A Fast Solver for Systems of Axisymmetric Ring Vortices

James H. Strickland, 1552
Donald E. Amos, 1422
Sandia National Laboratories
Albuquerque, New Mexico 87185

MASTER

Abstract

A method which is capable of efficient calculation of the axisymmetric flow field produced by a large system of ring vortices is presented in this report. The system of ring vortices can, in turn, be used to model body surfaces and wakes in incompressible unsteady axisymmetric flow fields. This method takes advantage of source point and field point series expansions which enables one to make calculations for interactions between groups of vortices which are in well separated spatial domains rather than having to consider interactions between every pair of vortices. In this work, series expansions for the stream function of the ring vortex system are obtained. Such expansions explicitly contain the radial and axial velocity components. A Fortran computer code RSOLV has been written to execute the fast solution technique to calculate the stream function and the axial and radial velocity components at points in the flow field. Test cases have been run to optimize the code and to benchmark the truncation errors and CPU time savings associated with the method. Non-dimensional truncation errors for the stream function and total velocity field are on the order of 5×10^{-5} and 3×10^{-3} respectively. Single precision accuracy produces errors in these quantities up to about 1×10^{-5} . For 100 vortices in the field, there is virtually no CPU time savings with the fast solver. For 10,000 vortices in the flow, the fast solver obtains solutions in about 1% to 3% of the time required for the direct solution technique. Simulations of vortices with square and circular cores were run in order to obtain expressions for the self-induced velocities of such vortices. Correlations for the non-dimensional self-induced velocity as a function of the non-dimensional ring radius were found to be almost identical for the square and circular core vortices. It appears that these correlations can be used to correctly convect discretized patches of vorticity in the flow field.

EB

Acknowledgements

The authors would like to thank Greg Homicz, Senior Member of Technical Staff in Sandia National Laboratories' Computational Aerodynamics Division (Organization 1556) for introducing them to the papers on fast multipole methods which forms the underlying basis for the present work.

Nomenclature

A_c	= Vortex core area.
$A_{n,m}$	= Source coefficient.
$A'_{n,m}$	= Translated source coefficient.
a	= Circular core radius.
$B_{n,m}$	= Field coefficient.
$B'_{n,m}$	= Translated field coefficient.
b	= Vortex ring or source box radius.
b_i	= Ring radius of i^{th} vortex.
b^+	= Non-dimensionalized b .
C_a	= Circular core axial velocity function.
C_s	= Square core axial velocity function.
$C_{n,m}$	= Coefficient of $I_{\mu\nu}^n$.
D_F	= Field domain.
D_S	= Source domain.
D_1	= Source domain 1.
D_2	= Source domain 2.
D_3	= Field domain 3.
D_4	= Field domain 4.
E	= Elliptic Integral of the first kind.
E_V	= Total velocity error.
E_ψ	= Stream function error.
F	= Ring vortex function.
F_i	= Ring vortex function for i^{th} vortex.
$F_{l,m,n}$	= The partial of F with respect to x^l , r^m , and b^n .
F^+	= Non-dimensionalized F .
f_{v1}	= Contribution from list 1 boxes to field variable.
f_{v3}	= Contribution from list 3 boxes to field variable.
f_{v24}	= Contribution from list 2 and 4 boxes to field variable.
f_{vtotal}	= Total field variable.
I	= Impulse of system of vortex rings.
$I_{\mu\nu}^n$	= Basic partial derivatives of F .
i	= Index.
J_0	= Zeroth order Bessel function.
J_1	= First order Bessel function.
J_μ	= μ^{th} order Bessel function.
j	= Index.
K	= Elliptic Integral of the second kind.
k	= Index or dummy variable.
L	= Index.
l	= Index or level.

Nomenclature (Continued)

M	= Recursion method for obtaining partials of F .
m	= Index.
N	= Maximum order of the partials of F .
N_c	= Maximum number of vortices in any childless box.
N_p	= Number of charged particles.
N_v	= Number of vortices.
n	= Index.
P	= Field point.
p	= Index.
$Q_{\nu-\frac{1}{2}}^m$	= Associated Legendre function of the second kind.
q	= Index.
R_F	= Radius of field domain.
R_S	= Radius of source domain.
r	= Radial coordinate.
r_c	= Radial spiral center.
r_f	= Radial distance from center of field domain 3 to field point P .
r_o	= Radial center of impulse.
r_p	= Radius at point P .
r^+	= Non-dimensionalized r .
S_l	= Box dimension at level l .
S_r	= Spiral radius.
s	= Square core size.
t_{cpu}	= Computer processing time.
U_o	= Axial core self-induced velocity.
u	= Axial velocity.
u_i	= Axial Velocity at i^{th} vortex due to other vortices.
u_{oi}	= Self-induced axial velocity at i^{th} vortex.
u_p	= Axial velocity at point P .
V_i	= Total velocity at point i obtained using fast solver.
V_{ei}	= Total velocity at point i obtained using direct method.
V_o	= Radial core self-induced velocity
v_i	= Radial Velocity at i^{th} vortex due to other vortices.
v_p	= Radial velocity at point P .
x	= Axial coordinate.
x_c	= Axial spiral center.
x_f	= Axial distance from center of field domain 3 to field point P .
x_i	= Axial coordinate of i^{th} vortex.
x_o	= Axial center of impulse.
x_p	= Axial distance to point P .

Nomenclature (Continued)

x^+	= Non-dimensionalized x .
z	= General argument of Bessel function.
Γ	= Strength of vortex.
Γ_i	= Strength of i^{th} vortex.
γ	= Argument of Elliptic integrals.
Δb_i	= Radial distance from center of source domain to i^{th} vortex.
Δb_s	= Radial distance from center of source domain 1 to source domain 2.
Δr_f	= Radial distance from center of field domain 3 to field domain 4.
Δx_f	= Axial distance from center of field domain 3 to field domain 4.
Δx_i	= Axial distance from center of source domain to i^{th} vortex.
Δx_s	= Axial distance from center of source domain 1 to source domain 2.
λ	= Argument of $Q_{\nu-\frac{1}{2}}^m$.
μ	= Index.
ν	= Index.
ρ	= Fluid density.
ψ	= Stream function.
ψ_i	= Stream function of i^{th} vortex.
ψ_{ei}	= Stream function at point i obtained using direct method.
ψ_p	= Stream function at point P .
ω	= Vorticity.

Contents

1	INTRODUCTION	11
1.1	Motivation for Work	11
1.2	General Methodology	11
2	SOURCE AND FIELD-POINT EXPANSIONS	15
2.1	Taylor Series Expansions	15
2.1.1	Expansion Within the Source Domain	15
2.1.2	Translation of the Source Domain	16
2.1.3	Expansion Within the Field Domain	16
2.1.4	Translation of the Field Domain	18
2.2	Partial Derivatives of F	19
2.2.1	Stream function Recursion Relationships	19
2.2.2	Evaluation of Basic Partial Derivatives	21
3	METHOD OF SOLUTION	25
3.1	Adaptive Mesh	25
3.1.1	Mesh Generation	25
3.1.2	Box Lists	27
3.2	Generation of Taylor Series Coefficients	31
3.2.1	A Coefficients	31
3.2.2	B Coefficients	31
3.3	Calculation of Field Variables	32
4	BENCHMARK TESTS	34
4.1	Error Associated with Truncation	35
4.2	Optimum Number of Vortices in a Box	38
5	SELF INDUCTION	44
5.1	Square Cores	45
5.2	Circular Cores	49
6	SUMMARY	52
	References	53

1 INTRODUCTION

1.1 Motivation for Work

Motivation for the present work results from a need to be able to perform fast yet accurate calculations of the axisymmetric flow field produced by large numbers of ring vortices. Such systems of ring vortices can be used, in turn, to model body surfaces and wakes in incompressible unsteady axisymmetric flow fields. Solutions for the flow over streamlined as well as bluff bodies can be simulated. Surface porosity and flexible geometries are also easily modeled using such methods. Bluff-body simulations present the greatest challenge since the wake structure and thus the positions of the ring vortices are not known apriori. The simulations for bluff bodies require that the wake structure evolve as a function of time, even for "steady flow" calculations. Large numbers of vortices ($100 < N_v < 10000$) are required for such simulations. Since the wake must be convected as part of the evolution process, the velocity perturbations from all vortices on all other vortices must be calculated at each time step. This requires an amount of work which is proportional to N_v^2 for each time step. If the amount of work can be reduced to being proportional to N_v or even $N_v \ln(N_v)$ then considerable CPU time will be saved in large simulations.

1.2 General Methodology

The genesis of this method comes from work done by Carrier, Greengard, and Rokhlin [1] in which an algorithm to calculate potential and force fields for a large system of charged particles was developed. In that work, it was shown that the asymptotic CPU time was of order N_p where N_p was the number of particles. This was contrasted to CPU times of order N_p^2 when traditional methods were used. This method takes advantage of source point and field point series expansions which enables one to make calculations for interactions between groups of particles which are in well separated spatial domains rather than having to consider interactions between every pair of particle. In the present work, the methodology concerning the use of the spatial domains in which vortex rings are located is essentially that used in reference [1] for charged particles. In the present work, the series expansion for the stream function of the vortex is used in place of the expansion for the potential of the charged particles of reference [1]. Likewise, the two velocity components produced at a point by the vortex rings are obtained explicitly from the stream-function expansion just as the two components of force produced by the charged particles are obtained from the potential function.

