the parabolic formula yields $\dot{f}_{-1} = \dot{f}_0 = 4$, and the Hyman's algorithm then gives $\dot{f}_{-1} = \dot{f}_0 = 3$. The results are of the wrong sign compared to the exact solutions.

The above examples, especially (2.26), show that one has to be cautious in extending the MP interval (2.7) as well as in using higher-order methods to define $\dot{f}_i$. For limiter functions, an extension of the monotonicity constraint for $r < 0$ can be obtained as follows: first, the region must contain the negative r-axis; second, due to the symmetric property (2.11), $g(-1) = 0$; third, in example (2.24), $\dot{f}_{-1}$ and $\dot{f}_0$ should be negative, i.e., for $-1 < r < 0$, $g(r)$ should be negative, and for $-\infty < r < -1$, $g(r)$ should be positive. Together with (2.12), these conditions imply that the graph of g lies in the region bounded by $g = 0$, $g = 1$, $g = r$, and $r = -1$, as shown by the shaded region for $r \leq 0$ in Fig. 4b. For $r \geq 0$, the condition is the same as the MP2 condition in the previous subsection. This enlarged monotonicity region is named the M2 region shown by Fig. 4b. Limiters in this region preserve monotonicity in the sense that if the data are monotone in $[x_i, x_{i+1}]$, then so is the interpolant. This follows because monotonicity of the data in $[x_i, x_{i+1}]$ implies $s_{i-1/2}$, $s_{i+1/2}$, and $s_{i+3/2}$ are of the same sign. Adjacent to strict local extrema, these limiters do not simply define $\dot{f}_i$ as 0; however, they are still only first-order accurate. Two examples of such limiters are Van Albada's limiter (2.23), and

$$\begin{aligned} g(r) &= \operatorname{median}(1, r, -r-1) \\ &= 1 + \operatorname{minmod}(r-1, -r-2), \end{aligned} \tag{2.27}$$

or

$$\begin{aligned} G(s,t) &= \operatorname{median}(s, t, -s-t) \\ &= t + \operatorname{minmod}(s-t, -s-2t). \end{aligned}$$

Note that for $r \geq 0$, the above limiter is identical to the minmod. One can also combine it for $r < 0$ with any limiter in subsection 2.2 for $r \geq 0$.

Recently, in [6], a second-order accurate constraint for $\dot{f}_i$ was developed based on Hyman's algorithm. This method essentially detects whether the value f_i is at the left or the right of a strict local extremum; in each case, a different constraint is enforced. Like Hyman's method, it may produce derivatives of the wrong sign under extreme conditions. In the case of example (2.24), the results are $\dot{f}_{-1} = \dot{f}_0 = 3$. As for example (2.26), the result is identical to that of the MP algorithm, which is "unrealistic": $\dot{f}_2 = 0$. In the next section, we present a systematic approach in which all data points, except those at the boundaries, are constrained by the same conditions which extend that of De Boor and Swartz. These extensions produce third-order accurate interpolants, allow nonmonotone interpolants for nonmonotone data, and behave well in the extreme conditions of the above examples.

3. Third-order accurate monotone interpolants. To obtain uniform third-order accuracy, it is logical to consider the parabola $P_i(x) = (x, p_i(x))$ through the points F_{i-1}, F_i and F_{i+1}. Denote the second divided difference by d_i, i.e.,

$$d_i = f_{[x_{i-1},x_i,x_{i+1}]} = \frac{s_{i+1/2} - s_{i-1/2}}{x_{i+1} - x_{i-1}}. \tag{3.1}$$

Then from the Newton formula,

$$p_{i+\theta}(x) = f_i + s_{i+1/2}(x - x_i) + d_{i+\theta}(x - x_i)(x - x_{i+1}) \tag{3.2}$$

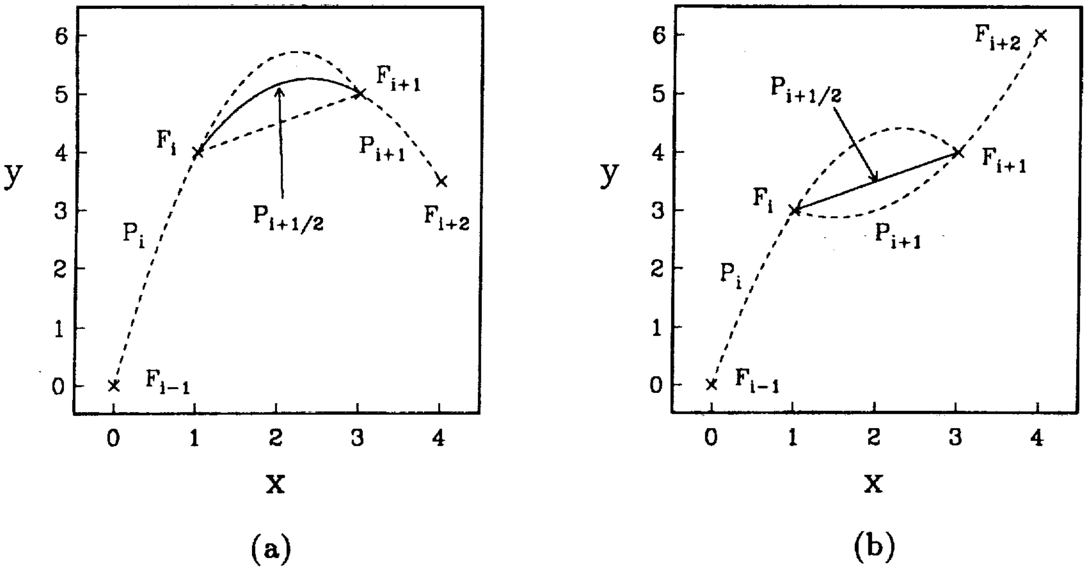
where $\theta = 0$ or $\theta = 1$. If $\{f_i\}$ are third or higher-order samples of a C^3 function f, then for all $x \in [x_{i-1}, x_{i+1}]$,

$$p_i(x) = f(x) + O(h^3), \qquad p_i'(x) = f'(x) + O(h^2). \tag{3.3}$$

3.1. Nonoscillatory parabolic interpolation. In the interval $[x_i, x_{i+1}]$, even when the data are monotone, neither p_i nor p_{i+1} are necessarily monotone. However, a monotone quadratic $p_{i+1/2}$ can be obtained by using Harten and Osher's nonoscillatory interpolation [15]. We present this method in our geometric framework below. Observe that the two parabolas P_i, P_{i+1}, and the line segment F_iF_{i+1} have the points F_i and F_{i+1} in common. Any two of these three curves are either identical or have no other point in common. Let $P_{i+1/2}$ be the curve which lies between the other two, i.e., the median of the above three curves, see Figs. 5a,b. Note that the equation for F_iF_{i+1} is given by (3.2) with $d_{i+\theta} = 0$. Let $d_{i+1/2}$ be the median of 0, d_i, and d_{i+1},

$$d_{i+1/2} = \mathrm{minmod}\,(d_i, d_{i+1}), \tag{3.4}$$

then the equation for $P_{i+1/2}$ is given by (3.2) with $\theta = 1/2$.



Figs. 5. *The parabolas* $P_{i+1/2}$.

Before proving the accuracy and monotonicity of $p_{i+1/2}$, we review some facts for the parabola $P(x) = (x, p(x))$ through F_i, F_{i+1} with second-divided difference (coefficient of x^2) d:

$$p(x) = f_i + s_{i+1/2}(x - x_i) + d(x - x_i)(x - x_{i+1}),$$

$$p'(x) = s_{i+1/2} + d(2x - x_i - x_{i+1}), \tag{3.5a}$$

$$p'(x_i) = s_{i+1/2} + d(x_i - x_{i+1}), \quad p'(x_{i+1}) = s_{i+1/2} + d(x_{i+1} - x_i). \tag{3.5b}$$

Denote

$$x_{i+1/2} = (x_i + x_{i+1})/2, \qquad f_{i+1/2} = (f_i + f_{i+1})/2, \tag{3.6}$$

then the two tangents to the parabola P at F_i and F_{i+1} intersect the vertical line $x = x_{i+1/2}$ at the same point $B_{i+1/2} = (x_{i+1/2}, b_{i+1/2})$ where

$$b_{i+1/2} = f_{i+1/2} - \frac{1}{2}d(\Delta x_{i+1/2})^2. \tag{3.7}$$

From (3.5a,b), and (3.7), p is monotone in $[x_i, x_{i+1}]$ if and only if any one of the following conditions holds:

$$|d| \leq \frac{|s_{i+1/2}|}{\Delta x_{i+1/2}}, \tag{3.8a}$$

$$p'(x_i) \in I\,[0, 2s_{i+1/2}], \tag{3.8b}$$

$$p'(x_{i+1}) \in I\,[0, 2s_{i+1/2}], \tag{3.8c}$$

$$b_{i+1/2} \in I\,[f_i, f_{i+1}]. \tag{3.8d}$$

To show accuracy for $p_{i+1/2}$, observe that since $d_{i+1/2}$ is the median of 0, d_i and d_{i+1}, it lies between d_i and d_{i+1}. Equations (3.2) with $\theta = 0$, 1, and 1/2 imply that $p_{i+1/2}(x)$ lies between $p_i(x)$ and $p_{i+1}(x)$ for all x. Similarly, equations (3.5a) with $d = d_i$, $d_{i+1/2}$, and d_{i+1} imply $p'_{i+1/2}(x)$ is the median of $s_{i+1/2}$, $p'_i(x)$, and $p'_{i+1}(x)$; therefore, $p'_{i+1/2}(x)$ lies between $p'_i(x)$ and $p'_{i+1}(x)$ for all x. As a consequence, it follows from (3.3) that for all $x \in [x_i, x_{i+1}]$,

$$p_{i+1/2}(x) = f(x) + O(h^3), \qquad p'_{i+1/2}(x) = f'(x) + O(h^2). \tag{3.9}$$

We turn now to monotonicity. Suppose the data are monotone in $[x_i, x_{i+1}]$. It will be shown that the same is true for $p_{i+1/2}$; therefore, (3.8a,b,c,d) hold for $p_{i+1/2}$. Consider for the moment the case $f_i \neq f_{i+1}$. By using a linear change of coordinates as in the proof of Lemma 1, we may assume $x_i = 0$, $x_{i+1} = 1$, and $f_i = 0$, $f_{i+1} = 1$. The data are thus increasing, and $f_{i-1} \leq 0$, $f_{i+2} \geq 1$. Because $s_{i+1/2} = 1$, $s_{i-1/2} \geq 0$, and $x_{i+1} - x_{i-1} > 1$, (3.1) implies $d_i < 1$. Similarly, $d_{i+1} > -1$. Since $d_{i+1/2}$ is the median of 0, d_i, and d_{i+1},

$$-1 < d_{i+1/2} < 1.$$

The above expression and (3.5a) imply

$$0 < p'_{i+1/2}(x) < 2 \tag{3.10}$$

for $x \in [0, 1]$. Consequently, $p_{i+1/2}$ is strictly increasing in $[0, 1]$. The case $f_i = f_{i+1}$ is trivial since the quadratic $p_{i+1/2}$ reduces to a constant function. This completes the proof.

Additional conclusions can be drawn if the data are monotone in $[x_i, x_{i+1}]$. Indeed, we will show that

$$p'_{i-1/2}(x_i)p'_{i+1/2}(x_i) \geq 0, \quad p'_{i+1/2}(x_{i+1})p'_{i+3/2}(x_{i+1}) \geq 0. \tag{3.11}$$

In the above proof, since $d_i < 1$ and $d_{i+1} > -1$, equations (3.5b) imply

$$p_i'(x_i) \geq 0, \qquad p_{i+1}'(x_{i+1}) \geq 0.$$

The facts that $s_{i-1/2} \geq 0$ and $s_{i+3/2} \geq 0$ then imply

$$p_{i-1/2}'(x_i) \geq 0, \qquad p_{i+3/2}'(x_{i+1}) \geq 0.$$

Expressions (3.11) follow from the above and (3.10). The case of $f_i = f_{i+1}$ is again trivial.

Furthermore, if the mesh is uniform, then in the proof of (3.10), $x_{i+1} - x_{i-1} = 2$, and the monotonicity of the data in $[x_i, x_{i+1}]$ implies

$$|d_{i+1/2}| \leq \frac{|s_{i+1/2}|}{2\Delta x_{i+1/2}}, \tag{3.12a}$$

$$p_{i+1/2}'(x) \in I\left[\frac{1}{2}s_{i+1/2}, \frac{3}{2}s_{i+1/2}\right] \tag{3.12b}$$

for all $x \in [x_i, x_{i+1}]$. The equal sign in (3.12a) and the closed interval in (3.12b) are necessary for the case $f_i = f_{i+1}$.

Similar to the definition of s_i, let

$$t_i = \text{minmod}\left[p_{i-1/2}'(x_i), p_{i+1/2}'(x_i)\right]. \tag{3.13}$$

From (3.9), t_i approximates $f'(x_i)$ to second-order accuracy. Note that the UNO scheme [15] is defined by setting $\dot{f}_i = t_i$. Loosely speaking, this scheme favors monotonicity over accuracy by defining $\dot{f}_i$ to be as close to 0 as possible from the above left and right slopes. It can also be considered as a high-order extension of the minmod limiter. In the next subsection, we present more accurate schemes which extend all limiters. Recall that in the second-order case presented in section 2, limiter functions simplify the algorithms by simultaneously satisfying the three conditions of symmetry (2.11), accuracy (2.12), and stability (continuity), together with the condition of monotonicity (2.13). In the third-order case below, the reverse is true: it is simpler to separate the former three conditions from the monotonicity condition. Consequently, for third or higher-order methods, limiters only need to satisfy these three conditions.

Uniform third-order monotonicity constraints are enforced in a separate step. In the next three subsections, we present three different third-order constraints.

3.2. M3 (monotone third-order) methods. To relax the monotonicity constraint of De Boor and Swartz, (2.6) is replaced by

$$\dot f_i \in I\left[0, 3s_{i+1/2}, \frac{3}{2}p'_{i+1/2}(x_i)\right], \qquad \dot f_{i+1} \in I\left[0, 3s_{i+1/2}, \frac{3}{2}p'_{i+1/2}(x_{i+1})\right]. \tag{3.14}$$

The above condition preserves monotonicity, i.e., if the data are monotone in $[x_i, x_{i+1}]$, and (3.14) holds, then the interpolant (2.2) is monotone in $[x_i, x_{i+1}]$. Indeed, by applying (3.8b,c) to $p_{i+1/2}$,

$$\frac{3}{2}p'_{i+1/2}(x_i),\ \frac{3}{2}p'_{i+1/2}(x_{i+1}) \in I\,[0, 3s_{i+1/2}]. \tag{3.15}$$

Therefore, the intervals in (3.14) reduce to those in (2.6), and monotonicity follows from Lemma 1. Next, expression (2.6′) is replaced by

$$\dot f_i \in I\left[0, 3s_{i-1/2}, \frac{3}{2}p'_{i-1/2}(x_i)\right], \qquad \dot f_i \in I\left[0, 3s_{i+1/2}, \frac{3}{2}p'_{i+1/2}(x_i)\right]; \tag{3.14′}$$

or $\dot f_i$ lies in the intersection of the above two intervals. The algorithm can be simplified by observing that the above intersection contains the interval $I\,[0, 3s_i, \frac{3}{2}t_i]$ since $\frac{3}{2}t_i$ lies in each of the intervals in (3.14′). Consequently, the MP constraint (2.7) is replaced by the following condition, which implies (3.14′),

$$\dot f_i \in I\left[0, 3s_i, \frac{3}{2}t_i\right]. \tag{3.16}$$

Observe that if the data are monotone in $[x_{i-1}, x_i]$ and $[x_i, x_{i+1}]$, then the above interval is identical to the MP interval. If the data are no longer monotone in $[x_{i-1}, x_i]$ or $[x_i, x_{i+1}]$, however, it may be strictly larger. The two end points of the interval in (3.16) can be obtained by taking the maximum and the minimum of the three arguments. Further simplification is made possible by showing that s_i and t_i are of the same sign. Indeed, $p'_{i-1/2}(x_i)$ is the median of $s_{i-1/2}$, $p'_i(x_i)$, and $p'_{i-1}(x_i)$; $p'_{i+1/2}(x_i)$ is the median of $s_{i+1/2}$, $p'_i(x_i)$, and $p'_{i+1}(x_i)$; consequently,

$$p'_{i-1/2}(x_i) \in I\,[s_{i-1/2}, p'_i(x_i)], \qquad p'_{i+1/2}(x_i) \in I\,[s_{i+1/2}, p'_i(x_i)]. \tag{3.17}$$

Because $p'_i(x_i) \in I\,[s_{i-1/2}, s_{i+1/2}]$ by (2.3), the above expressions imply

$$I\,[p'_{i-1/2}(x_i), p'_{i+1/2}(x_i)] \subset I\,[s_{i-1/2}, s_{i+1/2}].$$

Since $t_i = \text{median}\,(0, p'_{i-1/2}(x_i), p'_{i+1/2}(x_i))$, and $s_i = \text{median}\,(0, s_{i-1/2}, s_{i+1/2})$,

$$I\,[0, t_i] \supset I\,[0, s_i].$$

Consequently, s_i and t_i are of the same sign, and if $t_i = 0$, then $s_i = 0$. Expression (3.16), therefore, is identical to

$$\dot{f}_i \in I\left[0, \text{sgn}\,(t_i) \max(3|s_i|, \frac{3}{2}|t_i|)\right]. \tag{3.16'}$$

Note that if $t_i = 0$, it does not matter whether $\text{sgn}\,(t_i)$ is defined as -1, 0, or 1 in the above expression since the interval reduces to the point $\{0\}$.

