

## A Methodology for Applying a Moment to a Solid Finite Element Model

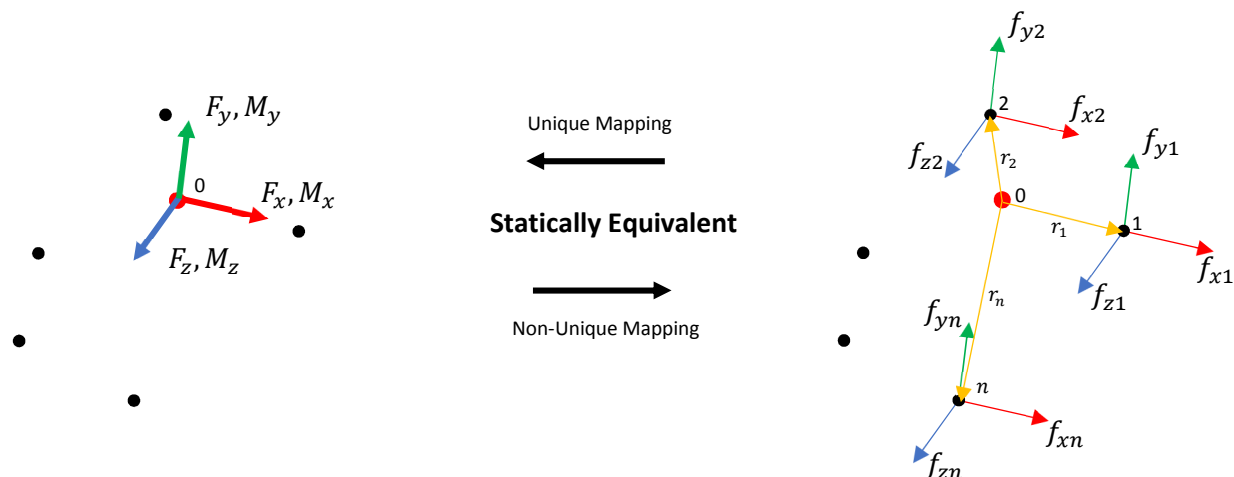
### Introduction

There are occasions where one wishes to apply 'point' moments to a solid finite element model. This is not directly possible since solid elements do not have rotational degrees of freedom. It is, however, possible by applying appropriate nodal forces to a small cloud of nodes surrounding the node of interest, i.e., the one where the point loads are to be applied, in a statically equivalent manner. In three dimensions the number of nodal force components is three times the number of nodes in the cloud of nodes and as soon as the number of nodes exceeds two, there is an infinity of possible statically admissible nodal forces. Additional constraints could be included if the nodal forces are forced to be distributed in a 'consistent' manner, i.e., in a work equivalent manner with the finite element formulation. We will, however, choose not to do this on the assumption, to be verified by the practising engineer, that the disturbance or pollution in the stress field local to the node cloud and resulting from the application of non-consistent loads will diffuse away, through St Venant's principle, so as not to influence the stresses in the region of interest – which is assumed to be 'remote' from the applied load.

The methodology described in this technical note has been developed into a small software application which is called Distributed Point Load, or DPL for short and this software is freely available from RMA's website – <https://www.ramsay-maunder.co.uk/software/>. We have used the 'point load' to indicate that the software is equally applicable to point forces and point moments.

### Methodology

The statically equivalent mapping process between point load and cloud nodal forces is illustrated in Figure 1 where it is indicated that whereas the mapping from cloud nodal forces to point loads is unique, the inverse mapping will generally be non-unique.



**Figure 1:** Schematic of transformation between point loads and cloud nodal forces

It is a simple matter to sum up the cloud nodal forces to determine the point forces and when weighted by the appropriate coordinate distances, to determine the point moments. In Eq. (1), the matrix  $[S]$  sums the contributions of the nodal forces  $\{f\}$  over the  $i = 1, 2, \dots, n$  nodes in the cloud that surround the node (node number 0) where the load vector  $\{F\}$  is to be applied.

$$[S]\{f\} = \{F\} \quad (1)$$

For the normal situation, where the number of nodal forces exceeds the six applied point loads, Eq. (1) can be expanded as shown in Eq. (2).