As indicated, the present method takes advantage of source point and field point series expansions which enables one to make calculations for interactions between groups of vortices which are in well separated spatial domains rather than having to consider interactions between each and every pair of vortices. For instance, consider the two domains D_S and D_F which are depicted in Figure 1. Domain D_S contains a set of vortices which are acting upon (producing perturbation velocities at) a set of points

Figure 1. Source and Field Point Domains

contained in domain D_F . In most cases, the set of points in domain D_F would represent the centers of a second set of vortices.

In order to implement this method, a Taylor series expansion for the stream function about the center of the source domain D_S is first obtained. This expansion includes all of the vortices which are inside D_S . The series converges for field points which are outside of D_S . In order to be somewhat conservative, the expansion about the center of the source domain is used only for field points which are separated by at least R_S from the boundary of the source domain. Thus, for field points which are within a circle of radius $2R_S$ with respect to the source domain center, the influence of vortices in domain D_S on the field point is computed directly. The center of the source domain may be shifted without actual evaluation of the Taylor series in order to allow one to form expansions efficiently for larger source domains which contain D_S .

A Taylor series expansion for the stream function can also be obtained about the center of the field domain D_F . The series converges for field points which are inside D_F . The expansion about the center of the field domain is used only for source points which are separated by at least R_F from the boundary of the field domain. Thus, for source points that are within a circle of radius $2R_F$ with respect to the field domain center, the influence of any vortex in domain D_S on a field point is computed directly. The center of the field point domain may also be shifted without actual evaluation of the Taylor series which allows one to form expansions for smaller field point domains contained in D_F .

In summary, the method allows one to calculate efficiently the influence of all of the vortices in D_S at all of the points in D_F when the two domains are "well separated" as

Figure 2. Adaptive Mesh (15 vortices per cell, maximum)

defined above. There is additional efficiency inherent in the method if one exploits the possibility of moving the centers of expansion without actually having to evaluate the Taylor series.

The method used to divide the flow field into domains is essentially that used in reference [1] for charged particles. The method requires that an adaptive mesh be generated that encloses all the vorticity in the flow. A box is first constructed which encloses all of the vortex rings in the flow. This box is subdivided into four equal boxes. If any of those four boxes contain more than a specified number of vortices, that particular box is subdivided into four more boxes. This process is continued until all of the boxes contain less than the specified number of vortices. Each box size is associated with a box level with the original box level being equal to 0. At each level, empty boxes are ignored. The ability to ignore regions of the flowfield in which vortex rings are not present is a significant feature of the method. Boxes which can be subdivided are parent boxes and those which cannot are childless boxes. Figure 2 provides an example of an adaptive mesh generated for a two-dimensional problem in which the maximum number of vortices per cell is 15.

A somewhat complicated procedure is used to define the separation condition between two boxes in the mesh and the influence of vortices in one box on the stream function in the other. In general, each box at each level has five possible relationships with other boxes in the mesh. A formal description of these "box lists" will be given in Section 3.1.2. In general, the five lists produce the following types of restrictions on the use of the Taylor series expansions:

- Direct calculations must be made. Taylor series expansions cannot be used.
- Source domain and field domain Taylor series expansions can be used.
- Source domain Taylor series expansions can be used, field domain expansions cannot.
- Field domain Taylor series expansions can be used, source domain expansions cannot.
- No calculations are necessary. Contributions from these distant boxes reside in the parent of this box.

2 SOURCE AND FIELD-POINT EXPANSIONS

As indicated previously, the basic solution method requires one to be able to write series expansions for source and field point domains. In addition, one must be able to translate the centers of the source and field point domains without actually having to evaluate the series expansions themselves. This section provides a description for obtaining the stream-function expansions and translations for axisymmetric vortex rings.

2.1 Taylor Series Expansions

Consider a single vortex ring located at an axial location $x = 0$ with a ring radius equal to b . The stream function at a field point (x, r) can be expressed according to Lamb [2] as:

$$\psi_i = \frac{1}{2} \Gamma_i b r \int_0^\infty e^{-kx} J_1(kr) J_1(kb) dk, \quad (1)$$

where Γ_i is the strength of the vortex ring and the J_1 functions are first order Bessel functions. The axial and radial velocity components are given by:

$$\begin{aligned} u &= \frac{1}{r} \frac{\partial \psi}{\partial r}, \\ v &= -\frac{1}{r} \frac{\partial \psi}{\partial x}. \end{aligned} \quad (2)$$

For the present purposes, let ψ_i be represented by:

$$\begin{aligned} \psi_i &= \frac{1}{2} \Gamma_i F(x, r, b), \\ F &\equiv b r \int_0^\infty e^{-kx} J_1(kr) J_1(kb) dk. \end{aligned} \quad (3)$$

2.1.1 Expansion Within the Source Domain

For a cluster of vortex rings as shown in Figure 3, a Taylor series expansion can be obtained which is valid at a field point P outside of a domain D_1 enclosing the cluster. The Taylor series for the stream function at point P resulting from this cluster can first be written symbolically as:

$$\psi = \sum_i \frac{1}{2} \Gamma_i F_i(x - \Delta x_i, r, b + \Delta b_i). \quad (4)$$

Formalizing the expansion yields a series in the partial derivatives of the function F . The coefficients A of this series are a function of the individual vortex strengths and their distance from the center of the expansion.

$$\begin{aligned} \psi &= \sum_{n=0}^{\infty} \sum_{m=0}^n A_{n-m,m} \frac{\partial^n F(x, r, b)}{\partial x^{n-m} \partial b^m}, \\ A_{n-m,m} &= \sum_i \frac{1}{2} \Gamma_i \frac{(-\Delta x_i)^{n-m} \Delta b_i^m}{(n-m)! m!}. \end{aligned} \quad (5)$$

Figure 3. Expansion About Center of Source Domain

The first summation in Equation 5 is, of course, calculated as a finite sum. Denoting the maximum value of n used in the calculation by N , later results show that accurate stream-function values are obtained for values of N on the order of 5 or 6.

2.1.2 Translation of the Source Domain

As indicated previously, it is advantageous to be able to move the center of the expansion given by Equation 5 such that the coefficients of the Taylor series in domain D_1 can be added to the coefficients of other domains contained in a larger domain D_2 . This situation is depicted in Figure 4. The Taylor series for the stream function at point P resulting from this translation can first be written symbolically as:

$$\psi = \sum_{n=0}^{\infty} \sum_{m=0}^n A_{n-m,m} \frac{\partial^n F(x - \Delta x_s, r, b + \Delta b_s)}{\partial x^{n-m} \partial b^m} \quad (6)$$

Formalizing the expansion again yields a series in the partial derivatives of the function F . The coefficients A' of this series are a function of the old coefficients A and the coordinates between the two centers of expansion.

$$\psi = \sum_{n=0}^{\infty} \sum_{m=0}^n A'_{n-m,m} \frac{\partial^n F(x, r, b)}{\partial x^{n-m} \partial b^m} \quad (7)$$

$$A'_{n-m,m} = \sum_{q=0}^m \sum_{p=q}^{q+n-m} A_{n-m-p+q,m-q} \frac{(-\Delta x_s)^{p-q} \Delta b_s^q}{(p-q)! q!}.$$

2.1.3 Expansion Within the Field Domain

A Taylor series expansion will now be obtained about the center of a field point domain D_3 due to the influence of vortices in D_2 . Referring to Figure 5 it can be seen

Figure 4. Translation of Source Domain Center

that this expansion will be valid at field points inside D_3 if D_2 and D_3 are well separated.

Figure 5. Expansion About Center of Field Domain

The Taylor series for the stream function at point P resulting from the expansion about the center of D_3 can be written symbolically as:

$$\psi = \sum_{n=0}^{\infty} \sum_{m=0}^n A'_{n-m,m} \frac{\partial^n F(x + x_f, r + r_f, b)}{\partial x^{n-m} \partial b^m} \quad (8)$$

Formalizing the expansion yields a power series in r_f and x_f . The coefficients B of this series are a function of the partial derivatives of the function F and the old coefficients A' .

$$\psi = \sum_{n=0}^{\infty} \sum_{m=0}^n B_{n-m,m} x_f^{n-m} r_f^m, \quad (9)$$

$$B_{n-m,m} = \sum_{p=0}^L \sum_{q=0}^p \frac{A'_{p-q,q}}{(n-m)! m!} \frac{\partial^{n+p} F(x, r, b)}{\partial x^{n-m+p-q} \partial r^m \partial b^q}.$$

In Equation 9, the summation limit L in the second equation will be equal to the maximum value of r less n itself (ie. $L = N - n$).

2.1.4 Translation of the Field Domain

The last Taylor series expansion involves the translation of the center of the field point domain. As Figure 6 depicts, the expansion for the domain D_3 can be used for any domain D_4 which is inside of D_3 by performing a simple translation of the expansion

Figure 6. Translation of Field Domain Center

center. The symbolic Taylor series expansion for the stream function at point P about the center of D_4 is:

$$\psi = \sum_{n=0}^{\infty} \sum_{m=0}^n B_{n-m,m} (x_f + \Delta x_f)^{n-m} (r_f + \Delta r_f)^m. \quad (10)$$

Formalizing the expansion again yields a power series in r_f and x_f . The coefficients B' of this series are a function of Δr_f and Δx_f and the old coefficients B .

$$\psi = \sum_{n=0}^{\infty} \sum_{m=0}^n B'_{n-r,m} x_f^{n-m} r_f^m, \quad (11)$$

$$B'_{n-m,m} = \sum_{p=0}^L \sum_{q=0}^p B_{n-m+p-q,m+q} \frac{\Delta x_f^{p-q} \Delta r_f^q (n-m+p-q)! (m+q)!}{(p-q)! q! (n-m)! m!}.$$

As before, the summation limit L in the second equation of Equation 11 will be equal to $N - n$.