We summarize the resulting algorithm below.

Theorem 1. *Let $s_{i+1/2}$, d_i be the first and second divided differences as in (2.1), (3.1). Let s_i, $d_{i+1/2}$ be the corresponding minmod defined by (2.8) and (3.4). Recall the slopes of the parabolas (3.5b) and expression (3.13),*

$$p'_{i-1/2}(x_i) = s_{i-1/2} + d_{i-1/2}(x_i - x_{i-1}),$$

$$p'_{i+1/2}(x_i) = s_{i+1/2} + d_{i+1/2}(x_i - x_{i+1}),$$

$$t_i = \text{minmod}\,[p'_{i-1/2}(x_i), p'_{i+1/2}(x_i)].$$

In addition, for $m + 2 \le i \le n - 2$, let the original $\dot{f}_i$ be defined by any continuous limiter function with property (2.12) applied to the left and right slopes $p'_{i-1/2}(x_i)$ and $p'_{i+1/2}(x_i)$, e.g., the arithmetic mean,

$$\dot{f}_i = \frac{1}{2}\,[p'_{i-1/2}(x_i) + p'_{i+1/2}(x_i)]. \tag{3.18}$$

The original $\dot{f}_i$ can also be defined by any second or higher-order formula for the derivative. Let the final $\dot{f}_i$ be defined by

$$t_{\max} = \text{sgn}\,(t_i) \max\left(3|s_i|, \frac{3}{2}|t_i|\right), \tag{3.19a}$$

$$\dot{f}_i \leftarrow \text{minmod}\,(\dot{f}_i, t_{\max}). \tag{3.19b}$$

Then the algorithm is stable. It preserves monotonicity, i.e., if the data are monotone in $[x_i, x_{i+1}]$ for any i, $m+2 \le i \le n-3$, then so is the interpolant. Furthermore, if $f \in C^3$, and $\{f_i\}$ are third or higher-order accurate, the final $\{\dot{f}_i\}$ are second-order; consequently, the interpolant (2.2) is third-order.

The above method is named M3-A (A stands for average) if (3.18) is used. The proof is now straightforward. First, stability follows since each of the above equations is stable. Next, monotonicity is preserved since (3.19a,b) imply (3.16), which in turn implies (3.14′) and (3.14). As for accuracy, observe that the original $\dot{f}_i$ defined by (3.18) or by a limiter with property (2.12) is second-order accurate. The interval in (3.16′) contains the value t_i, which is also second-order accurate. Any value which lies between the original $\dot{f}_i$ and t_i is therefore second-order, in particular, the final $\dot{f}_i$ defined by (3.19a,b).

Before presenting the next method, note the following. By considering the case $p'_{i-1/2}(x_i)$ and $p'_{i+1/2}(x_i)$ of the same sign or of opposite sign, the M3-A method (3.18), (3.19a,b) can be simplified to:

$$\dot{f}_i = \text{sgn}\,(t_i)\ \min\left[\frac{1}{2}|p'_{i-1/2}(x_i) + p'_{i+1/2}(x_i)|, \max\left(3|s_i|, \frac{3}{2}|t_i|\right)\right].$$

If the "Superbee" limiter is used, the resulting M3-B method can be expressed as

$$\dot{f}_i = \text{sgn}\,(t_i)\ \min\left[\max\left(|p'_{i-1/2}(x_i)|, |p'_{i+1/2}(x_i)|\right), \max\left(3|s_i|, \frac{3}{2}|t_i|\right)\right]. \tag{3.20}$$

Observe that in the above theorem, if one sets $d_i = 0$ for all i, then the M3-A and M3-B methods respectively reduce to the average and "Superbee" limiters.

The constraint (3.19a,b) has a five point stencil from $i-2$ to $i+2$. By using the quartic though these five data points (see formula (4.21)), one can obtain a fourth-order accurate $\dot{f}_i$. As shown in example (2.26), such $\dot{f}_i$ may have the wrong sign. This problem can be solved by enforcing the following condition which is a higher-order extension of the accuracy condition (2.12): $\dot{f}_i$ lies between $p'_{i-1/2}(x_i)$ and $p'_{i+1/2}(x_i)$, or

$$\dot{f}_i \leftarrow \text{median}\,(\dot{f}_i, p'_{i-1/2}(x_i), p'_{i+1/2}(x_i)). \tag{3.21}$$

The final $\dot{f}_i$ is then obtained by (3.19a,b). This method is named M3-quartic.

If the original $\dot{f}_i$ is defined by a limiter function which is bounded above by 3/2, then (3.19a,b) are redundant because the original $\dot{f}_i$ satisfies (3.16) *a priori*.

If the data represent a discontinuous function, one can replace (3.18) and (3.19a,b) by applying Van Albada's limiter (2.23) to $p'_{i-1/2}(x_i)$, $p'_{i+1/2}(x_i)$, and the resulting M3-V method is more accurate near discontinuities. (Incidentally, the graph of this limiter has a *V* shape.) It is also highly accurate near the smooth parts of the data because the limiter (2.23) satisfies $g'(1) = 1/2$. As for monotonicity, suppose that the data are monotone in $[x_i, x_{i+1}]$, then from (3.11), and the fact that this limiter is bounded above by $g(1+\sqrt{2}) = (1+\sqrt{2})/2 < 3/2$, one concludes that condition (3.16) is satisfied at x_i and x_{i+1}. Consequently, the M3-V method preserves monotonicity even though (3.19a,b) are omitted. Notice that with this method, condition (3.16) may not hold if $p'_{i-1/2}(x_i)p'_{i+1/2}(x_i) < 0$; however, an extension of (3.16), which is similar to the extension from the MP2 (Fig. 2c) to the M2 (Fig. 4b) constraint, holds in this case. Lastly, as in Theorem 1, accuracy and stability are trivial.

If the mesh is uniform, and the original $\dot{f}_i$ is defined by a limiter function which is bounded above by 2, e.g., the harmonic mean limiter, then it follows from (3.12b) that (3.19a,b) are redundant because the original $\dot{f}_i$ satisfies (3.16) *a priori*.

The algorithm of the above theorem is an extension of the author's SONIC schemes developed for conservation laws [16]. The results there show that the algorithm improves accuracy not only near strict local extrema, but also elsewhere (see subsection 3.5 below). Note that there are some similarities in the techniques of the SONIC schemes and (independently) Yang's recent artificial compression by slope modification [28]. The concepts, however, are very different: Yang's method modifies $\dot{f}_i = t_i$, the slopes of the UNO scheme defined in (3.13), by an adjustable factor obtained from numerical experiments; our methods define constraints which can be applied to any high-order accurate derivative.

3.3. MS3 (monotone simple third-order) constraint. To develop the second third-order constraint, we first define the minmod function with more than two arguments. Let $z_1, \ldots, z_k$ be real numbers, and

$$\alpha = \min(z_1, \ldots, z_k), \qquad \beta = \max(z_1, \ldots, z_k).$$

Let $\text{minmod}(z_1, \ldots, z_k)$ be the function which returns the smallest argument if all arguments are positive, the largest if they are all negative, and 0 otherwise. Then

$$\text{minmod}(z_1, \ldots, z_k) = \text{minmod}(\alpha, \beta).$$

The minmod function can also be expressed as

$$\text{minmod}(z_1, \ldots, z_k) = \lambda \min(|z_1|, \ldots, |z_k|)$$

where

$$\begin{aligned}\lambda &= \frac{1}{2}[\text{sgn}(z_1) + \text{sgn}(z_2)]\,\frac{1}{2}[\text{sgn}(z_1) + \text{sgn}(z_3)]\ldots\frac{1}{2}[\text{sgn}(z_1) + \text{sgn}(z_k)] \\ &= \frac{1}{2}[\text{sgn}(z_1) + \text{sgn}(z_2)]\,\frac{1}{2}[\text{sgn}(z_2) + \text{sgn}(z_3)]\ldots\frac{1}{2}[\text{sgn}(z_{k-1}) + \text{sgn}(z_k)].\end{aligned}$$

Observe that in the above expression for the minmod function, if $z_i = 0$, it does not matter whether $\text{sgn}(z_i)$ is defined as -1, 0, or 1. One can also take successive minmods to arrive at the same result, e.g.,

$$\text{minmod}(z_1, z_2, z_3) = \text{minmod}[\text{minmod}(z_1, z_2), z_3].$$

Notice that if $k = 2$, these definitions are consistent with those in section 2. The case of three arguments can be applied to the MP algorithm (2.9).

The second method to enlarge the MP interval is similar to the first method, except t_i in (3.16) is replaced by u_i:

$$\dot{f}_i \in I\left[0, 3s_i, \frac{3}{2}u_i\right] \tag{3.22}$$

where

$$u_i = \text{minmod}\left[p_i'(x_i), p_{i-1}'(x_i), p_{i+1}'(x_i)\right]. \tag{3.23}$$

Expression (3.22) can be simplified by showing that s_i, u_i, and $p_i'(x_i)$ are of the same sign. Indeed, since $p_i'(x_i)$ lies in the interval $I[s_{i-1/2}, s_{i+1/2}]$,

$$I[0, s_i] \subset I[0, p_i'(x_i)].$$

This implies s_i and $p_i'(x_i)$ are of the same sign, and if $p_i'(x_i) = 0$, then $s_i = 0$. Similar conclusions hold for u_i and $p_i'(x_i)$. Consequently,

$$I\left[0, 3s_i, \frac{3}{2}u_i\right] = I\left[0, \text{sgn}(p_i'(x_i)) \max\left(3|s_i|, \frac{3}{2}|u_i|\right)\right]. \tag{3.22$'$}$$

The corresponding algorithm is summarized below.

Corollary 1. *Let $s_{i+1/2}$, d_i be the first and second divided differences as in (2.1), (3.1). Let $\{p_i'(x_i), p_{i-1}'(x_i), p_{i+1}'(x_i)\}$ be the slopes of the parabolas given by (3.5b). Let s_i and u_i be the corresponding minmod defined by (2.8) and (3.23). Let $\dot{f}_i$ be any second, or higher-order approximation of $f'(x_i)$, and for $m+2 \le i \le n-2$, let the final $\dot{f}_i$ be defined by*

$$t_{\max} = \operatorname{sgn}(p_i'(x_i)) \max\left(3|s_i|, \frac{3}{2}|u_i|\right),$$

$$\dot{f}_i \leftarrow \operatorname{minmod}(\dot{f}_i, t_{\max}).$$

Then the algorithm is stable and preserves monotonicity, and the interpolant (2.2) is third-order accurate.

The stability of the above method is trivial. The proof of accuracy is similar to that of Theorem 1 with the observation that u_i is second-order accurate. Next, it will be shown that

$$I\left[0, 3s_i, \frac{3}{2}u_i\right] \subset I\left[0, 3s_i, \frac{3}{2}t_i\right]; \tag{3.24}$$

consequently, monotonicity follows. Indeed, since $p_{i-1/2}'(x_i) \in I\,[p_i'(x_i), p_{i-1}'(x_i)]$, and $p_{i+1/2}'(x_i) \in I\,[p_i'(x_i), p_{i+1}'(x_i)]$,

$$I\,[p_{i-1/2}'(x_i), p_{i+1/2}'(x_i)] \subset I\,[p_i'(x_i), p_{i-1}'(x_i), p_{i+1}'(x_i)];$$

therefore, $I\,[0, t_i] \supset I\,[0, u_i]$, and (3.24) holds. This completes the proof.

The above algorithm can be simplified further by defining

$$u_i = \min(|p_i'(x_i)|, |p_{i-1}'(x_i)|, |p_{i+1}'(x_i)|)$$

instead of (3.23), and the same conclusions still hold. A drawback of this simplification is that in example (2.24), if the original $\dot{f}_i$ is defined by $p_i'(x_i)$, then the results $\dot{f}_{-1} = \dot{f}_0 = 3$ are of the wrong sign.

Notice that due to (3.24), the method of the above corollary is not as general as that of Theorem 1. The next method, however, is more general than the above two.

3.4. MG3 (monotone general third-order) constraint. Using the four data points F_{i-1}, F_i, F_{i+1} and F_{i+2}, the monotonicity-preserving parabola $P_{i+1/2}$ defined in subsection 3.1 is, loosely speaking, "closest" to the straight line segment F_iF_{i+1}. We begin this subsection by defining a monotonicity-preserving parabola $\bar{P}_{i+1/2}$ which is "furthest" from the line segment. Denote $F_{i+1/2} = (x_{i+1/2}, f_{i+1/2})$ where $x_{i+1/2}$ and $f_{i+1/2}$ are defined by (3.6). Let $F^L_{i+1/2} = (x_{i+1/2}, f^L_{i+1/2})$ (or $F^R_{i+1/2}$) be the intersection of the line $F_{i-1}F_i$ (or $F_{i+1}F_{i+2}$) and the vertical line $x = x_{i+1/2}$ as shown in Fig. 6, then

$$f^L_{i+1/2} = f_i + \frac{1}{2}s_{i-1/2}\Delta x_{i+1/2}, \qquad f^R_{i+1/2} = f_{i+1} - \frac{1}{2}s_{i+3/2}\Delta x_{i+1/2} \tag{3.25}$$

where the superscripts L and R stand for extrapolation from left and right, respectively.

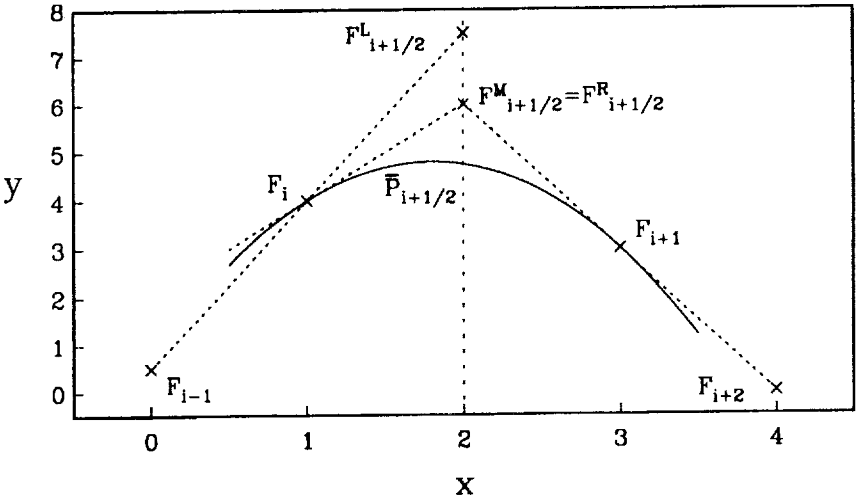


Fig. 6. *The parabola* $\bar{P}_{i+1/2}$.

