LEGIBILITY NOTICE

A major purpose of the Technical Information Center is to provide the broadest dissemination possible of information contained in DOE's Research and Development Reports to business, industry, the academic community, and federal, state and local governments.

Although a small portion of this report is not reproducible, it is being made available to expedite the availability of information on the research discussed herein.

Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7400-ENG-30

1

LA-UR--89- 590

DE89 007994

**TITLE IMPROVEMENTS OF THE KIVA-II COMPUTER PROGRAM
FOR NUMERICAL COMBUSTION**

AUTHOR(S) *P. J. O'Rourke, A. A. Amsden, T. D. Butler, and T. L. McKinley**

SUBMITTED TO **Third International Conference on Numerical Combustion,
May 23-26, 1989, Antibes France**

DISCLAIMER

This report was prepared in fulfillment of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

*Cummins Engine Company
Columbus, IN 47202

Los Alamos National Laboratory is pleased to provide this document to the public free of charge. However, it is requested that you acknowledge the source of this document when you use it for non-government purposes.

For more information, please request a copy of the Los Alamos National Laboratory Information Manual, which contains information on how to obtain documents from the Laboratory.

Los Alamos

**Los Alamos National Laboratory
Los Alamos, New Mexico 87545**

IMPROVEMENTS OF THE KIVA-II COMPUTER PROGRAM FOR NUMERICAL COMBUSTION

By

P. J. O'Rourke, A. A. Amsden, T. D. Butler
Group T-3, Mail Stop B216
Los Alamos National Laboratory
Los Alamos, NM 87545

and

T. L. McKinley
Cummins Engine Company
Columbus, IN 47202

ABSTRACT

This paper describes and illustrates the principal differences between the newly-released KIVA-II and the KIVA computer programs. Both programs are for the numerical calculation of two- and three-dimensional fluid flows with chemical reactions and sprays. Because of improvements to KIVA-II, it is faster, more accurate, and applicable to a wider variety of problems involving combustion and two-phase flow.

I. INTRODUCTION

The KIVA-II computer program¹ has been developed for the numerical solution of transient two- and three-dimensional fluid flows with chemical reactions and sprays. KIVA-II extends and enhances the capabilities of the earlier KIVA code,^{2,3} improving its numerical efficiency and accuracy and its ease-of use and versatility. In this paper we describe the principal differences between KIVA-II and KIVA and illustrate these with computational examples.

KIVA-II retains several of the basic features of the KIVA code, and users of KIVA should find the transition to KIVA-II to be straightforward. Finite differencing is based on a finite volume method called the ALE (Arbitrary Lagrangian-Eulerian) method.^{4,5} Spatial differences are formed with respect to a computational mesh composed of arbitrary hexahedrons. Spatial differencing is made conservative whenever possible by differencing integral forms of the conservation equations. Cartesian components of the velocity field are stored at cell vertices, and the linear momentum equations are differenced in a strictly conservative fashion. All other fluid variables are stored at cell centers. As in KIVA,¹ KIVA-II uses cell face velocities during a portion of the computational cycle to reduce the susceptibility of com

puted solutions to parasitic velocity modes and thereby largely eliminate the need for alternate node coupling.^{1,2}

Temporal differencing is based on a computational cycle that is divided into two phases - a Lagrangian phase and a rezone phase. In the rezone phase the vertices, which have moved with the fluid in the Lagrangian phase, are moved to new user specified positions, and the flow field is remapped onto the new computational mesh. The formulation allows a Lagrangian, Eulerian, or a mixed description. The arbitrary mesh can move to follow changes in combustion chamber geometry, such as occur in internal combustion engines.

Equations for an evaporating liquid fuel spray are solved using the stochastic particle method.^{6,7} Computational particles - which represent droplets of specified size, velocity, and temperature - interact with the fluid by exchanging mass, momentum, and energy in a conservative fashion. Probability distributions often govern the assignment of particle properties at injectors or the evolution of particle properties at downstream locations. When this is the case, the distributions are randomly sampled to determine the particle properties or their evolution.

