Methods for Generating Rough Surfaces in ANSYS

M. Kathryn Thompson
Mechanical Engineering Dept, MIT

Abstract

Many engineering fields, including tribology and MEMS, seek to use or alter the behavior of the system at the surface or at the interface between surfaces. Thus, the topography of the surface is of great interest and must be included in finite element simulations. Four methods for generating normally distributed rough surfaces using APDL are proposed. The resulting geometry, meshes, and solutions are then compared to demonstrate the strengths and weaknesses of the various methods.

Introduction

For the analysis of structures, users have historically chosen to model the gross geometry of the system and ignore the details to simplify the model and reduce computation times. Surface effects are generally ignored in the analysis, partly because of the difficulty in creating a reasonable model of the surface and partly because of the assumption that surface effects are less important. However, surface effects caused by surface topography are of major concern in some engineering applications and may become critical when contact is involved.

In tribological applications, real surfaces are subjected to a combination of normal loading and shear loading over long distances. This causes the asperities on the surfaces to interact in a non-uniform way, exerting forces on each other and leading to frictional forces and wear. In fluid sealing applications, the asperities must be compressed enough to close all potential leakage paths across the sealing surface. In MEMS applications, the surface finish or surface damage may be on the same order of magnitude as the desired geometry and may degrade the performance of the devices. In all of these applications, it is desirable to model the systems with the expected (or measured) surface topography to determine the impact (frictional losses, wear rate, sealing potential, failure, etc.) of surface roughness on the system itself. The key to predicting the impact of surface topography is the ability for the asperities in the system to deform in response to an applied load and for the program to solve for those asperity deformations with as little solution error as possible.

In 1996, Tian et al. argued that “for 3D rough surface contact problems with many asperities of arbitrary shape, [the] requirement of large number[s] of mesh elements makes the finite element approach unfeasible.” However, computational capabilities have become cheaper and more powerful every year, and it is becoming feasible to create and solve the meshes necessary to model arbitrarily shaped surface asperities for small to mid-sized geometries.

A variety of methods for creating rough surfaces in finite element models have been proposed. Thompson et al. created three-dimensional surfaces with a normal Gaussian distribution by moving the surface nodes. Hyun et al. created three dimensional self-
affine fractal surfaces by moving all of the nodes in the model “by a fraction of the local height that depends on the initial height \( z(0) \) of the node above the bottom of the elastic cube.”\(^3\) Bhushan, et al. created single asperities (hemispherical point, hemispherical scratch and groove (triangular) scratch) by modifying the solid model.\(^4\) However, there has been little or no discussion about the relative merit of the various methods based on the quality of solution that each produces or alternate methods for creating surface roughness.

This paper presents four methods of generating normally distributed rough surfaces using the ANSYS Parametric Design Language (APDL): (1) moving all nodes, (2) moving selected nodes by location, (3) creating key points and (4) creating shell primitives. The resulting geometry, meshes, and solutions are then compared to demonstrate the strengths and weaknesses of the various methods and recommendations are made based on the results.

**Parameterization of Surface Topography**

For this paper, ANSYS commands will be shown in capital letters. APDL commands are ANSYS commands prefaced by a “*.” Parameter names are shown in bold capital letters.

**Parameterization of General Surfaces**

Rough surfaces are composed of a large number of surface asperities which can be thought of as local deviations from the nominal or average surface height. Real surfaces may exhibit roughness (high frequency asperities), waviness (medium frequency asperities) and surface form (the general shape of the surface neglecting roughness and waviness). Surfaces may also exhibit lay, which is a directional characteristic of the surface “such as a parallel, circular, radial or multidirectional pattern.”\(^5\)

Consider a simple surface in Cartesian coordinates that is characterized by a single modal distribution of roughness and does not exhibit waviness, surface form or lay. The rough surface can then be fully described by six parameters. Two parameters should describe the asperity density, or the amount of space between asperities: one in \( x \) and another in \( y \). Three parameters should describe the asperity size: length (\( x \)), width (\( y \)) and height or depth (\( \pm z \)). The last parameter should describe the asperity shape and define any coupling between the first five parameters if appropriate. These parameters may have values that are constants, functions, or statistical distributions.

