Representation of Velocity Gradient Effects in a Gaussian Puff Model

R. I. SYKES AND D. S. HENN

A.R.A.P. Group, Titan Research and Technology Division, Titan Corporation, Princeton, New Jersey

(Manuscript received 7 March 1995, in final form 19 June 1995)

ABSTRACT

The Gaussian puff model framework is extended to provide a description of velocity shear distortion effects. An efficient splitting–merging algorithm is presented so that a maximum puff size can be specified for a calculation. This localizes the Gaussian puffs so that they represent only a limited region of the flow and the accuracy of the representation is therefore controlled. The model is shown to perform well on the deformational flow of Smolarkiewicz, providing an accurate calculation of the highly distorted solution. The extended puff methodology allows practical applications of an efficient Lagrangian dispersion technique in complex flow fields.

1. Introduction

Gaussian puff models are a well-known technique for describing the dispersion of a contaminant species in the atmosphere. Puff models represent one of two types of Lagrangian techniques, the other being the particle method, which represents the dispersing field using a collection of elements that move with the mean flow. Lagrangian techniques avoid the numerical diffusion problems associated with Eulerian grid methods and give an efficient representation of the wide range of scales in dispersion from a localized source. In the particle methodology, a large number of discrete particles are dispersed with either a random walk (e.g., Reid 1979; Diehl et al. 1982; Thomson 1987) or a deterministic gradient diffusion (e.g., Lange 1978), and local concentrations are calculated from the local particle-number density. In contrast, puff models maintain a description of the spatial spread of the Lagrangian element in addition to its centroid location so that a continuous description of the concentration field is obtained using a small number of elements (Bass 1980).

The particle technique is generally considered to be the more fundamental approach of the two Lagrangian methods, with the ability to represent inhomogeneous flow effects such as wind shear and turbulence variations. Each particle samples a point location within the flow, so the inhomogeneity can be explicitly sampled by a collection of particles. This generality is obtained at the expense of computational time since a large number of particles must be released if an accurate concentration calculation is required. Scalar concentrations are determined by counting the number of particles within a given volume, and this becomes increasingly inaccurate as the particles are dispersed and the number density decreases. In contrast, Gaussian puff methods provide a continuous description of the concentration field with a relatively small number of puffs and can represent arbitrarily small concentrations using the assumed spatial distribution. However, the simple representation as a collection of spreading Gaussian puffs is less able to represent inhomogeneous flow effects, since the puff samples only the flow field at the centroid location. Velocity field variations on the scale of the Gaussian distribution are neglected in the conventional puff method.

A compromise methodology has been suggested by Hurley (1994), wherein the vertical distribution is represented by the particle technique, but each particle is associated with a lateral Gaussian spread. This assumption significantly reduces the computational requirements since the particle description is restricted to one dimension. The technique exploits the fact that, in many cases, the inhomogeneity in the atmosphere is confined to the vertical direction and provides the generalized capability of the particles for treating vertical dispersion and shear. The method is specialized, however, and still requires a relatively large number of particles to define accurate local concentrations.

In this paper, we describe a generalization of the Gaussian puff method that can accurately represent arbitrary three-dimensional shear effects. The method is based on a complete description of the shear distortions in conjunction with an efficient splitting–merging scheme to maintain appropriate overlap while restricting the growth of individual puffs. The size is limited by splitting puffs that grow larger than the specified limit and defines a spatial resolution for the calculation. The Lagrangian resolution is chosen in the same way as an Eulerian resolution so that the velocity field vari-

Corresponding author address: Dr. R. Ian Sykes, ARAP Group, Titan Research and Technology Division, 50 Washington Rd., P.O. Box 2229, Princeton, NJ 08543-2229.

ations across a puff are small and a localized representation of prescribed accuracy is obtained. The puffs can be considered Lagrangian elements with Gaussian shape, and the objective of the splitting–merging scheme is to maintain a smoothly overlapping distribution as the puffs are advected and distorted by the flow field. The method is described in the next section, and model results for an idealized two-dimensional test problem are presented in section 3.

2. Numerical method

Conventional puff schemes describe the two physical processes of advection and diffusion very simply. The centroid of each puff is transported by the mean velocity field, and the Gaussian spread is increased by the diffusivity. Many models specify different spread rates for the vertical and horizontal direction, but each puff is essentially a spreading Gaussian ellipsoid moving with the mean wind. The two main difficulties with the conventional method are the inability to represent local shear distortions and the unbounded increase in puff size, which eventually makes a local approximation invalid. The first problem is addressed by consideration of the full Gaussian moment description, while the second problem requires a redistribution of the puff mass among smaller puffs so that the growth of an individual puff is limited.

The conventional representation of the Gaussian shape using a vertical and a horizontal spread is clearly restrictive. The moments are dependent upon the particular choice of coordinate system, and a complete description must consider the full moment tensor

$$\sigma_{ij} = \frac{1}{Q}\int_V x_i' x_j' c\, dV, \tag{1}$$

where Q is the total mass of the puff, c is the local concentration, and x_i' is the coordinate vector relative to the puff centroid. The puff mass and centroid are defined as

$$Q = \int_V c\, dV \tag{2}$$

$$\bar{x}_i = \frac{1}{Q}\int_V x_i c\, dV. \tag{3}$$

The local puff concentration c is represented by the generalized Gaussian distribution

$$c(x) = \frac{Q}{(2\pi)^{3/2}[\det(\sigma)]^{1/2}} \exp\left(-\frac{1}{2}\sigma_{ij}^{-1} x_i' x_j'\right). \tag{4}$$

The general Gaussian distribution therefore requires six moments since the second-rank tensor σ_{ij} is symmetric.

