Verification of a Viscous Computational Aeroacoustics Code using External Verification Analysis

AIAA Paper #2015-2225

Daniel Ingraham[1] Ray Hixon[2]

[1] Acoustics Branch
NASA Glenn Research Center
Cleveland, Ohio

[2] Mechanical, Industrial, & Manufacturing Engineering Department
University of Toledo
Toledo, Ohio

21st AIAA/CAES Aeroacoustics Conference
June 22nd, 2015

Motivation

- Computational Aeroacoustics (CAA) codes are increasingly used to simulate complex physics, make design decisions, etc.
- All software has bugs[1]. How can we know we're getting the right answer out of our CAA code?

Code Verification

- Code verification: “are we solving the equations right?” All about the math/numerics. Goal is to detect bugs.
- Gold-standard of code verification: the order-of-accuracy test. Does the error behave as we expect, i.e., converge at the rate of the order-of-accuracy of the code's schemes, p?
  - $\epsilon \approx Ah^p$
- So, we need the code's error, i.e., the difference between the code's solution and a reference solution.
- Where do we get a reference solution?
  - Method of Exact Solutions (MES)
  - Method of Manufactured Solutions (MMS)

Reference Solutions for Code Verification: State of the Art

Method of Exact Solutions (MES)

Look in a textbook (or paper, etc.) for a solution to the PDE the code solves.

- Advantages: Simple, no modifications to PDE code.
- Disadvantages: available solutions tend to be either tricky to evaluate numerically (infinite series, integrals, etc.), or too simple (terms go to zero, etc.), or both.

Method of Manufactured Solutions[2, 3] (MMS)

Start with the solution you want, then change the PDE to make it work.

- Advantages: General and flexible — user chooses the solution.
- Disadvantages: Source terms must be added to the PDE, and thus the code, which can be quite complicated for non-trivial PDEs.

The Ideal Code Verification Method

- The complication of MMS source terms appears to limit the popularity of MMS.
- Characteristics of an ideal code verification method?
  - Flexibility of MMS: user has control over the form of the solution.
  - Unobtrusiveness of MES: no modifications of PDE, or code.
- External Verification Analysis (EVA) is intended to fulfill these requirements.

Goals of this work

- Show how EVA can provide a reference solution for the nonlinear Navier-Stokes equations suitable for code verification,
- Use EVA to verify a high-order, viscous Computational Aeroacoustics code.

EVA: An Approximate Solution to the Cauchy Problem

- Express PDE as a Cauchy (initial value) problem,

$$\frac{\partial u}{\partial t} = H\left(x, t, \frac{\partial^\alpha u}{\partial x^\alpha}\right); \quad u(x, t = 0) = v(x)$$

- Use a Taylor series to approximate $u(x, t)$ as

$$u\,(x, t) \approx u(x, 0) + t\frac{\partial u(x, 0)}{\partial t} + \frac{1}{2}t^2\frac{\partial^2 u(x, 0)}{\partial t^2} + \cdots \frac{t^p}{p!}\frac{\partial^p u}{\partial t^p}.$$

- The EVA approach: use the above truncated series combined with an analytic initial condition as a reference solution for code verification.
  - Cauchy-Kowalewski (CK) recursion: repeatedly differentiate governing equation, eventually expressing. . .

$$\frac{\partial^n u}{\partial t^n} \to \frac{\partial^m u}{\partial x^m}$$

  - Differentiate the analytic initial condition exactly.
  - Details in the paper!

A Recurrence Relation for (Derivatives of) the Navier-Stokes Equations

- Recurrence relation for the continuity equation:

$$\frac{\partial^{b+d+n+1}\sigma}{\partial x^b\,\partial y^d\,\partial t^{n+1}} = -\sum_{a=0}^{b}\binom{b}{a}\sum_{c=0}^{d}\binom{d}{c}\sum_{m=0}^{n}\binom{n}{m}\Bigg[\frac{\partial^{b-a+d-c+n-m}u}{\partial x^{b-a}\,\partial y^{d-c}\,\partial t^{n-m}}\,\frac{\partial^{a+1+c+m}\sigma}{\partial x^{a+1}\,\partial y^{c}\,\partial t^{m}} + \frac{\partial^{b-a+d-c+n-m}v}{\partial x^{b-a}\,\partial y^{d-c}\,\partial t^{n-m}}\,\frac{\partial^{a+c+1+m}\sigma}{\partial x^{a}\,\partial y^{c+1}\,\partial t^{m}} - \frac{\partial^{b-a+d-c+n-m}\sigma}{\partial x^{b-a}\,\partial y^{d-c}\,\partial t^{n-m}}\left(\frac{\partial^{a+1+c+m}u}{\partial x^{a+1}\,\partial y^{c}\,\partial t^{m}} + \frac{\partial^{a+c+1+m}v}{\partial x^{a}\,\partial y^{c+1}\,\partial t^{m}}\right)\Bigg].$$

- Using the specific volume $\sigma = \frac{1}{\rho}$ (makes the math easier).
- Not as bad as it looks. Like Dyson and Goodrich[4], using Leibniz rule to differentiate products.

The Initial Condition

- Final piece of the puzzle: initial condition (IC). EVA uses an analytic IC.
- Since EVA solves a Cauchy (read: initial value) problem, no particular boundary conditions (BCs) are enforced. So an IC with a "footprint" helps us avoid problems with the PDE code's BCs.
- The IC currently implemented in EVA tool:

$$\begin{aligned}\phi(x,y,z) = {} & \overline{\phi} \\ & + \widetilde{\phi}\exp\left(-\frac{\log(2)}{b^2}\left[(x-x_0)^2+(y-y_0)^2+(z-z_0)^2\right]\right) \\ & \cdot \sin\left(k_x x + k_y y + k_z z + \theta\right) \\ & = \overline{\phi} + \widetilde{\phi}E(x,y,z)\cdot I(x,y,z)\end{aligned}$$

- Need to be able to calculate arbitrary-order derivatives of the initial condition.
- Benefit from analytic derivatives: EVA solution is completely *unaffected* by grid/mesh used!