2.2 Partial Derivatives of F

There are several methods which one could use to obtain the various partial derivatives of the function F . One method would be to simply differentiate Equation 3 with respect to x , r , and or b the required number of times. The method used in the present work makes use of the partial differential equation for the stream function, which is satisfied by Equation 3, to develop relationships between various partial derivatives of F . This eliminates the need to directly calculate many of the partials since the equation for the stream function can be used to obtain recursion relationships.

2.2.1 Stream function Recursion Relationships

The axisymmetric stream-function equation for irrotational flow is given by Lamb [2] as:

$$\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial r^2} = \frac{1}{r} \frac{\partial \psi}{\partial r}. \quad (12)$$

Equation 12 used in conjunction with Equation 3 yields two equations in F if one takes advantage of the symmetry between r and b :

$$\begin{aligned} \frac{\partial^2 F}{\partial x^2} + \frac{\partial^2 F}{\partial r^2} &= \frac{1}{r} \frac{\partial F}{\partial r}, \\ \frac{\partial^2 F}{\partial x^2} + \frac{\partial^2 F}{\partial b^2} &= \frac{1}{b} \frac{\partial F}{\partial b}. \end{aligned} \quad (13)$$

Equations represented by Equation 13 can be differentiated to obtain the following relationships for the $(l + m + n)^{th}$ order partials of F :

$$\begin{aligned} \frac{\partial^{l+m+n} F}{\partial x^l \partial r^m \partial b^n} &= -\frac{\partial^{l+m+n} F}{\partial x^{l+2} \partial r^{m-2} \partial b^n} + \sum_{j=0}^{m-2} \frac{(-1)^j}{r^{j+1}} \frac{(m-2)!}{(m-2-j)!} \frac{\partial^{l+m+n-j-1} F}{\partial x^l \partial r^{m-j-1} \partial b^n}, \\ \frac{\partial^{l+m+n} F}{\partial x^l \partial r^m \partial b^n} &= -\frac{\partial^{l+m+n} F}{\partial x^{l+2} \partial r^m \partial b^{n-2}} + \sum_{j=0}^{n-2} \frac{(-1)^j}{b^{j+1}} \frac{(n-2)!}{(n-2-j)!} \frac{\partial^{l+m+n-j-1} F}{\partial x^l \partial r^m \partial b^{n-j-1}}. \end{aligned} \quad (14)$$

In the first equation represented by Equation 14, $m = 0, 1$ is forbidden as is $n = 0, 1$ in the second equation. Therefore, for each value of $p = l + m + n$ the following four partial

derivatives cannot be obtained from Equation 14:

$$\frac{\partial^p F}{\partial x^p}, \quad \frac{\partial^p F}{\partial x^{p-1} \partial r}, \quad \frac{\partial^p F}{\partial x^{p-1} \partial b}, \quad \frac{\partial^p F}{\partial x^{p-2} \partial r \partial b}. \quad (15)$$

These four "basic" partial derivatives must be calculated directly. For each value of p there are $\frac{1}{2}(p+1)(p+2)$ partial derivatives of order p . The equations indicated by Equation 14 represent $\frac{1}{2}(p+1)(p+2) - 4$ independent equations for each value of p . The first of the two equations given in Equation 14 can be used to obtain $\frac{1}{2}p(p-1)$ partials. The second of the two equations given in Equation 14 can be used to obtain another set of $\frac{1}{2}p(p-1)$ partials. The two sets are not entirely independent. A recursive solution can be obtained for each value of p by first assuming that the partials in Equation 15 are already calculated. The calculation starts by using the first equation indicated by Equation 14 with the highest derivative of x . The order of r is systematically reduced while the order of b is increased. If the first equation becomes invalid ($m = 0, 1$) then the second equation of Equation 14 is used.

As an example of the method to obtain the mixed partials of F , consider the case for $p = 5$. It is assumed that all of the mixed partials for $p = 0, 1, 2, 3, 4$ have already been obtained. The following four "basic" fifth order partials are computed directly using a method which will be discussed in the next section:

$$\frac{\partial^5 F}{\partial x^5}, \quad \frac{\partial^5 F}{\partial x^4 \partial r}, \quad \frac{\partial^5 F}{\partial x^4 \partial b}, \quad \frac{\partial^5 F}{\partial x^3 \partial r \partial b}. \quad (16)$$

From the two differential equations represented by Equation 14, it can be seen that a fifth order partial can be determined from a previously calculated fifth order partial in addition to a sum of previously calculated lower order partials. Symbolizing the mixed partials of F by:

$$F_{l,m,n} \equiv \frac{\partial^{l+m+n} F}{\partial x^l \partial r^m \partial b^n}, \quad (17)$$

then Equation 14 can be represented by two equations of the form:

$$\begin{aligned} F_{l,m,n} &= -F_{l+2,m-2,n} + \dots \text{lower order terms}, \quad (a) \\ F_{l,m,n} &= -F_{l+2,m,n-2} + \dots \text{lower order terms}. \quad (b) \end{aligned} \quad (18)$$

Both equations can be represented by:

$$F_{l,m,n} = -F_{i,j,k} + \dots \text{lower order terms}. \quad (19)$$

Table 1 shows the sequence by which all of the fifth order partials are obtained. The first column labeled M where $M=(a), (b)$, or (d) indicates whether the partial was calculated from Equation 18 (a) or (b) or directly as indicated by (d). The second column represents the l, m, n partial of F which is being computed while the third column represents the i, j, k partial of F which is being used in the computation.

Table 1. Recursion Sequence for $p = 5$.

M	l, m, n	i, j, k
(d)	5,0,0	—
(d)	4,1,0	—
(d)	4,0,1	—
(d)	3,1,1	—
(a)	3,2,0	5,0,0
(b)	3,0,2	5,0,0
(a)	2,3,0	4,1,0
(a)	2,2,1	4,0,1
(b)	2,1,2	4,1,0
(b)	2,0,3	4,0,1
(a)	1,4,0	3,2,0
(a)	1,3,1	3,1,1
(a)	1,2,2	3,0,2
(b)	1,1,3	3,1,1
(b)	1,0,4	3,0,2
(a)	0,5,0	2,3,0
(a)	0,4,1	2,2,1
(a)	0,3,2	2,1,2
(a)	0,2,3	2,0,3
(b)	0,1,4	2,1,2
(b)	0,0,5	2,0,3

2.2.2 Evaluation of Basic Partial Derivatives

From the definition of F in Equation 3, it can be seen that if one takes the n^{th} derivative with respect to x , k^n is added to the integrand. Successive differentiations of the Bessel functions can be made to yield linear combinations of J_0 and J_1 by repeated application of the following differentiation formulas:

$$\begin{aligned}\frac{dJ_0(z)}{dz} &= -J_1(z), \\ \frac{dJ_1(z)}{dz} &= J_0(z) - \frac{1}{z}J_1(z).\end{aligned}\tag{20}$$

In other words, repeated differentiation with respect to r or b will produce linear combinations of $J_0(kr)$, $J_1(kr)$, $J_0(kb)$, and $J_1(kb)$. Therefore, the various partial derivatives of F can be expressed in terms of linear combinations of the following nondimensional integrals:

$$I_{\mu\nu}^n = (-1)^{n+\mu+\nu} b^n r \int_0^\infty e^{-kx} k^n J_\mu(kb) J_\nu(kr) dk, \tag{21}$$

where μ and ν can be equal to 0 or 1. Thus, for a given value of n only four integrals occur. These integrals are directly related to the four "basic" partial derivatives of F

discussed in the previous section. They can be obtained by taking the indicated partial derivatives of F , as defined in Equation 3, using the differentiation formulas given in Equation 20.

$$\begin{aligned}
\frac{\partial^n F}{\partial x^n} &= (-1)^n br \int_0^\infty e^{-kx} k^n J_1(kb) J_1(kr) dk = b^{1-n} I_{11}^n, \\
\frac{\partial^n F}{\partial x^{n-1} \partial b} &= (-1)^{n+1} br \int_0^\infty e^{-kx} k^n J_0(kb) J_1(kr) dk = b^{1-n} I_{01}^n, \\
\frac{\partial^n F}{\partial x^{n-1} \partial r} &= (-1)^{n+1} br \int_0^\infty e^{-kx} k^n J_1(kb) J_0(kr) dk = b^{1-n} I_{10}^n, \\
\frac{\partial^n F}{\partial x^{n-2} \partial r \partial b} &= (-1)^n br \int_0^\infty e^{-kx} k^n J_0(kb) J_0(kr) dk = b^{1-n} I_{00}^n.
\end{aligned} \tag{22}$$

Furthermore, if the variables (F, x, r) are nondimensionalized by b such that:

$$(F^+, x^+, r^+) \equiv \left(\frac{F}{b}, \frac{x}{b}, \frac{r}{b} \right), \tag{23}$$

and the partial derivatives with respect to (x^+, r^+, b^+) are defined as:

$$\left(\frac{\partial^n}{\partial x^{+n}}, \frac{\partial^n}{\partial r^{+n}}, \frac{\partial^n}{\partial b^{+n}} \right) \equiv \left(b^n \frac{\partial^n}{\partial x^n}, b^n \frac{\partial^n}{\partial r^n}, b^n \frac{\partial^n}{\partial b^n} \right), \tag{24}$$

then the nondimensional "basic" partial derivatives of F are equal to the integrals I_{11}^n , I_{01}^n , I_{10}^n , and I_{00}^n according to:

$$\begin{aligned}
\frac{\partial^n F^+}{\partial x^{+n}} &= I_{11}^n, \\
\frac{\partial^n F^+}{\partial x^{+n-1} \partial b^+} &= I_{01}^n, \\
\frac{\partial^n F^+}{\partial x^{+n-1} \partial r^+} &= I_{10}^n, \\
\frac{\partial^n F^+}{\partial x^{+n-2} \partial r^+ \partial b^+} &= I_{00}^n.
\end{aligned} \tag{25}$$