$$\begin{bmatrix} 1/r_1 & 1/r_2 & 1/r_n & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/r_1 & 1/r_2 & 1/r_n & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1/r_1 & 1/r_2 & 1/r_n \\ 0 & 0 & \dots & 0 & 0 & 0 & 0 & 0 & 0 \\ +z_1/r_1 & +z_2/r_2 & +z_n/r_n & -z_1/r_1 & -z_2/r_2 & \dots & -z_n/r_n & +y_1/r_1 & +y_2/r_2 & \dots & +y_n/r_n \\ -y_1/r_1 & -y_2/r_2 & -y_n/r_n & 0 & 0 & 0 & -x_1/r_1 & -x_2/r_2 & -x_n/r_n \\ +x_1/r_1 & +x_2/r_2 & +x_n/r_n & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} f_{x1} \\ f_{x2} \\ \vdots \\ f_{xn} \\ f_{y1} \\ f_{y2} \\ \vdots \\ f_{yn} \\ f_{z1} \\ f_{z2} \\ \vdots \\ f_{zn} \end{Bmatrix} = \begin{Bmatrix} (1-h)F_x \\ (1-h)F_y \\ (1-h)F_z \\ M_x \\ M_y \\ M_z \end{Bmatrix} \quad (2)$$

The parameter  $h$  in Eq. (2), is used to distribute some of the point forces to node number 0. If  $h = 0$  then the nodal forces at node number 0 are zero whereas if  $h = 1$  the total force is applied. The value of this parameter chosen for use in DPL is  $h = 1/2$ , i.e., it is 'hard' coded into the software. The nodal forces at node number 0 are then uniquely defined as shown in Eq. (3).

$$\begin{Bmatrix} f_{x0} \\ f_{y0} \\ f_{z0} \end{Bmatrix} = h \begin{Bmatrix} F_x \\ F_y \\ F_z \end{Bmatrix} \quad (3)$$

In Eq. (2) we have applied a radial weighting by dividing by the radius, i.e., the distance between node  $i$  and node 0. This weighting localises the larger nodal forces in the cloud to be adjacent to the point load. DPL, allows the user to decide whether or not to apply this radial weighting. If the user does not want this weighting then Eq. (2) is modified such that  $r_i = \text{unity}$  for  $1 \leq i \leq n$ . Note, however, that the stress distribution remote from the applied load is independent of whether or not radial weighting is applied and, so, provided the region of interest is 'sufficiently remote' from the applied load then the choice of weighting is irrelevant. It should be noted regarding the radial weighting that there is a remote possibility that the weighting could lead to ill-conditioning in the system of equations as a cloud node arbitrarily close to node number 0 would have a very large weighting compared to other more distant nodes. This is an unlikely scenario in practice but the user should always check the magnitude of the residual force and moment that is output by DPL to ensure that the solution provided is viable and not polluted by such ill-conditioning. The user could, of course, choose not to use radial weighting at all and in the example shown later it will be seen that it makes very little practical difference to the solution.

The dimensions of  $[S]$  are  $6 \times 3n$  and it has maximum rank of 6. In practice, and depending on the coordinates of the nodes, it may have a lower value, i.e.,  $\text{rank}[S] \leq 6$ . In contrast, the rank of the transpose of  $[S]$  will generally be full, i.e.,  $\text{rank}[S]^T = 6$ .

The system of equations, Eq. (1), is generally underdetermined, i.e.,  $n > 2$  and there is an infinity of possible solutions to the problem. A solution can, however, be determined through a least squares approach, i.e., by finding the vector  $\{f\}$  that minimises the Euclidian norm of the vector  $\{F\} - [S]\{f\}$ . This leads to the set of equations given in Eq. (4) which need to be solved for  $\{f\}$ .

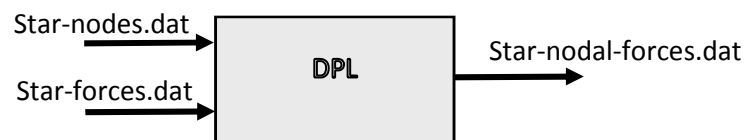
$$[S]^T[S]\{f\} = [S]^T\{F\} \quad (4)$$

If  $\text{rank}[S] < 6$  then  $[S]^T[S]$  is singular and there is no unique least squares solution for  $\{f\}$  using conventional decomposition techniques. It is, however, still possible to obtain a solution to the problem using singular value decomposition of  $[S]^T[S]$  or the pseudoinverse of Moore-Penrose. In the software this approach will be adopted irrespective of whether  $[S]$  is rank deficient or not.

Once the solution is obtained then the nodal forces are recovered directly from the vector  $\{f\}$  which, if radial weighting is chosen, needs to be divided by the appropriate nodal radius. The forces at node 0 are determined from Eq. (3). The solution is checked by evaluating the Euclidian norm of the vector  $\{F\} - [S]\{f\}$  for the forces and moments. These values are reported by DPL so that the user might be assured of the statical equivalence of the solution vector offered by the software.

## Software Application

A software application has been written to produce a statically equivalent set of nodal forces for a cloud of nodes surrounding the loaded node. The application is called DPL (Distributed Point Load). It takes input from two files and produces a single output file as illustrated in Figure 2.



