Whipple Shield Topology Optimisation Concept Study using LS-OPT

K. Fowler, T. Oudes

May 2, 2022

Contents

1. Introduction

This report aims to record the concept development of a topology optimisation routine for Whipple shield application. LS-OPT was implemented along with MATLAB and LS-PREPOST packages. Predetermined nodal coordinates relating to the geometry of the rear wall of a Whipple shield model were optimised with respect to the average displacement of the back side of the rear wall. The optimisation objective was to minimise this average displacement.

The routine operated by firstly creating a model in LS-PREPOST, running the model in LS-DYNA, outputting the desired response variable (the average displacement of the back side of the rear wall) and iterating the design by altering the input parameters (specified nodal coordinates on rear wall) according to the optimisation objective until an optimum solution was achieved. Figure 1 shows the LS-OPT user interface to assist with visualisation of the optimisation routine.

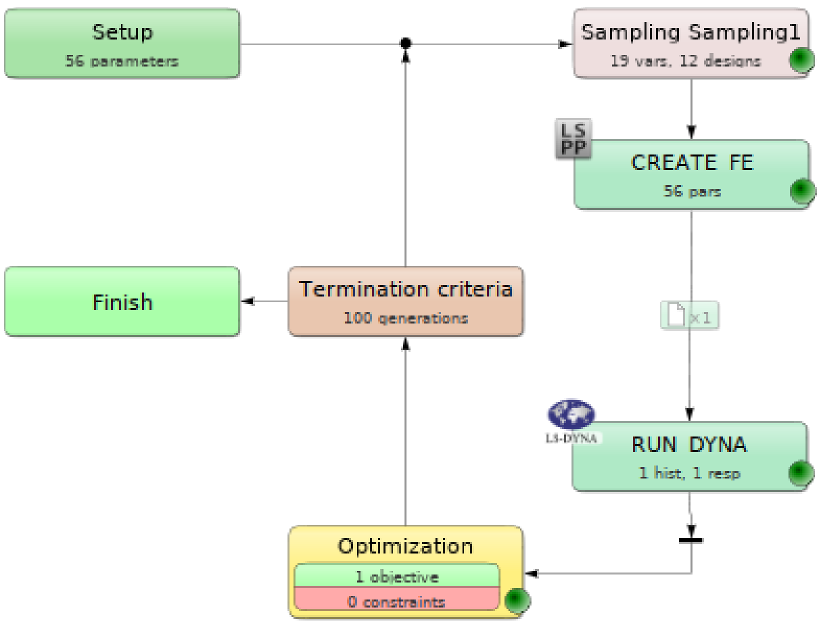


Figure 1: Overview of optimisation routine.

The *CREATE FE* block was required to remodel the rear wall (the mesh arrangement and nodal coordinates) for each optimisation iteration. The *RUN DYNA* block then used LS-DYNA to run each simulation on the cluster. With 56 parameters, of which 19 were variable, and 12 simulations per iteration, the total time taken for the optimisation routine to run was 4.5 hours.

1.1. General Methodology Overview

The following list summarises the methodology implemented.

- Model geometry and mesh characteristics defined within command files
- Parameter file created as a command file input, containing variables used in the optimisation routine
- Key files generated from command files
- Key files input into structure key file

- Optimisation routine set-up using the LS-OPT user interface, selecting inputs and defining command lines to access the cluster
- Objectives and constraints defined, ensuring parameters correctly detected
- Baseline run conducted to check for initial errors
- Optimisation routine normal run conducted

2. Whipple shield Model

The Whipple shield geometry was determined as shown in Figure 2. The dual wall set up was established to represent debris impacting two AA6070-T6 plates at 6.7 km/s. The projectile (ball) was modelled using AA6061-T6. The projectile, front and back plates are referred to as ball, target and wall throughout, respectively. The object of optimisation was the rear wall exclusively.

Figure 2: Dual plate Whipple shield set-up.

Due to computational efficiency, the optimisation routine was run using a quasi-2D model. Using a strict 2D model was not possible as the FEM/DES method implemented did not allow for this and a full 3D model was extremely time consuming. Therefore, a 3D model with a thickness of one element size in the z-direction (in the conventional coordinate system) was used to approximate the behaviour of a 2D plate. This was achieved by constraining the FEM parts in the z-direction as well as applying the original model boundary conditions on the target and rear wall. A further study was then completed to extend the 2D optimisation to a 3D model.

3. LS-PREPOST

3.1. Command files

A command file was necessary as the mesh could not be hard coded into the key file to be used by LS-DYNA because changing position of the specified nodes in the rear wall during the optimisation routine changed the numbering of each node and hence the mesh arrangement. The command file was used to update the mesh every time the node coordinates changed to keep the mesh arrangement consistent with the updated geometry.

Using LS-PREPOST, two command files (.cfile) were created to incorporate all the necessary geometries and mesh characteristics required to model the Whipple shield. All operations in LS-PREPOST were written within the command file. The ball and target information was contained in a separate command file to the rear wall. This was to allow changes to be easily made to the geometry of each part during the set up to make modelling more convenient, however, was not strictly necessary.

Node sets were defined to account for boundary conditions in these files. The boundary conditions had to be updated on each iteration as the numbering of the nodes changed. The command file for the rear wall (create_mesh_wall.cfile) also included a call function for the parameter command file ($openccommand "para01.cfile"$) in which the x and y nodal coordinates were fully defined. This allowed the values to be easily altered when required.

