

# Efficient Code Verification Using the Residual Formulation of the Method of Manufactured Solutions

Clarence O. E. Burg\*

*Mississippi State University, Mississippi State, Mississippi, 39762, U.S.A.*

Vasanth K. Murali†

*Washington University at St. Louis, St. Louis, Missouri, U.S.A.*

The Method of Manufactured Solutions is a code verification method that modifies the governing equations solved within a code by adding a source term to drive the solution towards a predetermined analytic function. By solving the modified equations on a sequence of grids and comparing the differences between the converged solution and manufactured solution, the order of accuracy of the implementation can be determined. The method of manufactured solutions combines the benefits of comparing with an exact solution without the need to derive an exact solution to the governing equations. However, in its current form, highly converged solutions on a sequence of grids are required which can be quite costly and difficult to obtain. In this paper, the method of manufactured solutions is used in a different fashion that removes the need for converged solutions by considering only the residual of the discretized governing equations rather than the solution, thus avoiding the computational cost and difficulties inherent in obtaining highly converged solutions. Furthermore, this new approach is quite similar to the method for analyzing a discretization method to determine the order of accuracy of that method via Taylor's series expansions. This new approach is demonstrated to yield the same order of accuracy as the original method of manufactured solutions using three different cases - one-dimensional porous media equation, one-dimensional St. Venant equations and two-dimensional unstructured Euler simulations.

## Nomenclature

$p$	Order of Accuracy
$h$	Typical Step Size
$I(h)$	Error Measure
$Q$	Converged Solution
$Q^e$	Exact Solution
$Q^i$	Computed Solution on Grid $i$
$\chi$	Computational Domain
$t$	Time
$V$	Volume
$R$	Residual

## I. Introduction

When a computational simulation is used to approximate the physics of a particular problem, the computer yields a set of numbers that represent approximations to physical quantities. The computer, however, does not tell the user whether the numbers are correct or realistic. How does one determine whether those numbers are consistent with the governing equations, and how does one determine how accurate those numbers are? These questions are the principle concerns of verification and validation.

---

\* Assistant Research Professor, Computational Simulation and Design Center, 2 Research Blvd., Starkville, MS, 39759, Member AIAA.

† Graduate Research Assistant, Department of Chemical Engineering, Washington University, One Brookings Drive, St. Louis, MO, 63130-4899, Student Member AIAA.

Two different classes of error exist in a computational simulation - errors associated with modeling the physical phenomenon with a mathematical expression and errors associated with solving a continuous mathematical expression on a discrete set of points using a finite precision machine. Code validation addresses the first class of error and involves comparing computational results with experimental results to determine the usefulness of the computational tool in predicting the physical phenomenon. Code verification is the process of quantifying the error introduced by discretizing the continuous governing equations and solving them on finite precision machines.

Code verification is the process of identifying sources of error within a numerical simulation. Typical sources of error include round-off error resulting from the use of finite precision machines, oscillations in the solution due to the lack of a strongly converged solution and discretization error associated with the choice of discretization method. Round-off error for steady-state simulations is probably of little concern due to the self-correcting nature of steady-state simulations. Oscillations in the solution near convergence are often noticed for complicated geometries with complicated physics, especially in three-dimensional flow and for turbulent flow, and the error associated with these oscillations can be estimated by banding the oscillations from above and below. Discretization error is the error that results from the differences between the discretized equation and the continuous partial differential equation, and for most simulations, this error is the critical one to analyze.

Code verification is typically accomplished via a grid refinement study where the solution on a sequence of grids is compared either with a known exact solution or with the solution on a highly resolved grid. As the grid is resolved, the error between the solution on the current grid and the exact solution should reduce at a rate consistent with the order of accuracy of the method. By demonstrating that this rate is achieved asymptotically, the code can be deemed "verified". However, there are many difficulties with grid refinement studies. First, exact solutions may be difficult to obtain and may result in closed-form solutions involving infinite series, which are problematic to compute numerically. But without an exact solution, there is no guarantee that the solution to which the code converges is the solution to the governing equations, even if the order of accuracy is demonstrated. For instance, if a multiplicative factor for one of the terms in the governing equation is omitted in the code, then the solutions may converge well, but they will clearly converge to the wrong answer.

The Method of Manufactured Solutions (MMS), developed by Steinberg and Roache<sup>1,2</sup>, is a methodical approach to determine the order of accuracy of the implementation of a method to discretize a partial differential equation (PDE). This approach combines the best features of grid refinement studies and comparisons with exact solutions, by introducing a source term that drives the solution towards a given analytic function. By comparing the converged solution on a sequence of successively refined grids with the manufactured exact solution, the order of accuracy of the solution and hence of the implementation can be determined. Unlike a typical grid refinement study, the solution on these grids should converge to a known, analytic solution, and unlike comparison with exact solutions, the analytic solution need not be a solution to the governing equations, so the method of manufactured solutions can be applied to complicated sets of equations on complicated geometries.