**Figure 2:** DPL mnemonic

We have used the term ‘star’ here to conjure up the idea of nodes surrounding a central node. The format of the input files is described below.

**star-nodes.dat** – this file contains a list of node numbers and coordinate as follows:

Node Number	X	Y	Z	U	V	W	Other
72	0.50000	2.0000	0.0000	0.00	0.00	0.00	
70	0.60000	2.0000	0.0000	0.00	0.00	0.00	
71	0.55000	2.0000	0.0000	0.00	0.00	0.00	
73	0.45000	2.0000	0.0000	0.00	0.00	0.00	
74	0.40000	2.0000	0.0000	0.00	0.00	0.00	

**The values are space or tab delimited and the only stipulation is that the loaded node (node number 0) should be the first node listed.** Note that as only the first four numbers on each line are read by DPL, additional data to the right of these can be included – this is useful for example when outputting nodal coordinates directly from ANSYS.

**star-forces.dat** – this file contains the forces and moments applied at the loaded node, i.e., the entries in the  $\{F\}$  vector as rows of six numbers as shown below:

Force/Moment	Force/Moment	Force/Moment	Force/Moment	Force/Moment	Force/Moment	Other
0D0	0D0	0D0	0D0	0D0	1D0	
0D0	1D0	0D0	0D0	0D0	0D0	

The values are space or tab delimited and solution vectors of nodal forces are produced for each row of force/moments supplied in this file.

**Star-Nodal-Forces.dat** - Although the application is not software specific, the output is written for use with ANSYS and the forces at the nodes are written in a format suitable for reading directly into this FE software. For each  $\{F\}$  vector supplied, the nodal forces are listed and separated with a load case header. The output corresponding to the input files specified above is as follows:

Star-Nodal-Forces.dat - Notepad

```
File Edit Format View Help
LOAD CASE NUMBER 1, Radial Weighting = F
F,70,FY, 0.400000E+01
F,71,FY, 0.200000E+01
F,73,FY,-0.200000E+01
F,74,FY,-0.400000E+01
LOAD CASE NUMBER 2, Radial Weighting = F
F,72,FY, 0.500000E+00
F,70,FY, 0.125000E+00
F,71,FY, 0.125000E+00
F,73,FY, 0.125000E+00
F,74,FY, 0.125000E+00
```

Ln 12, Co

Star-Nodal-Forces.dat - Notepad

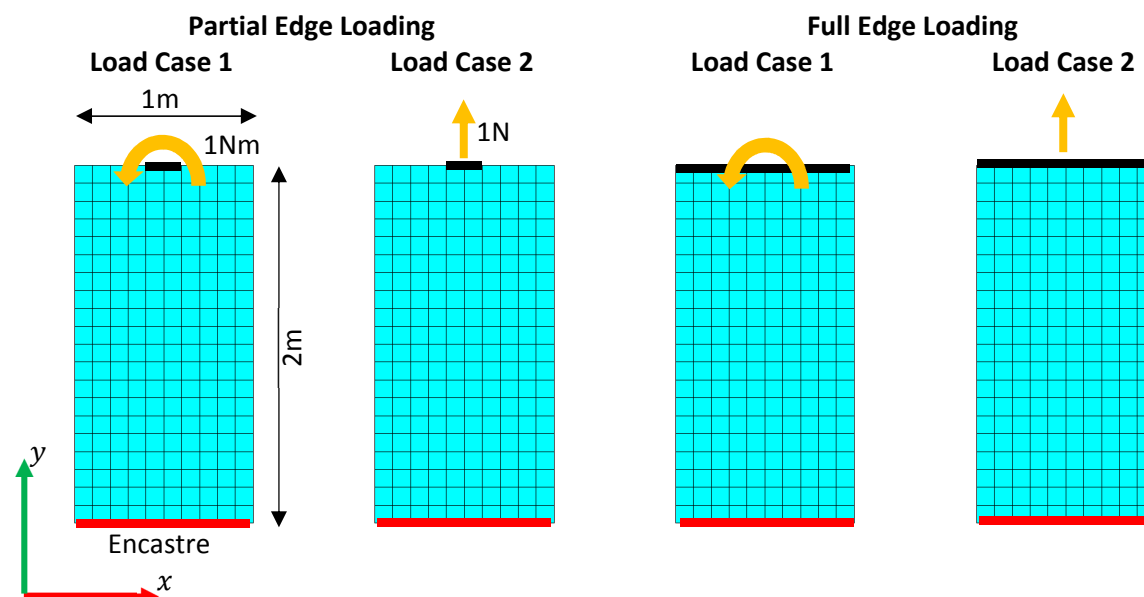
```
File Edit Format View Help
LOAD CASE NUMBER 1, Radial Weighting = T
F,70,FY, 0.250000E+01
F,71,FY, 0.500000E+01
F,73,FY,-0.500000E+01
F,74,FY,-0.250000E+01
LOAD CASE NUMBER 2, Radial Weighting = T
F,72,FY, 0.500000E+00
F,70,FY, 0.500000E-01
F,71,FY, 0.200000E+00
F,73,FY, 0.200000E+00
F,74,FY, 0.500000E-01
```