The differences between KIVA-II and KIVA are summarized in Table 1.

TABLE I
DIFFERENCES BETWEEN KIVA II AND KIVA

1. Computational Efficiency Improvements
 - a. Coupled, implicit differencing of diffusion and pressure wave propagation terms
 - b. Subcycled calculation of convection
 - c. 2-D to 3-D converter
2. Numerical Accuracy Improvements
 - a. Optional quasi-second order upwind convection scheme
 - b. Generalized mesh diffusion algorithm
 - c. Method for computing droplet turbulent dispersion when the time step Δt exceeds the droplet turbulence correlation time
3. New or Improved Physical Submodels
 - a. $k-\epsilon$ turbulence model
 - b. Revised subgrid scale turbulence model
 - c. Model for droplet aerodynamic breakup

4. Improvements in Ease-of-Use and Versatility
 - a. More flexible mesh generation for internal combustion engines
 - b. Inflow/Outflow boundaries
 - c. Simplified velocity boundary conditions
 - d. Gravitational terms
 - e. Library of thermophysical properties of common hydrocarbons
 - f. Alphabetized epilogue giving FORTRAN variables and their definitions.

All of these differences are fully documented in the KIVA-II report. In this paper we describe and illustrate the most important changes, which are 1a, 1b, 2a, 3a, 3b, and 3c of Table I.

II. NUMERICAL EFFICIENCY IMPROVEMENTS

The main criticism of the KIVA code was its slowness. This was largely due to the fact that explicit difference schemes were used for diffusion and convection terms, and a subcycled explicit scheme was used to calculate pressure wave propagation.² Each of these schemes had stability conditions that limited the size of the computational timestep. The stability conditions for the convection and diffusion terms usually controlled the main computational timestep, and often limited it to very small values. The diffusional stability criterion was particularly restrictive because the timestep was proportional to the square of the computational cell dimension. In low Mach number problems, the subcycle timestep for calculating pressure waves could become very small and result in further inefficiency.

To improve computational efficiency we have implemented two major changes in KIVA-II. First, implicit differencing is used to calculate the diffusion terms and terms associated with pressure wave propagation. Second, although it still uses explicit schemes, the convection calculation can be repeated or subcycled an arbitrary number of times using timesteps that satisfy the explicit stability criterion for convection and that are submultiples of the main computational timesteps. Because of these changes, the main computational timestep in KIVA II is no longer controlled by stability criteria, but by accuracy criteria associated with fluid accelerations and cell distortions during the Lagrangian phase and with coupling spray and chemical source terms to the fluid flow.¹

The implicit equations for diffusion and pressure wave propagation are solved during the Lagrangian phase by a modified SIMPLE method⁸ with individual equations solved by the conjugate residual method.⁹ It should be noted that we first tried a noniterative alternative to the SIMPLE method that was similar to the PISO¹⁰

method, but we found this to be unsatisfactory. Although the numerical truncation errors of the PISO-like method could be shown to be formally of second-order in the timestep Δt , in practice these errors were sometimes unacceptably large. The conjugate residual method⁹ was chosen because of its rapid convergence properties and its low computer storage requirements and because it is vectorizable on computers that offer this feature.

A novel feature of the implicit differencing in KIVA-II is that the amount of implicitness is variable. Weighted averages of the old- and new-time values of pressures and diffused quantities are automatically calculated. The weighting factor for the pressures is based on the Courant number $c\Delta t/\Delta x$, where c is the speed of sound, and the weighting factor for diffused quantities is based on the local diffusion Courant number $D\Delta t/\Delta x^2$, where D is the diffusivity. The weighting factors are chosen to ensure numerical stability and also to reduce the amount of implicitness and thereby to improve computational efficiency. When the Courant number is small enough, KIVA-II automatically uses explicit schemes for which no costly iterative solution is required. When Courant numbers are large enough that implicitness is required, KIVA-II uses partially-implicit schemes that can be solved more quickly than fully-implicit schemes by iterative methods such as the conjugate residual method.