Surface roughness and other surface topography may be a function of more than one roughness (waviness, form or lay) distribution\(^6\). Bimodal or multimodal systems would require up to six more parameters for each additional distribution mode, whether the additional mode is an additional roughness distribution or a contribution from waviness or surface form. The affect of lay can be incorporated by careful choice of coordinate system and asperity density parameters.
Parameterization of Generated Surfaces

Since the focus of this work is to demonstrate the viability and quality of various surface generation methods, the generated surfaces were chosen to have a simple geometry. The asperity densities in x and y were assumed to be equal to each other and set to a single parameter: ASPDENS = 1000 asperities / unit length, or one asperity every millimeter. The asperity sizes in x and y were assumed to be equal to each other and set to a value of 1/ASPDENS, thus each asperity is 1mm x 1mm square. The asperity size in z was assumed to be a normal Gaussian distribution with a mean of 0 and a standard deviation of 0.33 * MAXASPZ meters. In three of the four methods, no shape function was specified. In the last case, the asperities were assumed to have a pyramidal shape.

Rough Surface Generation Procedure

Procedure for Generating Roughness Values

Each of the four surfaces was generated from the same array of roughness values. Those arrays were created using the following procedure:

1. Create a (LENGTH*ASPDENS+1) by (WIDTH*ASPDENS+1) array named SURFKPZ to hold the mask of the nodes on the contact surface using the *DIM command to hold the random numbers for the surface. This generates an 11 x 11 array where each (ii,jj) entry represents the height of the asperity peak z for that (x,y) coordinate.
2. Use a DO loop to fill the array with random numbers using the *VFILL command with mean value of 0 and a standard deviation of 0.333333. In this way, the surface roughness can be scaled to any peak value.
3. Multiply the random number array by the peak amplitude MAXASPZ of the surface using the *VOPER command.

The APDL commands are shown below:

```
*DIM,SURFKPZ,ARRAY,LENGTH*ASPDENS+1,WIDTH*ASPDENS+1
*DO,ii,1,WIDTH*ASPDENS+1
  *VFILL,SURFKPZ(1,ii),GDIS, 0,0.3333333,
  *VOPER,SURFKPZ(1,ii),SURFKPZ(1,ii),MULT,MAXASPZ
*ENDDO
```

The array values created by using the procedure outlined above are shown in figure 1. The graphical user interface (GUI) only allows ten values per dimension of the array to be viewed at once so only values for SURFKPZ(1,1) through SURFKPZ(10,10) are shown. These values are all raised to powers of 10^-6 to 10^-8. The exponential power can only be viewed in the GUI if the individual cell is selected and scrolled all the way to the right.
Random Number Generation in ANSYS

In ANSYS, random numbers are generated using a recursive relation involving a seed number. Since the relation depends on the current and previous value of the seed, an “identical set of random numbers will be obtained” every time the same start value for the seed is used. In ANSYS, the start value for the seed is reset each time the program is closed and restarted for verification purposes so it was possible to generate the rough surfaces for each of the four proposed methods using identical arrays, making direct comparison of the surfaces possible.

Procedure for Moving All Nodes

The procedure to create a rough surface by moving nodes in the model selected by location is as follows:

1. Create a volume LENGTH x WIDTH x DEPTH using the BLOCK command.
2. Set the element size to 1/ASPDENS. This ensures one element per desired asperity location.
3. Mesh the volume using SOLID185 elements.
4. Detach the finite element model from the solid model using the MODMSH,DETACH command.
5. Select all nodes at the surface (z = DEPTH).
6. Use a *DO loop to select each surface node individually by its (x,y,z) coordinate and then use the NMODIF command to move the z coordinate of that node from DEPTH to DEPTH+SURFKPZ(ii+1,jj+1).

