

Benchmark Problems for Simulating Electric Fields Near Sharp Corners

Doug Craigen, PhD

Integrated Engineering Software / Enginia Research
Winnipeg, Manitoba, Canada R3H 0G4

INDEX

[The Problem](#)

[Jackson's Solution](#)

[Case 1: Razor Edge](#)

[Case 2: 90 Degree Corner](#)

[Summary](#)

Integrated Engineering Software - Website Links

[Home](#)

[Products](#)

[Support](#)

[Technical Papers](#)

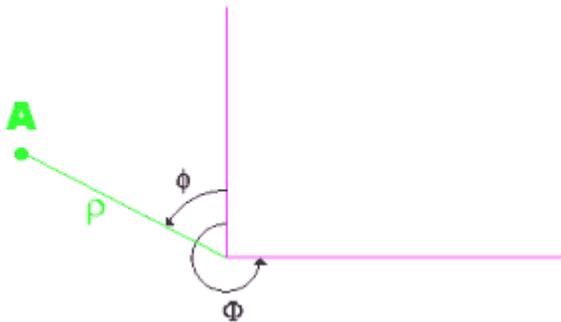
"Page Down" or use scroll bars to read the article

Benchmark Problems for Simulating Electric Fields Near Sharp Corners

Doug Craigen, PhD
Integrated Engineering Software/Enginia Research
Winnipeg, Manitoba, Canada R3H 0G4

INDEX

- [The Problem](#)
- [Jackson's Solution](#)
- [Case 1: Razor Edge](#)
- [Case 2: 90° Corner](#)
- [Summary](#)



The Problem

One of the most common questions in Electromagnetic simulation is "what/where is the maximum field?". When any kind of corner or edge is present the question becomes tricky. A corner is mathematically a singularity. You cannot ask what is the direction of the field at the corner - that depends on the direction from which you approach it. Furthermore, in cases like that to the left, the magnitude of the field is infinite at the corner. Hence, examining the field near a corner should produce increasingly high fields. The question posed is not meaningful because it depends on how close you approach the corner.

Of course, no physical object has a perfect corner - there is some degree of roundness. Also, at the locations where the fields become extremely high, other phenomena will come into play to limit the field. So, while the field will have a maximum value, the reasons for that value are not trivial. Users of Electromagnetic CAE tools are often lulled into a false sense of security by answers that look physically correct. However, suppose you draw the corner as sharp - the correct answer for the model as you constructed it is a singular field at the corner. If your CAE tools do not predict this, they are giving the wrong answer to the question you posed. This may or may not be the correct answer to the real physical problem, but if it is correct it was by luck rather than because it was able to solve the correct physical problem. If you set the solver to a more accurate convergence, or refine the mesh around the corner, you should get a higher "maximum field". This is a Russian Roulette approach to simulation.

At ENGINIA we've always contended that the correct approach is to insist the tool gives the correct answer to the question as posed. This provides the user with a tool they can use in an intelligent fashion rather than by relying on luck. The key is then simply to pose the question to reflect the actual physical problem realistically. Our first level of benchmarking is thus some analytic cases where uncertainty in measurement and modeling of the test case is not an issue. If you are using any of our competitors' CAE tools, you should be able to construct the models below in a matter of seconds for comparison.

Jackson's Solution

"Classical Electrodynamics, 2nd Edition" by J.D. Jackson addresses the generic problem illustrated above in section 2.11. He shows that close enough to a corner the voltage takes the form:

$$\Phi(\rho, \phi) = V + \sum_{m=1}^{\infty} a_m \rho^{m\pi/\beta} \sin(m\pi\phi/\beta)$$

Using this form we can create problems in which the a_m 's can be determined to have an analytically defined corner problem. An example is given in figure 2:

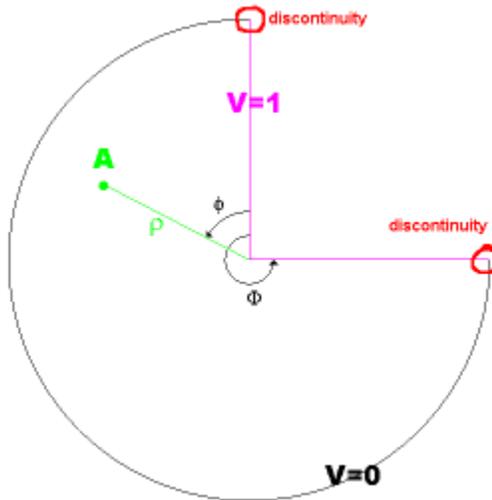


FIGURE 2: an analytically solvable problem

In this problem one cannot, of course, have the 1 V and 0 V sections touch. However, treating them as coming infinitesimally close presents no problem to the math. In any model that is constructed a small discontinuity will be needed at these locations.

Setting the radius of the arc to 1 m, and considering P anywhere on that arc:

$$0 = 1 + \sum_{m=1}^{\infty} a_m \sin(m\pi\phi/\beta)$$

this is a basic Fourier series problem with a well known solution:

$$a_m = \frac{4}{m\pi} \text{ for odd } m$$

So the field components are:

$$E_\rho = -\frac{\partial\Phi}{\partial\rho} = \frac{4}{\beta} \sum_{\text{odd } m} \rho^{(\frac{m\pi}{\beta}-1)} \sin(m\pi\phi/\beta)$$

and

$$E_\phi = -\frac{1}{\rho} \frac{\partial\Phi}{\partial\phi} = \frac{4}{\pi} \sum_{\text{odd } m} \frac{1}{m} \rho^{(\frac{m\pi}{\beta}-1)} \cos(m\pi\phi/\beta)$$

Case #1: "Razor Edge" ($\beta = 2\pi$)

This is the most extreme case possible for very sharp edges, thin foils etc. The chosen model is shown in Figure 3:

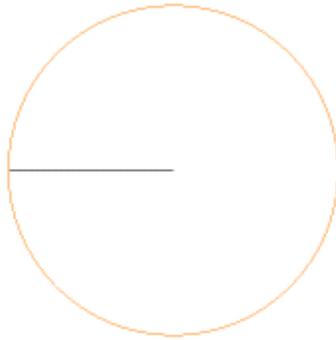


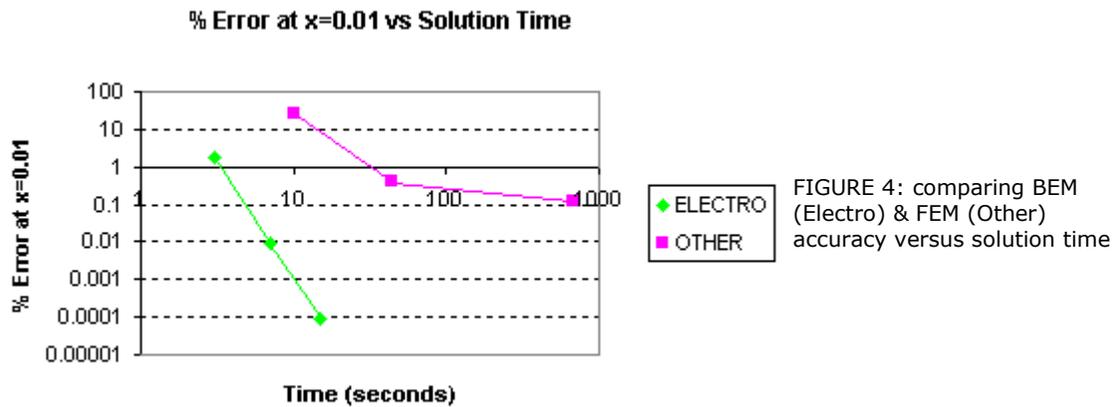
FIGURE 3: the Razor Edge model

The model simply consists of a 1 m radius circle set at 0 V and a line segment extending from 0, 0 to -0.99, 0 and set at 1 V. This model will be analyzed in our 2D electric simulation package ELECTRO.

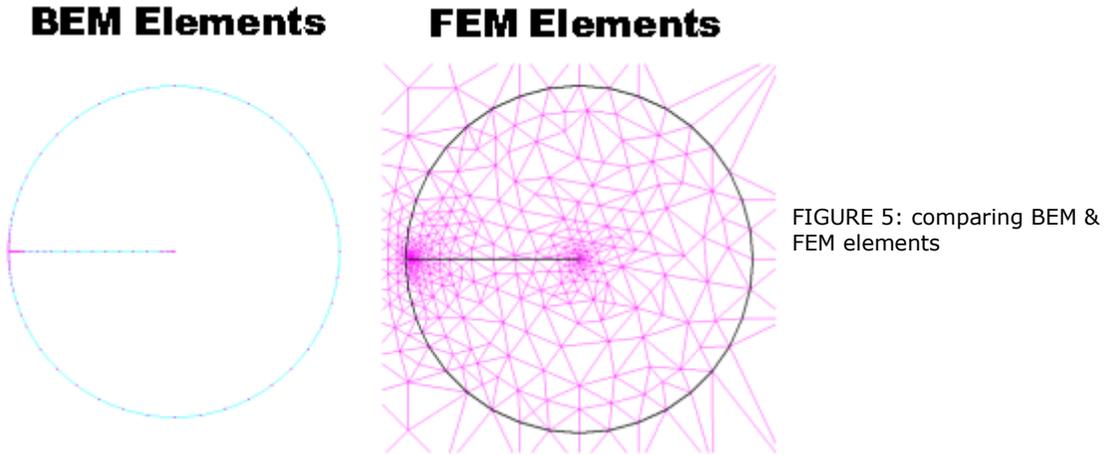
Prior to version 6.1 Electro users only had a "Boundary Element Method" (BEM) solver. We still recommend using this solver for most electric simulation problems, and especially ones where small features are present. To illustrate while avoiding any wrong impressions about the quality of our FEM solver, we compare ELECTRO with one of our commercial competitors who have been using the "Finite Element Method" (FEM) for many years.

For simplicity in interpreting the results, a series of standard sampling points are given along the positive x axis. Here the y component of E is zero.

As a quick test, the value is probed at $x=0.01$ - meaning we want the field at a distance from the corner of about 1% of the size of the piece. Clearly the longer we give the solver the better the solution can be.



Using the "Boundary Element Method" enables ELECTRO to find better solutions faster. The reason is obvious if one examines the mesh used for calculation by each method:



BEM elements are purely surface elements in this problem, used to model the charge distribution which results in the applied voltages. The FEM elements are areas used to compute the local voltage. Not only is it easier to get a mesh including very small detail with BEM, but the electric field is computed by integration from the charge, which incurs less numerical error than differentiating the voltage.

The following three tables examine the difference between the two approaches in more detail:

Default Solutions (Electro: 3 seconds, Other: 10 seconds) % error in ()'s				
x	Correct E	ELECTRO	OTHER	Notice that: <ul style="list-style-type: none"> ▪ ELECTRO (BEM) is 3 times faster ▪ ELECTRO produces more accurate numbers in every case ▪ even at x=0.8, a non-challenging location, ELECTRO is about 7 times more accurate <p>Based on years of comparing numeric solvers, we expect to see this same general trend for <i>any</i> FEM code compared to BEM on this problem.</p>
0.8	0.395423635	0.39512 (0.078%)	0.3975 (0.53%)	
0.1	1.83015317	1.8262 (0.22%)	1.861 (1.7%)	
0.01	6.303166063	6.185 (1.9%)	8.048 (28%)	
0.001	20.11157327	16.94 (16%)	12.66 (37%)	
0.0001	63.65561168	33.55 (47%)	12.94 (80%)	
0.00001	201.3148353	94.09 (53%)	12.96 (94%)	
0.000001	636.6191357	94.09 (85%)	12.97 (98%)	

Both programs can easily be set to solve more accurately, this was just what you get by using the default settings. Below are the answers obtained with a higher accuracy setting for each program:

Modified Solver Setup (Electro: 7 seconds, Other: 44 seconds) % error in ()'s				
x	Correct E	ELECTRO	OTHER	Notice that:
0.8	0.395423635	0.3954183	0.39557	

		(0.0014%)	(0.037%)	<ul style="list-style-type: none"> both programs now produce better answers the ELECTRO solution is 6 times faster than the "Other" solution ELECTRO now produces better than 1% accuracy down to 1/10,000th of the geometry scale
0.1	1.83015317	1.830127 (0.0014%)	1.83046 (0.017%)	
0.01	6.303166063	6.302590 (0.0091%)	6.32766 (0.39%)	
0.001	20.11157327	20.09547 (0.080%)	20.372 (1.3%)	
0.0001	63.65561168	63.13316 (0.82%)	82.51 (30%)	
0.00001	201.3148353	184.0527 (8.6%)	101.66 (50%)	
0.000001	636.6191357	405.6363 (36%)	102.97 (84%)	

Modified Solver Setup (Electro: 15 seconds, Other: 686 seconds) % error in ()'s				
x	Correct E	ELECTRO	OTHER	Notice that:
0.8	0.395423635	0.3954213 (0.00059%)	3.95455 (0.0079%)	<ul style="list-style-type: none"> ELECTRO has again improved significantly for all locations "Other" only made significant improvement for 0.8, 0.1 and 0.01 with ELECTRO in 15 seconds the solution can be accurate to within 2-3% for scales within a millionth of the geometry size.
0.1	1.83015317	1.830155 (0.00010%)	1.83144 (0.070%)	
0.01	6.303166063	6.303172 (0.000094%)	6.29532 (0.12%)	
0.001	20.11157327	20.11122 (0.0018%)	19.6486 (2.3%)	
0.0001	63.65561168	63.64274 (0.020%)	55.6532 (12.6%)	
0.00001	201.3148353	200.88 (0.22%)	106.412 (47%)	
0.000001	636.6191357	621.25 (2.4%)	108.311 (83%)	

With both methods one can choose to refine the mesh locally in order to obtain a better answer in the region of interest. However, the advantages of BEM for this problem type will show up whether it is the program refining the mesh, or an educated user.

Case #2: 90° Corner ($\beta = \frac{3}{2}\pi$)

Figure 6 shows a suggested benchmark problem with a 90° corner. This is typical of many more problem types than case #1. The singularity is not as severe, but the problem is made challenging by also including the very thin gap between the pie piece at 1 V and the cylinder at 0 V. This is more like a real world problem in that there is a corner, but other geometry will also influence the solution. In fact, one can also use this as a benchmark for known solution for the E field in the gap. For a small gap this is a challenging problem too.

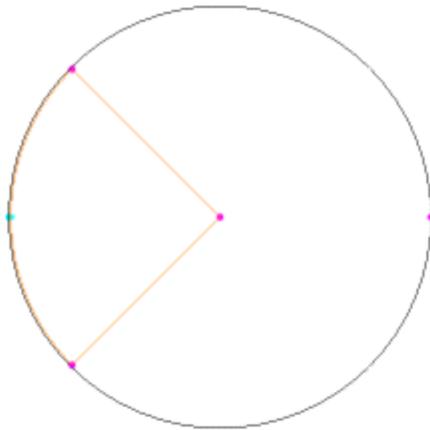


FIGURE 6: 90° Corner Problem

Below is a summary of the results, this time just focussing on a "long" calculation (where the word "long" has to be understood relative to which solver type is being used).

Modified Solver Setup (Electro: 20 seconds, Other: 323 seconds) % error in ()'s			
x	Correct E	ELECTRO	OTHER
0.8	0.52469989	0.524710 (0.0019%)	0.524786 (0.01%)
0.1	1.747623467	1.747621 (0.00014%)	1.7504 (0.15%)
0.01	3.931432953	3.931284 (0.0038%)	3.93625 (0.12%)
0.001	8.4874	8.4854 (0.023%)	8.567 (0.93%)
0.0001	18.287	18.202 (0.46%)	18.61 (1.8%)
0.00001	39.40	35.80 (9.1%)	19.74 (50%)
0.000001	84.88	57.10 (32.7%)	19.82 (77%)

Again BEM shows superior speed and accuracy. This can be double-checked by using the new FEM code within ELECTRO.

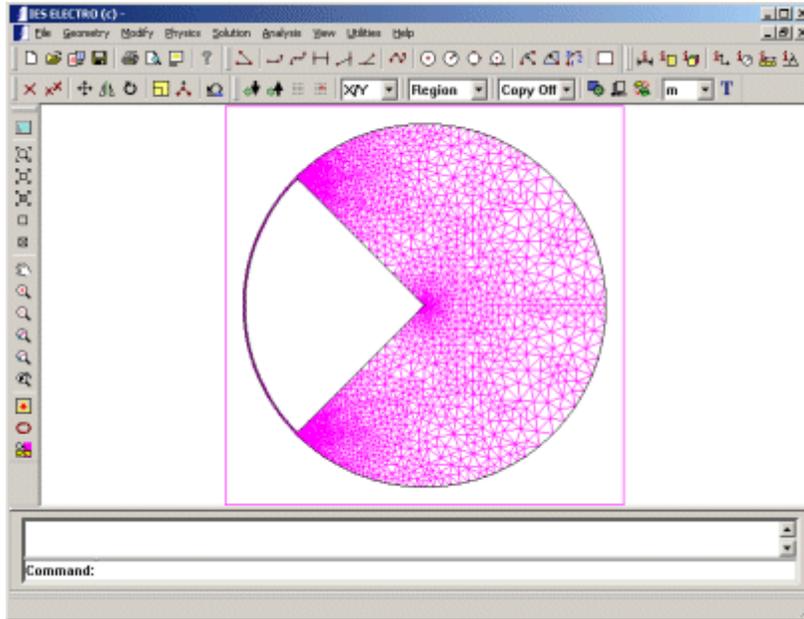


FIGURE 7: ELECTRO FEM Elements

x	Correct E	ELECTRO FEM Solution 90 seconds	Notice that:
0.8	0.52469989	0.5229 (0.34%)	<ul style="list-style-type: none"> ▪ ELECTRO FEM solution was 90 seconds compared to 323 for "Other" FEM program, the errors are higher in some locations and lower in others ▪ overall conclusions for FEM vs BEM for this problem type are unchanged
0.1	1.747623467	1.74528 (0.13%)	
0.01	3.931432953	3.821 (2.8%)	
0.001	8.4874	8.4891 (0.02%)	
0.0001	18.287	18.567 (1.5%)	
0.00001	39.40	34.89 (11%)	
0.000001	84.88	39.47 (53%)	

Summary

A simple to setup yet challenging to solve problem type has been proposed to benchmark the ability of CAE simulation software to give accurate field values in the vicinity of corners or sharp edges in a model. Two cases of this type illustrate why we believe the Boundary Element Method is the preferred solver type for this class of problems.

Users with other programs are invited to perform their own calculations for comparison and to take our free 30 day evaluation to verify for themselves the relative speed and accuracy. A user of the "Other" program in this study would have good reason to worry their results around small features are not sufficiently accurate to properly predict when fields will be high enough for arcing/flashover to occur.

The computer used in this study was running Windows 2000 at 2.4 GHz.