The four integrals in Equation 22 can be evaluated in terms of associated Legendre functions of the second kind. These functions can be denoted by $Q_{\nu-\frac{1}{2}}^m$ where for the present purposes, m and ν are integers. In order to write the four integrals in terms of associated Legendre functions, integration formula 6.6123 in reference [3] is first used. This equation can be written as:

$$\int_0^\infty e^{-kx} J_\nu(kb) J_\nu(kr) dk = \frac{(br)^{-\frac{1}{2}}}{\pi} Q_{\nu-\frac{1}{2}} \left(\frac{x^2 + r^2 + b^2}{2br} \right). \tag{26}$$

From Equations 21 and 26, it can be shown that:

$$\begin{aligned} I_{00}^0 &= \frac{1}{\pi} \left(\frac{r}{b} \right)^{\frac{1}{2}} Q_{-\frac{1}{2}}(\lambda), \\ I_{11}^0 &= \frac{1}{\pi} \left(\frac{r}{b} \right)^{\frac{1}{2}} Q_{\frac{1}{2}}(\lambda), \\ \lambda &= \frac{x^2 + r^2 + b^2}{2br}. \end{aligned} \quad (27)$$

In order to obtain I_{00}^n and I_{11}^n , the first two expressions in Equation 27 are differentiated n times. The integrals I_{10}^n and I_{01}^n can be obtained from I_{00}^{n-1} or I_{11}^{n-1} by manipulation of terms in Equation 22 and by use of the differentiation formulas of Equation 20. The results can be expressed by:

$$\begin{aligned} I_{00}^n &= b^n \frac{\partial^n I_{00}}{\partial x^n} = \frac{b^n}{\pi} \left(\frac{r}{b} \right)^{\frac{1}{2}} \frac{\partial^n Q_{-\frac{1}{2}}(\lambda)}{\partial x^n}, \\ I_{11}^n &= b^n \frac{\partial^n I_{11}}{\partial x^n} = \frac{b^n}{\pi} \left(\frac{r}{b} \right)^{\frac{1}{2}} \frac{\partial^n Q_{\frac{1}{2}}(\lambda)}{\partial x^n}, \\ I_{10}^n &= -b^n r \frac{\partial}{\partial b} \left(\frac{I_{00}^{n-1}}{b^{n-1} r} \right) = b^{n-1} \frac{\partial}{\partial r} \left(\frac{I_{11}^{n-1}}{b^{n-2}} \right), \\ I_{01}^n &= -b^n r \frac{\partial}{\partial r} \left(\frac{I_{00}^{n-1}}{b^{n-1} r} \right) = b^{n-1} \frac{\partial}{\partial b} \left(\frac{I_{11}^{n-1}}{b^{n-2}} \right). \end{aligned} \quad (28)$$

The n^{th} partial derivatives with respect to x of $Q_{-\frac{1}{2}}(\lambda)$ and $Q_{\frac{1}{2}}(\lambda)$ can be expressed in terms of the ordinary derivatives with respect to λ by the following sums:

$$\begin{aligned} \frac{\partial^n Q_{-\frac{1}{2}}(\lambda)}{\partial x^n} &= \sum_{m \geq \frac{n}{2}}^n \frac{n! 2^{m-n}}{(2m-n)!(n-m)!} \left(\frac{\partial \lambda}{\partial x} \right)^{2m-n} \left(\frac{\partial^2 \lambda}{\partial x^2} \right)^{n-m} \frac{d^m Q_{-\frac{1}{2}}(\lambda)}{d\lambda^m}, \\ \frac{\partial^n Q_{\frac{1}{2}}(\lambda)}{\partial x^n} &= \sum_{m \geq \frac{n}{2}}^n \frac{n! 2^{m-n}}{(2m-n)!(n-m)!} \left(\frac{\partial \lambda}{\partial x} \right)^{2m-n} \left(\frac{\partial^2 \lambda}{\partial x^2} \right)^{n-m} \frac{d^m Q_{\frac{1}{2}}(\lambda)}{d\lambda^m}. \end{aligned} \quad (29)$$

Useful partial derivatives of λ with respect to x , r , and b are given as:

$$\begin{aligned} \frac{\partial \lambda}{\partial x} &= \frac{x}{br}, \quad \frac{\partial^2 \lambda}{\partial x^2} = \frac{1}{br}, \\ \frac{\partial \lambda}{\partial r} &= \frac{r^2 - x^2 - b^2}{2r^2 b}, \\ \frac{\partial \lambda}{\partial b} &= \frac{b^2 - x^2 - r^2}{2b^2 r}. \end{aligned} \quad (30)$$

The ordinary derivatives of $Q_{-\frac{1}{2}}(\lambda)$ and $Q_{\frac{1}{2}}(\lambda)$ in Equation 29 can be obtained in terms of higher order associated Legendre functions as per equation 8.7524 in reference [3].

$$\begin{aligned} \frac{d^m Q_{-\frac{1}{2}}(\lambda)}{d\lambda^m} &= (\lambda^2 - 1)^{-\frac{m}{2}} Q_{-\frac{1}{2}}^m, \\ \frac{d^m Q_{\frac{1}{2}}(\lambda)}{d\lambda^m} &= (\lambda^2 - 1)^{-\frac{m}{2}} Q_{\frac{1}{2}}^m. \end{aligned} \quad (31)$$

The terms $Q_{-\frac{1}{2}}^m$ and $Q_{\frac{1}{2}}^m$ are understood to be equivalent to $Q_{-\frac{1}{2}}^m(\lambda)$ and $Q_{\frac{1}{2}}^m(\lambda)$ respectively. From Equations 28-31, one obtains expressions for $I_{\mu\nu}^n$ given by:

$$\begin{aligned}
I_{11}^n &= \sum_{m \geq \frac{n}{2}}^n C_{n,m} \frac{Q_{\frac{1}{2}}^m}{(\lambda^2 - 1)^{\frac{m}{2}}}, \\
I_{00}^n &= \sum_{m \geq \frac{n}{2}}^n C_{n,m} \frac{Q_{-\frac{1}{2}}^m}{(\lambda^2 - 1)^{\frac{m}{2}}}, \\
I_{10}^n &= - \sum_{m \geq \frac{n-1}{2}}^{n-1} C_{n-1,m} \left[\left(\frac{1 - x^{+2} - r^{+2}}{2r^{+2}} \right) \frac{Q_{-\frac{1}{2}}^{m+1}}{(\lambda^2 - 1)^{\frac{m+1}{2}}} - \frac{(m + \frac{1}{2})Q_{-\frac{1}{2}}^m}{(\lambda^2 - 1)^{\frac{m}{2}}} \right], \\
I_{01}^n &= - \sum_{m \geq \frac{n-1}{2}}^{n-1} C_{n-1,m} \left[\left(\frac{r^{+2} - x^{+2} - 1}{2r^{+2}} \right) \frac{Q_{-\frac{1}{2}}^{m+1}}{(\lambda^2 - 1)^{\frac{m+1}{2}}} - \frac{(m + \frac{1}{2})Q_{-\frac{1}{2}}^m}{r^{+}(\lambda^2 - 1)^{\frac{m}{2}}} \right], \\
C_{n,m} &= \frac{1}{\pi} \frac{n! 2^{m-n} x^{+(2m-n)} r^{+(\frac{1}{2}-m)}}{(2m-n)!(n-m)!}.
\end{aligned} \tag{32}$$

The associated Legendre functions $Q_{-\frac{1}{2}}^m$ and $Q_{\frac{1}{2}}^m$ are calculated using a set of recursive relationships given by equation 8.7323 in reference [3].

$$Q_{\nu}^{n+2} = -\frac{2(n+1)\lambda}{(\lambda^2 - 1)^{\frac{1}{2}}} Q_{\nu}^{n+1} + (\nu - n)(\nu + n + 1) Q_{\nu}^n, \quad n = 0, 1, 2, 3, \dots, \tag{33}$$

where ν is equal to $+\frac{1}{2}$ or $-\frac{1}{2}$ and where λ is the argument of Q_{ν}^n . As can be seen from Equation 33, starting values of Q_{ν} and Q_{ν}^1 are required. Q_{ν} is understood to be equivalent to Q_{ν}^0 . The values for $Q_{\frac{1}{2}}$ and $Q_{-\frac{1}{2}}$ can be obtained in terms of the elliptic integrals E and K of the first and second kind, respectively.

$$\begin{aligned}
Q_{-\frac{1}{2}} &= \gamma K(\gamma), \\
Q_{\frac{1}{2}} &= \lambda \gamma K(\gamma) - \frac{2}{\gamma} E(\gamma), \\
\gamma &\equiv \left(\frac{2}{\lambda + 1} \right)^{\frac{1}{2}}.
\end{aligned} \tag{34}$$

Values for $Q_{\frac{1}{2}}^1$ and $Q_{-\frac{1}{2}}^1$ can be obtained from equations 8.7344 and 8.7341 of reference [3] respectively.

$$\begin{aligned}
Q_{\frac{1}{2}}^1 &= \lambda Q_{-\frac{1}{2}}^1 + \frac{1}{2}(\lambda^2 - 1)^{\frac{1}{2}} Q_{-\frac{1}{2}}, \\
Q_{-\frac{1}{2}}^1 &= \frac{1}{2} \frac{(Q_{\frac{1}{2}} - \lambda Q_{-\frac{1}{2}})}{(\lambda^2 - 1)^{\frac{1}{2}}} = -\frac{E}{\gamma(\lambda^2 - 1)^{\frac{1}{2}}}.
\end{aligned} \tag{35}$$

The elliptic integrals were calculated using a polynomial approximation given in reference [4]. Calculated values agreed to 6 places when compared to the more exact algorithm due to Bulirsch [5, 6]. The associated Legendre functions generated by this method have been checked against the tables of Lowan [7] and found to be in excellent agreement.