Ln 1, Col

### Software Verification

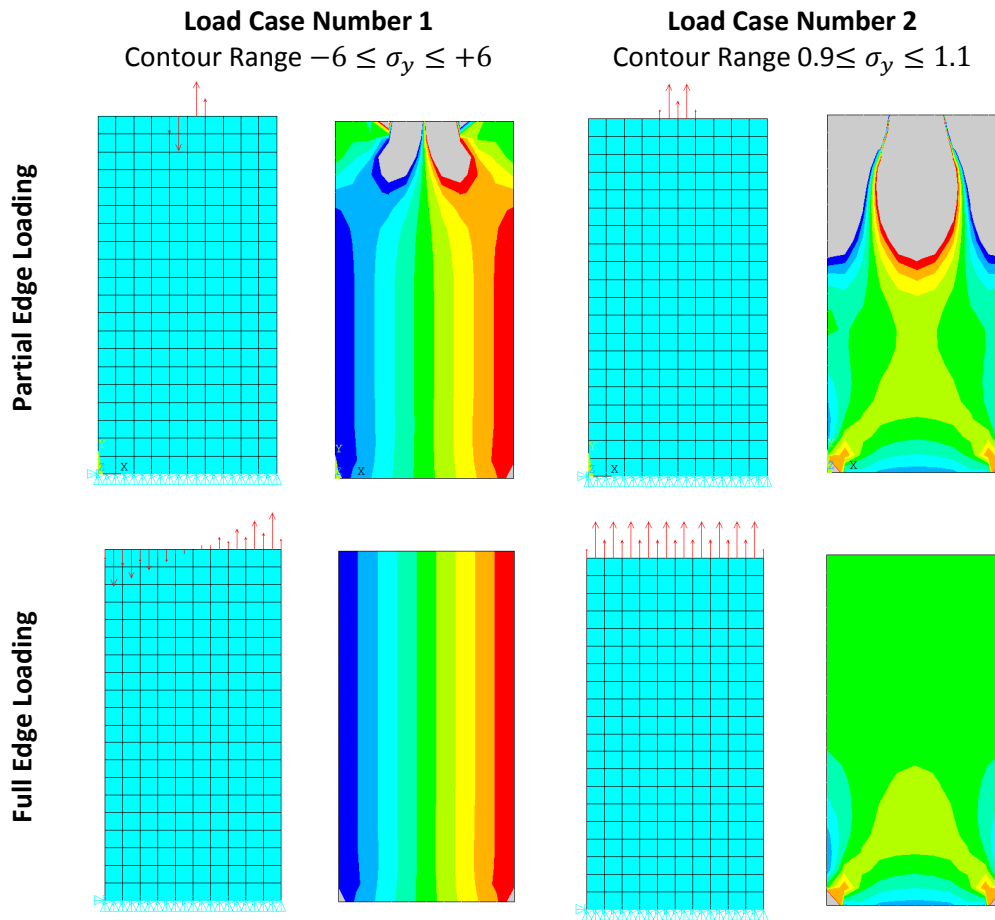
Although the software was developed for three-dimensional continua, we will choose, for ease of explanation and understanding, to consider a two-dimensional plane stress continuum example. The model lies in the  $x, y$  plane ( $z = 0$ ) and has translational freedoms in the two coordinate directions. As such, only three components of the force vector  $\{F\}$  are non-zero, i.e.,  $F_x$ ,  $F_y$  and  $M_z$ .

The example considered is a built-in or encastre beam member of thickness 1m and length 2m. At the free end of the beam we wish to apply unit loads and will consider two load cases,  $F_y=1\text{N}$  and  $M_z=1\text{Nm}$ . We will look at cases where the load is distributed over a 0.2m central region and when it is distributed over the complete 1m thickness of the beam. The FE mesh is shown in Figure 3 with the four cases identified. The mesh comprises 0.1m square higher-order (eight-noded) elements.