The idea of convective subcycling is that we calculate convection using an explicit scheme and timestep Δt_c that is an integral submultiple of the main computational timestep Δt . The timestep Δt_c must satisfy the Courant condition $u_r \Delta t_c / \Delta x < 1$, based on the fluid velocity u_r relative to the computational grid, but because there is no limit on the number of subcycles, we can have $u_r \Delta t / \Delta x > 1$. Convective subcycling was fairly easy to implement in KIVA-II because the calculation of convection, which is performed in the rezone phase, is split from the calculation of the remaining fluid flow terms. The method saves computational time because the rezone calculation typically takes only ten percent of the time of the Lagrangian phase calculation. Assuming the rezone phase calculation takes a fraction f of the computational time of the Lagrangian phase and that we subcycle N times per computational cycle, one can show that a calculation with convective subcycling takes a fraction

$$\frac{1 + Nf}{N(1 + f)}$$

of the time of a calculation without subcycling.

Convective subcycling has advantages and a disadvantage relative to the alternative of implicit differencing of convection terms. Generally, it is more accurate because if we subcycle, the timesteps we use to calculate convection are smaller and thus temporal truncation errors are smaller. Also because the calculation is explicit it is easier to use convective subcycling in conjunction with nonlinear monotone difference schemes. Such a scheme has been incorporated in KIVA-II and will be described in the next section. The disadvantage is that convective subcycling is generally slower than implicit differencing. With a little extra computational work convection could be calculated implicitly at the same time that the diffusion terms are calculated (the description "Lagrangian phase" would then no longer apply), but we believe the advantages of convective subcycling outweigh this one disadvantage.

Table II gives computational time comparisons between KIVA and KIVA-II for some two- and three-dimensional internal combustion engine calculations. The two codes were equally fast on a two-dimensional coarse mesh calculation, but KIVA-II was faster on two- and three-dimensional calculations in which the meshes were more refined and the timestep used by KIVA was limited by either convective or diffusional stability criteria. Generally, the more refined one's mesh is, the faster KIVA-II will be in comparison to KIVA. We also performed two coarse mesh calculations using KIVA-II in which the timestep was required to be less than 1.0×10^{-5} s.

TABLE II
COMPUTATIONAL TIME COMPARISONS FOR
2-D AND 3-D ENGINE CALCULATIONS

	<u>KIVA</u>	<u>KIVA-II</u>
2-D Calculations (6.25×10^{-3} s problem time)		
Course Mesh (440 cells)		
Unrestricted Δt	49.5'	53.5'
$\Delta t_{max} = 1 \times 10^{-5}$ s	---	130.8'
$\Delta t_{max} = 1 \times 10^{-5}$ s, fully implicit diffusion	---	176.8'
Fine Mesh (1760 cells)	5.13'	3.93'
3-D UPS Engine Calculations¹¹ (-118" to +92", 8712 cells)	3.0 hrs	1.44 hrs

With this small timestep KIVA-II automatically differenced the diffusion terms explicitly, but we also performed a calculation in which we overrode this automatic feature and differenced the diffusion terms fully implicitly. The fully-implicit calculation was thirty-five percent slower, showing the benefit of using variable implicitness.

III. NUMERICAL ACCURACY IMPROVEMENT

The major improvement in numerical accuracy of KIVA-II results from the implementation of a monotone scheme for convection that we call quasi-second-order upwind (QSOU) differencing. For a definition of monotone schemes and a detailed discussion of the QSOU method, the reader is referred to Appendix M of Ref. 1. Here we simply discuss and illustrate some of the general features of QSOU.

