

NASA/TM—2001-210985



# Technique for Very High Order Nonlinear Simulation and Validation

Rodger W. Dyson  
Glenn Research Center, Cleveland, Ohio

---

July 2001

## The NASA STI Program Office . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program Office plays a key part in helping NASA maintain this important role.

The NASA STI Program Office is operated by Langley Research Center, the Lead Center for NASA's scientific and technical information. The NASA STI Program Office provides access to the NASA STI Database, the largest collection of aeronautical and space science STI in the world. The Program Office is also NASA's institutional mechanism for disseminating the results of its research and development activities. These results are published by NASA in the NASA STI Report Series, which includes the following report types:

- **TECHNICAL PUBLICATION.** Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA's counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.
- **TECHNICAL MEMORANDUM.** Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.
- **CONTRACTOR REPORT.** Scientific and technical findings by NASA-sponsored contractors and grantees.

- **CONFERENCE PUBLICATION.** Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or cosponsored by NASA.
- **SPECIAL PUBLICATION.** Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.
- **TECHNICAL TRANSLATION.** English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services that complement the STI Program Office's diverse offerings include creating custom thesauri, building customized data bases, organizing and publishing research results . . . even providing videos.

For more information about the NASA STI Program Office, see the following:

- Access the NASA STI Program Home Page at <http://www.sti.nasa.gov>
- E-mail your question via the Internet to [help@sti.nasa.gov](mailto:help@sti.nasa.gov)
- Fax your question to the NASA Access Help Desk at 301-621-0134
- Telephone the NASA Access Help Desk at 301-621-0390
- Write to:  
NASA Access Help Desk  
NASA Center for AeroSpace Information  
7121 Standard Drive  
Hanover, MD 21076

NASA/TM—2001-210985



# Technique for Very High Order Nonlinear Simulation and Validation

Rodger W. Dyson  
Glenn Research Center, Cleveland, Ohio

National Aeronautics and  
Space Administration

Glenn Research Center

---

July 2001

## Acknowledgments

I would like to thank Dr. John Goodrich for introducing me to high order schemes. In addition, I thank Dr. Ray Hixon, Dr. James Scott, and Dr. Edmane Envia for offering suggestions and encouragement.

Available from

NASA Center for Aerospace Information  
7121 Standard Drive  
Hanover, MD 21076

National Technical Information Service  
5285 Port Royal Road  
Springfield, VA 22100

Available electronically at <http://gltrs.grc.nasa.gov/GLTRS>

# Technique for Very High Order Nonlinear Simulation and Validation

Roger W. Dyson  
National Aeronautics and Space Administration  
Glenn Research Center  
Cleveland, Ohio 44135

Finding the sources of sound in large nonlinear fields via direct simulation currently requires excessive computational cost. This paper describes a simple technique for efficiently solving the multidimensional nonlinear Euler equations that significantly reduces this cost and demonstrates a useful approach for validating high order nonlinear methods. Up to 15<sup>th</sup> order accuracy in space and time methods were compared and it is shown that an algorithm with a fixed design accuracy approaches its maximal utility and then its usefulness exponentially decays unless higher accuracy is used. It is concluded that at least a 7<sup>th</sup> order method is required to efficiently propagate a harmonic wave using the nonlinear Euler equations to a distance of 5 wavelengths while maintaining an overall error tolerance that is low enough to capture both the mean flow and the acoustics.

## 1. Introduction

It is well known that aeroacoustic computations require high order numerical schemes to efficiently capture the multiscale phenomena occurring over a relatively large time interval. But the question of how high has not been answered because of a number of reasons, such as the following:

- Acoustics may be mathematically solved in a number of ways such as via acoustic analogies <sup>1</sup>, direct simulations <sup>2</sup>, and perturbation techniques <sup>3</sup>.
- Commonly used numerical approaches such as Compact <sup>4</sup>, Essentially Non-Oscillatory <sup>5</sup>, Dispersion Relation Preserving <sup>6</sup>, and Galerkin <sup>7</sup> schemes are difficult to extend to very high order.
- The numerical requirements vary considerably due to the wide variety of acoustical phenomena which include linear propagation and scattering, inviscid gust interactions with objects, receptivity effects, and turbulent noise generation.

Since most Computational Aeroacoustics (CAA) techniques have a fixed design accuracy and resolution, they offer limited opportunities for exploring the effects of design accuracy on the simulation. Instead, an arbitrary accuracy Modified Expansion Solution Approximation

(MESA) technique <sup>8,9,10</sup> is used here. This method was recently implemented <sup>11</sup> in a way that greatly simplified its use and demonstrated the possibility of subgrid scale resolution by including the spatial derivatives of the primitive variables on the grid and by using very high design accuracies <sup>12</sup>. For these reasons, the MESA scheme is extended here to the nonlinear Euler equations so that the design accuracy required for an efficient direct numerical acoustical simulation can be quantified.

In addition, as the accuracy requirements increase so does the complexity of the nonlinear simulator. Many subtle errors can be introduced, particularly for nonlinear systems which have few analytical solutions available for testing with. Therefore, the Method of Manufactured Solutions <sup>13</sup> is applied here and found to be a useful tool for developing and testing very high order methods on complex nonlinear systems.

## 2. Problem Description

The following form of the two-dimensional nonlinear Euler equations are used <sup>14,15</sup>:

$$\frac{\partial \rho}{\partial t} + u \frac{\partial \rho}{\partial x} + v \frac{\partial \rho}{\partial y} + \rho \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) = Q_1 \quad (2.1)$$

$$\frac{\partial p}{\partial t} + u \frac{\partial p}{\partial x} + v \frac{\partial p}{\partial y} + \gamma p \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) = Q_2 \quad (2.2)$$

$$\frac{\partial(\rho u)}{\partial t} + \frac{\partial(\rho u^2 + p)}{\partial x} + \frac{\partial(\rho uv)}{\partial y} = Q_3 \quad (2.3)$$

$$\frac{\partial(\rho v)}{\partial t} + \frac{\partial(\rho uv)}{\partial x} + \frac{\partial(\rho v^2 + p)}{\partial y} = Q_4 \quad (2.4)$$

This form does not explicitly use the energy primitive variable since the equation of state was used to eliminate it, resulting in only four unknown variables:  $\rho$ ,  $p$ ,  $u$ ,  $v$ , which are the density, static pressure, x-velocity, and y-velocity respectively. The ratio of specific heats is defined as  $\gamma = \frac{c_p}{c_v} = 1.4$ .

Ordinarily, the source terms ( $Q_1, Q_2, Q_3, Q_4$ ) are zero. But, for testing purposes, these terms will be modified so that the following analytical solution is defined:

$$\rho(x, y, t) = a_1 \cos(k_x \pi x) \cos(k_y \pi y) \cos(k_t \pi t) + c_1 \quad (2.5)$$

$$u(x, y, t) = a_2 \cos(k_x \pi x) \cos(k_y \pi y) \cos(k_t \pi t) + c_2 \quad (2.6)$$

$$v(x, y, t) = a_3 \cos(k_x \pi x) \cos(k_y \pi y) \cos(k_t \pi t) + c_3 \quad (2.7)$$

$$p(x, y, t) = a_4 \cos(k_x \pi x) \cos(k_y \pi y) \cos(k_t \pi t) + c_4 \quad (2.8)$$

This simple harmonic solution is used to test the resolution of each method as their design accuracy is modified and it enables comparisons to be made with previous linearized results which also used this test <sup>12</sup>. The source terms,  $Q_i$ , are then found using computer algebra and the resulting code is verified using the Method of Manufactured Solutions <sup>16,17</sup>. The source terms corresponding to this analytical solution are shown in Appendix A.

It is important to insure the density and pressure variables,  $\rho(x, y, t)$  and  $p(x, y, t)$ , are greater than zero when choosing the free parameters, ( $a_i$  and  $c_i$ ), because the numerical

approach will otherwise be numerically unstable due to excessive roundoff error from division and multiplication by small numbers. The linearized Euler equations were previously solved with a linear MESA scheme <sup>12</sup> which did not have this source of instability. However, since  $p$  and  $\rho$  are not perturbation quantities in Eqs. (2.1-2.4), they will never be negative in practical applications and therefore this instability is only of concern while imposing the artificial solution, Eq. (2.5).

### 3. Numerical Approach

The numerical approach described here will produce single step, explicit,  $O = N(s + 1) - 1$  ( $\forall$  integer  $s \geq 0$  and  $N \geq 1$ ) order of accuracy in space and time methods for the solution of the two-dimensional nonlinear Euler equations using a  $N \times N$  stencil containing the primitive variables and their mixed spatial derivatives,  $\frac{\partial}{\partial x^{0\dots s} \partial y^{0\dots s}}$ .