This method was originally proposed for code verification of structured flow solvers on structured grids, typically grids with little geometry complexity, such as squares or cubes. As such, the sequence of grids generated for the refinement study did not need to be produced by grid doubling. However, for more realistic geometries, Eca<sup>3</sup> has introduced the concept of geometrically similar grids, where the elements in the refined grid union to form each element in the coarse, so that grid doubling is required. This method of manufactured solutions has recently been extended to unstructured solvers on structured grids<sup>4</sup> and to unstructured solvers on triangular unstructured grids<sup>5,6</sup>.

In order to avoid the computational cost of a grid refinement study and the need to obtain highly converged solutions, a new version of the method of manufactured solutions has been derived that directly compares the error in the implementation of the discretization method. When the order of accuracy of a discretization method is analyzed algebraically, the error in a discretization method is defined as the difference between the discretization method and the partial differential equation when acting upon a smooth analytic function. Typically, the order of accuracy is found via a Taylor's Series expansion of the discretized equations about a common point, and the result is compared with the partial differential equation. However, the order of accuracy derived from a grid refinement study compares the differences in the converged solutions, which represents the effects of the discretization method but does not directly interrogate the discretization method. This new version of the method of manufactured solutions isolates the discretization error inherent in the discretization method, and hence can calculate the order of accuracy of an implementation without the need for a converged solution or for any iterations of the algorithm towards convergence.

This paper first presents the important concepts for grid refinement methods to obtain an estimate of the order of accuracy, then the original formulation of the method of manufactured solutions is presented, followed by the residual formulation of the method of manufactured solutions. After the methods are presented, these methods are tested on three different sets of equations - the 1D porous media equation, the 1D St. Venant equations and the 2D Euler

equations.

## II. Code Verification Methods

To determine the accuracy of a numerical implementation, two methods are predominate, 1) comparison with an exact solution and 2) use of grid refinement studies. In the first method, a simple geometry with an known, relatively simple analytic solution is used for comparison with the numerical code. The assumption is that as the typical grid spacing tends to zero, the results from the numerical code will converge to the exact solution. By comparing the difference between the exact solution and the numerical solution on two different grids, the order of accuracy can be obtained. Some quantity of interest  $I$  is measured, such as the drag for the geometry. So for the exact solution, the exact value of the quantity  $I_{exact}$  is known and the assumption that for a grid with typical grid spacing  $h$ , this quantity can be expanded via Taylor's series expansion as

$$I(h) = I_{exact} + A_1 h^1 + A_2 h^2 + A_3 h^3 + \dots \quad (1)$$

If the code implementation is second-order accurate,  $A_1$  will be zero and  $A_2$  will be non-zero, as may be the other coefficients. However, since only a finite number of simulations will be performed, the Taylor's series is truncated, which may result in non-zero entries for each term in the series. As a result, another form of the expansion is often used, or

$$I(h) = I_{exact} + Ah^p \quad (2)$$

where  $A$  and  $p$  are unknowns. If the code implementation is second-order accurate, the exponent  $p$  will converge to 2.0 as the grid is refined, because the influence of the higher order terms in the expansion will become increasingly less noticeable.

If an exact solution is used, then  $I_{exact}$  is assumed to be known, then two unknowns exist, which are the coefficient  $A$  and the exponent  $p$ , so the solution on two grids is necessary. For instance, if the exact solution is known and two grids with typical grid spacings  $h_1$  and  $h_2$  are used to generate a numerical value for  $I(h_1)$  and  $I(h_2)$ , then solving to remove  $A$  from the equation yields

$$\left(\frac{h_2}{h_1}\right)^p = \frac{I(h_2) - I_{exact}}{I(h_1) - I_{exact}} \quad (3)$$

and the order of accuracy is approximately

$$p = \frac{\log\left(\frac{I(h_2) - I_{exact}}{I(h_1) - I_{exact}}\right)}{\log\left(\frac{h_2}{h_1}\right)} \quad (4)$$

If the exact solution is not known, then the solution on three different grids with spacing  $h_1$ ,  $h_2$  and  $h_3$  are required, or

$$\begin{aligned} I(h_1) &= I_{exact} + Ah_1^p \\ I(h_2) &= I_{exact} + Ah_2^p \\ I(h_3) &= I_{exact} + Ah_3^p \end{aligned} \quad (5)$$

Typically, the spacings are related, in that  $h_2 = rh_1$  and  $h_3 = rh_2$  where  $r$  is some refinement ratio. Using a constant grid refinement ratio of  $r$ , then the order of accuracy is

$$p = \frac{\log\left(\frac{I(h_3) - I(h_2)}{I(h_2) - I(h_1)}\right)}{\log(r)} \quad (6)$$

When using a grid refinement study, the assumption is that the grids lie within the "asymptotic range", which means that the primary source of error in the simulation arises from the leading order error term and that the other sources of error are insignificant. In reality for complicated two-dimensional and for most three-dimensional simulations, the size of grid required to be in the "asymptotic region" may be excessively large, so the assumption that the leading order error term is the dominant error term may not be valid. Thus, more than the minimum number of grids is often used to determine the order of accuracy. When using 4 or more grids, several different combinations of grids can be used to determine the order of accuracy and more complicated formulae, such as those reported by Eca<sup>3</sup>, can be used,

which will probably lead to significant differences in the reported order of accuracy. The warning, then, is to be aware of these limitations and assumptions and to take care when using these methodologies.