The output of the two command files were saved as two separate key files containing the ball and target data (mesh_ball_target.key), and rear wall data (mesh_ball_target.key), using the $'savekeyword "..."'$ command.

Additional aspects to note:

- Node sets were defined using the LS-PREPOST *box* function because selecting the nodes using the *area* function caused problems when model iterations were generated as this function relates to the coordinates seen on the screen during node selection rather than the geometries of the model itself.
- Command files had to be renamed in order to save changes to multiple iterations, as saving an updated model overrides the previous iteration if located in the same folder
- It was not possible to perform mathematical operations (additions, divisions, etc.) in LS-PREPOST. Therefore, it was more convenient to write an additional MATLAB file to write the command file to easily account for geometrical changes.
- Identifying the correct part numbers for each component was included within the original command files using the *PART keyword.

3.2. Structure Key file

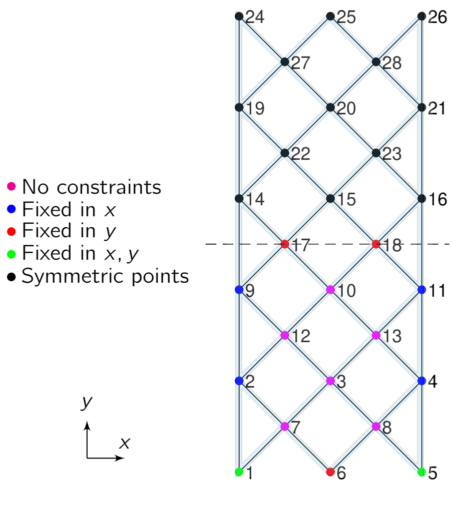
A structure file was created which included all LS-PREPOST keywords required to run the simulation in LS-DYNA. For example: *CONTROL_TIMESTEP, *BOUNDARY_SPC_SET_ID, etc. The ball and target key file, and rear wall key file were called into the structure key file using an *INCLUDE command. The structure key file was originally adapted from an existing k.file to aid code accuracy. It should be noted that the boundary condition node sets should correspond to the correct ID and be included within this file.

4. LS-OPT Optimisation Routine

Figure 1 displays the complete optimisation routine as set up in LS-OPT. The corresponding routine blocks are further discussed in the following sections.

4.1. Parameter Set-up

The x and y coordinates of each selected node were set as the optimisation input parameters. Figure 3 shows the constraints applied to each node. A continuous range was defined for each variable in the respective degree of freedom, each with a starting, maximum and minimum value. A constant was defined for the nodes fixed in x and y, and a dependence relationship was defined to account for the line of symmetry indicated in Figure 3. The value/range of each nodal coordinate effectively define the constraints of the model.



Variables	
x	y
-	y_2
x_3	y_3
-	y_4
x_6	-
x_7	y_7
x_8	y_8
-	y_9
x_{10}	y_{10}
-	x_{11}
x_{12}	y_{12}
x_{13}	y_{13}
x_{17}	-
x_{18}	-

Figure 3: Rear wall with all degrees of freedom.

Note: Parameters are automatically detected by LS-OPT if the parameter file is located in the same folder as the optimisation routine file. Only user defined parameters need to be manually input

4.2. Creating Finite Elements

Figure 4 shows the implementation of LS-PREPOST within the LS-OPT routine. '*create_mesh_wall.cfile*' is the input file required to generate the rear wall mesh with each routine iteration. The updated mesh details can then be used in LS-DYNA to re-run the simulation with the next set of nodal coordinates.

The command line shown in Figure 4 accesses LS-PREPOST to allow the command file to run. The line $'-nographics'$ was useful in speeding up the process as graphics were not necessary at this stage. The fields specifying the execution resources as shown in Figure 4, can be ignored.

4.3. Running LS-DYNA

To run the simulation in LS-DYNA, the files had to be transferred to the correct directory. The source file was copied to the destination file as shown in Figure 5.