The Computational Aeroacoustics Code: BASS

- Broadband Aeroacoustic Stator Simulator from NASA Glenn
- High-order, parallel, block-structured finite-difference Computational Aeroacoustics solver with explicit, optimized time-marching
- Inviscid or **Viscous**, 2D or **3D**
- Spatial differencing schemes
  - E_2: Explicit 2nd-order
  - E_6: Explicit 6th-order
  - DRP: Tam & Webb Dispersion Relation Preserving[5]
  - C_6: Hixon's[6] prefactored form of Lele's[7] compact 6th-order
- Time marching schemes
  - RK4L, RK5L: Jameson[8] four- and five-stage Runge-Kutta,
  - RK56: Stanescu and Habashi[9] five/six stage Runge-Kutta,
  - RK7S, RK67: Allampalli et al.[10] High-Accuracy Large-step Explicit Runge-Kutta (HALE-RK) seven and six/seven-stage schemes.
- Modified BASS to use constant viscous properties.

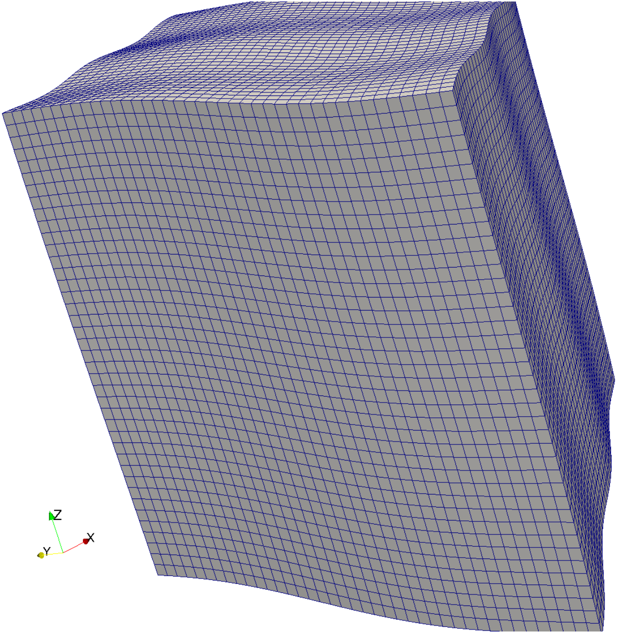
Verification Process

1. Identify (at least) formal order-of-accuracy of CAA code's numerical schemes,
2. Choose test case parameters,
3. Run EVA,
4. Run CAA code with range of "discretization measures" (grid spacings or time step sizes),
5. Calculate error and some error norm,
6. Calculate error norm convergence rate, compare to CAA code's formal order-of-accuracy.

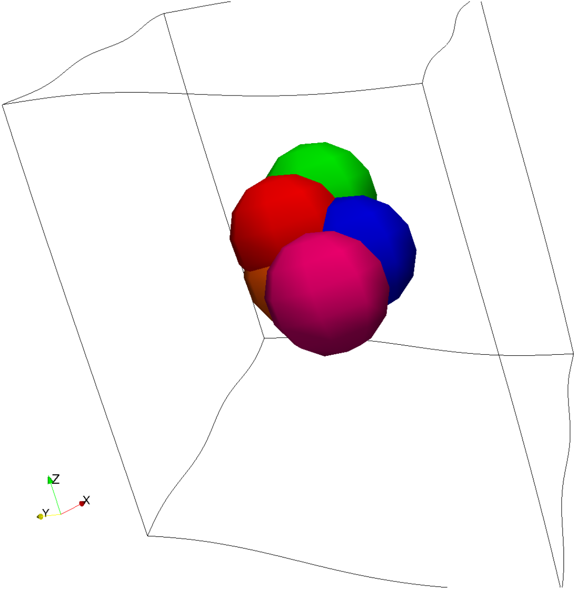
Spatial Test Case Parameters

- Skewed, rotated, and “perturbed” 3D grid series.
- RK7S time marching scheme with a single very small time step $\Delta t = 0.0005$ (far-field acoustic velocity is 1).
- Initial condition has an approximate wavenumber $\widetilde{k} = \frac{2\pi}{0.25} = 8\pi$, which, when combined with the grid spacings chosen, give a range of points-per-wavelength from 2 to 50.