For our present purposes we shall consider only the advection equation

$$\frac{\partial c}{\partial t} + \frac{\partial}{\partial x_i}(u_i c) = 0; \tag{5}$$

that is, diffusion is neglected here. The advection equation is linear and can therefore be considered to apply to each Lagrangian puff separately. If we consider the velocity field to be locally described by a linear variation in space, then the centroid and Gaussian moments evolve according to the relations

$$\frac{d\bar{x}_i}{dt} = u_i(\bar{x}_i, t) \tag{6}$$

$$\frac{d\sigma_{ij}}{dt} = \sigma_{ik}\frac{\partial u_j}{\partial x_k} + \sigma_{jk}\frac{\partial u_i}{\partial x_k}. \tag{7}$$

The linearization of the velocity field is valid for sufficiently small regions, and this can be ensured provided the puff moments remain small in comparison with the scale of the velocity field variations.

The effective volume of the generalized Gaussian described by the six independent moments σ_{ij} is proportional to the square root of the determinant $D = \det(\sigma)$. The volume of a fluid parcel is conserved in an incompressible flow, and conservation of the determinant is also implied by (7) provided the velocity field $\mathbf{u}$ is solenoidal. It is important to conserve the determinant numerically, and we now describe a simple scheme with this property.

The evolution of the Gaussian moments at time t over one time step Δt, that is, from $\sigma_{ij}(t)$ to $\sigma_{ij}(t+\Delta t)$, is computed as a sequence of intermediate steps with $\hat{\sigma}_{ij}^{(m)}$ denoting the result of step m. The initial step, with $m = 0$, accounts for the diagonal velocity gradients using the relation

$$\hat{\sigma}_{\alpha\beta}^{(0)} = \sigma_{\alpha\beta}(t)\exp\left[\Delta t\left(\frac{\partial u_\alpha}{\partial x_\alpha} + \frac{\partial u_\beta}{\partial x_\beta}\right)\right], \tag{8}$$

where the Greek subscripts imply no summation. The six off-diagonal velocity gradients are then applied sequentially using the scheme

$$\hat{\sigma}_{\alpha\alpha}^{(m)} = \hat{\sigma}_{\alpha\alpha}^{(m-1)} + S_{\alpha\beta}[2\hat{\sigma}_{\alpha\beta}^{(m-1)} + S_{\alpha\beta}\hat{\sigma}_{\beta\beta}^{(m-1)}] \tag{9a}$$

$$\hat{\sigma}_{\alpha\beta}^{(m)} = \hat{\sigma}_{\alpha\beta}^{(m-1)} + S_{\alpha\beta}\hat{\sigma}_{\beta\beta}^{(m-1)} \tag{9b}$$

$$\hat{\sigma}_{\alpha\gamma}^{(m)} = \hat{\sigma}_{\alpha\gamma}^{(m-1)} + S_{\alpha\beta}\hat{\sigma}_{\beta\gamma}^{(m-1)}, \tag{9c}$$

where $S_{\alpha\beta} = \Delta t\,\partial u_\alpha/\partial x_\beta$, $\alpha \neq \beta$, and $\gamma \neq \alpha$ or β. The subscripts corresponding to step m, where $1 \leqslant m \leqslant 6$, can be determined in the following (arbitrary) way as

$$\alpha = \text{int}\left(\frac{m+1}{2}\right) \tag{10a}$$

$$\beta = \text{mod}_3\left[\text{int}\left(\frac{m+4}{2}\right)\right] \tag{10b}$$

$$\gamma = \text{mod}_3(7-m), \tag{10c}$$

where $\text{mod}_3(n) \equiv n - 3\,\text{int}[(n-1)/3]$. It should be noted that the symmetry of the moment tensor is used in applying (9) at each intermediate step; the remaining moments are unaffected by the corresponding velocity

gradient component. The advanced time level $\sigma_{ij}(t + \Delta t)$ is then finally obtained as $\hat{\sigma}_{ij}^{(6)}$.

The scheme given by (8) and (9) preserves the value of the determinant exactly, as shown in the appendix. The relations (9) are strictly only first-order accurate in time, and the sequential application of the velocity gradients also reduces the overall accuracy. However, the conservation of the determinant is a more important property than accuracy for long-term stability of the scheme. Conservation of the determinant is vital not only because it eliminates numerical diffusion (which would increase the determinant) but also because it also avoids the nonphysical situation of decreasing the determinant, which can result in unrealizable moments. This situation could result from the application of strong persistent wind shear even in the presence of physical diffusion.

The puff centroid is advanced using the second-order Adams–Bashforth method

$$\bar{x}_i(t + \Delta t) = \bar{x}_i(t) + \Delta t\left[\frac{3}{2}\, u_i(t) - \frac{1}{2}\, u_i(t - \Delta t)\right], \tag{11}$$

where $u_i(t)$ is the velocity field at $\bar{x}_i(t)$, that is, the velocity at the current time step and centroid, and $u_i(t - \Delta t)$ is the velocity at the previous time step and centroid.

The moment evolution equation (7) can produce elongation of a puff along the direction of the wind shear, and at some stage the linear representation of the velocity field over the extent of the Gaussian puff becomes invalid. A similar effect is caused by diffusive growth of the puff, although we have not considered diffusion in our simplified equations. A grid-based method for splitting puffs was presented by Sykes and Henn (1992), where the moment method of Egan and Mahoney (1972) was extended to include some shear effects. In this scheme, grid cells can be thought of as each containing a Lagrangian puff. As the puffs move and spread into neighboring grid cells in a time step, the masses are redistributed so as to maintain a single puff within each cell. We wish to avoid any numerical grid in the current method, however, so the redistribution cannot be based on the rectangular grid cells.

