Seven Deadly Sins of Numerical Computation

Brian J. McCartin

1. INTRODUCTION. The modern Christian concept of Seven Deadly Sins [**1**, p. 22] is the culmination of a tradition dating back at least to Aristotle and the Stoics and with roots in Hellenistic Judaism as well. Although their precise number, version, and order have fluctuated over time, one concept has remained fixed: certain ruin will befall the sinner who does not avoid these vices.

That a similar situation obtains in numerical computation has been recognized for at least 40 years [**2**]. However, the faithful require a periodic reminder [**3**]. It is in this spirit that we herein deliver a sermon on Seven Deadly Sins of Numerical Computation.

Admittedly, the choice of offenses included reflects the prejudices of the author but this is natural since at one time or another he has committed each one of them. Lastly, just as the Seven Deadly Sins of Christianity correspond to the Seven Beatitudes, we offer advice for a virtuous life in numerical computing.

2. SEVEN DEADLY SINS. In the interest of breadth of coverage, one Deadly Sin has been selected from each of the areas: interpolation, data fitting, equation solving, numerical differentiation, numerical integration, ordinary differential equations, and partial differential equations.

2.1 Deadly Sin #1: Algorithmic Idolatry. Certain numerical algorithms constitute such a step beyond their predecessors and become so widespread in their application that numerical practitioners become blind to their shortcomings. Adherents of these techniques sometimes exhibit a religious fervor akin to idol worship. An excellent example is provided by polynomial splines.

One of the truly revolutionary concepts to emerge in 20th century numerical computing has been the introduction of the cubic spline by I. J. Schoenberg [**4**]. (Another, Richardson extrapolation, will be encountered in our treatment of Deadly Sin #4.) Not only did this resolve such outstanding problems as Runge's phenomenon (non-uniform convergence of equidistant interpolation by polynomials to real C^∞ functions) but it also laid the foundation for much of modern computer-aided design.

As major a leap forward as this represented, it was not a panacea. As is evident from Figure 1, despite the smoothness (C^2) and rapid convergence ($O(h^4)$ for equally spaced points a distance h apart), the cubic spline may violate the inherent monotonicity/convexity properties of the data (i.e., it is not shape-preserving—it can "wiggle"). This problem may be overcome by interpreting the cubic spline as a thin flexible beam [**5**] and adding sufficient tension to remove the offensive excursions (the spline in tension). To avoid an excessively "kinky" curve, the tension should not be applied uniformly [**6**] over the data set (the exponential spline).

A procedure for automatically applying just enough tension to remove "wiggles" is available [**7**]. The result of applying this algorithm is also shown in Figure 1. An appealing feature of this approach is that it subordinates the cubic spline. That is,

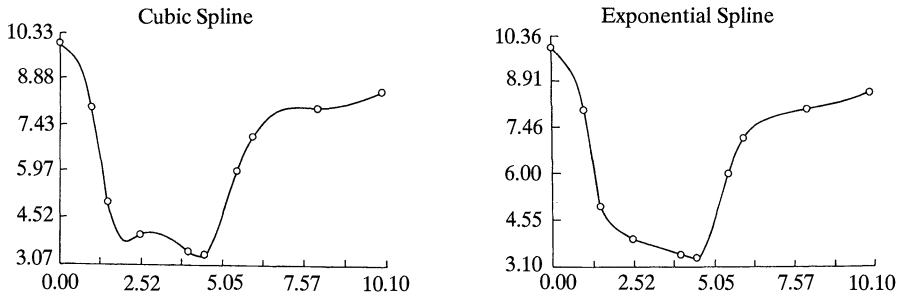

Figure 1. Getting Rid of Wiggles

the cubic spline supplies the zeroth iterate to the exponential spline tension parameter selection algorithm. Thus, if there are no violations of monotonicity/convexity then the cubic spline is accepted. The reader interested in further exploration of this topic might profitably begin with [**8**, p. 139].

2.2 Deadly Sin #2: Using An Inappropriate Model. It must be confessed that many numerical computors expend considerable effort in developing sophisticated numerical procedures for problem formulations that bear little resemblance to physical reality. In point of fact, it is challenging in the extreme to formulate an a priori mathematical model that reasonably conforms to a natural process. What typically transpires is the formulation of a tentative model which is then *validated* a posteriori against experimental data. It is at this stage that inadequacies in the problem formulation surface, resulting in a modification of the model. This process is iterated until adequate agreement is achieved between prediction and measurement. A prime example is the fitting of experimental measurements to theoretical models.

A pervasive problem in applied numerical computing is the fitting of free parameters appearing in theoretical models to experimentally measured data. Often, these parameters are not directly measurable and their values can be inferred only by fitting the models to data. However, the theoretical models typically fail to account for the finite precision of the measurements themselves. We are referring here to errors with nonzero mean that arise from pervasive sources such as thermal noise.

For example, excited atoms in a gas emit light with spectral lineshape given by the Lorentz distribution

$$I(\lambda) = \frac{I_0}{1 + 4(\lambda - \lambda_0)^2/\Gamma^2}, \tag{1}$$

where I is the intensity, λ is the wavelength, λ_0 is the resonant wavelength, Γ is the full width at half maximum, and I_0 is the resonant intensity [**9**, p. 138]. Given measurements of $I(\lambda)$, we must determine the "best" values of λ_0, Γ, and I_0. This determination then allows scientists to identify the atom emitting the light as well as to infer its environment (e.g., the pressure of the gas).

Figure 2 displays experimental data for ambient light obtained by an optical multichannel analyzer. The dashed curve in Figure 2 shows the least-squares curve fit to this data by (1). However, observe that these data do not taper off to zero as predicted by the Lorentz lineshape.

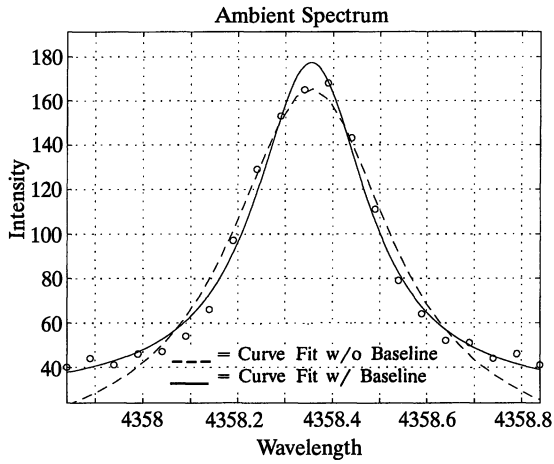

Figure 2. The Importance of Baselines

The source of this difficulty is that all experimental measurements are inescapably subject to such physical constraints as the Uncertainty Principle. This introduces a baseline into the problem, which must be explicitly accounted for in the theoretical model. We must replace (1) by

$$I_B(\lambda) = \frac{I_0}{1 + 4(\lambda - \lambda_0)^2/\Gamma^2} + B \tag{2}$$

and must fit the additional baseline parameter, B, along with all the other parameters in the least-squares model.

The result is the solid curve shown in Figure 2. Including the baseline has a dramatic effect on the computed value of Γ. If this correction were not made then the atomic environment would be erroneously determined. A reasonable embarkation point for further study of model verification/modification is provided by [**10**, p. 19].

2.3 Deadly Sin #3: Temptation From Ill-Conditioning. Engineers typically reject a mechanical or electrical device as useless (and dub it unstable) if a small relative change in its input produces a large relative change in its output. The corresponding numerical concept is that of *ill-conditioning*. We illustrate how this phenomenon arises in the context of root finding.

One of the most vexing problems in numerical computation lies in assessing the accuracy of an approximate solution, $\hat{x}$, to a nonlinear equation, $f(x_*) = 0$, obtained by fixed point iteration. The numerical neophyte is tempted to base an assessment solely on how closely the numerical solution comes to satisfying the equation, i.e., by the magnitude of the residual, $f(\hat{x})$. However, an inspection of Figure 3 reveals the fallaciousness of this reasoning.

The function graphed there is

$$y = f(x) = 10e^{5(x-2)}(x - 1)(x - 2) \tag{3}$$

with roots at $x = 1, 2$. However, these two roots have very different complexions. The function is so shallow at $x = 1$ ($y'(1) = -10e^{-5} \approx -.067$) that the residual may be small while $\hat{x}$ is still far from x_*. In this case, the naive logic would tempt us to terminate the iterative process prematurely. On the other hand, the function

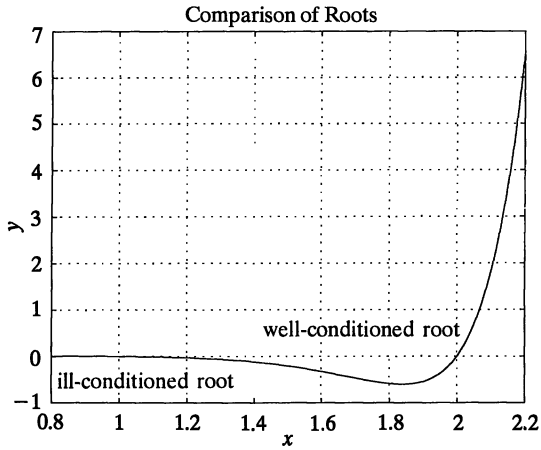

Figure 3. Differently Conditioned Roots

is so steep at $x = 2$ ($y'(2) = 10$) that the residual might be large while $\hat{x}$ is in fact close to x_*. In this event, more iterations than are necessary would be performed. This is clearly the lesser of these two evils. Obviously, we can make these disparities in slope arbitrarily large by a suitable choice of $f(x)$.

These considerations may be given a quantitative formulation as follows [**11**, p. 85]. If $\hat{f}(x)$ is our approximation to $f(x)$ induced by roundoff errors during function evaluation and δ measures the accuracy to which we can evaluate $f(\hat{x})$, i.e., $|\hat{f}(\hat{x}) - f(\hat{x})| \leq \delta$, then $|f'(x)| \geq M$ for all x close to x_* (assumed to be a simple root) implies the method-independent error estimate

$$|\hat{x} - x_*| \leq \frac{|\hat{f}(\hat{x})| + \delta}{M}. \tag{4}$$

Since the best we can achieve in practice is to produce an $\hat{x}$ satisfying $\hat{f}(\hat{x}) = 0$, the limit to the accuracy we can attain is

$$|\hat{x} - x_*| \leq \frac{\delta}{M}. \tag{5}$$

Returning now to Figure 3, if we assume that both function values can be computed to the same accuracy then we dub the root $x = 1$ ill-conditioned because of its small derivative value; the root $x = 2$ is well-conditioned due to its large derivative value. The long and the short of this analysis is that reliance on the magnitude of the residual in assessing accuracy must be tempered by a consideration of the precision to which we may perform function evaluations and also of the slope of the function at the root. Of course, this problem is only exacerbated when we pass to systems of equations, whether linear or nonlinear. Comprehensive treatments of conditioning and rounding (the subject of the next section) appear in [**12**, p. 9].

2.4 Deadly Sin #4: Ruination By Rounding. Interacting with a digital computer can be an unsettling affair. Despite all our school years spent studying the real number continuum, we are rudely awakened to the fact that we must, in practice, content ourselves with some finite set of numbers. This consequence of finite-precision arithmetic introduces roundoff error into numerical computation (even upon

input!). Let us explore, via example, how this affects traditional mathematical arguments based upon properties of the real number system.

All of the commonly used numerical differentiation rules possess error estimates of the form

$$D(f) = f'(a) = D_h + ch^r + o(h^r), \tag{6}$$

where D_h is our finite difference approximation, c is the asymptotic error constant, h is the mesh width, and r is the order of approximation [**13**, p. 300]. However, such estimates assume infinite precision arithmetic, which is not available on a fixed-word-length computer.

The portion of the error accounted for in (6) is called the *truncation* or *discretization error*. The additional error due to finite word length is referred to as the *roundoff error*. The difficulty lies in the fact that, while the truncation error goes to zero with h, the roundoff error becomes unbounded. Consequently, error terms such as that appearing in (6) present the illusion that unlimited accuracy is achievable simply by refining the mesh.

For the sake of definiteness, consider the central difference formula

$$D_h = \frac{f(a+h) - f(a-h)}{2h}, \tag{7}$$

with $c = -f'''(a)/6$ and $r = 2$. If we account only for the roundoff error, $E_{\pm}$, incurred by function evaluations then we obtain a computed value of

$$f_c' = \frac{f(a+h) + E_+ - f(a-h) - E_-}{2h} = D_h + \frac{E_+ - E_-}{2h}$$

$$= D(f) + \frac{E_+ - E_-}{2h} + ch^2 + o(h^2). \tag{8}$$

Thus, we determine that the actual error is composed of two quite dissimilar pieces: a parabolic truncation error and a hyperbolic roundoff error. As seen in Figure 4, this implies the existence of an optimum mesh width, h_{opt}, below which the total error increases. The perplexing facet of this phenomenon is that the precise determination of h_{opt} is impossible in practice.

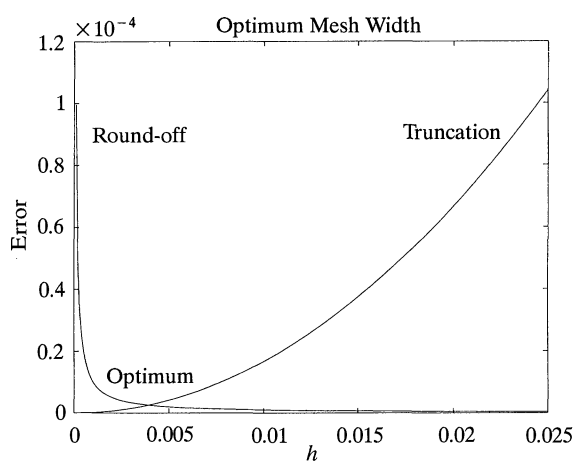

Figure 4. The Trade-Off in Mesh Refinement

Fortunately, we may obtain highly accurate derivative approximations without excessively refining the mesh! This miracle is achieved by that previously alluded to "Great Idea" in 20th century numerical computation: Richardson extrapolation. Rather than attempt a general explication of this remarkable device [**13**, p. 333], we instead particularize to the special case of the central difference approximation to the first derivative.

We begin by computing a coarse approximation

$$D(f) = D_{2h} + 4ch^2 + o(h^2), \tag{9}$$

and a fine approximation

$$D(f) = D_h + ch^2 + o(h^2). \tag{10}$$

We then subtract (9) from (10) and solve for the dominant term in the error

$$ch^2 = \frac{D_h - D_{2h}}{3} + o(h^2). \tag{11}$$

Finally, we insert (11) into (10) and obtain

$$D(f) = D'_h + o(h^2); \quad D'_h = \frac{4D_h - D_{2h}}{3}. \tag{12}$$

D'_h is the *Richardson extrapolant*, which constitutes a refined approximation so long as

$$\frac{D_h - D_{2h}}{D_{h/2} - D_h} \approx 4. \tag{13}$$

Of course, (12) and (13) are themselves subject to roundoff error, but, since we would presumably be using a larger value for h than for (7), the effects would be less pronounced.

This process may be repeated to yield successively more refined approximations. A similar device is applicable to higher order derivatives and integrals, as well as to approximating differential equations, where vast savings in computer resources may be so reaped. See [**14**, p. 38] for an extensive treatment including the related Shanks transformation.

2.5 Deadly Sin #5: Numerical Philistinism. The surest symptom of computational amateurism is the tendency to rely solely upon brute force numerical power rather than on the prudent use of a hybrid analytical/numerical formulation. Typically, this results in an immense waste of computational resources and a concomitant reduction in numerical resolution. Moreover, for many problems, a frontal assault would lead to an intractable computational problem. We illustrate this affliction by considering an application involving an integral with a highly oscillatory integrand.

In the scattering of acoustic or electromagnetic waves by a cylindrical obstacle [**15**, p. 193], we require the "scattering cross section" (see Figure 5 for notation)

$$\chi(\theta) = \lim_{r\to\infty} \frac{|u_S|^2}{|u_I|^2}, \tag{14}$$

where u_I and u_S are the incident and scattered fields, respectively. Physically, this is the scattered power per unit length in the direction θ normalized by that of the incident field.

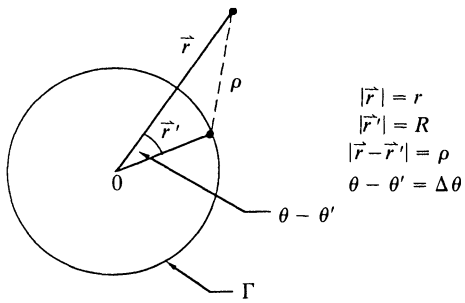

Figure 5. Far-Field Computation

Using Huygens' Principle, we may reduce this to the computation of an integral around a circular contour surrounding the scatterer (see Figure 5):

$$\chi(\theta) = \frac{R^2}{4k} \cdot \left| \int_0^{2\pi} F\left[u_S(\vec{r}')\right] e^{jkR\cos\Delta\theta}\, d\theta' \right|^2;$$

$$F\left[u_S(\vec{r}')\right] = \left[\frac{\partial u_S(\vec{r}')}{\partial R} - (jk\cos\Delta\theta) u_S(\vec{r}') \right], \tag{15}$$

where k is the free-space wave number of the incident wave.

For large values of kR, the integrand in (15) is highly oscillatory. If we were to attempt to numerically integrate this directly using, say, the composite trapezoidal rule then the number of panels required to resolve these oscillations would be so immense as to be prohibitive.

Instead, we first make the approximation

$$\int_0^{2\pi} \left[\frac{\partial u_S(\vec{r}')}{\partial R} - (jk\cos\Delta\theta) u_S(\vec{r}') \right] e^{jkR\cos\Delta\theta}\, d\theta'$$

$$\approx \sum_i \left\{ \frac{\partial \overline{u_S^i}}{\partial R} \int_{\theta_i'}^{\theta_{i+1}'} \Psi(R, \Delta\theta)\, d\theta' - jk\overline{u_S^i} \int_{\theta_i'}^{\theta_{i+1}'} \cos\Delta\theta \cdot \Psi(R, \Delta\theta)\, d\theta' \right\}, \tag{16}$$

with $\Psi(R, \Delta\theta) = e^{jkR\cos\Delta\theta}$, where $\overline{u_S^i}$ is the average of u_S over the ith panel. A linear approximation is then made to $\cos\Delta\theta$ on each panel. The remaining integrals are then evaluated analytically. The resulting approximate expression for $\chi(\theta)$ assumes the form of a discrete convolution and hence may be very evaluated efficiently using FFT techniques.

What we have done here is to employ strategically the trapezoidal rule's linear approximation in such a way as to render explicitly evaluable the resulting integral over a panel, thus analytically resolving the oscillations there. For high-frequency incident waves, this results in manifold savings in arithmetic operations over the brute force approach. This is especially important since $\chi(\theta)$ is required for many observation angles, θ.

Note that we are dealing here not with analytical approximations to the physical model prior to numerical solution but, rather, with the formulation of numerical schemes that minimize approximations and maximize analytical evaluation. A wealth of such ideas for numerical quadrature is contained in [**16**, p. 345].

2.6 Deadly Sin #6: Central Differencing Singular Perturbations. It hardly seems necessary to point out that a great number of physical processes are governed by second order ordinary differential equations. However, it is important to observe that in many such applied problems it is the first derivative terms that dominate the process under scrutiny. These are singular perturbation problems and require caution in the extreme for their numerical solution. The sheer breadth of these problems is breathtaking [**17**, p. 1]: pollutant dispersal in rivers, vorticity transport in aerodynamics, atmospheric pollution, kinetic theory, semiconductor devices, groundwater transport, financial modelling, melting,... .

A representative model problem is provided by the steady-state, convection-diffusion equation in one dimension

$$-Du'' + vu' = 0; \quad u(0) = 0, u(1) = 1, \tag{17}$$

where D, the diffusion coefficient, and v, the convective velocity, are given positive constants.

The exact solution to this problem is

$$u(x) = \frac{e^{Pe \cdot x} - 1}{e^{Pe} - 1}, \tag{18}$$

where $Pe = v/D$ is the *Peclet number* and measures the strength of convection relative to diffusion [**18**, p. 24]. In fluid dynamics, the analogous quantity is called the *Reynolds number* and measures the strength of inertial force relative to viscous force. In any case, we are concerned here with $Pe \gg 1$ (convection-dominated flow), in which case a boundary layer arises on the right, $x = 1$ (see Figure 6 where $Pe = 50$).

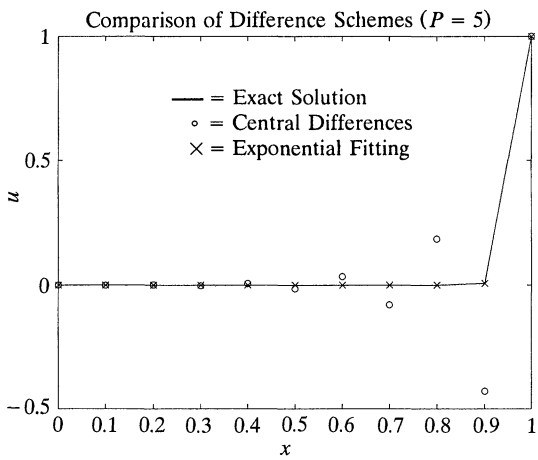

Figure 6. Resolving a Boundary Layer

If we discretize (17) using central differences with mesh width $\Delta x = 1/N$, we arrive at the discrete equation

$$(2 + P)u_{i-1} - 4u_i + (2 - P)u_{i+1} = 0, \tag{19}$$

where $P = Pe \cdot \Delta x = v\Delta x/D$ is the *cell Peclet number* (a.k.a. *cell Reynolds number*). The exact solution to this difference equation satisfying the boundary conditions $u_1 = 0$ and $u_{N+1} = 1$ is

$$u_i = \frac{q^{i-1} - 1}{q^N - 1} \quad (i = 1, \ldots, N + 1); \quad q = \frac{2 + P}{2 - P}. \tag{20}$$

This approximate solution for $P = 5$ is plotted in Figure 6 and displays prominent oscillations as we approach the boundary layer. These oscillations may be suppressed by using upwind differences, but at a prohibitive cost. The accuracy would then drop from second order to only first order.

An elegant way out of this dilemma is to employ exponential fitting [**19**, p. 18]. In this technique, the discrete equation (19) is replaced by

$$B(-P)u_{i-1} - [B(P) + B(-P)]u_i + B(P)u_{i+1} = 0, \tag{21}$$

where $B(z) = z/(e^z - 1)$ is the Bernoulli function. The exact solution to (21) with identical boundary conditions is

$$u_i = \frac{q^{i-1} - 1}{q^N - 1} \quad (i = 1, \ldots, N+1); \quad q = \frac{B(-P)}{B(P)}. \tag{22}$$

This method automatically provides upwinding as $P \to \infty$ and is second order accurate uniformly in $\epsilon = 1/P$. That is, the coefficient of $(\Delta x)^2$ in the error bound is independent of ϵ. Figure 6 shows this exponentially fitted approximation. Notice the conspicuous absence of any oscillatory behavior.

This example shows that if we use exponential fitting rather than central differences then we achieve optimal results in both limiting cases: as $P \to 0$ we recover central differences and as $P \to \infty$ we obtain correct directional dependence without having to sacrifice accuracy. Thus, exponential fitting plays a role in singular perturbations that is directly analogous to that played by exponential splines in our treatment of Deadly Sin #1. Naturally, similar considerations apply to singularly perturbed partial differential equations [**20**, p. 43].

2.7 DEADLY SIN #7: UNCRITICAL USE OF NON-ORTHOGONAL MAPPINGS. This last Deadly Sin is perhaps the most serious, and is all-too-commonly committed. In fact, it is explicitly advocated in some quarters [**21**, p. 33]. It concerns approximating a partial differential equation as follows: first perform a not-necessarily-orthogonal mapping of the solution domain to a rectangle or other canonical domain, then transform the partial differential equation to this coordinate system, and finally discretize this transformed equation by straightforward finite differences. We now argue that, unless the underlying mapping is nearly orthogonal, this procedure is inappropriate and results in several highly undesirable consequences.

We take as our prototype the Dirichlet problem for Poisson's equation on the rhombus displayed in Figure 7:

$$-\Delta u = f \text{ in } D; \; u = g \text{ on } \partial D. \tag{23}$$

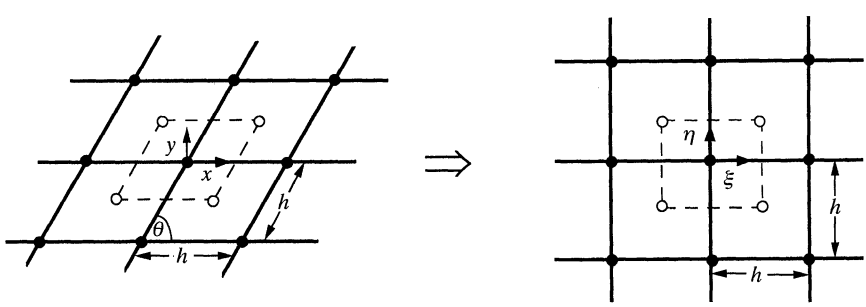

Figure 7. Shearing Transformation

While this problem may at first seem rather specialized, it is in fact prototypical of many physical processes such as heat conduction, fluid flow, and wave propagation, which require approximations to the Laplacian in complex geometry. According to the procedure recommended in [**21**, p. 34], we map this domain to the square, S, also shown in Figure 7, via a bilinear transformation, which in this instance reduces to the shearing transformation

$$\xi = x - (\cot\theta)y; \quad \eta = (\csc\theta)y. \tag{24}$$

In these sheared coordinates, the Poisson equation becomes

$$-\left[(\csc\theta)u_{\xi\xi} - 2(\cot\theta)u_{\xi\eta} + (\csc\theta)u_{\eta\eta}\right] = (\sin\theta)f. \tag{25}$$

Discretization within the square employing central differences yields the discrete equation

$$\csc\theta\cdot\left(-u_{i+1,j} - u_{i-1,j} - u_{i,j+1} - u_{i,j-1} + 4u_{i,j}\right) - \frac{1}{2}\cdot\cot\theta\cdot\left(-u_{i+1,j+1} + u_{i-1,j+1} + u_{i+1,j-1} - u_{i-1,j-1}\right) = (h^2\sin\theta)f_{i,j} \tag{26}$$

at each interior point of S. These equations are supplemented by the given Dirichlet conditions along ∂S.

The computational stencil for this scheme is displayed in Figure 8. This discrete operator is not of positive type nor is it diagonally dominant [**22**, p. 181]. Moreover, as we shall see, it unnecessarily expands the stencil to 9 points. The problem is not that we have transformed the equation but rather that the transformation employed is highly non-orthogonal.

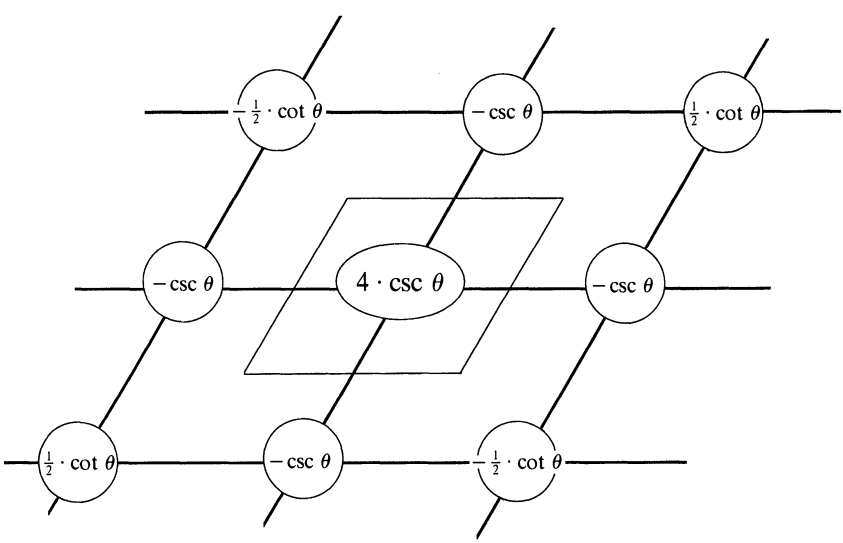

Figure 8. Stencil for Shearing Transformation

This may most easily be seen by remaining in the physical domain and discretizing the conservation form of the Poisson equation

$$-\oint_{\partial R}\frac{\partial u}{\partial n}\,ds = \int_R\int f\,dA \tag{27}$$

over the rhombus shown dashed in Figure 7. This process again leads to (26). The

source of difference weights of inappropriate sign may now be traced to the non-orthogonal alignment of the "grid lines" with the sides of this rhombic "control region".

This difficulty may be remedied by employing the Control Region Approximation [**23**],[**24**, p. 142]. We now use as our control region the Dirichlet polygon surrounding a generic grid point shown dashed in Figure 9, and discretize (27). This yields the discrete interior equation

$$(\csc\theta - \cot\theta)\cdot(-u_{i+1,j} - u_{i-1,j} - u_{i,j+1} - u_{i,j-1} + 4u_{i,j}) + \cot\theta\cdot(-u_{i-1,j+1} + 2u_{i,j} - u_{i+1,j-1}) = (h^2\sin\theta)f_{i,j}. \qquad (28)$$

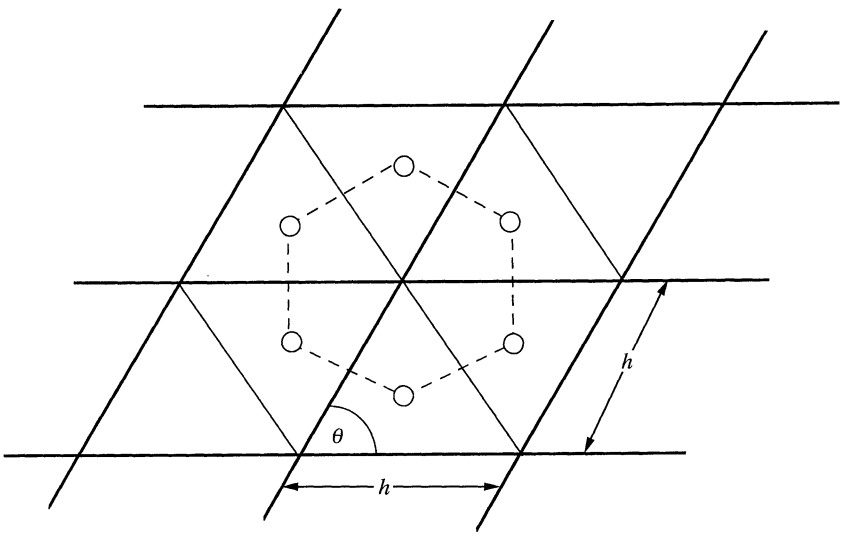

Figure 9. Control Region Approximation

Inspection of the 7-point computational stencil for this scheme displayed in Figure 10 reveals that the orthogonal alignment of grid lines and control region

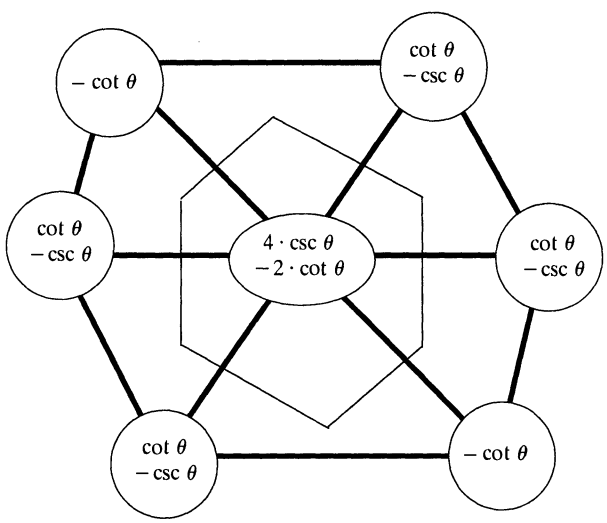

Figure 10. Stencil for Control Region Approximation

sides yields a discrete operator of positive type. Moreover, this discrete operator generates a corresponding irreducibly diagonally dominant, symmetric positive definite M-matrix [**25**, p. 202]. Thus, a discrete maximum principle, analogous to that for the continuous Laplacian, holds. This greatly facilitates error estimation.

The Control Region Approximation yields difference equations whose solution satisfies a discrete conservation law. In fact, the relationship between these two discretizations is vastly illuminated by consideration of $d =$ the difference of the left hand sides of (26) and (28):

$$d = \underbrace{(h^2 \sin\theta)}_{\text{area}} \cdot \left\{ \underbrace{\frac{h^2}{2} \cdot (\cos\theta) \cdot \left[(\cot^2\theta) \cdot u_{xxxx} + 2(\cot\theta) \cdot u_{xxxy} + u_{xxyy} \right]}_{\text{spurious source}} + o(h^2) \right\}. \tag{29}$$

Consequently, the shearing transformation introduces an additional $O(h^2)$, non-physical, spurious source term that becomes unbounded as $\theta \to 0$ with h fixed.

The moral of this story is clear: Unless one is prepared to make an *orthogonal* coordinate transformation, or one that is nearly so, one should remain in the physical domain and use the Control Region Approximation or some other physical-domain approximation [**26**, p. 17]. The problem that we have identified and remedied is symptomatic of the perils that lie dormant in any attempt to model the physical world mathematically: There are many seemingly plausible mathematical universes but only a single physical one! The Applied Mathematician must be forever vigilant against becoming intoxicated with the inherent beauty of the mathematical model and must instead remain firmly tethered to the underlying physics of the problem at hand.

3. CONCLUSION. While our list of numerical transgressions is hardly exhaustive and a reasonable argument could doubtless be made for replacing some of them by the reader's pet peeves, there is little doubt that abstinence from these indulgences will positively benefit one's numerical soul. A recurring theme throughout this sermon has been the need to be perpetually cognizant of the physical basis for any numerical computation. In fact, this might profitably be adopted as the First Commandment of Numerical Computation. Of course, this requires an in-depth study of the physical foundations of a mathematical problem—a sojourn upon which many Mathematicians seem reluctant to embark.

ACKNOWLEDGMENTS. The author thanks Pat Irish for locating and bringing to his attention [**1**] and Barbara Rowe for her assistance in the production of this paper. He is also eternally grateful to Dr. Joseph R. Caspar for impressing upon him early in his career that the Seventh Deadly Sin is indeed a mortal one.

REFERENCES

1. S. Schimmel, *The Seven Deadly Sins*, The Free Press, New York, 1992.
2. I. A. Stegun and M. Abramowitz, Pitfalls in Computation, *J. Soc. Indust. Appl. Math.* 4 (1956), 207–219.
3. G. E. Forsythe, Pitfalls in Computation, or Why a Math Book Isn't Enough, *Amer. Math. Monthly* 77 (1970), 931–956.
4. I. J. Schoenberg, Contribution to the Problem of Approximation of Equidistant Data by Analytic Functions, *Quart. Appl. Math.* 4 (1946), Part A: 45–99; Part B: 112–141.
5. D. G. Schweikert, An Interpolation Curve Using a Spline in Tension, *J. Math. Phys.* 45 (1966), 312–317.
6. S. Pruess, Properties of Splines in Tension, *J. Approx. Theory* 17 (1976), 86–96.

7. B. J. McCartin, Computation of Exponential Splines, *SIAM J. Sci. Stat. Comput.* 11 (1990), 242–262.
8. H. Späth, *Spline Algorithms for Curves and Surfaces*, Utilitas Mathematica, Winnipeg, 1974.
9. P. L. DeVries, *A First Course in Computational Physics*, Wiley, New York, 1994.
10. N. Bellomo and L. Preziosi, *Modelling Mathematical Methods and Scientific Computation*, CRC Press, Boca Raton, 1995.
11. L. Eldén and L. Wittmeyer-Koch, *Numerical Analysis*: *An Introduction*, Academic Press, New York, 1990.
12. N. J. Higham, *Accuracy and Stability of Numerical Algorithms*, SIAM, Philadelphia, 1996.
13. S. Conte and C. de Boor, *Elementary Numerical Analysis*: *An Algorithmic Approach*, 3rd Edition, McGraw-Hill, New York, 1980.
14. G. I. Marchuk and V. V. Shaidurov, *Difference Methods and Their Extrapolations*, Springer-Verlag, New York, 1983.
15. B. J. McCartin, L. J. Bahrmasel, and G. Meltz, "Application of the Control Region Approximation to Two-Dimensional Electromagnetic Scattering", Chapter 5 of *Differential Methods in Electromagnetic Scattering*, M. A. Morgan (Ed.), Elsevier, New York, 1988.
16. M. Abramowitz, "On the Practical Evaluation of Integrals", Appendix 1 of *Methods of Numerical Integration* by P. J. Davis and P. Rabinowitz, Academic Press, New York, 1975.
17. K. W. Morton, *Numerical Solution of Convection Diffusion Problems*, Chapman & Hall, London, 1996.
18. R. Peyret and T. D. Taylor, *Computational Methods for Fluid Flow*, Springer-Verlag, New York, 1983.
19. J. J. H. Miller, E. O'Riordan, and G. I. Shishkin, *Fitted Numerical Methods for Singular Perturbation Problems*, World Scientific, Singapore, 1996.
20. H.-G. Roos, M. Stynes, and L. Tobiska, *Numerical Methods for Singularly Perturbed Differential Equations*, Springer-Verlag, New York, 1996.
21. P. Knupp and S. Steinberg, *Fundamentals of Grid Generation*, CRC Press, Boca Raton, 1993.
22. G. E. Forsythe and W. R. Wasow, *Finite-Difference Methods for Partial Differential Equations*, Wiley, New York, 1960.
23. J. R. Caspar, D. E. Hobbs, and R. L. Davis, Calculation of Two-Dimensional Potential Cascade Flow Using Finite Area Methods, *AIAA J.* 19 (1980), 103–109.
24. B. J. McCartin, "Control Region Approximation for Electromagnetic Scattering Computations", *Computational Wave Propagation*, B. Engquist and G. A. Kriegsmann (Eds.), IMA Volumes in Mathematics and Its Applications, Vol. 48, Springer-Verlag, New York, 1997, pp. 141–164.
25. O. Axelsson, *Iterative Solution Methods*, Cambridge University Press, New York, 1996.
26. B. Heinrich, *Finite Difference Methods on Irregular Networks*, Birkhäuser Verlag, Boston, 1987.

BRIAN J. MCCARTIN graduated with Highest Distinction in Applied Mathematics from the University of Rhode Island and Summa Cum Laude in Music Theory from the Hartt School of Music. Also, he holds a doctorate in Applied Mathematics from the Courant Institute of Mathematical Sciences of New York University. He was formerly Head of the Computational Electromagnetics Program at United Technologies Corporation and Chairperson of the Department of Computer Science at Rensselaer Polytechnic Institute's Hartford, Connecticut campus.
Applied Mathematics, Kettering University, Flint, Michigan 48504-4898
bmccarti@kettering.edu