Furthermore, the means by which a grid is refined can affect the observed order of convergence. Eca<sup>3</sup> used the term “geometric similarity” to describe the types of grids that he used in his grid refinement study. One of the main characteristics of geometrically similar grids is that each element in the coarse grid is subdivided into smaller elements that union to form the coarse element. Hence, the volume of influence is maintained in a consistent manner from one grid to the next grid, and the faces of the control volume in the coarse grid are exactly replicated in the fine grid. Geometric similarity is achieved for one-dimensional grids by subdividing each interval into an integer number of sub-intervals, and it is achieved for two-dimensional triangulated unstructured grids by h-refinement without any quality improvements.

### III. Method of Manufactured Solutions

The method of manufactured solutions modifies the governing equations so as to drive the solution towards a predetermined “manufactured” solution. The continuous partial differential equation has the form and dependencies of

$$\mathcal{L}(Q, \chi, t) = f(\chi, t) \quad (7)$$

with boundary conditions

$$\mathcal{B}(Q, \chi, t) = g(\chi, t) \quad (8)$$

where  $Q$  is converged solution,  $\chi$  represents the computational domain,  $f(\chi, t)$  and  $g(\chi, t)$  are source terms that only depends on the computational domain and on time, but not on the solution  $Q$ . The implementation of the source terms can be verified easily by a grid refinement and comparison with the exact source terms, without solving for the flow variables  $Q$ .

Each flow variable represented in  $Q$  is set to some continuous function such as the exponential or sine or cosine functions, which will be termed  $Q^e(x, y, z)$ . These “manufactured” solutions are plugged into the governing partial differential equation to generate a new set of source terms, which modify the governing equations to

$$\mathcal{L}(Q, \chi, t) = \mathcal{L}(Q^e, \chi, t) \quad (9)$$

and

$$\mathcal{B}(Q, \chi, t) = \mathcal{B}(Q^e, \chi, t) \quad (10)$$

The geometric source terms  $f(\chi, t)$  and  $g(\chi, t)$  cancel out in the process, but as their implementation is easy to verify, this cancelation is acceptable.

Upon discretization, the right-hand-side can be evaluated exactly when the methodology and assumptions behind the discretization method are evaluated. For instance, when the finite volume method is used, the partial differential equations are integrated over the control volume, and when the finite element method is used, the partial differential equations multiplied by a weight function are integrated over each element. These integrals can be evaluated exactly because of the nature of the manufactured solutions, especially if the manufactured solutions are exponentials or trigonometric functions. Thus, using a symbolic manipulation software package such as Mathematica, these integrals can be evaluated exactly and implemented without modification into a FORTRAN or C code.

Because of the source terms, the discrete solution vector  $Q^c$  should converge to the exact solution, assuming a proper implementation of the discretized governing equations. Thus, the order of accuracy of the implementation can be verified by comparing the difference between the exact solution  $Q^e$  and the converged solution  $Q^c$  via some norm, such as the  $L_1$ -norm, or

$$Error_1(Q^e, Q^c) = \sum_{j=1}^N |Q_j^e - Q_j^c| V_j \quad (11)$$

where  $N$  is the number of nodes in the grid,  $Q_j^e$  represents the exact solution evaluated at the location of node  $j$  and  $V_j$  is the volume associated with node  $j$ . If the order of accuracy matches the theoretical order of accuracy, then the errors in the implementation are no worse than the errors in the discretization method, and the code can be said to be “verified”.

The manufactured solution can be any convenient function with the properties that the function is differentiable enough so that the leading order error terms in the Taylor’s series expansion are non-zero. For instance, for a second-order partial differential equation, such as the heat equation, and a third-order accurate discretization, then the manufactured solution must have at least five continuous derivatives to account for the two spatial derivatives in the PDE and

the non-zero third-order error term which can be found via three additional derivatives. Typically, exponentials, cosines and sines are convenient since they are infinitely differentiable, but under some circumstances, the anti-derivatives of these functions when the PDE is applied to them may not exist. In those cases, high-degree polynomials will suffice. The choice of manufactured solution does require some effort to avoid singularities such as division by zero or taking the square root of a negative number in some regions of the domain.

#### IV. Residual Formulation of the Method of Manufactured Solutions

The original formulation of the method of manufactured solutions requires the calculation of the solution  $Q^i$  on a sequence of successively resolved grids, which can be computationally expensive. Fortunately, the required information is already available, without the need for iterating towards the solution. In particular, by plugging the discrete values of the exact solution for grid  $\chi_i$  into the discretized governing equations,

$$\begin{aligned} R_i(Q^e) &= R_i(Q^i + Q^e - Q^i) = R_i(Q^i + \Delta Q^i) \\ &= R_i(Q^i) + \frac{\partial R_i}{\partial Q}(Q^i) (\Delta Q^i) + O(\Delta Q^i)^2 \\ &= \frac{\partial R_i}{\partial Q}(Q^i) (\Delta Q^i) + O(\Delta Q^i)^2 \end{aligned} \quad (12)$$