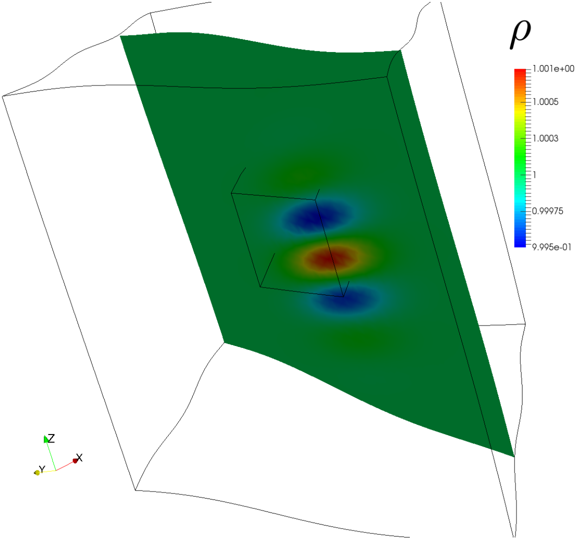
Test Case Parameters: Grid



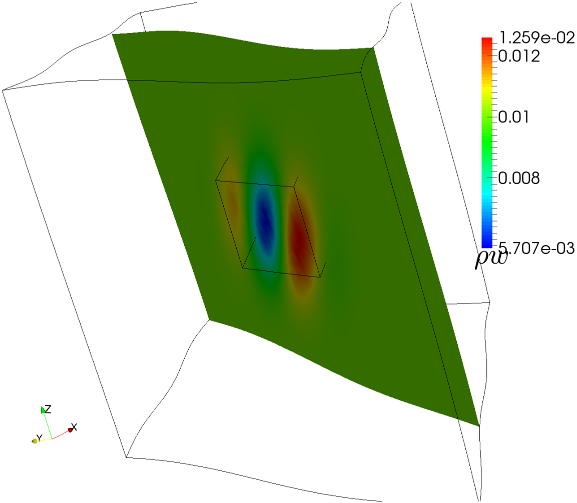
Test Case Parameters: Gaussian Centers



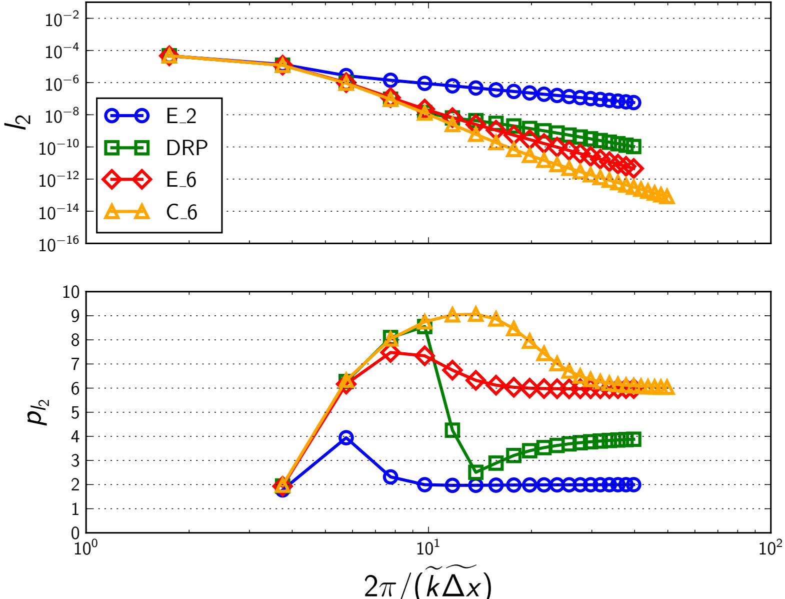
Test Case Parameters: ρ Initial Condition



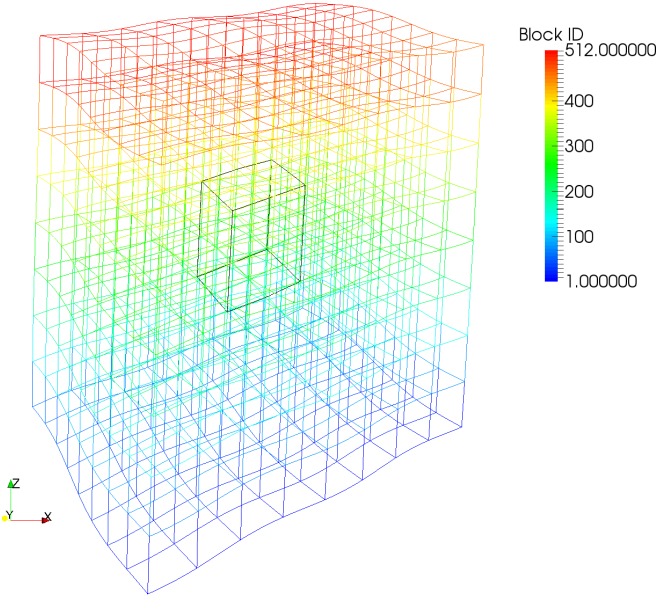
Test Case Parameters: ρw Initial Condition



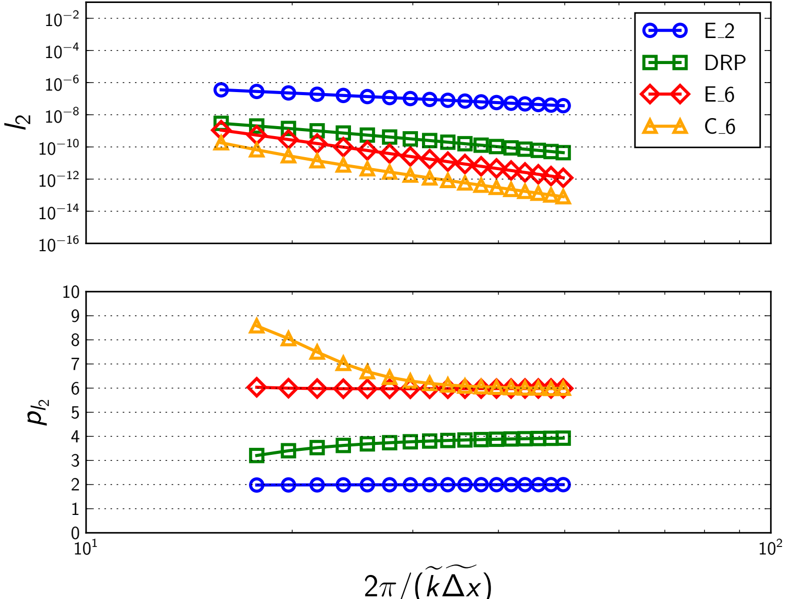
BASS Spatial Verif.: ρ, l_2, $A\Delta x^p$ Assumption



