JOURNAL OF COMPUTATIONAL PHYSICS **128,** 475–488 (1996)
ARTICLE NO. 0225

A Conservative Staggered-Grid Chebyshev Multidomain Method for Compressible Flows. II. A Semi-Structured Method

DAVID A. KOPRIVA

Supercomputer Computations Research Institute, The Florida State University, Tallahassee, Florida 32306

Received January 18, 1996; revised May 28, 1996

We present a Chebyshev multidomain method that can solve systems of hyperbolic equations in conservation form on an unrestricted quadrilateral subdivision of a domain. Within each subdomain the solutions and fluxes are approximated by a staggered-grid Chebyshev method. Thus, the method is unstructured in terms of the subdomain decomposition, but strongly structured within the subdomains. Communication between subdomains is done by a mortar method in such a way that the method is globally conservative. The method is applied to both linear and nonlinear test problems and spectral accuracy is demonstrated. © 1996 Academic Press, Inc.

1. INTRODUCTION

In the first paper [1], we introduced a staggered-grid Chebyshev multidomain method for the solution of inviscid compressible flow problems. The grid used was analogous to the fully staggered grids used in some finite difference computations of compressible flows, e.g. [2]. For the staggered-grid spectral method, the unknowns are approximated by global polynomials of degree N-1 in each space dimension, which pass through values defined at the Chebyshev–Gauss quadrature points. The fluxes are approximated by polynomials of degree N that pass through values defined on the Gauss–Lobatto points. Since the Gauss points fall strictly between the Lobatto points [3], the results is a staggering of the solution and flux values.

The staggered grid leads to a simpler and more flexible multidomain method than one based on a Lobatto grid alone, e.g. [4]. It is simpler because subdomain corners are not included in the approximation, so special conditions do not need to be derived for them. Its flexibility comes from the fact that only the normal fluxes, not the flux derivatives, need to be continuous across interfaces where the subdomains meet, making grid generation less restrictive. See [1] for details.

The flexibility of the method presented in [1] is still limited, however, by the restriction that the calculation of the unique flux along a subdomain interface requires the grid points to coincide there. We refer to this approximation as *conforming.* In general, the requirement that the interfaces be conforming means that the subdomains must intersect along an entire side or at a corner point. If two subdomains intersect along a side, then polynomial approximation orders must be the same along the interface between them [5]. An example of a two-subdomain conforming subdivision of a square is shown in Fig. 1a.

The limits imposed by the conforming restriction make it impossible to do local refinement by subdividing existing subdomains, or by increasing the polynomial order within selected subdomains. If refinement is necessary within one subdomain, it is necessary to refine also its neighbors. This makes the approximation more expensive than necessary, since the overall grid is often refined where refinement is not needed.

To be completely flexible, we would like the method to be able to use an arbitrary tiling of a domain by quadrilaterals. This would be similar to zonal finite difference methods that have long been in use in the finite difference community, e.g. [6]. Since within each subdomain the strong tensor product structure of the spectral approximation would remain, the result would be a semi-structured method. The flexibility of the semi-structured method would allow commonly available block structured grid generation methods [7] to be used to generate the subdomains.

The semi-structured method can be developed by loosening the restriction that the fluxes be continuous across an interface, giving a nonconforming patching of the subdomains. Figures 1b–1d show three nonconforming topologies. In the first (Fig. 1b), which we call order refinement, the subdomains intersect along a full side, but the approximation order changes across the interface. The second, Fig. 1c, shows a situation that comes from a subdivision of the conforming grid, Fig. 1a. Finally, Fig. 1d shows the fully nonconforming case, where the interface between two subdomains is not a full side of either.

Nonconforming spectral domain decomposition approximations for elliptic problems and for the incompressible Navier–Stokes equations on grids like those shown in Fig. 1 first appeared in the late 1980s [8–11]. Most notable was the mortar element method, in which a one-dimensional polynomial function space called a mortar was defined

0021-9991/96 $18.00

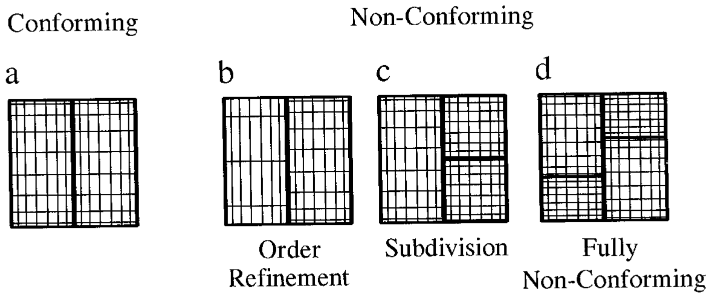

FIG. 1. Conforming vs nonconforming grids.

along subdomain interfaces. It was with this mortar space that the patching of the subdomains was accomplished. Details can be found in the cited references.

In this paper, we present a semi-structured method that uses a nonconforming mortar approximation for the solution of hyperbolic systems such as the Euler gas-dynamics equations. Interior to the subdomains, the method uses the conservative staggered-grid approximation presented in [1]. The result is a fully flexible approximation: Subdomains can be subdivided. Polynomial orders can be adjusted within subdomains without affecting neighboring subdomains. There is no restriction on how the subdomains tile the full domain, as long as they do not overlap.

The paper begins with a presentation of the equations, followed by a review of the staggered-grid conforming approximation. In Section 4, we introduce the mortar method for treating the subdomain interfaces. Test problems, both linear and nonlinear, are presented in Section 5 to show that the approximation is spectrally convergent. Conclusions are drawn in the last section.

2. THE EQUATIONS

In this paper we consider the approximation of hyperbolic systems in conservative form,

$$\frac{\partial \mathbf{Q}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} = \zeta \mathbf{H}, \tag{1}$$

where $\mathbf{Q}$ is the vector of solution unknowns and $\mathbf{F}(\mathbf{Q})$ and $\mathbf{G}(\mathbf{Q})$ are the advective flux vectors. For the Euler gas-dynamics equations in two space dimensions,

$$\mathbf{Q} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho e \end{bmatrix} \quad \mathbf{F} = \begin{bmatrix} \rho u \\ p + \rho u^2 \\ \rho u v \\ u(\rho e + p) \end{bmatrix} \quad \mathbf{G} = \begin{bmatrix} \rho v \\ \rho u v \\ p + \rho v^2 \\ v(\rho e + p) \end{bmatrix}. \tag{2}$$

In (2), $\rho e = p/(\gamma - 1) + \rho(u^2 + v^2)/2$ and we assume $\gamma = 1.4$. For two-dimensional problems, $\zeta = 0$. For axisymmetric problems, we interpret x as the axial coordinate and y as the radial coordinate, and we set $\zeta = 1$. For the gas-dynamics equations,

$$\mathbf{H} = \frac{1}{y} \begin{bmatrix} \rho v \\ \rho u v \\ \rho v^2 \\ v(\rho e + p) \end{bmatrix}. \tag{3}$$

In the multidomain approximation [4], the region under consideration is divided into K nonoverlapping subdomains, Ω^K. Each subdomain is mapped individually onto the unit square. Under the mapping, Eq. (1) becomes

$$\frac{\partial \tilde{\mathbf{Q}}}{\partial t} + \frac{\partial \tilde{\mathbf{F}}(\mathbf{Q})}{\partial X} + \frac{\partial \tilde{\mathbf{G}}(\mathbf{Q})}{\partial Y} = \zeta \tilde{\mathbf{H}}, \tag{4a}$$

where $\tilde{\mathbf{Q}} = J\mathbf{Q}$ and

$$\tilde{\mathbf{F}} = y_Y^N \mathbf{F} - x_Y^N \mathbf{G}, \quad \tilde{\mathbf{G}} = -y_X^N \mathbf{F} + x_X^N \mathbf{G}, \quad \tilde{\mathbf{H}} = J\mathbf{H}$$

$$J(X, Y) = x_X^N y_Y^N - x_Y^N y_X^N. \tag{4b}$$

3. THE CONFORMING STAGGERED-GRID APPROXIMATION

The staggered-grid approximation [1] computes the solution values, $\tilde{\mathbf{Q}}$, and the fluxes $\tilde{\mathbf{F}}$ and $\tilde{\mathbf{G}}$ on separate grids. These grids are tensor products of the Lobatto grid, X_j, and the Gauss grid, $\overline{X}_{j+1/2}$, mapped onto [0, 1]:

$$\begin{aligned} X_j &= \frac{1}{2}\left(1 - \cos\left(\frac{j\pi}{N}\right)\right), \quad j = 0, 1, ..., N, \\ \overline{X}_{j+1/2} &= \frac{1}{2}\left(1 - \cos\left(\frac{2j+1}{2N+2}\pi\right)\right), \quad j = 0, 1, ..., N-1. \end{aligned} \tag{5}$$

On the Lobatto and Gauss grids, we define two Lagrange interpolating polynomials,

$$l_j(\xi) = \prod_{\substack{i=0\\ i\neq j}}^{N} \left(\frac{\xi - X_i}{X_j - X_i}\right), \tag{6}$$

$$h_{j+1/2}(\xi) = \prod_{\substack{i=0\\ i\neq j}}^{N-1} \left(\frac{\xi - \overline{X}_{i+1/2}}{\overline{X}_{j+1/2} - \overline{X}_{i+1/2}}\right). \tag{7}$$

We see that $l_j(x) \in \mathbf{P}_N(x)$, and $h_{j+1/2}(\xi) \in \mathbf{P}_{N-1}$, where $\mathbf{P}_N$ is the space of polynomials of degree less than or equal to N.

