EGR 103L – Fall 2017

Graphics and Loops

Ian Hanus (ih52)
Lab Section 1B, Tuesday 8:30-11:20
22 October 2017

I understand and have adhered to all the tenets of the Duke Community Standard in completing every part of this assignment. I understand that a violation of any part of the Standard on any part of this assignment can result in failure of this assignment, failure of this course, and/or suspension from Duke University.

Contents

List of Figures

1 Chapra Problem 4.1

a	ϵ_s	maxit	$\sqrt{a}$	ϵ_a	iter
16	1e-02	5.000000	4.002257524798522e+00	3.36e+00	5
16	1e-02	12.000000	4.000000000000051e+00	1.59e-05	7
16	1e-08	5.000000	4.002257524798522e+00	3.36e+00	5
16	1e-08	12.000000	4.000000000000000e+00	1.27e-12	8
160	1e-02	5.000000	1.482664109800340e+01	5.22e+01	5
160	1e-02	12.000000	1.264911068004731e+01	7.89e-03	8
160	1e-08	5.000000	1.482664109800340e+01	5.22e+01	5
160	1e-08	12.000000	1.264911064067352e+01	0.00e+00	10
1600	1e-02	5.000000	1.052575377021292e+02	9.25e+01	5
1600	1e-02	12.000000	4.000000000060651e+01	5.51e-04	10
1600	1e-08	5.000000	1.052575377021292e+02	9.25e+01	5
1600	1e-08	12.000000	4.000000000000000e+01	1.52e-09	11
16000	1e-02	5.000000	1.005306930179416e+03	9.92e+01	5
16000	1e-02	12.000000	1.264911064067374e+02	1.86e-05	12
16000	1e-08	5.000000	1.005306930179416e+03	9.92e+01	5
16000	1e-08	12.000000	1.264911064067374e+02	1.86e-05	12
160000	1e-02	5.000000	1.000531194227066e+04	9.99e+01	5
160000	1e-02	12.000000	4.000285706905372e+02	1.20e+00	12
160000	1e-08	5.000000	1.000531194227066e+04	9.99e+01	5
160000	1e-08	12.000000	4.000285706905372e+02	1.20e+00	12

The quality of approximations can be measured by error, or the true value minus the approximation [1, p. 101]. The quality of the approximations are dependent on the factor that stopped the loop from continuing. If the stopping tolerance limited the number of iterations that the program went through, then the error of the approximations were within that stopping error. Therefore, if stopping error was the limiting factor, the times the loop was run with a smaller stopping error are more accurate. If the limiting factor was the number of iterations, the greater the number of iterations the closer the approximation will get to the real value. The size of the number that is being approximated is also a factor. The approximations of the larger numbers generally tend to have larger errors.

2 Palm Figure 6.1-2

Replication of figure 6.1-2 in Palm [2, p. 265]

3 Palm Problems 5.33

The temperature at the corner $x = y = 0$ is 1.47^{o}C

4 Palm Problem 4.28

Customer	**x location (mi)**	**y location (mi)**	**Volume (tons/week)**
1	10	-10	6
2	-11	-13	2
3	-8	-17	5
4	27	-26	2
5	-3	14	6
6	16	5	9
7	-26	22	7
8	14	-8	7
9	-17	-21	3
10	25	4	4

The best location for the distribution center is at coordinate (8,-2) for a total cost of 4.9605e+02 dollars.

A Codes

A.1 RunDivAvg.m

```
 % I have adhered to all the tenets of the
 % Duke Community Standard in creating this code.
 % Signed: [ih52]
%% Initialize the workspace
clear; format short e
%% Set up lists of parameters
a = [ones(1,4)*16, ...
     ones(1,4)*160, ...
     ones(1,4)*1600, ...
     ones(1,4)*16000, ...
     ones(1,4)*160000];
es = repmat([1e-2 1e-2 1e-8 1e-8], 1, 5);
maxit = repmat([5 12], 1, 10);
%% Run loop and store 20 sets of results
ea = a; iter = a; fx = a;
for k=1:20
    [fx(k), ea(k), iter(k)] = DivAvg(a(k), es(k), maxit(k));
end
%% Open file for writing
FID = fopen('DivAvgTable.tex', 'w');
%% Write the table to a file
% print the tabular line and a newline
fprintf(FID, '\n\\begin{tabular}{|ccc|ccc|}\\hline \n');
% print the table headers and a horizontal line then a newline
fprintf(FID, '$a$ & $\\epsilon_s$ & maxit & $\\sqrt{a}$ & $\\epsilon_a$ & iter\\\\ \\hline \n');
for k=1:20
    % print a line of the table - but no newline!
    fprintf(FID,'%0.0f & %0.0e & %f & %1.15e & %1.2e & %2.0f\\\\',a(k),es(k),maxit(k),fx(k),ea(k),iter(k
    % print a horizontal line every 4 row
    if mod(k,4) == 0 %%% YOUR LOGIC REPLACES THE 0 HERE
        fprintf(FID, '\\hline ');
    end
    % print a newline
    fprintf(FID, '\n');
end
fprintf(FID, '\\end{tabular}\n');
%% Close the file
fclose(FID);
```