The three diagonal moments $\sigma_{\alpha\alpha}$ give the effective length of the puff along each coordinate direction, and we choose to split a puff in the x direction, for example, when $\sigma_{11} > \Delta x^2$. Here Δx represents the limit of the x spread and should be chosen so that the linearization of the velocity field is valid for a Gaussian with smaller spread. Similar splitting criteria apply for the other two coordinate directions. We wish to represent the original Gaussian puff with several smaller, overlapping puffs that conserve all the puff moments and change only local concentration values by a small amount. For a split in the x direction, the original puff is replaced by two smaller puffs as follows. The new centroid locations are displaced by a fraction of the puff spread in the x direction and by a distance proportional to the off-diagonal moment in the other two coordinate directions. Thus,

$$\bar{x}_1^{(\alpha)} = \bar{x}_1 \pm r\sigma_{11}^{1/2} \tag{12a}$$

$$\bar{x}_2^{(\alpha)} = \bar{x}_2 \pm r\frac{\sigma_{12}}{\sigma_{11}^{1/2}} \tag{12b}$$

$$\bar{x}_3^{(\alpha)} = \bar{x}_3 \pm r\frac{\sigma_{13}}{\sigma_{11}^{1/2}}, \tag{12c}$$

where $\alpha = \{1, 2\}$. The diagonal moments are all reduced using the following relations

$$\sigma_{11}^{(\alpha)} = \sigma_{11}(1 - r^2) \tag{13a}$$

$$\sigma_{22}^{(\alpha)} = \sigma_{22}\left(1 - \frac{r^2\sigma_{12}^2}{\sigma_{11}\sigma_{22}}\right) \tag{13b}$$

$$\sigma_{33}^{(\alpha)} = \sigma_{33}\left(1 - \frac{r^2\sigma_{13}^2}{\sigma_{11}\sigma_{33}}\right), \tag{13c}$$

and the new off-diagonal moments are

$$\sigma_{12}^{(\alpha)} = \sigma_{12}(1 - r^2) \tag{14a}$$

$$\sigma_{13}^{(\alpha)} = \sigma_{13}(1 - r^2) \tag{14b}$$

$$\sigma_{23}^{(\alpha)} = \sigma_{23} - \frac{r^2\sigma_{12}\sigma_{13}}{\sigma_{11}}. \tag{14c}$$

Conservation of all puff moments is ensured by this procedure, and the moments of the new puffs are all smaller and are also realizable. The realizability constraints involve the Schwartz inequality between the diagonal and off-diagonal moments, for example, $\sigma_{12}^2 < \sigma_{11}\sigma_{22}$ and the strictly positive value for the determinant. These properties can be verified directly from the above relations.

Other splitting schemes may also be valid. For instance, splitting along the eigenvector associated with the maximum eigenvalue of the second-moment tensor would give a coordinate-independent scheme. However, the computational effort of such a scheme is probably unwarranted in light of the satisfactory performance of this simpler method.

The reduction in puff size is controlled by the parameter r, but a larger reduction in puff size reduces the amount of overlap between the new puffs and gives a poorer representation of the original Gaussian shape. The splitting scheme described by (12)–(14) creates a pair of Gaussians with a dimensionless separation

$$\delta_s = \frac{\Delta}{\sigma} = \frac{2r}{(1 - r^2)^{1/2}},$$

where Δ is the distance between the centroids and $\sigma = \sigma_{11}^{1/2}$ is the spread in the x direction. In general, the difference in concentration distribution resulting from the splitting operation is difficult to specify for an arbitrary Gaussian, but the simple spherical puff with diagonal moments σ^2 provides an idealized case. Figure

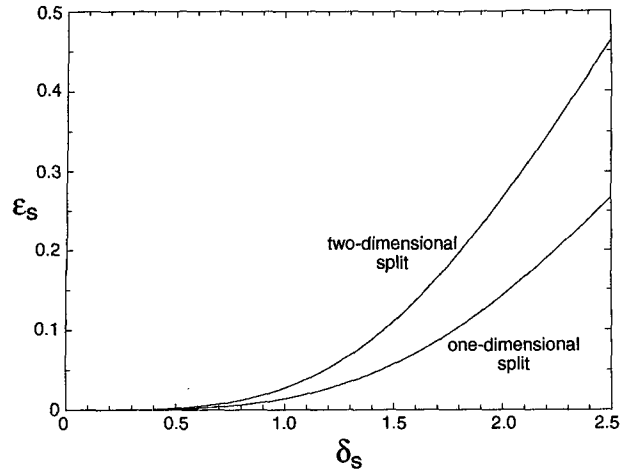

FIG. 1. Maximum dimensionless error from splitting a single Gaussian puff as a function of separation distance. The error is relative to the maximum value in the original Gaussian, and the separation distance of the puffs after splitting is relative to the original Gaussian spread. Results are shown for both one-dimensional and two-dimensional splits.

1 shows the maximum difference between the original Gaussian and the sum of the two smaller Gaussians after splitting in one dimension, that is, the x direction only, and also for a two-dimensional split in both x and y, which creates four puffs. The difference is given relative to the maximum concentration value in the original Gaussian and is plotted as a function of the dimensionless separation. A maximum local concentration change of 10% requires a dimensionless separation of less than 1.8 for a one-dimensional split, but must be less than 1.45 for two dimensions. In practice, there will usually be additional overlap from other puffs, and we have found that $r = 0.65$, that is, $\delta_s = 1.71$, gives the optimum size reduction with acceptable overlap. In diffusive applications, the puff size increases after splitting and reduces the effective separation; reasonable results are obtained with $r = 0.75$, that is, $\delta_s = 2.25$, in such cases.