The primitive variables and their spatial derivatives are advanced in time on a staggered grid <sup>12</sup> such that time advancement is always performed in the center of each stencil by:

$$\frac{\partial^{a+b} \rho(x, y, \Delta t)}{\partial x^a y^b} = \sum_{k=0}^O \frac{1}{k!} C_{a,b,k}^{\rho} \Delta t^k \quad (3.9)$$

$$\frac{\partial^{a+b} \rho(x, y, \Delta t) u(x, y, \Delta t)}{\partial x^a y^b} = \sum_{k=0}^O \frac{1}{k!} C_{a,b,k}^{\rho u} \Delta t^k \quad (3.10)$$

$$\frac{\partial^{a+b} \rho(x, y, \Delta t) v(x, y, \Delta t)}{\partial x^a y^b} = \sum_{k=0}^O \frac{1}{k!} C_{a,b,k}^{\rho v} \Delta t^k \quad (3.11)$$

$$\frac{\partial^{a+b} p(x, y, \Delta t)}{\partial x^a y^b} = \sum_{k=0}^O \frac{1}{k!} C_{a,b,k}^p \Delta t^k \quad (3.12)$$

where

$$C_{a,b,k}^f = \frac{\partial^{a+b+k} f(x, y, t = 0)}{\partial x^a y^b t^k} \quad (3.13)$$

The mixed space-time ( $C_{a,b,k}^f$ ) derivatives are found by differentiating the governing equations in space and time using the Cauchy-Kowalewski technique <sup>18</sup>. The difficulty in doing this with nonlinear systems is in differentiating products and this was cited in the past as a reason for not using it <sup>19</sup>. Fortunately, it can in fact be performed systematically using a multidimensional extension of the Leibniz rule <sup>20</sup>:

$$\frac{\partial^{a+b+k} f(x, y, t) g(x, y, t)}{\partial x^a y^b t^k} = \sum_{i=0}^a \sum_{j=0}^b \sum_{c=0}^k \left[ \binom{a}{i} \binom{b}{j} \binom{k}{c} \frac{\partial^{(a-i)+(b-j)+(k-c)} f(x, y, t)}{\partial x^{a-i} \partial y^{b-j} \partial t^{k-c}} \frac{\partial^{i+j+c} g(x, y, t)}{\partial x^i \partial y^j \partial t^c} \right] \quad (3.14)$$

By repeatedly applying the Leibniz rule to the governing equations, the explicit forms of

the coefficients,  $C_{a,b,k}^f$ , are found. These coefficients are evaluated with the following loop:

DO  $k = 1, O$ ; DO  $b = 0, O$ ; DO  $a = 0, O$

$C_{a,b,k}^{\rho(x,y,t)} =$

$$- \sum_{i=0}^a \binom{a}{i} \sum_{j=0}^b \binom{b}{j} \sum_{c=0}^{k-1} \binom{k-1}{c} \left[ C_{(a-i,b-j,(k-1)-c)}^{u(x,y,t)} C_{i+1,j,c}^{\rho(x,y,t)} + C_{(a-i,b-j,(k-1)-c)}^{\rho(x,y,t)} C_{i+1,j,c}^{u(x,y,t)} + C_{(a-i,b-j,(k-1)-c)}^{v(x,y,t)} C_{i,j+1,c}^{\rho(x,y,t)} + C_{(a-i,b-j,(k-1)-c)}^{\rho(x,y,t)} C_{i,j+1,c}^{v(x,y,t)} \right] + C_{a,b,k-1}^{Q_1}$$

$C_{a,b,k}^{p(x,y,t)} =$

$$- \sum_{i=0}^a \binom{a}{i} \sum_{j=0}^b \binom{b}{j} \sum_{c=0}^{k-1} \binom{k-1}{c} \left[ C_{(a-i,b-j,(k-1)-c)}^{u(x,y,t)} C_{i+1,j,c}^{p(x,y,t)} + \gamma C_{(a-i,b-j,(k-1)-c)}^{p(x,y,t)} C_{i+1,j,c}^{u(x,y,t)} + C_{(a-i,b-j,(k-1)-c)}^{v(x,y,t)} C_{i,j+1,c}^{p(x,y,t)} + \gamma C_{(a-i,b-j,(k-1)-c)}^{p(x,y,t)} C_{i,j+1,c}^{v(x,y,t)} \right] + C_{a,b,k-1}^{Q_2}$$

$C_{a,b,k}^{\rho(x,y,t)u(x,y,t)} =$

$$- \sum_{i_1=0}^a \sum_{j_1=0}^b \sum_{c_1=0}^{k-1} \sum_{i_2=0}^{i_1} \sum_{j_2=0}^{j_1} \sum_{c_2=0}^{c_1} \left[ \binom{a}{i_1} \binom{b}{j_1} \binom{k-1}{c_1} \binom{i_1}{i_2} \binom{j_1}{j_2} \binom{c_1}{c_2} \left( C_{a-i_1+1,b-j_1,(k-1)-c_1}^{r(x,y,t)} C_{i_1-i_2,j_1-j_2,c_1-c_2}^{u(x,y,t)} + 2C_{a-i_1,b-j_1,(k-1)-c_1}^{r(x,y,t)} C_{i_1-i_2,j_1-j_2,c_1-c_2}^{u(x,y,t)} C_{i_2+1,j_2,c_2}^{u(x,y,t)} + C_{a-i_1,b-j_1+1,(k-1)-c_1}^{r(x,y,t)} C_{i_1-i_2,j_1-j_2,c_1-c_2}^{u(x,y,t)} C_{i_2,j_2,c_2}^{v(x,y,t)} + C_{a-i_1,b-j_1,(k-1)-c_1}^{r(x,y,t)} C_{i_1-i_2,j_1-j_2+1,c_1-c_2}^{u(x,y,t)} C_{i_2,j_2,c_2}^{v(x,y,t)} + C_{a-i_1,b-j_1,(k-1)-c_1}^{r(x,y,t)} C_{i_1-i_2,j_1-j_2,c_1-c_2}^{u(x,y,t)} C_{i_2,j_2+1,c_2}^{v(x,y,t)} \right) - C_{a+1,b,k-1}^p \right] + C_{a,b,k-1}^{Q_3}$$

$C_{a,b,k}^{\rho(x,y,t)v(x,y,t)} =$

$$- \sum_{i_1=0}^a \sum_{j_1=0}^b \sum_{c_1=0}^{k-1} \sum_{i_2=0}^{i_1} \sum_{j_2=0}^{j_1} \sum_{c_2=0}^{c_1} \left[ \binom{a}{i_1} \binom{b}{j_1} \binom{k-1}{c_1} \binom{i_1}{i_2} \binom{j_1}{j_2} \binom{c_1}{c_2} \left( C_{a-i_1,b-j_1+1,(k-1)-c_1}^{r(x,y,t)} C_{i_1-i_2,j_1-j_2,c_1-c_2}^{v(x,y,t)} + 2C_{a-i_1,b-j_1,(k-1)-c_1}^{r(x,y,t)} C_{i_1-i_2,j_1-j_2,c_1-c_2}^{v(x,y,t)} C_{i_2,j_2+1,c_2}^{v(x,y,t)} + C_{a-i_1+1,b-j_1,(k-1)-c_1}^{r(x,y,t)} C_{i_1-i_2,j_1-j_2,c_1-c_2}^{v(x,y,t)} C_{i_2,j_2,c_2}^{u(x,y,t)} + C_{a-i_1,b-j_1,(k-1)-c_1}^{r(x,y,t)} C_{i_1-i_2+1,j_1-j_2,c_1-c_2}^{v(x,y,t)} C_{i_2,j_2,c_2}^{u(x,y,t)} + C_{a-i_1,b-j_1,(k-1)-c_1}^{r(x,y,t)} C_{i_1-i_2,j_1-j_2,c_1-c_2}^{v(x,y,t)} C_{i_2+1,j_2,c_2}^{u(x,y,t)} \right) - C_{a,b+1,k-1}^p \right] + C_{a,b,k-1}^{Q_4}$$

END DO; END DO

Perform Conversion Eq. (3.16)

END DO

(3.15)