where  $\Delta Q^i = Q^e - Q^i$  and  $R_i(Q^i) = 0$ , since  $Q^i$  is the converged solution for the equations on grid  $\chi_i$ . Thus, the error represented by  $R_i(Q^e)$  contains the same error term as the difference between the exact and computed solutions. If the residual linearly depends on the unknown  $Q$ , then Eqn (13) reduces to

$$R_i(Q^e) = \frac{\partial R_i}{\partial Q}(Q^i) \Delta Q^i \quad (13)$$

where  $\frac{\partial R_i}{\partial Q}(Q^i)$  is a constant matrix dependent only on metric terms, and the order of accuracy of the residual is clearly the same as the order based on the converged solution. Using this reasoning, if the residual is a polynomial in  $Q$ , as is the case with many sets of governing equations including the Navier-Stokes equations, then the additional terms represent higher order terms in the expression, and the leading order error term is the same. If the residual includes ratios of these flow variables or other combinations such as square roots that can generate singularities, then the manufactured solution must avoid these singularities, in which case the residual can be written as a convergent Taylor's series expansion, and the order of accuracy is again determined from the leading term.

Because of the equivalence of the residual form and the original formulation of the method of manufactured solutions, only the residual equations resulting from the discretized governing equations and the source term from the manufactured solution needs to be evaluated, and it only needs to be evaluated once, which is quite inexpensive in comparison to the cost of obtaining highly converged solutions on a sequence of successively refined grids. In fact, determining the order of accuracy directly from the residual rather than from the converged solutions mimics the method for determining the order of accuracy of the stencil via a Taylor's series expansion. When one determines the order of accuracy of a stencil, the terms are expanded about a common point in time and space assuming a sufficiently smooth function. After cancelations, the continuous governing equation is recovered, assuming a consistent discretization, and the extra terms are the leading order error terms and higher order error terms. This methodology presented herein works in a similar fashion.

#### V. 1D Porous Media Equation

The porous media equation governs the flow of water through porous media such as groundwater through an aquifer. It is governed by Darcy's law that states that the flow is proportional to the change in the pressure. Perhaps the simplest example is the flow of water through an aquifer due to the pumping action at a well. For this case, the flow is assumed to be radially symmetric about the well of radius  $R_w$  and flows through a confined aquifer with the radius of influence  $R_o$  (i.e., the outer boundary). Under these assumptions, the pressure within the aquifer is the piezometric head  $h(r, t)$  and the continuity of mass equation can be reduced to

$$\frac{S}{T} \frac{\partial h}{\partial t} = \frac{\partial^2 h}{\partial r^2} + \frac{1}{r} \frac{\partial h}{\partial r} \quad (14)$$

where  $S$  is the storage coefficient and  $T$  is the transmissivity. For steady-state flow, the temporal derivative is zero. Assuming a drawdown at the well of  $H_o - H_w$ , where  $H_o$  is the head at the outer boundary and  $H_w$  is the head at the

well, then the solution to the governing equations is

$$h(r) = H_o + \frac{(H_w - H_o)\log(r/R_o)}{\log(R_w/R_o)} \quad (15)$$

For the numerical discretization, the Galerkin finite element method was used on a one-dimensional uniform grid. The stencil for node  $i$  is obtained by multiplying the governing equation by the weight function  $\phi_i$  and integrating over the domain of influence, or

$$\int_{r_{i-1}}^{r_{i+1}} \phi_i \left[ \frac{\partial^2 h}{\partial r^2} + \frac{1}{r} \frac{\partial h}{\partial r} \right] dr \quad (16)$$

Upon discretization and after dividing through by the term generated by the temporal derivative (i.e.,  $\frac{\Delta r_{i+1} + \Delta r_i}{2}$ ), the discretization of each term at each interior node was

$$\begin{aligned} h_{rr} &\approx \left( \frac{h_{i+1} - h_i}{\Delta r_{i+1}} - \frac{h_i - h_{i-1}}{\Delta r_i} \right) \frac{2}{\Delta r_{i+1} + \Delta r_i} \\ \frac{h_r}{r} &\approx \left( \frac{h_{i+1} - h_i}{(\Delta r_{i+1})^2} \left[ r_{i+1} \ln \left( \frac{r_{i+1}}{r_i} \right) - \Delta r_{i+1} \right] + \frac{h_i - h_{i-1}}{(\Delta r_i)^2} \left[ \Delta r_i - r_{i-1} \ln \left( \frac{r_i}{r_{i-1}} \right) \right] \right) \frac{2}{\Delta r_{i+1} + \Delta r_i} \end{aligned} \quad (17)$$

where  $h_{rr}$  is the second spatial derivative of  $h$ ,  $\Delta r_{i+1} = r_{i+1} - r_i$  and  $\Delta r_i = r_i - r_{i-1}$ .

