

**THE UNIVERSITY OF TEXAS AT ARLINGTON, TEXAS  
DEPARTMENT OF ELECTRICAL ENGINEERING**

**EE 5329**

**Distributed Decision and Control**

**TAKE HOME EXAM 3**

**by**

**SOUTRIK PRASAD MAITI**

**1001569883**

**Presented to**

**Dr. Frank Lewis**

**May 1, 2018**

**EE 5329 Distributed Decision and Control**

**Spring 2018**

**Exam Pledge of Honor**

On all exams in this class - YOU MUST WORK ALONE.

***Any cheating or collusion will be severely punished.***

***It is very easy to compare your software code and determine if you worked together***

***It does not matter if you change the variable names.***

Please sign this form and include it as the first page of all of your submitted homeworks.

.......................................................………………………………………………………………........

Typed Name: Soutrik Maiti

***Pledge of honor:***

"On my honor I have neither given nor received aid on this homework.”

e-Signature: Soutrik Maiti

Problem 1:

*MATLAB Code* –

clc;

clear all;

close all;

%% A matrix for undirected tree

a = [0 0 1 0 0 0;

1 0 0 0 0 1;

1 1 0 0 0 0;

0 1 0 0 0 0;

0 0 1 0 0 0;

0 0 0 1 1 0;];

x1 = [1 1];

x2 = [1 -1];

x3 = [-1 -1];

x4 = [-1 1];

x5 = [2 0];

x6 = [-2 0];

x = [x1;x2;x3;x4;x5;x6];

x2\_V = -5:0.1:5;

v1 = zeros(length(x2\_V),length(x2\_V));

for m = 1:length(x2\_V)

for n = 1:length(x2\_V)

for j = 1:6

v = a(2,j)\*((x(j,:)-[x2\_V(m) x2\_V(n)])\*(x(j,:)-[x2\_V(m) x2\_V(n)])');

v1(m,n) = v1(m,n) + v;

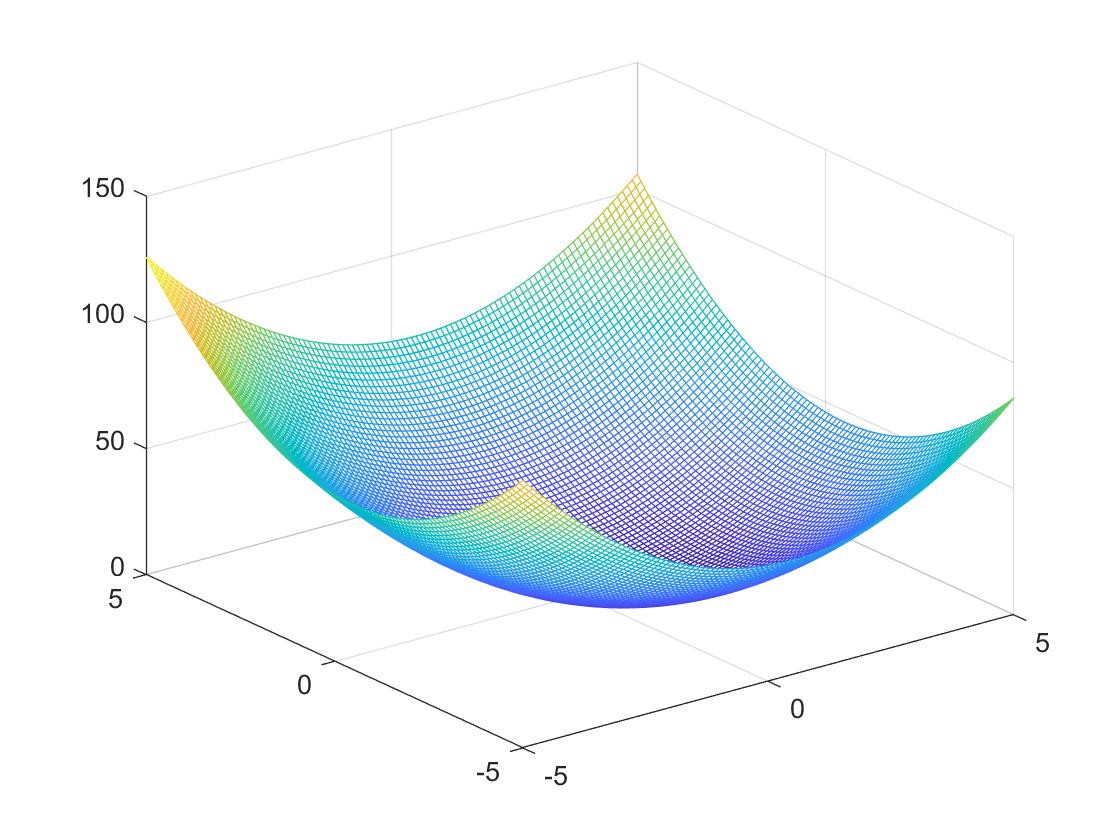
end

end

end

mesh(x2\_V,x2\_V,v1)