Let $\bar{P}^L$ (or $\bar{P}^R$) be the parabola through F_i, F_{i+1}, whose tangent at F_i (or F_{i+1}) is the line $F_{i-1}F_i$ (or $F_{i+1}F_{i+2}$). Similar to the definition of $P_{i+1/2}$, let $\bar{P}_{i+1/2}$ be the median of the parabolas $\bar{P}^L$, $\bar{P}^R$, and the line segment F_iF_{i+1}. Observe that $\bar{d}^L$, $\bar{d}^R$, which are the second divided differences of $\bar{P}^L$, $\bar{P}^R$, can be obtained by expressing the slopes of the tangents $F_{i-1}F_i$, $F_{i+1}F_{i+2}$ in the form (3.5b),

$$\bar{d}^L = \frac{s_{i+1/2} - s_{i-1/2}}{\Delta x_{i+1/2}}, \qquad \bar{d}^R = \frac{s_{i+3/2} - s_{i+1/2}}{\Delta x_{i+1/2}}. \tag{3.26}$$

Let

$$\bar{d}_{i+1/2} = \text{minmod}\,(\bar{d}^L, \bar{d}^R), \tag{3.27a}$$

then $\bar{d}_{i+1/2}$ is the second divided difference of $\bar{P}_{i+1/2}$, and from (3.26),

$$\bar{d}_{i+1/2} = \frac{\bar{s}_{i+1/2}}{\Delta x_{i+1/2}} \tag{3.27b}$$

where

$$\bar{s}_{i+1/2} = \operatorname{minmod}(s_{i+1/2} - s_{i-1/2}, s_{i+3/2} - s_{i+1/2}). \tag{3.28}$$

Observe that by (3.1) and (3.26), d_i lies between 0 and $\bar{d}^L$; d_{i+1} lies between 0 and $\bar{d}^R$; consequently, by (3.4) and (3.27a),

$$d_{i+1/2} \in I\,[0, \bar{d}_{i+1/2}]. \tag{3.29}$$

Note that if $p_{i+1/2}$ has a strict local extremum in (x_i, x_{i+1}), then the same is true for $\bar{p}_{i+1/2}$.

Next, define

$$f^M_{i+1/2} = \operatorname{median}(f_{i+1/2}, f^L_{i+1/2}, f^R_{i+1/2}), \tag{3.30}$$

then $F^M_{i+1/2} = (x_{i+1/2}, f^M_{i+1/2})$ is the intersection of the tangents to $\bar{P}_{i+1/2}$ at F_i and F_{i+1}. The slopes of these tangents are given by (3.5b),

$$\bar{p}'_{i+1/2}(x_i) = s_{i+1/2} - \bar{s}_{i+1/2}, \tag{3.31a}$$

$$\bar{p}'_{i+1/2}(x_{i+1}) = s_{i+1/2} + \bar{s}_{i+1/2}. \tag{3.31b}$$

Using definition (3.30), observe that if $f^M_{i+1/2}$ is a *strict local maximum*, i.e., $f^M_{i+1/2} > f_i$ and $f^M_{i+1/2} > f_{i+1}$, then from (3.25), the data $\{f_i\}$ have a *strict local maximum in* $[x_i, x_{i+1}]$, i.e.,

$$f_i > f_{i-1}, \quad \text{and} \quad f_{i+1} > f_{i+2}, \tag{3.32}$$

see Fig. 6. A similar statement holds for the minimum. Consequently, if $f^M_{i+1/2}$ is a *strict local extremum*, then the data $\{f_i\}$ have a *strict local extremum in* $[x_i, x_{i+1}]$. In particular, if the data are monotone in $[x_i, x_{i+1}]$, then $f^M_{i+1/2}$ is not a strict local extremum, i.e., $f^M_{i+1/2}$ lies between f_i and f_{i+1}. Therefore, by (3.8d), $\bar{P}_{i+1/2}$ is monotone in $[x_i, x_{i+1}]$, and by (3.8b,c),

$$\bar{p}'_{i+1/2}(x_i) \in I\,[0, 2s_{i+1/2}], \quad \bar{p}'_{i+1/2}(x_{i+1}) \in I\,[0, 2s_{i+1/2}]. \tag{3.33}$$

Similar to (3.14), (3.14′), we can now relax the constraints (2.6) and (2.6′) to

$$\dot{f}_i \in I\left[0, 3s_{i+1/2}, \frac{3}{2}\bar{p}'_{i+1/2}(x_i)\right], \quad \text{and} \quad \dot{f}_{i+1} \in I\left[0, 3s_{i+1/2}, \frac{3}{2}\bar{p}'_{i+1/2}(x_{i+1})\right]; \tag{3.34}$$

$$\dot{f}_i \in I\left[0, 3s_{i-1/2}, \frac{3}{2}\bar{p}'_{i-1/2}(x_i)\right], \quad \text{and} \quad \dot{f}_i \in I\left[0, 3s_{i+1/2}, \frac{3}{2}\bar{p}'_{i+1/2}(x_i)\right]. \qquad (3.34')$$

The intersection of the two intervals in (3.34′), denoted by $[\alpha_i, \beta_i]$, is nonempty (contains 0) and can be obtained by

$$\alpha_i^L = \min\left[0, 3s_{i-1/2}, \frac{3}{2}\bar{p}'_{i+1/2}(x_i)\right], \quad \beta_i^L = \max\left[0, 3s_{i-1/2}, \frac{3}{2}\bar{p}'_{i+1/2}(x_i)\right], \quad (3.35a)$$

$$\alpha_i^R = \min\left[0, 3s_{i+1/2}, \frac{3}{2}\bar{p}'_{i+1/2}(x_i)\right], \quad \beta_i^R = \max\left[0, 3s_{i+1/2}, \frac{3}{2}\bar{p}'_{i+1/2}(x_i)\right], \quad (3.35b)$$

$$\alpha_i = \max(\alpha_i^L, \alpha_i^R), \qquad \beta_i = \min(\beta_i^L, \beta_i^R). \qquad (3.35c)$$

Notice that a simplification similar to (3.16) does not work in this case because it may cause a loss of accuracy. To bring $\dot{f}_i$ into the interval $[\alpha_i, \beta_i]$, again we replace $\dot{f}_i$ by median $(\dot{f}_i, \alpha_i, \beta_i)$:

$$\dot{f}_i \leftarrow \dot{f}_i + \text{minmod}(\alpha_i - \dot{f}_i, \beta_i - \dot{f}_i). \qquad (3.36)$$

Next, we prove that the above algorithm gives a second-order accurate $\dot{f}_i$ if the original $\dot{f}_i$ is calculated by a second or higher-order method. In fact, we will show that

$$[\alpha_i, \beta_i] \supset I\left[0, 3s_{i+1/2}, \frac{3}{2}t_i\right], \qquad (3.37)$$

which implies $t_i \in [\alpha_i, \beta_i]$, and accuracy follows as in the proof of Theorem 1. To show (3.37), it suffices to show that the intervals in (3.34) contain the corresponding ones in (3.14), i.e.,

$$I\left[0, 3s_{i+1/2}, \frac{3}{2}\bar{p}'_{i+1/2}(x_i)\right] \supset I\left[0, 3s_{i+1/2}, \frac{3}{2}p'_{i+1/2}(x_i)\right], \qquad (3.38a)$$

$$I\left[0, 3s_{i+1/2}, \frac{3}{2}\bar{p}'_{i+1/2}(x_{i+1})\right] \supset I\left[0, 3s_{i+1/2}, \frac{3}{2}p'_{i+1/2}(x_{i+1})\right]. \qquad (3.38b)$$

Indeed, from (3.29),

$$p'_{i+1/2}(x_i) \in I\left[s_{i+1/2}, \bar{p}'_{i+1/2}(x_i)\right], \qquad p'_{i+1/2}(x_{i+1}) \in I\left[s_{i+1/2}, \bar{p}'_{i+1/2}(x_{i+1})\right], \quad (3.39)$$

which imply (3.38a,b). This completes the proof of accuracy.

The above method is summarized below.

Theorem 2. *Let α_i, β_i be defined as in (3.35a,b,c) using (3.28), (3.31a,b), and let the original $\{\dot{f}_i\}$ be calculated by (2.3) or some higher-order method for $m+2 \leq i \leq n-2$. Let the final $\{\dot{f}_i\}$ be obtained by (3.36). Then the algorithm is stable and monotonicity-preserving in $[x_i, x_{i+1}]$, $m+2 \leq i \leq n-3$. Moreover, if $f \in C^3$, and $\{f_i\}$ are third or higher-order accurate, then the final $\{\dot{f}_i\}$ are second-order; consequently, the interpolant (2.2) is third-order.*

We conclude this subsection with the following remarks:

Observe that a simplification similar to (3.16) may cause a loss of accuracy since in spite of (3.39), the interval $I\,[0, \text{minmod}\,(\bar{p}'_{i-1/2}(x_i), \bar{p}'_{i+1/2}(x_i))]$ may not contain $I\,[0, t_i]$.

Due to (3.37), the above method is more general than the previous two. As will be shown in the next subsection, the algorithm still provides "plenty of room" for $\dot{f}_i$ near strict local extrema if the coefficients 3/2 in expression (3.35a,b) are replaced by 1.

Suppose $\hat{d}_{i+1/2}$ is an accurate, second divided difference obtained by evaluating, e.g., the second derivative at $x = x_{i+1/2}$ of the cubic through F_{i-1}, F_i, F_{i+1}, and F_{i+2}. Then, in order that the corresponding parabola $\hat{P}_{i+1/2}$ is monotonicity-preserving, one replaces $\hat{d}_{i+1/2}$ by minmod$\,(\hat{d}_{i+1/2}, \bar{d}_{i+1/2})$. Note that Theorem 1 with $d_{i+1/2}$ replaced by $\hat{d}_{i+1/2}$ is still valid.

In practice, one can often avoid the calculations in the above theorem by using one of the following two tests:

1. Checking if $\dot{f}_i$ lies in the MP interval by testing if

$$\dot{f}_i(\dot{f}_i - 3s_i) \leq 0.$$

   If this condition holds, one moves on to the next point. Otherwise, one goes through the calculations of the above theorem.

2. Checking if the data have a strict local extremum in $[x_i, x_{i+1}]$ by testing if

$$(f_i - f_{i-1})(f_{i+1} - f_{i+2}) > 0.$$

   If the data have neither a strict local extremum in $[x_{i-1}, x_i]$, nor one in $[x_i, x_{i+1}]$, we simply use the MP algorithm (2.9).

3.5. More on the accuracy of third-order methods. Observe that if the data are on a parabola, then any of the above three methods M3, MS3, and MG3 reproduces the parabola (if the original $\{\dot{f}_i\}$ are calculated by (2.3) for the last two methods). However, if the data are on a cubic, these methods may not reproduce the cubic even if the original $\{\dot{f}_i\}$ are exact. We show an example of this case below.

Let $x_i = i$ for $i = -2, \ldots, 2$, and

$$f_{-1} = f_0 = f_1 = 0, \quad f_{-2} = -6, \quad \text{and} \quad f_2 = 6, \tag{3.40}$$

i.e., the data are on the cubic $f(x) = x^3 - x$. Clearly, the parabola P_0 through F_{-1}, F_0, and F_1 is identical to the x-axis. Since the interval $[\alpha_0, \beta_0]$ reduces to $\{0\}$, any of the above three algorithms yields $\dot{f}_0 = 0$. The exact solution is $f'(0) = -1$, and the algorithms do not reproduce the cubic. If h is small enough, however, the MG3 constraint reproduces the above cubic as shown below.

More generally, suppose the mesh is uniform, the data are fourth or higher-order samples of a C^4 function f, and the original $\{\dot{f}_i\}$ are third or higher-order accurate, then when do the above third-order constraints cause a loss of fourth-order accuracy? As shown at the end of subsection 2.1, at the part where $f'(x) \neq 0$, the MP algorithm, and therefore all third-order algorithms, does not alter $\dot{f}_i$ provided h is small enough; consequently, accuracy is preserved. Next, we show that adjacent to a strict local extremum at $x = x^*$ where $f''(x^*) \neq 0$, the M3 and MS3 constraints are essentially identical, and they may yield only second-order accurate $\dot{f}_i$. The MG3 constraint, on the other hand, does not alter $\dot{f}_i$ in this case, and thus, preserves accuracy. Indeed, for any x_i, after a linear change of coordinates, we may assume $x_i = 0$, and the Taylor series expansion of the exact solution at $x = x_i$ is of the form

$$f(x) = sx + dx^2 + ex^3 + O(h^4). \tag{3.41}$$

Higher-order terms will be omitted below when there is no ambiguity. By evaluating $f_{i-2}, \ldots, f_{i+2}$, and using (2.1), (3.1), and (3.5b),

$$s_{i-3/2} = s - 3dh + 7eh^2, \qquad s_{i+3/2} = s + 3dh + 7eh^2, \tag{3.42a}$$

$$s_{i-1/2} = s - dh + eh^2, \qquad s_{i+1/2} = s + dh + eh^2, \tag{3.42b}$$

$$d_{i-1} = d - 3eh, \qquad d_i = d + O(h^2), \qquad d_{i+1} = d + 3eh, \tag{3.43}$$

$$p'_{i-1}(x_i) = s - 2eh^2, \qquad p'_i(x_i) = s + eh^2, \qquad p'_{i+1}(x_i) = s - 2eh^2. \tag{3.44}$$

Since $f''(x^*) \neq 0$, in a small enough neighborhood of x^*, we may assume $d \neq 0$. The following equations (3.45–50) only use the fact that $d \neq 0$. Suppose h is small enough so that

$$|d| > 3|e|h. \tag{3.45}$$

If $de \geq 0$, then from (3.4), (3.5b), (3.27b), and (3.31a,b),

$$d_{i-1/2} = d - 3eh, \qquad d_{i+1/2} = d, \tag{3.46}$$

$$p'_{i-1/2}(x_i) = s - 2eh^2, \qquad p'_{i+1/2}(x_i) = s + eh^2, \tag{3.47}$$

$$\bar{d}_{i-1/2} = 2d - 6eh, \qquad \bar{d}_{i+1/2} = 2d. \tag{3.48}$$

$$\bar{p}'_{i-1/2}(x_i) = s + dh - 5eh^2, \qquad \bar{p}'_{i+1/2}(x_i) = s - dh + eh^2. \tag{3.49}$$

Note that since the mesh is uniform, $\bar{d}_{i\pm1/2} = 2d_{i\pm1/2}$. If $de \leq 0$, then

$$d_{i-1/2} = d, \qquad d_{i+1/2} = d + 3eh, \tag{3.46$'$}$$

$$p'_{i-1/2}(x_i) = s + eh^2, \qquad p'_{i+1/2}(x_i) = s - 2eh^2, \tag{3.47$'$}$$

$$\bar{d}_{i-1/2} = 2d, \qquad \bar{d}_{i+1/2} = 2d + 6eh. \tag{3.48$'$}$$

$$\bar{p}'_{i-1/2}(x_i) = s + dh + eh^2, \qquad \bar{p}'_{i+1/2}(x_i) = s - dh - 5eh^2. \tag{3.49$'$}$$

From (3.44), (3.47), and (3.47′), the M3 and the MS3 constraints are identical up to $O(h^3)$:

$$t_i = u_i = \text{minmod}\,(s + eh^2, s - 2eh^2). \tag{3.50}$$

They may yield only second-order accurate $\dot{f}_i$ adjacent to a strict local extremum because of the following reasoning: if $e \neq 0$, and $s \in I[-eh^2, 2eh^2]$ such that s is strictly $O(h^2)$, e.g., $s = eh^2$, then $t_i = u_i = 0$ by (3.50). By (3.45), $|d|h > 3|e|h^2$. Equations (3.42b) then imply $s_i = 0$. Consequently, the final $\dot{f}_i$ is equal to 0 by either of these two methods, and the error is $O(h^2)$. Note that if $|s| > 6|e|h^2$, then the M3 and MS3 constraints preserve high-order accuracy since they do not alter the original

$\dot{f}_i$ due to (3.50). A loss of fourth-order accuracy, therefore, may occur only in a small region around a strict local extremum whose length is $O(h^2)$.

On the contrary, the MG3 constraint preserves accuracy: if h is small enough, then up to $O(h^2)$, the interval $[\alpha_i, \beta_i]$ of the MG3 constraint defined in (3.35a,b,c) contains $I[s - dh, s + dh]$ due to (3.42b), (3.49), and (3.49$'$). Since $d \neq 0$, this constraint does not alter the original $\dot{f}_i = s + O(h^3)$. The MG3 constraint (and therefore, all third-order constraints), however, can cause a loss of accuracy near $\hat{x}$ where $f'(\hat{x}) = 0$, and $f''(\hat{x}) = 0$, as shown in the following example, see also [7].