Unfortunately, the splitting process can rapidly lead to excessive numbers of puffs unless some form of merging is employed to reduce the numbers. In general, splitting the puff distribution will create many overlapping puffs that can be merged together as a single Gaussian. The merging rules are actually much simpler than the splitting rules since the only requirement is moment conservation. Thus, if superscripts 1 and 2 represent two puffs deemed sufficiently close to each other to merge, then they can be combined as follows:

$$Q = Q^{(1)} + Q^{(2)} \tag{15a}$$

$$Q\bar{x}_i = Q^{(1)}\bar{x}_i^{(1)} + Q^{(2)}\bar{x}_i^{(2)} \tag{15b}$$

$$Q\sigma_{ij} = Q^{(1)}[\sigma_{ij}^{(1)} + \Delta\bar{x}_i^{(1)}\Delta\bar{x}_j^{(1)}] + Q^{(2)}[\sigma_{ij}^{(2)} + \Delta\bar{x}_i^{(2)}\Delta\bar{x}_j^{(2)}], \tag{15c}$$

where $\Delta\bar{x}_i^{(\alpha)} = \bar{x}_i^{(\alpha)} - \bar{x}_i$.

The difficulty associated with merging lies in deciding which puff pairs are eligible. In general, if we have N puffs, there are N^2 possible pair combinations and it rapidly becomes impractical to consider all such pairs when N is large. Some techniques have been suggested for ordering randomly located, multidimensional data with the objective of determining nearest neighbors (e.g., Boris 1986), but we have chosen to use an adaptive multigrid location scheme.

The basic concept of the grid location scheme is to assign each puff to a particular grid box; then a search for near neighbors can be carried out by searching over the neighboring grid cells. There are two technical difficulties that arise in the application of this concept. First, the grid-cell size should really be determined by the size of the puff itself, so that a puff overlaps only a finite number of neighboring grid cells. Second, we must account for an arbitrary number of puffs within each grid cell. Thus, we cannot simply allocate storage for a fixed grid with a fixed number of puffs in each cell.

The first problem is solved by means of an adaptive grid in the horizontal plane. In atmospheric cases, the vertical resolution is restricted by the stable stratification, so a fixed grid size, Δz, is used. However, the horizontal scale varies widely so an initial coarse grid is defined with size Δ_0. Each cell of this coarse grid can be individually refined to create four additional cells with size $\Delta_1 = \Delta_0/2$. The new cells are simply added to the end of the list of cells, and a pointer from the Δ_0 cell contains the location of the first of the four refined cells. This procedure can be continued as far as computer storage allows. This successive refinement procedure has the advantage that all coarser levels of the grid are available in addition to the most refined level. Each cell of the grid stores two integer variables, the pointer for the subsequent refinement (if it exists) and the number identifier of the puff located in that cell. The choice of refinement level for locating a puff on the adaptive grid is determined by the largest of the horizontal moments, that is, the grid size $\Delta_n = \Delta_0/2^n$ such that

$$\Delta_n < 4\sigma_{\max} \leqslant \Delta_{n-1},$$

where $\sigma_{\max}^2 = \max(\sigma_{11}, \sigma_{22})$. A schematic illustration of the puff allocation is shown in Fig. 2, where a single puff with index number α is located in cell 12 (denoted by C12). The puff requires two levels of grid refinement, and the grid system generated by this puff is also shown. The grid cells are indicated by the C number, and the two numbers stored for each cell are given in parentheses below. The first number is the first grid cell of a refined block of four cells and is zero for no refinement. The second number is the index of the puff contained in this cell, and a zero indicates no puff in the cell. When both numbers are zero for a cell, we have omitted the number pair in some cases. Thus, all cells except C12 contain no puffs and show the second number as zero. Cell 4 shows the refinement to the

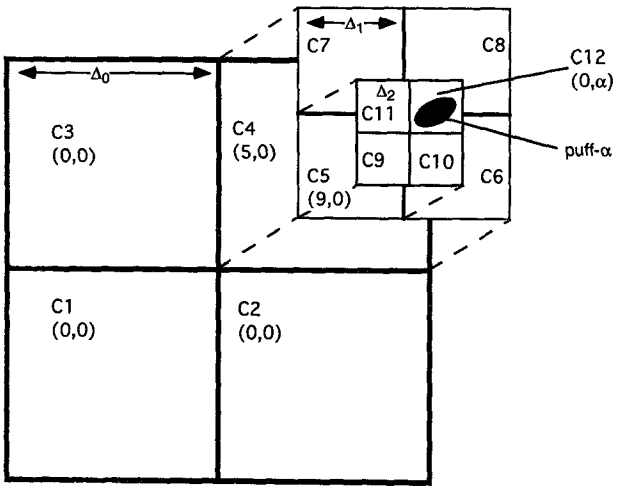

FIG. 2. Schematic illustration of the adaptive multigrid for locating puffs. Grid-cell numbers are represented as C*n*, and two levels of refinement are shown. See text for a description of the cell storage rules.

first level and points to cell 5, which is further refined and points to cell 9. Clearly, a large number of puffs will produce a much more complicated grid and index structure, but this simple adaptive refinement provides an efficient location scheme for locating arbitrary collections of puffs.

In locating the puffs on the grid, the index number of the first puff placed in a particular cell is stored in the grid-cell storage location, but if other puffs share the same cell, then their indices must be saved elsewhere. Unlimited numbers of puffs can be accommodated by using a linked list; each puff carries a pointer to designate the next puff in the cell. Thus, if puff number j is found to share a grid cell with a previous occupant, say puff number i, then the list pointer in puff i is set to point to puff j. In practice, the entire list for the cell must be scanned and the new puff added to the end of the list. However, since the cells are adapted to the puff size, it is highly unlikely that a single cell will contain more than a few puffs.