The QSOU method is a combination of Leith's method¹² or interpolated donor cell differencing, which is second-order accurate in space, and upwind or donor cell differencing, which is first-order accurate and monotone. QSOU aims to maximize the proportion of Leith's method that is used, while preserving monotonicity. There have been many schemes based on this same general concept,^{13,14,15} but QSOU is based primarily on the ideas of van Leer.¹⁶ By using a slightly more strict gradient-limiter than van Leer, QSOU ensures monotonicity in one-dimensional calculations. We have implemented a simple multidimensional extension in which fluxes in each coordinate direction depend only on gradients in that direction. This extension has given monotone results in all test cases, as evidenced by the following calculational results.

Figure 1 gives results from five calculations in which we convect a scalar function through a mesh of square cells and at a 45 degree angle to the mesh directions. The function has initial values equal to one in a square-shaped region and zero outside this region. The calculations differ in the convection schemes used; one uses QSOU and two others use explicit donor cell and interpolated donor cell. Only the explicit donor cell and interpolated donor cell schemes are available as options when using the KIVA code. Two calculations use the fully implicit donor cell and QUICK²⁸ schemes, which are not available in KIVA II or KIVA, and timesteps that are twenty five times larger than the timestep used for the explicit schemes. These larger time steps are chosen to show that although they are unconditionally stable, the fully implicit methods can be much less accurate than subcycled explicit methods when $u \Delta t / \Delta x > 1$. The results are plots of isopleths of the computed solution at a time when the exact solution is a translation of the initially square shaped region a distance equal to the side of the square in each mesh direction. The reported error is the

Fig. 1. Isopleths from calculations of convection of a square-shaped scalar profile.

root-mean-square error between the computed and exact solutions, averaged over the region in which the exact solution is unity. By this error criterion the QSOU solution is most accurate. The interpolated donor cell solution is more accurate than donor cell, but because interpolated donor cell is not monotone its solution assumes values that lie outside the range of the initial conditions. The implicit calculations using the large timestep are least accurate.

IV. NEW OR IMPROVED PHYSICAL SUBMODELS

A. Turbulence Models

The user of KIVA-II has the option of using one of two turbulence models: a two equation $k-\epsilon$ model that is not available in KIVA and a subgrid scale (SGS) model that is a modification of the one-equation model of KIVA.² In this section we give the equations of both models and illustrate some differences between them with a computational example.

The $k-\epsilon$ equations are the standard ones for incompressible flow,¹⁷ modified to include volumetric expansion effects¹⁸ and spray/turbulence interactions.¹⁹

$$\frac{\partial \rho k}{\partial t} + \nabla \cdot (\rho \mathbf{u} k) + \frac{1}{2} \rho k \nabla \cdot \mathbf{u} = \sigma \nabla \mathbf{u} + \nabla \cdot (\mu_T / \sigma_k \nabla k) - \rho \epsilon - W_s \quad (1)$$

and

$$\frac{\partial \rho \epsilon}{\partial t} + \nabla \cdot (\rho \mathbf{u} \epsilon) + \left(\frac{1}{2} c_{\epsilon 1} - c_{\epsilon 3} \right) \rho \epsilon \nabla \cdot \mathbf{u} = \frac{\epsilon}{k} c_{\epsilon 1} (\sigma : \nabla \mathbf{u}) + \nabla \cdot (\mu_T / \sigma_\epsilon \nabla \epsilon) - c_{\epsilon 2} \rho \frac{\epsilon^2}{k} - c_s \frac{\epsilon}{k} W_s \quad (2)$$

In these equations, ρ is the fluid mass density and $\mathbf{u}$ the fluid velocity. The stress tensor σ is given by

$$\sigma = \mu_T [\nabla \mathbf{u} + (\nabla \mathbf{u})^T] - \frac{1}{2} \mu_T \nabla \cdot \mathbf{u} \mathbf{I} \quad (3)$$

where the turbulent viscosity is obtained from

$$\mu_T = \rho c_\mu \frac{k^2}{\epsilon} + \mu_\ell \quad (4)$$

and μ_ℓ is the laminar viscosity. The spray interaction term W_s accounts for the work done by the turbulence in dispersing the spray droplets. KIVA-II uses standard values of the constants $c_{\epsilon 1}$, $c_{\epsilon 2}$, $c_{\epsilon 3}$, σ_k , σ_ϵ , and c_μ , and these are given in Table III.

