Gun3P - Static PIC Solver for DC Guns

Accelerator Code Workshop
Stanford, March 27 – 31, 2023

Computational Electrodynamics Department
SLAC National Accelerator Laboratory

Gun3P: Parallel electrostatic and self-magnetostatic solver for DC guns

SLAC

- **The electrostatic field E and the magneto-static flux density B are expressed in terms of scalar potential φ and vector**

$$\mathbf{E} = -\nabla \phi, \qquad \mathbf{B} = \nabla \times \mathbf{A}$$

Electrostatic solver

$$-\nabla \cdot \left(\varepsilon \nabla \phi \right) = 0$$

Magnetostatic solver

$$-\nabla \times \left(\mu^{-1} \nabla \times \mathbf{A} \right) = 0$$

Particle tracking

$$\frac{d\mathbf{p}}{dt} = q\left[\mathbf{E} + \mathbf{v} \times \mathbf{B} \right]$$

- **Proper boundary conditions are applied on surfaces**

Dirichlet

$$\mathbf{E} \times \hat{\mathbf{n}} = 0 \;\Rightarrow\; \phi = 0, \quad \phi = V_{\mathrm{cathode}}, \quad \phi = V_{\mathrm{anode}}$$

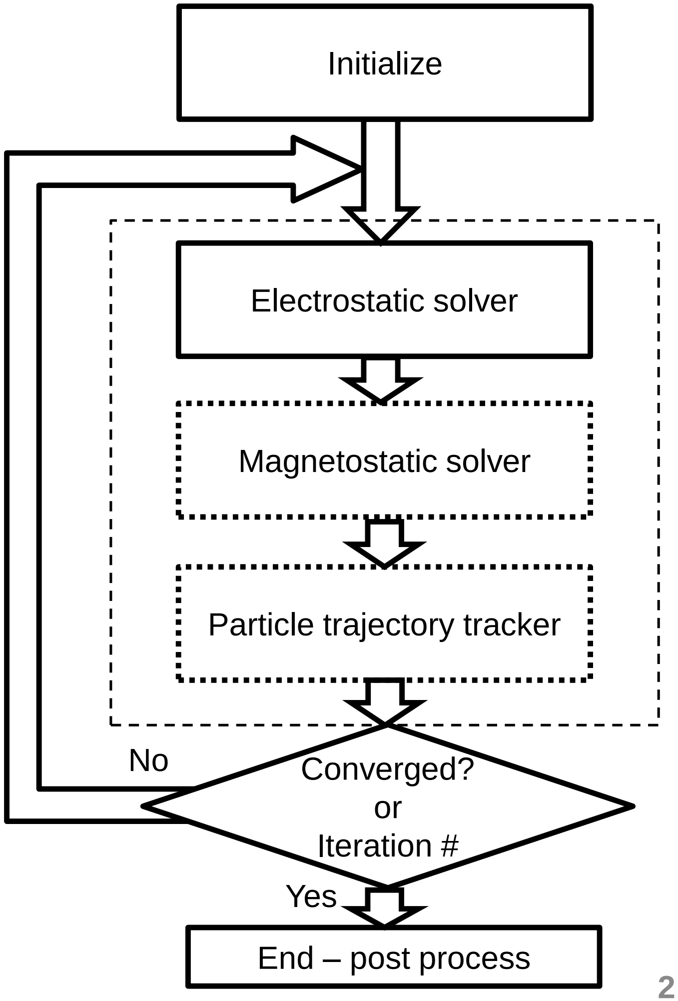
Metal surfaces (anode, cathode, focus electrodes, etc)

Neumann

$$\mathbf{E} \cdot \hat{\mathbf{n}} = 0 \;\Rightarrow\; \frac{\partial \phi}{\partial \hat{n}} = 0$$