The use of the two methodologies illustrated above provides a very efficient method for sorting a large number of puffs with arbitrary locations and moments, giving a compact list of puff pairs to test for possible merging. We note that a search over the list in each grid cell finds most of the candidate pairs, although puffs allocated to different cells but lying very close to each other across the boundary will be missed. In practical terms, it is more efficient to miss some merges rather than scan multiple cells. The actual merging of a candidate pair is based on the overlap integral of the two Gaussian functions, which can be written in the form

$$I = \frac{1}{(2\pi)^{3/2} D_{12}} \exp(-A_{12}),$$

where D_{12} is a combination of the two determinants and the exponential argument depends on both puff moments and their separation. This integral is computed as part of the concentration fluctuation variance calculation in atmospheric dispersion applications of the model (Sykes et al. 1993). For two identical, spherical Gaussians with separation distance Δ and diagonal moments σ^2, the argument can be written very simply as $A_{12} = \Delta^2/4\sigma^2$. A pair is merged if $A_{12} < \delta_m^2/4$, so that δ_m is a measure of the centroid separation relative to the size of the two puffs. We note that the separation is not a direct indication that the two puffs are nearly coincident, since they could be of different size or shape. However, the multigrid sorting scheme ensures that the pair have similar overall size and it is unlikely that two such puffs will arrive at the same spatial position with very different shear histories. A more reliable merge criterion can be developed to account for the shape differences as well as the centroid separation, but we have not found this to be necessary.

As discussed in connection with the splitting process, the merging criterion determines the extent of puff overlap, since a small value for δ_m will maintain more overlap before merging a pair of puffs. It is important to ensure consistency between the split and merge criteria, however, in the sense that a newly created split pair should not satisfy the merge criterion. In practice, we maintain a distinct gap between the two criteria, and a reasonable value for δ_m is found to be 1.41 in conjunction with the split criterion, $r = 0.65$; that is, $\delta_s = 1.71$.

3. Idealized test results

The simplest idealized shear flow is the homogeneous case $u_i = (\alpha z, 0, 0)$. This does not provide a significant test of our method, however, since the generalized moment equations, (7), give the exact solution for an initially Gaussian distribution. Another standard advection test is solid-body rotation (Chock and Dunker 1983), but this is also of limited value for a Lagrangian method. The rotational flow tests only the centroid transport equation since there is no local distortion; the initial distribution is advected without change apart from errors in the trajectory calculation. Results after six rotations of an initially conical distribution of unit height with a base diameter of 30 units and the center at 25 units from the center of a 100 × 100 grid are shown in Fig. 3. There are 628 time steps per revolution, and the splitting limit is set so $\Delta x = 2$. The initial cone is represented by 177 circular puffs with $\sigma_{11} = \sigma_{22} = 1$ so the puffs never split, and the contour levels are almost perfectly circular. There is evidence of some slight roughness but this is also present in the initial cone representation and indicates only that more puffs with more overlap would be necessary to produce smoother contours. The only noticeable error is that the cone centroid does not return exactly to its initial

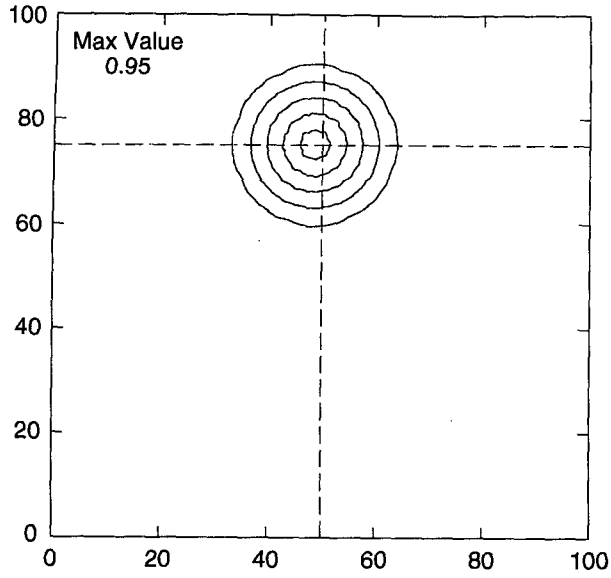

FIG. 3. Puff calculation results for the rotating cone test after six revolutions. The intersection of the dashed lines is the original centroid location. Contour levels are 0.02, 0.2, 0.04, 0.6, and 0.8.

position, and this is a measure of the accuracy of the advection scheme.

A more severe test of the Lagrangian scheme is provided by the deformation flow test proposed by Smolarkiewicz (1982). In this flow, a set of counterrotating cells distorts an initially conical scalar distribution. The flow field is defined by the streamfunction

$$\psi(x, y) = A \sin(kx) \cos(ky),$$

where $A = 8$, $k = 4\pi/L$, $L = 100$ units, and the initial scalar field is a cone of unit height with base diameter 30 centered at (50, 50). Staniforth et al. (1987) present the analytic scalar field solution for this flow, which is an increasing spiral within each rotation cell. As time proceeds, the number of spiral turns increases and the width of the arms decreases. Standard Eulerian tests have been conducted with a unit grid length, and accurate solutions are possible only for $t \leq T/50$ (e.g., Smolarkiewicz 1982; Bott 1989), where $T = 2637.6$ is the final integration time given in Smolarkiewicz's paper. Beyond this time, the stretching effect of the vortex reduces the spiral width below the Eulerian grid scale.

The Lagrangian puff scheme contains no explicit resolution limit on the Gaussian moments, but the maximum puff size must be specified so that the velocity field is resolved. For this test, we use the analytic definition for the velocity and investigate the effect of variations in the splitting criterion Δx. The practical limitation on the calculation is determined by the maximum number of puffs allowable, since the stretching spirals are resolved by an increasing number of puffs as they are stretched and split during the calculation.

The development of the concentration field using the puff model with $\Delta x = 1$ is shown in Fig. 4 along with the analytic solution at $t = T/50$ and $T/10$. The time step is $\Delta t = T/5000$, and the splitting and merging parameters are those given in the previous section. The spiral development is closely predicted at $t = T/50$ and also at $t = T/10$ with little degradation in the accuracy. It should be noted that the puff solution actually contains finer resolution than can be shown on these plots, and the absence of this small-scale information may contribute to some of the roughness evident in the contours. Figure 4 demonstrates the capability of the puff calculation to faithfully reproduce the very sharp gradients that develop near the vortex boundaries, even at the later time. In contrast, these gradients are already absent at $T/50$ in a calculation using Bott's (1989) higher-order Eulerian scheme. The absence of numerical diffusion is evidenced by the clear regions between the spiral arms.

