

## Appendix H

```

clc
close all
clear all

data=stlread('HemiellipsoidFlatSideFillet10_Surface.stl');
% Meshing of the file
V = vertexNormal(data);
% Calculating the angle of each normal vector
xx=V(:,1);
yy=V(:,2);
zz=V(:,3);
% Orientation matters, need to check what is the part vertical or "out-of-
% plane" direction, which goes in 'tan'
[pyt]=sqrt(xx.^2+zz.^2);
[tan]=pyt./yy;
[alpharadians]=atan(tan);
[alphadegrees]=alpharadians.*180./pi;

%Calculating the slope
for i= 2:length(data.Points(:,1))
    slope(i,1) = (data.Points(i,2) - data.Points(i-1,2))/ ...
    % Change the indices for difference orientations for the columns in
    %"data.Points"
    data.Points(i,1)-data.Points(i-1,1));
end

% Creating a matrix with vectors, angles, and positions
E=[V alphadegrees data.Points slope];
% Sort
% If mesh is organized in Z, use 7. 6 for Y and 5 for X
sortedData=sortrows(E,7);
% UmyTol to determine if 2 values should be considered the same
myTol = 0.01;
% Start index of group
grpStartIdx = 1;
% Stop index of group
grpStopIdx = 1;
% A cell array for the grouped data
groupedData = {};
atGroupIdx = 0;
% Iterate over each row of sortedData starting at second element
for k=2:size(sortedData,1)
    % Compare the first element with the previous
    % Change index according to what is defined in sortrows. The two
    % consecutive elements are considered to be identical so must be part of
    % same group
    if abs(sortedData(k,7)-sortedData(k-1,7)) < myTol
        grpStopIdx = k;
    else
        % The two consecutive elements are not identical since their
        % difference is greater than myTol so new group is needed
        grpStopIdx = k-1;
        % Copy all data (R,Z,TH) as a group to the cell array
        atGroupIdx = atGroupIdx + 1;
        groupedData{atGroupIdx} = sortedData(grpStartIdx:grpStopIdx,:);
        % Update next group indices
        grpStartIdx = k;
        grpStopIdx = k;
    end
end

```

```

    end
end

% Make sure elements are in the right order
% Use 5 for organised plotting against x-coordinate, 6 to organise in y,
% and 7 in z

% Select scan path as groupedData{1,:}. Only some will work, as shown as
commented
% Fillet10 282, 546 or 577(vertical 2045)
sortedGroupmiddle=sortrows(groupedData{1,282},5);
%sortedGroupmoved=sortrows(groupedData{1,306},5);
%sortedGroupflange=sortrows(groupedData{1,168},7);

% Plot the angle
figure
plot(sortedGroupmiddle(:,5),sortedGroupmiddle(:,4),'.-r');
% Un-comment the following to compare several scan-paths at once
% hold on
% plot(sortedGroupmoved(:,5),sortedGroupmoved(:,4),'.-m');
% hold on
% plot(sortedGroupflange(:,5),sortedGroupflange(:,4),'.-y');
ylim([-180 180])
title('Surface normal orientation');
xlabel('Position in x axis');
ylabel('Angle (°)');
legend('Middle section scan-path'); %, 'Top section scan-path', 'Flange
section scan-path');

% Plot the slope variation
figure
plot(sortedGroupmiddle(:,5),sortedGroupmiddle(:,8),'--k');
% Un-comment the following to compare several scan-paths at once
% hold on
% plot(sortedGroupmoved(:,5),sortedGroupmoved(:,8),'--b');
% hold on
% plot(sortedGroupflange(:,7),sortedGroupflange(:,8),'--c');
title('Slope variation along scan path');
xlabel('Position in x axis');
ylabel('dz/dx');
legend('Middle section scan-path'); %, 'Top section scan-path', 'Flange
section scan-path');

% Plots the surface with the surface normals
figure
trisurf(data,'FaceColor','cyan','FaceAlpha', 0.8);
axis equal
hold on
quiver3(data.Points(:,1),data.Points(:,2),data.Points(:,3), ...
        V(:,1),V(:,2),V(:,3),0.5,'Color','r');
hold off
xlabel('X direction');
ylabel('Y direction');
zlabel('Z direction');
hold on
x_line=sortedGroupmiddle(:,5);
y_line=sortedGroupmiddle(:,6);
z_line=sortedGroupmiddle(:,7);
plot3(x_line,y_line,z_line,'g','LineWidth',3)
% Un-comment the following to compare several scan-paths at once
% hold on

```

---

```
% x_line=sortedGroupmoved(:,5);
% y_line=sortedGroupmoved(:,6);
% z_line=sortedGroupmoved(:,7);
% plot3(x_line,y_line,z_line,'m','LineWidth',3)
% hold on
% x_line=sortedGroupflange(:,5);
% y_line=sortedGroupflange(:,6);
% z_line=sortedGroupflange(:,7);
% plot3(x_line,y_line,z_line,'y','LineWidth',3)
```