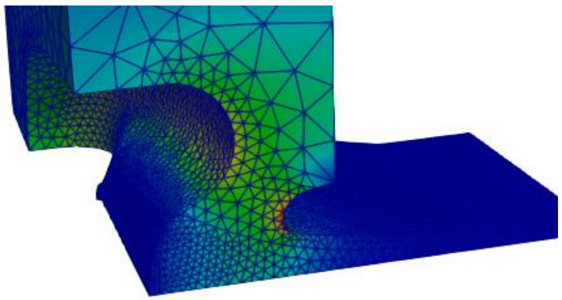
Symmetry surfaces

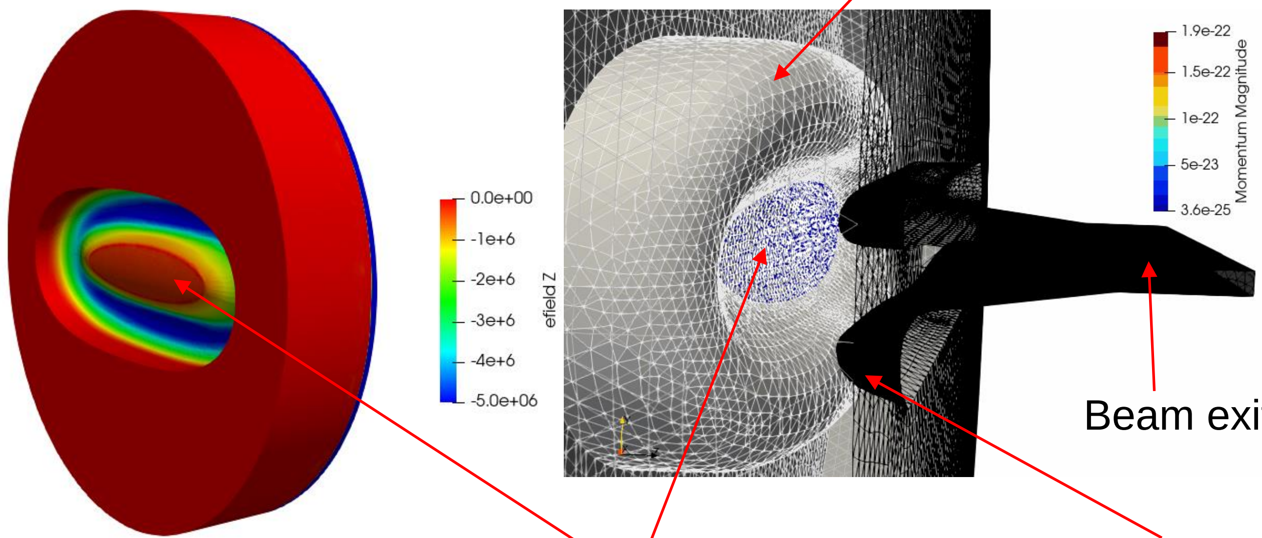
- **Tracking and iterations**
- **Exit at fixed number of iteration or when convergence is reached**



Gun3P Example - sheet beam klystron

(SBK) DC Gun





Unstructured mesh model of a sheet beam DC gun with Cubit

Gun3P calculates electrostatic, magnetostatic fields due to external and beam potential

Temporal evolution of electron trajectories calculated in Gun3p through tracking

Simulation with Gun3P

1) Generate geometry model (vacuum space)
2) Generate mesh - **CUBIT**
3) Convert Mesh – **ACDTool**
4) Electrostatic fields, beam current and particle trajectories **gun3p**
5) Postprocess – **ParaView**

Example: SBK DC gun

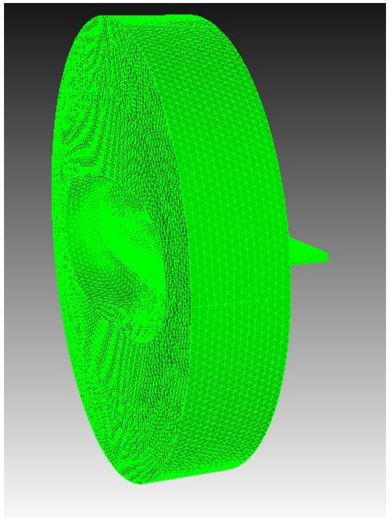
Directory: ../cw23/exercises/gun3p/LSBK

1) Generate mesh - CUBIT

SLAC

Go to working directory:

cd cw23/exercises/gun3p/LSBK



Launch Cubit: cubit

Make sure you copy SBK_3D.step into your directory

cd into directory where .step file is saved

for example:

```
>> cd "C:\Users\johndoe\cw23\exercises\gun3p\LSBK\"
```

Generate cavity:

Tools -> Play Journal File -> dcgun.jou:

generate mesh dcgun.gen

Convert Mesh - ACDTool

Convert mesh from genesis format to netcdf format /check mesh stats/check mesh quality

acdtool meshconvert dcgun.gen

Running acdtool for Mesh Conversion

Command: acdtool meshconvert dcgun.gen

- Copy Cubit genesis file and run script from local computer to NERSC, and logon to NERSC

  scp dcgun.gen run-meshconvert.batch user@perlmutter-p1.nersc.gov:directory_name/.
  ssh perlmutter-p1.nersc.gov

- Submit job on NERSC

  cd **directory_name**
  sbatch run-meshconvert.batch
  sqs

- Check output file acdtool.log for mesh correctness and mesh quality.