**Figure 3:** Four finite element models considered in this section

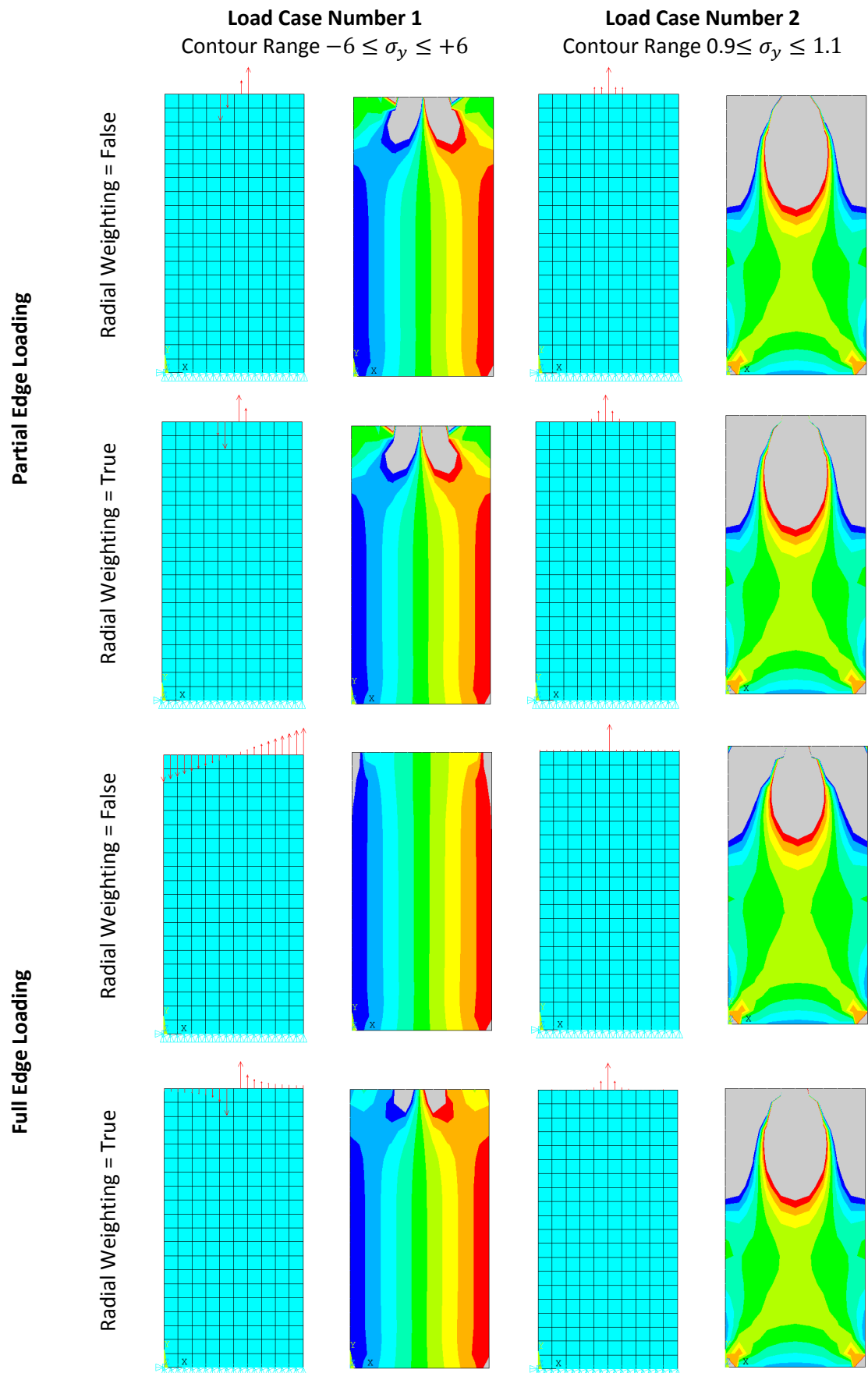
To provide some reference solutions to the problems, we will apply the nodal forces consistently to the loaded edge. Contours of the direct stress in the y direction ( $\sigma_y$ ) are shown in Figure 4.



**Figure 4:** Contours of  $\sigma_y$  from consistently applied loads

The best point for starting this discussion is with the full edge loading. When applied consistently the expected stress fields, linear for load case 1 and constant for load case 2, are immediately recovered under the load. The encastre support leads to reaction forces that are not the same as the consistent loads and so the stress field is disturbed adjacent to the encastre edge. When partial edge loading is applied, even consistently, the stresses under the load are disturbed by the fact that the load wants to diffuse to the complete section. The length over which the stresses are disturbed is about one half of the thickness and the thickness, respectively, for load cases 1 and 2 – St Venant’s principle in operation!