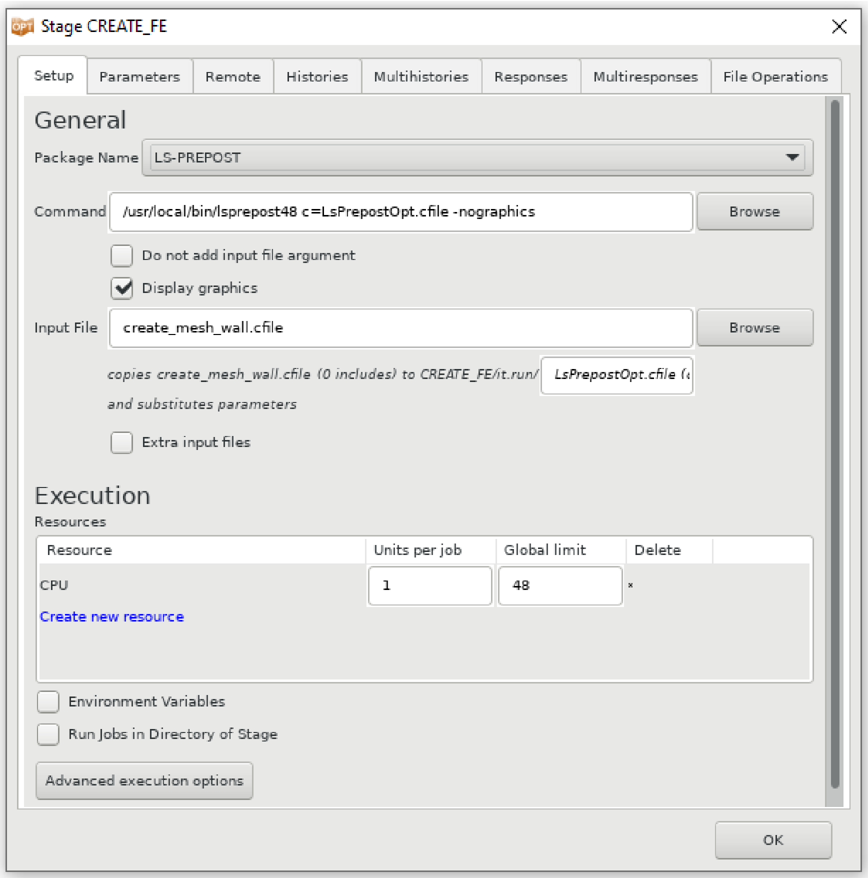


Figure 4: Create Finite Element set up in LS-OPT.

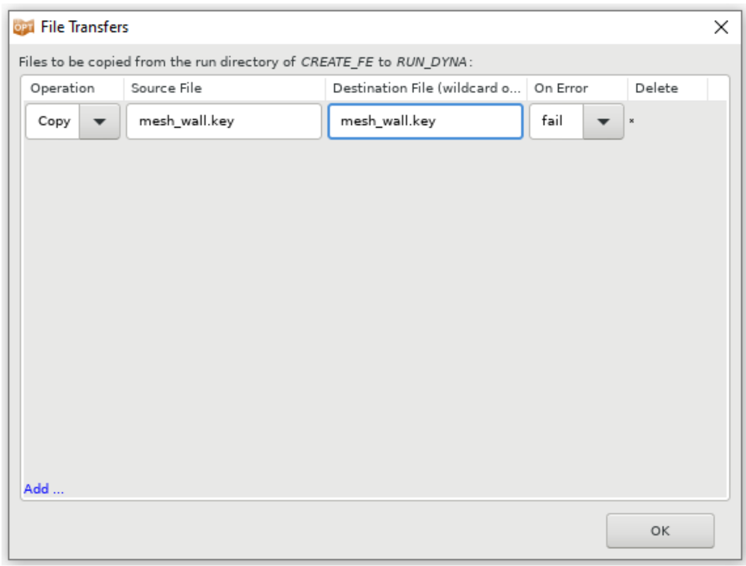


Figure 5: File transfer in LS-OPT.

The command required to access LS-DYNA is shown in Figure 6, where the input file is the structure key file containing the updated mesh information. An extra input file for the projectile and front plate was also included (the mesh_ball_target.key file).

The optimisation objective was to minimise the average displacement of the back side of the rear wall, and therefore this parameter was the required output during the optimisation routine. The output command file was generated using LS-PREPOST. The operation function was used to average the data curves for all nodes on the back side of the rear wall and the data history containing coordinates of the xy plot was saved accordingly.