Let the data be on the cubic $f(x) = x^3$, and let the mesh be uniform with $x_{i-1} = -\epsilon$, $x_i = 2\epsilon$ where $\epsilon > 0$. It follows that

$$f_{i-1} = -\epsilon^3, \quad f_i = 8\epsilon^3, \quad f_{i+1} = 125\epsilon^3.$$

Since the data are increasing, the interval $[\alpha_i, \beta_i]$ reduces to $[0, 3s_i] = [0, 9\epsilon^2]$. The exact solution, $f'(x_i) = 12\epsilon^2$, lies outside this interval, and the final $\dot{f}_i$ has error $O(\epsilon^2) = O(h^2)$.

We conclude this section by comparing the accuracy of $\dot{f}_i$ defined by different third-order methods. Observe that the (lowest-order) error of the UNO scheme (the h^2 term of $|t_i - f'(x_i)|$)lies between $|eh^2|$ and $|2eh^2|$ by (3.50). The error of the M3-B method also shares this property due to its definition, (3.47), and (3.47$'$). The parabolic formula defined by $\dot{f}_i = p_i'(x_i)$ has error $|eh^2|$; consequently, it is more accurate than the UNO and the M3-B methods. Away from strict local extrema, the error of the M3-A method is $|eh^2/2|$, which is roughly less than half that of the UNO, the M3-B, or the parabolic methods. Adjacent to strict local extrema, the error is at most $|2eh^2|$. Notice that one cannot eliminate the second-order error term by a limiter function due to (3.47) and (3.47$'$). However, this term can be eliminated by using (3.44):

$$\dot{f}_i = \frac{1}{6}\left[4p_i'(x_i) + p_{i-1}'(x_i) + p_{i+1}'(x_i)\right].$$

In fact, if the mesh is uniform, the above formula is identical to the quartic formula; consequently, the error is $O(h^4)$. In this case, the M3-quartic method consists of the above formula followed by (3.21) and then (3.19a,b). It yields a fourth-order accurate $\dot{f}_i$ except at strict local extrema and inflection points, where it may degenerate to second-order. At inflection points, the loss of accuracy is caused by (3.21), and at strict local extrema, by (3.19a,b).

In the next section, we extend the monotonicity intervals further and derive fourth-order accurate algorithms which reproduce a cubic when the data lie on one.

4. Fourth-order accurate monotone interpolants. There are three main problems in developing fourth-order monotone interpolants which reproduce a cubic when the data are on one. First, example (3.40) shows that there can be monotone data which lie on a nonmonotone cubic. If the interpolant reproduces the cubic, it cannot preserve monotonicity as defined in section 2. Therefore, one has to modify the definition of monotonicity. Next, second-order accurate $\dot{f}_i$ can be obtained by using formula (2.3), which has a three point stencil consisting of $i-1$, i, and $i+1$. For third-order accurate $\dot{f}_i$, one has to add one more point to the above stencil, and to preserve symmetry, the stencil is enlarged to five points from $i-2$ to $i+2$. If $\dot{f}_i$ is calculated using the quartic through these five data points, it can be of the wrong sign as shown in example (2.26). The second problem is thus to define the original $\dot{f}_i$ to third-order accuracy. The final problem is to enlarge the monotonicity interval.

4.1. Nonoscillatory cubic interpolation. To solve the above problems, we introduce the cubics $\tilde{q}_{i+1/2}(x)$ defined for $x_i \leq x \leq x_{i+1}$ whose graphs $\{\tilde{Q}_{i+1/2}(x) = (x, \tilde{q}_{i+1/2}(x))\}$ stay "close" to the straight line segments F_iF_{i+1} so that they do not create unwanted oscillations. These cubics can be considered as fourth-order extensions of the nonoscillatory parabolic interpolant $p_{i+1/2}$ defined in subsection 3.1. Conceptually, our cubics are somewhat similar to those of the ENO schemes [14] which were developed for conservation laws. The key difference is that the ENO cubic interpolation is unstable due to a switch which decides whether the stencil is enlarged to the left or the right. Our methods are stable because they employ the median function as shown below.

Let $q_{i+1/2}(x)$ be the cubic defined by the four points F_{i-1}, F_i, F_{i+1}, F_{i+2}, and let $e_{i+1/2}$ be the third-divided difference of $\{f_i\}$,

$$e_{i+1/2} = f_{[x_{i-1},x_i,x_{i+1},x_{i+2}]} = \frac{d_{i+1} - d_i}{x_{i+2} - x_{i-1}}. \tag{4.1}$$

From the Newton formula,

$$q_{i-1/2}(x) = p_i(x) + e_{i-1/2}(x - x_{i-1})(x - x_i)(x - x_{i+1}), \tag{4.2a}$$

$$q_{i+1/2}(x) = p_i(x) + e_{i+1/2}(x - x_{i-1})(x - x_i)(x - x_{i+1}), \tag{4.2b}$$

where p_i is defined in (3.2). Rewrite (4.2a) with index i replaced by $i + 1$,

$$q_{i+1/2}(x) = p_{i+1}(x) + e_{i+1/2}(x - x_i)(x - x_{i+1})(x - x_{i+2}). \tag{4.2c}$$

If $\{f_i\}$ are fourth or higher-order samples of a C^4 function f, then for all $x \in [x_{i-1}, x_{i+2}]$,

$$q_{i+1/2}(x) = f(x) + O(h^4), \qquad q'_{i+1/2}(x) = f'(x) + O(h^3). \tag{4.3}$$

In the interval $[x_i, x_{i+1}]$, the cubic $\tilde{Q}_{i+1/2}$ going through F_i, F_{i+1} is defined by the derivatives $\tilde{q}'_{i+1/2}(x_i)$ and $\tilde{q}'_{i+1/2}(x_{i+1})$. We present two different methods to define $\tilde{Q}_{i+1/2}$.

For the first method, in the interval $[x_{i-1}, x_{i+1}]$, consider the three curves $Q_{i-1/2}$, P_i and $Q_{i+1/2}$, which go through the points F_{i-1}, F_i and F_{i+1}. Any two of these curves are either identical or have no other point in common. Let Q_i be the curve which lies between the other two, and let

$$e_i = \operatorname{minmod}(e_{i-1/2}, e_{i+1/2}). \tag{4.4}$$

From (4.2a,b), and (4.4), the equation for Q_i can be expressed as

$$q_i(x) = p_i(x) + e_i(x - x_{i-1})(x - x_i)(x - x_{i+1}). \tag{4.5}$$

Upon differentiating the above equation,

$$q'_i(x_{i-1}) = p'_i(x_{i-1}) + e_i(x_{i-1} - x_i)(x_{i-1} - x_{i+1}), \tag{4.6a}$$

$$q'_i(x_i) = p'_i(x_i) + e_i(x_i - x_{i-1})(x_i - x_{i+1}), \tag{4.6b}$$

$$q'_i(x_{i+1}) = p'_i(x_{i+1}) + e_i(x_{i+1} - x_{i-1})(x_{i+1} - x_i). \tag{4.6c}$$

Observe that in the interval $[x_i, x_{i+1}]$, among the two cubics Q_i, Q_{i+1} and the straight line segment F_iF_{i+1}, there may not be a median curve as in the case of the parabolas P_i, P_{i+1} and F_iF_{i+1} shown in the previous section. However, one can take the median of the derivatives to define $\tilde{q}_{i+1/2}$,

$$\tilde{q}'_{i+1/2}(x_i) = \operatorname{median}[s_{i+1/2}, q'_i(x_i), q'_{i+1}(x_i)]. \tag{4.7}$$

Using (3.5b) and (4.6), equation (4.7) and (4.7) with x_i replaced by x_{i+1} imply

$$\tilde{q}'_{i+1/2}(x_i) = s_{i+1/2} - \Delta x_{i+1/2} \operatorname{minmod}[d_i + e_i(x_i - x_{i-1}), d_{i+1} + e_{i+1}(x_i - x_{i+2})], \quad (4.8a)$$

$$\tilde{q}'_{i+1/2}(x_{i+1}) = s_{i+1/2} + \Delta x_{i+1/2} \operatorname{minmod}[d_i + e_i(x_{i+1} - x_{i-1}), d_{i+1} + e_{i+1}(x_{i+1} - x_{i+2})]. \quad (4.8b)$$

Remark that if d_i and d_{i+1} are of the same sign, then the cubic $\tilde{Q}_{i+1/2}$ is the median of the three curves Q_i, Q_{i+1} and F_iF_{i+1}. To show this, observe first that in the interval $[x_i, x_{i+1}]$, the cubic $Q_{i+1/2}$ lies between the two parabolas P_i and P_{i+1} for arbitrary d_i, d_{i+1}. Indeed, by differentiating (4.2b,c),

$$\begin{aligned} q'_{i+1/2}(x_i) &= p'_i(x_i) + e_{i+1/2}(x_i - x_{i-1})(x_i - x_{i+1}), \\ &= p'_{i+1}(x_i) + e_{i+1/2}(x_i - x_{i+1})(x_i - x_{i+2}). \end{aligned} \quad (4.9)$$

Since $(x_i - x_{i-1})(x_i - x_{i+2}) < 0$, equations (4.9) imply

$$[q'_{i+1/2}(x_i) - p'_i(x_i)][q'_{i+1/2}(x_i) - p'_{i+1}(x_i)] \leq 0,$$

or

$$q'_{i+1/2}(x_i) \in I\,[p'_i(x_i), p'_{i+1}(x_i)]. \quad (4.10)$$

A similar statement with x_i replaced by x_{i+1} holds. Because any two of the three curves $Q_{i+1/2}$, P_i, and P_{i+1} are either identical or they do not intersect in (x_i, x_{i+1}), $Q_{i+1/2}$ lies between P_i and P_{i+1}, see Fig. 7. Next, since Q_i is the median of $Q_{i-1/2}$, $Q_{i+1/2}$ and P_i, it lies between $Q_{i+1/2}$ and P_i. Similarly, Q_{i+1} lies between $Q_{i+1/2}$ and P_{i+1}. Thus if $0 \leq d_i \leq d_{i+1}$, or $0 \geq d_i \geq d_{i+1}$ as in Fig. 7, the parabola P_i is closer to F_iF_{i+1} than P_{i+1} and $\tilde{Q}_{i+1/2}$ is identical to Q_i. A similar conclusion holds if d_{i+1} lies between 0 and d_i. If d_i and d_{i+1} are of opposite sign, however, the cubic $\tilde{Q}_{i+1/2}$ can be different from any of the six curves $Q_{i-1/2}$, $Q_{i+1/2}$, $Q_{i+3/2}$, P_i, P_{i+1}, and F_iF_{i+1} which are used to define it.

The above proof also shows that definition (4.7) is equivalent to

$$\tilde{q}'_{i+1/2}(x_i) = \operatorname{median}[p'_{i+1/2}(x_i), q'_i(x_i), q'_{i+1}(x_i)]$$

where $s_{i+1/2}$ in (4.7) is replaced by $p'_{i+1/2}(x_i)$ defined in (3.5b). Our next definition resembles (3.13) and will be used to extend the monotonicity interval,

$$\tilde{t}_i = \operatorname{minmod}[\tilde{q}'_{i-1/2}(x_i), \tilde{q}'_{i+1/2}(x_i)]. \quad (4.11)$$

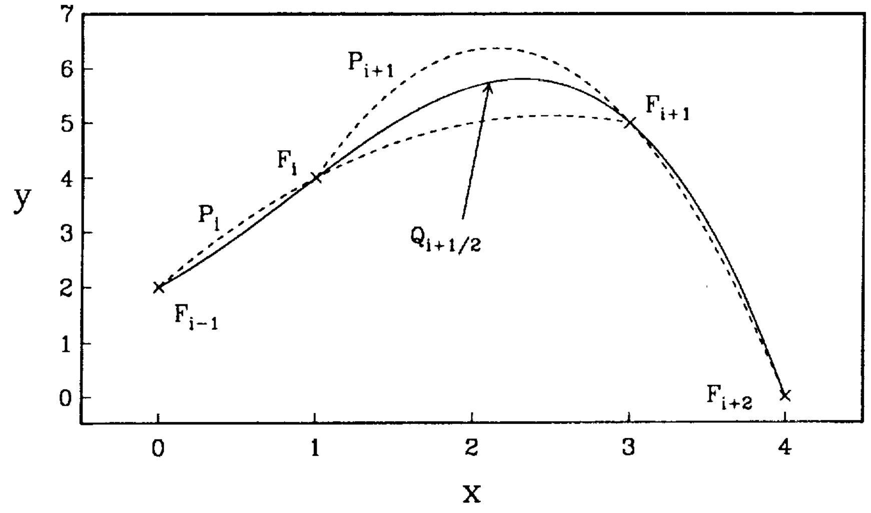


Fig. 7. $Q_{i+1/2}$ *lies between* P_i *and* P_{i+1}.

For the second method, $\tilde{Q}_{i+1/2}$ is again defined by the two slopes $\hat{q}'_{i+1/2}(x_i)$, $\hat{q}'_{i+1/2}(x_{i+1})$ where

$$\begin{aligned} u_{\min} &= \min\left[q'_{i-1/2}(x_i), q'_{i+1/2}(x_i), q'_{i+3/2}(x_i)\right], \\ u_{\max} &= \max\left[q'_{i-1/2}(x_i), q'_{i+1/2}(x_i), q'_{i+3/2}(x_i)\right], \\ \hat{q}'_{i+1/2}(x_i) &= \text{median}\left(s_{i+1/2}, u_{\min}, u_{\max}\right), \\ &= s_{i+1/2} + \text{minmod}\left(u_{\min} - s_{i+1/2}, u_{\max} - s_{i+1/2}\right). \end{aligned} \tag{4.12}$$

Similarly, $\hat{q}'_{i+1/2}(x_{i+1})$ is defined by replacing x_i by x_{i+1} in the above equations. Observe that

$$I\left[q'_i(x_i), q'_{i+1}(x_i)\right] \subset I\left[q'_{i-1/2}(x_i), q'_{i+1/2}(x_i), q'_{i+3/2}(x_i)\right],$$

consequently, from definitions (4.7) and (4.12),

$$I\left[s_{i+1/2}, \tilde{q}'_{i+1/2}(x_i)\right] \supset I\left[s_{i+1/2}, \hat{q}'_{i+1/2}(x_i)\right], \tag{4.13}$$

i.e., $\tilde{q}'_{i+1/2}(x_i)$ and $\hat{q}'_{i+1/2}(x_i)$ are on the same side relative to $s_{i+1/2}$ and $\hat{q}'_{i+1/2}(x_i)$ is closer to $s_{i+1/2}$ than $\tilde{q}'_{i+1/2}(x_i)$. Finally, as in the first method,

$$\hat{f}_i = \text{minmod}\left[\hat{q}'_{i-1/2}(x_i), \hat{q}'_{i+1/2}(x_i)\right]. \tag{4.14}$$

Notice that if the data are fourth-order accurate, then so are $\{\tilde{Q}_{i+1/2}\}$ defined by either of the above two methods. Because these cubics stay "close" to the straight line segments F_iF_{i+1}, they do not create unwanted oscillations. However, the derivatives of these cubics are discontinuous at $\{x_i\}$; therefore, they are only C^0. We use these cubics to define accurate, monotone C^1 interpolants below.

4.2. M4 (monotone fourth-order) methods. First, we modify the definition of monotonicity. The data are *q-monotone in* $[x_i, x_{i+1}]$ if they are monotone in $[x_i, x_{i+1}]$, and

$$\tilde{q}'_{i+1/2}(x_i) \in I\,[0, 3s_{i+1/2}], \qquad \tilde{q}'_{i+1/2}(x_{i+1}) \in I\,[0, 3s_{i+1/2}] \tag{4.15}$$

where $\tilde{q}'_{i+1/2}(x_i)$, $\tilde{q}'_{i+1/2}(x_{i+1})$ are defined by (4.8). Using Lemma 1, (4.15) implies that $\tilde{Q}_{i+1/2}$ is monotone in $[x_i, x_{i+1}]$. Observe that if the data are q-monotone, then (4.15) with $\tilde{q}'_{i+1/2}(x_i)$, $\tilde{q}'_{i+1/2}(x_{i+1})$ replaced by $\hat{q}'_{i+1/2}(x_i)$, $\hat{q}'_{i+1/2}(x_{i+1})$ is also satisfied due to (4.13).