3 METHOD OF SOLUTION

The present work is implemented by the computer code Ring SOLVer (RSOLV) developed at Sandia National Laboratories. This code requires that one specify a distribution of vortex rings by their axial locations, ring diameters and strengths and place such information in an input file. The code also requires one to specify the domain of interest, the maximum number of vortices in a box N_c , and the order N of the Taylor series to be used. RSOLV output consists of an output file with values of the stream function, axial velocity, and radial velocity at all of the vortex locations. The following describes the methodology used in the RSOLV code.

3.1 Adaptive Mesh

As mentioned in the introduction, the method used to divide the flow field into domains is essentially that used in reference [1] for a field of charged particles. Some minor differences in logic associated with the mesh generator of RSOLV and the mesh generator of reference [1] may exist. The description of mesh generation in the following section applies in the strictest sense to the present work. Similar things may be said about box lists in that while they are defined as in reference [1], some minor differences in logic associated with their use in RSOLV may exist.

3.1.1 Mesh Generation

The method requires that an adaptive mesh be generated that encloses all the N_v vortices of interest in the flow. A square box with dimensions S_0 is first constructed which encloses all of the chosen vortex rings. If there happen to be vortices outside of the chosen box or even on the box edges, they are discarded and an appropriate warning is issued. This box is subdivided into four equal boxes. If any of those four boxes contain more than a specified number of vortices N_c , that particular box is subdivided into four more boxes. Vortices residing on mesh lines are associated with the box whose center has a more positive x or r value than the x or r value defining the mesh line. This process is continued until all of the boxes contain less than the specified number of vortices. Each box size is associated with a box level l with the original box level being equal to 0. Therefore at a particular level l , the box dimension S_l is given by:

$$S_l = \frac{S_0}{2^l}. \quad (36)$$

At each level, empty boxes are ignored. The ability to ignore regions of the flowfield in which vortex rings are not present is a significant feature of the method. Boxes which can be subdivided are parent boxes and those which cannot are childless boxes. A list of vortices residing in each parent or childless box at each level is maintained. Each vortex is identified by a unique label (number). Figures 2 and 7 provide examples of the adaptive mesh generation capability of RSOLV.

(a) $N_v = 973$, $N_e = 20$

(b) $N_v = 200$, $N_e = 10$

Figure 7. Mesh Generation Examples

The mesh generator in the RSOLV code allows a maximum level of $l = 10$. The box dimension S_{10} at that level will be $1/1024^{th}$ of the original box dimension. In some instances there will be more than the specified number of vortices N_c in some boxes at $l = 10$. In such cases, a warning is issued and the code proceeds leaving the extra vortices in the boxes. It should be pointed out that the solution resolution is not limited by the maximum level l since interactions between vortices within the smallest box are still considered. It may also be possible that the maximum number of parent or childless boxes at a given level may be exceeded. This number is set at 1000 in the RSOLV code. If this number is exceeded a warning is issued along with a request to increase the value of N_c .

3.1.2 Box Lists

The purpose of defining a box list for each box at each level is to define the separation condition between the selected box and all other boxes in the mesh. This in turn allows one to efficiently calculate the influence of vortices in one box on the stream function in the other. In general, each box at each level has five possible relationships with other boxes in the mesh. The five lists define restrictions on the use of the Taylor series expansions between domains defined by the two boxes.

In order to formalize the box list definitions, let NBOXI refer to the mesh box for which a box list is being developed. As mentioned previously, parent boxes are boxes which can be subdivided into four smaller boxes, while childless boxes cannot. Thus, parent boxes contain more than N_c vortices. Empty boxes contain no vortices, while childless boxes contain from 1 to N_c vortices. A colleague box of NBOXI is a box which is adjacent to NBOXI and which has the same size (level) as NBOXI. There are at most eight colleague boxes associated with NBOXI. Definitions for the five boxlists are as follows:

- Box List 1- In order for NBOXI to have any list 1 boxes, NBOXI must itself be childless. If NBOXI is a childless box, then list 1 consists of NBOXI and all childless boxes at all levels which are adjacent to NBOXI. This list defines boxes which are not separated from each other and which cannot be subdivided. Direct calculations must be made. Taylor series expansions cannot be used.
- Box List 2- List 2 boxes of NBOXI are boxes which are children of the colleagues of NBOXI's parent that are well separated from NBOXI. NBOXI and its list 2 boxes can be either parent or childless boxes. NBOXI and its list 2 boxes will be the same size (level) and will be separated by at least the dimension of one of their sides. Both source domain and field domain Taylor series expansions can be used.
- Box List 3- In order for NBOXI to have any list 3 boxes, NBOXI must itself be childless. List 3 boxes can be either parent or childless boxes. List 3 boxes are descendants of the colleagues of NBOXI. The parent of the list 3 box must be

adjacent to NBOXI but the list 3 box must not itself be adjacent to NBOXI. List 3 boxes will always be smaller than NBOXI. NBOXI will be separated from the list 3 box by one box which is the same size as the list 3 box. Source domain Taylor series expansions can be used, field domain expansions cannot.

- Box List 4- NBOXI can be either a parent or childless box. List 4 boxes must be childless. NBOXI is a descendant of the colleagues of any list 4 box. The parent of NBOXI must be adjacent to the list 4 box but NBOXI must not itself be adjacent to the list 4 box. List 4 boxes will always be larger than NBOXI. NBOXI will be separated from the list 4 box by one box which is the same size as NBOXI. Field domain Taylor series expansions can be used, source domain expansions cannot.
- Box List 5- List 5 boxes consist of all boxes that are well separated from NBOXI's parent. No calculations are necessary. Contributions from these distant boxes reside in the parent of NBOXI.

In order to demonstrate how the set of box lists brings information into the field box NBOXI, consider the example given in Figures 8 and 9. NBOXI is the cross-hatched box shown in Figure 8 (a). Assume that the indicated mesh has been generated for a set of vortices and that there are no empty boxes. Therefore, one would expect every box to contribute to the stream function in NBOXI. From Figure 8 (a), it can be seen that there are contributions from NBOXI's own box list and that this list contains box lists 1-4. As indicated previously, box list 5 does not contribute directly and is thus not even indicated in Figures 8 and 9. In Figure 8 (b), there are several list 2 and 4 boxes that contribute to NBOXI's parent. Since this is a parent box, it does not have any list 1 or 3 boxes associated with it. The information from NBOXI's parent is fed into NBOXI by translation of the field domain center of the parent of NBOXI to the center of NBOXI. Information from NBOXI's grandparent's list boxes (see Figure 9 (c)) were previously fed into NBOXI's parent in the same fashion. As indicated in Figure 9 (d), NBOXI's great grandparent has a null box list and therefore does not contribute in this case. It can be noted that every box in the mesh has contributed to NBOXI. In general, it can be seen that the box list concept allows information to be passed down through successive levels in an orderly and efficient fashion.

(a) Contribution from NBOXI's own box list.

(b) Contribution from NBOXI's parent's box list.

Figure 8. Box List Example

(c) Contribution from NBOXI's grandparent's box list.

(d) Contribution from NBOXI's great grandparent's box list.

Figure 9. Box List Example (continued)

3.2 Generation of Taylor Series Coefficients

As mentioned previously, the Taylor series can be used only in cases where the source list box associated with a field box NBOXI is a list 2, list 3, or list 4 box. Information from the A coefficients is used in cases where NBOXI has list 2 or list 3 boxes, while information from the B coefficients is used in cases where NBOXI has list 2 or 4 boxes.

3.2.1 A Coefficients

The $A_{n-m,m}$ or $A'_{n-m,m}$ coefficients defined in Equations 5 and 7 are used to calculate the $B_{n-m,m}$ coefficient when the source box is a list 2 box. Also, for list 3 boxes, the A coefficients are used directly in the calculation of field variables. The A coefficients are calculated by first using Equation 5 to generate values for $A_{n-m,m}$ for all childless boxes. The centers of expansion for the smallest (highest level) child boxes are then shifted to their parent's centers using Equation 7. The contributions from the child boxes to the parent box are added together producing a set of A coefficients for the parent box. This process is repeated for decreasing levels (larger boxes) until the A coefficients are calculated for all parent and childless boxes at all levels. This process may be thought of as the process by which information concerning source domain expansions is promulgated from smaller domains into larger domains.

3.2.2 B Coefficients

The B coefficients are only used for list 2 and list 4 boxes. For list 2 boxes, the $B_{n-m,m}$ coefficients can be calculated from Equation 9. For list 4 boxes, the contribution from each vortex must be considered separately since the source domain is not well separated from the center of the field box NBOXI. The following equation is used to calculate the contribution from a group of I vortices in a list 4 box to the B coefficients in NBOXI:

$$B_{n-m,m} = \sum^I \frac{1}{2} \frac{\Gamma_i}{(n-m)!m!} \frac{\partial^n F(x - x_i, r, b_i)}{\partial x^{n-m} \partial r^m}. \quad (37)$$

Here, Γ_i is the strength of the i^{th} vortex. The center of the field box is at x, r while the i^{th} vortex center is at x_i, b_i . Therefore, the B coefficients are calculated by first using Equation 9 for list 2 boxes and Equation 37 for list 4 boxes to generate values for $B_{n-m,m}$ for all boxes. Next, the centers of expansion for the largest (lowest level) parent boxes which have list 2 or list 4 boxes associated with them are shifted to their children's centers using Equation 11. The contribution from the parent box to the child box is added together with the list 2 and list 4 contributions associated with the child box itself. This produces a set of B coefficients for the child box. This process is repeated for increasing levels (smaller boxes) until the B coefficients are calculated for all parent and childless boxes at all levels. This process may be thought of as the process by which information concerning field domain expansions is promulgated from larger domains into smaller domains.