A more quantitative comparison with the exact results can be seen in Fig. 5, which shows the solution along $y = 50$ at the same two times illustrated in Fig. 4. The agreement at $T/50$ with the analytical results is excellent. There is some evidence of small-scale

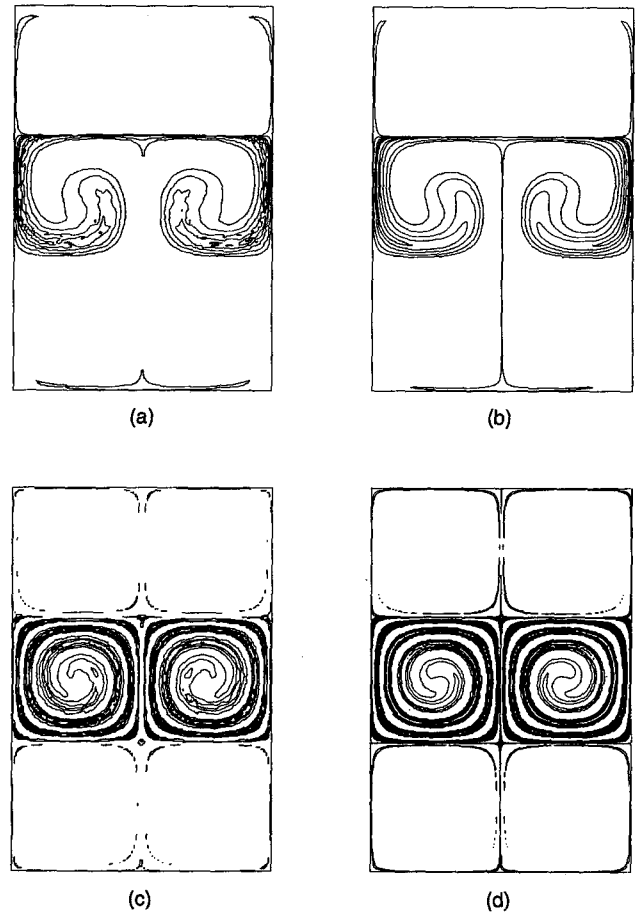

FIG. 4. Puff calculation results and analytic solution for the deformation flow with $\Delta x = 1$: (a) puff result and (b) analytic solution at $t = T/50$; (c) puff result and (d) analytic solution at $t = T/10$. Contour levels as in Fig. 3.

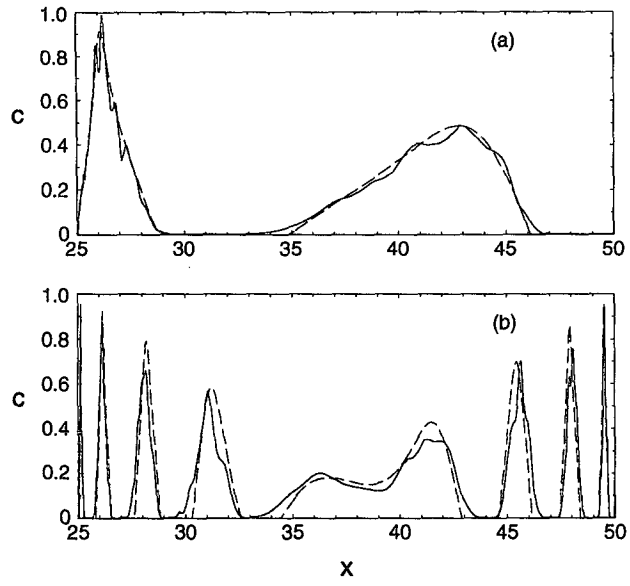

FIG. 5. Cross section at $y = 50$ for deformation flow test at (a) $T/50$, (b) $T/10$. Solid line is puff result, dashed line is analytic solution.

roughness, particularly where the solution is sharply peaked. This could be improved by increasing puff overlap but at the expense of more puffs. At $T/10$ there are some slight shifts in the positions of the peaks with respect to the exact solution; these may be due to advection errors or could result from the splitting–merging algorithms. However, considering the difficult nature of this calculation, the results show very good agreement with the exact solution.

As noted in Table 1, the integration to $T/50$ requires 36 s CPU and creates nearly 3000 puffs in the process. (The calculations were performed on a single processor of a Silicon Graphics 4D/240, which is approximately equivalent to a 486DX2/66.) This is comparable to an equivalent Eulerian calculation but clearly depends on the number of puffs produced by the splitting–merging scheme. To estimate the CPU required by an Eulerian scheme, we consider the very accurate, higher-order scheme of Prather (1986), since lower-order schemes such as SHASTA perform relatively poorly in even the simple rotating cone test (Chock and Dunker 1983). From Prather (1986), we get a CPU time of approximately 10^{-4} s CPU per grid cell per time step. (This assumes that an IBM 3033 is half the speed of the Silicon Graphics processor). Assuming a 100×100 grid with unit grid spacing and time step of $T/3768$ as in Smolarkiewicz (1982), we estimate the integration to $T/50$ would require about 75 s CPU. Thus, the Lagrangian puff method may be more efficient than an Eulerian scheme for some dispersion applications and, as noted above, better able to resolve the sharp gradients.

The dependence on the number of puffs is evident in the computational cost of extending the calculation to $T/10$. As the complexity of the scalar field increases,

TABLE 1. Number of puffs and CPU times required for three resolutions in computing Smolarkiewicz's (1982) cone deformation problem.