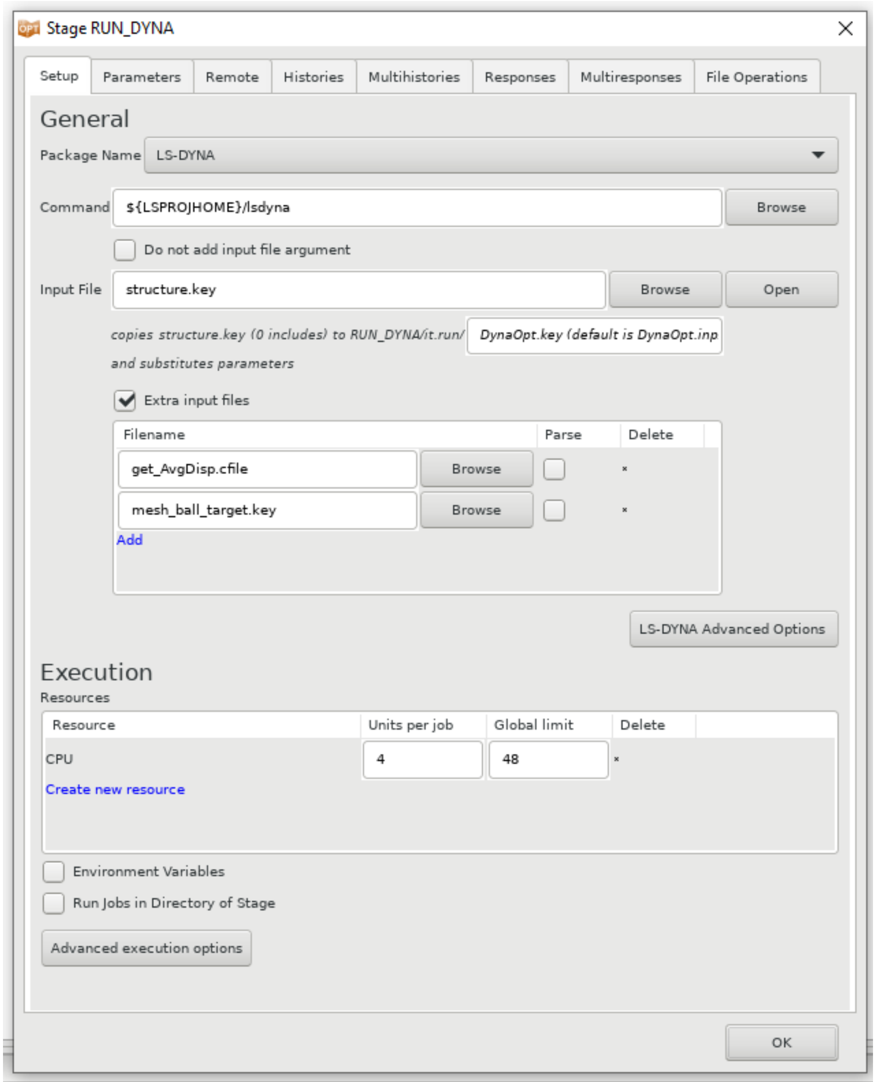


Figure 6: Calling LS-DYNA within LS-OPT.

An additional command file was also added as an extra input file to create a user defined response (get_AvgDisp.cfile). The history tab (shown in Figure 7) was used to obtain all the history information throughout the simulation from the LsoptHistory file that was generated and stored as TimeAvgDisp as user defined in LS-OPT.

In the responses tab (shown in Figure 8) a user defined response was created referring to the TimeAvgDisp. The results for the final value of each simulation were stored as the AvgDisp and used during the optimisation routine. The final value was consistently the maximum value and therefore this method was justified.

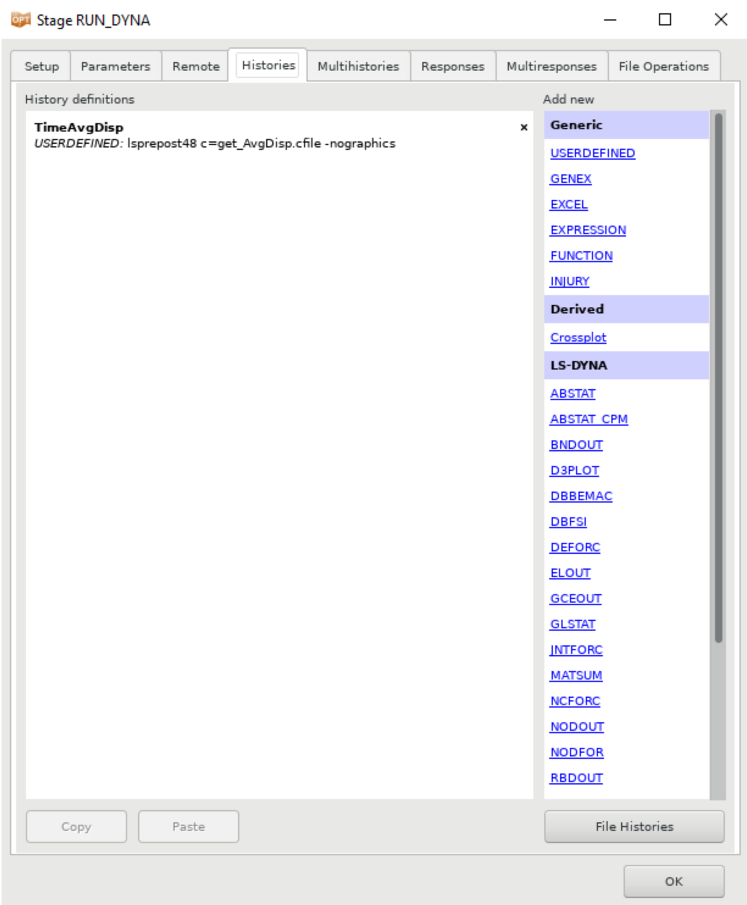


Figure 7: User defined response - Histories tab.

Additional aspects to note:

- Data written from get_AvgDisp.cfile should be read in an extension-less file named "LsoptHistory". User defined histories can only be loaded into LS-OPT with that particular file name.
- Optimisations with respect to energies (e.g. total energy) can be performed via the GLSTAT option in the tab "Responses". This negates the need for a user defined sub-routine. Option 3 should be selected in write to binary in the .key file.

  When working with displacements or an element selection within a region, the GLSTAT did not provide the correct information. For this reason the user defined routine was chosen.