3.3 Calculation of Field Variables

The field variables ψ_p , u_p , and v_p at a point p are calculated according to the type of list box containing the source information. For list 1 boxes, the field variable at point p must be calculated by considering the influence of each individual vortex in the source box on the point p . For list 2 and 4 source boxes, a Taylor series about the center of the field box is used. This series uses the B coefficients. For list 3 source boxes, a series containing the A coefficients is used. Although the source box information is contained in the A coefficients, the partial derivatives of F must be recalculated for each field point.

The field variables ψ_p , u_p , and v_p at a point p for list 1 source boxes are calculated from:

$$\begin{aligned}\psi_p &= \sum_{i=1}^I \frac{1}{2} \Gamma_i F(x_p - x_i, r_p, b_i), \\ u_p &= \frac{1}{r_p} \sum_{i=1}^I \frac{1}{2} \Gamma_i \frac{\partial F(x_p - x_i, r_p, b_i)}{\partial r}, \\ v_p &= -\frac{1}{r_p} \sum_{i=1}^I \frac{1}{2} \Gamma_i \frac{\partial F(x_p - x_i, r_p, b_i)}{\partial x}.\end{aligned}\tag{38}$$

Here, the field point location p is given by x_p and r_p while the location of the i^{th} vortex in the list 1 box is given by x_i and b_i . The strength of the i^{th} vortex is Γ_i while the total number of vortices in the list 1 box is I .

The field variables at a point p for list 2 and list 4 source boxes are calculated by:

$$\begin{aligned}\psi_p &= \sum_{n=0}^N \sum_{m=0}^n B_{n-m,m} x_f^{n-m} r_f^m, \\ u_p &= \frac{1}{r_p} \sum_{n=1}^N \sum_{m=1}^n m B_{n-m,m} x_f^{n-m} r_f^{m-1}, \\ v_p &= -\frac{1}{r_p} \sum_{n=1}^N \sum_{m=0}^{n-1} (n-m) B_{n-m,m} x_f^{n-m-1} r_f^m.\end{aligned}\tag{39}$$

Here, the distance from the field box center to the field point p is given by the axial and radial coordinates x_f and r_f . It should also be noted that the $B_{n-m,m}$ terms are the B coefficients associated with the field box containing point p which have been obtained according to the method of the previous section.

The field variables at a point p for list 3 source boxes are calculated by:

$$\begin{aligned}\psi_p &= \sum_{n=0}^N \sum_{m=0}^n A_{n-m,m} \frac{\partial^n F(x_p - x_i, r_p, b_i)}{\partial x^{n-m} \partial b^m}, \\ u_p &= \frac{1}{r_p} \sum_{n=0}^N \sum_{m=0}^n A_{n-m,m} \frac{\partial^{n+1} F(x_p - x_i, r_p, b_i)}{\partial x^{n-m} \partial r \partial b^m},\end{aligned}\tag{40}$$

$$v_p = -\frac{1}{r_p} \sum_{n=0}^N \sum_{m=0}^n A_{n-m,m} \frac{\partial^{n+1} F(x_p - x_i, r_p, b_i)}{\partial x^{n-m+1} \partial b^m}.$$

Here, the field point location p is given by x_p and r_p while the location of the i^{th} list 3 source box is given by x_i and b_i . The $A_{n-m,m}$ terms are the A coefficients associated with the i^{th} list 3 source box.

The contribution to a field variable at point p from list 1 and list 3 boxes is obtained by adding the contributions from all list 1 and list 3 boxes together. The contribution from all list 2 and 4 boxes is inherent in the B coefficients which are associated with the field box itself. This contribution is added to the contributions from all list 1 and list 3 source boxes. This may be illustrated by the following equation:

$$fv_{total} = \sum_{list\ 1} fv_1 + \sum_{list\ 3} fv_3 + fv_{24}, \quad (41)$$

where fv is any field variable.

4 BENCHMARK TESTS

The important dependent variables in the bench mark tests are the CPU run times and the truncation errors associated with the calculation of field variables. The independent variables are the number of vortices in the field, the distribution of vortices in the field, the number of terms used in the Taylor series, and the maximum number of vortices allowed in any childless box in the mesh. Based on the limited parametric study described below, the required number of terms in the Taylor series was obtained based on the truncation errors while the optimum maximum number of vortices in any childless box was obtained based on the CPU run times. The limited parametric study was conducted in the following fashion:

- A uniform distribution of 1024 vortices was used to examine the error associated with five different Taylor series truncation points (values of N). Runs were made for three different values of the maximum allowable number of vortices in any childless box N_c . Based on these 15 runs, a Taylor series truncation point was chosen based on a trade off between truncation error and CPU time t_{cpu} .
- Random distributions with 50, 100, 200, 300, 500, 1000, and 2000 vortices were used to determine the optimum maximum number of vortices N_c allowed in any childless box. The optimum range of values for N_c was obtained when t_{cpu} was found to be a minimum. Truncation errors from the 110 runs made as a course of this optimization were compared with the errors experienced in the previous set of runs.
- Spiral distributions with 100, 200, 500, 1000, and 2000 vortices were used to determine the optimum value of N_c . As in the previous case, the optimum range of values for N_c was obtained when t_{cpu} was found to be a minimum. Truncation errors from the 62 runs made as a course of this optimization were compared with the errors experienced in the previous two sets of runs.

Based on the parametric study, it was found that reasonable accuracy could be obtained if one includes all of the terms through the 5th order in the Taylor series expansions ($N=5$). Furthermore, if one limits the maximum number of vortices in any childless box to 20 ($N_c=20$) then t_{cpu} will be near the minimum in all cases. The following sections provide a detailed description of the benchmark tests. It should be noted that all of the benchmark tests reported in the following sections were run on the department 1500 VAXcluster on a VAX 8820 (SAV08). In order to compare CPU times on different machines, runs were made on a VAX 8600 (SAV07) and on the two VAX 8650's (SAV01 and SAV03). A uniform distribution of 1024 vortices provided the test case. Values of $N_c=20$ and $N = 5$ were used. The CPU times for the VAX 8820 and the two VAX 8650's were virtually identical ($t_{cpu} = 50 \pm 0.9$ seconds) while the CPU time for the VAX 8600 was about 37% longer.

4.1 Error Associated with Truncation

A measure of the error incurred by truncation of the Taylor series after the N^{th} order terms can be obtained for the velocity potential and total velocity fields by means of the following equations:

$$E_{\psi} = \left[\frac{\sum_{i=1}^{N_v} (\psi_i - \psi_{ei})^2}{\sum_{i=1}^{N_v} \psi_{ei}^2} \right]^{\frac{1}{2}}, \quad (42)$$

$$E_V = \left[\frac{\sum_{i=1}^{N_v} (V_i - V_{ei})^2}{\sum_{i=1}^{N_v} V_{ei}^2} \right]^{\frac{1}{2}}.$$

Here, ψ_i and V_i are the velocity potential and total velocity at point i as calculated from the truncated series. The quantities ψ_{ei} and V_{ei} are the velocity potential and total velocity at point i as calculated exactly using the direct method. As indicated, these error calculations are obtained from summations over all of the N_v vortices in the flow. Assuming that single precision accuracy means that any number stored in the computer will be accurate through 6 significant digits, then single precision round off errors will produce values of E_{ψ} and E_V up to about 1×10^{-5} . Therefore, errors calculated from Equation 42 which are equal to or less than 1×10^{-5} imply that the machine round off errors are possibly as large as the truncation errors. For such cases, one should not continue to add terms to the Taylor series expansions for added accuracy unless the machine precision is also increased.

In order to gain some insight into the magnitude of the truncation error, a uniform distribution of 1024 vortices was placed in a box defined by $-1 < x < 1$ and $0 < r < 2$. The vortices along the box border were shifted toward the inside of the box by a distance of 0.01 to avoid their rejection by the mesh generator. This distribution is shown in Figure 10. The errors, as defined by Equation 42, for the stream function and total velocity fields are plotted in Figure 11. These errors are plotted versus N which is the order of the terms in the Taylor series just prior to truncation. It is interesting to note that the truncation error is not a strong function of the maximum number of vortices N_c in a cell or box. The errors associated with the stream function are on the order of 5×10^{-5} and 1×10^{-5} for values of N equal to 5 and 6 respectively. Thus, truncation errors for the stream function with N equal to 5 or 6 are of the same order of magnitude as possible machine round off errors and are therefore acceptable for single precision accuracy. Truncation errors associated with the total velocity for N equal to 5 and 6 are approximately equal to 3×10^{-3} and 1×10^{-3} respectively. For the total velocity,

one must decide whether or not errors of such magnitude are acceptable since they are two orders of magnitude larger than machine round off errors. One would suspect that, in most cases, velocity calculations that are accurate to within 0.1% to 0.3% will be acceptable.