A.2 DivAvg.m

```
 % I have adhered to all the tenets of the
 % Duke Community Standard in creating this code.
 % Signed: [ih52]
function [fx, ea, iter] = DivAvg(a, es, maxit)
% DivAvg  Use Divide and Average to find square root
%   [fx, ea, iter] = DivAvg(a, es, maxit)
%     a: number of which to take the square root
%     es: stopping error
%     maxit: maximum number of iterations
%     fx: approximation of square root of a
%     ea: approximate relative error (%)
%     iter: number of iterations
% Based on IterMeth.m from Figure 4.2 on p. 94 of
% Applied Numerical Methods with MATLAB for
% Scientists and Engineers
% Steven C. Chapra, 3rd Edition

% Honor code

%% defaults:
if nargin<2|isempty(es),es=0.0001;end;
if nargin<3|isempty(maxit),maxit=50;end;
%% initialization
iter = 1; sol = a; ea = 100;
%% iterative calculation
while (1)
    oldsol = sol;
    sol = (sol + a./sol)./2;
    iter = iter + 1;
    if sol ~= 0
         ea = abs((sol-oldsol)./sol)*100;
    end
    if ea <= es | iter>=maxit
        break
    end
end
fx = sol;
end
```

A.3 PalmFigure612.m

```
 % I have adhered to all the tenets of the
 % Duke Community Standard in creating this code.
 % Signed: [ih52]
figure(1);clf
xlim([0,4])
ylim([0,4])
title('The Power Function {\it y = x^m} (ih52)')
xlabel('{\it x}')
ylabel('{\it y}')
x = linspace(0,4,1000);
hold on
for k = [-0.5,0,0.5,1,2]
    plot(x,x.^k,'k-')
end
text(1.25,3.4,'{\it m} = 2')
text(2.75,3.2,'{\it m} = 1')
text(3.25,2,'{\it m} = 0.5')
text(3.25,1.15,'{\it m} = 0')
text(2.75,0.4,'{\it m} = -0.5')
hold off
print -depsc PalmFigure
```

A.4 PalmProblem533.m

```
 % I have adhered to all the tenets of the
 % Duke Community Standard in creating this code.
 % Signed: [ih52]
clear; format short e
[x,y] = meshgrid(0:0.025:1);
T = 80.*exp(-(x-1).^2).*exp(-3.*(y-1).^2);
%% Surface Plot
figure(1);clf
xlabel('x (distance)')
ylabel('y (distance)')
title('Surface Plot of Palm 5.33')
surfc(x,y,T)
colorbar
colormap hot
print -depsc SurfacePlot533
%% Contour Plot
figure(2);clf
xlabel('x (distance)')
ylabel('y (distance)')
title('Contour Plot of  Palm 5.33')
contour(x,y,T,[0:10:80])
colormap hot
print -depsc ContourPlot533
%% Imagesc Plot
figure(3);clf
xlabel('x (distance)')
ylabel('y (distance)')
title('Imagesc Plot of Palm 5.33')
imagesc([0 1],[0 1],T)
colorbar
colormap hot
print -depsc ImagescPlot533
```