The mapping of each subdomain onto the unit square is done by a static isoparametric transformation. Let the vector function $\mathbf{g}(s)$, $0 \le s \le 1$, define a parametric curve. The polynomial of degree N that interpolates $\mathbf{g}$ at the Lobatto points is

$$\Gamma(s) = \sum_{j=0}^{N} \mathbf{g}(s_j) l_j(s). \tag{8}$$

Four such polynomial curves, $\Gamma_m(s)$, $m = 1, 2, 3, 4$, counted counterclockwise, bound each subdomain. As in [1], we map each subdomain onto the unit square by the linear blending formula,

$$\begin{aligned}\mathbf{x}^N(X, Y) = {} & (1 - Y)\Gamma_1(X) + Y\Gamma_3(X) + (1 - X)\Gamma_4(Y) \\ & + X\Gamma_2(Y) - \mathbf{x}_1(1 - X)(1 - Y) \\ & - \mathbf{x}_2 X(1 - Y) - \mathbf{x}_3 XY - \mathbf{x}_4(1 - X)Y,\end{aligned} \tag{9}$$

where the $\mathbf{x}_j$'s represent the locations of the corners of the subdomain, counted counterclockwise.

The solution unknowns are approximated at $(\overline{X}_{i+1/2}, \overline{Y}_{j+1/2})$, $i, j = 0, 1, ..., N - 1$, which we will call the Gauss/Gauss points. The interpolant through these unknowns is a polynomial in $\mathbf{P}_{N-1,N-1} = \mathbf{P}_{N-1} \otimes \mathbf{P}_{N-1}$,

$$\tilde{\mathbf{Q}}(X, Y) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \tilde{\mathbf{Q}}_{i+1/2,j+1/2} h_{i+1/2}(X) h_{i+1/2}(Y). \tag{10}$$

The horizontal fluxes are approximated at the Lobatto/Gauss points $(X_i, \overline{Y}_{j+1/2})$, $i = 0, 1, ..., N$; $j = 0, 1, ..., N - 1$, computed from the polynomial (10)

$$\begin{aligned}\tilde{\mathbf{F}}_{i,j+1/2} = {} & y_Y^N(X_i, \overline{Y}_{j+1/2})\mathbf{F}(\overline{Q}(X_i, \overline{Y}_{j+1/2})) \\ & - x_Y^N(X_i, \overline{Y}_{j+1/2})\mathbf{G}(\overline{Q}(X_i, \overline{Y}_{j+1/2})).\end{aligned} \tag{11}$$

Finally, the vertical fluxes are approximated at the Gauss/Lobatto points $(\overline{X}_{i+1/2}, Y_j)$, $i = 0, 1, ..., N - 1$; $j = 0, 1, ..., N$, are are computed as

$$\begin{aligned}\tilde{\mathbf{G}}_{i+1/2,j} = {} & -y_X^N(\overline{X}_{i+1/2}, Y_j)\mathbf{F}(\overline{Q}(\overline{X}_{i+1/2}, Y_j)) \\ & + x_X^N(\overline{X}_{i+1/2}, Y_j)\mathbf{G}(\overline{Q}(\overline{X}_{i+1/2}, Y_j)).\end{aligned} \tag{12}$$

The heart of the multidomain approximation is how the interfaces between subdomains are treated. In the conforming approximation (Fig. 1a) the interface points between two neighboring subdomains coincide. However, the two solutions at the interface need not, since they are computed independently from the interpolant through the Gauss/Gauss points in each subdomain. From these two values, however, a unique flux is computed so that the characteristic propagation of waves is accounted for. For linear problems, we use a flux vector splitting. For the Euler gas-dynamics equations, we use a Roe solver [9] with the entropy fix to compute the normal flux at the interface from these two values. Inflow, outflow, and wall boundaries can be treated by specifying the boundary conditions as the extra solution value in the Riemann solver. Details can be found in [1].

Once the fluxes are computed, we form the semi-discrete approximation for the solution on the Gauss/Gauss grid. For each subdomain

$$\begin{aligned}& \left.\frac{d\tilde{\mathbf{Q}}}{dt}\right|_{i+1/2,j+1/2} + \left[\frac{\partial\tilde{\mathbf{F}}}{\partial X} + \frac{\partial\tilde{\mathbf{G}}}{\partial Y}\right]_{i+1/2,j+1/2} \\ & \qquad = \zeta\tilde{\mathbf{H}}\big|_{i+1/2,j+1/2} \quad \begin{cases} i = 0, 1, ..., N - 1, \\ j = 0, 1, ..., N - 1, \end{cases}\end{aligned} \tag{13}$$

where the derivatives, defined as

$$\begin{aligned}\left.\frac{\partial\tilde{\mathbf{F}}}{\partial X}\right|_{i+1/2,j+1/2} &= \sum_{n=0}^{N} \tilde{\mathbf{F}}_{n,j+1/2} l_n'(\overline{X}_{i+1/2}) \\ \left.\frac{\partial\tilde{\mathbf{G}}}{\partial Y}\right|_{i+1/2,j+1/2} &= \sum_{m=0}^{N} \tilde{\mathbf{G}}_{i+1/2,m} l_m'(\overline{Y}_{j+1/2})\end{aligned} \tag{14}$$

are computed by matrix multiplication. Equation (13) is then integrated in time by a two-level low-storage Runge–Kutta scheme.

4. A NONCONFORMING MORTAR APPROXIMATION

The only differences between the conforming and the nonconforming approximations come from how the fluxes are to be computed along the interfaces between subdomains. In the conforming case (Fig. 1a), there are two solution values at each interface point from which a single flux can be computed directly. In the nonconforming cases (Figs. 1c–d), however, the grid lines do not necessarily match along the interface, so that point-by-point transfer of information cannot be made from a subdomain to its neighbor.

We have chosen to implement the transfer of information between subdomains by a mortar method [9]. The basic idea is that the mortar (the "cement") connects neighboring subdomains (the "bricks"). In our method, the two-dimensional subdomains communicate only with an intermediate one-dimensional construct, called a mortar, not with neighboring subdomains (Fig. 2). In practice, a projection of the solution values is made from the contributing subdomain faces onto a mortar. It is on the mortar, and not on the subdomains themselves, that the Riemann problem is solved to give a unique flux. The computed flux is then projected back onto the subdomain faces.

The use of a mortar has several advantages over direct subdomain-to-subdomain communication of solution values. First, each mortar will communicate with at most two subdomain faces. The flux computations on a mortar can be made independently of the subdomains that contribute to it. Finally, the work of computing the interface fluxes is not duplicated.

The projections from the subdomain faces to the mortars and back must be carefully chosen. We will use a least squares matching of the polynomials. It is possible to simply interpolate from the grid points on a subdomain face to a mortar and vice versa, since the polynomial approximations are already specified for the spectral method by Eq. (10). Either one can be performed computationally by matrix multiplication, although the matrix coefficients are different. For elliptic problems, however, it was shown [5, 11] that a least square projection was necessary to retain spectral convergence globally.

The conditions for determining the solutions of the hyperbolic system along the mortar are different from those in the elliptic case [11]. Since we are computing the solution to the system of equations in conservation form, we set two necessary conditions to be satisfied by the treatment of the interfaces. The first requirement is that the approximation retains global conservation. This will determine the choice of the projection operator from the mortar back onto the subdomain faces. The second requirement we will term the *outflow condition.*

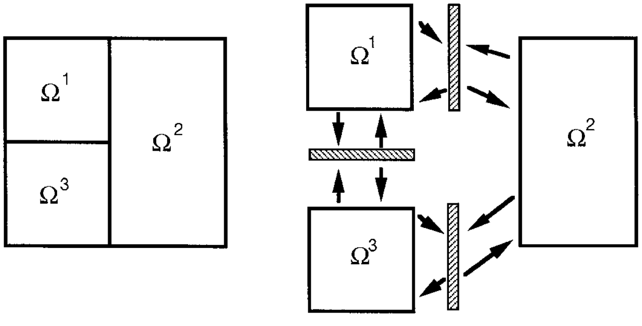

FIG. 2. Diagram of mortar communication between three subdomains that subdivide a square.