TABLE III

$k-\epsilon$ MODEL CONSTANTS

$c_{\epsilon 1} = 1.44$	$\sigma_k = 1.0$
$c_{\epsilon 2} = 1.92$	$\sigma_\epsilon = 1.3$
$c_{\epsilon 3} = -1.0$	$c_\mu = 0.09$

A value of c_s equal to 1.50 has been suggested²⁰ based on the postulate of length scale conservation, and has been found to give good agreement in experimental comparisons.¹⁹

The wall boundary conditions for these equations are

$$\nabla k \cdot \mathbf{n} = 0 \quad (5)$$

and

$$\varepsilon = \left| \frac{c_\mu}{\sigma_\varepsilon (c_{\varepsilon_2} - c_{\varepsilon_1})} \right|^{1/2} \frac{k^{3/2}}{L}, \quad (6)$$

where $\mathbf{n}$ is the unit normal to the wall and (6) is prescribed at the centers of cells that lie next to the wall and whose centers are distance L from the wall. In addition, when the $k - \varepsilon$ model is used, wall momentum and energy losses are calculated by matching to turbulent law-of-the-wall profiles. The details of this calculation are given in Appendix B of Ref. 1.

The SGS model of KIVA-II is implemented by solving Eqs. (1) and (2) and requiring that the local length scales computed from Eq. (6) be less than or equal to $m\Delta x$, where Δx is a typical computational cell dimension and m equals 3 or 4. More precisely, we take ε to be the maximum of its value computed from Eq. (2) and

$$\varepsilon' = \left| \frac{c_\mu}{\sigma_\varepsilon (c_{\varepsilon_2} - c_{\varepsilon_1})} \right|^{1/2} \frac{k^{3/2}}{m\Delta x} \quad (7)$$

In regions where $\varepsilon = \varepsilon'$ (or $L = m\Delta x$), the model reduces to a one-equation model that is identical to the SGS model of KIVA,² except some of the constants are slightly different. In regions where $\varepsilon > \varepsilon'$ (or $L < m\Delta x$) the model reduces to the full $k - \varepsilon$ model. The old SGS model in KIVA is in error in taking $L = m\Delta x$ because there are often regions of the flow field, for example near walls, where all length scales become less than $m\Delta x$. Near walls the velocity solution to one's turbulence equations should, under certain ideal conditions,¹ be the logarithmic velocity profile. This is true for the $k - \varepsilon$ model, but is not true for a model that assumes a constant length scale.

Although the $k - \varepsilon$ and SGS models of KIVA-II are formally very similar, the results they give can be very different because the SGS turbulent diffusivity can be

much smaller than the $k-\epsilon$ diffusivity. This is demonstrated by recent three-dimensional calculations we have performed of the decay of a swirling flow in a cylindrical vessel. The computational mesh is shown in Fig. 2. We initially specified an axisymmetric flow field with only an azimuthal or swirl velocity component. In experiments^{21, 22} such flow fields are observed to be unstable and to develop a precessional motion of the swirl center. We performed calculations using both $k-\epsilon$ and SGS, and plots of the swirl center location in the midplane of the cylinder are shown in Fig. 3. In the SGS calculation a precessional motion is obtained, but in the $k-\epsilon$ model the flow remains nearly axisymmetric. Apparently, the $k-\epsilon$ diffusivity is large enough to suppress the instability.