To determine the order of accuracy of these stencils, these formulae are expanded using Taylor's series expansions about a common point, such as  $h_i$ . The stencil for the term  $\frac{\partial^2 h}{\partial r^2}$  reduces to

$$h_{rr} + h_{rrr} \frac{\Delta r_{i+1} - \Delta r_i}{3} + h_{rrrr} \frac{\Delta r_{i+1}^2 - \Delta r_{i+1} \Delta r_i + \Delta r_i^2}{12} + O(\Delta r_{i+1}^3, \Delta r_i^3) \quad (18)$$

The stencil for the term  $h_r/r$  is

$$\frac{h_r}{r} + \left[ \frac{h_r}{3r^2} + \frac{h_{rr}}{2r} \right] (\Delta r_{i+1} - \Delta r_i) + \left[ \frac{h_r}{12r^2} - \frac{h_{rr}}{12r^2} + \frac{h_{rrr}}{12r} \right] (\Delta r_{i+1}^2 - \Delta r_{i+1} \Delta r_i + \Delta r_i^2) + O(\Delta r_{i+1}^3, \Delta r_i^3) \quad (19)$$

The term  $(\Delta r_{i+1} - \Delta r_i)$  is zero for uniform grids and much smaller than  $\Delta r$  for grids whose spacing grows geometrically, so that the influence of this term is undetectable, making this discretization second-order accurate. In addition, the third order term in both expansions has a factor of  $(\Delta r_{i+1} - \Delta r_i)$ , so that the third order error term is zero for uniform grids.

For this problem, the well had a radius of 0.02 units and the outer boundary was set at 1 units. The piezometric head at the well was 0, and the head at the outer boundary was 5 units, which allowed for a Dirichlet boundary condition. Using the exact solution as the target for a traditional grid refinement study, the error is measured as the difference between the highly converged computed solution and the exact solution, or

$$L_{error}^1 = \sum_{i=1}^N |h_{exact}^i - h_{computed}^i| \frac{r_{i+1} - r_{i-1}}{2} \quad (20)$$

Using this definition for the error, the order of accuracy can be verified as second-order, as is shown in Table 1, where  $N$  is the number of intervals. Using equation (4), the order of accuracy based on these errors is shown in the right column, and asymptotically approaches 2.

N	Error	Accuracy
100	0.02722986504	NA
200	0.00676605012	2.00880
400	0.00168880297	2.00231
800	0.00042202945	2.00059
1600	0.00010549746	2.00014

Table 1. Error between Computed and Exact Solution.

Since an analytic solution exists for this set of equations, the choice of function for the method of manufactured solutions could be that function. But for illustration purposes, the manufactured solution was chosen as  $h(r) = r^{10}$ . The source term for this function, then, is

$$h_{rr} + \frac{h_r}{r} = 90r^8 + \frac{10r^9}{r} = 100r^8 \quad (21)$$

Applying the finite element method to this source term, the contribution to the discrete residual is

$$\int_{r_{i-1}}^{r_{i+1}} \phi_i 100r^8 dr = \frac{90r_i^{10} + 10r_{i+1}^{10} - 100r_{i+1}r_i^9}{9\Delta r_{i+1}} + \frac{90r_i^{10} + 10r_{i-1}^{10} - 100r_{i-1}r_i^9}{9\Delta r_i} \quad (22)$$

This source term is simple enough to calculate by hand, but for more complicated functions, symbolic manipulation software such as Mathematica can be easily used to perform this task, and then output the results in FORTRAN or C format for direct copying into the code. After implementing this source term, the same example was run using Dirichlet boundary conditions consistent with the manufactured solution. The error based on the residual formulation of the method of manufactured solutions is shown in Table 2 and is defined as

$$L^1_{residualerror} = \sum_{i=1}^N |Residual_i| \quad (23)$$

Since the residual already includes the area contribution, it is not needed for the error calculation. As can be seen in Table 2, the order of accuracy from one grid to the next approaches 2 as the grid spacing is halved, which demonstrates second-order accuracy.

N	Residual Error	Accuracy
100	0.00064080995	NA
200	0.00016296669	1.9753
400	0.00004109210	1.9876
800	0.00001031714	1.9938
1600	0.00000258484	1.9969

Table 2. Residual Error For Porous Media Equation.

Similarly, the error between the converged solutions and the manufactured solution is shown in Table 3. This error also demonstrates second-order accuracy. However, due to the conditional stability of this algorithm, the computational cost of obtaining the converged solutions was quite large, even for this one-dimensional problem; whereas for the residual formulation, the computational cost was the cost of performing one iteration of the flow solver.

N	Variable Error	Accuracy
100	0.00009288762	NA
200	0.00002316852	2.00332
400	0.00000578882	2.00082
800	0.00000144699	2.00021
1600	0.00000036173	2.00005

Table 3. Variable Error Using Method of Manufactured Solutions.