The partial potential V2(x2) is shown in the following:



Problem 2:

MATLAB Code –

clc;

clear all;

close all;

%% A matrix for undirected tree

a = [0 0 1 0 0 0;

1 0 0 0 0 1;

1 1 0 0 0 0;

0 1 0 0 0 0;

0 0 1 0 0 0;

0 0 0 1 1 0;];

x1 = [1 1];

x2 = [1 -1];

x3 = [-1 -1];

x4 = [-1 1];

x5 = [2 0];

x6 = [-2 0];

x = [x1;x2;x3;x4;x5;x6];

x6\_V = -5:0.1:5;

v1 = zeros(length(x6\_V),length(x6\_V));

for m = 1:length(x6\_V)

for n = 1:length(x6\_V)

for j = 1:6

v = a(6,j)\*((x(j,:)-[x6\_V(m) x6\_V(n)])\*(x(j,:)-[x6\_V(m) x6\_V(n)])');

v1(m,n) = v1(m,n) + v;

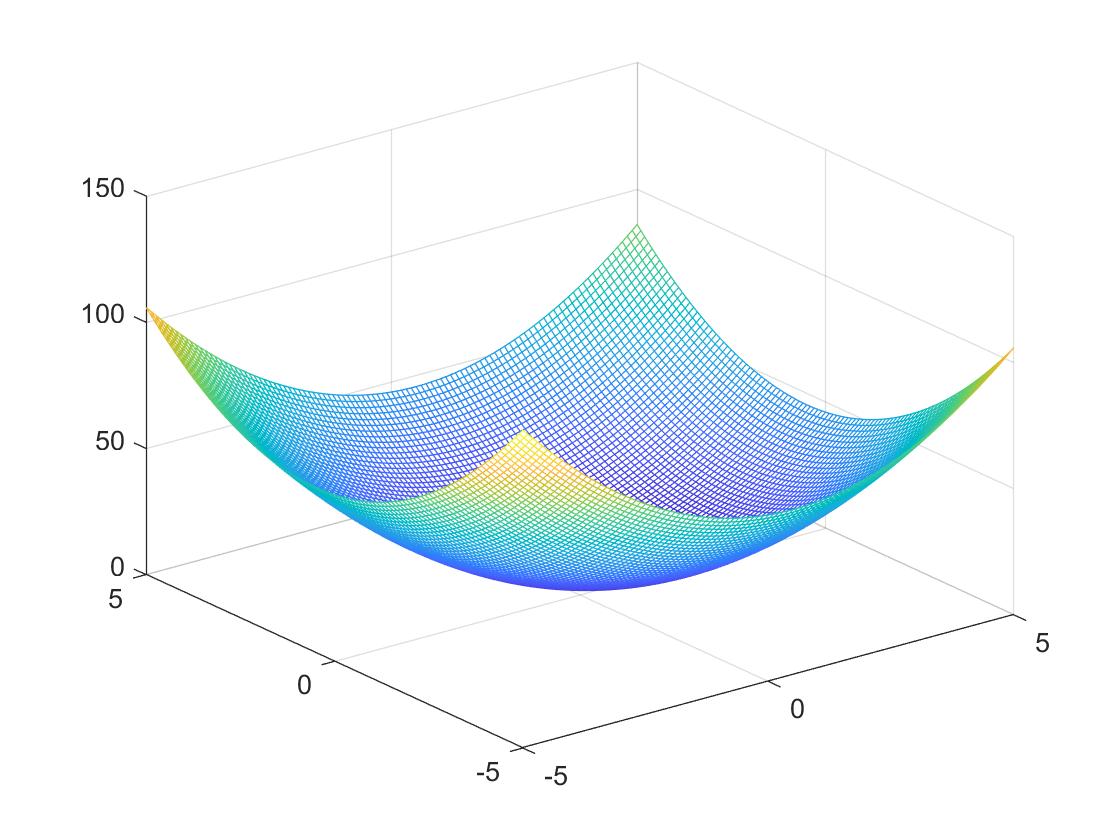
end

end

end

mesh(x6\_V,x6\_V,v1)