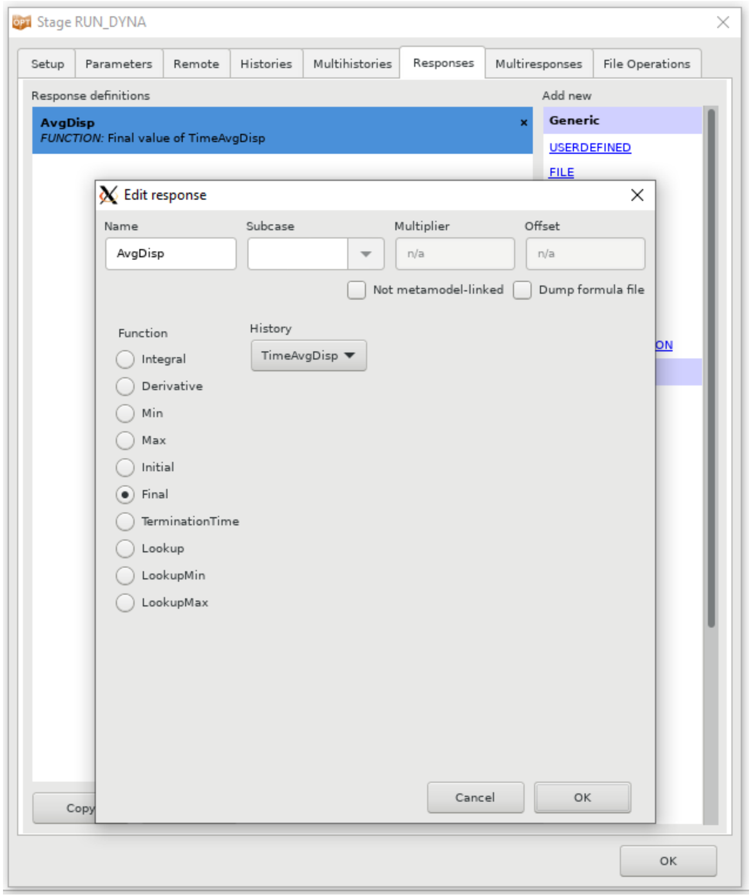


Figure 8: User defined response - Responses tab.

4.4. Running the Optimisation Routine

Table 1 shows the file names and corresponding extensions required to run the optimisation routine.

Table 1: Files required to run LS-OPT optimisation.

File Name	File Extension
optimisation	.lsopt
para01	.cfile
create_mesh_wall	.cfile
get_AvgDisp	.cfile
mesh_ball_target	.key
structure	.key
lsdyna	

Additional aspects to note concerning running the optimisation routine on the cluster:

- Conducting a 'baseline run' prior to running the complete optimisation ('normal run') checked for any initial simulation errors

- 12 designs were run simultaneously for each optimisation routine cycle as to complete the optimisation within a reasonable time frame without overloading the cluster.
- Under Execution (see in Figure6), the resources are stated. This includes the total number of available CPUs (global limit), and the number of CPUs per simulation (units per job).

  The number of units per job correspond to the global limit divided by the population size, i.e. 48/12 = 4.
- Altering the file within the WinSPC workspace converts files to Unix rather than Linux. The cluster requires Linux to run correctly and therefore this conversion should be avoided.
- The lsdyna file (a Linux compatible txt file) was required to submit jobs to the cluster to run the optimisation routine.
- The following line was added to the lsdyna file to remove *D3DUMP* files as these were large files unnecessary for the required results: $'exportLSTC\_D3DUMP = off'$
- All relevant files should be placed in the same folder before starting the optimisation routine
- Each time the optimisation routine was restarted, the folder in which the relevant files existed was cleared using the 'clean all' function
- *Xming* software was used in addition to LS-OPT. It was coupled with *PuTTY* which required downloading and set-up. *XLaunch* was run and the command *lsoptui* was required to open LS-OPT. Figure **??** shows the configuration set-up for PuTTY.

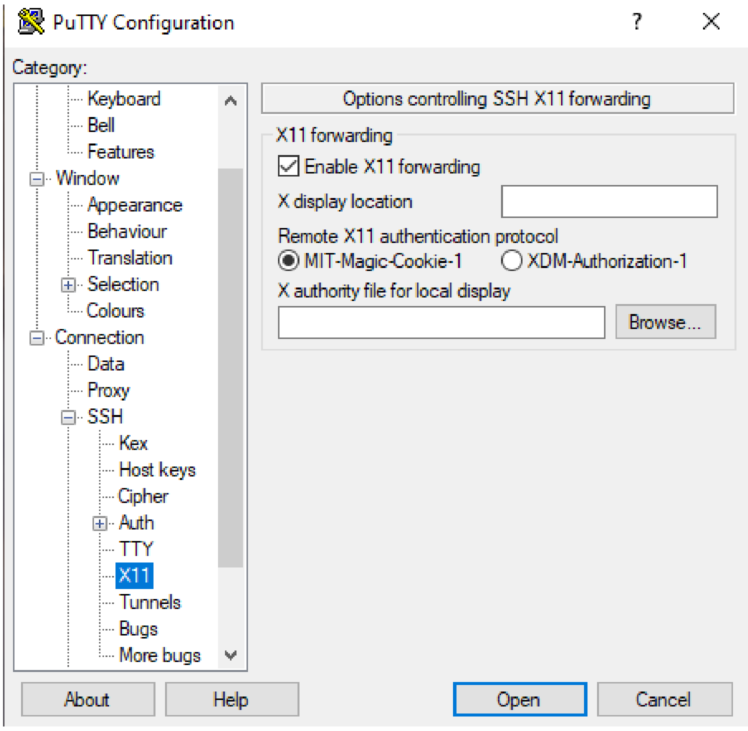


Figure 9: PuTTY Configuration.

4.5. Termination Criteria

The termination criteria was set to 100 generations with 0.1 max repeat optimum/generations. This infers that it takes 10 simulations with no change in the optimised model output for the routine to be complete. However, it was found that adequately accurate results were achieved before this value was reached.