The outflow condition arises from the fact that, in a hyperbolic problem, waves should pass through an interface unaffected by downwind contributions. If the problem is scalar, for instance, this means that of the two solutions at an interface, the solution from the subdomain from which the characteristic comes (the "upwind side") is used. To affect this choice with a mortar, it is necessary that the solution along the upwind face be unchanged after projecting onto the mortar and then back onto the face. If this is true, we say that the approximation satisfies the outflow condition.

Both the outflow condition and conservation can be satisfied by a least squares matching of the face and mortar solutions. Least squares projection can be viewed as requiring that any approximation errors be orthogonal to the polynomial space on which the solution is being projected. Equivalently, it can be viewed as truncation, or padding by zeros, of the orthogonal polynomial representation of the function. Thus, it ensures that the patching does not introduce errors into modes that are represented on the faces or the mortars.

We will describe mortar approximations for the three nonconforming topologies shown in Fig. 1. The first two topologies occur when a conforming topology is locally refined. The first means only that the polynomial order along a subdomain face differs from that of its neighbor. The second situation arises when a subdomain itself is subdivided without subdividing the neighbor. The last topology is the most general one and does not come from an initially conforming grid.

In the discussion that follows, we will consider only the approximation of a scalar problem

$$\frac{\partial \tilde{Q}}{\partial t} + \frac{\partial \tilde{F}(Q)}{\partial X} + \frac{\partial \tilde{G}(Q)}{\partial Y} = 0. \tag{15}$$

The extension to a system is direct. The only difference in the case of a system is that along the mortar a characteristic resolution of the two solutions must be made to compute the flux. That mortar flux calculation is identical to the conforming case [1].

4.1. *Order Refinement*

The simplest case of a nonconforming approximation (Fig. 1b) occurs when two subdomains meet along a full side, but the polynomial orders of the solution on either side of the interface are not the same. A schematic diagram of two contributing sides of subdomains "L" and "R" and the mortar is shown in Fig. 3. For this topology, the mortar, denoted by Ξ, extends the full length of both sides of the two subdomains. Note that the approximation needs to consider only the two subdomains whose faces coincide, since the staggered grid approximation does not include subdomain corners.

In the general case, the subdomain interfaces and the mortar between them will be defined through Eq. (8) by polynomial approximations, Γ, to the boundary curve, $\mathbf{g}$. To be exactly coincident, the three polynomial approximations must be identical. This condition automatically holds for the conforming approximation, but not for the nonconforming. The restriction of coincidence means that the order of the curves must be chosen to be no more than the lowest order of the contributing subdomains. Since the subdomain interfaces are artificial interior boundaries (often straight lines), we have the freedom to make this restriction.

Only the solution along the subdomain faces must be transferred to the mortar. So define the solutions along the faces as $U^{\mathrm{L}}_{j+1/2} = Q^{\mathrm{L}}(1, \overline{Y}_{j+1/2})$, $j = 0, 1, \ldots, M^{\mathrm{L}} - 1$, and $U^{\mathrm{R}}_{j+1/2} = Q^{\mathrm{R}}(0, \overline{Y}_{j+1/2})$, $j = 0, 1, \ldots, M^{\mathrm{R}} - 1$. The polynomials along the faces that interpolate these values are

$$\begin{aligned} U^{\mathrm{L}}(\xi) &= \sum_{j=0}^{M^{\mathrm{L}}-1} U^{\mathrm{L}}_{j+1/2} h^{\mathrm{L}}_{j+1/2}(\xi) \in \mathbf{P}_{M^{\mathrm{L}}-1}, \\ U^{\mathrm{R}}(\xi) &= \sum_{j=0}^{M^{\mathrm{R}}-1} U^{\mathrm{R}}_{j+1/2} h^{\mathrm{R}}_{j+1/2}(\xi) \in \mathbf{P}_{M^{\mathrm{R}}-1}, \end{aligned} \tag{16}$$

where $\xi \in [0, 1]$ is the local subdomain coordinate. On the mortar itself, we represent the two solutions as the polynomials ϕ^{L} and ϕ^{R} defined by

$$\phi^{\mathrm{L,R}}(\xi) = \sum_{j=0}^{J-1} \phi^{\mathrm{L,R}}_{j+1/2} h^{\Xi}_{j+1/2}(\xi) \in \mathbf{P}_{J-1}. \tag{17}$$

To be able to satisfy the outflow condition, the polynomial order of the mortar space must be sufficiently large to include both $\mathbf{P}_{M^{\mathrm{L}}-1}$ and $\mathbf{P}_{M^{\mathrm{R}}-1}$. So as not to require an excessive amount of work to compute the projections, we choose $J = \max(M^{\mathrm{L}}, M^{\mathrm{R}})$.

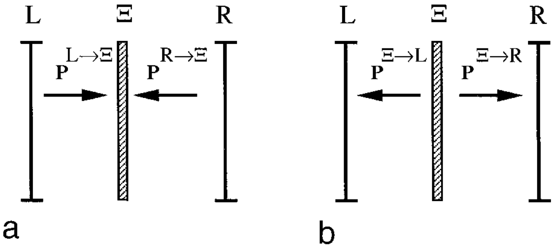

FIG. 3. Schematic of order refinement: (a) subdomain to mortar projections; (b) mortar to subdomain projections.

To compute the flux for each subdomain face, a three-step procedure is used. First, the two solutions $U^{\mathrm{L,R}}(\xi)$ are projected onto the mortar space by projections $\mathbf{P}^{\mathbf{L}\to\Xi}$ and $\mathbf{P}^{\mathbf{R}\to\Xi}$ to give $\phi^{\mathrm{L,R}}$. The two values on the mortar are then used to compute a unique mortar flux, which is evaluated as if the mortar is a conforming interface. Finally, the mortar flux is projected back onto the subdomain faces by the projections $\mathbf{P}^{\Xi\to\mathrm{L}}$ and $\mathbf{P}^{\Xi\to\mathrm{R}}$. These projection operators are chose so that the approximation is globally conservative and satisfies the outflow condition.

4.1.1. *Subdomain → Mortar Projection.* A diagram of the projection of a solution onto a mortar for the order refinement case is shown in Fig. 3a. Since we have chosen the order of the polynomial approximation on the mortar (the "mortar order") to be equal to the maximum order of the polynomials on the two contributing subdomains, one of the projection operators is the identity. For convenience, we will assume that $J = M^{\mathrm{R}}$ so that $\phi^{\mathrm{R}}(\xi) = U^{\mathrm{R}}(\xi)$ and $\mathbf{P}^{\mathbf{R}\to\Xi} = \mathbf{I}$.

For the projection of the lower order space onto the mortar, we use the unweighted L^2 projection. Thus, we ask that the approximation along the mortar satisfy

$$\min_{\phi^{\mathrm{L}} \in \mathbf{P}_J} \int_0^1 (\phi^{\mathrm{L}} - U^{\mathrm{L}})^2 \, d\xi \tag{18a}$$

$$\Leftrightarrow \int_0^1 (\phi^{\mathrm{L}} - U^{\mathrm{L}}) h^{\Xi}_{m+1/2}(\xi) \, d\delta = 0, \quad m = 0, 1, \ldots, J - 1. \tag{18b}$$

Substitution of the definitions (16) and (17) into (18b) gives

$$\begin{aligned} &\sum_{j=0}^{N^{\mathrm{L}}-1} U^{\mathrm{L}}_{j+1/2} \left[\int_0^1 h^{\mathrm{L}}_{j+1/2} h^{\Xi}_{m+1/2} \, d\xi \right] \\ &\quad = \sum_{j=0}^{J-1} \phi^{\mathrm{L}}_{j+1/2} \left[\int_0^1 h^{\Xi}_{j+1/2} \, h^{\Xi}_{m+1/2} \, d\xi \right], \\ &m = 0, 1, \ldots, J - 1. \end{aligned} \tag{19}$$

Now, define the matrix elements

$$\begin{aligned} S^{\mathrm{L}}_{mj} &= \int_0^1 h^{\mathrm{L}}_{j+1/2} h^{\Xi}_{m+1/2} \, d\xi \\ M_{mj} &= \int_0^1 h^{\mathrm{L}}_{j+1/2} h^{\Xi}_{m+1/2} \, d\xi \end{aligned} \tag{20}$$

so that (19) becomes $\mathbf{S}^{\mathrm{L}}\mathbf{U}^{\mathrm{L}} = \mathbf{M}\Phi$, where $\mathbf{U}$ is the vector of discrete solution values along the face and Φ is the vector of solution values along the mortar. The integrals in (20) can be computed exactly by a Clenshaw–Curtis quadrature [3] on $2J + 2$ Lobatto points. We then define the projection operator by

$$\Phi = \mathbf{P}^{\mathrm{L}\to\Xi}\mathbf{U}^{\mathrm{L}} = \mathbf{M}^{-1}\mathbf{S}^{\mathrm{L}}\mathbf{U}^{\mathrm{L}}. \qquad (21)$$

The matrix $\mathbf{P}^{\mathbf{R}\to\Xi}$ can be computed once at the beginning of the computation and stored for use as needed.