DO  $a, b : (a + b = 0, 1, \dots, 2O)$  and  $(0 \leq a, b \leq O)$

$C_{a,b,k}^u =$

$$\begin{aligned} & \frac{1}{C_{0,0,0}^\rho} C_{a,b,k}^{\rho u} - \frac{1}{C_{0,0,0}^\rho} \left( \sum_{i=0}^{a-1} \sum_{j=0}^{b-1} \sum_{c=0}^{k-1} \binom{a}{i} \binom{b}{j} \binom{k}{c} C_{a-i,b-j,k-c}^\rho C_{i,j,c}^u + \right. \\ & \quad \sum_{i=0}^{a-1} \sum_{c=0}^k \binom{a}{i} \binom{k}{c} C_{a-i,0,k-c}^\rho C_{i,b,c}^u + \\ & \quad \sum_{j=0}^{b-1} \sum_{c=0}^k \binom{b}{j} \binom{k}{c} C_{0,b-j,k-c}^\rho C_{a,j,c}^u + \\ & \quad \sum_{i=0}^{a-1} \sum_{j=0}^{b-1} \binom{a}{i} \binom{b}{j} C_{a-i,b-j,0}^\rho C_{i,j,k}^u + \\ & \quad \left. \sum_{c=0}^{k-1} \binom{k}{c} C_{0,0,k-c}^\rho C_{a,b,c}^u \right) \end{aligned}$$

$C_{a,b,k}^v =$

$$\begin{aligned} & \frac{1}{C_{0,0,0}^\rho} C_{a,b,k}^{\rho v} - \frac{1}{C_{0,0,0}^\rho} \left( \sum_{i=0}^{a-1} \sum_{j=0}^{b-1} \sum_{c=0}^{k-1} \binom{a}{i} \binom{b}{j} \binom{k}{c} C_{a-i,b-j,k-c}^\rho C_{i,j,c}^v + \right. \\ & \quad \sum_{i=0}^{a-1} \sum_{c=0}^k \binom{a}{i} \binom{k}{c} C_{a-i,0,k-c}^\rho C_{i,b,c}^v + \\ & \quad \sum_{j=0}^{b-1} \sum_{c=0}^k \binom{b}{j} \binom{k}{c} C_{0,b-j,k-c}^\rho C_{a,j,c}^v + \\ & \quad \sum_{i=0}^{a-1} \sum_{j=0}^{b-1} \binom{a}{i} \binom{b}{j} C_{a-i,b-j,0}^\rho C_{i,j,k}^v + \\ & \quad \left. \sum_{c=0}^{k-1} \binom{k}{c} C_{0,0,k-c}^\rho C_{a,b,c}^v \right) \end{aligned}$$

END DO (3.16)

After each loop of the  $k$  index in Eq. (3.15), the following momentum coefficients are known:

$$\begin{aligned} & C_{a,b,k}^{\rho u} \text{ with } (a, b) : (a, b = 0, 1, 2, \dots, O) \\ & C_{a,b,k}^{\rho v} \text{ with } (a, b) : (a, b = 0, 1, 2, \dots, O) \end{aligned} \quad (3.17)$$

But the  $C_{a,b,k}^u$  and  $C_{a,b,k}^v$  coefficients are required to evaluate the coefficients,  $C_{0 \dots O, 0 \dots O, k+1}^f$  at the next  $k + 1$  loop. By repeatedly applying the Leibniz' rule to Eq. (3.17) an explicit expression for the  $C_{a,b,k}^u$  and  $C_{a,b,k}^v$  coefficients is found, Eq. (3.16), in terms of already known variables. Note, however, that these coefficients must be evaluated in the sequence

of increasing derivative order (i.e., Find  $C_{a,b,k}^u$  and  $C_{a,b,k}^v$  in the order  $(a,b):(a+b=0,1,\dots,2O)$ . By applying this conversion step once for each k-loop increment in Eq. (3.15), we can efficiently continue increasing the index k until all the coefficients, Eq (3.13), are found.

#### 4. Determining the Spatial Derivatives

The procedure described in the last section will advance the nonlinear Euler equations in time at  $(x,y)$  with arbitrary accuracy if all the spatial derivatives:

$$C_{a,b,0}^f \forall (a,b) : a,b = 0,1,\dots,O$$

are known at  $(x,y)$ . However, only the spatial derivatives:

$$C_{a,b,0}^f \forall (a,b) : a,b = 0,1,\dots,s$$

are available on each grid point,  $(x_i, y_j)$ . Therefore, the higher order spatial derivatives,  $(a,b \geq s+1)$ , need to be interpolated. In addition, since time advancement is performed using a staggered grid, both the known derivatives on the grid and the higher order unknown derivatives need to be interpolated to the stencil center,  $(x=0, y=0)$ . A simple method for accomplishing this on a  $N=2$  size stencil was shown previously<sup>12</sup>. It is completely generalized here to include any size stencil ( $N \geq 1$ ) using any number of spatial derivatives,  $s \geq 0$ , and at any location such as on solid boundaries which require one-sided stencils.

A one-dimensional osculating spatial interpolant using the primitive variables and their spatial derivatives at the grid points,  $(x_0, x_1, \dots, x_{N-1})$ , is:

$$C_{0,0,0}^{f_N}(x) = \sum_{k=0}^{N-1} \sum_{i=k(s+1)}^{k(s+1)+s} \left[ Q_{i,i} \left( \prod_{j=0}^{k-1} (x - x_j) \right)^{s+1} (x - x_k)^{i-k(s+1)} \right] \quad (4.18)$$

where the  $Q_{i,i}$  are computed as follows:

$$\begin{aligned} &DO \ k = 0, N-1; DO \ i = k(s+1), k(s+1)+s; \\ &DO \ j = 0, i - k(s+1) \\ &\quad Q_{i,j} = (1/j!) \frac{\partial^j f(x_k)}{\partial x^j} \\ &END \ DO; END \ DO; END \ DO \end{aligned} \quad (4.19)$$

$$\begin{aligned} &DO \ m = 1, N-1; DO \ k = 1, N-m \\ &DO \ i = (s+1)(m+k-1), (k+m)s+m+k-1 \\ &DO \ j = i+1 - (s+1)k, i+s+1 - k(s+1) \\ &\quad Q_{i,j} = \frac{(Q_{i,j-1} - Q_{i-1,j-1})}{m\Delta x} \\ &END \ DO; END \ DO; END \ DO; END \ DO \end{aligned} \quad (4.20)$$

The real value of this algebraic form is the simplicity of interpolating the derivatives; Ordinarily, by the product rule of differentiation, each differentiation of Eq (4.18), would result in an equation that grew in size by a factor of  $N$ , up to  $N^O$  terms, and would render

very high order methods intractable. Instead, the following explicit algebraic form avoids this issue:

$$C_{a,0,0}^{f_N(x)} = \sum_{k=0}^{N-1} \sum_{i=k(s+1)}^{k(s+1)+s} Q_{i,i} Z_{a,i,k} \quad (4.21)$$

where the function  $Z_{a,i,k}$  is independent of space and time and can be computed simply as follows:

$$\begin{aligned} &DO \ a = 0, (N-1)(s+1) + s \\ &DO \ i = 0, s \\ &Z_{a,i,0} = (x - x_0)^{(i-a)} \prod_{e=0}^{a-1} (i - e) \\ &END \ DO \\ &END \ DO \\ &DO \ a = 0, (N-1)(s+1) + s \\ &DO \ k = 1, N-1 \\ &DO \ i = k(s+1), k(s+1) + s \\ &Z_{a,i,k} = \\ &\quad \sum_{r_0=0}^a \sum_{r_1=0}^{a-r_0} \dots \sum_{r_{k-1}=0}^{a-\sum_{p=0}^{k-2} r_p} \left( \prod_{\psi=0}^{k-1} \left[ \binom{a - \sum_{p=0}^{\psi-1} r_p}{r_\psi} \right] \right. \\ &\quad * \ (x - x_0)^{(s+1-(a-\sum_{p=0}^{k-1} r_p))} (x - x_k)^{(i-k(s+1)-r_{k-1})} \prod_{\alpha=1}^{k-1} \left[ (x - x_\alpha)^{(s+1-r_{\alpha-1})} \right. \\ &\quad \left. * \prod_{e_0=0}^{(a-\sum_{p=0}^{k-1} r_p)-1} [s+1-e_0] \prod_{e_k=0}^{(r_{k-1}-1)} [i-k(s+1)-e_k] \prod_{\beta=1}^{k-1} \prod_{e_\beta=0}^{(r_{\beta-1}-1)} [s+1-e_\beta] \right] \left. \right) \\ &END \ DO \\ &END \ DO \\ &END \ DO \end{aligned} \quad (4.22)$$

This technique is an extension of Hermitian divided-difference interpolation<sup>21</sup> and it was derived by repeatedly applying the generalized Leibniz' rule in Eq. (3.14). For a fixed stencil size on a uniform grid, Eq. (4.22) needs to be computed only once since it is constant for all space and time. This results in significant computational savings since only the  $Q_{i,i}$  variables require further evaluation as the stencil location changes, but even these stay constant as higher order derivatives are evaluated using Eq. (4.21) on the same stencil.