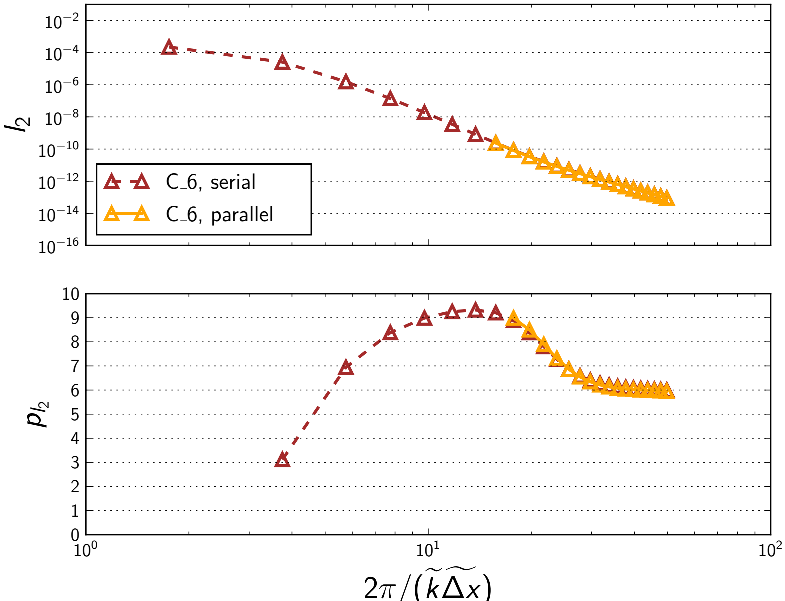
BASS Spatial Verif.: Multiblock Grid for Parallel Runs



BASS Spatial Verif.: ρ, l_2, $A\Delta x^p$ Assumption, Parallel



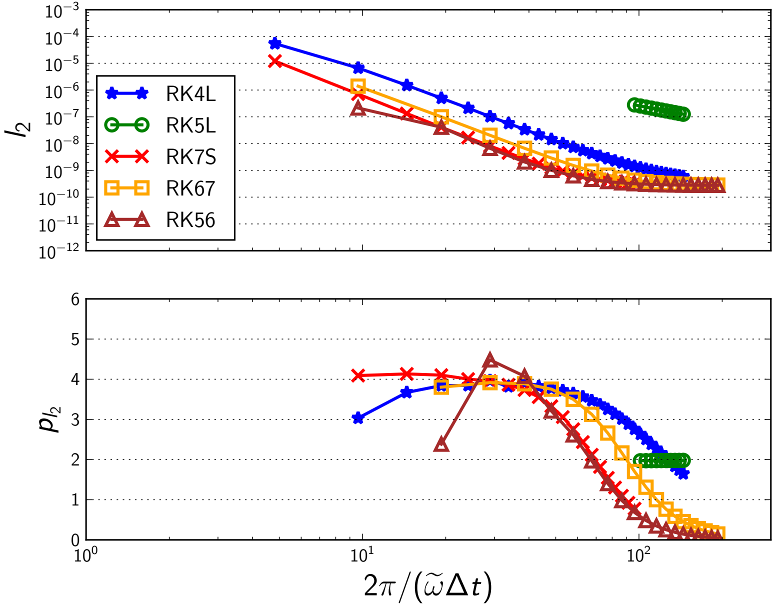
BASS Spatial Verif.: ρv, C_6 Serial vs. Parallel



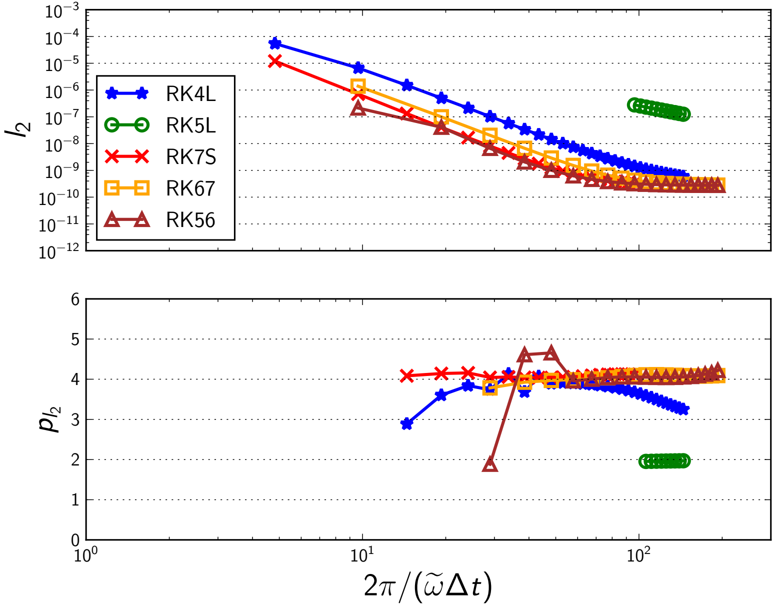
Temporal Test Case Parameters

- Grid and initial condition identical to the spatial test cases.
- Used E_6 scheme for spatial differencing.
- Marched to a final time level of $t = 0.05$ with an EVA target truncation error of 10^{-12}, which required a 26th-order Taylor series.
- Combination of time step size and initial condition gives a steps-per-period range of 5 to 200.