Since the computation of the projection operator requires the inversion of the matrix $\mathbf{M}$, it is important to consider the conditioning of that system. We find numerically that the growth of the condition number in the maximum norm is weak with matrix size, $\kappa \approx 1.36J^{0.79}$. For the maximal polynomial orders that we typically use ($J \approx 20$), $\kappa \approx 15$.

4.1.2. *Mortar → Subdomain Projection.* Once the mortar functions (17) are computed, the normal flux is calculated as described in [1]. This flux must then be projected back onto the subdomains (Fig. 3b). Let $\Psi \in \mathbf{P}_{J-1}$ be the mortar contravariant flux and let $\tilde{F}^{\mathrm{L}}$ and $\tilde{F}^{\mathrm{R}}$ be the two subdomain fluxes to be computed from it. Since $M^{\mathrm{R}} = J$, we have immediately that $\tilde{F}^{\mathrm{R}} = \Psi$. To get the flux on the left, we require that

$$\int_0^1 (\tilde{F}^{\mathrm{L}} - \Psi) h^{\mathrm{L}}_{m+1/2}\, d\xi = 0, \quad m = 1, 2, ..., M^{\mathrm{L}} - 1, \qquad (22)$$

i.e., that the projection error is orthogonal to $\mathbf{P}_{M^{\mathrm{L}}-1}$. Then, in matrix form,

$$\mathbf{M}^{\mathrm{L}}\tilde{\mathbf{F}}^{\mathrm{L}} = \mathbf{S}^{\mathrm{L}}\Psi, \qquad (23)$$

where

$$M^{\mathrm{L}}_{m,j} = \int_0^1 h^{\mathrm{L}}_{j+1/2} h^{\mathrm{L}}_{m+1/2}\, d\xi, \quad m, j = 0, 1, ..., M^{\mathrm{L}} - 1$$

$$S^{\mathrm{L}}_{m,j} = \int_0^1 h^{\Xi}_{j+1/2} h^{\mathrm{L}}_{m+1/2}\, d\xi, \quad \begin{cases} m = 0, 1, ..., M^{\mathrm{L}} - 1, \\ j = 0, 1, ..., J - 1. \end{cases}$$

As before, these integrals can be computed exactly by quadrature. From (23) the projection of the flux onto the subdomain face is

$$\tilde{\mathbf{F}}^{\mathrm{L}} = \mathbf{P}^{\Xi\to\mathrm{L}}\Psi = (\mathbf{M}^{\mathrm{L}})^{-1}\mathbf{S}\Psi. \qquad (24)$$

4.2. *Subdomain Refinement*

The next level of flexibility allows subdomains be subdivided locally. For simplicity, we consider only the case when a side is subdivided into two. The refinement to three or more subdomains is a simple extension of this refinement.

When a subdomain is refined as shown in Fig. 1c, there are two possible choices for the mortars (Fig. 4). In the first, two mortars coincide with the "short" faces of Ω^2 and Ω^3. The second choice uses a single mortar that coincides with the "long" face of Ω^1.

We choose the topology in Fig. 4a, since it is the one that can satisfy the outflow condition. In the second case, the projection from Ω^2 and Ω^3 onto the mortar is a projection of a piecewise polynomial space onto a single polynomial space. The former is the larger space, since it includes approximations that are discontinuous at the point where Ω^2 and Ω^3 meet. The outflow condition requires that the projection of face values from Ω^2 and Ω^3 onto a mortar, and the subsequent projection back onto the faces returns the original polynomial functions. This is clearly impossible in case (4b), since the projection back onto the faces returns a continuous function. By using two mortars, however, as shown in Fig. 4a, it is possible to construct projections that recover the original polynomials on all three subdomains. This situation differs from the mortar element method for elliptic problems [11], which require stronger regularity conditions than hyperbolic problems.

As before, the curves for the subdomain surfaces and mortars must coincide. In the case of subdivision, this can be accomplished by choosing the interface to be a polynomial whose order is the lowest of the orders, three polynomials being used along the faces of the contributing interfaces. This polynomial should be defined along the "long" face of Ω^1. Again, we have the freedom to do this when defining subdomain boundaries.

We also define as before the solution approximations along a face as

$$\begin{aligned} U^1(\xi) &= \sum_{j=0}^{M^1-1} U^1_{j+1/2} h^1_{j+1/2}(\xi) \in \mathbf{P}_{M^1-1} \\ U^2(\xi) &= \sum_{j=0}^{M^2-1} U^2_{j+1/2} h^2_{j+1/2}(\xi) \in \mathbf{P}_{M^2-1} \\ U^3(\xi) &= \sum_{j=0}^{M^3-1} U^3_{j+1/2} h^3_{j+1/2}(\xi) \in \mathbf{P}_{M^3-1} \end{aligned} \qquad (25)$$

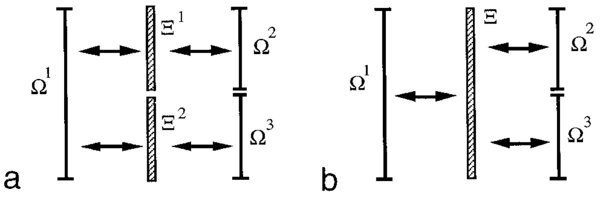

FIG. 4. Two mortar configurations for subdomain refinement.

for local subdomain coordinates $\xi \in [0, 1]$. We also define four mortar functions

$$\phi^{1,(\mathrm{L,R})}(z) = \sum_{j=0}^{J^1-1} \phi_{j+1/2}^{1(\mathrm{L,R})} h_{j+1/2}^{\Xi^1}(z) \in \mathbf{P}_{J^1-1}$$
$$\phi^{2,(\mathrm{L,R})}(z) = \sum_{j=0}^{J^2-1} \phi_{j+1/2}^{2(\mathrm{L,R})} h_{j+1/2}^{\Xi^2}(z) \in \mathbf{P}_{J^2-1} \tag{26}$$

which are functions of the local mortar coordinate, $z \in [0, 1]$. Finally, we define the variables o^k and s^k to be the offset and the scale of a mortar with respect to the subdomain Ω^k that contributes to it. Thus, for $z \in [0, 1]$, $\xi^k = o^k + s^k z$.

The orders of the mortar polynomials must be chosen sufficiently high so that the outflow condition can be satisfied. This means that the mortar order must be at least as large as the largest subdomain order of all contributing subdomains. Thus, we choose $J^1 = \max(M^1, M^2)$ and $J^2 = \max(M^1, M^3)$.

4.2.1. *Subdomain → Mortar Projections.* To compute the mortar functions (26), we also use the unweighted L^2 projection. Thus, we seek polynomials on the two mortars that best approximate the polynomial along the contributing face. For each mortar Ξ and each subdomain contributor Ω, this means that we require

$$\int_0^1 (\phi(z) - U(o + sz)) h_{j+1/2}^{\Xi}\, dz = 0, \quad j = 0, 1, \ldots, J. \tag{27}$$

Then the vector of the solution values along the mortar can be computed by

$$\Phi = \mathbf{P}^{\Omega\to\Xi}\mathbf{U} = \mathbf{M}^{-1}\mathbf{S}\mathbf{U}, \tag{28}$$

where, this time

$$M_{m,j} = \int_0^1 h_{j+1/2}^{\Xi} h_{m+1/2}^{\Xi}\, dz\, m, \quad j = 0, 1, \ldots, J-1, \tag{29a}$$

$$S_{m,j}^{\Omega} = \int_0^1 h_{j+1/2}^{\Xi} h_{m+1/2}^{\Omega}(o + sz)\, dz, \quad \begin{array}{l} m = 0, 1, \ldots, M-1, \\ j = 0, 1, \ldots, J-1. \end{array} \tag{29b}$$

Note that the matrices in (20) are just special cases of those in (29) with $o = 0$ and $s = 1$.

4.2.2. *Mortar → Subdomain Projections.* Once the flux is computed on the mortars, as if the approximation is conforming, it is projected back onto the subdomain faces. The projections $\mathbf{p}^{\Xi^1\to\Omega^2}$ and $\mathbf{p}^{\Xi^2\to\Omega^3}$ are exactly as described by (22)–(24). If the mortar order and the subdomain polynomial order are the same, the projection simplifies to the identity, and a simple copy of the flux from the mortar to the face can be made.