Standard Output from acdtool Mesh Conversion

Output file: acdtool.log

```
TOTALS:
  elements: 459206
  coordinates: 85736

EULER CHARACTER:
  Surf Euler Char = 2
  Vol Euler Char = 1
  Euler Char is OK.

ASPECT RATIO:
  min      = 1.02675
  max      = 3.34323 <- GREAT
  average  = 1.59004
  std dev  = 0.195627

SHAPE MEASURE:
  min      = 0.185159 <- GREAT
  max      = 0.999495
  average  = 0.803014
  std dev  = 0.107582
```

```
ELEMENT VOLUME:
  min      = 1.74816e-10
  max      = 1.18662e-07
  average  = 2.52024e-08
  std dev  = 1.6924e-08

BOUNDING BOX:
  min      = (-0.21971, -0.21971, -0.02)
  max      = (0.21971, 0.21971, 0.16)

EDGE LENGTH:
  min      = 0.00112911
  max      = 0.0134253
  average  = 0.00597493
  std dev  = 0.00218956

Existing boundary IDs:
1 2 3 4 10 12
```

```
Material IDs      Number of Elems:
1:                459206
```

2) Gun3P simulation: Input file

SLAC

Input file: dcgun.gun3p

```
DCGunProblem:
{
  RunId: test
  SymmetryFactor: 1
  //JustGenerateTreecotree: On
  ElectricAbsoluteTolerance: 0.
  ElectricRelativeTolerance: 1.00e-12
  ElectricAbsoluteTolerance2: 0.
  ElectricRelativeTolerance2: 1.00e-12
  MagneticAbsoluteTolerance: 0.
  MagneticRelativeTolerance: 1.00e-12
  MaxIterations: 60
//  FirstIterationForMagnetostaticProblem: 1
  Save iteration data: On
}
```

```
ElectrostaticProblem:
{  AnodeBoundaryId: 12
  MeshFile: dcgun.ncdf
  BasisOrder: 2
  CurvedSurfaces: on
  Charge averaging factor: 0.1
  VolumeMaterial: { Id: 1
    MediumName: Vacuum PhysicalParameter:
 {  Name: Permittivity    ConstantValue: 1.} }
   Write dofs to disk: on
  Debug bd dofs mapping: Off
  Debug matrix assembly: Off
Use reduced solver: On
  Outputs: {   Dofs for vizualization: On }
```

```
Boundary: {    Id: 1    ConditionType: Dirichlet DirichletValue: 0.  }
Boundary: {    Id: 2    ConditionType: Dirichlet DirichletValue : -500  }
Boundary: {    Id: 3    ConditionType: Neumann NeumannValue : 0.  }
Boundary: {    Id: 4    ConditionType: Dirichlet DirichletValue : 0.  }
Boundary: {    Id: 10    ConditionType: Neumann NeumannValue : 0.  }
Boundary: {    Id: 12    ConditionType: Dirichlet DirichletValue : 115000.  }
```

```
Compute non zero initial guess: On
LinearSolver: {    Solver:           CG
 Preconditioner:    CHOLESKY
PrintFrequency:    50
QuietMode:        0
 AbsoluteTolerance: 1.00e-99
Tolerance:        1.00e-16
MaxIterations:     5000  }}
```

2) Gun3P simulation: Input file (Cont'd)