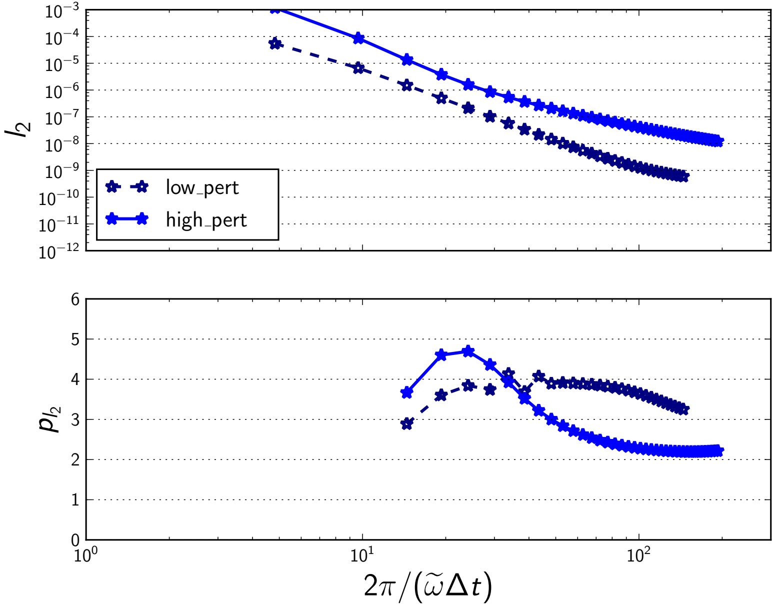
BASS Temporal Verif.: ρ, l_2, $A\Delta t^p$ Assumption



BASS Temporal Verif.: ρ, l_2, $A\Delta t^p + B$ Assumption



BASS Temporal Verif.: RK4L Low vs. High Amp.



Conclusions

- EVA provides a flexible method for obtaining a reference solution suitable for code verification
    - Extended to support 3D nonlinear Navier-Stokes.
    - No modification to CAA code required.
    - EVA doesn't care about the quality of the grid (just like an exact or manufactured solution).
- EVA was used to verify the spatial and temporal schemes in BASS, a high-order CAA solver.
    - Each scheme eventually converged at the expected rate, strongly verifying the code.
    - Overall, error and convergence rates corresponded well to linear analysis of the schemes (see paper).
    - Needed more general $A\Delta t^p + B$ assumption when calculating convergence rate for the high-order time-marching schemes.
    - Saw linear/nonlinear behavior for RK4L scheme (4th-order linear, 2nd-order nonlinear).

Future Work

- Current limitations of the EVA approach to code verification:
  - Boundary conditions,
  - Non-constant properties (i.e., viscosity and thermal conductivity),
  - Finding the asymptotic range.
- More codes(?)

The End

Thanks!

This work was supported by the NASA Advanced Air Transport Technology (AATT) Project. The authors would like to thank Dr. Edmane Envia of the NASA Glenn Research Center, who was the technical monitor for this work.

References I

[1] Les Hatton. "The T experiments: errors in scientific software". In: *Computing in Science and Engineering* 4.2 (1997), pp. 27–38. URL: http://www.computer.org/csdl/mags/cs/1997/02/c2027.pdf (visited on 08/13/2014).

[2] Stanly Steinberg and Patrick J Roache. "Symbolic manipulation and computational fluid dynamics". In: *Journal of Computational Physics* 57.2 (Jan. 1985). 00079, pp. 251–284. ISSN: 0021-9991. DOI: 10.1016/0021-9991(85)90045-2. URL: http://www.sciencedirect.com/science/article/pii/0021999185900452 (visited on 04/11/2014).

References II

[3] Patrick J. Roache. “Code Verification by the Method of Manufactured Solutions”. In: *Journal of Fluids Engineering* 124.1 (Nov. 2001), pp. 4–10. ISSN: 0098-2202. DOI: 10.1115/1.1436090. URL: http://dx.doi.org/10.1115/1.1436090 (visited on 06/10/2015).

[4] Rodger W. Dyson. *Technique for Very High Order Nonlinear Simulation and Validation*. NASA Technical Memorandum NASA/TM 2001-210985. Cleveland, OH, United States: NASA Glenn Research Center, July 2001. URL: http://ntrs.nasa.gov/search.jsp?R=20010090341 (visited on 05/29/2015).

References III

[5] Christopher K. W. Tam and Jay C. Webb. "Dispersion-Relation-Preserving Finite Difference Schemes for Computational Acoustics". In: *Journal of Computational Physics* 107.2 (Aug. 1993), pp. 262–281. ISSN: 0021-9991. DOI: 10.1006/jcph.1993.1142. URL: http://www.sciencedirect.com/science/article/pii/S0021999183711423 (visited on 11/11/2014).

[6] R. Hixon. "Prefactored Small-Stencil Compact Schemes". In: *Journal of Computational Physics* 165.2 (Dec. 2000). 00079, pp. 522–541. ISSN: 0021-9991. DOI: 10.1006/jcph.2000.6631. URL: http://www.sciencedirect.com/science/article/pii/S0021999100966312 (visited on 05/29/2014).

References IV

[7] Sanjiva K. Lele. "Compact finite difference schemes with spectral-like resolution". In: *Journal of Computational Physics* 103.1 (Nov. 1992), pp. 16–42. ISSN: 0021-9991. DOI: 10.1016/0021-9991(92)90324-R. URL: http://www.sciencedirect.com/science/article/pii/002199919290324R (visited on 08/22/2014).

[8] A. Jameson. "Time Dependent Calculations Using Multigrid, with Applications to Unsteady Flows Past Airfoils and Wings". In: *AIAA 10th Computational Fluid Dynamics Conference*. AIAA 91-1596. Honolulu, HI, June 1991.