The projection from the mortars to a subdomain is a little more complicated in the case where two mortars contribute to a subdomain, as is the case for Ω^1 in Fig. 4a. The piecewise polynomial that represents the fluxes along the mortars, possibly discontinuous, must be used to compute the continuous polynomial along the face. As before, we seek the best polynomial on the face that approximates the mortar solutions in the least squares sense. Posing the projection problem in this way means that there is no ambiguity if a grid point falls at the intersection of the two mortars, unlike the situation that would result if pointwise interpolation were used.

To obtain the least squares projection, we seek the flux that satisfies

$$\int_0^{o^1} (\tilde{F}(\xi) - \Psi^{\Xi^2}(\xi)) h_{j+1/2}(\xi)\, d\xi$$
$$+ \int_{o^1}^1 (\tilde{F}(\xi) - \Psi^{\Xi^1}(\xi)) h_{j+1/2}(\xi)\, d\xi, \quad j = 1, 2, \ldots, M-1,$$
$$= 0, \tag{30}$$

and we define

$$\Psi = \begin{cases} \Psi^{\Xi^1}\left(\dfrac{\xi - o^1}{s^1}\right), & o^{1+} \le \xi \le 1, \\ \Psi^{\Xi^2}\left(\dfrac{\xi - o^2}{s^2}\right), & 0 \le \xi \le o^{1-}. \end{cases} \tag{31}$$

If we now define

$$S_{i,j}^{(k)} = s^k \int_0^1 h_{i+1/2}^{\Xi} h_{j+1/2}^{k}(o^k + s^k z)\, dz, \tag{32a}$$

$$M_{ij} = \int_0^1 h_{i+1/2}(\xi) h_{k+1/2}(\xi)\, d\xi, \tag{32b}$$

we can write

$$\tilde{\mathbf{F}}^1 = \sum_{k=1}^{2} \mathbf{P}^{(k)}\Psi^{(k)}, \tag{33}$$

where $\mathbf{P}^{(k)} = \mathbf{M}^{-1}\mathbf{S}^{(k)}$ is the projection matrix.

4.3. *Fully Nonconforming Interfaces*

The final topology, shown in Fig. 1d, is characterized by subdomains whose faces only partially overlap. In order to satisfy the outflow condition, we choose the mortars to cover the intersections of two subdomain faces, as shown in Fig. 5. We note again that this choice is different than the choice one must make in the elliptic case [9], where the concept of a supermortar was introduced to give the solution sufficient regularity.

In practice, this situation is handled as in subdivision. The interfaces are defined so that they coincide. The mortar orders are chosen to be the maximum of the orders of the polynomials of the contributing subdomains. The computation of the mortar functions is done by Eq. (28) for all the subdomains, since none of the projection operators is the identity. Once the mortar flux is computed along each mortar, it is projected back onto the subdomain faces by (33), where the upper limit on the sum is equal to the number of mortars that contribute to the subdomain face.

4.4. *Properties of the Nonconforming Approximation*

The use of the unweighted L^2 projections gives the mortar approximation two desired properties: The approximation is globally conservative and the outflow condition is satisfied. If, in addition, the subdomain faces and mortars coincide, the method remains free-stream preserving.

It is simplest to show conservation for order refinement. In that case, we need only to show that

$$\int_0^1 (\tilde{F}^{\mathrm{R}} - \tilde{F}^{\mathrm{L}})\, d = 0. \tag{34}$$

This is because the integration of Eq. (13) with $\zeta = 0$ over all subdomains leaves only integrals of the flux over the boundaries. Conservation follows if the interface flux contributions cancel. By design,

$$(\tilde{F}^{\mathrm{L}} - \Psi, v) = 0 \quad \forall v \in \mathbf{P}_{M^{\mathrm{L}}-1} \tag{35a}$$

$$(\tilde{F}^{\mathrm{R}} - \Psi, w) = 0 \quad \forall w \in \mathbf{P}_{M^{\mathrm{R}}-1}, \tag{35b}$$

where $(u, v) = \int_0^1 uv\, d\xi$. Since $1 \in \mathbf{P}_{M^{\mathrm{L}}-1} \subset \mathbf{P}_{M^{\mathrm{R}}-1} = \mathbf{P}_{J-1}$, the result follows. A similar argument can be constructed for subdomain refinement and fully nonconforming interfaces. We remark that (34) would not hold if the mortar fluxes are merely interpolated to the subdomain faces from the mortar.

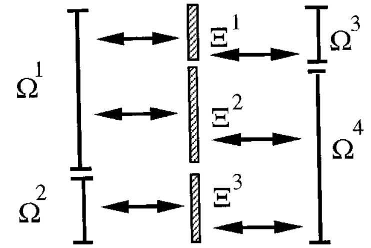

FIG. 5. Mortar topology for a fully nonconforming approximation.

The outflow condition is also satisfied by our choice of projections. Again, it is simplest to show this for order refinement. In terms of the projection operators, the outflow condition means that $\mathbf{P}^{\Xi\to\mathrm{L}}\mathbf{P}^{\mathrm{L}\to\Xi} = \mathbf{I}$. If we call $\Phi = \mathbf{P}^{\mathrm{L}\to\Xi}\mathbf{U}$ and $\mathbf{U}^* = \mathbf{P}^{\Xi\to\mathrm{L}}\Phi$ this is true if $\mathbf{U} = \mathbf{U}^*$. By construction

$$(U - \Phi, v) = 0 \quad \forall v \in \mathbf{P}_{J-1} \tag{36a}$$

and

$$(U^* - \Phi, w) = 0 \quad \forall w \in \mathbf{P}_{M^{\mathrm{L}}-1} \subset \mathbf{P}_{J-1}. \tag{36b}$$

Then $(U - \Phi, w) = 0 \quad \forall w \in \mathbf{P}_{M^{\mathrm{L}}-1}$, from which we see that $(U - U^*, w) = 0 \quad \forall w \in \mathbf{P}_{M^{\mathrm{L}}-1}$. The result follows from the fact that $U - U^* \in \mathbf{P}_{M^{\mathrm{L}}-1}$.

For the subdomain refinement case, satisfaction of the outflow condition requires that the projection operators satisfy the relations

$$\begin{aligned} \mathbf{P}^{\Xi^1\to\Omega^1}\mathbf{P}^{\Omega^1\to\Xi^1} + \mathbf{P}^{\Xi^2\to\Omega^1}\mathbf{P}^{\Omega^1\to\Xi^2} &= \mathbf{I} \\ \mathbf{P}^{\Xi^1\to\Omega^2}\mathbf{P}^{\Omega^2\to\Xi^1} &= \mathbf{I} \\ \mathbf{P}^{\Xi^2\to\Omega^3}\mathbf{P}^{\Omega^3\to\Xi^2} &= \mathbf{I} \end{aligned} \tag{37}$$

which can be shown by arguments similar to that above.

Satisfaction of the outflow condition leads us to conclude that there should be no degradation of the maximum time step that can be used in the nonconforming case compared to that required in an equivalent conforming case. Essentially, along a subdomain face inflow characteristics are specified as boundary conditions, while the outflow characteristics are unaffected. The time step, then, is determined by the approximation of the equations within the subdomains. This is exactly the situation that occurs in the conforming case. Our experience is consistent with this expectation.

Finally, the method remains free-stream preserving, provided that the subdomain interfaces coincide exactly. Again, this is most easily argued in the case of order refinement.

To see how the method can preserve a constant freestream, consider a curved interface with the topology shown in Fig. 1b. For uniform flow, with Q constant so that $F(Q) = G(Q) = 1$, the projection of the solution from the faces to the mortar by (21) is exact. The mortar flux computed pointwise along the mortar then defines a polynomial

$$\Psi = y(Y)_Y - x(Y)_Y \tag{38}$$

of degree less than or equal to J. By design, polynomials $x(Y)$ and $y(Y)$ are of low enough order to be identical along both subdomain faces and along the mortar. Thus, the projection (24) of the mortar flux, Ψ, onto the faces is also exact, so it returns the same values as would have been computed locally on the subdomain faces. On each subdomain, then, Eq. (13) becomes

$$\left.\frac{d\tilde{Q}}{dt}\right|_{i+1/2,j+1/2} + \left[\frac{\partial}{\partial X}(y_Y^N - x_Y^N) + \frac{\partial}{\partial Y}(-y_X^N + x_X^N)\right|_{i+1/2,j+1/2} = 0. \tag{39}$$

The situation is now identical to the conforming case. Since $\mathbf{x}^N \in P_{N,N}$ on a given subdomain,

$$\begin{aligned}\frac{\partial}{\partial X}\left(\frac{\partial \mathbf{x}^N}{\partial Y}\right)_{i+1/2,j+1/2} &= \sum_{k,l=0}^{N} x_{k,l}^N l_k'(\overline{X}_{i+1/2}) l_l'(\overline{Y}_{j+1/2}) \\ &= \frac{\partial}{\partial Y}\left(\frac{\partial \mathbf{x}^N}{\partial X}\right)_{i+1/2,j+1/2}\end{aligned} \tag{40}$$

so that

$$\left.\frac{d\tilde{\mathbf{Q}}}{dt}\right|_{i+1/2,j+1/2} = \mathbf{0}, \quad \begin{cases} i = 0, 1, ..., N-1, \\ j = 0, 1, ..., N-1. \end{cases} \tag{41}$$

4.5. *Mortar Algorithm*

The algorithm for the nonconforming approximation is the same as the conforming one, except for the manner in which the interface fluxes are computed. At the start of a calculation, after the grid connections, mortar positions, offsets, and scales are computed, the projection matrices are computed and stored. Then at each stage of the time integration, we use the following algorithm, based on the staggered grid method of [1]:

Algorithm I. (Non-conforming staggered grid).