In addition, higher order interpolants may reuse the information from lower order interpolants to efficiently and adaptively increase local accuracy:

$$C_{a,0,0}^{f_N(x)} = C_{a,0,0}^{f_{N-1}(x)} = \sum_{i=(N-1)(s+1)}^{(N-1)(s+1)+s} Q_{i,i} Z_{a,i,k} \quad (4.23)$$

Multidimensional spatial interpolations are efficiently accomplished using the tensor product of a series of one-dimensional interpolants<sup>11,22</sup>.

## 5. Explicit forms for the Analytic Solution and its Source Terms

Computer algebra can calculate the derivatives of the analytical solution, Eq. (2.5), and the source terms,  $C_{a,b,k-1}^{Q_i}$  in Eq. (3.15), and then automatically write the necessary FORTRAN code<sup>23,13</sup>. However, for very high order algorithms the code becomes too large and complicated for most compilers and therefore it is necessary to develop an explicit form of those expressions.

First define the Lagrangian operator as:

$$L_{a,n} = \frac{\left(\prod_{i=0}^{n-1} \text{Mod}(a, 4) - i\right) \left(\prod_{i=n+1}^a \text{Mod}(a, 4) - i\right)}{\left(\prod_{i=0}^{n-1} (n - i)\right) \left(\prod_{i=n+1}^a (n - i)\right)} \quad (5.24)$$

And then define:

$$C_d^{\cos(\psi z)} = (L_{d,0} \cos(\psi z) - L_{d,1} \sin(\psi z) - L_{d,2} \cos(\psi z) + L_{d,3} \sin(\psi z)) \psi^d \quad (5.25)$$

$$C_d^{\sin(\psi z)} = (L_{d,0} \sin(\psi z) + L_{d,1} \cos(\psi z) - L_{d,2} \sin(\psi z) - L_{d,3} \cos(\psi z)) \psi^d \quad (5.26)$$

where  $\psi$  is any constant,  $d = a, b$ , or,  $k$ , and  $z = k_x \pi x$ ,  $k_y \pi y$ , or,  $k_t \pi t$ .

The derivatives of the assumed analytical solution, Eq. (2.5), may then be written as:

$$C_{a,b,k}^{\rho_{anal}} = \delta_{a0} \delta_{b0} \delta_{k0} c_1 + a_1 (k_x \pi)^a (k_y \pi)^b (k_t \pi)^k C_a^{\cos(k_x \pi x)} C_b^{\cos(k_y \pi y)} C_k^{\cos(k_t \pi t)} \quad (5.27)$$

$$C_{a,b,k}^{u_{anal}} = \delta_{a0} \delta_{b0} \delta_{k0} c_2 + a_2 (k_x \pi)^a (k_y \pi)^b (k_t \pi)^k C_a^{\cos(k_x \pi x)} C_b^{\cos(k_y \pi y)} C_k^{\cos(k_t \pi t)} \quad (5.28)$$

$$C_{a,b,k}^{v_{anal}} = \delta_{a0} \delta_{b0} \delta_{k0} c_3 + a_3 (k_x \pi)^a (k_y \pi)^b (k_t \pi)^k C_a^{\cos(k_x \pi x)} C_b^{\cos(k_y \pi y)} C_k^{\cos(k_t \pi t)} \quad (5.29)$$

$$C_{a,b,k}^{p_{anal}} = \delta_{a0} \delta_{b0} \delta_{k0} c_4 + a_4 (k_x \pi)^a (k_y \pi)^b (k_t \pi)^k C_a^{\cos(k_x \pi x)} C_b^{\cos(k_y \pi y)} C_k^{\cos(k_t \pi t)} \quad (5.30)$$

where  $\delta_{i0}$  is the Kronecker delta.

The source terms involve products of transcendental functions which may be differentiated explicitly using the Leibniz' rule, Eq. (3.14), and are given by:

$$\begin{aligned} C_d^{\text{trig}_1(\psi_1 z) \text{trig}_2(\psi_2 z)} &= \sum_{i=0}^d \binom{d}{i} C_{d-i}^{\text{trig}_1(\psi_1 z)} C_i^{\text{trig}_2(\psi_2 z)} \\ C_d^{\text{trig}_1(\psi_1 z) \text{trig}_2(\psi_2 z) \text{trig}_3(\psi_3 z)} &= \sum_{i=0}^d \sum_{i_2=0}^i \binom{d}{i} \binom{i}{i_2} C_{d-i}^{\text{trig}_1(\psi_1 z)} C_{i-i_2}^{\text{trig}_2(\psi_2 z)} C_{i_2}^{\text{trig}_3(\psi_3 z)} \end{aligned}$$

where  $\text{trig}_i$  is either the cos or sin function and  $\psi_i$  is some constant.

The explicit forms of the derivatives of the source terms using these definitions are listed in Appendix A. In this way, only a few pages of FORTRAN code are required to compute all the necessary very high order derivatives of the analytical solution and source terms.

The Method of Manufactured solutions permits any analytical solution to be imposed, but note that not all solutions and their induced source terms will be explicitly differentiable as they are here. Therefore, additional care is required in the selection of analytical solutions for testing very high order methods for otherwise the resulting code will become unwieldy.

## 6. Stability

Deriving analytical stability conditions for the nonlinear Euler equations is generally difficult<sup>24</sup> and necessary stability conditions (but generally not sufficient) are typically derived by linearizing the conservative governing equations using their constant coefficient quasi-linear form:

$$\frac{\partial U}{\partial t} + A \frac{\partial U}{\partial x} + B \frac{\partial U}{\partial y} = 0 \quad (6.31)$$

where A and B are the flux Jacobians which are held constant for a von Neumann stability analysis. However, in this work, the energy equation is not in conservation form and therefore this form of analysis is not available.

Instead, engineering rules of thumb were applied here and tested numerically. Let the Courant number be defined by<sup>25</sup>:

$$\sigma = \left( \frac{c_x}{\Delta x} + \frac{c_y}{\Delta y} \right) \Delta t \quad (6.32)$$

where  $c_x = \max(\|u\| + c)$ , which is the magnitude of the greatest wave velocity in the x-direction over the entire grid (and similarly for  $c_y$  in the y direction). In addition, for most single-step explicit numerical formulations in two-dimensions the Courant number must be  $\sigma \leq 0.5$  to maintain numerical stability<sup>26</sup>.

Since the Cauchy-Kowalewski procedure is essentially a Taylor series method, a simple model problem may be used to demonstrate numerical stability limits. For example<sup>27</sup>, if the test equation (for  $\lambda < 0$ ):

$$y' = \lambda y \quad (6.33)$$

is time advanced using an  $n^{th}$  order Taylor series in time, it will be stable if the following series (which is generally convergent as  $n \rightarrow \infty$ ) is satisfied:

$$\|1 + \Delta t \lambda + \left(\frac{1}{2}\right) \Delta t^2 \lambda^2 + \dots + \left(\frac{1}{n!}\right) \Delta t^n \lambda^n\| < 1 \quad (6.34)$$

At low accuracy the maximum stable time step will vary, but it soon approaches a fixed value as the accuracy increases. In general, however, high-order explicit schemes usually have more severe stability restrictions on  $\Delta t$ <sup>28</sup> and this is evident in Table 3.

The following time step size was found to provide numerically stable solutions for up to 15<sup>th</sup> order accuracy on a uniform Cartesian grid:

$$\Delta t = \frac{\sigma}{\max_{i,j} \left( \frac{\|u_{i,j}\| + \|v_{i,j}\| + 2c}{\Delta x} \right)} \quad (6.35)$$

where the speed of sound is defined by :

$$c = \sqrt{\gamma \frac{p(x,y,t)}{\rho(x,y,t)}} = \sqrt{\frac{\gamma(a_4 + c_4)}{c_1 - a_1}} \quad (6.36)$$

Numerical instability did occur when the solution was under-resolved as shown in Table 6. Here, the 3<sup>rd</sup> order method became unstable by the time  $t = 799.98$ , but the 5<sup>th</sup>, 7<sup>th</sup>, and 9<sup>th</sup> order methods were stable to at least 50 wavelengths of propagation. This suggests that nonlinear aliasing effects can be delayed significantly by increasing the accuracy of the simulation. Similar results are seen in Table 3.

