Applying Pressure Loads in Node Network Models

Lawrence Smith | *lasm4254@colorado.edu*

I. Basic Method

Many soft robotic actuators use fluid pressure to transmit mechanical power or achieve desired displacements. In simulating these components, the ability to apply uniform pressure boundary conditions is indispensable. A uniform pressure load is a specific type of surface traction with the following properties:

- it acts everywhere normal (perpendicular) to the surface
- it has a constant per-unit-area value

Node network models discretize a body into point masses which are connected by links that imitate the stiffness and viscous properties of continuum matter. Force contributions are summed at each node from the surrounding connections to other nodes. These forces produce accelerations at each node, which are numerically integrated forward in time.

In order to apply a boundary condition which is defined as *distributed evenly* across a surface, we need to resolve it to forces that act on the nodes of the surface. At every evaluation of nodal forces, a model should index through each face of the model which is exposed to the applied pressure and perform the calculaiton described below.

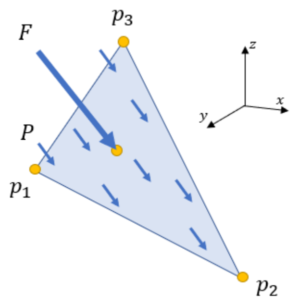

Fig. 1: Triangular area bounded by three points in space, with uniform distributed pressure load P resolved to a total force vector $\mathbf{F}$ acting at the centroid of the triangle

Consider a triangluar surface defined by three points in space p_1, p_2, p_3, with an applied pressure load of magnitude P. The total force acting on the face is

$$\mathbf{F} = P * A * \mathbf{n} \tag{1}$$

where A is the triangle area and $\mathbf{n}$ is the surface normal.

This total force is distributed among the three nodes of the triangle - conveniently for our assumptions, it is distributed exactly evenly between the three nodes, regardless of the shape of the triangle. This is because the resultant force vector F always passes through the centroid of the triangle.

$$f_1 = f_2 = f_3 = f_i = \frac{1}{3}\mathbf{F} \tag{2}$$

The area of the triangle can be computed by taking half the cross product of two vectors formed by the three points:

$$A = |\frac{cross(p_2 - p_1, p_3 - p_1)}{2}| \tag{3}$$

The normal vector to the surface can be found by taking the same cross product, then dividing by the norm to produce a vector with unit magnitude:

$$\mathbf{n} = \frac{cross(p_2 - p_1, p_3 - p_1)}{|cross(p_2 - p_1, p_3 - p_1)|} \tag{4}$$

When substituting expressions 1, 3, and 4 into 2, a convenient cancellation occurs, leaving:

$$f_i = \frac{P}{6} cross(p_2 - p_1, p_3 - p_1) \tag{5}$$

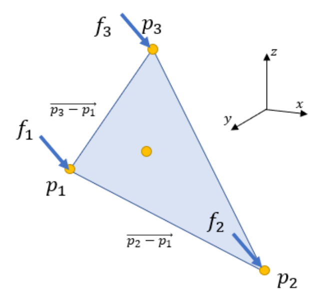

Fig. 2: Resolving the pressure load to nodal forces f_i

The execution speed of this calculation can be improved by carrying out the cross product operation and expressing f_i as an algebraic function of p_1, p_2, p_3:

$$f_i = \frac{P}{6}\begin{bmatrix} (p_{1y} - p_{3y})(p_{1z} - p_{2z}) - (p_{1y} - p_{2y})(p_{1z} - p_{3z}) \\ (p_{1x} - p_{2x})(p_{1z} - p_{3z}) - (p_{1x} - p_{3x})(p_{1z} - p_{2z}) \\ (p_{1x} - p_{3x})(p_{1y} - p_{2y}) - (p_{1x} - p_{2x})(p_{1y} - p_{3y}) \end{bmatrix} \tag{6}$$

If a body is represented by hexahedral unit cells in a node network model, the faces exposed to pressure loads will be quadrilaterals. The exact method described above can still be used to apply pressure loads however - each face is simply divided into two triangles first.

II. Notes of Caution

The unit normal $\mathbf{n}$ is signed, and if the ordering of p_1, p_2, p_3 in equation 6 is not consistent between faces, pressure forces will act in unpredictable directions.

In cases where spatially uniform pressure cannot be assumed, this calculation does not hold. The total force vector will $\mathbf{F}$ will not intersect the centroid of the triangle, (a surface integral is required to determine its position), and the nodal forces f_i are not equal.

If the body in question undergoes finite deformation, the direction and/or magnitude of the load may change during the analysis (force nonlinearity). This captures physical buckling and snap-through behaviors, but can lead to instabilities if appropriate solution methods (e.g. Riks) are not implemented.