1. Interpolate the Gauss/Gauss point solution values to the Gauss/Lobatto and the Lobatto/Gauss points.

2. Compute the interior point fluxes **F** and **G** from the interpolated values.

3. Compute the interface fluxes:

   (a) Project the interface solution values onto the mortars.

   (b) Compute the mortar fluxes.

   (c) Project the mortar fluxes back onto the subdomain faces.

4. Compute the boundary fluxes by applying the boundary conditions.

5. Compute spatial derivatives at the Gauss/Gauss points.

6. Update the solution at the Gauss/Gauss points.

7. Repeat Steps 1–6 until done.

The mortar approximation adds little to the cost of the calculation of the subdomain fluxes as N gets large, provided that the projection matrices are computed and stored at the beginning of a computation. The bulk of the computations occur in matrix multiplication operations, which require $O(N^2)$ multiplications, where N is the order of the matrix. Assuming that the same number of points is used in each space dimension, the work required to compute the interior fluxes for the Euler gas-dynamics equations is $O(N^2(16N + 72))$. The subdomain faces are one space dimension less and the work required to do step (3) of the algorithm above is $O(N(16N + 132))$. Thus, the work required to treat the interfaces relative to the interior work is asymptotically $O(1/N)$.

5. EXAMPLES

In this section, we use the semi-structured algorithm to compute solutions to both linear and nonlinear hyperbolic problems. We first solve a two-variable linear system using nonconforming topologies and compare the convergence to the convergence using alternative conforming grids. We then present an example where the solution is localized, and show that the computational cost for the same error can be reduced significantly by using the nonconforming interface treatment.

We also solve three problems using the Euler gas-dynamics equations. The first problem is that of subsonic flow from a point source, for which there is an exact solution. The convergence rates of conforming and alternative nonconforming grids are compared. We then show that exponential convergence is obtained when solving the problem on a complex, multiply connected subdomain topology. The second problem is a steady subsonic flow over a circular bump. We show that exponential convergence of the entropy is obtained for both conforming and nonconforming approximations. The nonconforming approximation, however, takes only half the computer time for the same accuracy. Finally, as an example of a transonic flow problem we solve flow in an axisymmetric converging–diverging nozzle. The computed results for that problem are compared to experimental data.

5.1. *Linear Model Problem*

We begin by considering steady solutions to the system

$$\mathbf{Q}_t + \mathbf{F}_x + \mathbf{G}_y = \mathbf{S}(x, y), \tag{42a}$$

where

$$\mathbf{Q} = \begin{bmatrix} u \\ v \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} 3v - u \\ 3u - v \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} 2u + 4v \\ 4u + 2v \end{bmatrix}. \tag{42b}$$

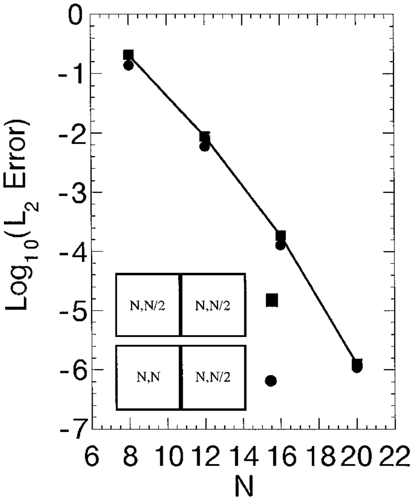

FIG. 6. Comparison of conforming and nonconforming errors for order refinement.

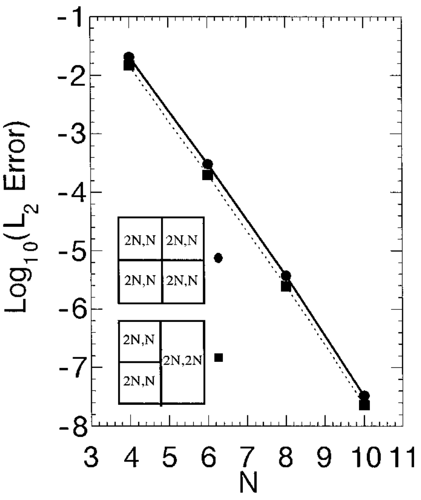

FIG. 7. Comparison of L_2 errors for conforming and nonconforming grids.

The source term is chosen so that the exact steady solution is

$$u = e^{\sin(2\pi x)\sin(2\pi y)}$$

$$v = e^{xy}.$$

Exponential convergence is observed for all three nonconforming grid topologies shown in Fig. 1b. First, we consider order refinement on an equal subdivision of the rectangle [0, 2] × [0, 1]. Figure 6 compares the L$_2$ errors of a conforming grid with two subdomains with those of a nonconforming approximation. In both cases we observe exponential convergence. For order refinement alone, we would expect the error to be dominated by the lowest order approximation, and this is the case.

Exponential convergence is also observed when a subdomain is subdivided. Figure 7 compares the error of an equal four-subdomain conforming decomposition of the unit square with a subdivided approximation. As before, we observe that the error is dominated by the Nth-order polynomial approximation.

Finally, we consider a fully nonconforming subdivision of the unit square, shown in Fig. 8. In this case, we choose a subdivision that is between two conforming approximations in its resolution. The first conforming subdivision divides the square into four equal subdomains. The second subdivides the vertical direction into thirds and the horizontal in half. Again, we observe exponential decay of the error, and that error lies between the errors of the two conforming approximations.

One of the main reasons to use a nonconforming grid is to compute efficiently solutions where local refinement of the grid is needed. As an example, we compute the solution of (42) on the unit square where the source terms are chosen so that the steady solution is

$$u = e^{-15((x-1)+(y-1))}$$

$$v = e^{-50(x^2+y^2)}.$$

In this problem, the solution variations are concentrated in the upper left and lower right corners as shown in Fig. 9a. We solve the problem on two grids, also shown in Fig. 9. The nonconforming grid, which has increased resolution only where needed, has 44% of the number of grid points of the conforming grid when $N = 10$.

Figure 10 compares the convergence of the error for the

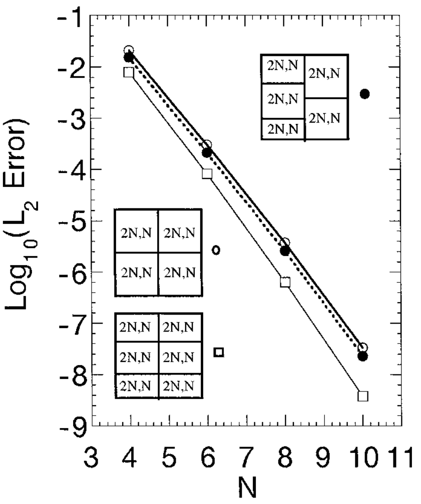

FIG. 8. Comparison of conforming and fully nonconforming errors.

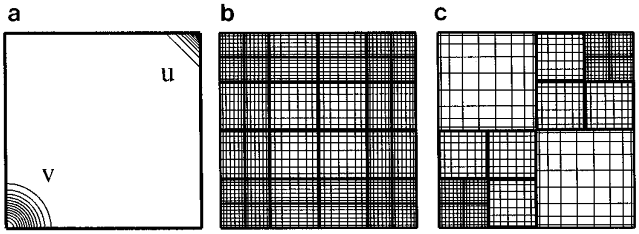

FIG. 9. Solution and grids for (38) where local refinement is needed: (a) solutions; (b) conforming grid; (c) nonconforming grid.