Appendix A. get_AvgDisp.cfile

```
//
// COMMAND FILE TO RETRIEVE AVERAGE DISPLACEMENT OF BACK SIDE OF BACK WALL DATA
//
open d3plot d3plot
genselect target node
loadstate 1;
loadquat 0.000000 0.000000 0.000000 1.000000;
loadgrot 1.000000 0.000000 0.000000 0.000000 0.000000 1.000000 0.000000 0.000000
    0.000000 0.000000 1.000000 0.000000 0.000000 0.000000 0.000000 1.000000 ;
loadzoom 0.409600;
loadpan 52.876713 0.227008;
loadeyepos 0.000000 0.000000 0.000000;
loadupvect 0.000000 1.000000 0.000000;
genselect node add box in 130.9 -1000.000000 -0.100000 135.000000 1000.000000 2.0
ntime 8
xyplot 1 operation average_curves all
xyplot 1 savefile xypair LsoptHistory 1
deletewin 1
quit
```

Appendix B. structure.key

```
$# LS-DYNA Keyword file created by LS-PrePost(R) V4.8.25 - 04Jan2022
$# Created on Apr-5-2022 (16:30:58)
*KEYWORD
*TITLE
$#                                                                         title
LS-DYNA keyword deck by LS-PrePost
*CONTROL_ACCURACY
$#     osu       inn    pidosu      iacc
         1         4         0         0
*CONTROL_BULK_VISCOSITY
$#      q1        q2      type     btype    tstype
       1.5      0.06        -2         0         0
*CONTROL_CONTACT
$#  slsfac    rwpnal    islchk    shlthk    penopt    thkchg     orien    enmass
       0.2       0.0         1         0         1         0         1         0
$#  usrstr    usrfrc     nsbcs    interm     xpene     ssthk      ecdt   tiedprj
         0         0         0         0       4.0         0         0         0
$#   sfric     dfric       edc       vfc        th     th_sf    pen_sf
       0.0       0.0       0.0       0.0       0.0       0.0       0.0
$#  ignore    frceng   skiprwg    outseg   spotstp   spotdel   spothin
         2         1         0         0         0         0       0.0
$#    isym    nserod    rwgaps    rwgdth     rwksf      icov    swradf    ithoff
         0         0         1       0.0       1.0         0       0.0         0
$#  shledg    pstiff    ithcnt    tdcnof     ftall    unused    shltrw    igactc
         0         0         0         0         0                 0.0         0
*CONTROL_DISCRETE_ELEMENT
$#   ndamp     tdamp     frics     fricr     normk    sheark       cap       vtk
       0.0       0.0       0.0       0.0      0.01       0.0         0         0
$#   gamma       vol       ang       gap    unused    ignore      nbuf  parallel

$#   lnorm    lshear    unused     fricd        dc      ncrb        bt        dt
         0         0                 0.0       0.0         0       0.01.00000E20
$#      cp        tc      tfac
       0.0       0.0       0.0
$# idesoft    sofscl    unused     iskip
         0       0.1                  20
*CONTROL_ENERGY
$#    hgen      rwen    slnten     rylen     irgen
         2         1         2         2         1
*CONTROL_HOURGLASS
$#     ihq        qh
         3       0.1
*CONTROL_OUTPUT
$#   npopt    neecho    nrefup    iaccop     opifs    ipnint    ikedit    iflush
         0         0         0         0       0.0         0       100      5000
$#   iprtf    ierode   tet10s8    msgmax    ipcurv      gmdt   ip1dblt      eocs
         0         1         2        50         0       0.0         0         0
$#   tolev    newleg    frfreq     minfo    solsig    msgflg    cdetol
         2         0         1         0         0         0      10.0
$# phschng    demden   icrfile   spc2bnd    penout    shlsig   hisnout    engout
         0         0         0         0         0         0         0         0
$#    insf    isolsf      ibsf      issf    mlkbag
         0         0         0         0         0
*CONTROL_SOLID
$#   esort   fmatrix   niptets    swlocl    psfail   t10jtol    icohed    tet13k
         1         0         4         1         0       0.0         0         0
$#   pm1     pm2     pm3     pm4     pm5     pm6     pm7     pm8     pm9    pm10
       0       0       0       0       0       0       0       0       0       0
*CONTROL_SOLUTION
$#    soln       nlq     isnan     lcint     lcacc     ncdcf     nocop
```

```
         0         0         0       400         1         1         0
*CONTROL_TERMINATION
$#  endtim    endcyc     dtmin    endeng    endmas     nosol
6.00000E-5         0       0.0       0.01.000000E8         0
*CONTROL_TIMESTEP
$#  dtinit    tssfac      isdo    tslimt     dt2ms      lctm     erode     ms1st
       0.0       0.1         0       0.0       0.0         0         0         0
$#  dt2msf   dt2mslc     imscl    unused    unused     rmscl    unused      ihdo
       0.0         0         0                           0.0                   0
*DATABASE_GLSTAT
$#      dt    binary      lcur     ioopt
1.00000E-8         3         0         1
*DATABASE_GLSTAT_MASS_PROPERTIES
$#      dt    binary      lcur     ioopt
1.00000E-8         3         0         1
*DATABASE_MATSUM
$#      dt    binary      lcur     ioopt
1.00000E-8         3         0         1
*DATABASE_NODOUT
$#      dt    binary      lcur     ioopt   option1   option2
       0.0         3         0         1       0.0         0
*DATABASE_SLEOUT
$#      dt    binary      lcur     ioopt
1.00000E-8         3         0         1
*DATABASE_BINARY_D3PLOT
$#      dt      lcdt      beam     npltc    psetid
       0.0         0         0        30         0
$#   ioopt      rate    cutoff    window      type      pset
         0       0.0       0.0       0.0         0         0
*DATABASE_EXTENT_BINARY
$#   neiph     neips    maxint    strflg    sigflg    epsflg    rltflg    engflg
        20        20         3         0         1         1         1         1
$#  cmpflg    ieverp    beamip     dcomp      shge     stssz    n3thdt   ialemat
         0         0         0         1         1         1         2         1
$# nintsld   pkp_sen      sclp     hydro     msscl     therm    intout    nodout
         0         0       1.0         2         0         0
$#    dtdt    resplt     neipb     quadr     cubic
         0         0         0         0         0
*CONTACT_ERODING_NODES_TO_SURFACE_ID
$#     cid                                                                 title
         3
$#    ssid      msid     sstyp     mstyp    sboxid    mboxid       spr       mpr
        50        12         4         2         0         0         0         0
$#      fs        fd        dc        vc       vdc    penchk        bt        dt
      0.01      0.01       0.0       0.0       5.0         0       0.01.00000E20
$#     sfs       sfm       sst       mst      sfst      sfmt       fsf       vsf
       1.0       1.0       0.0       0.0       1.0       1.0       1.0       1.0
$#    isym    erosop      iadj
         1         1         1
$#    soft    sofscl    lcidab    maxpar     sbopt     depth     bsort    frcfrq
         1       0.1         0     1.025       2.0         2         1         1
*SET_NODE_GENERAL_TITLE
Particles
$#     sid       da1       da2       da3       da4    solver
        50       0.0       0.0       0.0       0.0MECH
$#  option        e1        e2        e3        e4        e5        e6        e7
PART              10        30         0         0         0         0         0
*SET_PART_LIST_TITLE
Parts
$#     sid       da1       da2       da3       da4    solver
        12       0.0       0.0       0.0       0.0MECH
$#    pid1      pid2      pid3      pid4      pid5      pid6      pid7      pid8
```

```
         1         2         3         0         0         0         0         0
*CONTACT_ERODING_SINGLE_SURFACE
$#     cid                                                                 title
$#    ssid      msid     sstyp     mstyp    sboxid    mboxid       spr       mpr
         0         0         5         0         0         0         0         2
$#      fs        fd        dc        vc       vdc    penchk        bt        dt
      0.01      0.01       0.0       0.0       5.0         0       0.01.00000E20
$#     sfs       sfm       sst       mst      sfst      sfmt       fsf       vsf