Another advantage of the residual form of MMS is the ability to isolate and study each component of the flow solver. For instance, the order of accuracy for the first term  $h_{rr}$  can be studied without the influence of the other term. Or, if more complicated boundary conditions are imposed, their influence and order of accuracy can be studied without reference to the interior nodes. In particular, for the term  $h_{rr}$ , the residual form of MMS using the function  $h(r) = r^{10}$  results in errors that are on the order of machine precision. When investigated further, it was discovered that for a uniform grid, this solution is an exact solution of the resultant MMS equation shown in Eqn (23), or

$$h_{i+1} - 2h_i + h_{i-1} = 18r_i^{10} + r_{i+1}^{10} + r_{i-1}^{10} - 10r_i^9(r_{i+1} + r_{i-1}) \quad (24)$$

when  $h_{i+1} = r_{i+1}^{10}$ ,  $h_i = r_i^{10}$  and  $h_{i-1} = r_{i-1}^{10}$ . When the second term  $h_r/r$  was studied, the error using the residual formulation reduced at a rate consistent with a second-order discretization. When the flow solver was run with only this term activated, it was discovered that this isolated stencil is unconditionally unstable whether it is an implicit or explicit discretization, which agrees with the von Neumann analysis for this stencil. Hence, the method of manufactured solutions could not be used in the original formulation to determine the order of accuracy of the implementation of  $h_r/r$ , but the residual formulation could be used.

## VI. 1D St. Venant Equations

The St. Venant Equations model flow in channels and rivers and represent a simplification of the Navier-Stokes Equations by assuming a hydrostatic pressure distribution in the vertical direction. The flow variables are the depth  $h$  and the discharge rate  $p$ , and the coefficients are Manning's friction coefficient  $n$ , gravity  $g$ , the bed slope  $S_o$  and the hydraulic radius  $R_w$ . The one-dimensional St. Venant Equations can be expressed as

$$\begin{aligned} \frac{\partial h}{\partial t} + \frac{\partial p}{\partial x} &= 0 \\ \frac{\partial p}{\partial t} + \frac{\partial}{\partial x} \left( \frac{p^2}{h} + \frac{1}{2}gh^2 \right) &= gh \left( S_o - \frac{n^2 p^2}{h^2 R_w^{4/3}} \right) \end{aligned} \quad (25)$$

These equations were discretized using the flux difference splitting finite volume method similar to the method used by Whitfield<sup>7</sup> for the 2D Shallow Water Equations. Using a manufactured solution of  $h(x) = e^{-x/1500}$  and  $p(x) = e^{x/1500}$ , where 1500 is the length of the channel, the code was verified on a sequence of grids using an  $L_2$ -norm of the difference between the computed and exact manufactured solutions, or

$$E_1(Q^e, Q^i) = \sum_{j=0}^N \Delta x_j (|h^e(x_j) - h^i(x_j)| + |p^e(x_j) - p^i(x_j)|) \quad (26)$$

where  $\Delta x_j$  is the length of the control volume associated with node  $j$ . The results of this error norm, which is the result of the original formulation of the method of manufactured solutions is shown in Table 4.

N	Variable Error	Accuracy	Comp. Cost
100	0.042862708	N.A.	5.32
200	0.010705630	2.00135	18.26
400	0.002675181	2.00066	69.24
800	0.000668642	2.00033	235.52
1600	0.000167141	2.00017	910.08
3200	0.000041783	2.00008	1588.71

Table 4. Error based on Original Formulation.

where the third column represents the order of accuracy as determined via Equation (4). As the grid spacing is successively halved, the order of accuracy of this implementation approaches second order. In the fourth column, the computational cost (in seconds) of obtaining a highly converged solution is provided and shows that the cost for the largest grid is almost half an hour.

Using the residual form of the method of manufactured solutions, the error is the  $L_1$ -norm of the residual at each grid location for the exact solution, or

$$E_2(Q^e) = \sum_{j=0}^N |R_1(Q^e)_j| + |R_2(Q^e)_j| \quad (27)$$

where  $R_1$  and  $R_2$  are the results for the discretization of the first and second equation in the St. Venant Equations. These errors are tabulated in Table 5, and again demonstrate second-order accuracy. From this example, the order of accuracy is the same for the residual form and the original formulation of the method of manufactured solutions, indicating that the order of accuracy calculated from the residual form of the method of manufactured solutions is as useful as the order of accuracy calculated from the original formulation.



N	Residual Error	Accuracy	Comp. Cost
100	0.000729183538	N.A.	0.00363
200	0.000188734839	1.94992	0.00723
400	0.000048006195	1.97507	0.01445
800	0.000012105478	1.98756	0.02882
1600	0.000003039431	1.99379	0.05775
3200	0.000000761495	1.99689	0.11564

Table 5. Error based on Residual Form.

The fourth column of Table 5 shows the computational cost (in seconds) of the residual formulation. These costs are orders of magnitude less than the cost of the original formulation, due to the fact that the residual must be calculated only once for this method, instead of repeatedly as the solver iterates to obtain the converged solution. In addition, the Jacobian calculation is not needed, and the solution to the matrix equation is not required, both of which reduce the computational cost. This computational savings will be even more pronounced when applied to two- and three-dimensional simulations.

N	Variable Error	Accuracy
100	0.000142583963	N.A.
200	0.000017693074	3.01055
400	0.000002212223	2.99961
800	0.000000276456	3.00038
1600	0.000000034550	3.00029
3200	0.000000004318	3.00025

Table 6. Third-Order Error based on Original Formulation.