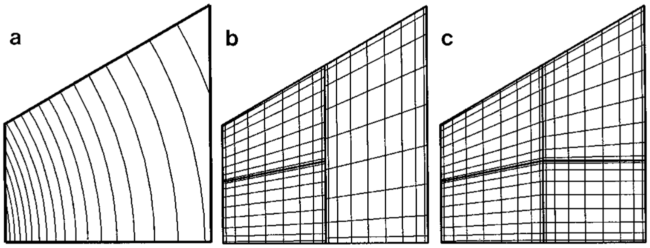

FIG. 11. Diffuser solution and nonconforming and conforming grids: (a) Mach contours; (b) nonconforming grid; (c) conforming grid.

two grids shown in Fig. 9. We find that the errors for both grids are the same to one digit. However, at $N = 10$, the computational cost of the nonconforming grid computation is 46% of the cost of the conforming grid computation. Since, at best, we would expect the nonconforming grid to take 44% of the time of the conforming one, we see that the overhead due to the mortar projections is only about 5%.

5.2. *Euler Gas-Dynamics Equations*

5.2.1. *Point Source Flow.* We now consider the solution of the flow of a steady, irrotational gas exiting from a point, which can be solved exactly by a hodograph transformation [13]. The streamlines are radial and the level curves of the Mach number, pressure, and density are circles centered on the source. We will compute this flow in two geometries. The first geometry represents flow in an expanding duct, where two streamlines are chosen as walls of the duct. The second geometry is that of a rectangular region with three cut-out circles.

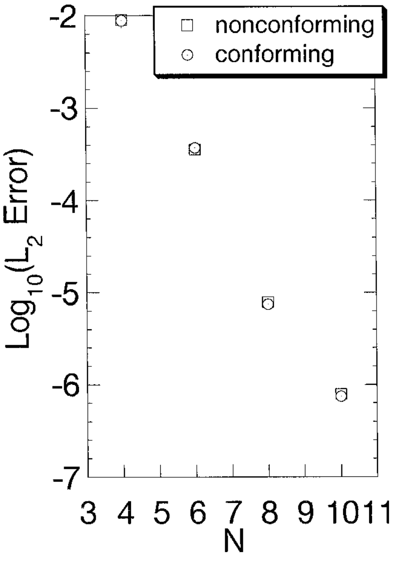

FIG. 10. Convergence of linear model problem (38) on the two grids shown in Fig. 9.

The first geometry represents steady flow in an expanding two-dimensional duct with straight walls (Fig. 11). The lower wall was chosen to be the line $y = 0$ and the upper wall the line $y = x \tan(\pi/6)$. The exact solution chosen was the one that takes on the Mach number $M = 0.6$ at the lower left corner. We compute this flow on the two grids shown in Fig. 11. An examination of the error using a single domain approximation indicates that most of the contribution of the error occurs near the lower left corner. Thus, we set up the nonconforming grid as shown in Fig. 11b. For comparison, we also compute the solution on the conforming grid shown in Fig. 11c. The errors for the two grids are plotted in Fig. 12 and are the same to one digit.

Exponential convergence can be obtained on complex geometries, too, if the solution is smooth. In Fig. 13, we show the Mach contours and grid for the solution of the point source flow in a rectangular domain with three cut-out holes. The point source was placed at the center of the bottom circle. This grid has fully nonconforming subdo-

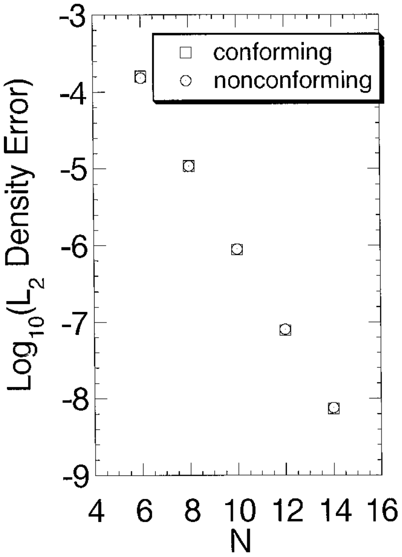

FIG. 12. Convergence of the density error for the two grids of Fig. 11.

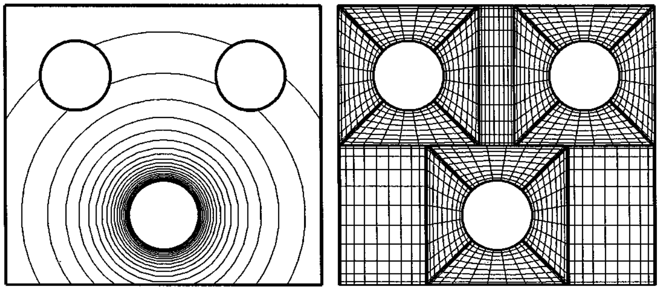

FIG. 13. Mach contours and multiply connected grid for the point source flow.

main interfaces. The exact solution was used to compute the boundary conditions on all the boundaries. Figure 14 shows that the error converges exponentially with N.

5.2.2. *Subsonic Flow over a Circular Bump.* Our next example is the solution of a Mach 0.3 subsonic flow over a circular bump. This flow was computed on two grid topologies, shown in Fig. 15. A wall boundary condition was specified along the bottom. At the left, right, and top boundaries, the uniform free-stream condition was specified as the external input to the boundary Riemann problems. Initially, the free-stream solution was specified everywhere, and then the boundary conditions were imposed. A comparison of the pressure computed on the conforming and nonconforming grids is also shown on Fig. 15.

This problem does not have an exact solution. However, since the free-stream is irrotational and homentropic, the entropy must remain constant everywhere. That this is not true computationally is due to spatial approximation errors. Figure 16 shows the exponential convergence of the entropy error for the two grids shown in Fig. 15.

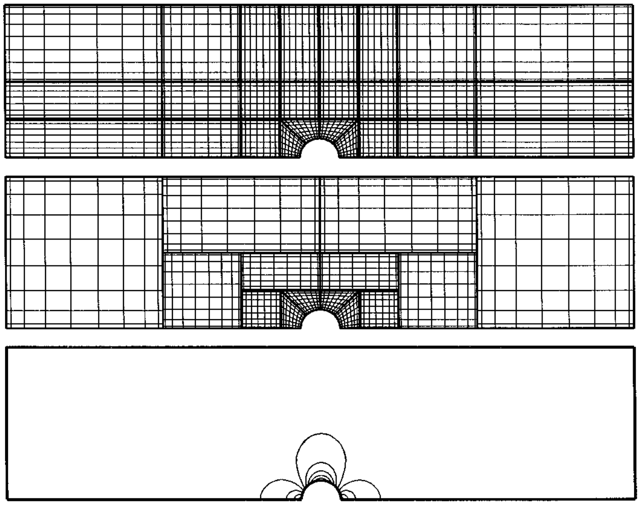

FIG. 15. Pressure contours and grids for $M = 0.3$ flow over a circular bump. The pressure contours for the conforming grid are plotted with dashed lines; the nonconforming solution is plotted with solid lines.

For $N = 10$, the nonconforming grid in Fig. 15 has 50% of the number of degrees of freedom of the conforming grid. We find that the CPU time for the nonconforming approximation to get to steady state is 44% of the time required by the conforming one, so the work required by the mortar projections is negligible.

5.2.3. *Transonic Flow in a Converging–Diverging Nozzle.* Our final example is that of a transonic flow in an

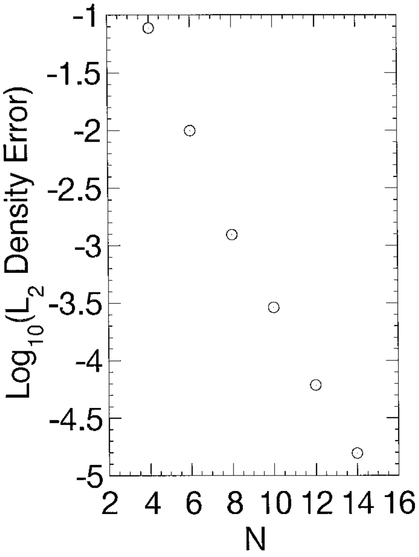

FIG. 14. Convergence of the density for the grid in Fig. 13.

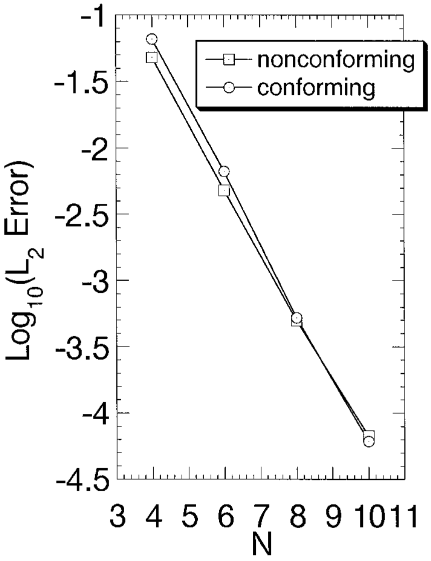

FIG. 16. Convergence of the entropy for the two grids shown in Fig. 15.

