BMS INSTITUTE OF TECHNOLOGY AND MANAGEMENT

Department Of Electronics And Telecommunication

VI SEMESTER

COMPUTER COMMUNICATION NETWORKS (18TE63)

DISTANCE VECTOR ROUTING ALGORITHM IN MATLAB

COURSE COORDINATOR
Dr. Seema Singh

PRESENTED BY
S Varsha - 1BY20ET048

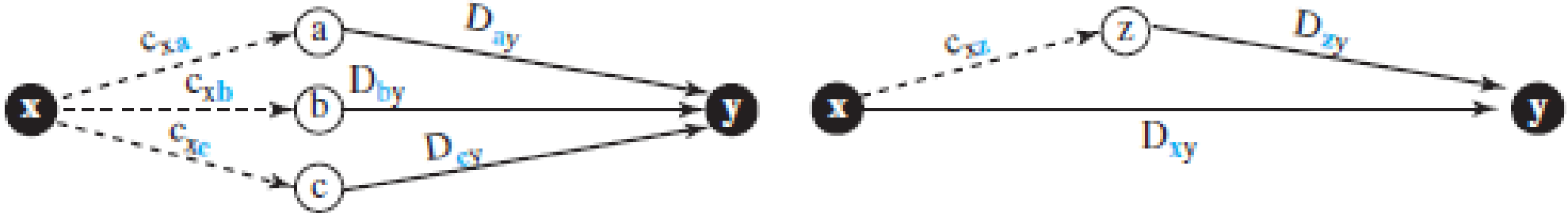
INTRODUCTION

- Unicast routing is when a datagram is destined for only one destination (one-to-one delivery).
- Unicast routing in the Internet can be done only by using hierarchical routing: routing in several steps using different routing algorithms.
- In distance-vector routing, the first thing each node creates is its own least-cost tree with the rudimentary information it has about its immediate neighbors.
- The incomplete trees are exchanged between immediate neighbors to make the trees more and more complete and to represent the whole internet.

Bellman-Ford equation

$$Dxy = \min\{(cxa + Day), (cxb + Dby), (cxc + Dcy), \ldots\}$$

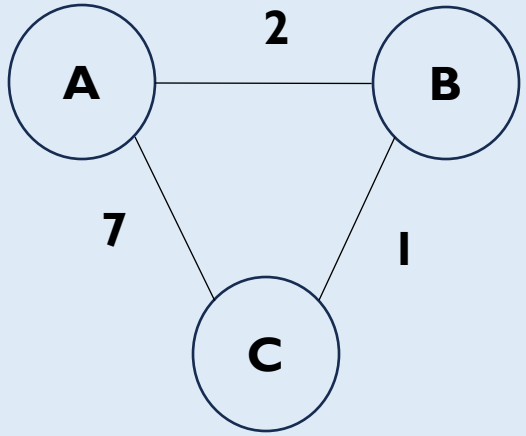
Figure 20.3 *Graphical idea behind Bellman-Ford equation*



a. General case with three intermediate nodes

b. Updating a path with a new route

EXAMPLE



	A	B	C
A	0	2	7
B	2	0	1
C	7	1	0

	A	B	C
A	0	2	3
B	2	0	1
C	3	1	0

PROGRAM

```
dist=zeros(1,20);
from=zeros(1,20);
rt=zeros(1,10);
rt=struct ;
rt.dist='';
rt.from='';
costmat=zeros(20,20);
i=0;
j=0;
k=0;
count=0;
nodes=input("Enter the number of nodes : ");
disp("Enter the cost matrix :");
```

```
for i=1:nodes
    for j=1:nodes
        fprintf('enter cost(%d,%d): ',i,j);
        costmat(i,j)=input('');
        costmat(i,i)=0;
        rt(1,i).dist(1,j)=costmat(i,j);
        rt(1,i).from(1,j)=j;
    end
end
```