Figure 10. Uniform Distribution of 1024 Vortices

Deciding whether one should use $N=5$ or $N=6$ is a trade off between truncation error and CPU time t_{cpu} . In order to gain some appreciation for the difference in t_{cpu} for $N=5$ versus $N=6$, consider the plot shown in Figure 12. As can be seen from this figure, t_{cpu} increases by about 50% for $N_c=10$ between $N=5$ and $N=6$ while for $N_c=20$ the increase is about 15%. For $N_c=100$ there is a 7% increase. The trend for t_{cpu} to increase more rapidly with increasing N for smaller values of N_c is a general one which is attributable to the time required to calculate the A and B coefficients associated with the Taylor series expansions. Since the increase in t_{cpu} is non-trivial between $N=5$ and $N=6$, a value of $N=5$ will be selected, albeit somewhat arbitrary. This value will be used in the subsequent section to evaluate the relationship between t_{cpu} and N_c and to obtain an optimum value for N_c .

(a) Stream-function error plot

(b) Total-velocity error plot

Figure 11. Error Plots for Uniform Distribution of 1024 Vortices

Figure 12. CPU Times for Uniform Distribution of 1024 Vortices

4.2 Optimum Number of Vortices in a Box

In order to optimize the maximum number of vortices N_c allowed in any childless box with respect to t_{cpu} , random and spiral type distributions were used. Examples of these distributions are shown in Figure 13. The cases in Figure 13 represent two extreme distributions with vortices being highly localized in the spiral distribution especially near its center. It was originally hoped that a single value of N_c could be found which would minimize t_{cpu} for both distributions over a range of N_v values. Fortunately, this turned out to be the case. It is presently assumed that this optimization will be valid for most other distributions. Optimization runs were made for random distributions with 50, 100, 200, 300, 500, 1000, and 2000 vortices. All of the vortices were chosen to be of equal strength. Vortices in the random distributions were not allowed to be any closer to the box border than 0.003 times the length of a side of the box or no closer than 0.1. Optimization runs were made for spiral distributions with 100, 200, 500, 1000, and 2000 vortices. Equations for points x_p and r_p on the spiral curve are given by:

$$\begin{aligned}
 x_p &= x_c + S_r \cos \theta, \\
 r_p &= r_c + S_r \sin \theta, \\
 S_r &= \frac{2\theta}{\pi} \left(\sin^2 \theta + \frac{1}{4} \cos^2 \theta \right)^{-\frac{1}{2}},
 \end{aligned} \tag{43}$$

where $x_c=16.00$ and $r_c=10.67$. The angle θ (in radians) is measured counter clockwise about the point (x_c, r_c) from an axis parallel to the x axis. Vortices of equal strength were placed on the spiral curve at equal intervals of θ where θ ranges from 0 to 2.8π .

(a) Example random distribution

(b) Example spiral distribution

Figure 13. Typical Distributions Used in Optimization of N_c

Figure 14. CPU Times for Random Distribution of 1000 Vortices

Figure 14 shows a typical plot of t_{cpu} versus N_v . From this figure, it can be seen that the CPU time is a minimum for values of N_v equal to any integer between 10 and 50. The maximum value of t_{cpu} occurs for $N_v=1000$. Since there are only 1000 vortices in this distribution, the mesh consists of a single box for $N_v=1000$ which corresponds to a direct simulation. Plots similar to Figure 14 were made for the twelve random and spiral distributions. Examination of those plots revealed that if one chooses $N_v=20$ that a near optimum (minimum) value of t_{cpu} will be obtained.

Comparisons of t_{cpu} using the direct simulation versus using the fast solver with $N_v=20$ and $N=5$ for the random distributions are shown in Figure 15 (a). As can be seen from this figure, there is no advantage in using the fast solver for distributions which have less than about 100 vortices. For situations where there are more than 100 vortices, one can write equations for the straight lines plotted through the data. For the random distribution shown in Figure 15 (a), the CPU time for the direct simulation can be expressed by:

$$t_{cpu} = 3.85 \times 10^{-4} N_v^2. \quad (44)$$

On the other hand, the CPU time for the fast solver used on the random distribution can be expressed by;

$$t_{cpu} = 8.90 \times 10^{-3} N_v^{1.25}. \quad (45)$$

Therefore, the fast solver will run in about 3% of the time required for the direct simulation for the case where there are 10,000 vortices in the flow.

(a) Random Distributions

(b) Spiral Distributions

Figure 15. CPU Time Comparisons Between Direct and Fast Solvers

For the spiral distribution whose CPU times are shown in Figure 15 (b), the CPU time for the direct simulation is given by:

$$t_{cpu} = 3.85 \times 10^{-4} N_v^2. \quad (46)$$

As would be expected, the CPU times required for direct simulations of either the random or spiral distributions are the same. The CPU time for the fast solver used on the spiral distribution can be expressed by;

$$t_{cpu} = 4.08 \times 10^{-2} N_v. \quad (47)$$

In this case, the fast solver will run in about 1% of the time required for the direct simulation when there are 10,000 vortices in the flow.

At this point, it is appropriate to check the truncation errors produced using the fast solver on the random and spiral distributions to see if they are about as one might expect. From the work done in Section 4.1 with $N=5$ the truncation error for the stream function E_ψ was on the order of 5×10^{-5} while the error in total velocity E_V was on the order of 3×10^{-3} . Truncation errors are plotted in Figure 16 for the random and spiral distributions versus the total number of vortices N_v . As can be seen from this figure, the order of magnitude of these errors are relatively independent of the number of total vortices N_v in the distributions. One exception is the error associated with $N_v=50$. For $N_v=50$, the errors are very small due to the fact that the simulation is essentially a direct simulation since there will be at most four boxes produced by the mesh generator for a random distribution with $N_c=20$. Excluding the data for $N_v=50$, the truncation error for the stream function E_ψ is on the order of 5×10^{-5} while the error in total velocity E_V is on the order of 1×10^{-3} . These errors are roughly of the same order of magnitude as those found for the uniform distribution of Section 4.1.

(a) Stream-function errors ($\circ$ - random, $\triangle$ - spiral)

(b) Total-velocity errors ($\circ$ - random, $\triangle$ - spiral)

Figure 16. Truncation Errors for Random and Spiral Distributions

5 SELF INDUCTION

All of the velocity calculations which have been made thus far have ignored the self-induced velocity associated with each vortex ring. If one were attempting to calculate the convection velocity of a particular ring, the magnitude of the self-induced velocity would be of obvious importance. Some very interesting discussions concerning the motion and subsequent distortion of vortex cores may be found in a recent work by Shariff, Leonard, and Ferziger [8]. The present work will be limited to obtaining the self-induced velocity at the center of impulse for a circular or square-core ring vortex.

According to Lamb [2], the self-induced axial velocity U_o of a vortex ring of radius b with a circular core of radius a is given by:

$$U_o = \frac{\Gamma}{4\pi b} \left[\ln \left(\frac{8b}{a} \right) - \frac{1}{4} \right], \quad (48)$$

where Γ is the vortex strength. This equation is strictly valid for a vortex which has a small core ($a \ll b$) with a uniform distribution of vorticity. If this equation is written in terms of the vorticity ω , in place of the circulation, one obtains:

$$U_o = \frac{a\omega}{4b} \left[\ln \left(\frac{8b}{a} \right) - \frac{1}{4} \right]. \quad (49)$$

From Equation 49, it can be seen that whether or not the self-induced velocity is important depends to a large extent on the core size a . For instance, suppose that one were interested in obtaining the velocity at a point in a uniform vorticity field ($\omega = \text{constant}$) and that the field was simulated with a number N_v of uniformly distributed discrete vortices. The appropriate core radius for each vortex would be proportional to the square root of N_v . In the limit as $N_v \rightarrow \infty$, $a \rightarrow 0$. From Equation 49 as $a \rightarrow 0$, $U_o \rightarrow 0$. This suggests that by using a large number of discrete vortices to represent a vorticity field, the self-induced velocities may be made small enough to be neglected. On the other hand, if one chooses to use a coarse discretization of the vorticity field then some equation similar to Equation 49 must be used. In the following sections, vortex cores which have uniform distributions of vorticity within them will be examined to determine how many discrete vortices are required to obtain their proper self-induced velocities. In addition, correlations will be developed which can be used to obtain correct self-induced velocities for coarse discretizations.

A somewhat general discussion concerning the motion of vortex rings with cores can be found in reference [2]. For a collection of i vortices, a centroid based on the impulse of the vortices is first obtained. The impulse for the system of vortices is given by:

$$I = \pi\rho \sum_i \Gamma_i b_i^2, \quad (50)$$

where ρ is the fluid density and Γ_i and b_i are the strength and radius of the i^{th} vortex. The time rate of change of impulse represents a force which is zero for a collection of

"free" vortices such as would be encountered in a wake flow. Assuming that the vortices are free, the impulse given by Equation 50 is a constant. The x and r coordinates of the center of impulse (x_o, r_o) are given by:

$$\begin{aligned} x_o &= \frac{\sum_i \Gamma_i b_i^2 x_i}{\sum_i \Gamma_i b_i^2} , \\ r_o^2 &= \frac{\sum_i \Gamma_i b_i^2}{\sum_i \Gamma_i} . \end{aligned} \tag{51}$$

A single vortex (with the strength of the sum of the i vortices) has the same impulse as the collection of i vortices when placed at r_o . The value of x_o represents the axial centroid of the impulse for the collection of i vortices. The motion of the center of impulse (x_o, r_o) can be obtained by differentiating the expressions in Equation 51 with respect to time t . Since the impulse is constant, r_o is a constant and the radial velocity V_o of the center of impulse is equal to zero. On the other hand, the axial velocity U_o of the center of impulse is not, in general, zero and is given by:

$$U_o = \frac{\sum_i \Gamma_i b_i (b_i u_i + 2x_i v_i)}{\sum_i \Gamma_i b_i^2} , \tag{52}$$

where u_i and v_i are the axial and radial velocities at the i^{th} vortex.