A.5 PalmProblem428.m

```
 % I have adhered to all the tenets of the
 % Duke Community Standard in creating this code.
 % Signed: [ih52]
clear; format short e
Data = load('DataTable.dat');
Xloc = Data(:,2);
Yloc = Data(:,3);
Volume = Data(:,4);
Elements = numel(Xloc);
figure(1);clf
xlim([-30 30])
ylim([-30 30])
xlabel('{\it x} (miles)')
ylabel('{\it y} (miles)')
title('Customer Map (ih52)')
for k = 1:Elements
    text(Xloc(k),Yloc(k),num2str(k))
end
print -depsc CustomerMap428
%% Graph 2
figure(2);clf
[X,Y] = meshgrid(-30:2:30);
TotalCost = zeros(31);
for k = 1:Elements
    if k == 1
        Distance = sqrt((X-Xloc(k)).^2+(Y-Yloc(k)).^2);
        Cost = 0.5.*Distance.*Volume(k);
        TotalCost = Cost;
    else
        Distance = sqrt((X-Xloc(k)).^2+(Y-Yloc(k)).^2);
        Cost = 0.5.*Distance.*Volume(k);
        TotalCost = Cost + TotalCost;
    end
end
meshc(X,Y,TotalCost)
xlabel('{\it x} (miles)')
ylabel('{\it y} (miles)')
zlabel('Cost ($)')
title('Mesh with Contours (ih52)')
MinCost = min(TotalCost(:))
MinCostLoc = find(MinCost == TotalCost);
MinCostXLoc = -30 + 2*floor(MinCostLoc./31)
MinCostYLoc = -30 + 2*mod(MinCostLoc,31)
colormap copper
print -depsc MeshContours428
%% Graph 3
figure(3); clf
surf(X,Y,TotalCost)
shading interp
colormap copper
view(2)
colorbar
xlabel('{\it x} (miles)')
ylabel('{\it y} (miles)')
title('Surface Plot from Above (ih52)')
```

```
print -depsc SurfacePlot428
%% Graph 4
figure(4); clf
contour(X,Y,TotalCost,10)
colormap copper
colorbar
xlabel('{\it x} (miles)')
ylabel('{\it y} (miles)')
title('Contour Plot (ih52)')
print -depsc ContourPlot428
```