       1.0       1.0       0.0       0.0       1.0       1.0       1.0       1.0
$#    isym    erosop      iadj
         1         1         1
$#    soft    sofscl    lcidab    maxpar     sbopt     depth     bsort    frcfrq
         2       0.1         0     1.025       5.0        35         1         1
*SECTION_SOLID_TITLE
Target_Wall_Particles
$#   secid    elform       aet    unused    unused    unused    cohoff    unused
         1         1         0                                       0
*MAT_USER_DEFINED_MATERIAL_MODELS_TITLE
AA6070-T6
$#     mid        ro        mt       lmc       nhv    iortho     ibulk        ig
         22.70000E-9        47        48        20         0         1         2
$#   ivect     ifail    itherm    ihyper      ieos      lmca    unused    unused
         1         1         0         0         1         0
$        k         g         e        pr      beta       xsi    cp*rho     alpha
$#      p1        p2        p3        p4        p5        p6        p7        p8
  58333.33  26923.08   70000.0       0.3       0.0       0.9     2.4572.32000E-5
$    e0dot        tr        tm        t0
$#      p1        p2        p3        p4        p5        p6        p7        p8
1.00000E-4     293.0     893.0     293.0         0       0.0       0.0       0.0
$        a         b         n         c         m
$#      p1        p2        p3        p4        p5        p6        p7        p8
       350      2.64      0.05     0.103       1.0         0       0.0       0.0
$       q1        c1        q2        c2
$#      p1        p2        p3        p4        p5        p6        p7        p8
      30.1     185.9      72.8       7.7         0       0.0       0.0       0.0
$       dc        wc
$#      p1        p2        p3        p4        p5        p6        p7        p8
       1.0       115         0       0.0       0.0       0.0       0.0       0.0
$       tc      tauc       pco
$#      p1        p2        p3        p4        p5        p6        p7        p8
       804.01.00000E20     -1000         0       0.0       0.0       0.0       0.0
*EOS_LINEAR_POLYNOMIAL_TITLE
Linear
$#   eosid        c0        c1        c2        c3        c4        c5        c6
         2       0.0  58333.33       0.0       0.0       0.0       0.0       0.0
$#      e0        v0
       0.0       1.0
$*BOUNDARY_SPC_SET_ID
$$#     id                                                               heading
$        1Fix Z
$$#   nsid       cid      dofx      dofy      dofz     dofrx     dofry     dofrz
$         1         0         0         0         1         0         0         0
$$#     id                                                               heading
$        2Fix Y
$$#   nsid       cid      dofx      dofy      dofz     dofrx     dofry     dofrz
$         2         0         0         1         0         0         0         0
*SECTION_SOLID_TITLE
Ball
$#   secid    elform       aet    unused    unused    unused    cohoff    unused
         2         1         0                                       0
*MAT_USER_DEFINED_MATERIAL_MODELS_TITLE
Aluminium
```

```
$#     mid        ro        mt       lmc       nhv    iortho     ibulk        ig
         12.70000E-9        47        48        20         0         1         2
$#   ivect     ifail    itherm    ihyper      ieos      lmca    unused    unused
         1         1         0         0         1         0
$        k         g         e        pr      beta       xsi    cp*rho     alpha
$#      p1        p2        p3        p4        p5        p6        p7        p8
  58333.33  26923.08   70000.0       0.3       0.0       0.9     2.4572.32000E-5
$    e0dot        tr        tm        t0
$#      p1        p2        p3        p4        p5        p6        p7        p8
1.00000E-4     293.0     893.0     293.0         0       0.0       0.0       0.0
$        a         b         n         c         m
$#      p1        p2        p3        p4        p5        p6        p7        p8
     292.6       0.0       0.0      0.02       1.0         0       0.0       0.0
$       q1        c1        q2        c2
$#      p1        p2        p3        p4        p5        p6        p7        p8
       2.7    2160.7     707.6      8.94         0       0.0       0.0       0.0
$       dc        wc
$#      p1        p2        p3        p4        p5        p6        p7        p8
       1.0     278.0         0       0.0       0.0       0.0       0.0       0.0
$       tc      tauc       pco
$#      p1        p2        p3        p4        p5        p6        p7        p8
     804.01.00000E20     -1000         0       0.0       0.0       0.0       0.0
*PART
$#                                                                         title
Particle Target
$#     pid     secid       mid     eosid      hgid      grav    adpopt      tmid
        10         1         2         2         0         0         0         0
*PART
$#                                                                         title
Particle Wall
$#     pid     secid       mid     eosid      hgid      grav    adpopt      tmid
        20         1         2         2         0         0         0         0
*PART
$#                                                                         title
Particle Ball
$#     pid     secid       mid     eosid      hgid      grav    adpopt      tmid
        30         1         1         2         0         0         0         0
*INITIAL_VELOCITY_GENERATION
$#      id      styp     omega        vx        vy        vz     ivatn      icid
         3         2       0.0   6700000       0.0       0.0         0         0
$#      xc        yc        zc        nx        ny        nz     phase    irigid
       0.0       0.0       0.0       0.0       0.0       0.0         0         0
*DEFINE_ADAPTIVE_SOLID_TO_DES_ID
$#     did                                                               heading
         1
$#    ipid     itype        nq     ipdes     isdes       rsf    outdes     ibond
         1         0         1        10         1       1.0         0         0
*DEFINE_ADAPTIVE_SOLID_TO_DES_ID
         2
         2         0         1        20         2       1.0         0         0
*DEFINE_ADAPTIVE_SOLID_TO_DES_ID
         3
         3         0         1        30         2       1.0         0         0
*INCLUDE
$#                                                                      filename
mesh_ball_target.key
*INCLUDE
$#                                                                      filename
mesh_wall.key
*BOUNDARY_SPC_SET_ID
$#      id                                                               heading
         0Fix Z Target and Ball
```

```
$#     nsid       cid      dofx      dofy      dofz     dofrx     dofry     dofrz
          1         0         0         0         1         0         0         0
$#       id                                                             heading
          2Fix Z Wall
$#     nsid       cid      dofx      dofy      dofz     dofrx     dofry     dofrz
          4         0         0         0         1         0         0         0
$#       id                                                             heading
          3Fix XY Target Top
$#     nsid       cid      dofx      dofy      dofz     dofrx     dofry     dofrz
          2         0         1         1         0         0         0         0
$#       id                                                             heading
          4Fix XY Target Bottom
$#     nsid       cid      dofx      dofy      dofz     dofrx     dofry     dofrz
          3         0         1         1         0         0         0         0
$#       id                                                             heading
          5Fix XY Wall
$#     nsid       cid      dofx      dofy      dofz     dofrx     dofry     dofrz
          5         0         1         1         0         0         0         0
*END
```