	No. of puffs		CPU (s)	
Δx	$T/50$	$T/10$	$T/50$	$T/10$
1	2981	28 387	35.8	4372
2	419	12 291	12.9	1210
4	242	1535	7.3	119

the calculation requires more puffs to represent the solution. The time step is fixed, so the number of steps is increased by a factor of 5 relative to the result for $T/50$, but the CPU increases by more than a factor of 100, taking over 4300 s and generating 28 000 puffs. However, an accurate Eulerian integration out to this time would require such fine resolution that it would likely demand even larger memory storage and CPU time. (Using the estimates above with a 1000×1000 grid yields over 10 h CPU.) Although integrating out to this time is perhaps an overly severe test of a numerical scheme since such deformations would not in practice persist for long periods, it does demonstrate the ability of the current method to resolve a wide range of scales, including those that would be below the grid size in a conventional Eulerian scheme.

The high fidelity of the model calculation shown above suggests that a lower-resolution calculation may still produce acceptable results. Therefore, we now examine the quality of solutions generated using coarser resolution and fewer puffs. The results at $T/50$ are shown since this is a realistic limit for an Eulerian cal-

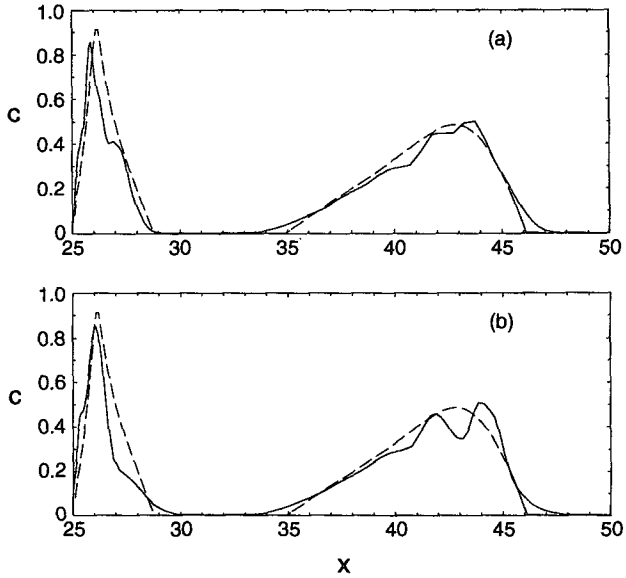

FIG. 6. Effect of resolution on puff result. Cross sections at $y = 50$ and $T/50$ for deformation flow test with (a) $\Delta x = 2$, (b) $\Delta x = 4$. Solid line is puff result; dashed line is analytic solution.

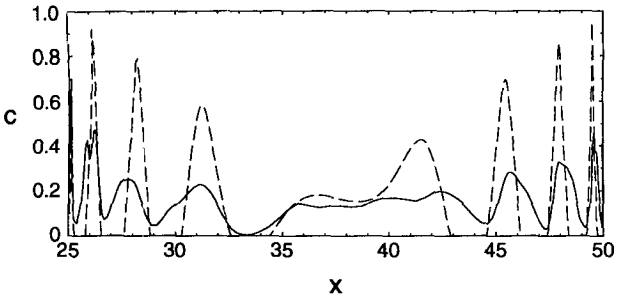

FIG. 7. Cross sections at $y = 50$ and $T/10$ for deformation flow test with $\Delta x = 2$. Solid line is puff result; dashed line is analytic solution.

culation by way of comparison. Figure 6 shows the model solutions at $y = 50$ for $\Delta x = 2$ and $\Delta x = 4$. Both calculations show good agreement with the analytical solution, although inferior to the $\Delta x = 1$ case. However, the computation times noted in Table 1 are significantly less than for the higher-resolution case.

Figure 7 shows the solution along $y = 50$ for the $\Delta x = 2$ case at $T/10$, demonstrating the long-time performance of the method when the velocity field is not well resolved. The calculated solution essentially diffuses the spikes so that the peak values are reduced and become nearly uniform. Also, the widths of the spikes increase so that the clear areas in the theoretical solution (and higher resolution case) are now just areas of relatively low concentration. The solution with $\Delta x = 4$ produces an almost well-mixed cell at $t = T/10$, and there is also some transfer of material into neighboring cells. This is similar to the results from a coarsely resolved Eulerian calculation where the vortex cell boundary does not coincide with the numerical grid. The Lagrangian scheme performance is thus qualitatively similar to the Eulerian schemes and does require appropriate resolution of the velocity field for accurate results. The Lagrangian scheme is able to represent smaller-scale variations in the scalar field, however, as they are generated by the velocity gradients.

4. Conclusions

A Lagrangian puff method based on a generalized Gaussian representation has been presented. The method extends the conventional puff methodology to describe the effects of velocity shears by computing the arbitrary distortions of individual puffs and also by splitting puffs that become larger than a specified size. This allows flow inhomogeneity to be treated accurately, since each puff represents only a localized region of limited size. To avoid exponential growth in the number of puffs, an efficient scheme for merging overlapping puffs has also been implemented, and conservation rules for both splitting and merging have been described. The new scheme provides the efficiency and scale adaptability of a Lagrangian method with the means to describe complex flow field distortions. The Gaussian representation also gives a continuous description of the scalar field and avoids the counting problems of discrete particle methods.

The Lagrangian method has been demonstrated to perform well on the deformational flow test of Smolarkiewicz (1982), in which a continuous stretching motion distorts the initial scalar distribution into an increasingly complicated shape. The adaptive capability of the Lagrangian scheme allows an accurate calculation of the distortion without artificial diffusion for a relatively long time until it is eventually limited by computer resources.