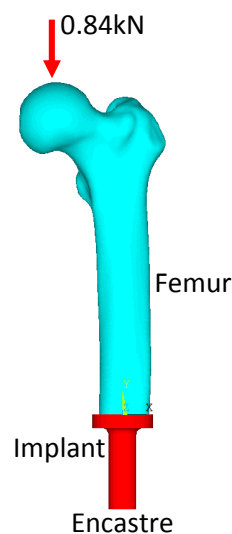
The contours of  $\sigma_y$  obtained by loads (non-consistent) from DPL are shown in Figure 5 for both load cases and for radially weighted and non-weighted distributions. When the load is applied to the complete edge, there remains a disturbance in the stress directly under the load and this is directly caused by the fact that the loads were not applied in a consistent manner, c.f., the second row of Figure 4. When partial edge loading is adopted then significant disturbances in the stress are seen irrespective of whether the loads are applied consistently or not. This indicates that the dominating factor in terms of disturbance in the stress field is that due to load diffusion from the partially loaded edge to the full section width and not the precise nature of the way in which the load was applied.



**Figure 5:** Contours of  $S_y$  from DPL applied loads

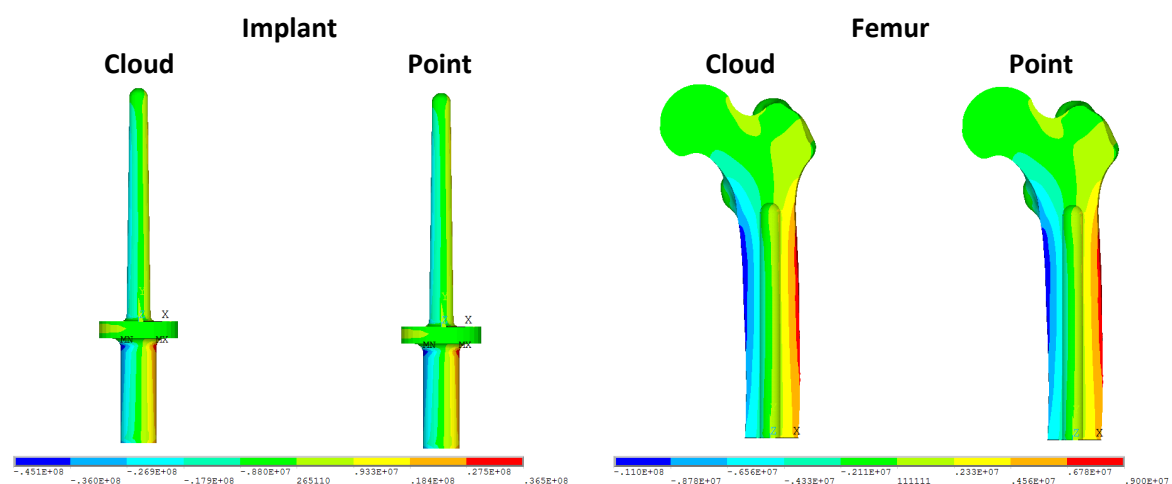
## Practical Example

The software application, DPL, has been used in research in the biomedical engineering group at University College London who focus on joint and limb replacement and salvage. Skeletally anchored amputation prostheses (SAAP) have been clinically trialled and further work is being conducted – of key interest here is the stress field at the bone transaction face as this is key to SAAP survival. These stresses are obtained by finite element analysis with the joint reaction force being applied to the femoral head. The load includes forces and moments and these change in a cyclic manner as the patient walks and so a simple method of applying these forces and moment was required. The finite element model of the transected femur and implant are shown in Figure 6. A point load is applied to the femoral head and the bottom of the implant is fully restrained.



**Figure 6:** Finite element model of truncated femur and metallic implant

DPL was used to distribute the applied forces and moments to a small cloud of nodes on the femoral head. To verify the approach, DPL was used to produce distributed forces that equilibrate with a point force. The reactions at the support were then compared and the stresses in the region of interest. Contours of the vertical direct stress are shown in Figure 7 for the two components and with the point and cloud loading and are seen, as expected, to be essentially identical.



**Figure 7:** Finite element stresses in implant and femur for cloud and point loading

Whilst this practical example has only shown results for a point forces, the software will, in due course, be used to apply both forces and moments as derived from a rigid-body dynamic analysis of the gait cycle.

## Closure

The practising engineer may sometimes have a need to apply point moments to his/her continuum finite element models. Such models do not explicitly include rotational degrees of freedom and therefore some indirect approach is required. This may be done through kinematic coupling or, as shown here, by applying a statically equivalent set of nodal forces to a cloud of nodes local to the point where the moment is applied. In general, and even with a small number of nodes in the cloud, the nodal forces are not uniquely determined from the applied load, i.e., there is an infinity of ways in which cloud nodal forces may be determined to equilibrate with the applied load. One way to resolve this problem is to adopt a least-squares fit. However, depending on the location of the cloud nodes, the system of equations may not only be underdetermined but also rank deficient. Conventional matrix decomposition techniques then fail. In this technical note singular value decomposition is used to overcome this potential hurdle and provides a robust approach to solving the least squares problem irrespective of any rank deficiency that may be present.