B Figures

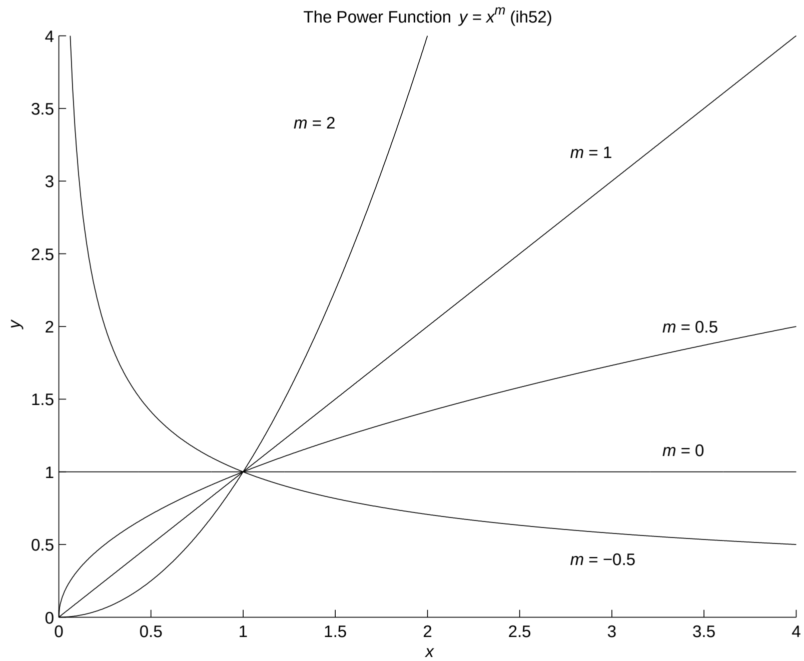

Figure 1: Palm Figure 6.1-2

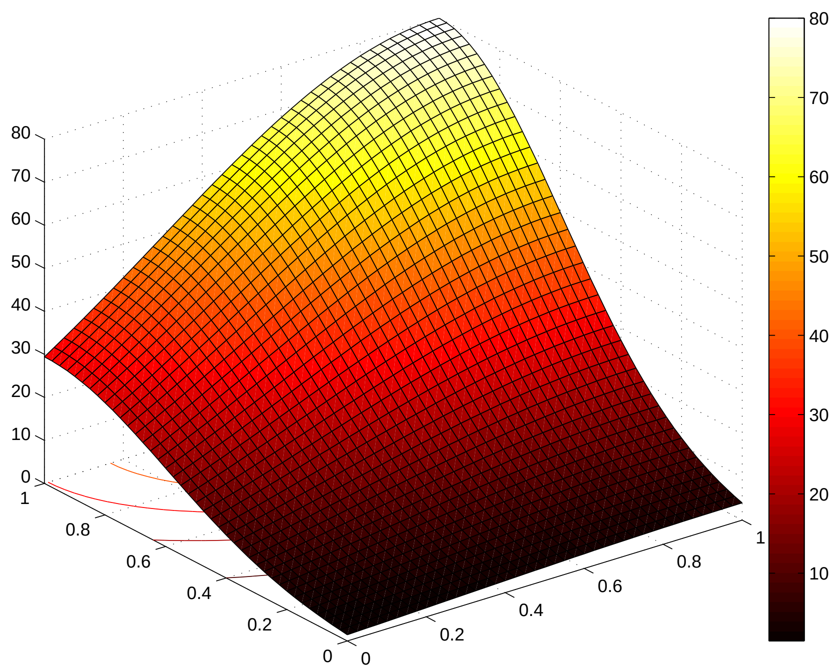

Figure 2: Surface Plot of Palm Problem 5.33

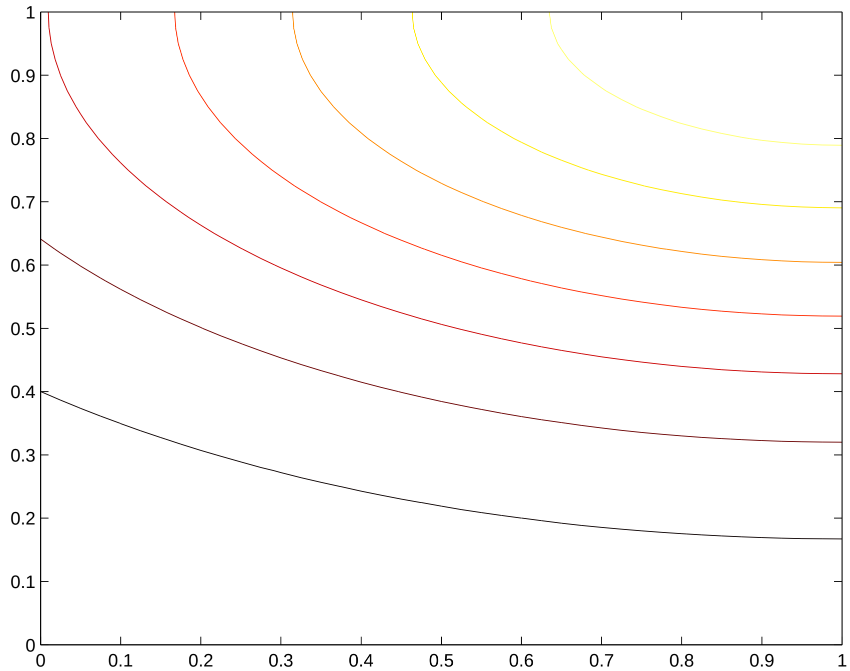

Figure 3: Contour Plot of Palm 5.33

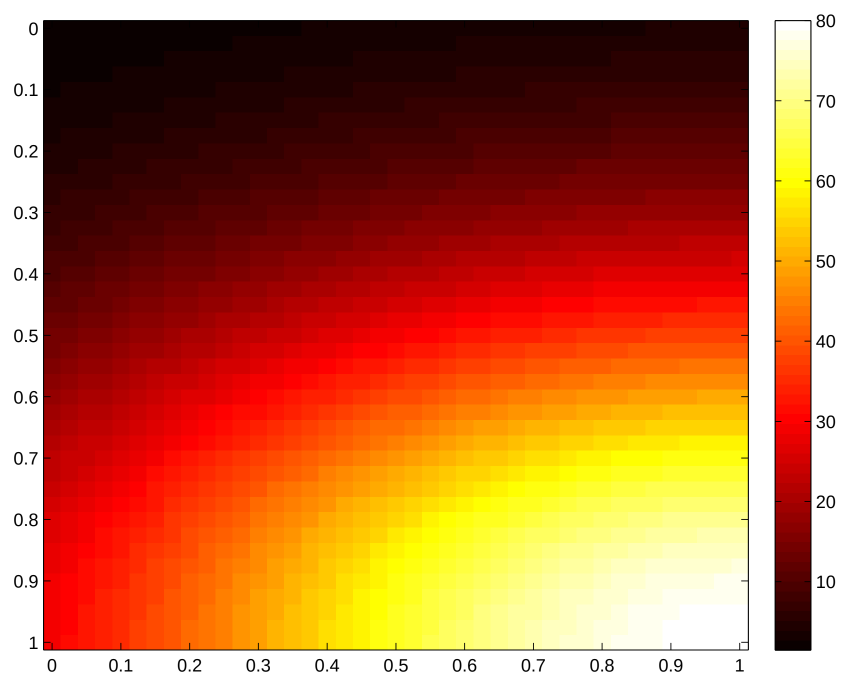