In practical applications, it will be necessary to filter out the inevitable unresolved high frequencies. However, it is encouraging to observe that despite the very low numerical dissipation, filters were not required in this work.

## 7. Numerical Results

A comparison of the effects of design accuracy on the efficiency and resolution of the overall simulation was performed by using a  $2 \times 2$  stencil to produce  $2s + 1$  order schemes. These schemes were used to propagate a single wavelength the following distances:

- A single time step as shown in Tables 1-2,
- Five wavelengths as shown in Tables 3-5,
- Fifty wavelengths as shown in Table 6.

The wavenumbers, amplitudes, and displacements defined in Eq. (2.5) were set to  $k_i = 1$ ,  $a_i = 1$  and  $c_i = 2$ , respectively, except in Table 6 which has  $c_i = 10$ .

The expected error,  $y_s$ , of a method with design accuracy  $O = 2s + 1$  can be extrapolated using the data in Table 1 since the design accuracy determines the rate of error change as the grid is refined. Starting with the known numerical error,  $e_s$ , in Table 1 with  $x_s = 2$ , the following relationship:

$$y_s = \frac{e_s}{(2^{O(x_s-2)})} \quad (7.37)$$

will provide the expected error at  $x_s = \lceil \log_2 \frac{\text{Gridpoints}}{\text{Wavelength}} \rceil$ . Figure 1 plots this equation. Note that, due to computer roundoff, errors less than  $10^{-15}$  are only possible if quadruple precision floating point numbers are available.

By solving for  $x_s$  in Eq. (7.37), it is possible to estimate the number of grid points required to meet a particular error level  $y_s$ .

$$x_s = \frac{\log_2 e_s - \log_2 y_s}{O} + 2 \quad (7.38)$$

The total number of grid points in a two-dimensional unit domain required for propagating a single wavelength is then given by  $(2^{x_s} + 1)^2$ . In addition, each grid point contains the four primitive variables  $\rho$ ,  $p$ ,  $u$ , and  $v$  as well as their,  $O^2$ , spatial derivatives resulting in a total memory cost of:

$$\text{cost}_{\text{memory}} = 4(2s + 1)^2(2^{x_s} + 1)^2 \quad (7.39)$$

Table 1. Numerical error at  $t=0.05$ ,  $\sigma=0.25$ , 4 grid points per wavelength, 25 grid points  
 $c_i = 2, a_i = 1, k_i = 1, \Delta t = .05$ , Single Step

s	Order	Error, $e_s$	Seconds/Grid Point, $c_s$
1	3	$1.57 \cdot 10^{-2}$	.0012
2	5	$2.74 \cdot 10^{-4}$	.015
3	7	$2.54 \cdot 10^{-6}$	.1172
4	9	$1.44 \cdot 10^{-8}$	.678
5	11	$5.40 \cdot 10^{-11}$	3
6	13	$1.42 \cdot 10^{-13}$	11
7	15	$1.78 \cdot 10^{-15}$	36
8	17	$3.55 \cdot 10^{-15}$	95
9	19	roundoff	238

Table 2. Numerical error at  $t=0.05$ ,  $\sigma=0.5$ , 8 grid points per wavelength, 81 grid points  
 $c_i = 2, a_i = 1, k_i = 1, \Delta t = .05$ , Single Step

s	Order	Error, $e_s$	Seconds/Grid Point, $c_s$
1	3	$9.55 \cdot 10^{-4}$	.0016
2	5	$3.95 \cdot 10^{-6}$	.0183
3	7	$1.18 \cdot 10^{-8}$	.1449
4	9	$3.46 \cdot 10^{-11}$	.839
5	11	$4.85 \cdot 10^{-14}$	3.7
6	13	roundoff	13

Table 3. Numerical error at  $t=10$ ,  $\sigma=0.5$ , 4 grid points per wavelength, 25 grid points  
 $c_i = 2, a_i = 1, k_i = 1, \Delta t = 2.4755 \cdot 10^{-2}$ , 403 Steps, 5 wavelengths

s	Order	Error, $e_s$	Total Seconds	Seconds / Grid Point, $c_s$
1	3	UNSTABLE	UNSTABLE	UNSTABLE
2	5	$1.13 \cdot 10^{-1}$	188.22	7
3	7	$1.67 \cdot 10^{-3}$	1497.53	59
4	9	$1.50 \cdot 10^{-5}$	8610.51	344
5	11	$5.21 \cdot 10^{-6}$	39125.30	1565
6	13	UNSTABLE	UNSTABLE	UNSTABLE

Table 4. Numerical error at  $t=10$ ,  $\sigma=0.5$ , 8 grid points per wavelength, 81 grid points  
 $c_i = 2, a_i = 1, k_i = 1, \Delta t = 1.2377 \cdot 10^{-2}$ , 807 Steps, 5 wavelengths

s	Order	Error, $e_s$	Total Seconds	Seconds / Grid Point, $c_s$
1	3	$1.47 \cdot 10^{-0}$	166.90	2
2	5	$3.95 \cdot 10^{-3}$	1392.09	17
3	7	$9.73 \cdot 10^{-6}$	13328.04	164
4	9	$1.64 \cdot 10^{-8}$	64907.56	801
5	11	$1.45 \cdot 10^{-11}$	275939.53	3406
6	13	$8.29 \cdot 10^{-13}$	999763.10	12342

Table 5. Numerical error at  $t=10$ ,  $\sigma=0.5$ , 16 grid points per wavelength, 289 grid points  $c_i = 2, a_i = 1, k_i = 1, \Delta t = 6.188 \cdot 10^{-3}$ , 1615 Steps, 5 wavelengths

s	Order	Error, $e_s$	Total Seconds	Seconds / Grid Point, $c_s$
1	3	$1.48 \cdot 10^{-1}$	935	3.2
2	5	$2.54 \cdot 10^{-4}$	10198	35
3	7	$2.02 \cdot 10^{-7}$	80299	277
4	9	$1.29 \cdot 10^{-10}$	463537	1603

Table 6. Numerical error at  $t=100$ ,  $\sigma=0.5$ , 8 grid points per wavelength, 81 grid points  $c_i = 10, a_i = 1, k_i = 1, \Delta t = 5.077 \cdot 10^{-3}$ , 19692 Steps, 50 wavelengths

s	Order	Error, $e_s$	Total Seconds	Seconds / Grid Point, $c_s$
1	3	UNSTABLE	UNSTABLE	UNSTABLE
2	5	$1.05 \cdot 10^{-3}$	65407	807
3	7	$8.65 \cdot 10^{-6}$	406932	5023
4	9	$1.08 \cdot 10^{-8}$	1662013	20518

In addition, the total time per grid point for each method,  $c_s$  is shown in Table 1. However, this time reflects the cost of a single time step. As  $x_s$  increases (grid density) the maximum stable time step will decrease for explicit methods (to maintain CFL condition). This is accounted for by including the factor,  $2^{x_s-2}$ , for a total time cost of:

$$cost_{time} = c_s 2^{x_s-2} (2^{x_s} + 1)^2 \quad (7.40)$$

Fig. 2 compares the relative costs of propagating a wavelength over a range of error tolerances,  $10^{-15} \leq y_s \leq 10^0$ .

Generally, a direct acoustical simulation requires an error of less than  $10^{-6}$  to capture both the acoustical and mean flow phenomena. By this standard, the correct propagation of a wave cannot be realistically achieved with a  $3^{rd}$  order method because of the exponential growth in its cost as shown in Fig. 2. In addition, additional error will accumulate at each time step and as Tables 3-5 indicate, a higher order method is required to propagate out to a distance of 5 wavelengths. As Table 6 shows, at least  $7^{th}$  order accuracy is required for propagating 50 wavelengths with 8 grid points per wavelength.

## 8. Conclusions

The procedures shown here provide arbitrary accuracy in both space and time for nonlinear Euler simulations using only a few pages of FORTRAN code. The usefulness of these procedures hinges on the required accuracy of a given simulation.

If an accuracy of  $10^{-10}$  is required at time  $t = 0.05$  to maintain a given error tolerance at some future time, then Fig. 2 indicates the  $3^{rd}$  and  $5^{th}$  order methods could not accomplish this because of exponential cost growth; Here, the optimal design accuracy is  $11^{th}$  order. In addition, if the error tolerance is decreased to  $10^{-15}$  because of the desire to run the simulation over a larger time period, it is most optimal to use a  $15^{th}$  or higher order method.