The method has been coded up as a small Fortran software application called Distributed Point Load (DPL). DPL was verified using a plane stress beam problem and the stresses resulting from the application of cloud nodal forces were examined and compared with those where the nodal forces were generated in a manner consistent with the finite element method. It was interesting to observe that when the loads were distributed over only a part of the edge then the disturbance in the stress due to the application of inconsistent nodal forces was swamped by that due to load diffusion from the partial to the full section. It was also noted that different loading forms require different distances for full load diffusion; in the case considered, the pure axial load required about twice the distance than for the bending load.

The practical example of the femur/implant model demonstrated that DPL functions well for three-dimensional models and offers a robust and rapid approach to applying moments to solid models.

## Acknowledgement

The author is grateful to Kirstin Ahmed, PhD student in Biomedical Engineering at University College London, for evaluating the software application DPL and providing the figures used in the Practical Example section of this technical note.

## Appendix 1: Software Download & Installation

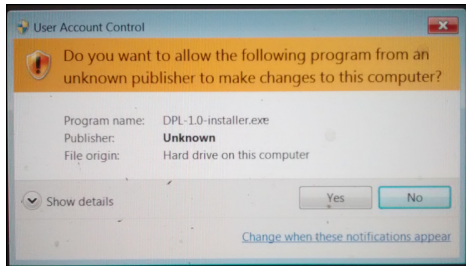
The Windows installer for the free software application DPL can be downloaded from the RMA website at:

<https://www.ramsay-maunders.co.uk/software/distributed-point-load-dpl/>

### Software Installation

After downloading the DPL installer file, you should run the installer by double clicking on the file icon. You will then receive the following warning that the software Publisher is Unknown to your computer.



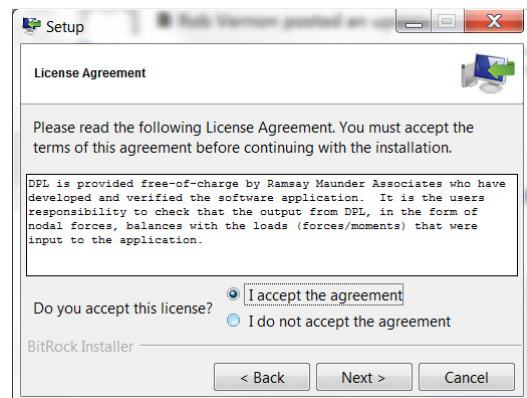
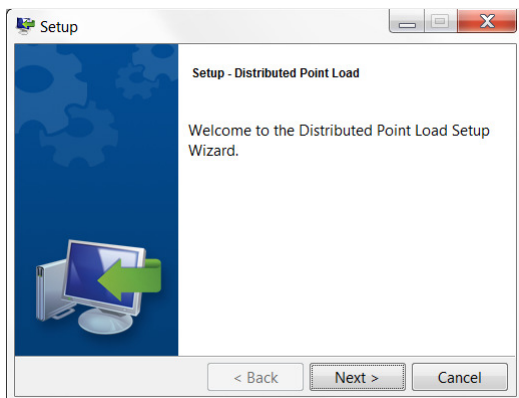


This warning arises because RMA have not obtained a Software Publisher Certificate (SPC) from Microsoft:

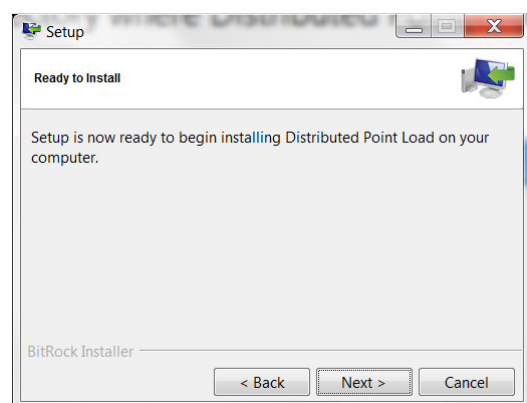
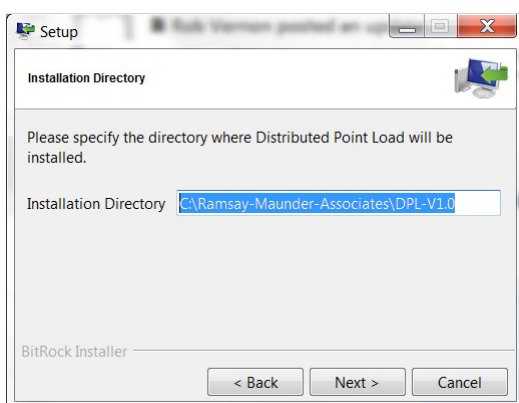
<https://docs.microsoft.com/en-us/windows-hardware/drivers/install/obtaining-a-software-publisher-certificate--spc->

Provided you believe RMA to be a reputable supplier of software then you should answer 'yes' to this question.