[9] D. Stanescu and W. G. Habashi. "2N-Storage Low Dissipation and Dispersion Runge-Kutta Schemes for Computational Acoustics". In: *Journal of Computational Physics* 143.2 (1998), pp. 674–681.

References V

[10] Vasanth Allampalli, Ray Hixon, M. Nallasamy, and Scott D. Sawyer. "High-accuracy large-step explicit Runge–Kutta (HALE-RK) schemes for computational aeroacoustics". In: *Journal of Computational Physics* 228.10 (June 2009), pp. 3837–3850. ISSN: 0021-9991. DOI: 10.1016/j.jcp.2009.02.015. URL: http://www.sciencedirect.com/science/article/pii/S0021999109000849 (visited on 11/11/2014).

[11] Patrick Knupp and Kambiz Salari. *Verification of Computer Codes in Computational Science and Engineering*. en. CRC Press, Oct. 2002. ISBN: 9781584882640.

[12] *Journal of Computational Physics* 274 (2014). ISSN: 0021-9991.

[13] *Computers & Fluids* 103 (2014). ISSN: 0045-7930.

References VI

[14] Rodger W. Dyson and John W. Goodrich. “An Automated Approach to Very High Order Aeroacoustic Computations in Complex Geometries”. In: *6th AIAA/CEAS Aeroacoustics Conference*. Lahaina, HI: AIAA, June 2000, pp. 1–41. URL: http://ntrs.nasa.gov/search.jsp?R=20000120419 (visited on 05/29/2015).

[15] Rodger W. Dyson and John W. Goodrich. “Automated Approach to Very High-Order Aeroacoustic Computations. Revision”. In: *AIAA Journal* 39.3 (Mar. 2001). URL: http://ntrs.nasa.gov/search.jsp?R=20010061355 (visited on 05/29/2015).

[16] Cristobal E. Castro. “High Order ADER FV/DG Numerical Methods for Hyperbolic Equations”. Doctoral Thesis. Trento, Italy: University of Trento, May 2007.

References VII

[17] C.E. Castro and E.F. Toro. “Solvers for the high-order Riemann problem for hyperbolic balance laws”. In: *Journal of Computational Physics* 227.4 (2008), pp. 2481–2513. ISSN: 0021-9991. DOI: 10.1016/j.jcp.2007.11.013. URL: http://journals.ohiolink.edu/ejc/article.cgi?issn=00219991&issue=v227i0004&article=2481_sfthrpfhbl (visited on 06/01/2015).

MES Example

From Knupp & Salari[11], the exact solution to unsteady homogeneous heat conduction in a solid sphere is

$$T(r,\mu,\phi,t) = \frac{1}{\pi}\sum_{n=0}^{\infty}\sum_{p=1}^{\infty}\sum_{m=0}^{n}\frac{e^{-\alpha\lambda_{np}^2 t}}{N(m,n)M(\lambda_{np})}J_{n+1/2}(\lambda_{np}r)P_n^m(\mu)$$

$$\int_0^b\int_{\mu_0}^1\int_0^{2\pi} r'^{3/2}J_{n+1/2}(\lambda_{np}r')P_n^{-m}(\mu')\cos\left(\phi-\phi'\right)F(r',\mu',\phi')\,d\phi'\,d\mu'\,dr'$$

where

$$N(m,n) = \left(\frac{2}{2n+1}\right)\frac{(n+m)!}{(n-m)!}$$

$$M(\lambda_{np}) = \frac{b^2}{2}\left[J_{n+1/2}\left(\lambda_{np}r\right)\right]^2.$$

Takeaway

Exact solutions are simultaneously too complex, too simple.

What is the Method of Manufactured Solutions[2, 3]? I

- The "backwards approach:" start with the solution you want, then change the PDE to make it work.
- Simple example with the linear advection equation:
  1. The manufactured solution:
  $$\hat{u}(x,t) = \sin(x - bt)$$
  2. The PDE:
  $$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0,$$
  3. Get the MMS source term by putting the manufactured solution in the PDE:
  $$\begin{aligned} S(x,t) &= \frac{\partial \hat{u}}{\partial t} + a\frac{\partial \hat{u}}{\partial x} \\ &= \left[\sin(x-bt)\right]_t + a\left[\sin(x-bt)\right]_x \\ &= (a-b)\cos(x-bt). \end{aligned}$$

What is the Method of Manufactured Solutions[2, 3]? II

4. Add $S(x, t)$ to the PDE to make $\hat{u}$ an exact solution:

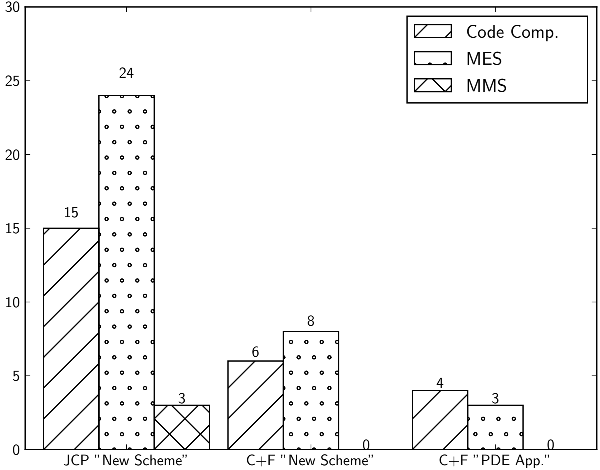
$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = S(x, t) = (a - b)\cos(x - bt).$$

5. Use $\hat{u}$ as the reference solution.

- Advantages
  - General: applicable to any PDE.
  - Flexible: user chooses the solution, so all aspects of PDE code can be tested
- Disadvantages
  - MMS source term can be complicated (but computer algebra system software [CAS] helps)
  - PDE code must allow for user-specified distributed sources, or be made to.

