This is a brief explanation for "meshgen3.m". This is not official report, Please not use without using the program

These figures (Figure1,2,6,7) are results when I input following data. I will explain about procedure with this results. Following table (Table:1) is described the input data for "meshgen3.m".

**Step 0**

Firstly describe the demanded shape from the input data. Bule broken line means edge of the demanded rectangle. Red broken line means extension direction. So the red line should suit element diagonal.
<table>
<thead>
<tr>
<th>Valuable name</th>
<th>input number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ox</td>
<td>-0.01m</td>
<td>x position for left bottom apex of Rectangle (blue big dot on Figure 1)</td>
</tr>
<tr>
<td>Oy</td>
<td>-0.01m</td>
<td>y position for left bottom apex of Rectangle (blue big dot on Figure 1)</td>
</tr>
<tr>
<td>Length</td>
<td>0.02m</td>
<td>Length of the Rectangle</td>
</tr>
<tr>
<td>Height</td>
<td>0.02m</td>
<td>Height of the Rectangle</td>
</tr>
<tr>
<td>Element</td>
<td>0.001m</td>
<td>square-shaped element(before extension)</td>
</tr>
<tr>
<td>theta</td>
<td>$45^\circ$</td>
<td>Angle between X axis and Element-diagonal</td>
</tr>
<tr>
<td>ExtensionCentrePos</td>
<td>[0.0 , 0.0 ]</td>
<td>Position of the Maximum extended element</td>
</tr>
<tr>
<td>MaxDiagonal</td>
<td>0.0018m</td>
<td>Diagonal length of the Maximum extended element</td>
</tr>
<tr>
<td>Eccentricity</td>
<td>0.98</td>
<td>Eccentricity of the ellipse that is used for deciding extended ratio for each element (yellow points on Figure 8)</td>
</tr>
</tbody>
</table>

Table 1: input data

**Step 1**

Move the ExtensionCentrePos to origin and rotate theta degree clockwise. But moving to origin is needless in this case because I choose origin for ExtensionCentrePos. You can easily understand that the ExtensionCentrePos will be moved to origin when you input different number of ExtensionCentrePos. That procedure changes the coordinate for make the node-relationship simply. The red line should be fitted with X axis. So the Extended Base Nodes on the extension (yellow dots on Figure8), that will make other nodes after, move on the X axis and other nodes will keep symmetrical position about Y axis.
Step 2

Next, we have to decide the size of the net. If we choose too large net to cover the rectangle, the calculation time becomes longer and that is far from realistic situation.

In order that, we have to find the proper size of the net. To find the size of the net, we have to consider the border of the deformed(extended) net.

See Figure 3, the extended net has the different length of the diagonal(Blue line).

Hence the half number of the diagonal is $n$ and these diagonals are named $a_1, a_2, \cdots, a_n$. The nodes position on the X axis are following.

$$(X, Y) = (-2 \sum_{i=1}^{n} a_i, 0), (-2 \sum_{i=1}^{n-1} a_i, 0), \cdots, (-2a_1, 0), (0, 0), (2a_1, 0), \cdots, (2 \sum_{i=1}^{n-1} a_i, 0), (2 \sum_{i=1}^{n} a_i, 0)$$

By the way, thinking about the Element length, the deformed net keeps the relation between each truss. Actually, the truss that have same color in
Figure 3: ellipse algorithm 2

Figure 4 should keep parallelism. So, if we have the progression $a_n$, we can find the positional relation between all nodes (Figure 4).

Then we have to make the (decreasing) progression $a_n$. To make the progression $a_n$, we will use an ellipse equation. Ellipse equation is following.

\[
\frac{x^2}{A^2} + \frac{y^2}{C^2} = 1 \quad (1)
\]
\[
e = \sqrt{1 - \frac{C^2}{A^2}} \quad (2)
\]

Hence, $2A$ is called major-axis, $2C$ is called minor-axis, and e is called eccentricity.

See Figure 5.
At first, put decreasing dots at even intervals (red dots on figure 5). Then, we can calculate the x position on the ellipse curve (blue circle on Figure 5). From eq. (1), the equation is following.

\[ a_n = A \sqrt{1 - \frac{c_n^2}{B^2}} \]  
\[ = A \sqrt{1 - \frac{c_n^2}{A^2(1-e^2)}} \]

If we decide A as ”MaxDiagonal” in Table 1, \( a_n \) will be a decreasing progression.

Therefor, from the node that is left end of the extension line (the position is \((-2 \sum_{i=1}^{n} a_i, 0)\)), the border of the deformed net can be calculated.

Because the Size of the border is depend on the number of the element, check iteratively the element number that deformed net will cover the rectangle.

See Figure 6, the blue line is the border with element number \( n = 3 \). The green line is the border with \( n = 5 \). And red line is border with \( n = 14 \). When \( n \) reache to 14, the border covers the Rectangle for the first time. (Red line cover the bule broken line)
Step 3

For the next step, we need remove the node that lie outside of the rectangle. Firstly find the node which lie outside the rectangle. (Bule area on the Figure 7) But it's not perfect for membrane-element and truss-element. Cause of some elements crossing the rectangle border, we need to remove the imperfect element. (I choose the way that checking the each element which is connected to the outside node or not.)
Step 4

Finally, we have to return the initial rectangle position. That is moving origin to the ExtensionCentrePos and rotate theta degree counterclockwise. This is the result for the demanded rectangle. (Figure 8)
Figure 7: Figure 1 (right-bottom)

Figure 8: Figure 2: generated mesh (Rectangle)