The Lagrangian scheme is numerically efficient for dispersion calculations since the field is calculated only in regions of significant concentration. The accuracy depends on the model resolution, specified through the puff splitting criterion, and can be determined in the same manner as for an Eulerian calculation. The scheme is robust, in the sense that coarser resolution degrades the solution without instability, since all puff moments are conserved at each stage of the calculation.

Acknowledgments. This work was supported by the U.S. Defense Nuclear Agency under Contract DNA 001-93-C-0171.

Disclaimer. The views expressed in this article are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

APPENDIX

Conservation of the Moment Tensor Determinant

To demonstrate that (8) conserves the determinant of the moment tensor, we start by writing out the matrix elements explicitly:

$$\det[\hat{\sigma}^{(0)}] = \begin{vmatrix} \sigma_{11}e^{2S_{11}} & \sigma_{12}e^{S_{11}+S_{22}} & \sigma_{13}e^{S_{11}+S_{33}} \\ \sigma_{12}e^{S_{11}+S_{11}} & \sigma_{22}e^{2S_{22}} & \sigma_{23}e^{S_{22}+S_{33}} \\ \sigma_{13}e^{S_{11}+S_{33}} & \sigma_{23}e^{S_{22}+S_{33}} & \sigma_{33}e^{2S_{33}} \end{vmatrix}, \tag{A1}$$

where $S_{\alpha\alpha} = \Delta t \partial u_\alpha / \partial x_\alpha$ (no summation). It can be seen that the matrix in (A1) is simply the original matrix σ pre- and postmultiplied by the diagonal matrix

$$\begin{bmatrix} e^{S_{11}} & 0 & 0 \\ 0 & e^{S_{22}} & 0 \\ 0 & 0 & e^{S_{33}} \end{bmatrix}.$$

Therefore, (A1) becomes $\det[\hat{\sigma}^{(0)}] = \det(\sigma)e^{2(S_{11}+S_{22}+S_{33})} = \det(\sigma)$ for a divergence-free velocity field.

To demonstrate that the relations (9) preserve the determinant, we again write out the elements of the matrix explicitly. We do this for $m = 1$ only, but the results hold for the other intermediate steps since they involve merely a permutation of indices. We have

$$\det[\hat{\sigma}^{(1)}] = \begin{vmatrix} \sigma_{11} + S_{12}(2\sigma_{12} + S_{12}\sigma_{22}) & \sigma_{12} + S_{12}\sigma_{22} & \sigma_{13} + S_{12}\sigma_{23} \\ \sigma_{12} + S_{12}\sigma_{22} & \sigma_{22} & \sigma_{23} \\ \sigma_{13} + S_{12}\sigma_{23} & \sigma_{23} & \sigma_{33} \end{vmatrix}, \quad \text{(A2)}$$

where the carets and superscripts on σ_{ij} on the right-hand side have been dropped for clarity and $S_{12} = \Delta t \partial u/\partial y$. It is apparent that the matrix in (A2) can be constructed in two steps by 1) multiplying the second row of the original matrix (at $m = 0$) by S_{12} and adding the result to the first row and 2) multiplying the second column of the new matrix by S_{12} and adding the result to the first column. Since a fundamental property of a matrix is that its determinant is unchanged by adding a constant multiple of one row or column to another row or column, each step preserves the determinant of the original matrix.

REFERENCES

Bass, A., 1980: Modelling long range transport and diffusion. *Second Joint Conf. on Applications of Air Pollution Meteorology,* New Orleans, LA, Amer. Meteor. Soc., 193–215.

Boris, J., 1986: A vectorized "near neighbors" algorithm of order N using a monotonic logical grid. *J. Comput. Phys.,* **66,** 1–20.

Bott, A., 1989: A positive definite advection scheme obtained by nonlinear renormalization of the advective fluxes. *Mon. Wea. Rev.,* **117,** 1006–1015.

Chock, D. P., and A. M. Dunker, 1983: A comparison of numerical methods for solving the advection equation. *Atmos. Environ.,* **17,** 11–24.

Diehl, S. R., D. T. Smith, and M. Sydor, 1982: Random-walk simulation of gradient-transfer processes applied to dispersion of stack emission from coal-fired power plants. *J. Appl. Meteor.,* **21,** 69–82.

Egan, B. A., and J. R. Mahoney, 1972: Numerical modeling of advection and diffusion of urban area source pollutants. *J. Appl. Meteor.,* **11,** 312–322.

Hurley, P., 1994: PARTPUFF—A Lagrangian particle-puff approach for plume dispersion modeling applications. *J. Appl. Meteor.,* **33,** 285–294.

Lange, R., 1978: ADPIC—A three-dimensional computer code for the study of pollutant dispersal and deposition under complex conditions. *J. Appl. Meteor.,* **17,** 320–329.

Prather, M. J., 1986: Numerical advection by conservation of second-order moments. *J. Geophys. Res.,* **91,** 6671–6681.

Reid, J. D., 1979: Markov chain simulations of vertical dispersion in the neutral surface layer for surface and elevated releases. *Bound.-Layer Meteor.,* **16,** 3–22.

Smolarkiewicz, P. K., 1982: The multidimensional Crowley advection scheme. *Mon. Wea. Rev.,* **110,** 1968–1983.

Staniforth, A., J. Côté, and J. Pudykiewicz, 1987: Comments on "Smolarkiewicz's deformational flow." *Mon. Wea. Rev.,* **115,** 894–900.

Sykes, R. I., and D. S. Henn, 1992: An improved moment conservation method for the advection–diffusion equation. *J. Appl. Meteor.,* **31,** 112–118.

——, S. F. Parker, D. S. Henn, and W. S. Lewellen, 1993: Numerical simulation of ANATEX tracer data using a turbulence closure model for long-range dispersion. *J. Appl. Meteor.,* **32,** 929–947.

Thomson, D. J., 1987: Criteria for the selection of stochastic models of particle trajectories in turbulent flows. *J. Fluid Mech.,* **180,** 529–556.