B. Droplet Aerodynamic Breakup Model

Recently it has been found²³ that droplet aerodynamic breakup is important in certain engine sprays. This finding is consistent with an earlier study²⁴ that it is primarily an aerodynamic mechanism that causes atomization of liquid jets. In response to these findings, we have developed and incorporated in KIVA-II a model for droplet aerodynamic breakup. We now summarize the basis for the model and give additional evidence for the importance of droplet breakup in diesel-type sprays.

We call our model the Taylor-Analogy Breakup (TAB) method because it is based on an analogy proposed by G. I. Taylor²⁵ between an oscillating droplet and a spring/mass system. The restoring force of the spring is analogous to surface tension

Fig. 2. Top and perspective views of computational mesh for swirling flow calculations.

Fig. 3. Swirl center location in $k-\epsilon$ [left, $(x/R)_{max} \approx .0025$] and SGS [right, $(x/R)_{max} \approx .05$] calculations. R is the cylinder radius.

forces, and external forces on the spring/mass system are analogous to aerodynamic forces on the droplet. To the analogy we have added damping forces due to liquid viscosity. A droplet breaks up when its distortion exceeds a certain critical value. Details concerning the equations solved and their solution procedure can be found in Refs. 1 and 26.

The model has been used in comparison with experimental measurements of diesel sprays,²⁶ and we give here a result that was not previously reported. Good agreement was obtained between computed and measured spray penetrations and drop sizes for an axisymmetric, nonvaporizing spray. We wished to use the computer program to assess the importance of aerodynamic breakup in the experimental sprays. One measure of how well the spray is broken up is the spray surface area S per unit length along the axis of the spray. An equation for S can be derived from the spray equation,²⁷ and for a steady, nonvaporizing spray this is

$$S = \frac{u_a}{v} + u_a \frac{S}{v} = \frac{dS}{dt}_{coll} + \frac{dS}{dt}_{breakup} \quad (8)$$

The term dS/dt_{coll} is the time rate of change of S due to collisions and will be negative since collisions frequently result in coalescences, which reduce spray surface area. The term $dS/dt_{breakup}$ is the time rate of change of S due to breakups and will be

positive. The quantity u_s is a surface-area averaged velocity. We calculated S , u_s , $dS/dt|_{coll}$, and $dS/dt|_{breakup}$ by time-averaging the computed steady solution for 500 computational cycles. The values of $\partial S/\partial z$ and $\partial u_s/\partial z$ were then found by simple numerical differentiation.

Figure 4 shows plots of the terms $S \partial u_s/\partial z$, $u_s \partial S/\partial z$, $-dS/dt|_{coll}$, and $-dS/dt|_{breakup}$ as functions of distance along the axis of the spray. It is seen that at all axial locations the collision and breakup terms are much larger than the other two and that they nearly balance each other, with differences between them being comparable in magnitude to the other two terms. If collisions or breakups, individually or in combination, were ignored, the rate of change of S for an observer moving with velocity u_s down the spray axis, which is $u_s \partial S/\partial z$, would differ from the computational result. This supports the contention that to calculate diesel-type sprays accurately, it is necessary to include the effects of both droplet collisions and breakups.

Fig. 4. Terms in S equation ($\text{cm}^2/\text{cm s}$) vs. distance from the injector (cm).

ACKNOWLEDGMENT

This work was supported by the U.S. Department of Energy, Office of Energy Utilization Research, Energy Conversion and Utilization Technologies Program.