Input file: dcgun.gun3p

Here calculate beam self magnetic field **(optional)**

```
MagnetostaticProblem:
{MeshFile: dcgun.ncdf
BasisOrder: 2
CurvedSurfaces: on
Charge averaging factor: 0.1  VolumeMaterial:  {
 Id: 1
 MediumName: Vacuum PhysicalParameter:  {
 Name: Permittivity
ConstantValue: 1.
                                   }
}
Write dofs to disk: on
Debug bd dofs mapping: Off
Debug matrix assembly: Off
Use reduced solver: On
Outputs:
{    Dofs for vizualization: On }
```

```
Boundary:  {    Id: 1    ConditionType: Dirichlet DirichletValue: 0.  }
Boundary:  {    Id: 2    ConditionType: Dirichlet DirichletValue : 0  }
Boundary:  {    Id: 3    ConditionType: Dirichlet DirichletValue : 0.  }
Boundary:  {    Id: 4    ConditionType: Dirichlet DirichletValue : 0.  }
Boundary:  {    Id: 10    ConditionType: Dirichlet DirichletValue : 0.  }
Boundary:  {    Id: 12    ConditionType: Dirichlet DirichletValue : 0  }
```

```
Compute non zero initial guess: On
LinearSolver:  {    Solver:           MUMPSFLOAT
 Preconditioner:    CHOLESKY
PrintFrequency:    50
QuietMode:        0
 AbsoluteTolerance: 1.00e-99
Tolerance:        1.00e-16
MaxIterations:     5000  }}
```

2) Gun3P simulation: Input file (Cont'd)

SLAC

Input file: dcgun.gun3p

Here calculate beam trajectory

```
Tracker:
{  JobName: ./gun3p_results/OUTPUT
   Particlefile: partpath
    t: 6.0e-6
    Generate output for vizualization: Off
    Print emission statistics: On
    Backward:  {
     t: 6.0e-6
     Generate output for vizualization: Off
    }
Domain:  {
t0: 0.0
 dt: 1.0e-12
MaxImpacts: 2
LowEnergy: 0.0
HighEnergy: 1.0e+99
InitialEnergy: 0.0
```

```
dt backward: 1.0e-12
Backward velocity pattern: 0
Minimum backward velocity factor: 0.01
Strategy for dt: 0
Emission Nx:   200
Emission Ny:   100
Emission Nz:   300.
Tracking Nx:   200.
Tracking Ny:   100.
  Tracking Nz:   300
  Tracking box x min: -0.11
  Tracking box y min: -0.04
  Tracking box z min: -0.001
  Tracking box x max: 0.11
  Tracking box y max: 0.04
  Tracking box z max: 0.18  }
```

```
Monitor:  {    Type: PlaneCrossingsVsT
Name: screen
Plane point: 0., 0., 0.15999
Plane normal: 0., 0., 1.
Compute densities: On
Max x for densities: 0.110
Max y for densities: 0.004
X number of densities: 11
  Y number of densities: 80  }
```

2) Gun3P simulation: Input file (Cont'd)

SLAC

Input file: dcgun.gun3p

Here lots of options for emission model

```
MeshInterface:
{   Type: pd
    Localizer: {     Type: USS_Curve    } }
Emitter:  {
Type: 6    // Child Langmuir
BoundaryID: 8
t0: -1.0e-12
t1:  1.0e-12
N:  1.0e+0      // number of unit particles in the macroparticle
M:  9.10938e-31   // real mass of a unit particle
Q: -1.60218e-19
d:  0.1e-3
Sample Type: 1    // 1: center    // 2: stochastic
Child Langmuir:    {Whole boundary is emitter: On
Use electric field direction for initial forward velocity: on
Strategy: 6   //fixed distance for strategy pass 6
Use average J: On
Backward velocity factor: 0.66666667
Forward velocity factor: 1.
Min allowed distance: 0.01e-3
Max allowed distance: 5.00e-3
Given delta phi: 100.0
Max allowed emission distance wrt z axis: 1.000000
First inter cycle factor: 1.0. //default is 0.5
Inter cycle averaging factor: 1.0. //default is 0.5
Max allowed total current: 1.35e+9
Reference z for emission surface: 1.50e-3
Particles per face: 1    }  }
```

```
// Material Type:  1: Reflector    2: Absorber    3: Secondary emitter 4: Test
//surface    5: SymmetryPlane
Material:  {BoundarySurfaceID: 1    Type: 2 }
Material:  {    BoundarySurfaceID: 2    Type: 2  }
 Material:  {    BoundarySurfaceID: 3    Type: 2  }
Material:  {    BoundarySurfaceID: 4    Type: 2  }
Material:  {    BoundarySurfaceID: 6    Type: 2  }
Material:  {    BoundarySurfaceID: 10    Type: 2  }
Material:  {    BoundarySurfaceID: 11    Type: 2  }

}
```

Running gun3p

- Copy gun3p startup file and run script from local computer to NERSC, and logon to NERSC

  scp dcgun.gun3p run-gun3p.batch user@perlmutter-p1.nersc.gov:directory_name/.
  ssh perlmutter-p1.nersc.gov