Figure 4: Imagesc Plot of Palm 5.33

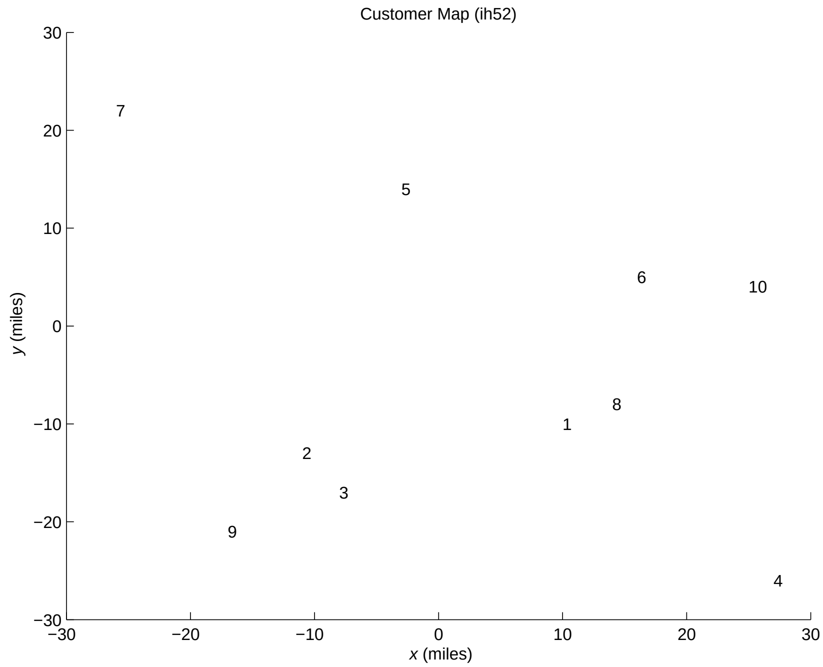

Figure 5: Customer Map of Palm 4.28

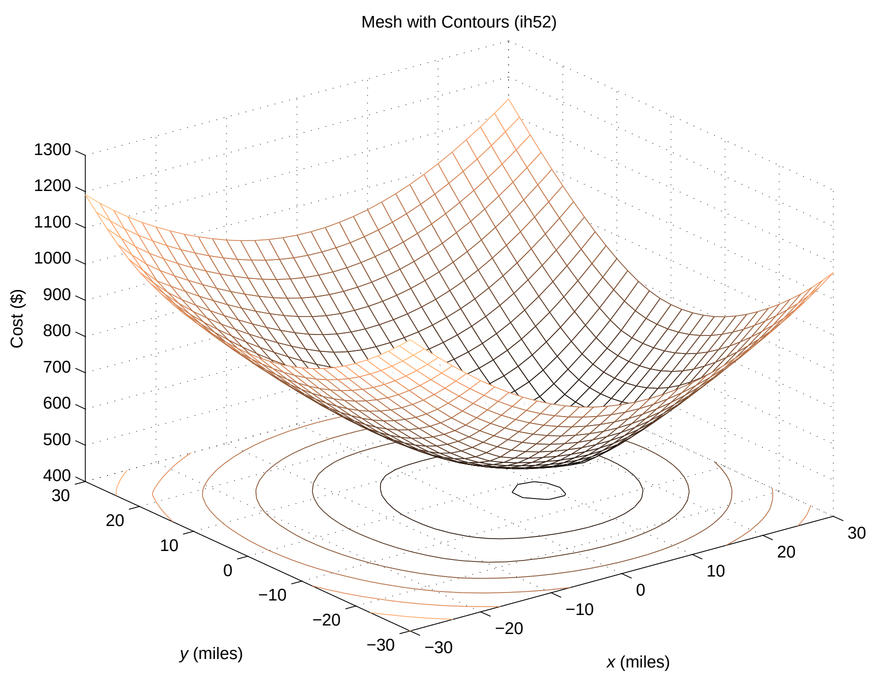

Figure 6: Mesh with Contours Plot of Palm 4.28

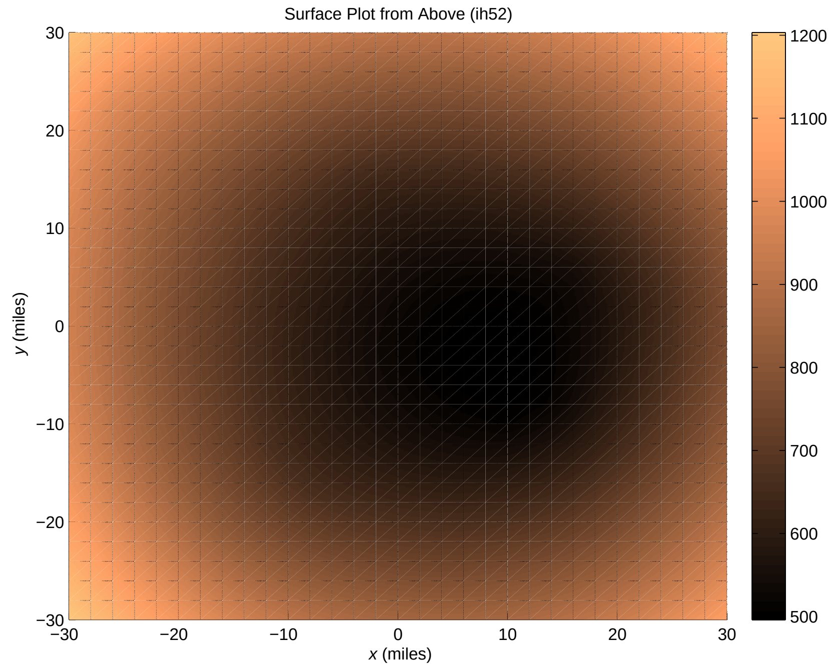

Figure 7: Surface Plot from Above of Palm 4.28

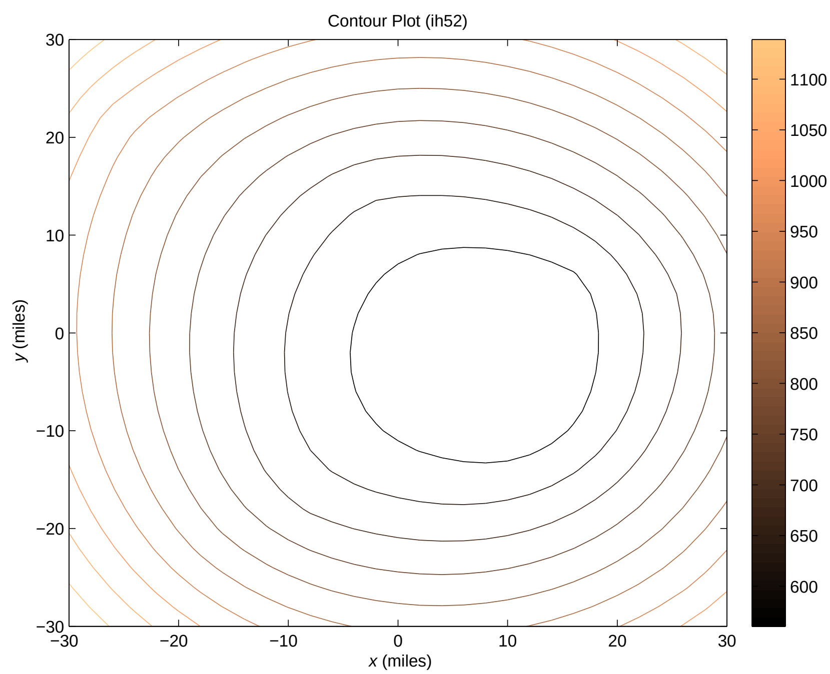

Figure 8: Contour Plot of Palm 4.28

References

[1] Chapra, Steven C., *Applied Numerical Methods with MATLAB for Engineering and Scientists*. McGraw-Hill, New York, 4th Edition, 2018.

[2] Palm, William J., *Introduction to MATLAB for Engineers*. McGraw-Hill, New York, 3rd Edition, 2011.