The plot of V6(x6) as x6 varies is as follows:



Problem 3:

MATLAB Code -

%% A matrix for undirected tree

a = [0 0 1 0 0 0;

1 0 0 0 0 1;

1 1 0 0 0 0;

0 1 0 0 0 0;

0 0 1 0 0 0;

0 0 0 1 1 0;];

%% Position of agents

x1 = [1 1];

x2 = [1 -1];

x3 = [-1 -1];

x4 = [-1 1];

x5 = [2 0];

x6 = [-2 0];

x = [x1;x2;x3;x4;x5;x6];

%% varying x2

x2\_V = -5:0.1:5;

v1 = ones(length(x2\_V),length(x2\_V));

%% Algorithm for given equation

for m = 1:length(x2\_V)

for n = 1:length(x2\_V)

for j = 1:6

if j == 1||j == 6

v = a(2,j)\*((x(j,:)-[x2\_V(m) x2\_V(n)])\*(x(j,:)-[x2\_V(m) x2\_V(n)])');

v1(m,n) = v1(m,n) \* v;

end

end

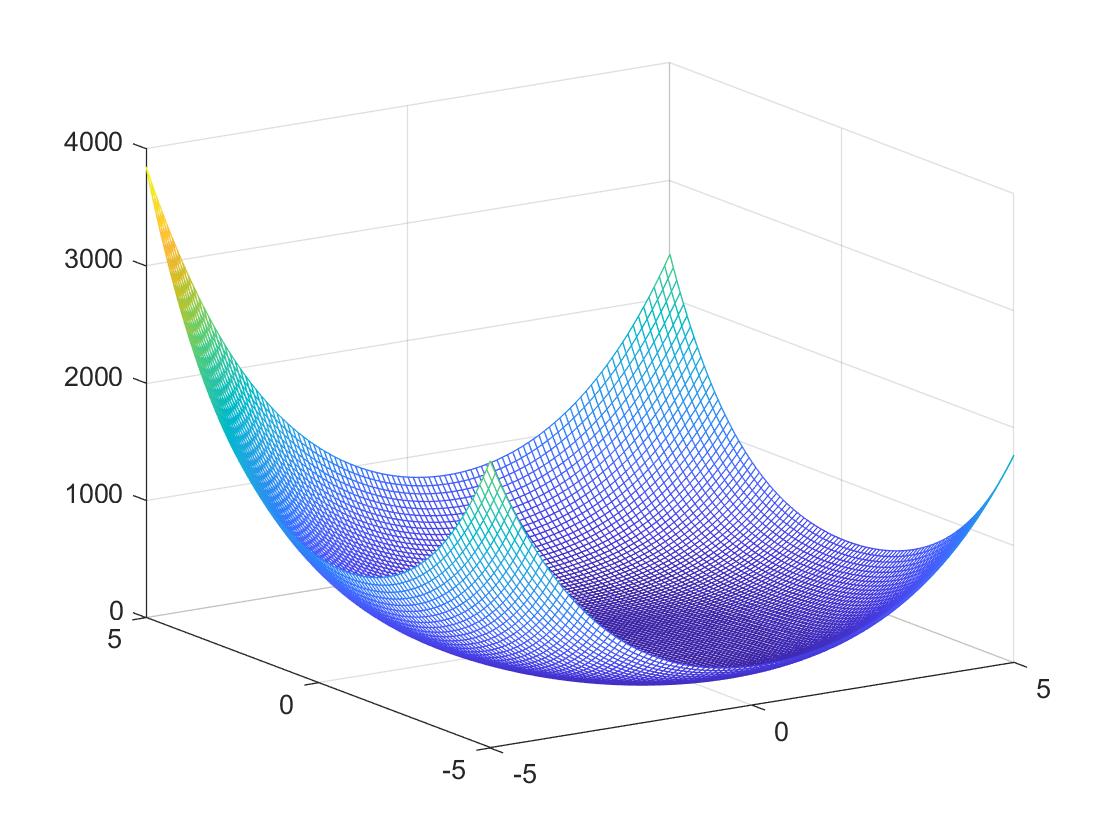
end

end

%% 3D plot of given equation

mesh(x2\_V,x2\_V,v1)

The plot is as follows:



b) The new function gives a flat base whereas the original Laplacian gave a curved bottom having a particular minimum as seen from the image.