5.1 Square Cores

Simulations of vortex rings with square cores were run in order to develop an understanding of the number of discrete vortices required to properly represent the core region. It was assumed that the vorticity in the core was uniform. Following the discussion in the previous section, one would expect that if the core region is represented by a large number of discrete vortices that the proper self-induced velocity for the ring will be obtained. The approach that was taken to determine the number N_v of discrete vortices required in the core was to increase N_v until the velocity U_o calculated using Equation 52 became invariant. Cases were run for eight values of N_v and three values of b/s , where s is equal to one half of the distance along a side of the square core and b is the radius to the center of the core. Results from this series of simulations are shown in Figure 17. As can be seen from this figure, the self-induced velocity becomes essentially constant for $N_v > 100$. For example, the induction velocity increases by only 1 to 2% between $N_v=100$ and $N_v=1000$.

Figure 17. Self-Induced Velocity Simulations of Square Core Vortex Rings

Next, a set of simulations were run in order to obtain an equation similar to Equation 48 which can be used to calculate the self-induced velocity for a vortex ring with various ring radius to core size ratios b/s . It was first assumed that the self-induced velocity for a vortex ring with a square core can be written in the form:

$$U_o = \frac{\Gamma}{4\pi b} \left[\ln \left(\frac{8b}{s} \right) - C_s \right], \quad (53)$$

where C_s is some function of b/s . Simulations were run for ten different values of b/s by placing 1089 vortices uniformly in the square cores. Equation 53 was solved for C_s using values of U_o obtained from Equation 52. A plot of the resulting values of C_s as a function of b/s is shown in Figure 18. As can be seen from this figure, the maximum value of C_s is equal to 1.011 for $b/s=1$. For large values of b/s , $C_s \rightarrow 0.423$. A very good fit to the data in Figure 18 can be obtained from the following equation which is shown as the chain dot line:

$$C_s = .423 + 1.064 \exp \left(- .593 \frac{b}{s} \right). \quad (54)$$

In Figure 19, the simulated values of $\frac{U_o b}{\Gamma}$ are plotted along with a curve (shown as a chain dot line) obtained by using Equation 54 in Equation 53. The fit between the correlation and data is seen to be quite good.

At this point, the self-induced velocity of a square core can be obtained by simulating it with a large number of discrete vortices in which the self-induced velocity of an individual discrete vortex is ignored or one can use Equations 53 and 54 to calculate the self-induced velocity for a single square core vortex. What about the situation

Figure 18. C_s Function for Square Core Vortex Rings

Figure 19. Self-Induced Velocities of Square Core Vortex Rings

where one wishes to discretize the core with a few vortices as opposed to one or many? It can be demonstrated that if one adds the "local" self-induced velocity u_{oi} to u_i in Equation 52 that the correct value of $\frac{U_o b}{\Gamma}$ will be obtained regardless of the number of vortices N_v used to discretize the core. The local self-induced velocity is obtained from Equations 54 and 53 with Γ replaced by $\Gamma/\sqrt{N_v}$ and s replaced by $s/\sqrt{N_v}$ to correctly reflect values of Γ and s for the local cell. The local value of b is also used. Results of the modification which includes the local self-induced velocity is shown in Figure 20. The unmodified versus modified results can be compared by observing Figures 17 and 20 respectively. This result is significant in that it suggests that one can obtain the correct self-induced velocity of vortices which simulate a flow field without regard to the level of discretization.

Figure 20. Modified Self-Induced Velocity Simulations of Square Core Vortex Rings

The center of impulse (x_o, r_o) of a square core was obtained from Equation 51 applied to a 1089 vortex simulation. Since the vorticity distribution about the center of the core is symmetrical, x_o is equal to the axial location of the core center. The radial location of the center of impulse is shown in Figure 21. The following equation, which is shown as the chain dot curve, fits the data reasonably well:

$$\frac{r_o}{b} = 1.0 + .586 \exp \left(-1.33 \frac{b}{s} \right) . \quad (55)$$

Figure 21. Radial Center of Impulse of Square Core Vortex Rings

5.2 Circular Cores

Relationships for the circular core vortex rings were obtained by simulating the core with a set of 973 uniformly spaced vortices. An example of the vortex placement can be seen in Figure 7(a). Simulations were first run in order to obtain an equation similar to Equation 48 which can be used to calculate the self-induced velocity for a vortex ring with various ring radius to core radius ratios b/a . Here a is the core radius. The self-induced velocity for a vortex ring with a circular core was assumed to be of the form:

$$U_o = \frac{\Gamma}{4\pi b} \left[\ln \left(\frac{8b}{a} \right) - C_a \right], \quad (56)$$

where C_a is some function of b/a . Simulations were run for ten different values of b/a . Equation 56 was solved for C_a using values of U_o obtained from Equation 52. A plot of the resulting values of C_a as a function of b/a is shown in Figure 22. As can be seen from this figure, the maximum value of C_a is equal to 0.085 for $b/a=1$. For large values of b/a , $C_a \rightarrow 0.250$. A good fit to the data in Figure 22 can be obtained from the following equation which is shown as the chain dot line:

$$C_a = .250 + 1.064 \exp \left(-.669 \frac{b}{a} \right). \quad (57)$$

Figure 22. C_a Function for Circular Core Vortex Rings

It is interesting to note that an estimate for C_a could have been obtained from Equations 54 and 53 by assuming an equivalence between the core areas A_c for a square and circular core (ie., $A_c = 4s^2 = \pi a^2$). This assumption yields an equation identical to Equation 57 except the constant term is equal to 0.302 instead of 0.250. Therefore, the C_a plot of Figure 22 would be displaced upward by 0.052.

In Figure 23 the simulated values of $\frac{U_c b}{\Gamma}$ are plotted along with a curve (shown as a chain dot line) obtained by using Equation 57 in Equation 56. The fit between the correlation and data is seen to be quite good. Based on the comments of the previous paragraph, if $\frac{U_c b}{\Gamma}$ for the square and circular cores were plotted on the same plot versus $\frac{b}{\sqrt{A_c}}$, the data for the square core would be above that of the circular core data by only .004 (ie., $.052/4\pi$).

The center of impulse (x_o, r_o) of a circular core was obtained from Equation 51 applied to the 973 vortex simulation. Since the vorticity distribution about the center of the core is symmetrical, x_o is equal to the axial location of the core center. The radial location of the center of impulse is shown in Figure 24. The following equation is shown as the chain dot curve:

$$\frac{r_o}{b} = 1.0 + .586 \exp \left(-1.50 \frac{b}{a} \right) . \quad (58)$$

Equation 58 was obtained by making the substitution $4s^2 = \pi a^2$ in Equation 55. Equation 58 fits the data reasonably well.

Figure 23. Self-Induced Velocities of Circular Core Vortex Rings

Figure 24. Radial Center of Impulse of Circular Core Vortex Rings

6 SUMMARY

The work accomplished can be summarized as follows:

- Source and field point Taylor series expansions have been developed for axisymmetric ring vortices. These series are calculated using associated Legendre functions in conjunction with recursion relationships derived from stream function considerations. Relationships which allow the source and or field expansion points to be translated within their domains of convergence were also developed.
- A Fortran computer code RSOLV has been written to generate the adaptive mesh, box lists, and Taylor series expansions associated with the fast solution technique. The code also executes the fast solution technique to calculate the stream function and the axial and radial velocity components at points in the flow field.
- Test cases have been run to optimize the RSOLV code and to benchmark the truncation errors and CPU time savings associated with the method. The Taylor series were truncated after 5th order terms and the maximum number of vortices allowed in a childless box was set at 20. Truncation errors for the stream function and total velocity field were on the order of 5×10^{-5} and 3×10^{-3} respectively. For 100 vortices in the field there was virtually no CPU time savings with the fast solver. For 10,000 vortices in the flow, the fast solver obtained solutions in about 1% to 3% of the time required for the direct solution technique.
- Simulations for vortices with square and circular cores were run in order to obtain expressions for the self-induced velocities of such vortices. In each case, the vorticity was uniform in the core. Correlations for the non-dimensional self-induced velocity as a function of the non-dimensional ring radius were found to be almost identical for the square and circular core vortices. It appears that these correlations can be used to correctly convect discretized patches of vorticity in the flow field.

References

- [1] Carrier, J., Greengard, L., and Rokhlin, V., "A Fast Adaptive Multipole Algorithm For Particle Simulations," *SIAM Journal on Scientific and Statistical Computing*, Vol.9, No.4, pp. 669-686, July, 1988.
- [2] Lamb, H., *Hydrodynamics*, 6th Edition, Dover Publications, New York, p.239, 1932.
- [3] Gradshteyn, I. S. and Ryzhik, I. M., *Table of Integrals, Series, and Products*, Academic Press, 1980.
- [4] Hart, J. F., *Computer Approximations*, SIAM Series in Applied Mathematics, Wiley and Sons Inc., New York, pp.151-154, 1968.
- [5] Bulirsch, Roland, *Numerische Mathematik*, vol.7, p.78, 1965.
- [6] Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T., *Numerical Recipes*, Cambridge University Press, Cambridge, p.187, 1988.
- [7] Lowan, A. N., *Tables of Associated Legendre Functions*, Columbia University Press, New York, 1945.
- [8] Shariff, K., Leonard, A., Ferziger, J. H., "Dynamics of a Class of Vortex Rings," NASA Technical Memorandum 102257, Ames Research Center, December, 1989.

END

DATE FILMED

12 / 13 / 90