Next, the original $\dot{f}_i$ is defined by applying any continuous limiter function with property (2.12) to the left and right slopes $\tilde{q}'_{i-1/2}(x_i)$ and $\tilde{q}'_{i+1/2}(x_i)$, e.g., the arithmetic mean as in (3.18),

$$\dot{f}_i = \frac{1}{2}\left[\tilde{q}'_{i-1/2}(x_i) + \tilde{q}'_{i+1/2}(x_i)\right]. \tag{4.16}$$

The corresponding method is named M4-A. If the data represent a discontinuous function, the Van Albada's limiter (2.23), which results in the M4-V method, is preferred. Similar to the M3-V method, the monotonicity algorithm (4.20) below is omitted for the M4-V method.

Finally, we relax the monotonicity constraint (3.14) to

$$\dot{f}_i \in I\left[0, 3s_{i+1/2}, \frac{3}{2}p'_{i+1/2}(x_i), \tilde{q}'_{i+1/2}(x_i)\right], \tag{4.17a}$$

$$\dot{f}_{i+1} \in I\left[0, 3s_{i+1/2}, \frac{3}{2}p'_{i+1/2}(x_{i+1}), \tilde{q}'_{i+1/2}(x_{i+1})\right]. \tag{4.17b}$$

More generally, we can extend any constraint of section 3. For simplicity, the following constraint extends (3.16),

$$\dot{f}_i \in I\left[0, 3s_i, \frac{3}{2}t_i, \tilde{t}_i\right] \tag{4.18}$$

where t_i, $\tilde{t}_i$ are defined by (3.13) and (4.11).

The fourth-order methods are summarized below.

Theorem 3. *Let $s_{i+1/2}$, d_i, $e_{i+1/2}$ be respectively the first, second, and third divided differences; let s_i, $d_{i+1/2}$, e_i be the corresponding minmod as in (2.8), (3.4), and (4.4). Let $\{p'_{i-1/2}(x_i), p'_{i+1/2}(x_i)\}$ and t_i be given by (3.5b) and (3.13); let $\tilde{q}'_{i+1/2}(x_i)$, $\tilde{q}'_{i+1/2}(x_{i+1})$, and $\tilde{t}_i$ be defined by either (4.8a,b) and (4.11), or (4.12) and (4.14). In addition, let $\dot{f}_i$ be defined by a continuous limiter function with property (2.12), e.g., the average (4.16), or by any third or higher-order formula for the derivative. Let the final $\dot{f}_i$ be defined by*

$$t_{\min} = \min\left(0, 3s_i, \frac{3}{2}t_i, \tilde{t}_i\right), \tag{4.19a}$$

$$t_{\max} = \max\left(0, 3s_i, \frac{3}{2}t_i, \tilde{t}_i\right), \tag{4.19b}$$

$$\dot{f}_i \leftarrow \dot{f}_i + \operatorname{minmod}(t_{\min} - \dot{f}_i, t_{\max} - \dot{f}_i). \tag{4.20}$$

Then the algorithm is stable. Moreover, if the data are q-monotone in $[x_i, x_{i+1}]$ for any i, $m+3 \le i \le n-4$, then the interpolant (2.2) is monotone in $[x_i, x_{i+1}]$. As for accuracy, if $f \in C^4$, and $\{f_i\}$ are fourth or higher-order accurate, the final $\{\dot{f}_i\}$ are third-order; consequently, the interpolant (2.2) is fourth-order.

The proof of stability is again straightforward. To prove accuracy, observe that the second half of (4.3) implies $q'_{i-3/2}$, $q'_{i-1/2}$, $q'_{i+1/2}$, and $q'_{i+3/2}$ at $x = x_i$ approximate the true derivative $f'(x_i)$ to $O(h^3)$. Therefore, the same is true for q'_{i-1}, q'_i, q'_{i+1} because their definitions use the median function. It follows that $\tilde{q}'_{i-1/2}$, $\tilde{q}'_{i+1/2}$, and then $\tilde{t}_i$, are third-order accurate. Since the original $\dot{f}_i$ is defined a limiter function with property (2.12), it lies between $\tilde{q}'_{i-1/2}(x_i)$ and $\tilde{q}'_{i+1/2}(x_i)$, and thus is also third-order. Because the final $\dot{f}_i$ by (4.20) lies between the original $\dot{f}_i$ and $\tilde{t}_i$, the accuracy claim follows. Next, suppose the data are q-monotone in $[x_i, x_{i+1}]$, then $\tilde{q}'_{i+1/2}(x_i) \in I[0, 3s_{i+1/2}]$. Definition (4.11) implies $\tilde{t}_i \in I[0, \tilde{q}'_{i+1/2}(x_i)]$. Consequently, $\tilde{t}_i \in I[0, 3s_{i+1/2}]$. Since q-monotonicity implies that the data are monotone in $[x_i, x_{i+1}]$, it follows that $\frac{3}{2}t_i \in I[0, 3s_{i+1/2}]$. The interval $I[0, 3s_i, \frac{3}{2}t_i, \tilde{t}_i]$ is thus contained in $I[0, 3s_{i+1/2}]$. Expression (4.18) then implies $\dot{f}_i \in I[0, 3s_{i+1/2}]$. Similar arguments hold for $\dot{f}_{i+1}$, and the monotonicity of the interpolant (2.2) follows from Lemma 1. This completes the proof.

Note that instead of the average (4.16), one can use the following fourth-order quartic formula to define the original $\dot{f}_i$:

$$\dot{f}_i = p_i'(x_i) + (x_i - x_{i-1})(x_i - x_{i+1})\frac{e_{i-1/2}(x_{i+2} - x_i) + e_{i+1/2}(x_i - x_{i-2})}{x_{i+2} - x_{i-2}}. \tag{4.21}$$

As in (3.21), to avoid derivatives of the wrong sign, before applying (4.19a,b) and (4.20), one replaces the original $\dot{f}_i$ by

$$\dot{f}_i \leftarrow \text{median}\,(\dot{f}_i, \tilde{q}_{i-1/2}'(x_i), \tilde{q}_{i+1/2}'(x_i)). \tag{4.22}$$

The resulting method is called the M4-quartic method.

To obtain MS4 constraint, which extends the MS3 constraint, one simply replaces t_i, $\tilde{t}_i$ in (4.19a,b) by u_i, $\tilde{u}_i$ where u_i is defined in (3.23), and

$$\tilde{u}_i = \text{minmod}\left[q_{i-3/2}'(x_i), q_{i-1/2}'(x_i), q_{i+1/2}'(x_i), q_{i+3/2}'(x_i)\right].$$

This constraint, however, does not provide the original $\dot{f}_i$. If the quartic formula (4.21) is used for $\dot{f}_i$, then (4.22) is necessary to avoid derivatives of the wrong sign. Since $\tilde{q}_{i-1/2}'(x_i)$ and $\tilde{q}_{i+1/2}'(x_i)$ are needed, one might as well use the algorithm of Theorem 3.

5. Boundary conditions. The above methods apply only to the interior points. Near the boundaries, one must use one-sided algorithms. We carry out the arguments for the left boundary, and only occasionally, formulas for the right boundary are presented to show the symmetry between the two boundaries. For third-order methods, $\dot{f}_m$ is defined by the slope of the parabola P_{m+1} at x_m,

$$\dot{f}_m = p_{m+1}'(x_m) = s_{m+1/2} + d_{m+1}(x_m - x_{m+1}), \tag{5.1}$$

$$\dot{f}_n = p_{n-1}'(x_n) = s_{n-1/2} + d_{n-1}(x_n - x_{n-1}).$$

For the quartic or the fourth-order methods, the obvious choices for $\dot{f}_m$ and $\dot{f}_{m+1}$ are the slopes of the cubic $Q_{m+3/2}$. It follows from differentiating (4.2b) that

$$\dot{f}_m = s_{m+1/2} + d_{m+1}(x_m - x_{m+1}) + e_{m+3/2}(x_m - x_{m+1})(x_m - x_{m+2}),$$

$$\dot{f}_{m+1} = s_{m+1/2} + d_{m+1}(x_{m+1} - x_m) + e_{m+3/2}(x_{m+1} - x_m)(x_{m+1} - x_{m+2}).$$

As for the constraints, the second, third, and fourth-order ones presented above have respectively a three, five, and seven-point stencil from $i - k$ to $i + k$, $k = 1, 2$ or 3.

Near the boundaries, the high-order constraints are replaced by appropriate lower-order ones, e.g., at $m+2$, one of the third-order constraints, or the MP constraint, and at $m+1$, the MP constraint. Since the definition of monotonicity of the data in $[x_i, x_{i+1}]$ involves the values f_{i-1} and f_{i+2}, at the left boundary, we cannot have monotonicity of the data in $[x_m, x_{m+1}]$. In this interval, the interpolant is monotone provided condition (2.6) is enforced:

$$\dot{f}_m \leftarrow \text{minmod}\,(\dot{f}_m, 3s_{m+1/2}), \qquad \dot{f}_n \leftarrow \text{minmod}\,(\dot{f}_n, 3s_{n-1/2}). \tag{5.2}$$

If the data are smooth and have a strict local extremum next to the boundary, then the above constraints may "clip" the extremum as shown in subsection 2.3. Consequently, whenever possible, the boundaries should be chosen at the smooth part of the data and away from strict local extrema.

For the M3 methods, boundary conditions which provide smoother transitions with the interior points than (5.1) can be obtained as follows. The task is to define the parabola P_m. The simplest way is to let P_m be identical to P_{m+1},

$$d_m = d_{m+1}. \tag{5.3}$$

P_m can also be defined by the two points F_m, F_{m+1}, and the slope $q'_{m+3/2}(x_m)$. From (4.2b),

$$q_{m+3/2}(x) = p_{m+1}(x) + e_{m+3/2}(x - x_m)(x - x_{m+1})(x - x_{m+2}).$$

By expanding $q'_{m+3/2}(x_m)$, and using (3.5b), the above definition for P_m is equivalent to

$$d_m = d_{m+1} + e_{m+3/2}(x_m - x_{m+2}). \tag{5.4}$$

$$d_n = d_{n-1} + e_{n-3/2}(x_n - x_{n-2}).$$

The third method to define P_m is obtained by setting p''_m equal to $q''_{m+3/2}(x_m)$, or equivalently,

$$d_m = d_{m+1} + e_{m+3/2}(2x_m - x_{m+2} - x_{m+1}). \tag{5.5}$$

Observe that if the mesh is uniform, (5.4) and (5.5) respectively reduce to

$$d_m = d_{m+1} + \frac{2}{3}(d_{m+1} - d_{m+2}),$$

$$d_m = d_{m+1} + (d_{m+1} - d_{m+2}).$$

Expressions (5.3), (5.4) and (5.5) for d_m correspond to extrapolation to the boundary from the interior, with (5.4) and (5.5) being more accurate than (5.3). It is well-known that higher-order extrapolation may cause problems. In this case, however, the essential quantity is $d_{m+1/2}$, which is the median of d_m, d_{m+1}, and 0. Therefore, d_m takes effect only when it lies between 0 and d_{m+1}, and consequently, the above methods behave at least as well as the commonly used one-sided interpolation (5.1). To define $\dot{f}_m$, we simply set

$$\dot{f}_m = p'_{m+1/2}(x_m). \tag{5.6}$$

Clearly, if one uses (5.3), the resulting $\dot{f}_m$ is the same as (5.1). If (5.4) is being used, then $\dot{f}_m$ by (5.6) is identical to

$$\dot{f}_m = \text{median}\left[s_{m+1/2}, p'_{m+1}(x_m), q'_{m+3/2}(x_m)\right]. \tag{5.7}$$

Note that to assure the monotonicity of the interpolant in $[x_m, x_{m+1}]$ and $[x_{m+1}, x_{m+2}]$, the MP constraint must be enforced at $m+1$, and (5.2) at m.

Finally, for the M4 methods, the first task is to define $Q_{m+1/2}$. Let

$$\gamma = f_{[x_m,\ldots,x_{m+4}]} = \frac{e_{m+5/2} - e_{m+3/2}}{x_{m+4} - x_m}, \tag{5.8}$$

and let $r_{m+2}(x)$ be the equation of the quartic interpolating $f_m, \ldots, f_{m+4}$. Then by applying the Newton formula, and setting $e_{m+1/2} = r^{(3)}_{m+2}(x_{m+1/2})/3!$,

$$e_{m+1/2} = e_{m+3/2} + \gamma(x_m + x_{m+1} - x_{m+2} - x_{m+3}). \tag{5.9}$$

Next, we define $\tilde{Q}_{m+1/2}$,

$$\tilde{q}'_{m+1/2}(x_m) = \text{median}\,[s_{m+1/2}, q'_{m+1/2}(x_m), q'_{m+3/2}(x_m)], \tag{5.10}$$

and similarly, $\tilde{q}'_{m+1/2}(x_{m+1})$ is defined by replacing x_m by x_{m+1} in the above equation. Consequently, at $i = m+1$, and $i = m+2$, there is enough information for the algorithm in Theorem 3. At the left boundary, set

$$\dot{f}_m = \tilde{q}'_{m+1/2}(x_m). \tag{5.11}$$

6. Numerical examples. The methods presented above can be separated into two types: methods which combine $\dot{f}_i$ defined by an interpolation formula with the MP, M3, MS3, MG3, or M4 constraint belong to type I; those which yield the slopes on the left and the right side of i before employing a limiter function belong to type II. For type I methods, besides the parabolic (2.3) and the quartic formula (4.21), we also test Hyman's fourth-order finite-difference formula [17],

$$\dot{f}_i = \frac{-f_{i+2} + 8f_{i+1} - 8f_{i-1} + f_{i-2}}{-x_{i+2} + 8x_{i+1} - 8x_{i-1} + x_{i-2}}. \tag{6.1a}$$

If the mesh is uniform, the above formula is identical to the quartic formula. For convenience, we refer to (6.1a) as the FD4 formula. At the left boundary, the following cubic formulas are used:

$$\dot{f}_m = \frac{-22f_m + 36f_{m+1} - 18f_{m+2} + 4f_{m+3}}{-22x_m + 36x_{m+1} - 18x_{m+2} + 4x_{m+3}}, \tag{6.1b}$$

$$\dot{f}_{m+1} = \frac{-2f_m - 3f_{m+1} + 6f_{m+2} - 2f_{m+3}}{-2x_m - 3x_{m+1} + 6x_{m+2} - 2x_{m+3}}. \tag{6.1c}$$

Formulas for the right boundary are obtained by replacing $m+$ by $n-$ in the above equations. Type I methods to be tested are the combinations of the parabolic, the quartic, or the FD4 formulas with one of the constraints MP, MS3, MG3, M3, or M4. Type II includes methods which employ limiter functions. The eight limiters to be tested are: the minmod (2.14), abbreviated by MD; the Van Albada (2.23), V; the average (2.19), A; the "Superbee" (2.17), B; and occasionally, the harmonic mean (2.15), HM; the Fritsch-Butland (2.16), FB; the limiter (2.20), AR (average rational); and the limiter (2.21), AQ (average cubic). Combining these limiters with the M3, M4, or no constraint yields 24 methods. For the M3 methods, we use the boundary condition (5.5), and for the M4, (5.10) and (5.11). Since there are 39 methods altogether, for each data set below, we can only present some selected results for these methods.

The data sets include both rough and smooth ones. The rough data sets, which are monotone, are used to test the "visually pleasing" or "too flat" properties of the methods. The smooth data sets are nonmonotone. They provide information on accuracy.

6.1. Monotone data. The first data set appearing below is an example from Akima, [1], [9], [10], and the second is the Fritsch-Carlson RPN 14 data [10], [17].

Akima data

x	f
3.	10.
5.	10.
6.	10.
8.	10.
9.	10.5
11.	15.
12.	50.
14.	60.
15.	85.