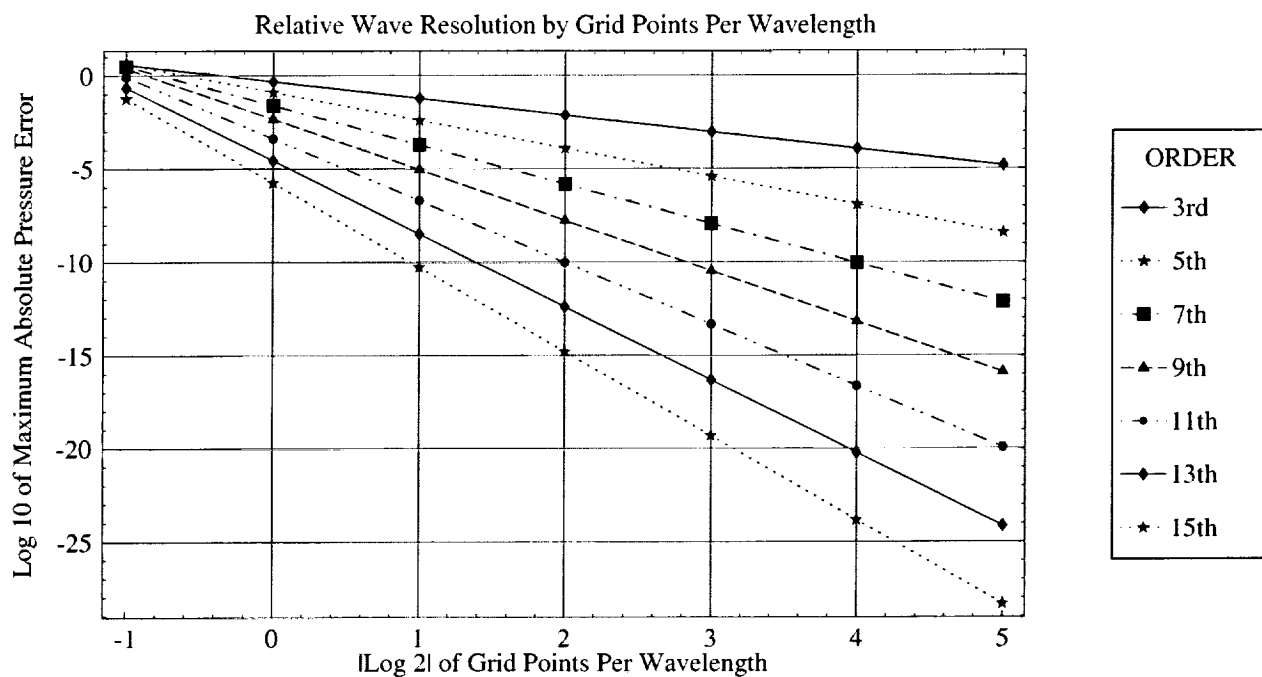


Figure 1: Resolution Studies

These results show that a fixed order algorithm approaches its maximal utility and then quickly becomes useless. The nonlinear implementation of the MESA schemes described here can be used to adjust the design accuracy arbitrarily and it offers an opportunity for solving the more demanding acoustical simulations in smooth flows.

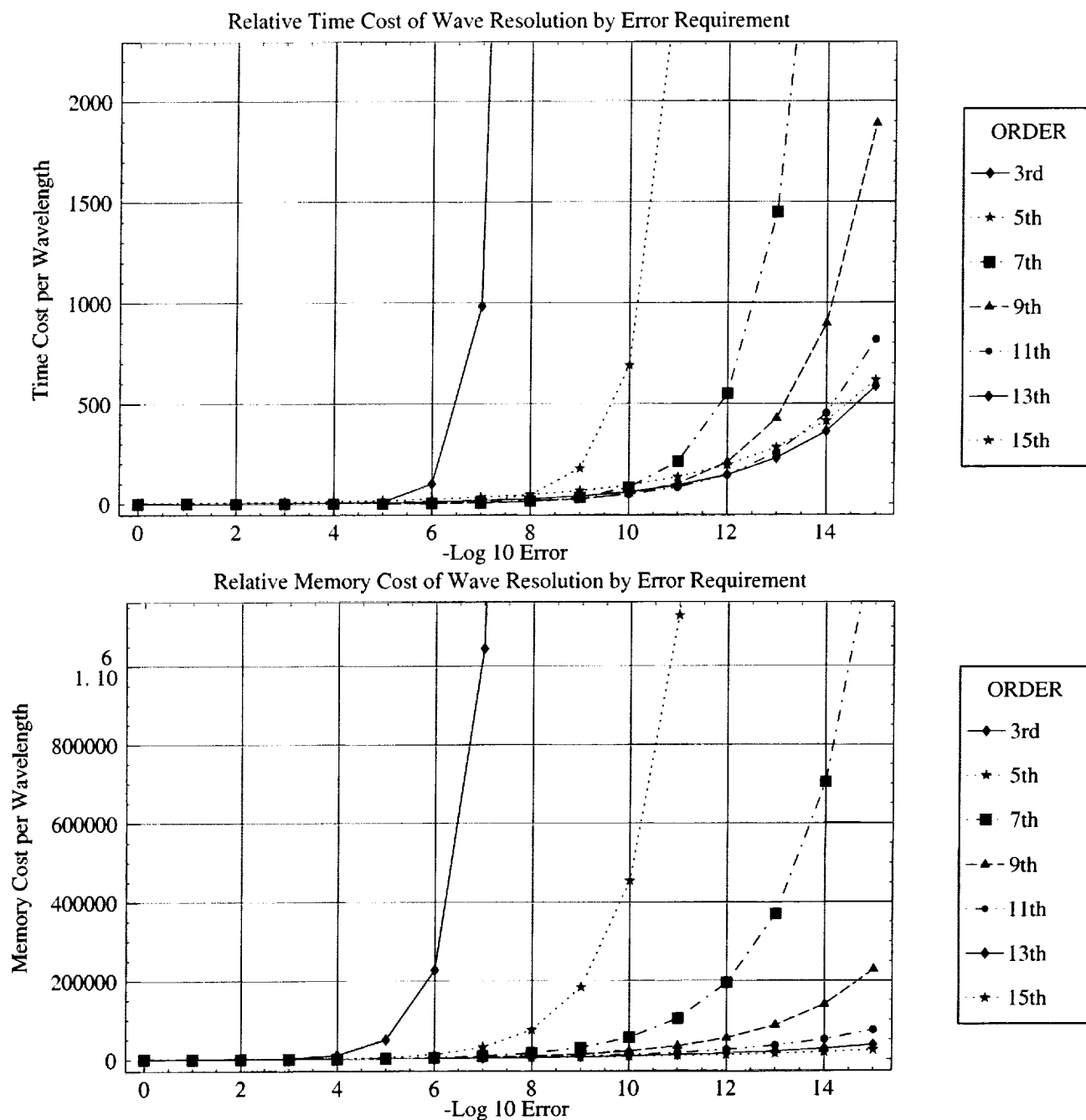


Figure 2: Time and Memory Efficiency Comparison For Errors Up To  $10^{-15}$

## Appendix A

Sources terms  $Q_1$ ,  $Q_2$ ,  $Q_3$ , and  $Q_4$  in Eqs. (2.1, 2.2, 2.3, 2.4) and their explicit derivatives are defined below using the definitions in section 5:

$$\begin{aligned}
C_{a,b,k}^{Q_1} = & -a_1 k_t \pi k_{xyt} C_a^{\cos(k_x \pi x)} C_b^{\cos(k_y \pi y)} C_k^{\cos(k_t \pi t)} - \\
& a_2 c_1 k_x \pi k_{xyt} C_k^{\cos(k_t \pi t)} C_b^{\cos(k_y \pi y)} C_a^{\sin(k_x \pi x)} - \\
& a_1 c_2 k_x \pi k_{xyt} C_k^{\cos(k_t \pi t)} C_b^{\cos(k_y \pi y)} C_a^{\sin(k_x \pi x)} - \\
& a_1 a_2 k_x \pi k_{xyt} C_k^{\cos(k_t \pi t)^2} C_b^{\cos(k_y \pi y)^2} C_a^{\sin(2k_x \pi x)} - \\
& a_3 c_1 k_y \pi k_{xyt} C_k^{\cos(k_t \pi t)} C_a^{\cos(k_x \pi x)} C_b^{\sin(k_y \pi y)} - \\
& a_1 c_3 k_y \pi k_{xyt} C_k^{\cos(k_t \pi t)} C_a^{\cos(k_x \pi x)} C_b^{\sin(k_y \pi y)} - \\
& a_1 a_3 k_y \pi k_{xyt} C_k^{\cos(k_t \pi t)^2} C_a^{\cos(k_x \pi x)^2} C_b^{\sin(k_y \pi y)} \\
C_{a,b,k}^{Q_2} = & -a_4 k_t \pi k_{xyt} C_a^{\cos(k_x \pi x)} C_b^{\cos(k_y \pi y)} C_k^{\sin(k_t \pi t)} - \\
& a_4 c_2 k_x \pi k_{xyt} C_a^{\sin(k_x \pi x)} C_b^{\cos(k_y \pi y)} C_k^{\cos(k_t \pi t)} - \\
& a_2 c_4 \gamma k_x \pi k_{xyt} C_a^{\sin(k_x \pi x)} C_b^{\cos(k_y \pi y)} C_k^{\cos(k_t \pi t)} - \\
& a_2 a_4 k_x \pi k_{xyt} C_k^{\cos(k_t \pi t)^2} C_b^{\cos(k_y \pi y)^2} C_a^{\cos(k_x \pi x) \sin(k_x \pi x)} - \\
& \gamma a_2 a_4 k_x \pi k_{xyt} C_k^{\cos(k_t \pi t)^2} C_b^{\cos(k_y \pi y)^2} C_a^{\cos(k_x \pi x) \sin(k_x \pi x)} - \\
& a_4 c_3 k_y \pi k_{xyt} C_a^{\cos(k_x \pi x)} C_b^{\sin(k_y \pi y)} C_k^{\cos(k_t \pi t)} - \\
& a_3 c_4 \gamma k_y \pi k_{xyt} C_a^{\cos(k_x \pi x)} C_b^{\sin(k_y \pi y)} C_k^{\cos(k_t \pi t)} - \\
& \frac{a_3 a_4}{2} k_y \pi k_{xyt} C_k^{\cos(k_t \pi t)^2} C_a^{\cos(k_x \pi x)^2} C_b^{\sin(2k_y \pi y)^2} - \\
& \gamma \frac{a_3 a_4}{2} k_y \pi k_{xyt} C_k^{\cos(k_t \pi t)^2} C_a^{\cos(k_x \pi x)^2} C_b^{\sin(2k_y \pi y)^2} \\
C_{a,b,k}^{Q_3} = & -a_2 c_1 k_t \pi k_{xyt} C_a^{\cos(k_x \pi x)} C_b^{\cos(k_y \pi y)} C_k^{\sin(k_t \pi t)} - \\
& a_1 c_2 k_t \pi k_{xyt} C_a^{\cos(k_x \pi x)} C_b^{\cos(k_y \pi y)} C_k^{\sin(k_t \pi t)} - \\
& 2a_1 a_2 k_t \pi k_{xyt} C_a^{\cos(k_x \pi x)^2} C_b^{\cos(k_y \pi y)^2} C_k^{\cos(k_t \pi t) \sin(k_t \pi t)} - \\
& a_4 k_x \pi k_{xyt} C_k^{\cos(k_t \pi t)} C_b^{\cos(k_y \pi y)} C_a^{\sin(k_x \pi x)} - \\
& 2a_2 c_1 c_2 k_x \pi k_{xyt} C_k^{\cos(k_t \pi t)} C_b^{\cos(k_y \pi y)} C_a^{\sin(k_x \pi x)} - \\
& a_1 c_2^2 k_x \pi k_{xyt} C_k^{\cos(k_t \pi t)} C_b^{\cos(k_y \pi y)} C_a^{\sin(k_x \pi x)} - \\
& 2a_2^2 c_1 k_x \pi k_{xyt} C_k^{\cos(k_t \pi t)^2} C_b^{\cos(k_y \pi y)^2} C_a^{\cos(k_x \pi x) \sin(k_x \pi x)} - \\
& 4a_1 a_2 c_2 k_x \pi k_{xyt} C_k^{\cos(k_t \pi t)^2} C_b^{\cos(k_y \pi y)^2} C_a^{\cos(k_x \pi x) \sin(k_x \pi x)} - \\
& 3a_1 a_2^2 k_x \pi k_{xyt} C_k^{\cos(k_t \pi t)^3} C_b^{\cos(k_y \pi y)^3} C_a^{\cos(k_x \pi x) \cos(k_x \pi x) \sin(k_x \pi x)} - \\
& a_3 c_1 c_2 k_y \pi k_{xyt} C_k^{\cos(k_t \pi t)} C_a^{\cos(k_x \pi x)} C_b^{\sin(k_y \pi y)} - \\
& a_2 c_1 c_3 k_y \pi k_{xyt} C_k^{\cos(k_t \pi t)} C_a^{\cos(k_x \pi x)} C_b^{\sin(k_y \pi y)} -
\end{aligned}$$

$$\begin{aligned}
& a_1 c_2 c_3 k_y \pi k_{xyt} C_k^{\cos(k_t \pi t)} C_a^{\cos(k_x \pi x)} C_b^{\sin(k_y \pi y)} - \\
& 2a_2 a_3 c_1 k_y \pi k_{xyt} C_k^{\cos(k_t \pi t)^2} C_a^{\cos(k_x \pi x)^2} C_b^{\cos(k_y \pi y) \sin(k_y \pi y)} - \\
& 2a_1 a_3 c_2 k_y \pi k_{xyt} C_k^{\cos(k_t \pi t)^2} C_a^{\cos(k_x \pi x)^2} C_b^{\cos(k_y \pi y) \sin(k_y \pi y)} - \\
& 2a_1 a_2 c_3 k_y \pi k_{xyt} C_k^{\cos(k_t \pi t)^2} C_a^{\cos(k_x \pi x)^2} C_b^{\cos(k_y \pi y) \sin(k_y \pi y)} - \\
& 3a_1 a_2 a_3 k_y \pi k_{xyt} C_k^{\cos(k_t \pi t)^3} C_a^{\cos(k_x \pi x)^3} C_b^{\cos(k_y \pi y) \cos(k_y \pi y) \sin(k_y \pi y)} \\
C_{a,b,k}^{Q_4} = & -a_3 c_1 k_t \pi k_{xyt} C_a^{\cos(k_x \pi x)} C_b^{\cos(k_y \pi y)} C_k^{\sin(k_t \pi t)} - \\
& a_1 c_3 k_t \pi k_{xyt} C_a^{\cos(k_x \pi x)} C_b^{\cos(k_y \pi y)} C_k^{\sin(k_t \pi t)} - \\
& 2a_1 a_3 k_t \pi k_{xyt} C_a^{\cos(k_x \pi x)^2} C_b^{\cos(k_y \pi y)^2} C_k^{\cos(k_t \pi t) \sin(k_t \pi t)} - \\
& a_3 c_1 c_2 k_x \pi k_{xyt} C_k^{\cos(k_t \pi t)} C_b^{\cos(k_y \pi y)} C_a^{\sin(k_x \pi x)} - \\
& a_2 c_1 c_3 k_x \pi k_{xyt} C_k^{\cos(k_t \pi t)} C_b^{\cos(k_y \pi y)} C_a^{\sin(k_x \pi x)} - \\
& a_1 c_2 c_3 k_x \pi k_{xyt} C_k^{\cos(k_t \pi t)} C_b^{\cos(k_y \pi y)} C_a^{\sin(k_x \pi x)} - \\
& 2a_2 a_3 c_1 k_x \pi k_{xyt} C_k^{\cos(k_t \pi t)^2} C_b^{\cos(k_y \pi y)^2} C_a^{\cos(k_x \pi x) \sin(k_x \pi x)} - \\
& 2a_1 a_3 c_2 k_x \pi k_{xyt} C_k^{\cos(k_t \pi t)^2} C_b^{\cos(k_y \pi y)^2} C_a^{\cos(k_x \pi x) \sin(k_x \pi x)} - \\
& 2a_1 a_2 c_3 k_x \pi k_{xyt} C_k^{\cos(k_t \pi t)^2} C_b^{\cos(k_y \pi y)^2} C_a^{\cos(k_x \pi x) \sin(k_x \pi x)} - \\
& 3a_1 a_2 a_3 k_x \pi k_{xyt} C_k^{\cos(k_t \pi t)^3} C_b^{\cos(k_y \pi y)^3} C_a^{\cos(k_x \pi x) \cos(k_x \pi x) \sin(k_x \pi x)} - \\
& a_4 k_y \pi k_{xyt} C_k^{\cos(k_t \pi t)} C_a^{\cos(k_x \pi x)} C_b^{\sin(k_y \pi y)} - \\
& 2a_3 c_1 c_3 k_y \pi k_{xyt} C_k^{\cos(k_t \pi t)} C_a^{\cos(k_x \pi x)} C_b^{\sin(k_y \pi y)} - \\
& a_1 c_3^2 k_y \pi k_{xyt} C_k^{\cos(k_t \pi t)} C_a^{\cos(k_x \pi x)} C_b^{\sin(k_y \pi y)} - \\
& 2a_3^2 c_1 k_y \pi k_{xyt} C_k^{\cos(k_t \pi t)^2} C_a^{\cos(k_x \pi x)^2} C_b^{\cos(k_y \pi y) \sin(k_y \pi y)} - \\
& 4a_1 a_3 c_2 k_y \pi k_{xyt} C_k^{\cos(k_t \pi t)^2} C_a^{\cos(k_x \pi x)^2} C_b^{\cos(k_y \pi y) \sin(k_y \pi y)} - \\
& 3a_1 a_3^2 k_y \pi k_{xyt} C_k^{\cos(k_t \pi t)^3} C_a^{\cos(k_x \pi x)^3} C_b^{\cos(k_y \pi y) \cos(k_y \pi y) \sin(k_y \pi y)}
\end{aligned}$$