- Submit job on NERSC

  cd **directory_name**
  sbatch run-gun3p.batch
  sqs

- Transfer data form NERSC to local computer

  scp -r user@perlmutter-p1.nersc.gov:directory_name/ .*

gun3p Results

Gun3P result files are in the gun3p_results directory:

- *Screen output – **gun3p_results/gun3p.log***
- *Field and Particle results in subdirectory*

OUTPUT

*Time-domain field volume monitors: partpath_t0000*ps.out.mod*

Particles

- *Micro perveance for each iteration*

- *To get the electric field .mod file run*

  run-gun3p-postprocess.batch

  for the specific iteration

srun -n 1 --cpu_bind=cores /global/cfs/cdirs/ace3p/perlmutter/CPU/gun3pOutputConverter dcgun.gun3p e 30

electric fields

Iteration # (optional)

srun -n 1 --cpu_bind=cores /global/cfs/cdirs/ace3p/perlmutter/CPU/gun3pOutputConverter dcgun.gun3p m 30

magnetic field

gun3p Results: Convergence & Benchamrk

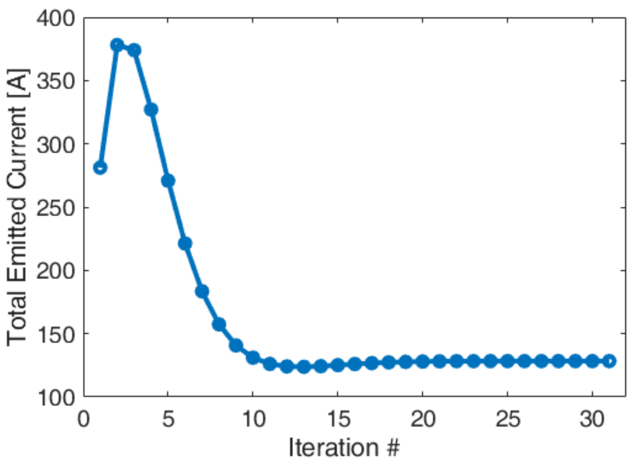
- *Screen output – **gun3p_results/gun3p.log***

gun3p iter 3: E rel. tol. = 0.002677669016, H rel. tol. = 0.000000000000, converged = 0
Gun3p iter 3: at emitter, total number of emitted particles = 11268, total current (with symmetry factor, uA) = **327035360.583706**
Final result, plane monitor ‘screen': ex = 419.326556, ey = 49.646728, (ex+ey)/2 = 234.486642, er = 1679.159719, maxCrossingR = 0.088783
Final result, plane monitor ‘screen': nex = 296.616015, ney = 35.118249, (nex+ney)/2 = 165.867132, ner = 1187.775154

gun3p iter 4: E rel. tol. = 0.002020987464, H rel. tol. = 0.000000000000, converged = 0
Gun3p iter 4: at emitter, total number of emitted particles = 11268, total current (with symmetry factor, uA) = **270856592.158773**
Final result, plane monitor screen: ex = 444.309539, ey = 50.446245, (ex+ey)/2 = 247.377892, er = 2161.949293, maxCrossingR = 0.091402
Final result, plane monitor ‘screen ': nex = 314.236528, ney = 35.677948, (nex+ney)/2 = 174.957238, ner = 1529.031857
……

- *Benchmarked against MICHELLE* →

MICHELLE is a commercial 3D finite element Gun Code



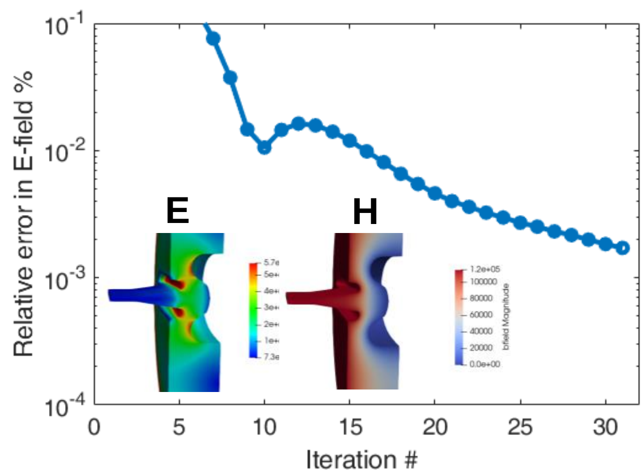


TABLE III

INPUT VALUES USED FOR GUN3P AND MICHELLE SIMULATIONS.