Fritsch-Carlson data

x	f
7.99	0.
8.09	2.76429E-5
8.19	4.37498E-2
8.7	0.169183
9.2	0.469428
10.	0.943740
12.	0.998636
15.	0.999919
20.	0.999994

Figs. 8, 9 and 10, 11 show the respective interpolation curves for the Akima and the Fritsch-Carlson data. Since these data are monotone, the M3, MS3, and MG3 constraints are identical to the MP constraint.

The methods presented in Figs. 8 and 10 are: (a) parabolic, (b) MG3-parabolic, (c) FD4, (d) M4-quartic, (e) MG3-FD4, (f) M4-A, (g) harmonic mean, and (h) Fritsch-Butland. Note that the interpolants on the right are generally more "visually pleasing" than those on the left. The interpolants in Figs. 8(a,c) of the parabolic and FD4 methods are oscillatory. Since the mesh is irregular, $\dot{f}_n$ at the right boundary of the FD4 method has the wrong sign in Fig. 8(c). Although the monotonicity constraint corrected $\dot{f}_n$ to 0 in Fig. 8(e), it is still unacceptable. Consequently, the FD4 method (6.1) should only be used when the mesh is smooth. Fig. 8(g) shows that the harmonic mean limiter, as observed by Fritsch and Butland [9], produces interpolants which are "too flat". Four of the methods which yield "visually pleasing" interpolants are presented in the right column: the constrained parabolic, constrained quartic, M4-A, and Fritsch-Butland. Except for those of the boundary, similar remarks hold for Figs. 10.

In Figs. 9 and 11, results for well-known (M2) limiters are presented on the left, and the corresponding M3 methods are on the right: (a) minmod, (b) UNO, (c) Van Albada, (d) M3-V, (e) average, (f) M3-A, (g) "Superbee", and (h) M3-B. The interpolants of the minmod, Van Albada, UNO, and M3-V methods, shown by Figs. 9

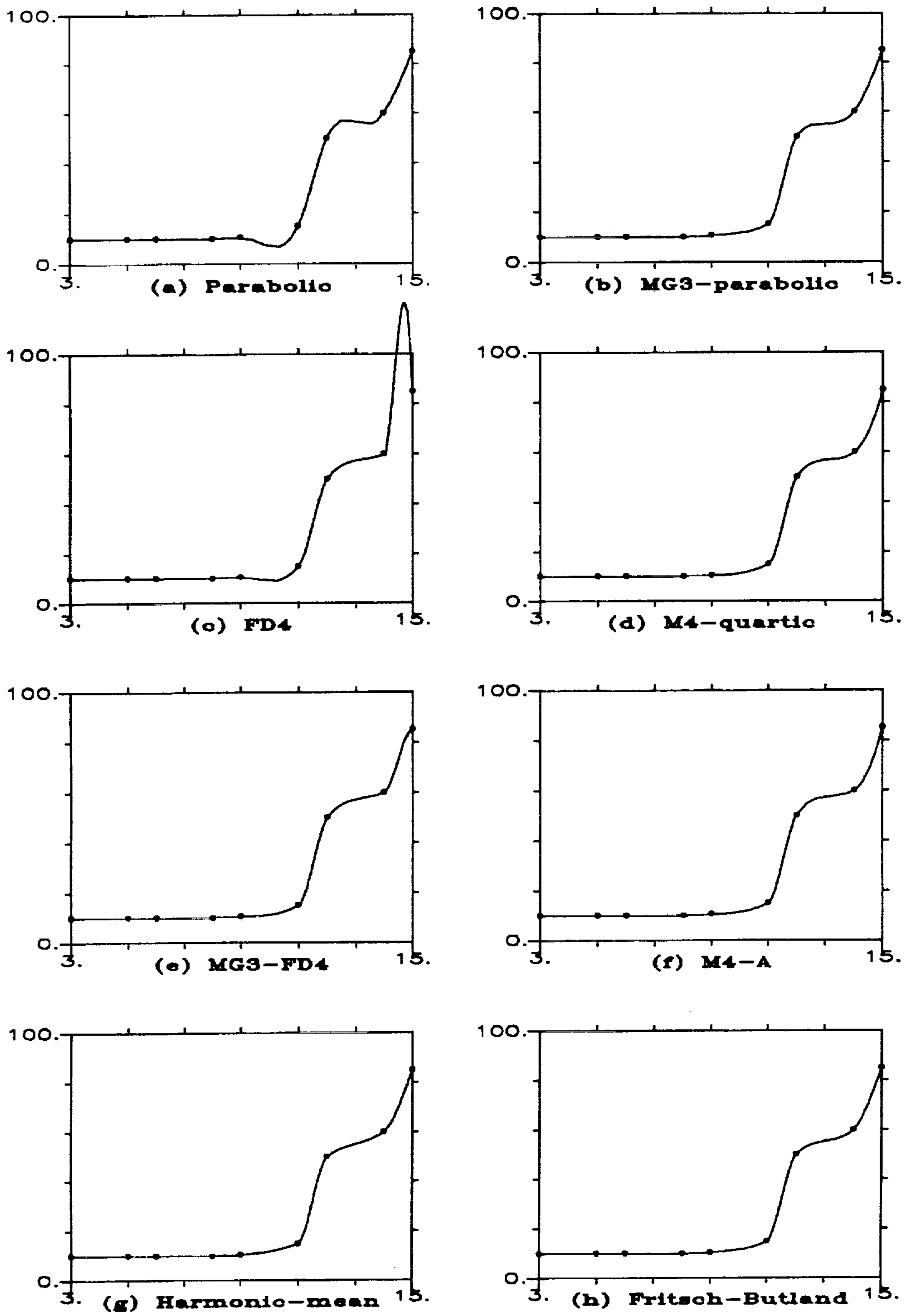
and 11 (a,b,c,d), are "too flat". Although the M3 methods are more accurate, the results are essentially the same as those of the M2 methods since the data are rough. The interpolants for the AR and AQ limiters are not shown because they are essentially the same as those of the average. Methods which yield "visually pleasing" interpolants are presented at the bottom half of each figure: the average limiter, "Superbee", M3-A, and M3-B.

For rough and monotone data sets, "visually pleasing" interpolants are produced by the following methods: the constrained (MP, MS3, or MG3) parabolic; the average, Fritsch-Butland, and "Superbee" limiters; the M3-A, M3-B, and M3-quartic; and the M4-A, M4-B, and M4-quartic.

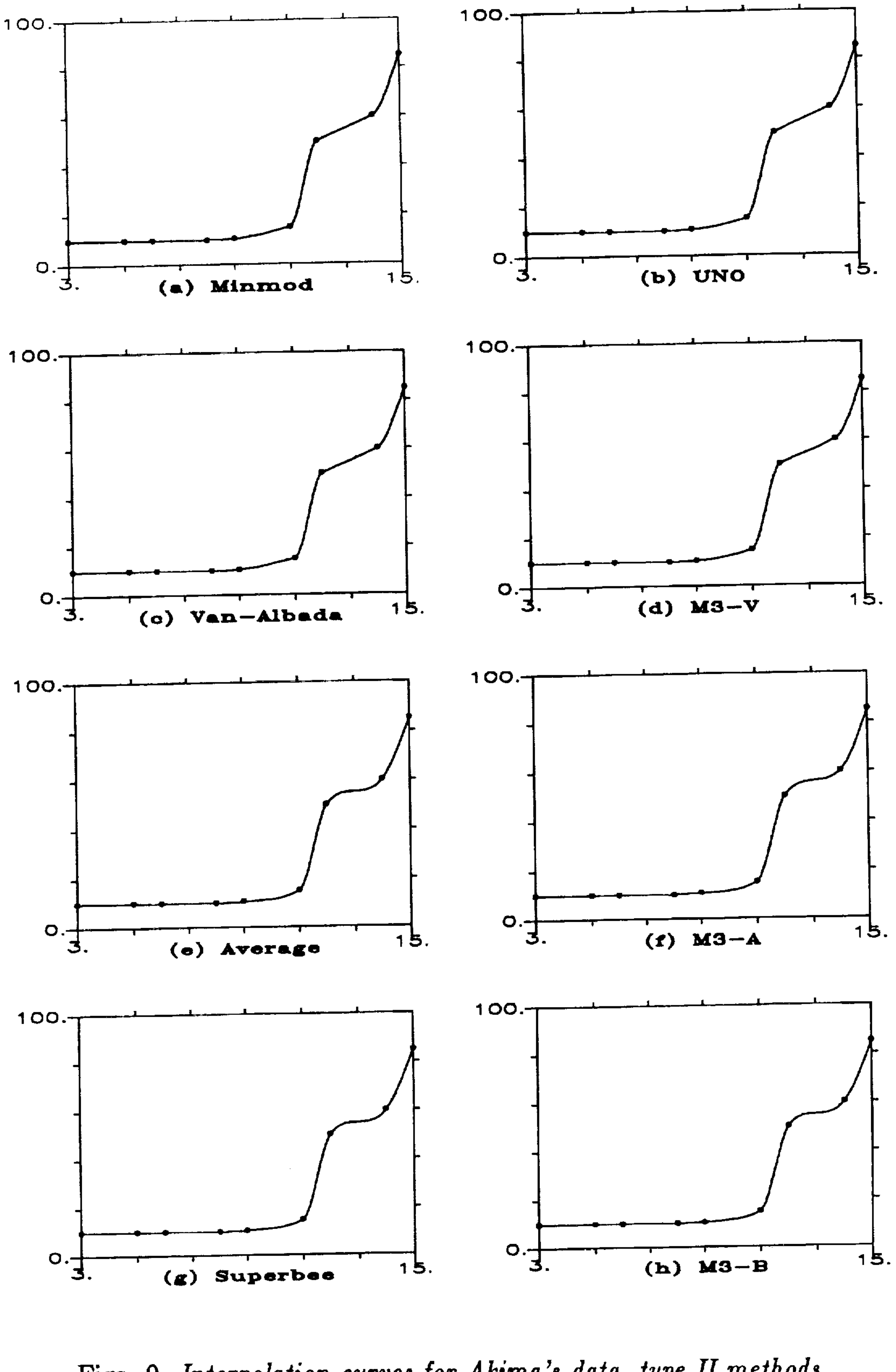
6.2. Nonmonotone data. Following Hyman [17], we interpolate the function $f(x) = e^{-x^2}$ on a uniform mesh. Note that the data are smooth and have a local maximum. The results for $N = n - m = 8$ mesh intervals are presented in Figs. 12–14. Since the mesh is uniform, the FD4 formula gives identical results as the quartic formula, and the MP-parabolic method is the same as the average limiter method.

Fig. 12 shows the results for the domain $[-2.8, 3.6]$. The methods presented are of type I: (a) parabolic, (b) MG3-parabolic, (c) MP-parabolic, (d) MS3-parabolic, (e) quartic, (f) MG3-quartic, (g) MP-quartic, and (h) MS3-quartic. Figs. 12(c,g) show that the MP constraint clips the strict local extrema, while the MS3 and MG3 constraints have no effect in this case. Observe that in Figs. 12(a,e), near the boundaries, the parabolic and the quartic methods yield nonmonotone interpolants, while all the constrained methods preserve monotonicity. Accurate monotone interpolants are produced by the methods presented in the right column: (b) MG3-parabolic, (d) MS3-parabolic, (f) MG3-quartic, and (h) MS3-quartic.

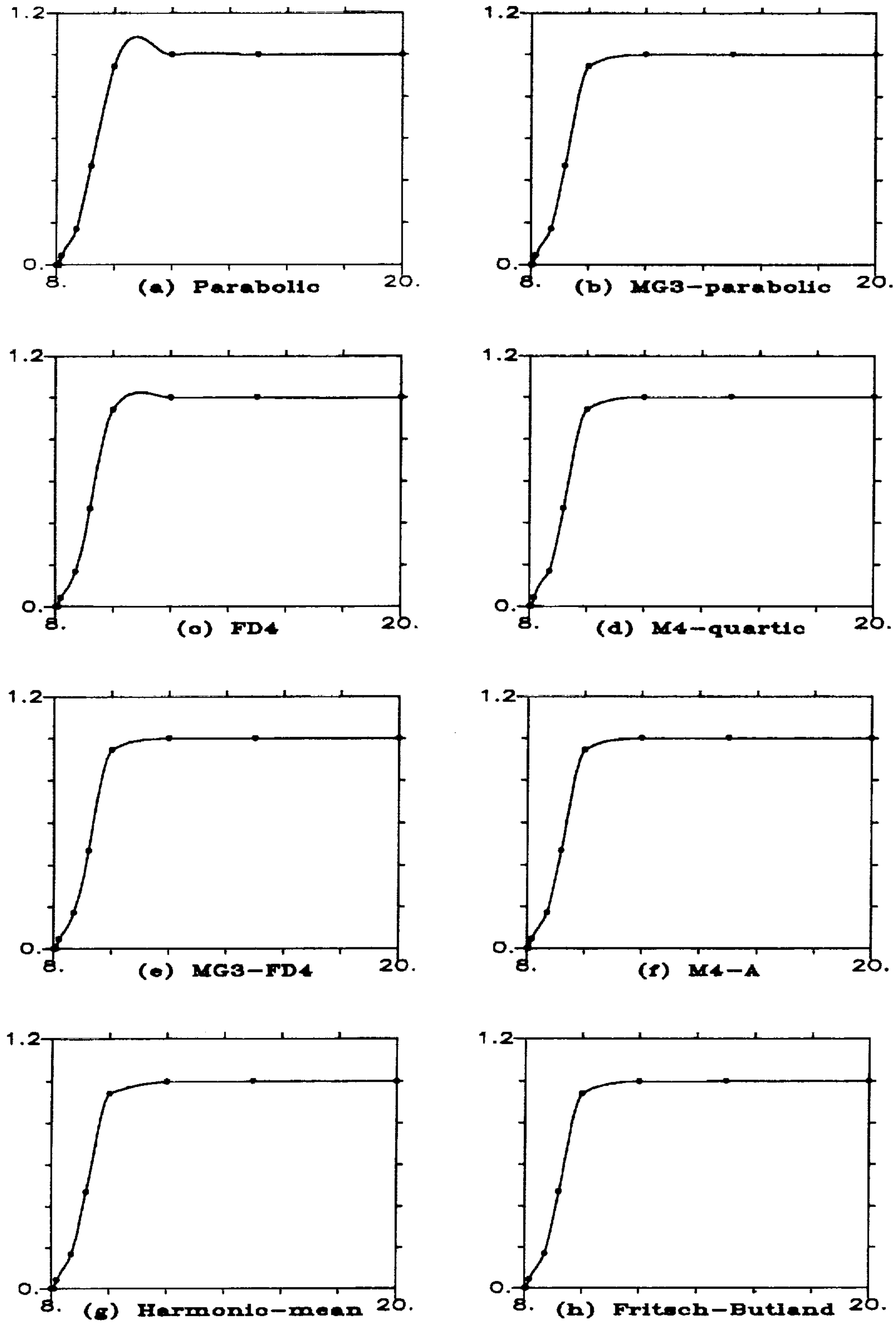
Figs. 13 are identical to Figs. 12 except the domain is shifted to the left: $[-2.9, 3.5]$. Figs. 13(b,f) show that near extrema, the MG3 constraint always provides "plenty of room" so that it has no effect. Note that for the three mesh intervals around the maximum, the interpolants in Figs. 13(b,f) are respectively identical to those of the unconstrained methods in Figs. 13(a,e). However, Figs. 13(d,h) show that the MS3 constraint occasionally does not provide "enough room", especially for a coarse mesh.



Figs. 8. *Interpolation curves for Akima's data.*



Figs. 9. *Interpolation curves for Akima's data, type II methods.*



(a) Parabolic

(b) MG3-parabolic

(c) FD4

(d) M4-quartic

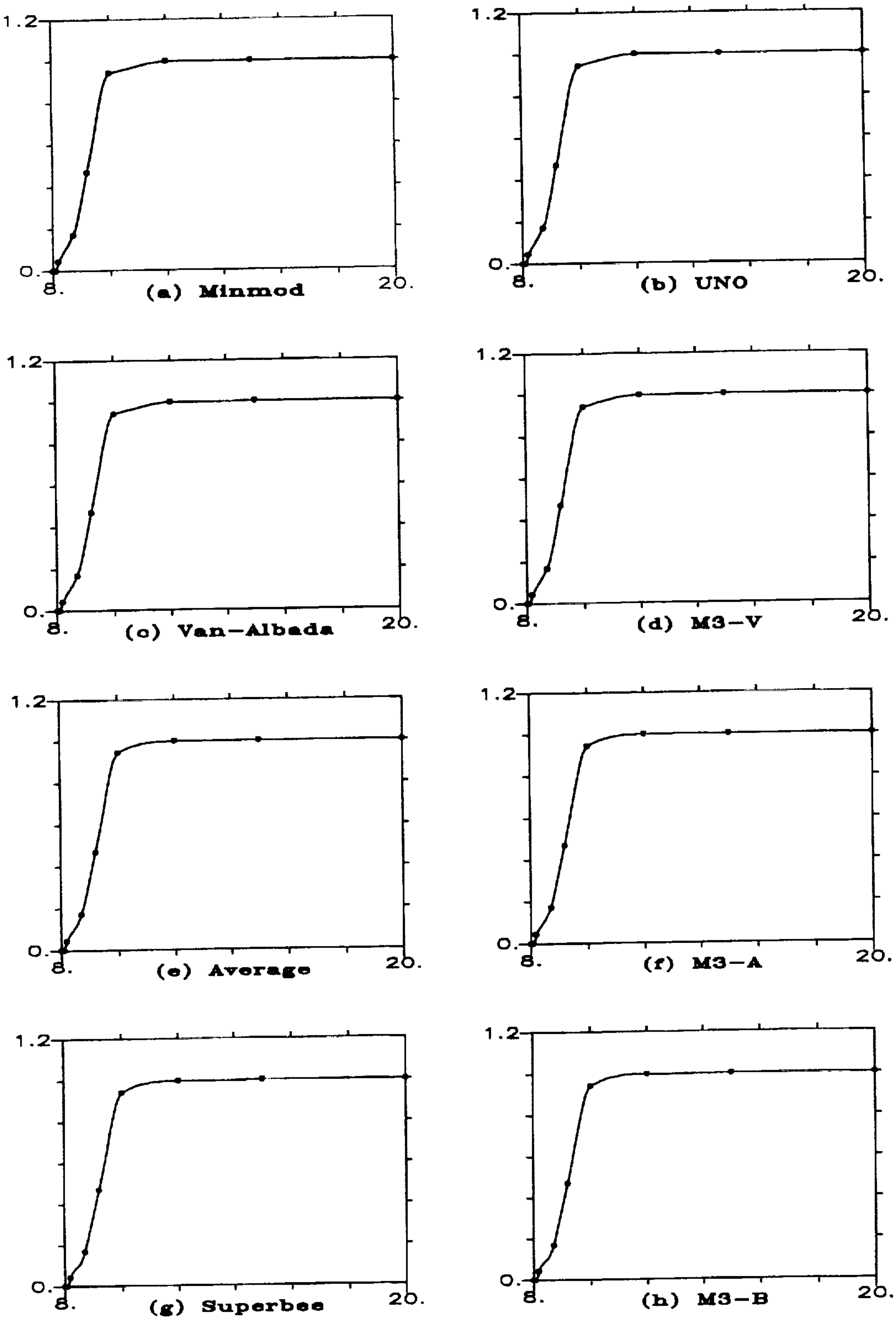
(e) MG3-FD4

(f) M4-A

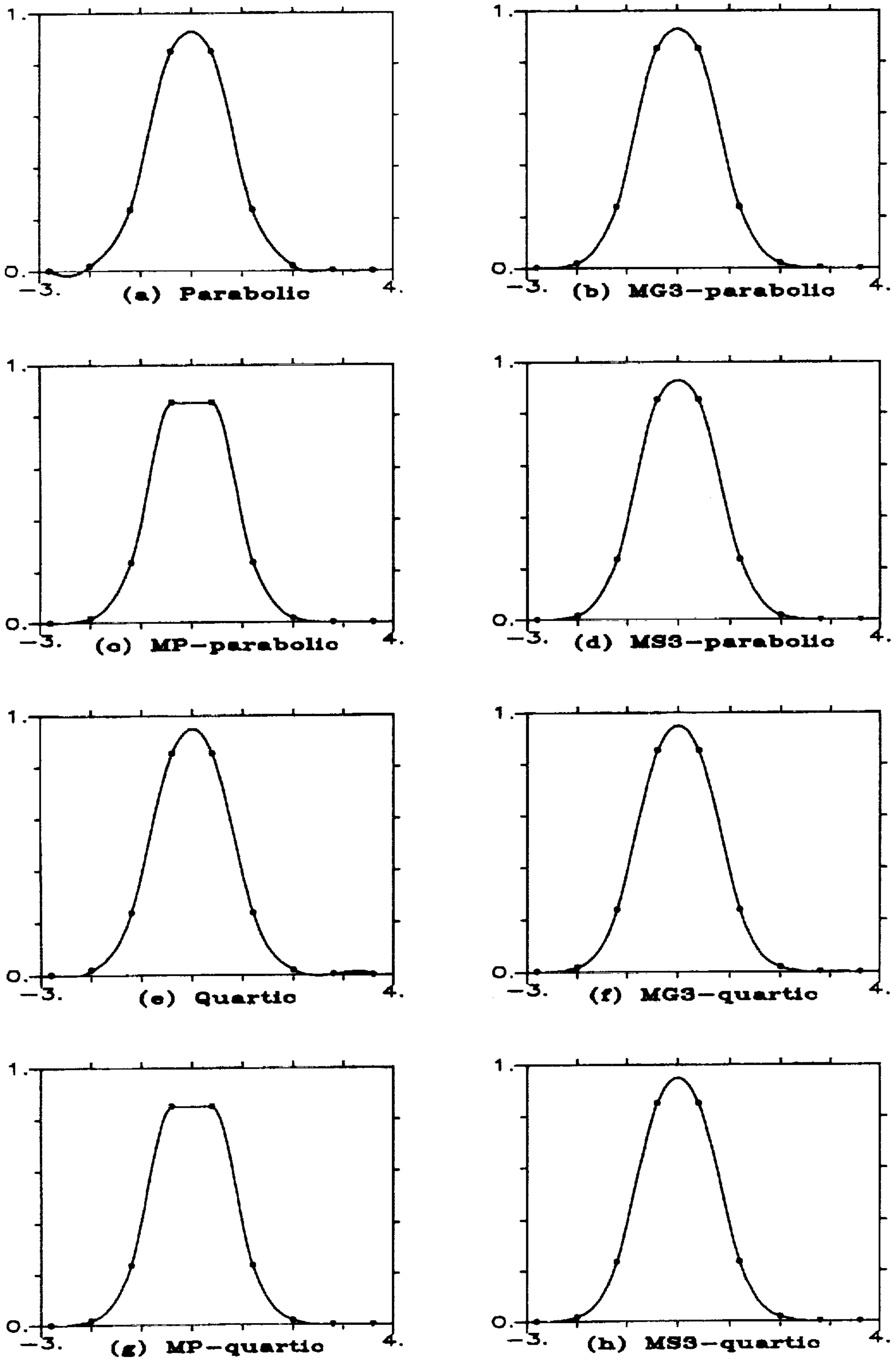
(g) Harmonic-mean

(h) Fritsch-Butland

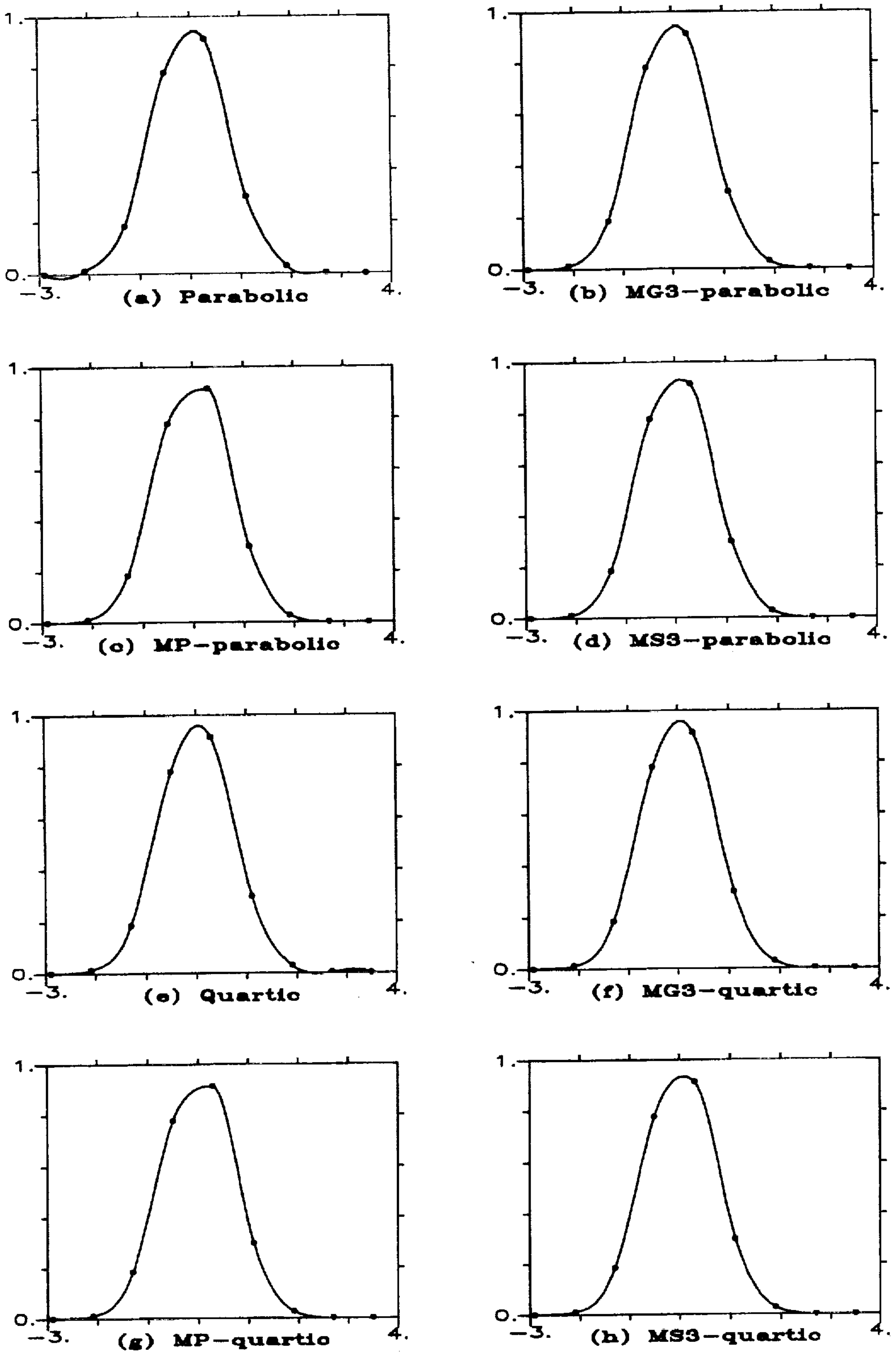
Figs. 10. *Interpolation curves for Fritsch-Carlson's data.*



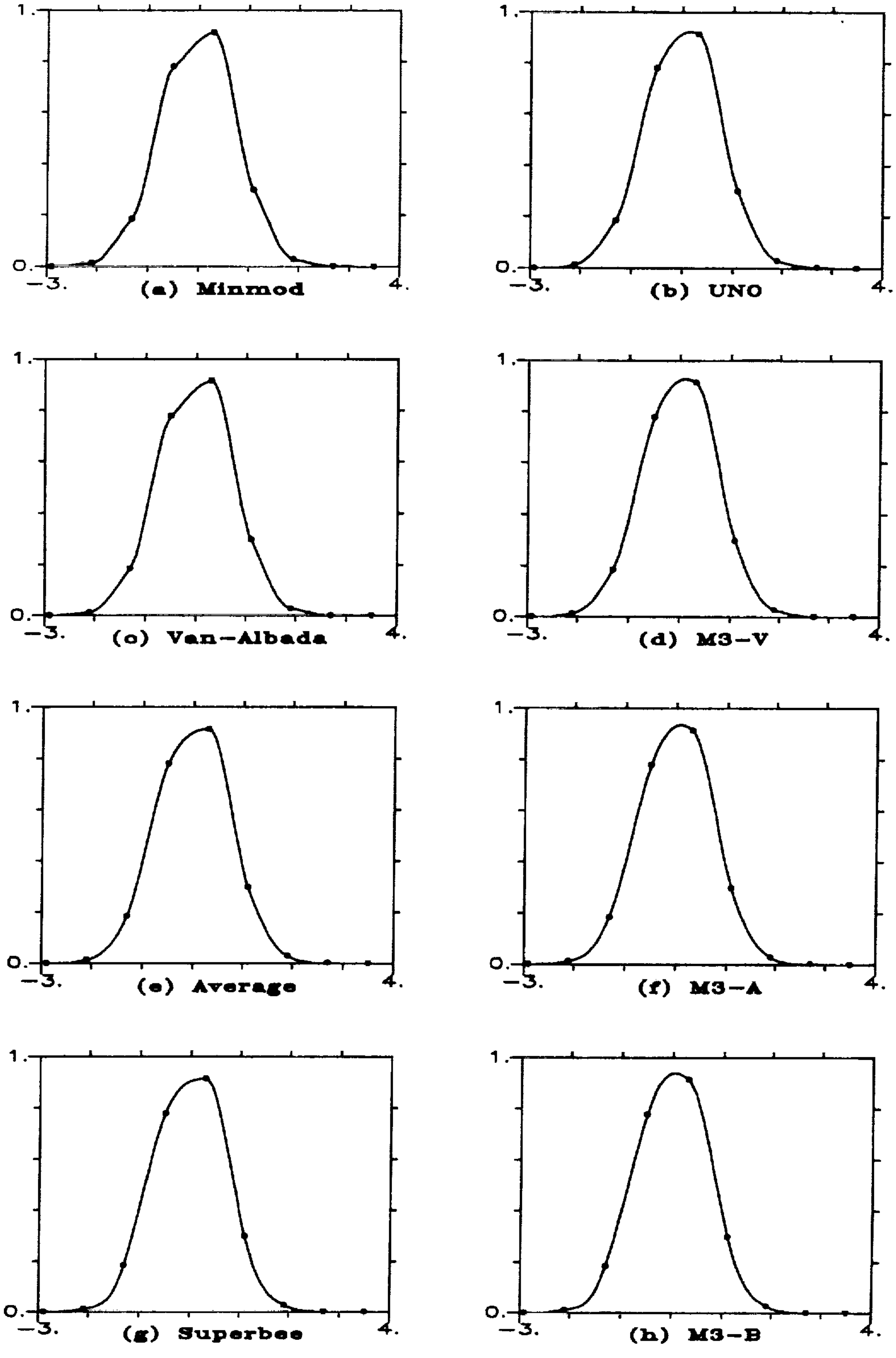
Figs. 11. *Interpolation curves for Fritsch-Carlson's data, type II methods.*



Figs. 12. *Interpolation curves for* $f(x) = e^{-x^2}$ *on the domain* $[-2.8, 3.6]$*, type I methods.*



Figs. 13. *Interpolation curves for* $f(x) = e^{-x^2}$ *on the domain* $[-2.9, 3.5]$, *type I methods.*



(a) Minmod

(b) UNO

(c) Van−Albada

(d) M3−V

(e) Average

(f) M3−A

(g) Superbee

(h) M3−B

Figs. 14. *Interpolation curves for* $f(x) = e^{-x^2}$ *on the domain* $[-2.9, 3.5]$, *type II methods.*

Although the interpolants are "somewhat flat" at the local maximum in these figures, they do reach a strict local maximum at an interior point of the mesh interval. These results are consistent with the analysis in subsection 3.5. Also note that the MP constraint produces even "flatter" interpolants in Figs. 12(c,g), and the local maximum of the interpolants occur at a mesh point.

In Figs. 14, the data are the same as those of Figs. 13; but the methods presented are of type II: the second and third-order limiter methods listed above in Figs. 9 and 11. Since the data are smooth, the interpolants of the M3 methods in the right column are considerably smoother than those of the M2 limiters in the left. Note that the interpolants given by the "Superbee" and the M3-B methods in Figs. 14(g) and (h) are a little "fuller" than the others in the same column. As shown below, they also have larger errors, that is, the "visually pleasing" properties of the interpolants by these methods are obtained at the cost of losing some accuracy.