## References

1. Lee, J., Cho, K., Lee, S. "Application of acoustic analogy to automotive engine-cooling fan noise prediction", AIAA J., Vol. 38 No. 6 pp. 1095-1098 JUN 2000
2. Colonius T, Lele S.K., Moin P. "Sound generation in a mixing layer" J. of Fluid Mechanics, Vol. 330 pp. 375-409 JAN 10 1997
3. Morris P.J., Long L.N., Bangalore A., Wang Q.Z. "A parallel three-dimensional computational aeroacoustics method using nonlinear disturbance equations", J. Comp. Phys., Vol.133 No.1, pp. 56-74 MAY 1 1997
4. Povitsky, A., and Morris, P.J., "A Higher-Order Compact Method in Space and Time Based On Parallel Implementation of the Thomas Algorithm", J. Comp. Phys., 161, 182-203 (2000)
5. Yee, H.C., Sandham, N.D., and Djomehri, M.J. "Low-dissipative High-Order Shock-Capturing Methods Using Characteristic-Based Filters", J. Comp. Phys., Vol. 150, 199-238, 1999.

6. Tam, C.K.W. "Computational Aeroacoustics: Issues and Methods", *AIAA J.*, Vol. 33, No. 10, Oct. 1995.
7. Harari, I., Avraham, D., "High-order finite element methods for acoustic problems", *J. Comp. Acoust.*, Vol. 5: (1) pp. 33-51 MAR 1997
8. Goodrich, J.W. "Accurate Finite Difference Algorithms for Computational Aeroacoustics", *Computational Fluid Dynamics Review 1998*, edited by M.M Hafez and K. Oshima, World Scientific, Singapore, 1998.
9. Goodrich, J.W., "A Comparison of Numerical Methods for Computational Aeroacoustics", *AIAA 99-1943*, May 1999.
10. Goodrich, J.W. "A Low Order and a High Order Solution for a Converging-Diverging Nozzle Problem", *NASA/CP-2000-209790*, pp. 265-271, Aug. 2000.
11. Dyson, R.W., Goodrich, J.W. "A Very High Order, Adaptable MESA Implementation for Aeroacoustic Computations", *NASA/TM-2000-209944*, April 2000.
12. Dyson, R.W., Goodrich, J.W., "Automated Approach to Very High Order Aeroacoustics Computations", *AIAA J.*, March 2001.
13. Roache, P.J., "Verification of codes and calculations", *AIAA J.*, Vol. 36: (5) pp. 696-702, MAY 1998
14. Laney, C.B., *Computational Gasdynamics*, Cambridge University Press, 1998, pp. 9-10.
15. Thompson, K.W., *Time-Dependent Boundary Conditions for Hyperbolic Systems, II*, *J. Comp. Phys.*, Vol. 89, pp. 439-461, 1990
16. Kambiz Salari, Patrick Knupp, Code Verification by the Method of Manufactured Solutions, Sandia National Laboratory Report, SAND2000-1444, June 2000
17. Roache, P.J., *Verification and Validation in Computational Science and Engineering*, Hermosa Publishers, 1998, pp.65-90
18. Laney, C.B., *Computational Gasdynamics*, Cambridge University Press, 1998, pp. 316-323.
19. Garabedian, P.R., "The Method of Power Series: The Cauchy-Kowalewski Theorem," *Partial Differential Equations*, 2<sup>nd</sup> ed., Chelsea, New York, 1986, pp. 16.
20. Fulks, W, *Advanced Calculus: An introduction to analysis*, John Wiley & Sons, 1978, pp. 109.
21. Powell, M.J.D., "Hermite Interpolation," *Approximation Theory and Methods*, Cambridge University Press, 1981, pp. 53-57.
22. Botha, J.F., and Pinder, G.F., "Two-Dimensional Polynomial Approximation," *Fundamental Concepts in the Numerical Solution of Differential Equations*, Wiley, New York, 1983, pp. 17-22.
23. Akers R.L., Baffes P., Kant E., Randall C., Steinberg S., Young R.L. "Automatic synthesis of numerical codes for solving partial differential equations", *Math. Comp. in Simulation*, Vol. 45 No. 1-2, pp. 3-22, JAN 31 1998
24. Hirsch, C. *Numerical Computation of Internal and External Flows: Volume 2: Computational Methods for Inviscid and Viscous Flows*, John Wiley & Sons, 1988, pp. 379-384.
25. Thompson, K.W., *Lecture Series in Computational Fluid Dynamics*, NASA TM 100010, August 1987.
26. Hoffman, K.A., Chiang, S.T., *Computational Fluid Dynamics For Engineers – Volume II*, Engineering Education System, 1993, pp. 15.
27. Burden, R.L., Faires, J.D. *Numerical Analysis, Fourth Edition*, PWS-Kent Publishing Company, Boston, 1989, pp. 304.
28. Fletcher, C.A.J. "Computational Techniques for Fluid Dynamics 1: Fundamental and General Techniques", Springer-Verlag, Berlin, Heidelberg, 1991,

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE July 2001	3. REPORT TYPE AND DATES COVERED Technical Memorandum		
4. TITLE AND SUBTITLE  Technique for Very High Order Nonlinear Simulation and Validation		5. FUNDING NUMBERS  WU-781-30-11-0		
6. AUTHOR(S)  Rodger W. Dyson				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)  National Aeronautics and Space Administration John H. Glenn Research Center at Lewis Field Cleveland, Ohio 44135-3191		8. PERFORMING ORGANIZATION REPORT NUMBER  E-12839		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)  National Aeronautics and Space Administration Washington, DC 20546-0001		10. SPONSORING/MONITORING AGENCY REPORT NUMBER  NASA TM-2001-210985		
11. SUPPLEMENTARY NOTES  Responsible person, Rodger W. Dyson, organization code 5940, 216-433-9083.				
12a. DISTRIBUTION/AVAILABILITY STATEMENT  Unclassified - Unlimited Subject Category: 71  Available electronically at <a href="http://gltrs.grc.nasa.gov/GLTRS">http://gltrs.grc.nasa.gov/GLTRS</a> This publication is available from the NASA Center for AeroSpace Information, 301-621-0390.			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words)  Finding the sources of sound in large nonlinear fields via direct simulation currently requires excessive computational cost. This paper describes a simple technique for efficiently solving the multidimensional nonlinear Euler equations that significantly reduces this cost and demonstrates a useful approach for validating high order nonlinear methods. Up to 15th order accuracy in space and time methods were compared and it is shown that an algorithm with a fixed design accuracy approaches its maximal utility and then its usefulness exponentially decays unless higher accuracy is used. It is concluded that at least a 7th order method is required to efficiently propagate a harmonic wave using the nonlinear Euler equations to a distance of 5 wavelengths while maintaining an overall error tolerance that is low enough to capture both the mean flow and the acoustics.				
14. SUBJECT TERMS  High order; Nonlinear Euler; Multidimensional; Acoustics			15. NUMBER OF PAGES 23	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT	