Report on Dipole Source Simulation

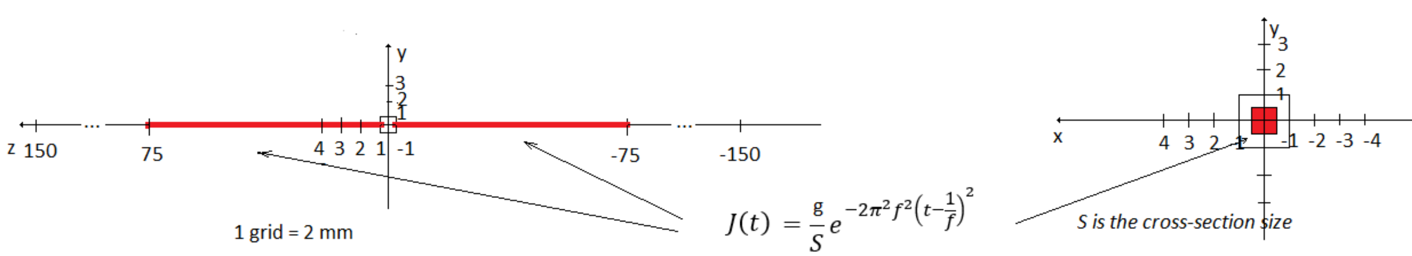
David Ge (dge893@gmail.com)

April 22, 2021

Contents

Simulation task description

Basic requirements



Computing domain: (0.3 m, 0.3 m, 0.6 m)

Space voxel size: 0.002 m

Gaussian Waveform current:

$$W(t) = e^{-2\pi^2 f^2 \left(t - \frac{1}{f}\right)^2}$$

Simulation time:

$$0.1\ \mu s = 10^{-7} seconds$$

The rods are perfect electric conductor.

PML thickness is 0.02 meter = 10 grids.

Source Code Implementing Dipole Source

The Gaussian Waveform is implemented in DipoleSourceTss.h.

Dipole field source is implemented in DipoleSourceTss.h and DipoleSourceTss.cpp

https://github.com/DavidGeUSA/TSS/blob/master/Source%20Code%20V2/TssTest/DipoleSourceTss.h

https://github.com/DavidGeUSA/TSS/blob/master/Source%20Code%20V2/TssTest/DipoleSourceTss.cpp

Create Task File for the simulation task description

A task file dipole.task is created for the simulation. The task file is at https://github.com/DavidGeUSA/TSS/blob/master/Source%20Code%20V2/taskFiles/dipole.task

"Space voxel size: 0.002 m" is described in the task file by

```
TSS.DS=0.002
TSS.DT=3.85e-12
```

The above settings specify $\Delta_s = 0.002, \Delta_t = 3.85 \times 10^{-12}$

"Simulation time: $10^{-7} seconds$" is described in the task file by

```
SIM.MAXTIMES=26000
```

We have $26000 \times 3.85 \times 10^{-12} \approx 10^{-7}$

"Computing domain: (0.3 m, 0.3 m, 0.6 m)" is described in the task file by

```
TSS.NX=150
TSS.NY=150
TSS.NZ=300
//ds*nx=0.3=x-axis length, xmin=-0.15=-(half x-axis length) -> x=0 is at the center
TSS.XMIN=-0.15
TSS.YMIN=-0.15
TSS.ZMIN=-0.3
```

PML setting:

```
PML.DISABLE=false
PML.THICK=10
PML.ALPHA=1
PML.BETA=1
PML.X.LOW=true
PML.X.HIGH=true
PML.Y.LOW=true
PML.Y.HIGH=true
PML.Z.LOW=true
PML.Z.HIGH=true
PML.POWER=3
```

Field Source Considerations

In the task file, we specify the field source to use the C++ class DipoleSourceTss:

```
TSS.SOURCE.CLASS=DipoleSourceTss
```

If the Gaussian waveform $W(t) = e^{-2\pi^2 f^2 \left(t-\frac{1}{f}\right)^2}$ represents current density then the Maxwell's equation is

$$case\ 1{:}\quad \frac{\partial E}{\partial t} = \frac{1}{\varepsilon}\nabla \times H - \frac{\sigma}{\varepsilon}E - \frac{1}{\varepsilon}e^{-2\pi^2 f^2 \left(t-\frac{1}{f}\right)^2}$$

If the Gaussian waveform $W(t) = e^{-2\pi^2 f^2 \left(t-\frac{1}{f}\right)^2}$ represents current then we need to divide it by the cross-section of the dipole rod to get the current density:

$$case\ 2{:}\quad \frac{\partial E}{\partial t} = \frac{1}{\varepsilon}\nabla \times H - \frac{\sigma}{\varepsilon}E - \frac{1}{\varepsilon}\frac{1}{\Delta_s^2}e^{-2\pi^2 f^2 \left(t-\frac{1}{f}\right)^2}$$

To accommodate both cases, we introduce a magnitude factor g to the Gaussian waveform:

$$W(t) = ge^{-2\pi^2 f^2 \left(t-\frac{1}{f}\right)^2}$$

The Maxwell's equation to be used becomes

$$\frac{\partial E}{\partial t} = \frac{1}{\varepsilon}\nabla \times H - \frac{\sigma}{\varepsilon}E - \frac{1}{\varepsilon}\frac{g}{\Delta_s^2}e^{-2\pi^2 f^2 \left(t-\frac{1}{f}\right)^2}$$

In the task file, the magnitude factor g is specified by SOURCE.MAGNITUDE

For case 1, because $\Delta_s^2 = 0.000004$, we use

```
SOURCE.MAGNITUDE=0.000004
```

For case 2, we use

```
SOURCE.MAGNITUDE=1
```

It also allows us to use other values for the magnitude.

We use GAUSSIAN.FREQUENCY to specify the value for the center frequency f :

```
GAUSSIAN.FREQUENCY=1e+9
```

Output

By default, at each simulation time step, two binary files are generated to record field values at every space node point, one file is for E field and the other for H field.

We are going to run 26000 time steps, we may not want to generate 26000x2=52000 files. Use TSS.REC_INTERVAL to reduce the number of files. For example,

```
TSS.REC_INTERVAL=100
```

Now it will generate field files every 100 time steps. Totally it will generate 26000x2/100=520 field files.

Run the simulation

Suppose the file dipole.task is in the folder taskFiles, we can run the simulation by following command

```
$./EM /W./tmp /D./_data /K300 /T./taskFiles/dipole.task
```

The following statistics is generated for SOURCE.MAGNITUDE=1 in dipole.task (only show the first 6 time steps):

```
Time	Energy	Max Divg E (||E||)	Avg Divg E (avg ||E||)Max Divg H (||H||)	Avg Divg H (avg ||H||)
0	2.61915e-025	0 (0)	0 (1.86385e-012)	0 (1.58101e-322)	0 (0)
100	5.54805e-014	4.73695e-012 (5.44368)	1.37013e-015 (0.186069)	1.39666e-014 (0.00711344)	4.84603e-018 (1.4708e-006)
200	3.9421e-008	6.0633e-009 (6575.49)	2.97691e-012 (176855)	1.10731e-011 (4.82872)	9.482e-015 (0.730486)
300	2.59632e-006	1.98876e-007 (82704.2)	7.21701e-011 (1.77383e+007)	1.42645e-010 (3.0574)	1.80647e-013 (5.19794)
400	3.53129e-006	5.04466e-007 (156607)	1.75428e-010 (2.50624e+007)	3.23509e-010 (0.0297094)	3.25219e-013 (0.472803)
500	3.52521e-006	6.20882e-007 (153208)	2.33245e-010 (2.50597e+007)	6.24836e-010 (0.0319486)	3.94048e-013 (0.186986)
600	3.51331e-006	4.65661e-007 (155333)	2.76134e-010 (2.49686e+007)	7.42199e-010 (0.0199422)	4.41985e-013 (0.232678)
```

Change dipole.task to use

```
SOURCE.MAGNITUDE=0.000004
TSS.REC_BASEFILE=dp4
```

Changing TSS.REC_BASEFILE is for not overwriting previous simulation data.

Run the simulation again, a new statistics file is generated (only show the first 6 time steps):

Time	Energy	Max Divg E (||E||)	Avg Divg E (avg ||E||)	Max Divg H (||H||)	Avg Divg H (avg ||H||)
0	4.19064e-036	0 (0)	0 (2.98217e-023)	0 (1.58101e-322)	0 (0)
100	8.87688e-025	2.03288e-017 (3.60092e-005)	5.43615e-021 (2.9771e-012)	5.26216e-020 (2.84465e-008)	1.9209e-023 (2.35328e-017)
200	6.30735e-019	2.31296e-014 (0.0360018)	1.1918e-017 (2.82967e-006)	4.7338e-017 (1.92857e-005)	3.79125e-020 (1.16878e-011)
300	4.15411e-017	1.4803e-012 (0.525269)	2.87785e-016 (0.000283813)	5.18163e-016 (2.29737e-005)	7.19982e-019 (8.31671e-011)
400	5.65006e-017	1.11022e-012 (0.40461)	6.98778e-016 (0.000400999)	1.15144e-015 (5.48898e-008)	1.29797e-018 (7.56485e-012)
500	5.64033e-017	1.77636e-012 (0.612831)	9.29027e-016 (0.000400956)	1.82265e-015 (9.78859e-008)	1.5736e-018 (2.99178e-012)
600	5.6213e-017	3.40468e-012 (0.621334)	1.10234e-015 (0.000399497)	2.43035e-015 (3.26188e-008)	1.76583e-018 (3.72285e-012)

As expected the energy is much smaller due to the much smaller waveform magnitude.

Problem in using large cells

The advantage of the 4-th order or higher order TSS algorithm is its very high accuracy, allowing large space voxel sizes. For point field sources, it is not a problem to use large space voxel sizes. But for continuous field sources, using large voxel sizes causes big problem due to large digitization errors of the field source.

Point Source

Let's first examine the case of a point source. It is a Ricker source applied at the center of a computing domain. For details, see https://github.com/DavidGeUSA/TSS/blob/master/TSS_Linux_Windows.pdf

Using $\Delta_s = 0.03$, at space point (2.4, 2.4, 2.4), the TSS algorithm produces following E field values.

Time Index	E_x	E_y	E_z
0	12.5874	-25.1747	12.5874
1	12.5796	-25.1593	12.5796
2	12.5565	-25.113	12.5565
3	12.518	-25.036	12.518
4	12.4642	-24.9283	12.4642
5	12.3951	-24.7902	12.3951
6	12.3109	-24.6218	12.3109
7	12.2117	-24.4234	12.2117
8	12.0977	-24.1954	12.0977
9	11.969	-23.938	11.969
10	11.8258	-23.6516	11.8258
11	11.6684	-23.3368	11.6684
12	11.497	-22.994	11.497
13	11.3119	-22.6238	11.3119
14	11.1133	-22.2267	11.1133
15	10.9016	-21.8032	10.9016
16	10.6771	-21.3542	10.6771
17	10.4401	-20.8803	10.4401
18	10.1911	-20.3821	10.1911
19	9.93028	-19.8606	9.93028
20	9.65818	-19.3164	9.65818
21	9.37519	-18.7504	9.37519
22	9.08173	-18.1635	9.08173
23	8.77825	-17.5565	8.77825
24	8.46521	-16.9304	8.46521
25	8.14309	-16.2862	8.14309
26	7.81237	-15.6247	7.81237

27	7.47356	-14.9471	7.47356
28	7.12715	-14.2543	7.12715
29	6.77367	-13.5473	6.77367
30	6.41363	-12.8273	6.41363
31	6.04758	-12.0952	6.04758
32	5.67604	-11.3521	5.67604
33	5.29953	-10.5991	5.29953
34	4.91859	-9.83718	4.91859
35	4.53371	-9.06741	4.53371
36	4.14535	-8.2907	4.14535
37	3.75393	-7.50786	3.75393
38	3.35978	-6.71956	3.35978
39	2.96311	-5.92622	2.96311
40	2.564	-5.12801	2.564
41	2.16234	-4.32469	2.16234
42	1.75782	-3.51565	1.75782
43	1.34993	-2.69986	1.34993
44	0.937966	-1.87593	0.937966
45	0.521139	-1.04228	0.521139
46	0.0986749	-0.19735	0.0986749
47	-0.32999	0.659981	-0.32999
48	-0.764976	1.52995	-0.764976
49	-1.2057	2.41139	-1.2057
50	-1.65061	3.30121	-1.65061
51	-2.09702	4.19403	-2.09702
52	-2.54102	5.08204	-2.54102
53	-2.97762	5.95523	-2.97762
54	-3.40101	6.80202	-3.40101
55	-3.80519	7.61038	-3.80519
56	-4.18464	8.36928	-4.18464
57	-4.53517	9.07035	-4.53517
58	-4.8547	9.7094	-4.8547
59	-5.14384	10.2877	-5.14384
60	-5.40611	10.8122	-5.40611

Reduce the space cell size by half: $\Delta_s = 0.015$. Run a simulation again. The TSS algorithm produces following E field values at space point (2.4, 2.4, 2.4). Note that the Time Index values is doubled.

Time Index	E_x	E_y	E_z
0	12.5874	-25.1747	12.5874
2	12.5796	-25.1593	12.5796
4	12.5565	-25.113	12.5565
6	12.518	-25.036	12.518
8	12.4642	-24.9283	12.4642
10	12.3951	-24.7902	12.3951
12	12.3109	-24.6218	12.3109
14	12.2117	-24.4234	12.2117
16	12.0977	-24.1954	12.0977
18	11.969	-23.938	11.969
20	11.8258	-23.6517	11.8258
22	11.6684	-23.3369	11.6684
24	11.497	-22.9941	11.497
26	11.3119	-22.6238	11.3119
28	11.1133	-22.2267	11.1133
30	10.9016	-21.8033	10.9016
32	10.6771	-21.3542	10.6771
34	10.4401	-20.8803	10.4401
36	10.1911	-20.3821	10.1911
38	9.93029	-19.8606	9.93029
40	9.65819	-19.3164	9.65819
42	9.3752	-18.7504	9.3752
44	9.08173	-18.1635	9.08173
46	8.77825	-17.5565	8.77825
48	8.46522	-16.9304	8.46522

50	8.1431	-16.2862	8.1431
52	7.81238	-15.6248	7.81238
54	7.47356	-14.9471	7.47356
56	7.12716	-14.2543	7.12716
58	6.77368	-13.5474	6.77368
60	6.41365	-12.8273	6.41365
62	6.04761	-12.0952	6.04761
64	5.6761	-11.3522	5.6761
66	5.29966	-10.5993	5.29966
68	4.91885	-9.8377	4.91885
70	4.53422	-9.06844	4.53422
72	4.14633	-8.29265	4.14633
74	3.75573	-7.51145	3.75573
76	3.36297	-6.72594	3.36297
78	2.96858	-5.93716	2.96858
80	2.57303	-5.14607	2.57303
82	2.1767	-4.3534	2.1767
84	1.77973	-3.55946	1.77973
86	1.38187	-2.76375	1.38187
88	0.982238	-1.96448	0.982238
90	0.578988	-1.15798	0.578988
92	0.169116	-0.338231	0.169116
94	-0.251511	0.503022	-0.251511
96	-0.687528	1.37506	-0.687528
98	-1.14265	2.28529	-1.14265
100	-1.61738	3.23475	-1.61738
102	-2.10642	4.21284	-2.10642
104	-2.59682	5.19365	-2.59682
106	-3.06849	6.13697	-3.06849
108	-3.49824	6.99647	-3.49824
110	-3.86748	7.73495	-3.86748
112	-4.17105	8.34211	-4.17105
114	-4.42315	8.8463	-4.42315
116	-4.65552	9.31104	-4.65552
118	-4.90626	9.81251	-4.90626
120	-5.20236	10.4047	-5.20236

From the above data we can see that at the same times and at the same space point, the two simulations produced almost the same field values, showing that doubling the space step size caused little differences. Thus, using TSS, we can reduce simulation calculations by increasing space step size. That is, run simulations faster while achieving higher accuracy, comparing to the conventional YEE algorithm.

Continuous sources

When a continuous source is used, the benefits shown previously disappear due to the digitization errors of the field source.

For the dipole source, let's use $\Delta_s = 0.002$ and $\Delta_s = 0.004$ and do two simulations. At space point (0.05, 0.05, 0), the two simulations produced following E field values.

$\Delta_s = 0.002$:

Time Index	E_x	E_y	E_z	
0	0	0	0	
100	0.00296364		0.00296364	-0.00405862
200	15.923	15.923	-4.48125	
300	216.152	216.152	-33.0495	
400	142.15	142.15	19.0796	
500	217.804	217.804	33.7943	
600	141.971	141.971	-12.8219	

$\Delta_s = 0.004$:

Time Index	E_x	E_y	E_z	
0	0	0	0	
50	0.00290906		0.00290906	-0.0156266
100	8.40039	8.40039	-87.6141	
150	-186.393	-186.393	-939.713	
200	-135.814	-135.814	-496.815	
250	-175.994	-175.994	-43.1511	
300	-132.792	-132.792	-29.1067	

We can see that the two simulations quickly go different ways.

Solution

One solution to the problem is to use various voxel sizes. In space covering a continuous field source, use small voxel size. In space away from continuous field sources, use large voxel size.