

Code Verification for Practically Singular Equations

Brian A. Freno^a and Neil R. Matula^a

^a*Sandia National Laboratories, Albuquerque, NM 87185*

1. Introduction

The method-of-moments implementation of the electric-field integral equation (EFIE) yields many code-verification challenges due to the various sources of numerical error and their possible interactions [1, 2]. Matters are further complicated by singular integrals, which arise from the presence of a Green's function. To address these singular integrals, an approach is presented in [1] wherein both the solution and Green's function are manufactured. Because the arising equations are poorly conditioned, they are reformulated as a set of constraints for an optimization problem that selects the solution closest to the manufactured solution. In this paper, we demonstrate how, for such practically singular systems of equations, computing the truncation error by inserting the exact solution into the discretized equations cannot detect certain orders of coding errors. On the other hand, the discretization error from the optimal solution [1] is a more sensitive metric that can detect orders less than those of the expected convergence rate.

2. Truncation and Discretization Error

Let $\mathcal{L}(\cdot)$ and $\mathcal{L}_h(\cdot)$ respectively denote the operators representing the continuous and discretized equations. The truncation error $\tau_h(\cdot)$ is the difference between them [3, 4]:

$$\mathcal{L}_h(\mathbf{v}) = \mathcal{L}(\mathbf{v}) + \tau_h(\mathbf{v}). \quad (1)$$

Let \mathbf{u} denote the solution to the continuous equations ($\mathcal{L}(\mathbf{u}) = \mathbf{0}$) and \mathbf{u}_h denote the solution to the discretized equations ($\mathcal{L}_h(\mathbf{u}_h) = \mathbf{0}$). Inserting a Taylor-series expansion of $\mathcal{L}(\mathbf{v})$ about $\mathbf{v} = \mathbf{u}$ into (1) and evaluating at $\mathbf{v} = \mathbf{u}_h$ yields

$$\left. \frac{\partial \mathcal{L}}{\partial \mathbf{v}} \right|_{\mathbf{u}} \cdot \mathbf{e}_h + \mathcal{O}(\|\mathbf{e}_h\|^2) = -\tau_h(\mathbf{u}_h), \quad (2)$$

where $\mathbf{e}_h = \mathbf{u}_h - \mathbf{u}$ is the discretization error. Therefore, (2) relates the truncation error to the discretization error.

The truncation error can be computed from (1) by setting $\mathbf{v} = \mathbf{u}$:

$$\tau_h(\mathbf{u}) = \mathcal{L}_h(\mathbf{u}). \quad (3)$$

As described in Section 5.5.6.1 of [4], evaluating (3) across a series of systematically refined discretizations can be used to compute the convergence rate of \mathbf{e}_h without solving for \mathbf{u}_h . However, there are known drawbacks to using this approach [4].

3. The Method-of-Moments Implementation of the EFIE

The variational form of the EFIE is: find \mathbf{u} , such that

$$a(\mathbf{u}, \mathbf{v}) = (\mathbf{f}, \mathbf{v}) \quad \forall \mathbf{v} \in \mathbb{V}, \quad (4)$$

where \mathbf{f} is a source, \mathbb{V} is the test space, and the sesquilinear form and inner product are defined by

$$\begin{aligned} a(\mathbf{u}, \mathbf{v}) &= \alpha \int_S \bar{\mathbf{v}}(\mathbf{x}) \cdot \int_{S'} \mathbf{u}(\mathbf{x}') G(\mathbf{x}, \mathbf{x}') dS' dS + \beta \int_S \nabla \cdot \bar{\mathbf{v}}(\mathbf{x}) \int_{S'} \nabla' \cdot \mathbf{u}(\mathbf{x}') G(\mathbf{x}, \mathbf{x}') dS' dS, \\ (\mathbf{u}, \mathbf{v}) &= \int_S \mathbf{u}(\mathbf{x}) \cdot \bar{\mathbf{v}}(\mathbf{x}) dS. \end{aligned} \quad (5)$$

In (5), α and β are constants, G is a Green's function, and the bar notation denotes complex conjugation. If the domain is open, the component of \mathbf{u} normal to the boundary is zero.

To solve (4), we approximate \mathbf{u} with \mathbf{u}_h using a Galerkin method with Rao–Wilton–Glisson (RWG) basis functions $\phi_j(\mathbf{x})$ [5]: $\mathbf{u}_h(\mathbf{x}) = \sum_{j=1}^n u_j^h \phi_j(\mathbf{x})$, where n is the total number of unknowns. Defining \mathbb{V}_h to be the span of the RWG basis functions, the Galerkin approximation of the original problem is now: find $\mathbf{u}_h \in \mathbb{V}_h$, such that

$$a(\mathbf{u}_h, \phi_i) = (\mathbf{f}, \phi_i) \quad (6)$$

for $i = 1, \dots, n$.