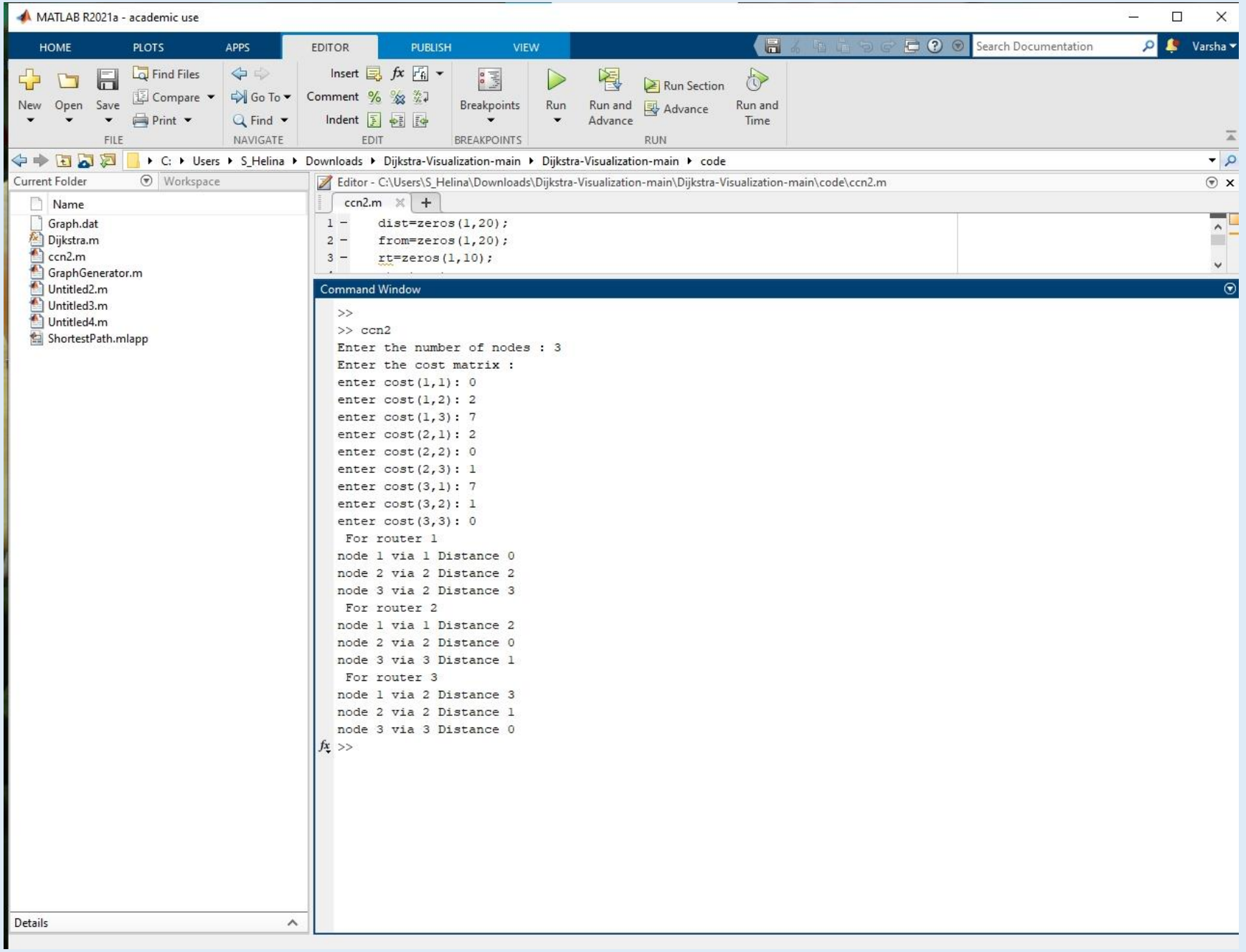
```
while 1
    count=0;
    for i=1:nodes
        for j=1:nodes
            for k=1:nodes

if(rt(1,i).dist(1,j)>costmat(i,k)+rt(1,k).dist(1,j))

rt(1,i).dist(1,j)=rt(1,i).dist(1,k)+rt(1,k).dist(1,j);
                    rt(1,i).from(1,j)=k;
                    count=count+1;
                end
            end
        end
    end
    if count==0
        break;
    end
end
```

```
for i=1:nodes
    fprintf(" For router %d",i);
    disp(' ');
    for j=1:nodes
        fprintf("node %d via %d Distance
%d",j,rt(1,i).from(1,j),rt(1,i).dist(1,j));
        disp(' ');
    end
end
end
```

OUTPUT


MATLAB R2021a - academic use
HOME
PLOTS
APPS
EDITOR
PUBLISH
VIEW
Search Documentation
Varsha
New
Open
Save
Find Files
Compare
Print
Go To
Find
Insert
Comment
Indent
Breakpoints
Run
Run and Advance
Run Section
Advance
Run and Time
FILE
NAVIGATE
EDIT
BREAKPOINTS
RUN
C: ▸ Users ▸ S_Helina ▸ Downloads ▸ Dijkstra-Visualization-main ▸ Dijkstra-Visualization-main ▸ code
Current Folder
Workspace
Name
Graph.dat
Dijkstra.m
ccn2.m
GraphGenerator.m
Untitled2.m
Untitled3.m
Untitled4.m
ShortestPath.mlapp
Details
Editor - C:\Users\S_Helina\Downloads\Dijkstra-Visualization-main\Dijkstra-Visualization-main\code\ccn2.m
ccn2.m
1 – dist=zeros(1,20);
2 – from=zeros(1,20);
3 – rt=zeros(1,10);
Command Window
>>
>> ccn2
Enter the number of nodes : 3
Enter the cost matrix :
enter cost(1,1): 0
enter cost(1,2): 2
enter cost(1,3): 7
enter cost(2,1): 2
enter cost(2,2): 0
enter cost(2,3): 1
enter cost(3,1): 7
enter cost(3,2): 1
enter cost(3,3): 0
For router 1
node 1 via 1 Distance 0
node 2 via 2 Distance 2
node 3 via 2 Distance 3
For router 2
node 1 via 1 Distance 2
node 2 via 2 Distance 0
node 3 via 3 Distance 1
For router 3
node 1 via 2 Distance 3
node 2 via 2 Distance 1
node 3 via 3 Distance 0
>>