REFERENCES

1. A. A. Amsden, P. J. O'Rourke, and T. D. Butler, "KIVA-II: A Computer Program for Chemically Reactive Flows with Sprays," Los Alamos National laboratory report (in preparation).
2. A. A. Amsden, J. D. Ramshaw, P. J. O'Rourke, and J. K. Dukowicz, "KIVA: A Computer Program for Two- and Three-Dimensional Fluid Flows with Chemical Reactions and Fuel Sprays," Los Alamos National Laboratory report LA-10245-MS (February 1985).
3. A. A. Amsden, J. D. Ramshaw, L. D. Cloutman, and P. J. O'Rourke, "Improvements and Extensions to the KIVA Computer Program," Los Alamos National Laboratory report LA-10534-MS (October 1985).
4. C. W. Hirt, A. A. Amsden, and J. L. Cook, *J. Comput. Phys.* **14**, 227 (1974).
5. W. E. Pracht, *J. Comput. Phys.* **17**, 132 (1975).
6. J. K. Dukowicz, *J. Comput. Phys.* **35**, 229 (1980).
7. P. J. O'Rourke, "Collective Drop Effects in Vaporizing Liquid Sprays," Ph.D. Thesis 1532 T, Princeton University (August 1981).
8. S. V. Patankar, *Numerical Heat Transfer and Fluid Flow* (Hemisphere Publishing Corporation, Washington D.C., 1980).
9. P. J. O'Rourke and A. A. Amsden, "Implementation of a Conjugate Residual Iteration in the KIVA Computer Program," Los Alamos National Laboratory report LA-10849-MS (October 1986).
10. R. I. Issa, *J. Comput. Phys.* **62**, 1, 40 (1986).
11. P. J. O'Rourke and A. A. Amsden, "Three Dimensional Numerical Simulations of the UPS 292 Stratified Charge Engine," SAE Technical Paper 870597 (1987).
12. P. J. Roache, *Computational Fluid Dynamics* (Hermosa Publishers, Albuquerque, New Mexico, 1982).
13. S. T. Zalesak, *J. Comp. Phys.* **31**, 335 (1979).
14. M. Chapman, *J. Comp. Phys.* **44**, 84 (1981).
15. L. D. Cloutman, "Numerical Experiments with CONCHAS SPRAY: Installation of the FLOE Algorithm," Los Alamos National Laboratory report LA-10626-MS (February 1986).
16. B. van Leer, *J. Comput. Phys.* **32**, 101 (1979).
17. B. E. Launder and D. B. Spalding, *Mathematical Models of Turbulence* (Academic Press, New York, 1972).
18. W. C. Reynolds, "Modeling of Fluid Motions in Engines - An Introductory Overview," in *Combustion Modeling in Reciprocating Engines*, eds. J. N. Mattav and C. A. Amann (Plenum Press, New York, 1981).

19. R. D. Reitz and R. Diwakar, "Structure of High Pressure Fuel Sprays," SAE Technical Paper 870598 (1987).
20. P. J. O'Rourke, oral presentation to the Twenty-third Direct Injected Stratified Charge Working Group Meeting, Los Alamos National Laboratory, March 1986.
21. T. M. Dyer, "Characterization of One- and Two-Dimensional Homogeneous Combustion Phenomena in a Constant Volume Bomb," SAE paper 790353 (1979).
22. S. Wahiduzzaman and C. R. Ferguson, "Convective Heat Transfer from a Decaying Swirling Flow within a Cylinder," 8th International Heat Transfer Conference, Paper #86 IHTC-253 (August 1986).
23. R. D. Reitz and R. Diwakar, "Effect of Drop Breakup on Fuel Sprays," SAE Paper 860469 (1986).
24. R. D. Reitz and F. V. Bracco, "On the Dependence of Spray Angle on Nozzle Design and Operating Conditions," SAE Paper 790494 (1979).
25. G. I. Taylor, "The Shape and Acceleration of a Drop in a High Speed Air Stream" in *The Scientific Papers of G. I. Taylor*, ed. G. K. Batchelor Vol. III (University Press, Cambridge, 1963).
26. P. J. O'Rourke and A. A. Amsden, "The TAB Method for Numerical Calculation of Spray Droplet Breakup," SAE Technical Paper 872089 (1987).
27. F. A. Williams, *Combustion Theory* (Benjamin/Cummings, Menlo Park, California, 1985).
28. B. P. Leonard, *Comput. Meths. Appl. Mech. Engrg.* 19, 59 (1979).