In the next examples, for comparison, the domain is identical to that of Hyman: $[-1.7, 1.9]$. In Table 1, the root mean square error,

$$\text{RMS error} = \left(\frac{1}{3.6} \int_{-1.7}^{1.9} [Pf(x) - e^{-x^2}]^2 \, dx \right)^{1/2}$$

of the interpolants is compared as the mesh is refined from $N = 8$ to $N = 64$. The last column shows the result for an irregular mesh with $N = 32$, $\Delta x_{m+3/2} = 3\Delta x_{m+1/2}$, and every other mesh interval is of the same length. For type I methods, the MG3 constraint has no effect, while the MS3 constraint has no effect only when the mesh is fine enough, that is, when $N \geq 16$. It can be seen that the FD4 formula is highly accurate for a uniform mesh; however, for an irregular mesh, it is much worse than the parabolic and the quartic formulas. Similarly, the average formula is identical to the MP-parabolic method for a uniform mesh, but when the mesh is irregular, it has a larger error. Observe that the MS3, MG3-parabolic, and all the M3 methods are more accurate than the M2-limiter methods. Among the limiters, those with $g'(1) = 1/2$ are more accurate than those without.

Note that for the minmod (or "Superbee") limiter, $\dot{f}_i - f'(x_i)$ and $\dot{f}_{i+1} - f'(x_{i+1})$ are of the same sign in the monotone part of the data since each $\dot{f}_i$ is closer to (or further from) 0 relative to the exact solution $f'(x_i)$. Consequently, $Pf(x) - f(x)$ changes sign in $[x_i, x_{i+1}]$, and the root mean square error favors these methods. A more accurate norm is the average error of $\dot{f}_i$:

$$\text{Average error} = \frac{1}{N+1}\sum_{i=m}^{n} |\dot{f}_i - f'(x_i)|.$$

Method	N = 8	N = 16	N = 32	N = 64	N = 32 irr.
parabolic	4.2E-3	4.1E-4	4.3E-5	4.9E-6	5.0E-5
MG3-parabolic	4.2E-3	4.1E-4	4.3E-5	4.9E-6	5.0E-5
MS3-parabolic	4.9E-3	4.1E-4	4.3E-5	4.9E-6	5.0E-5
MP-parabolic	5.9E-3	2.3E-3	8.0E-5	2.5E-5	8.2E-4
quartic	3.4E-3	7.4E-5	2.3E-6	1.2E-7	7.5E-6
FD4	3.4E-3	7.4E-5	2.3E-6	1.2E-7	2.0E-3
MG3-FD4	3.4E-3	7.4E-5	2.3E-6	1.2E-7	2.0E-3
MS3-FD4	4.5E-3	7.4E-5	2.3E-6	1.2E-7	2.0E-3
MP-FD4	5.6E-3	2.4E-3	6.9E-5	2.4E-5	2.0E-3
M2-Limiters:					
Minmod	1.1E-2	2.9E-3	5.3E-4	1.2E-4	1.3E-3
Van Albada	9.9E-3	2.6E-3	3.3E-4	6.5E-5	1.2E-3
Average	5.9E-3	2.3E-3	8.0E-5	2.5E-5	1.1E-3
Superbee	7.9E-3	3.3E-3	4.7E-4	1.2E-4	1.2E-3
Fritsch-Butland	6.8E-3	2.4E-3	1.5E-4	4.5E-5	1.1E-3
M3-					
MD or UNO	5.2E-3	4.4E-4	4.8E-5	5.3E-6	6.2E-5
V	4.5E-3	3.5E-4	3.7E-5	3.4E-6	8.1E-5
A	4.3E-3	4.1E-4	3.8E-5	3.4E-6	8.5E-5
B	5.9E-3	8.6E-4	9.2E-5	1.0E-5	1.7E-4
M4-					
MD	3.2E-3	1.7E-4	8.1E-6	4.4E-7	1.3E-5
V	2.4E-3	1.2E-4	3.9E-6	1.8E-7	1.0E-5
A	2.7E-3	1.8E-4	5.5E-6	1.8E-7	1.4E-5
B	3.4E-3	2.6E-4	9.6E-6	5.1E-7	2.1E-5

Table 1: *RMS errors for* $f(x) = e^{-x^2}$

Method	N = 8	N = 16	N = 32	N = 64	N = 32 irr.
parabolic	5.8E-2	1.6E-2	4.0E-3	9.9E-4	3.1E-3
MG3-parabolic	5.8E-2	1.6E-2	4.0E-3	9.9E-4	3.1E-3
quartic	4.1E-2	1.7E-3	9.9E-5	6.1E-6	7.4E-5
M4-quartic	4.8E-2	1.7E-3	9.9E-5	6.1E-6	7.4E-5
M3-quartic	5.4E-2	1.7E-3	9.9E-5	6.1E-6	7.4E-5
MP-quartic	6.2E-2	2.3E-2	8.6E-4	3.9E-4	7.4E-3
M2-Limiters:					
Minmod	1.8E-1	1.0E-1	4.7E-2	2.4E-2	5.0E-2
Van Albada	1.3E-1	6.0E-2	1.7E-2	5.7E-3	2.8E-2
Average	7.6E-2	3.6E-2	4.7E-3	1.4E-3	2.8E-2
Average Rational	9.8E-2	4.3E-2	9.0E-3	2.9E-3	2.7E-2
Average Cubic	7.4E-2	4.2E-2	1.0E-2	4.7E-3	2.8E-2
Harmonic Mean	1.1E-1	4.7E-2	1.1E-2	3.7E-3	2.7E-2
Fritsch-Butland	7.6E-2	3.5E-2	9.0E-3	5.8E-3	2.8E-2
Superbee	9.9E-2	7.7E-2	4.0E-2	2.2E-2	4.7E-2
M3-					
MD or UNO	7.1E-2	1.8E-2	4.6E-3	1.1E-3	3.7E-3
V	5.5E-2	9.5E-3	2.5E-3	5.6E-4	2.8E-3
A	5.6E-2	1.1E-2	2.6E-3	5.6E-4	3.0E-3
AR	5.3E-2	1.1E-2	2.6E-3	5.6E-4	2.9E-3
AQ	6.3E-2	1.4E-2	2.9E-3	5.8E-4	3.2E-3
HM	5.2E-2	1.0E-2	2.5E-3	5.6E-4	2.9E-3
FB	6.0E-2	1.5E-2	3.9E-3	9.0E-4	4.2E-3
B	7.9E-2	2.8E-2	7.3E-3	1.8E-3	7.3E-3
M4-					
MD	4.2E-2	5.3E-3	6.1E-4	7.6E-5	4.6E-4
V	2.8E-2	2.4E-3	1.5E-4	1.2E-5	1.8E-4
A	3.4E-2	3.5E-3	1.9E-4	1.2E-5	2.4E-4
AR	3.3E-2	3.5E-3	1.9E-4	1.2E-5	2.4E-4
AQ	3.9E-2	3.6E-3	1.9E-4	1.2E-5	2.4E-4
HM	3.2E-2	3.5E-3	1.9E-4	1.2E-5	2.4E-4
FB	3.8E-2	4.4E-3	3.3E-4	3.3E-5	3.4E-4
B	4.8E-2	6.9E-3	7.2E-4	8.5E-5	6.5E-4

Table 2: *Average errors of* $\dot{f}_i$ *for* $f(x) = e^{-x^2}$

In Table 2, the average errors are listed. As the mesh spacing is reduced by a factor of 2, the error of the parabolic method is reduced by a factor of roughly 4, the quartic method by 16, the minmod and "Superbee" limiter by 2, and the M3 methods by 4. The error of the M3-A method is about half that of the UNO, M3-B, and parabolic methods. This result is consistent with the analysis in subsection 3.5. Also note that limiters with $g'(1) = 1/2$ are more accurate than those without. Among limiters with this property, the average errors are essentially the same. The errors of the M4-MD and M4-B methods reduce by a factor of roughly 8 as the mesh is refined, and for the M4-A method, the factor is larger than 8 in this case. The M3-B and M4-B methods have the largest error in the M3 and M4 classes, respectively.

The examples in this subsection show that accurate, monotonicity-preserving, and "visually pleasing" interpolants are produced for smooth data by the following methods: MS3, MG3, M3-parabolic and quartic, M3-V, M3-A, M4-V, M4-A, and M4-quartic. These methods, except for the M3-V and M4-V, also produce monotonicity-preserving and "visually pleasing" interpolants for rough data, as shown in the previous subsection.

7. Discussion and conclusions. The above methods can be applied to conservation laws, see [16], [29] for some preliminary results in the piecewise linear case. We hope to report further results for the piecewise linear and parabolic methods in a forthcoming paper.

For rough, smooth, monotone or nonmonotone data sets, the methods discussed produce stable, monotonicity-preserving, and "visually pleasing" C^1 Hermite cubic interpolants which are at least third-order accurate. The two simplest methods are the MS3-parabolic and M3-A methods, where the latter has an average error for the derivatives roughly half that of the former. Without increasing the stencil, the M3-quartic method is fourth-order accurate—the highest possible order of accuracy for a cubic interpolant— except adjacent to strict local extrema and inflection points, where the accuracy degenerates to third-order. Uniform fourth-order accuracy can be obtained by using the M4-A or M4-quartic methods.

References

[1] H. Akima, *A new method of interpolation and smooth curve fitting based on local*

procedures, J. Assoc. Comput. Mach., 17 (1970), pp. 589-602.

[2] J. Butland, *A method of interpolating reasonable-shaped curves through any data,* Proc. of Computer graphics 80, Online Publications, Northwood Hills, Middlesex, U.K., 1980, pp. 409-422.

[3] R. E. Carlson & F. N. Fritsch, *Monotone piecewise bicubic interpolation,* SIAM J. Numer. Anal. 22 (1985), pp. 386-400.

[4] C. de Boor, *A Practical Guide to Splines*, Springer-Verlag, New York, 1978.

[5] C. de Boor & B. Swartz, *Piecewise monotone interpolation,* J. Approx. Theory, 21 (1977), pp. 411-416.

[6] R. L. Dougherty, A. Edelman & J. M. Hyman, *Nonnegativity-, monotonicity-, or convexity-preserving cubic and quintic Hermite interpolation,* Math. Comp. 52 (1989), pp. 471-494.

[7] S. C. Eisenstat, K. R. Jackson & J. W. Lewis, *The order of monotone piecewise cubic interpolation,* SIAM J. Numer. Anal. 22 (1985), pp. 1220-1237.

[8] J. C. Ferguson & K. Miller, *Characterization of shape in a class of third degree algebraic curves*, TRW Report 5322-3-5, 1969.

[9] F. N. Fritsch & J. Butland, *A method for constructing local monotone piecewise cubic interpolants,* SIAM J. Sci. Statist. Comput. 5 (1984), pp. 300-304.

[10] F. N. Fritsch & R. E. Carlson, *monotone piecewise cubic interpolation,* SIAM J. Numer. Anal. 17 (1980), pp. 238-246.

[11] J. E. Fromm, *A method for reducing dispersion in convective difference schemes,* J. Comp. Phys.3 (1968), pp. 176-189.

[12] A. Harten, *High resolution schemes for hyperbolic conservation laws,* J. Comp. Phys. 49 (1983), pp. 357-393.

[13] A. Harten, *ENO schemes with subcell resolution,* J. Comp. Phys. 83 (1989), pp. 148-184.

[14] A. Harten, B. Engquist, S. Osher & S. R. Chakravarty, *Uniformly high-order accurate essentially nonoscillatory schemes. III,* J. Comp. Phys. 71 (1987), pp. 231-303.

[15] A. Harten & S. Osher, *Uniformly high-order accurate nonoscillatory schemes. I,* SIAM J. Numer. Anal. 24 (1987), pp. 279-309.

[16] H. T. Huynh, *Second-order accurate nonoscillatory schemes for scalar conservation laws,* Proceedings of the Sixth International Conference on Numerical Methods in Laminar and Turbulent Flows, 1989, Pineridge Press, Swansea, U.K., 1989, pp. 25-38.

[17] J. M. Hyman, *Accurate monotonicity preserving cubic interpolation,* SIAM J. Sci. Statist. Comput. 4 (1983), pp. 645-654.

[18] S. Osher & S. R. Chakravarthy, *High resolution schemes and the entropy condition,* SIAM J. Numer. Anal. 21 (1984), pp. 955-984.

[19] P. J. Rasch & D. L. Williamson, *On shape-preserving interpolation and semi-Lagrangian transport,* SIAM J. Sci. Statist. Comput. 11 (1990), pp. 656-687.

[20] P. L. Roe *Some contributions to the modelling of discontinuous flows,* Lectures on Applied Mathematics 22-part 2 , Amer. Math. Soc. (1985), pp. 163-194.

[21] P. L. Roe *Finite-volume methods for the compressible Navier-Stokes equations,* Proceedings of the Fifth International Conference on Numerical Methods in Laminar and Turbulent Flows, 1987, Pineridge Press, Swansea, U.K., 1987, pp. 2088-2101.

[22] P. L. Roe *A survey of upwind differencing techniques,* Proceedings of the Eleventh International Conference on Numerical Methods in Fluid Dynamics, 1988, Lecture Notes in Physics 323 Springer-Verlag, Berlin/New York, 1989, pp. 69-78.

[23] R. C. Swanson & E. Turkel, *On central-difference and upwind schemes,* NASA Contractor Report 182061, ICASE Report No. 90-44, 1990 (to be published).

[24] P. K. Sweby, *High resolution schemes using flux limiters for hyperbolic conservation laws,* SIAM J. Numer. Anal. 21 (1984), pp. 995-1011.

[25] G. D. van Albada, B. van Leer and W. W. Roberts, Jr., *A comparative study of computational methods in cosmic gas dynamics,* Astronom. and Astrophys. 108 (1982), pp. 76-84.

[26] B. van Leer, *Towards the ultimate conservative difference scheme. II. Monotonicity and conservation combined in a second-order scheme,* J. Comp. Phys. 14 (1974), pp. 361-370.

[27] B. van Leer, *Towards the ultimate conservative difference scheme. IV. A new approach to numerical convection,* J. Comp. Phys. 23 (1977), pp. 276-298.

[28] H. Yang, *An artificial compression method for ENO schemes: the slope modification method,* J. Comp. Phys. 89 (1990), pp. 125-160.

[29] J. W. Yokota & H. T. Huynh, *A nonoscillatory, characteristically convected, finite volume scheme for multidimensional convection problems,* NASA Technical Memorandum 102354, 1990 (unpublished).

[30] S. Zalesak, *A preliminary comparison of modern shock-capturing schemes: Linear advection,* Advances in Computer methods for Partial Differential Equations, VI, IMACS, North-Holland, Amsterdam/New York, 1987.

NASA
National Aeronautics and Space Administration

Report Documentation Page

1. Report No.
NASA TM-103789

2. Government Accession No.

3. Recipient's Catalog No.

4. Title and Subtitle
Accurate Monotone Cubic Interpolation

5. Report Date
March 1991

6. Performing Organization Code

7. Author(s)
Hung T. Huynh

8. Performing Organization Report No.
E-6066

10. Work Unit No.
505-62-52

9. Performing Organization Name and Address
National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Ohio 44135-3191

11. Contract or Grant No.

13. Type of Report and Period Covered
Technical Memorandum

12. Sponsoring Agency Name and Address
National Aeronautics and Space Administration
Washington, D.C. 20546-0001

14. Sponsoring Agency Code

15. Supplementary Notes
Responsible person, Hung T. Huynh, (216) 433-5852.

16. Abstract

Monotone piecewise cubic interpolants are simple and effective. They are generally third-order accurate, except near strict local extrema where accuracy degenerates to second-order due to the monotonicity constraint. Algorithms for piecewise cubic interpolants, which preserve monotonicity as well as uniform third and fourth-order accuracy, are presented. The gain of accuracy of obtained by relaxing the monotonicity constraint in a geometric framework in which the median function plays a crucial role.

17. Key Words (Suggested by Author(s))
Monotonicity-preserving
Cubic Hermite interpolation

18. Distribution Statement
Unclassified – Unlimited
Subject Category 64

19. Security Classif. (of this report)
Unclassified

20. Security Classif. (of this page)
Unclassified

21. No. of pages
62

22. Price*
A04

*For sale by the National Technical Information Service, Springfield, Virginia 22161