To verify the order of accuracy of (6), we can manufacture \mathbf{f} [1] so that (6) becomes

$$a(\mathbf{u}_h, \phi_i) = a(\mathbf{u}, \phi_i). \quad (7)$$

In (7), \mathbf{u} is a manufactured solution that coerces \mathbf{u}_h to \mathbf{u} and permits the truncation error and discretization error to be measured. Let $\mathbf{u}_n(\mathbf{x}) = \sum_{j=1}^n u_j^n \phi_j(\mathbf{x})$ denote the basis-function representation of \mathbf{u} . The truncation error is

$$\tau_{hi}(\mathbf{u}) = a(\mathbf{u}_n, \phi_i) - (\mathbf{f}, \phi_i) = a(\mathbf{u}_n, \phi_i) - a(\mathbf{u}, \phi_i). \quad (8)$$

As an alternative to the continuous form of the discretization error $\mathbf{e}_h = \mathbf{u}_h - \mathbf{u}$, we can measure the discrete form of the discretization error

$$\mathbf{e}^h = \mathbf{u}^h - \mathbf{u}^n. \quad (9)$$

However, the integrals in (7) cannot be accurately computed due to the presence of the Green's function, which yields a singularity when $\mathbf{x} = \mathbf{x}'$. In [1], this challenge is mitigated by manufacturing the Green's function as well. As a result, the matrix becomes practically singular, admitting infinite solutions \mathbf{u}^h . Therefore, \mathbf{u}^h is chosen by selecting the closest choice to \mathbf{u}^n that satisfies (7). The implications for the truncation and discretization errors as code-verification metrics for such singular systems of equations are discussed in the next section.

4. Singular Systems of Equations

Given their similar properties, in [6], expectations for the RWG basis functions are based on those for rooftop basis functions. Likewise, in this section, we consider the one-dimensional analogy with piecewise linear basis functions $\phi_j(x)$. We restrict the solution to real numbers and set $S = [a, b]$ in (5). We consider the extreme case of $G(x, x') = 1$, which yields a singular system of equations. As a result, (7) becomes: find $u_h \in H_0^1$, such that

$$a(u_h, \phi_i) = a(u, \phi_i) \quad (10)$$

for $i = 1, \dots, n$, and

$$a(u, \phi_i) = \alpha \int_a^b \phi_i(x) \int_a^b u(x') dx' dx + \beta \int_a^b \frac{d\phi_i}{dx}(x) \int_a^b \frac{du}{dx'}(x') dx' dx. \quad (11)$$

Since $\int_a^b \frac{d\phi_i}{dx}(x) dx = 0$, (11) is reduced to

$$a(u, \phi_i) = \alpha \int_a^b \phi_i(x) \int_a^b u(x') dx' dx.$$

For a uniform discretization, after dividing both sides by αh , where $h = (b - a)/N$ and $N = n + 1$ is the number of elements, the system of equations for (10) is

$$h \begin{bmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{bmatrix} \begin{Bmatrix} u_1^h \\ \vdots \\ u_n^h \end{Bmatrix} = \int_a^b u(x) dx \begin{Bmatrix} 1 \\ \vdots \\ 1 \end{Bmatrix}, \quad (12)$$

which can be written as $\mathcal{L}_h(u_h) = \mathbf{A}\mathbf{u}^h - \mathbf{b} = \mathbf{0}$, where $\mathbf{A} = h\mathbf{1}_{n \times n}$ and $\mathbf{b} = \left(\int_a^b u(x)dx\right) \mathbf{1}_{n \times 1}$. For the truncation error, the division by h ensures that

$$\left. \frac{\partial \mathcal{L}_i}{\partial v} \right|_u \cdot e_h = \frac{1}{\alpha h} a(e_h, \phi_i) = \frac{1}{h} \int_a^b \phi_i(x) \int_a^b e_h(x') dx' dx = \int_a^b e_h(x) dx,$$

such that the truncation and discretization errors are of the same order in (2).

4.1. Original System

Accounting for the boundary conditions $u(a) = u(b) = 0$, the trapezoidal-rule approximation of the integral in (12) is

$$\int_a^b u(x)dx = h \sum_{j=1}^n u_j^n + \mathcal{O}(h^2), \quad (13)$$

such that $\mathbf{b} = \mathbf{A}\mathbf{u}^n + \mathcal{O}(h^2)$ and (12) becomes

$$\mathcal{L}_h(u_h) = \mathbf{A}(\mathbf{u}^h - \mathbf{u}^n) + \mathcal{O}(h^2) = \mathbf{0}. \quad (14)$$

From (3) and (14),

$$\tau_h(u) = \mathcal{L}_h(u) = \tilde{\tau}_h(u) + \mathcal{O}(h^2), \quad (15)$$

where

$$\tilde{\tau}_h(u) = \mathbf{A}(\mathbf{u}^n - \mathbf{u}^n) = \mathbf{0},$$

such that the truncation error $\tau_h(u)$ is $\mathcal{O}(h^2)$, as expected. Because \mathbf{A} is singular, a unique solution cannot be computed and a meaningful discretization error cannot be obtained.

If an error $\delta \neq 0$ that is $\mathcal{O}(h^r)$, where $r \geq 0$, is introduced in $A_{i,j}$ in (12), such that it becomes $(1+\delta)A_{i,j}$, $\tilde{\tau}_h(u)$ in the truncation error (15) becomes

$$\tilde{\tau}_h(u) = \delta u_j^n h \begin{Bmatrix} \mathbf{0}_{i-1 \times 1} \\ \mathbf{1}_{1 \times 1} \\ \mathbf{0}_{n-i \times 1} \end{Bmatrix}. \quad (16)$$

If j is fixed, as n is increased, x_j will approach a and $u_j^n = u(x_j)$ will be asymptotically proportional to the leading term of the Taylor series expansion at $x = a + jh$ about $x = a$. If that term is $\mathcal{O}(h^q)$, $\tau_h(u)$ (15) will be $\mathcal{O}(h^{\min\{q+r+1, 2\}})$. If the error is introduced in a column corresponding to a fixed x location (e.g., $j = N/2$), $\tau_h(u)$ will be $\mathcal{O}(h^r + 1)$. Therefore, if j is fixed and $q + r \geq 1$ or the error is spatially fixed and $r \geq 1$, the error will go undetected.

4.2. Reduced System

To perform code verification for a singular system of equations, such as that in (12), an approach is presented in [1], which uses a QR factorization of the transpose of the matrix to select the \mathbf{u}^h closest to \mathbf{u}^n that satisfies the system of equations. To simplify the factorization, we divide both sides of (12) by h once more:

$$\begin{bmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{bmatrix} \begin{Bmatrix} u_1^h \\ \vdots \\ u_n^h \end{Bmatrix} = \frac{1}{h} \int_a^b u(x)dx \begin{Bmatrix} 1 \\ \vdots \\ 1 \end{Bmatrix}, \quad (17)$$

which can be written as $\mathbf{A}\mathbf{u}^h = \mathbf{b}$, where $\mathbf{A} = \mathbf{1}_{n \times n}$ and $\mathbf{b} = \frac{1}{h} \left(\int_a^b u(x)dx\right) \mathbf{1}_{n \times 1}$. Performing a pivoted QR factorization of \mathbf{A}^T , such that

$$\mathbf{A}^T \mathbf{P} = [\mathbf{Q}_1, \mathbf{Q}_2] \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{bmatrix} = \mathbf{Q}_1 \mathbf{R}_1,$$

the optimal solution is

$$\mathbf{u}^h = \mathbf{Q}_1 \mathbf{u}' + \mathbf{Q}_2 \mathbf{Q}_2^T \mathbf{u}^n, \quad (18)$$

where \mathbf{u}' is the solution to (17) when the size is reduced to its rank:

$$\mathbf{u}' = (\mathbf{R}_1^T)^\dagger \mathbf{P}^T \mathbf{b},$$

where $(\cdot)^\dagger$ denotes the Moore–Penrose pseudoinverse. Noting that $\mathbf{Q}\mathbf{Q}^T = \mathbf{Q}_1 \mathbf{Q}_1^T + \mathbf{Q}_2 \mathbf{Q}_2^T = \mathbf{I}_{n \times n}$, (18) can be written as

$$\mathbf{u}^h = \mathbf{Q}_1 \mathbf{u}' - \mathbf{Q}_1 \mathbf{Q}_1^T \mathbf{u}^n + \mathbf{u}^n. \quad (19)$$

4.2.1. Without an Error

The factorization is $\mathbf{P} = \mathbf{I}_{n \times n}$, $\mathbf{Q}_1 = n^{-1/2} \mathbf{1}_{n \times 1}$, $\mathbf{R}_1 = n^{1/2} \mathbf{1}_{1 \times n}$, and $(\mathbf{R}_1^T)^\dagger = n^{-3/2} \mathbf{1}_{1 \times n}$. The first term of (19) is

$$\mathbf{Q}_1 \mathbf{u}' = \mathbf{Q}_1 (\mathbf{R}_1^T)^\dagger \mathbf{P}^T \mathbf{b} = \frac{1}{n^2} \mathbf{1}_{n \times n} \mathbf{b} = \frac{1}{nh} \left(\int_a^b u(x) dx \right) \mathbf{1}_{n \times 1}. \quad (20)$$

Inserting (13) into (20) yields

$$\mathbf{Q}_1 \mathbf{u}' = \frac{1}{n} \mathbf{1}_{n \times n} \mathbf{u}^n + \mathcal{O}(h^2). \quad (21)$$

The second term of (19) is

$$\mathbf{Q}_1 \mathbf{Q}_1^T \mathbf{u}^n = \frac{1}{n} \mathbf{1}_{n \times n} \mathbf{u}^n. \quad (22)$$

Inserting (21) and (22) into (19), the discretization error (9) is

$$\mathbf{e}^h = \tilde{\mathbf{e}}^h + \mathcal{O}(h^2), \quad (23)$$

where

$$\tilde{\mathbf{e}}^h = \left(\frac{1}{n} \mathbf{1}_{n \times n} - \frac{1}{n} \mathbf{1}_{n \times n} + \mathbf{I}_{n \times n} - \mathbf{I}_{n \times n} \right) \mathbf{u}^n = \mathbf{0}.$$

The discretization error \mathbf{e}^h is $\mathcal{O}(h^2)$, as expected.

4.2.2. With an Error

We now consider the case when an error $\delta \neq 0$ is introduced in $A_{i,j}$ in (17), such that it becomes $(1 + \delta)A_{i,j}$. If $i = 1$, the factorization is $\mathbf{P} = \mathbf{I}_{n \times n}$,

$$\mathbf{Q}_1 = \frac{1}{\gamma} \begin{bmatrix} \mathbf{1}_{j-1 \times 1} & \eta \xi^{-1/2} \mathbf{1}_{j-1 \times 1} \\ \eta \mathbf{1}_{1 \times 1} & -\xi^{1/2} \mathbf{1}_{1 \times 1} \\ \mathbf{1}_{n-j \times 1} & \eta \xi^{-1/2} \mathbf{1}_{n-j \times 1} \end{bmatrix},$$

$$\mathbf{R}_1 = \frac{1}{\gamma} \begin{bmatrix} \gamma^2 \mathbf{1}_{1 \times 1} & \zeta \mathbf{1}_{1 \times n-1} \\ \mathbf{0}_{1 \times 1} & \delta \xi^{1/2} \mathbf{1}_{1 \times n-1} \end{bmatrix},$$

$$(\mathbf{R}_1^T)^\dagger = \frac{1}{\delta \gamma} \begin{bmatrix} \delta \mathbf{1}_{1 \times 1} & \mathbf{0}_{1 \times n-1} \\ -\zeta \xi^{-1/2} \mathbf{1}_{1 \times 1} & \gamma^2 \xi^{-3/2} \mathbf{1}_{1 \times n-1} \end{bmatrix},$$

where $\xi = n - 1$, $\eta = 1 + \delta$, $\zeta = n + \delta$, and $\gamma = \sqrt{\xi + \eta^2}$. If $i \neq 1$, the factorization is $\mathbf{P} = \mathbf{I}_{n \times n}$,

$$\mathbf{Q}_1 = n^{-1/2} \begin{bmatrix} \mathbf{1}_{j-1 \times 1} & -\xi^{-1/2} \mathbf{1}_{j-1 \times 1} \\ \mathbf{1}_{1 \times 1} & \xi^{1/2} \mathbf{1}_{1 \times 1} \\ \mathbf{1}_{n-j \times 1} & -\xi^{-1/2} \mathbf{1}_{n-j \times 1} \end{bmatrix},$$

$$\mathbf{R}_1 = n^{-1/2} \begin{bmatrix} n \mathbf{1}_{1 \times i-1} & \zeta \mathbf{1}_{1 \times 1} & n \mathbf{1}_{1 \times n-i} \\ \mathbf{0}_{1 \times i-1} & \delta \xi^{1/2} \mathbf{1}_{1 \times 1} & \mathbf{0}_{1 \times n-i} \end{bmatrix},$$

$$(\mathbf{R}_1^T)^\dagger = \frac{n^{-1/2} \xi^{-3/2}}{\delta} \begin{bmatrix} \delta \xi^{1/2} \mathbf{1}_{1 \times i-1} & \mathbf{0}_{1 \times 1} & \delta \xi^{1/2} \mathbf{1}_{1 \times n-i} \\ -\zeta \mathbf{1}_{1 \times i-1} & n \xi \mathbf{1}_{1 \times 1} & -\zeta \mathbf{1}_{1 \times n-i} \end{bmatrix}.$$

The first term of (19) is

$$\begin{aligned} \mathbf{Q}_1 \mathbf{u}' &= \mathbf{Q}_1 (\mathbf{R}_1^T)^\dagger \mathbf{P}^T \mathbf{b} = \frac{1}{\delta \xi^2} \begin{bmatrix} \eta \mathbf{1}_{j-1 \times i-1} & -\xi \mathbf{1}_{j-1 \times 1} & \eta \mathbf{1}_{j-1 \times n-i} \\ -\xi \mathbf{1}_{1 \times i-1} & \xi^2 \mathbf{1}_{1 \times 1} & -\xi \mathbf{1}_{1 \times n-i} \\ \eta \mathbf{1}_{n-j \times i-1} & -\xi \mathbf{1}_{n-j \times 1} & \eta \mathbf{1}_{n-j \times n-i} \end{bmatrix} \mathbf{b} \\ &= \frac{1}{\xi h} \int_a^b u(x) dx \begin{Bmatrix} \mathbf{1}_{j-1 \times 1} \\ \mathbf{0}_{1 \times 1} \\ \mathbf{1}_{n-j \times 1} \end{Bmatrix}. \end{aligned} \quad (24)$$

Inserting (13) into (24) yields

$$\mathbf{Q}_1 \mathbf{u}' = \frac{1}{\xi} \begin{bmatrix} \mathbf{1}_{j-1 \times n} \\ \mathbf{0}_{1 \times n} \\ \mathbf{1}_{n-j \times n} \end{bmatrix} \mathbf{u}^n + \mathcal{O}(h^2). \quad (25)$$

The second term of (19) is

$$\mathbf{Q}_1 \mathbf{Q}_1^T \mathbf{u}^n = \left(\begin{bmatrix} \mathbf{0}_{j-1 \times j-1} & \mathbf{0}_{j-1 \times 1} & \mathbf{0}_{j-1 \times n-j} \\ \mathbf{0}_{1 \times j-1} & \mathbf{1}_{1 \times 1} & \mathbf{0}_{1 \times n-j} \\ \mathbf{0}_{n-j \times j-1} & \mathbf{0}_{n-j \times 1} & \mathbf{0}_{n-j \times n-j} \end{bmatrix} + \frac{1}{\xi} \begin{bmatrix} \mathbf{1}_{j-1 \times j-1} & \mathbf{0}_{j-1 \times 1} & \mathbf{1}_{j-1 \times n-j} \\ \mathbf{0}_{1 \times j-1} & \mathbf{0}_{1 \times 1} & \mathbf{0}_{1 \times n-j} \\ \mathbf{1}_{n-j \times j-1} & \mathbf{0}_{n-j \times 1} & \mathbf{1}_{n-j \times n-j} \end{bmatrix} \right) \mathbf{u}^n. \quad (26)$$

Inserting (25) and (26) into (19), $\tilde{\mathbf{e}}^h$ in the discretization error $\mathbf{e}^h = \tilde{\mathbf{e}}^h + \mathcal{O}(h^2)$ (23) is

$$\begin{aligned} \tilde{\mathbf{e}}^h &= \left(- \begin{bmatrix} \mathbf{0}_{j-1 \times j-1} & \mathbf{0}_{j-1 \times 1} & \mathbf{0}_{j-1 \times n-j} \\ \mathbf{0}_{1 \times j-1} & \mathbf{1}_{1 \times 1} & \mathbf{0}_{1 \times n-j} \\ \mathbf{0}_{n-j \times j-1} & \mathbf{0}_{n-j \times 1} & \mathbf{0}_{n-j \times n-j} \end{bmatrix} + \frac{1}{\xi} \begin{bmatrix} \mathbf{0}_{j-1 \times j-1} & \mathbf{1}_{j-1 \times 1} & \mathbf{0}_{j-1 \times n-j} \\ \mathbf{0}_{1 \times j-1} & \mathbf{0}_{1 \times 1} & \mathbf{0}_{1 \times n-j} \\ \mathbf{0}_{n-j \times j-1} & \mathbf{1}_{n-j \times 1} & \mathbf{0}_{n-j \times n-j} \end{bmatrix} \right) \mathbf{u}^n \\ &= u_j^n \left(- \begin{Bmatrix} \mathbf{0}_{j-1 \times 1} \\ \mathbf{1}_{1 \times 1} \\ \mathbf{0}_{n-j \times 1} \end{Bmatrix} + \frac{h}{b-a} \begin{Bmatrix} \mathbf{1}_{j-1 \times 1} \\ \mathbf{0}_{1 \times 1} \\ \mathbf{1}_{n-j \times 1} \end{Bmatrix} \right). \end{aligned} \quad (27)$$

If j is fixed, as n is increased, x_j will approach a and $u_j^n = u(x_j)$ will be asymptotically proportional to the leading term of the Taylor series expansion at $x = a + jh$ about $x = a$. If that term is $\mathcal{O}(h^q)$, (23) will be $\mathcal{O}(h^{\min\{q, 2\}})$. If the error is introduced in a column corresponding to a fixed x location, (23) will be $\mathcal{O}(h^0)$ if $u(x_j) \neq 0$. Therefore, if $q < 2$ or the error is spatially fixed, the error will be detected. Note that, while the column index of the error appears in (27), δ does not.

4.3. Summary

In the absence of coding errors, the truncation error computed from the singular equations and the discretization error computed from the optimal solution are both $\mathcal{O}(h^2)$. However, the discretization error is more effective at detecting coding errors than the truncation error.

For coding errors with an index-fixed column, the discretization error can detect errors if $q < 2$, whereas the truncation error can only detect errors if $q < 1 - r$. For coding errors with a physically fixed column, the discretization error can detect any errors, whereas the truncation error can only detect errors if $r < 1$.

Case	$u(x)$	q	Location of δ , (i, j)			
			a	b	c	d
1	$\sin \pi x$	1	—	(1, 2)	(2, $N/4$)	($N/2$, $N/2$)
2	$\sin \pi x^2$	2	—	(1, 2)	(2, $N/4$)	($N/2$, $N/2$)

Table 1: Simplified analogy: Cases.

Case	(α, β)	Location of δ , (i, j)			
		a	b	c	d
1	(1, 1)	—	(1, 2)	(2, $f(N)$)	($f(N)$, $f(N)$)
2	(1, 0)	—	(1, 2)	(2, $f(N)$)	($f(N)$, $f(N)$)
3	(0, 1)	—	(1, 2)	(2, $f(N)$)	($f(N)$, $f(N)$)

Table 2: EFIE: Cases.

5. Numerical Examples

We demonstrate the observations of Section 4 with numerical examples in this section for the simplified one-dimensional analogy to the EFIE (12), as well as for the EFIE (7).

5.1. Simplified Analogy

For the simplified analogy, we consider the cases in Table 1 for the domain $[a, b] = [0, 1]$. Cases 1a and 2a have no errors, Cases 1b and 2b have an error of $\delta = 1/20$ at fixed row and column indices, Cases 1c and 2c have the error at a fixed row index and spatially fixed column, and Cases 1d and 2d have the error at spatially fixed rows and columns. The leading term of the Taylor series expansion at $x = a + h$ about $x = a$ is $\mathcal{O}(h^q)$, where $q = 1$ for Case 1 and $q = 2$ for Case 2.

Figure 1 provides a comparison of the error metrics $\tau_h(u)$ from (15), \mathbf{e}^h from (23), $\tilde{\tau}_h(u) - \tau_h(u)$ from (15), and $\tilde{\mathbf{e}}^h - \mathbf{e}^h$ from (23). $\tau_h(u)$ is $\mathcal{O}(h^2)$ not only for Cases 1a and 2a, but also for Cases 1b and 2b, such that the fixed-index errors go undetected. Cases 1c, 1d, 2c, and 2d are $\mathcal{O}(h)$, such that the spatially fixed errors are detected, but consistency is falsely implied. \mathbf{e}^h is $\mathcal{O}(h^2)$ for Cases 1a and 2a, as well as Case 2b since the error introduced in Case 2b is $\mathcal{O}(h^2)$ since $q = 2$. Case 1b is $\mathcal{O}(h)$ and Cases 1c, 1d, 2c and 2d are $\mathcal{O}(h^0)$, such that those errors are detected. Finally, $\tilde{\tau}_h(u) - \tau_h(u)$ verifies the expression in (16) and $\tilde{\mathbf{e}}^h - \mathbf{e}^h$ verifies the expression in (27), as only the $\mathcal{O}(h^2)$ terms remain.

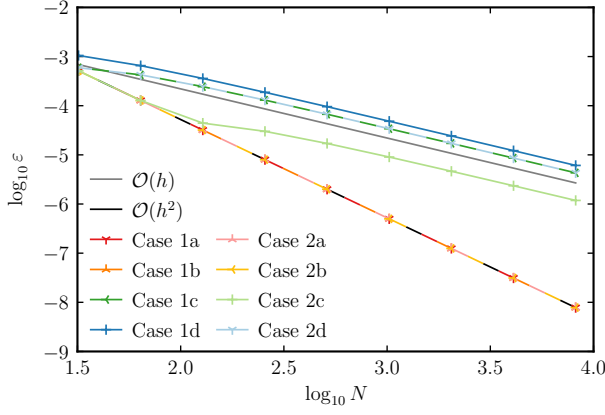
5.2. EFIE

For the EFIE, we consider the cases in Table 2 for the domain $(x, y) \in [-1, 1] \times [0, 1]$. For these cases, $\mathbf{u}(\mathbf{x}) = \{\cos(\pi x/2) \cos(\pi y/4), \cos(\pi x/4) \sin(\pi y)\}$, and $G(\mathbf{x}, \mathbf{x}') = 1 - \|\mathbf{x} - \mathbf{x}'\|_2^2/5$. If j is fixed, as n is increased, (x_j, y_j) will approach $(-1, 0)$, and the leading term of the Taylor series expansion is $\mathcal{O}(h)$. For the EFIE, N denotes the number of triangles used to discretize the domain and $h = 2/\sqrt{N}$.

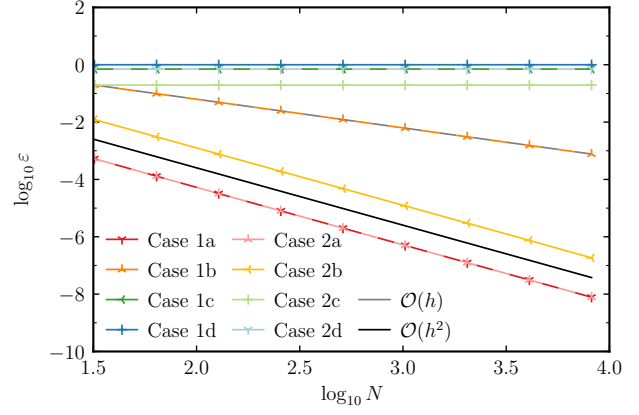
Cases 1a, 2a, and 3a have no errors. Cases 1b, 2b, and 3b have an error of $\delta = 1/100$ at fixed row and column indices. For Cases 1c, 2c, and 3c, the row index of the error is fixed, and the column indices $f(N)$ are those associated with the solution values at $x = 0$. For Cases 1d, 2d, and 3d, the row and column indices of the error are spatially fixed and denoted by $f(N)$.

Figure 2 provides a comparison of the error metrics $\tau_h(\mathbf{u})$ from (8) and \mathbf{e}^h from (9). For the EFIE, the truncation error is computed by factoring out h^2 , so that

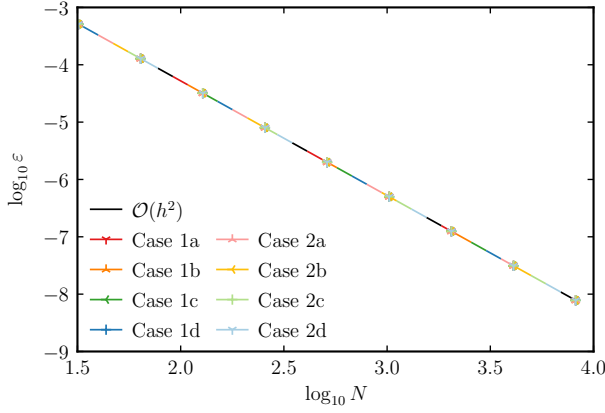
$$\left. \frac{\partial \mathcal{L}_i}{\partial \mathbf{v}} \right|_{\mathbf{u}} \cdot \mathbf{e}_h = \frac{a(\mathbf{e}_h, \phi_i)}{h^2},$$



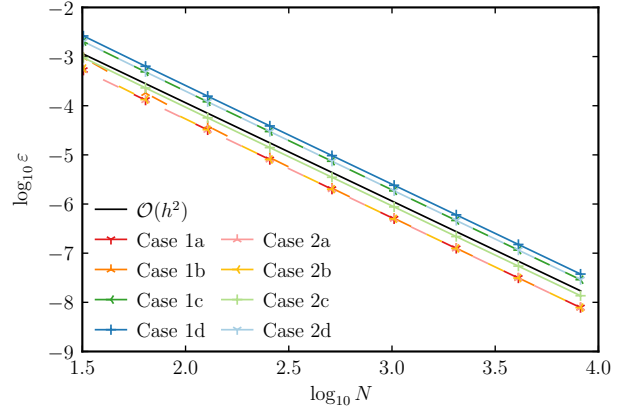
(a) $\varepsilon = \|\tau_h(u)\|_\infty$



(b) $\varepsilon = \|e^h\|_\infty$

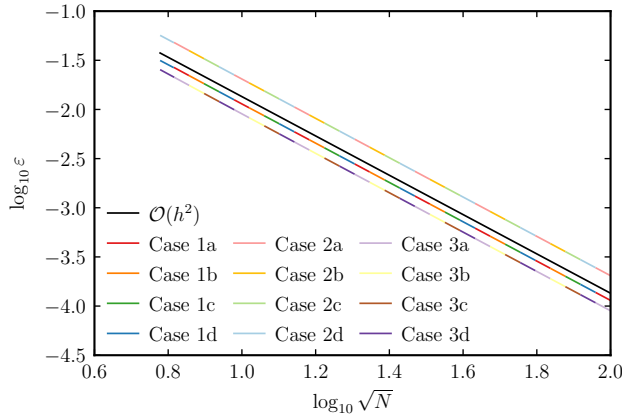


(c) $\varepsilon = \|\tilde{\tau}_h(u) - \tau_h(u)\|_\infty$

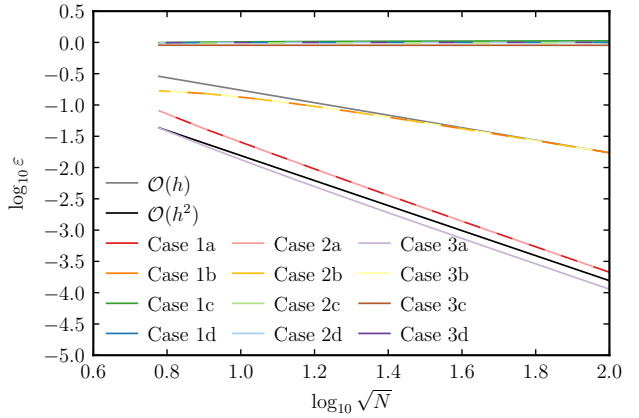


(d) $\varepsilon = \|\tilde{e}^h - e^h\|_\infty$

Figure 1: Simplified analogy: Convergence comparison of error metrics ε .



(a) $\varepsilon = \|\tau_h(u)\|_\infty$



(b) $\varepsilon = \|e^h\|_\infty$

Figure 2: EFIE: Convergence comparison of error metrics ε .

and the truncation and discretization errors are of the same order in (2). For the cases with δ ,

$$\tau_{h_i}(\mathbf{u}) = \frac{\delta u_j^n}{h^2} a(\phi_j, \phi_i) + \mathcal{O}(h^2). \quad (28)$$

In (28), $a(\phi_j, \phi_i)$ is $\mathcal{O}(h^4)$ due to the four-dimensional integrals of the basis functions. Therefore, $\tau_h(\mathbf{u})$ is $\mathcal{O}(h^2)$ not only for the cases without error, but for all of the cases with error, such that none of the errors are detected. \mathbf{e}^h is $\mathcal{O}(h^2)$ for Cases 1a, 2a, and 3a. Cases 1b, 2b, and 3b are $\mathcal{O}(h)$, and Cases 1c, 1d, 2c, 2d, 3c, and 3d are $\mathcal{O}(h^0)$, such that these errors are detected.

6. Conclusions

In this paper, we showed how, for a singular system of equations, computing the truncation error by inserting the exact solution into the discretized equations cannot detect certain orders of coding errors. However, the discretization error from the optimal solution is a more effective metric.

Acknowledgments

This article has been authored by employees of National Technology & Engineering Solutions of Sandia, LLC under Contract No. DE-NA0003525 with the U.S. Department of Energy (DOE). The employees own all right, title, and interest in and to the article and are solely responsible for its contents. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this article or allow others to do so, for United States Government purposes. The DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan <https://www.energy.gov/downloads/doe-public-access-plan>.

References

- [1] B. A. Freno, N. R. Matula, W. A. Johnson, Manufactured solutions for the method-of-moments implementation of the electric-field integral equation, *Journal of Computational Physics* 443 (2021). [doi:10.1016/j.jcp.2021.110538](https://doi.org/10.1016/j.jcp.2021.110538).
- [2] B. A. Freno, N. R. Matula, J. I. Owen, W. A. Johnson, Code-verification techniques for the method-of-moments implementation of the electric-field integral equation, *Journal of Computational Physics* 451 (2022). [doi:10.1016/j.jcp.2021.110891](https://doi.org/10.1016/j.jcp.2021.110891).
- [3] J. H. Ferziger, M. Perić, *Computational Methods for Fluid Dynamics*, Springer, 2002. [doi:10.1007/978-3-642-56026-2](https://doi.org/10.1007/978-3-642-56026-2).
- [4] W. L. Oberkampf, C. J. Roy, *Verification and Validation in Scientific Computing*, Cambridge University Press, 2010. [doi:10.1017/cbo9780511760396](https://doi.org/10.1017/cbo9780511760396).
- [5] S. Rao, D. Wilton, A. Glisson, Electromagnetic scattering by surfaces of arbitrary shape, *IEEE Transactions on Antennas and Propagation* 30 (3) (1982) 409–418. [doi:10.1109/TAP.1982.1142818](https://doi.org/10.1109/TAP.1982.1142818).
- [6] K. F. Warnick, *Numerical Analysis for Electromagnetic Integral Equations*, Artech House, 2008.