This procedure forces the x and y mesh size to correspond to the x and y asperity size and the mesh density to correspond to the asperity density. All vertical deviations between the asperity height and the nominal surface are contained within single elements.
The APDL commands are shown below:

```
MODMSH,DETACH
*DO,ii,0,LENGTH*ASPDENS
  *DO,jj,0,WIDTH*ASPDENS
    NSEL,S,LOC,Z,DEPTH
    NSEL,R,LOC,X,ii*(1/ASPDENS)
    NSEL,R,LOC,Y,jj*(1/ASPDENS)
    NMODIF,ALL,,DEPTH+SURFKPZ(ii+1,jj+1)
  *ENDDO
*ENDDO
```

Figure 2. Mesh Created By Moving All Nodes – Actual Scale (left), 100x Displacement (right)

The mesh geometry created by moving all nodes is shown in figure 2. The mesh on the left is shown with actual displacement scaling. The mesh on the right was created with displacement values one hundred times larger than those used to solve the model and is included to allow easy visualization of the generated surface.

**Procedure for Moving All Nodes and Selected Nodes By Location**

The procedure to move only selected nodes by location is almost the same as the procedure to move all nodes. However, in the “by location” case, the element size must be set to some multiple smaller than $1/\text{ASPDENS}$. For this work, the element size for the “by location” case was set to $1/(2*\text{ASPDENS})$.

This procedure lessens the degree of coupling between the mesh size and the asperity size (allowing the asperity size to now be a function of the mesh size instead of equal to it). However the mesh and asperity densities are still coupled and all vertical deviations between the asperity height and the nominal surface are still contained within single elements.
The mesh geometry created by moving nodes by location is shown in figure 3. The mesh for the “by location” case is twice as dense as the previous case. The asperities have the same heights at the same locations as the previous case but the asperities are visibly different. The geometry of the new rough surface is more discontinuous with much less of the surface area covered by asperities.

**Procedure for Creating Key Points**

The procedure to create a rough surface by creating key points is as follows:

1. Use a *DO loop to create key points at x and y intervals of 1/ASPDENS and with z values at DEPTH + the values in the SURFKPZ array.
2. Use a *DO loop to create lines between key points in the x direction.
3. Use a *DO loop to create lines between key points in the y direction.
4. Use a *DO loop to select lines by location and create areas for each asperity. Since the key points are not co-planar, ANSYS uses Coons patches to generate the surfaces. This step results in the rough surface at the top of the block.
5. Create four more key points at the base of the block and use bottom up solid modeling to create the block volume.
6. Use the VMESH,ALL command to create nodes and elements.

This procedure decouples the solid model geometry and the finite element mesh and makes it possible to have multiple elements per asperity.

Selected APDL commands are shown below:

```
*DO,ii,0,LENGTH*ASPDENS
  *DO,jj,0,WIDTH*ASPDENS
    kk = ii*(1/ASPDENS)*jj,DEPTH+SURFKPZ(ii+1,jj+1)
  *ENDDO
*ENDDO
*DO,ii,0,ASPDENS*LENGTH
  kk = ii*(ASPDENS*WIDTH+1)
  *DO,jj,1,WIDTH*ASPDENS
```
The mesh geometry generated by creating key points is shown in figure 4. The surface topography for the “key points” case is nearly identical to the “all nodes” case; however the mesh quality is significantly better and can be refined if necessary.

**Procedure for Creating Shell Primitives**

The method of creating rough surfaces by creating shell primitives is similar to creating key points. In the key point method, the rough surface area was created from the bottom up and then the block volume was created later. In the shell primitive method, the rough surface is created from the top down by using primitives available in ANSYS or by using custom macros and the block volume is created later.

The procedure to create a rough surface by creating shell primitives is as follows:

1. Use a *DO loop to create shell primitives at x and y intervals of 1/ASPDENS and with z values at DEPTH + the values in the SURFKPZ array.
2. Create four more key points at the base of the block and use bottom up solid modeling to create the block volume.

3. Since the center point of the asperity is located at the same (x,y) interval as the asperities in the other models, the asperities generated using this method effectively hang off the edges of the block. These over-hanging asperities must be trimmed by creating dummy volumes at the edges of the block and using the VSBV command to subtract the overhanging portions from the main volume.

4. Use the VMESH,ALL command to create nodes and elements.

This procedure also decouples the solid model geometry and the finite element mesh and makes it possible to have multiple elements per asperity. It also makes the implementation of specific asperity shape functions relatively easy.