Appendix C. lsdyna file

```
#!/bin/sh
#
# Create a new jobscript
#
rm -f jobdyna
cat >jobdyna <<EOF
#!/bin/bash

#SBATCH -J lsdyna
#SBATCH -o lsdyna.%j.out
#SBATCH -N 1
//  4 is the amount of CPUs reserved on the cluster
#SBATCH -n 4
#SBATCH --ntasks-per-core=1

export LSTC_LICENSE=network
export LSTC_LICENSE_SERVER=129.241.89.45
export LSTC_D3DUMP=off
export LSTC_BINARY=32ieee
export LSOPT_HOST=172.16.0.1
export LSOPT_PORT=$LSOPT_PORT

source /opt/intel/compilers_and_libraries/linux/bin/compilervars.sh intel64
source /opt/intel/impi/2018.4.274/bin64/mpivars.sh intel64

export I_MPI_ADJUST_BCAST=1
export I_MPI_ADJUST_ALLREDUCE=5
export I_MPI_PIN_CELL=core
export I_MPI_PIN_PROCESSOR_EXCLUDE_LIST=48-95
export I_MPI_FABRICS=shm:ofa
export I_MPI_PRINT_VERSION=on
export I_MPI_DEBUG=4
export I_MPI_DEBUG_OUTPUT=jobstat

$LSOPT/wrapper mpirun -bootstrap slurm -prefork -n \$SLURM_NTASKS /opt/lstc/mppdyna-eos -v2 $1
EOF
#
#  submit the jobscript to the slurm que
#
sbatch jobdyna
```