MMS in the “Real World”

- Surveyed a recent volume of the Journal of Computational Physics[12] and Computers & Fluids[13] (total of 69 articles)
  - Was code verification performed? If so, how?



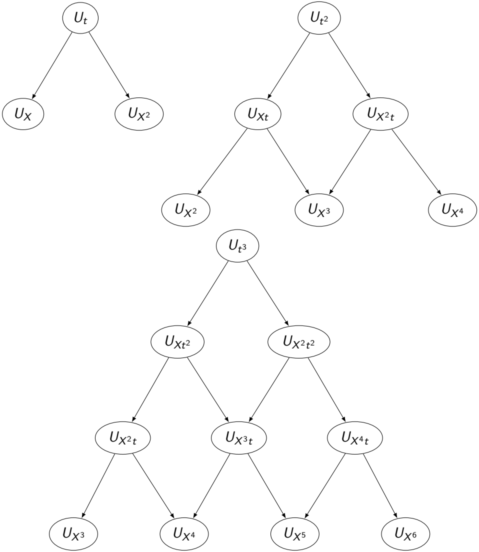
Cauchy-Kowalewski Recursion: What is it?

- Cauchy-Kowalewski (CK) recursion: fancy math technique for getting the temporal derivatives needed for the Taylor series coefficients by repeatedly differentiating the original PDE.
  - Used in the proof of the Cauchy-Kowalewski theorem, a local existence and uniqueness theorem for PDEs.
  - Occasionally used to integrate PDEs numerically:
    - Dyson & Goodrich's MESA [14, 4, 15]
    - Castro & Toro's ADER [16, 17]

CK Recursion: How it works

- Goal: to express $\frac{\partial^m u}{\partial t^m}$ in terms of $\frac{\partial^n u}{\partial x^n}$.
- Two "phases" CK recursion:
  1. Find $\frac{\partial^n u}{\partial t^n}$ by taking t-derivative $\frac{\partial^{n-1} u}{\partial t^{n-1}}$,
  2. Look at $\frac{\partial^n u}{\partial t^n}$, finding all unknown mixed-x-t derivatives by taking x-derivatives of known expressions, repeating until only pure x-derivatives remain.
- See paper for all the (very exciting!) details.

CK Recursion: Viscous Burgers Dependency Graphs



Differentiating Governing Equations with the Leibniz Rule

- Problem: CK recursion requires repeated differentation of the governing equation (here, 3D nonlinear Navier-Stokes).
- The terms in the flow equations that give us trouble are products ($u\frac{\partial v}{\partial x}$, etc.).
- Is there a pattern?
  - $u\frac{\partial v}{\partial x}$
  - $\frac{d}{dx}\left(u\frac{\partial v}{\partial x}\right) = \frac{\partial u}{\partial x}\frac{\partial v}{\partial x} + u\frac{\partial^2 v}{\partial x^2}$
  - $\frac{d^2}{dx^2}\left(u\frac{\partial v}{\partial x}\right) = \frac{\partial^2 u}{\partial x^2}\frac{\partial v}{\partial x} + 2\frac{\partial u}{\partial x}\frac{\partial^2 v}{\partial x^2} + u\frac{\partial^3 v}{\partial x^3}$
  - $\frac{d^3}{dx^3}\left(u\frac{\partial v}{\partial x}\right) = \frac{\partial^3 u}{\partial x^3}\frac{\partial v}{\partial x} + 3\frac{\partial^2 u}{\partial x^2}\frac{\partial^2 v}{\partial x^2} + 3\frac{\partial u}{\partial x}\frac{\partial^3 v}{\partial x^3} + u\frac{\partial^4 v}{\partial x^4}$

Differentiating Governing Equations with the Leibniz Rule

- The Leibniz rule: generalization of the product rule from calculus:

$$(f \cdot g)^{(n)} = \sum_{k=0}^{n} \binom{n}{k} f^{(k)} g^{(n-k)}$$

  - $\binom{n}{k}$: (n, k) binomial coefficient (Pascal's triangle).
- So,
  - $(f \cdot g)^{(0)} = fg$
  - $(f \cdot g)^{(1)} = f^{(1)}g + fg^{(1)}$
  - $(f \cdot g)^{(2)} = f^{(2)}g + 2f^{(1)}g^{(1)} + fg^{(2)}$
  - $(f \cdot g)^{(3)} = f^{(3)}g + 3f^{(2)}g^{(1)} + 3f^{(1)}g^{(2)} + fg^{(3)}$
- Approach: As in Dyson[4], repeatedly use Leibniz rule to differentiate with respect to each variable (x, y, z, t), giving recurrence relation for all the derivatives needed for CK recursion.

Recurrence Relation for the Navier-Stokes

- The x-momentum equation:

$$
\begin{aligned}
\frac{\partial^{b+d+n+1} u}{\partial x^{b}\,\partial y^{d}\,\partial t^{n+1}} = -\sum_{a=0}^{b}\binom{b}{a}\sum_{c=0}^{d}\binom{d}{c}\sum_{m=0}^{n}\binom{n}{m}\Bigg[& \\
& \frac{\partial^{b-a+d-c+n-m} u}{\partial x^{b-a}\,\partial y^{d-c}\,\partial t^{n-m}}\,\frac{\partial^{a+1+c+m} u}{\partial x^{a+1}\,\partial y^{c}\,\partial t^{m}} \\
& + \frac{\partial^{b-a+d-c+n-m} v}{\partial x^{b-a}\,\partial y^{d-c}\,\partial t^{n-m}}\,\frac{\partial^{a+c+1+m} u}{\partial x^{a}\,\partial y^{c+1}\,\partial t^{m}} \\
& + \frac{\partial^{b-a+d-c+n-m} \sigma}{\partial x^{b-a}\,\partial y^{d-c}\,\partial t^{n-m}}\,\frac{\partial^{a+1+c+m} p}{\partial x^{a+1}\,\partial y^{c}\,\partial t^{m}} \\
& - \mu\frac{\partial^{b-a+d-c+n-m} \sigma}{\partial x^{b-a}\,\partial y^{d-c}\,\partial t^{n-m}}\Bigg(\frac{4}{3}\,\frac{\partial^{a+2+c+m} u}{\partial x^{a+2}\,\partial y^{c}\,\partial t^{m}} \\
& + \frac{1}{3}\,\frac{\partial^{a+1+c+1+m} v}{\partial x^{a+1}\,\partial y^{c+1}\,\partial t^{m}} \\
& + \frac{\partial^{a+c+2+m} u}{\partial x^{a}\,\partial y^{c+2}\,\partial t^{m}}\Bigg)\Bigg]
\end{aligned}
$$

Order-of-Accuracy Calculation

Two different approaches used here:

- Simplest approach: assume

  $$\epsilon_i = \phi_i - \Phi \approx Ah_i^p$$

  Two unknowns (A, p), requires error from two CAA runs.

- Slightly more complicated: assume

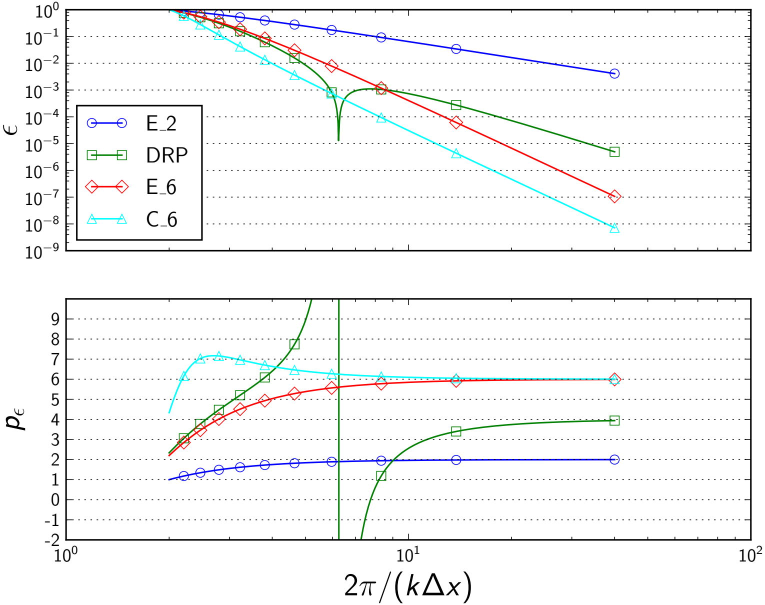
  $$\epsilon_i \approx Ah_i^p + B$$

  which has three unknowns (A, B, p) — needs three CAA runs.

Spatial Verification: Fourier Analysis

- At a minimum, need to know scheme's order-of-accuracy to verify with EVA (or anything else) and the order-of-accuracy test
  - But a more detailed understanding of a scheme's performance helps with setting up test cases and interpreting results
- Fourier analysis for finite differencing schemes: use scheme to differentiate a single Fourier component $f(x) = e^{ikx}$, then compare to the exact value.

FD Schemes: Fourier Analysis Error and Conv. Rate



Temporal Verification: Fourier Analysis

- Fourier analysis for time-marching schemes: use the scheme to integrate the ODE

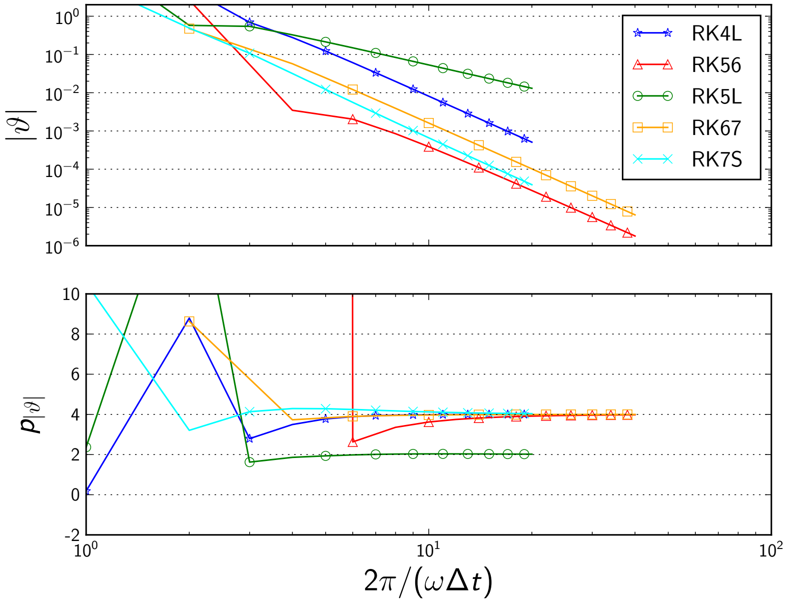
$$\frac{du}{dt} = -i\omega u$$

which has the exact solution

$$u(t) = u_0 e^{-i\omega t}$$

where $u_0 = u(0)$.

Temporal Schemes: Global Error and Convergence Rate



10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
$|\vartheta|$
RK4L
RK56
RK5L
RK67
RK7S
10
8
6
4
2
0
-2
$p_{|\vartheta|}$
10^{0}
10^{1}
10^{2}
$2\pi/(\omega\Delta t)$