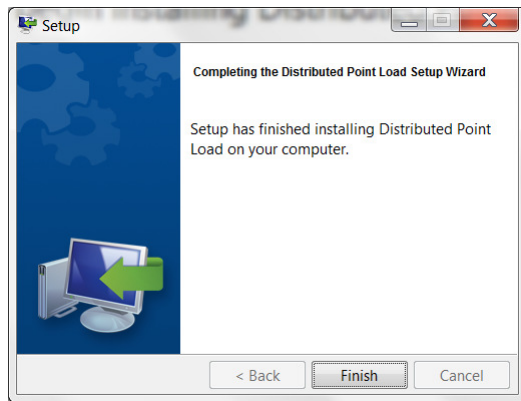
You will then see the following dialogs. The second dialog outlines the license agreement between RMA and yourself which must be accepted if you wish to install the software.



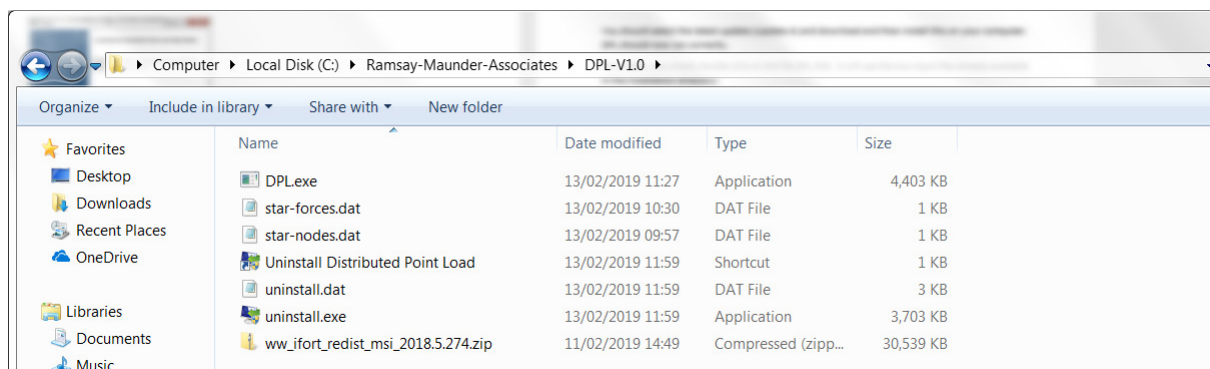
The next dialog tells you where the installer intends to place the software on your computer. A default directory is specified but this may be changed to suit your purposes.



The final dialog simply informs you that the software has been installed.



The following files are added to your selected installation directory.

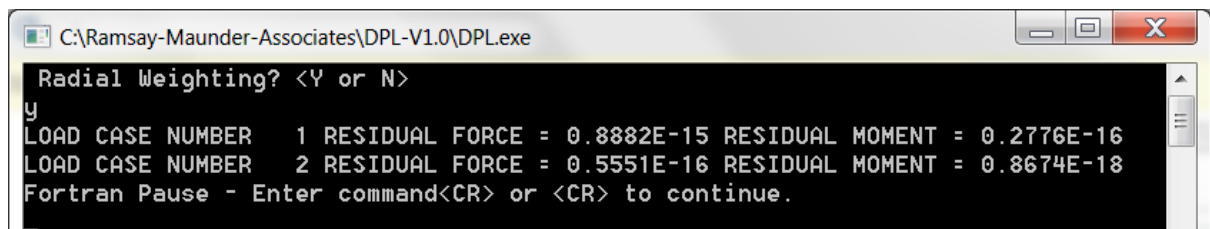


The application DPL may not run at this stage unless you have the appropriate Intel libraries already installed on your computer. If you haven't then these may be obtained by unpacking the file:

ww-ifort-redist-msi-2018.5.274.zip file

and running the resulting msi file.

To run DPL, you simply double click on the file DPL.EXE. It will use the two input files already available in the installation directory and once the user has selected whether or not to apply radial weighting the residuals are reported for each load case as shown below:



The output file from DPL is written and to check that DPL is running as intended the contents of this file can be compared with those presented in the section Software Application earlier in this technical note.