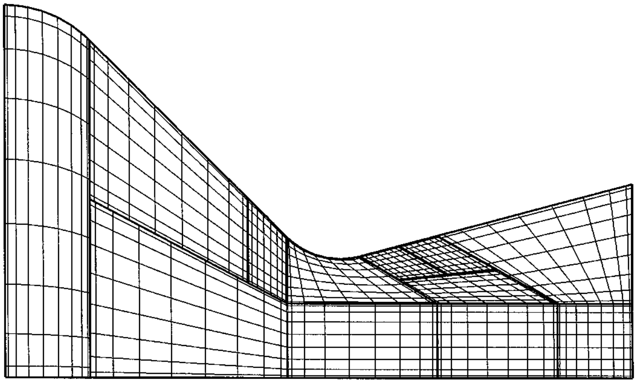

FIG. 17. Nozzle shape and grid for transonic flow computation.

axisymmetric converging diverging nozzle. We use the nozzle of Cuffel *et al.* [14]. The nozzle has a converging section with half angle of 45° and a diverging section of 15°. The experimental tests were done in air with a stagnation temperature of 540 R and a stagnation pressure of 70 psia. The nozzle geometry and grid are shown in Fig. 17. We have increased the resolution in the neighborhood of nozzle wall curvature singularities by subdivision of a conforming grid (see [1]). Boundary conditions and scaling were treated as in Ref. [1].

Results computed for the nozzle are shown in Figs. 18–20. Contours of the Mach number are compared to the experimentally determined positions in Fig. 18. Wall values of the pressure and Mach number are shown in Figs. 19 and 20 for different subdomain resolutions.

6. SUMMARY

It was difficult to do local refinement of the grid with the original staggered-grid multidomain approximation [1]. That method required subdomains to intersect either along a full side or at a single point. In addition, along adjoining faces the approximation's polynomial orders had to be the same. This made it impossible to subdivide a subdomain or to increase locally the polynomial order as necessary to resolve a local feature in the solution.

In this paper, we have described a semi-structured method that uses the staggered grid scheme interior to subdomains. It relaxes the restriction that the fluxes be continuous at subdomain interfaces and allows for nonconforming interfaces. This makes it possible to use a general quadrilateral tiling of a computational domain. Subdomains can be subdivided as necessary. Within each subdomain the polynomial order can be set independently of its neighbors. More generally, subdomains need intersect only partially along a side.

In the semi-structured version, the interfaces are treated by a mortar method. The solutions along subdomain faces are first projected onto a one-dimensional construct called

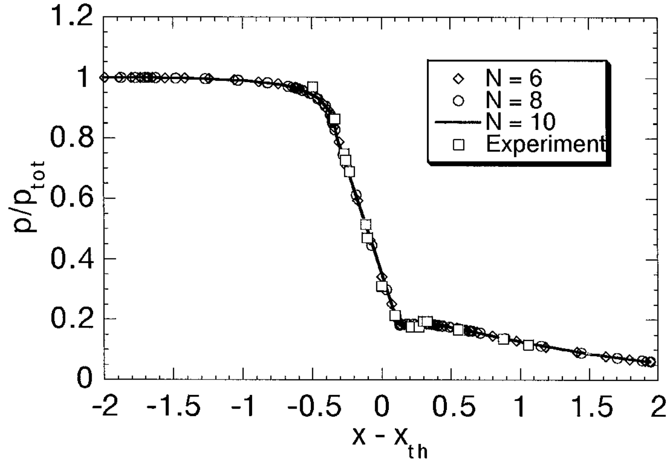

FIG. 19. Wall pressure of the converging–diverging nozzle.

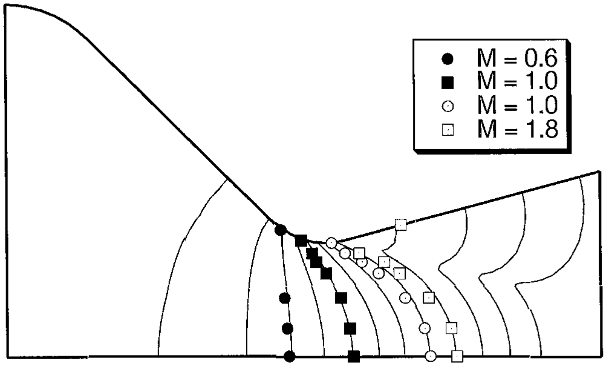

FIG. 18. Computed and measured (symbols) Mach contours in the nozzle.

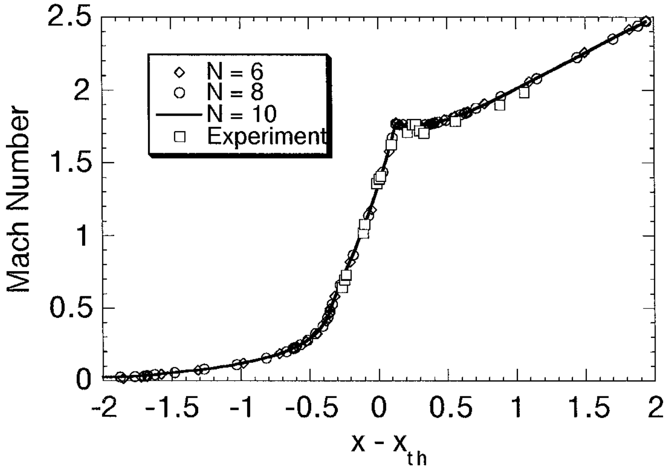

FIG. 20. Wall Mach number for the converging–diverging nozzle.

a mortar. Along the mortar a unique flux is computed as if the approximation is conforming. That flux is then projected back onto the subdomain faces to be used to update the solutions within the subdomains. Asymptotically, the work involved in this process is small when compared to the work required to update the solutions in the interiors of the subdomains. The mortar approximation is designed so that the method is globally conservative.

The method has been applied to both linear and nonlinear smooth problems for which exact solutions are known. In all cases, spectral convergence was observed. The advantage of being able to refine locally the grid was also seen in the reduced cost of the nonconforming approximations.

ACKNOWLEDGMENTS

The author thanks Dr. C. Mavriplis for many helpful discussions on the mortar element method. This research was supported in part by the U.S. Department of Energy Contract DE-FC05-85ER250000, by the National Science Foundation Grant DMS-9404322 and by NAS1-19480, while the first author was in residence at the Institute for Computer Applications in Science and Engineering (ICASE), NASA Langley Research Center, Hampton, VA 23681-0001.

REFERENCES

1. D. A. Kopriva and J. H. Kolias, *J. Comput. Phys.* **125**, 244 (1996).
2. F. H. Harlow and A. A. Amsden, *J. Comput. Phys.* **8,** 197 (1971).
3. C. Canuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang, *Spectral Methods in Fluid Mechanics* (Springer-Verlag, New York, 1987).
4. D. A. Kopriva, *J. Comput. Phys.* **96,** 428 (1991).
5. C. A. Mavriplis, Ph.D. dissertation, MIT, Feb. 1989 (unpublished).
6. M. M. Rai, AIAA paper 84-0164, January 1984 (unpublished).
7. J. F. Thompson, Z. U. Warsi, and C. W. Mastin, *Numerical Grid Generation* (North Holland, New York, 1985).
8. D. Funaro, *Numer. Math.* **52,** 329 (1988).
9. Y. Maday, C. Mavriplis, and A. T. Patera, "Non-Conforming Mortar Element Methods: Application to Spectral Discretizations" in *Domain Decomposition Methods* (SIAM, Philadelphia, 1989), p. 392.
10. G. Anagnostou, Y. Maday, C. Mavriplis, and A. T. Patera, "On the Mortar Element Method: Generalization and Implementation" in *Third International Conference on Domain Decomposition Methods for Partial Differential Equations,* edited by R. Glowinski (SIAM, Philadelphia, 1990), p. 157.
11. C. Bernardi, Y. Maday, and A. T. Patera, "A New Nonconforming Approach to domain decomposition: The Mortar Element Method", in *Nonlinear Partial Differential Equations and their Applications*, edited by H. Brezis and J. L. Lions (Pitman/Wiley, London/New York, 1992).
12. P. L. Roe, *J. Comput. Phys.* **43,** 357 (1981).
13. R. Courant and K. O. Fridrichs, *Supersonic Flow and Shock Waves* (Springer-Verlag, New York, 1976).
14. R. F. Cuffel, L. F. Back, and P. F. Massier, *AIAA J.* **7,** 1364 (1969).