Problem	Input Parameter	Gun3P Value	MICHELLE Value
All	# cycles	41	80
	# elems	1,316,530 tetrahedra	1,073,000 hexahedrons
	q	2	1
	# cpus	48 (1.9 GHz each)	1 (3.0 GHz)
E	V_{focus} (V)	-500	-500
	V_{anode} (V)	115,000	115,000
	$p_{\mathbf{E}}$	3	1
B	$p_{\mathbf{B}}$	1	unknown
T	$d(\mu\text{m})$	100	unknown
	N	145,675	96,064
	Δt (ps)	2	unknown

TABLE IV

UTPUT VALUES OBTAINED BY GUN3P AND MICHELLE SIMULATIONS.

Output parameter	Gun3P Value	MICHELLE Value
I(A)	129.3	129.3
$J_{cathode}$ (A/cm^2)	min = 1.7528 max = 2.3622	min = 1.7500 max = 2.4000
$\epsilon_{y,z=18\text{cm}}$ (π mm-mrad)	4.3	unknown
$\rho_{z=14\text{cm}}$ (mC/m^3)	max = 3.9	unknown
Hitting percentage	0%	0%
# dof$_{\mathbf{E}}$	6,114,694	unknown
# dof$_{\mathbf{B}}$	1,599,393	unknown
# time steps/cycle	785	unknown
Run time (hrs)	5.5	63.6

Prudencio *et al.*, SLAC-PUB-13097, Jan. 2008

Gun3P as an Electrostatic Field Solver

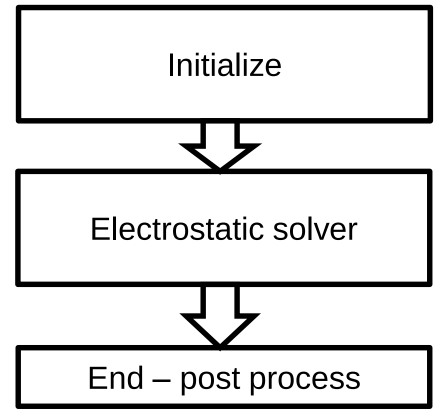
SLAC

- **Gun3P can solve the electrostatic fields with <u>no beam</u>**

```
DCGunProblem:
{
  RunId: test
  SymmetryFactor: 1
  ElectricAbsoluteTolerance: 0.
  ElectricRelativeTolerance: 1.00e-12
  ElectricAbsoluteTolerance2: 0.
  ElectricRelativeTolerance2: 1.00e-12
MaxIterations: 60
Save iteration data: On
}
```

```
ElectrostaticProblem:
{  AnodeBoundaryId: 12
  MeshFile: dcgun.ncdf
  BasisOrder: 2
  CurvedSurfaces: on
  Charge averaging factor: 0.1
  VolumeMaterial: { Id: 1
    MediumName: Vacuum PhysicalParameter:
 {  Name: Permittivity    ConstantValue: 1.} }
   Write dofs to disk: on
  Debug bd dofs mapping: Off
  Debug matrix assembly: Off
Use reduced solver: On
  Outputs: {   Dofs for vizualization: On
  FileName: ElectrostaticField.data  Nx: 40  Ny: 50  Nz: 60 }
```

- Electrostatic fields can be exported on a structured (Cartesian) grid




```
Boundary:  {    Id: 1    ConditionType: Dirichlet DirichletValue: 0.  }
Boundary:  {    Id: 2    ConditionType: Dirichlet DirichletValue : -500  }
Boundary:  {    Id: 3    ConditionType: Neumann NeumannValue : 0.  }
Boundary:  {    Id: 4    ConditionType: Dirichlet DirichletValue : 0.  }
Boundary:  {    Id: 10    ConditionType: Neumann NeumannValue : 0.  }
Boundary:  {    Id: 12    ConditionType: Dirichlet DirichletValue : 115000.  }
```

```
Compute non zero initial guess: On
LinearSolver: {    Solver:            CG
 Preconditioner:    CHOLESKY
PrintFrequency:    50
QuietMode:         0
 AbsoluteTolerance: 1.00e-99
Tolerance:         1.00e-16
MaxIterations:     5000  }}
```