The APDL commands to create the surface are shown below:

```
*DO,ii,0,LENGTH*ASPDENS
  *DO,jj,0,WIDTH*ASPDENS
  PYR4S,(1/ASPDENS)*ii,(1/ASPDENS)*jj,DEPTH,(1/ASPDENS),(1/ASPDENS),SURFKP
  Z(ii+1,jj+1)
*ENDDO
*ENDDO
```

For this method, four sided pyramidal shaped asperities were chosen. Since ANSYS does not offer pyramidal shell primitives, a custom macro was written. The macro PYR4S.mac takes six arguments: x, y and z coordinates for the center of the base, length of the pyramid base, width of the pyramid base and height of the pyramid tip. The macro returns four triangular areas in a pyramidal arrangement. The base area of the pyramid is not created.

![Figure 5. Mesh Created By Creating Shell Primitives – Actual Scale (left), 100x Displacement (right)](image-url)
The mesh geometry generated by creating shell primitives is shown in figure 5. As in the “by location” case, the surface topography in the “shell primitives” case is discontinuous since the base of each pyramidal shell is at the nominal surface. Two adjacent asperities with similar depths would create two adjacent pits for the “shell primitives” case where the same asperities would create one longer pit in the “creating key points” case.

The asperity geometry is mesh independent so the mesh quality is significantly better than that created by either “node” method and can be refined if necessary.

**Unimplemented Procedures**

Three other methods for creating rough surfaces were considered but not implemented: creating volume primitives, moving all nodes (throughout the model, not just surface nodes), and element sculpting.

In ANSYS, a number of volume primitives are available. It is possible to create a block volume and asperity volumes and then use Boolean add or subtract operations to incorporate the asperity volumes into the surface of the block volume. This would create a surface identical to the “shell primitives” case but could be used to modify existing geometry. This method was not implemented because of problems with the robustness of large scale Boolean operations in ANSYS.

As stated earlier, Hyun, et al. implemented a version of moving all nodes in the model in the z direction to prevent all of the asperity height from being incorporated into a single element. However, this was considered a brute force approach and an inelegant solution.

Similarly, element sculpting, or the addition and removal of elements or series of elements at various locations to produce asperities that contain multiple elements was considered a brute force approach and impractical on a large scale.

**Second Contact Surface**

A second contact surface must be present to interact with the rough surface. For this work, a smooth, infinitely stiff contact surface was chosen. The infinitely stiff contact surface is shown in figure 6 as the upper, purple surface. It was created using the following procedure:

1. Obtain the maximum z key point coordinate $\text{KPZMAX}$ on the rough surface using the *GET command.
2. Offset the current working plane to $\text{KPZMAX}$.
3. Create a rectangular area $\text{LENGTH} \times \text{WIDTH}$ on the working plane.
4. Set the number of element divisions along region boundary lines to be “1” using the ESIZE command. This will mesh the area with a single element.
5. Mesh the area with a target 170 element using the AMESH command.

The ANSYS commands are shown below:

ASEL,NONE
*GET,\text{KPZMAX},KP,0,MXLOC,Z
WPOFFS,,\text{KPZMAX}
RECT,0,LENGTH,0,WIDTH
TYPE,2
ESIZE,,1
AMESH,ALL

**Figure 6. Key Point Generated Element Mesh for Upper (Infinitely Stiff) Surface and Lower (Rough Deformable) Volume – Side View (right), Angled view (left)**

A pilot node was created at one corner of the infinitely stiff contact surface to permit boundary conditions to be applied to that surface. The ANSYS Element Reference states that a “pilot node provides a convenient, powerful way to assign boundary conditions such as rotations, translations, moments, temperature, and voltage on an entire rigid target surface. You assign the conditions only to the pilot node, eliminating the need to assign boundary conditions to individual nodes and reducing the chance of error.”\textsuperscript{8} Finally, the orientation of the single contact element was verified (reversed in this case) to ensure that both contact surfaces faced each other. The pilot node is shown as a small, purple asterisk or star on the left hand side of the upper contact surface.