As further demonstration, if the slope  $S_o$  and the friction coefficient  $n$  are set to 0, then third and fourth order compact stencils are available. Using the MUSCL-scheme<sup>7</sup>, a third order stencil can be obtained. The source terms must be eliminated since their discretization is only second-order accurate and would dominate the higher-order effects. The  $L_1$ -error based on the difference between the exact and computed solutions is shown in Table 6, and the error based on the residuals is shown in Table 7. Again, both methods show a third-order convergence rate.

N	Residual Error	Accuracy
100	0.0000035165437	N.A.
200	0.0000004362233	3.01102
400	0.0000000543199	3.00551
800	0.0000000067770	3.00276
1600	0.0000000008463	3.00138
3200	0.0000000001058	3.00050

Table 7. Third-Order Error based on Residual Form.

A fourth-order stencil can be obtained by using two values to the left and to the right of each interface. However, this stencil is not stable, so that converged solutions could not be compared with the exact solution. However, the order of accuracy of the stencil can be determined via the residual form of MMS, as is shown in Table 8. As these results show, the fourth order accurate scheme is so accurate that the influence of round-off error is seen starting at the grid with 800 nodes, and this influence dominates for the two most resolved grids. However, for the first three grids, the order of accuracy is clearly fourth order.

N	Residual Error	Accuracy
100	0.00000000459628	N.A.
200	0.0000000028494	4.0117
400	0.0000000001774	4.0056
800	0.0000000000201	3.1417
1600	0.0000000000326	N.A.
3200	0.0000000000637	N.A.

Table 8. Fourth-Order Error based on Residual Form.

## VII. 2D Unstructured Euler Equations

The final example deals with an two-dimensional unstructured finite volume flow solver of the steady-state incompressible Euler equations, which are show below

$$\begin{aligned}
 \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} &= 0 \\
 \frac{\partial(u^2 + P)}{\partial x} + \frac{\partial uv}{\partial y} &= 0 \\
 \frac{\partial uv}{\partial x} + \frac{\partial(v^2 + P)}{\partial y} &= 0
 \end{aligned} \tag{28}$$

where  $u$  and  $v$  are the velocities in the  $x$  and  $y$  directions and  $P$  is the dynamic pressure. The manufactured solutions were chosen to be exponentials as is shown below

$$\begin{aligned}
 P^e &= e^{2x+2y} \\
 u^e &= e^{x+y} \\
 v^e &= e^{x+y}
 \end{aligned} \tag{29}$$

The computational domain was an equilateral triangle subdivided equally into equilateral sub-triangles, as is shown in Figure 1. In addition, Murali<sup>5,6</sup> applied this approach to a triangulated square grid and to a grid about a NACA0012 airfoil. The sequence of grids was obtained by using h-refinement where each edge was split in the middle and each triangle was replaced by 4 smaller triangles.

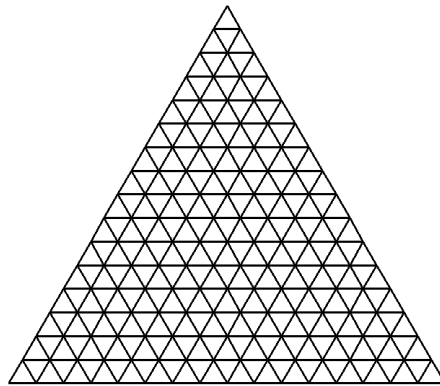


Figure 1. Equilateral Triangle.

The number of nodes and the number of triangles for each grid is shown in Table 9, ranging from excessively coarse to highly refined.

Grid Level	Nodes	Triangles
1	15	16
2	45	64
3	153	256
4	561	1024
5	2145	4096
6	8385	16384
7	33153	65536

Table 9. Nodes and Triangles in Equilateral Triangles.

Both inviscid and viscous simulations were studied using both the original and residual forms of the method of manufactured solutions. The flux differencing formulation of the finite volume method was used for these discretizations, and are further described by Hyams<sup>8,9</sup>.

The results for the inviscid case are shown in Tables 10, 11, and 12 using an  $L_1$  norm for the difference between the computed solutions and the exact solution for each variable. The order of accuracy is clearly converging towards 2, even though for the coarser grids the observed order of accuracy is different from 2.

Triangles	Pressure $P$ Error	Accuracy
16	0.0000105048	N.A.
64	0.00000247905	2.083
256	0.00000627157	1.983
1024	0.00000159472	1.975
4096	0.000000402268	1.987
16384	0.000000101005	1.994
65536	0.0000000253041	1.997

Table 10. Pressure Error for Euler Equations.

Triangles	X-Velocity $u$ Error	Accuracy
16	0.00000174646	N.A.
64	0.000000615283	1.505
256	0.000000202460	1.604
1024	0.0000000649553	1.640
4096	0.0000000195274	1.734
16384	0.00000000540187	1.854
65536	0.00000000142033	1.927

Table 11. X-Velocity Error for Euler Equations.

Triangles	Y-Velocity $v$ Error	Accuracy
16	0.00000391432	N.A.
64	0.000000678723	2.528
256	0.000000133879	2.342
1024	0.0000000389832	1.780
4096	0.0000000117515	1.730
16384	0.00000000331684	1.825
65536	0.000000000883976	1.908

Table 12. Y-Velocity Error for Euler Equations.