The ANSYS commands to create the pilot node and reverse the node orientation are shown below:

\begin{verbatim}
*GET,NMIN,NODE,,NUM,MIN
TSHAPE,PILOT
E,NMIN
ESURF,,REVERSE
\end{verbatim}

**Analysis**

Contact analysis for all four cases was performed using a fully parameterized 3D model. Units were in the MKS system (meters, Kelvin, seconds). The element mesh used solid185 or solid187 elements, depending on whether or not a mapped mesh was necessary. After the first set of elements was produced, contact 173 or contact 174 elements were supplied to create a deformable contact surface. Key option five for the 173 and 174 elements was set to “1” to allow the element to close the gap between contact surfaces using auto CNOF.
Material properties were assigned based on polyetheretherketone (PEEK). The Young’s Modulus was assumed to be 4.6e9 Pa, the Poisson’s Ratio was assumed to be 0.4 and the yield stress was assumed to be 100e6 Pa. This material choice was made to ensure that stresses remained in the linear elastic regime.

A zero displacement constraint in x was placed on all nodes at x = 0. Similarly, zero displacement constrains in y and z were placed on all nodes at y = 0 and z = 0 and the pilot node was also constrained in x and y. Since the pilot node was located at (0,0,MAZKPZ), these boundary conditions constrained both the rough volume and the second contact surface. The z constraint for the pilot node was then removed to allow the infinitely stiff surface to move downward under the pre-load force. Finally, the downward pre-load force of – 500 N was applied to the pilot node, supplying the contact pressure for the system.

Results

Post processing returned three plots: a contour plot of the nodal solution for contact pressure, a contour plot of the element solution for contact pressure, and an element plot of the structural energy error. Contour plots are from solutions with MAXASPZ = 1e-5. The maximum structural energy error for solutions with MAXASPZ = 1e-6, 1e-5, 1e-4 and 1e-3 are listed in Table 1.

Solution Quality Metrics

There are three metrics for determining the quality of a surface generation method: quality of the surface, the quality of the solution and flexibility and robustness of the method.

The quality of the surface is determined by how closely the generated surface based on the supplied parameters resembles the desired surface. The solution must have some relationship to the physical situation that it is based on or the solution has no meaning and thus no value to the user.

The quality of the solution is a function of the number, shape and placement of elements in the mesh. If a good quality mesh is present both in the bulk of the material and in the asperities, a good quality solution should be achieved. This can be determined by comparing the nodal (averaged) solution to the element (non-averaged) solution. If the solutions very closely resemble one another, then the solution discontinuities across element boundaries are very small and the user can have a high degree of confidence in the solution. In addition, solution quality can be determined by calculating the structural energy error. Structural energy error is a “measure of the discontinuity of the stress field from element to element” which can be used as an estimation of “the amount of solution error due specifically to mesh discretization.”

Several factors affect the flexibility and robustness of the method. Some of these factors are: whether or not the method can be used on imported CAD geometry, whether the method scales well for large models, whether the method produces good geometry
without generating errors in the program and whether the method can be modified for more complex geometries.

**Contact Pressure Plots**

Since the comparison of the nodal and element stress solutions is addressed, in part, by the discussion of structural energy error, the nodal and element contact pressure plots were chosen for qualitative comparisons of the averaged and non-averaged solutions.

The contact pressure plots for the surface created by moving all nodes are shown in figure 7. The two plots look different with the nodal solution producing smooth curves and the element solution having distinct element boundaries visible along solution lines. In addition to the differences in the shape of the plots, the location of maximum contact pressure is different for the two plots. In the nodal plot, the maximum contact pressure is located at the bottom center of the surface while it is located in the top right hand corner in the element plot. While the maximum value of contact pressure for the plots are within 100% of each other, the shape disparities still may be cause for re-evaluation of the quality of results.

![Figure 7. Contact Pressure Created by Moving All Nodes– Nodal Solution (left), Element Solution (right)](image)

The contact pressure plots for the surface created by moving selected nodes by location are shown in figure 8. The two plots look similar with areas of maximum contact pressure in the same location and maximum contact pressure values within 5% of each other. The element solution shows more contact than the nodal solution which is unusual and probably a result of the non-asperity elements averaging out the presence of low contact pressure regions on the surface. Overall, the two solutions are better matched than the solution above. However the shape of the asperity is very clearly influencing the shape of the contact region and ultimately there may be issues with this surface failing to resemble a real surface in addition to the coupling between the finite element mesh and the asperity geometry.
Figure 8. Contact Pressure Created by Moving Nodes by Location– Nodal Solution (left), Element Solution (right)

The contact pressure plots for the surface created by creating key points are shown in figure 9. In this case, the agreement between the two solutions is excellent. The maximum contact pressure values are within 14% of each other and the shapes are very similar.