The error associated with each equation in the set of governing equations is shown in Tables 13, 14 and 15. The order of accuracy seen from the residual equations matches the order of accuracy from the converged solutions, and it is better behaved in that the error ratio monotonically converges towards the ideal value and the influence of the higher order error terms can be clearly seen.

Triangles	Continuity Residual Error	Accuracy
16	0.0009337660	N.A.
64	0.0003753253	1.315
256	0.0001136083	1.724
1024	0.00003101255	1.873
4096	0.000008088478	1.939
16384	0.000002064616	1.970
65536	0.0000005214924	1.985

Table 13. Continuity Residual Error for Euler Equations.

Triangles	X-Momentum Residual Error	Accuracy
16	0.0002495181	N.A.
64	0.00008251329	1.596
256	0.00002329190	1.825
1024	0.000006166476	1.917
4096	0.000001585209	1.960
16384	0.0000004017942	1.980
65536	0.0000001011372	1.990

Table 14. X-Momentum Residual Error for Euler Equations.

Triangles	Y-Momentum Residual Error	Accuracy
16	0.0003263460	N.A.
64	0.0001035595	1.657
256	0.00002876701	1.848
1024	0.000007564698	1.927
4096	0.000001939237	1.963
16384	0.0000004909055	1.982
65536	0.0000001234934	1.991

Table 15. Y-Momentum Residual Error for Euler Equations.

## VIII. Conclusion

A new, more efficient formulation for the method of manufactured solutions (MMS) has been presented. The method of manufactured solutions is a widely used means of determining the order of accuracy of an implementation, in order to determine whether the algorithm has been correctly implemented. The original formulation of the method of manufactured solutions required converged solutions on a sequence of refined grids, which was computationally expensive to obtain; and for realistic problems, obtaining a sufficiently converged solution may not even be possible.

The new formulation only considers the residual of the discretized equations and not the converged solution in order to determine the order of accuracy. Hence, the computational cost of the method is significantly reduced. Furthermore, this new formulation can be applied to individual components of the flow solver, such as the viscous terms or the boundary conditions, without reference to the other components, so that the order of accuracy of each piece of a flow solver can be individually analyzed.

This new formulation was demonstrated on three different problems and was shown to yield the same order of accuracy as the original formulation. These illustrative problems included one-dimensional flow through porous media which is an example of a parabolic partial differential equation and was solved using a Galerkin finite element approach. The second example dealt with the one-dimensional St. Venant equations, which are hyperbolic in nature and was discretized using a finite volume approach. In addition to second order results, third and fourth order spatial accuracy was demonstrated for this problem. The third case dealt with the two-dimensional incompressible Euler equations solved via node-centered finite volume approach on unstructured triangulated grids. For each of these cases, the residual formulation showed the same order of accuracy as the original formulation of the method of manufactured solutions, and examples of how the residual formulation can be used to study individual components of the flow solver were provided as well.

## References

- <sup>1</sup>Steinberg, S., and Roache, P. J., *Symbolic Manipulation and Computational Fluid Dynamics*, AIAA Journal, vol. 22, pp. 1390-1394, Oct. 1984.
- <sup>2</sup>Roache, P. J., *Verification of Codes and Calculations*, AIAA Journal, vol. 36, pp. 696-702, May, 1998.
- <sup>3</sup>Eca, L., and Hoekstra, M., *An Evaluation of Verification Procedures for CFD Applications*, 24th Symposium on Naval Hydrodynamics, Fukuoka, Japan, July, 2002.
- <sup>4</sup>Roy, C. J., Smith, T. M., and Ober, C. C., *Verification of a Compressible CFD Code Using the Method of Manufactured Solutions*, AIAA Paper 2002-3110, 32nd AIAA Fluid Dynamics Conf., St. Louis, June, 2002.
- <sup>5</sup>Murali, V.K., *Code Verification Using the Method of Manufactured Solutions*, Master's Thesis, Mississippi State Univ., Dec., 2002.
- <sup>6</sup>Murali, V. K., Burg, C. O. E., *Verification of 2D Navier-Stokes Codes by the Method of Manufactured Solutions*, AIAA Paper 2002-3109, 32nd AIAA Fluid Dynamics Conf., St. Louis, June, 2002.
- <sup>7</sup>Whitfield, D. L., *Numerical Solution of the Shallow Water Equations*, Mississippi State University Report, MSSU-EIRS-ERC-96-4, July, 1996.
- <sup>8</sup>Hyams, D. G., *An Investigation of Parallel Implicit Solution Algorithms for Incompressible Viscous Flow Problems*, Ph.D. Dissertation, Mississippi State University, May, 2000.
- <sup>9</sup>Hyams, D. G., Sreenivas, K., Sheng, C., Briley, W. R., Marcum, D. L., and Whitfield, D. L., *An Investigation of Parallel Implicit Solution Algorithms for Incompressible Flows on Multi-element Unstructured Topologies*, AIAA Paper-2000-0271, 38th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Jan., 2000.