Figure 9. Contact Pressure Generated by Creating Key Points– Nodal Solution (left), Element Solution (right)

The contact pressure plots for the surface created by primitive shells are shown in figure 10. Again, the shape of the plots are in excellent agreement. The location of maximum contact pressure is in the same place for both plots although the maximum values are only within 67% of each other. Again, the asperity shape is clearly having an affect on the contact geometry and it may be necessary to round off the tips of the pyramids to yield better results in the future.
**Structural Energy Error**

The four surface generation methods were used in conjunction with MAXASP asperity scaling factors spanning four orders of magnitude to explore the variation in structural energy error and the robustness of the surface generation method. The meshes for the volumes “creating key points” and “shell primitives” were automatically generated by the program without an input from the user. The results are listed in Table 1.

### Table 1. Structural Energy Error for Various Surfaces

<table>
<thead>
<tr>
<th>MAXASP</th>
<th>All Nodes</th>
<th>By Location</th>
<th>Key Points</th>
<th>Shells</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max SERR</td>
<td>0.000001</td>
<td>3.45E-07</td>
<td>8.13E-08</td>
<td>4.30E-08</td>
</tr>
<tr>
<td>Max SERR</td>
<td>0.00001</td>
<td>1.16E-05</td>
<td>7.43E-06</td>
<td>7.30E-07</td>
</tr>
<tr>
<td>Max SERR</td>
<td>0.001</td>
<td>1.74E-04</td>
<td>1.87E-04</td>
<td>2.82E-05</td>
</tr>
<tr>
<td>Max SERR</td>
<td>0.001</td>
<td>N/A</td>
<td>N/A</td>
<td>4.06E-04</td>
</tr>
</tbody>
</table>

All four surface generation methods had very low structural energy errors for small asperities. In this case, “small” asperities are defined as vertical nodal displacements that increase the element height by 100% or less for the first two surface generation methods.

However, when the asperity sizes start to approach the size of the elements, nodal displacements begin to increase the element height by 200 – 300% or more, which can generate poorly shaped elements and lead the program to refuse to produce a solution. When MAXASP was equal to 0.001m, the program would not generate a solution for either of the surfaces generated by nodal displacement which is why a value of structural energy error is not listed for either method at that scaling value.

There is no value for structural energy error for the surface generated by shell primitives for the largest value of MAXASP for a different reason. At that length scale and for that
particular surface, contact in the model occurs at a single point. Since the pyramidal shaped asperities have sharp tips, there is a numerical singularity at that location. The program will generate a solution but the solution is essentially meaningless. Only the surface created by generating key points was able to produce a meaningful solution at the largest value of $\text{MAXASP}$. This is most likely due to the presence of the Coons patches which helps to smooth out the surface topography and ensure multiple points of contact. Modifying the mesh in an attempt to improve the error resulted in similar contact singularities and the user is encouraged to use the initial mesh produced by the program.

**Conclusion**

Overall, the method of creating rough surfaces by generating key points produces the best results in terms of nodal/element solution agreement, structural energy error, flexibility, and robustness. Creation of surfaces by generating primitive shells may be a useful method, especially for verification of analytical models that are based on assumptions about asperity shape. While the creation of rough surfaces through the modification of finite element entities produces better results in terms of nodal/element solution agreement and structural energy error than anticipated, especially for small asperity heights, the user is still cautioned to examine results very carefully to ensure validity of solution when using said method.

**Acknowledgements**

The author would like to thank the Cambridge-MIT Institute for support for the research and writing of this article. She would also like to thank Prof. Nam P. Suh, Prof. Alexander H. Slocum, and Dr. John M. Thompson for their input, feedback and support.

**References**

7. Section 21.2.2.1. ANSYS Theory Reference
8. TARGE170 ANSYS Element Reference
9. Section 5.3.6. ANSYS Basic Analysis Procedures Guide