SCAM: Scenario-based Clustering Algorithm for Mobile Ad Hoc networks

V. S. Anitha & M. P. Sebastian
National Institute of Technology Calicut
Kerala

07.01.2009

Contents

Introduction

- Ad hoc networks are wireless, infrastructure-less, multi-hop, dynamic network established by a collection of mobile nodes.

- MANET Applications

- Motivation for this Work

Main Contribution

- Proposes a scenario-based, adaptive and distributed clustering algorithm for MANETs
  - (*k*, *r*)-Dominating set based Clustering algorithm
  - Provides scalability, stability and load balancing

Related works

- Link cluster Algorithm [1]
- Lowest-ID algorithm [2]
- Highest- Degree Algorithm[3]
- Least Cluster head Change (LCC) [4]
- Distributed Clustering Algorithm [5]
- MOBIC algorithm [6]
- Connected Dominating set based algorithm [7]
- Weighted Clustering Algorithm [8]

Design Requirements

- Selection of optimum number of cluster heads
- Distributed
- Scalability
- Stability
- Mechanism to prevent the clusters from growing too large
- Cluster maintenance mechanism

Parameters for cluster head election

1. Clusterhead redundancy, parameter ***k***.
2. Bounded distance, parameter ***r***
3. Degree of the node

 Total number of nodes within the transmission range

$$\sum_{u \in V, u \neq v} \{D_{uv} < T_x, transmission\ range\}$$

Parameters …

4. Energy level of a node

   Energy depletion may lead to partitioning of the network and interruption in communication

5. Local stability of a node-

   Friss' transmission formula

   Received power ,

$$P_r = P_t * G_t * G_r * \frac{\lambda^2}{(4 * \pi * R)^2}$$

Parameters …

Therefore the approximate distance at time t is

$$D_t^{v,u} = \frac{k}{\sqrt{P_r}}$$

$$RM_t^{v,u} = D_t^{v,u} - D_{t-1}^{v,u}$$

$$SDRM = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(RM_i - \overline{RM})^2}$$

where

$$\overline{RM} = \frac{1}{N}(RM_1^{v,u} + RM_2^{v,u} + ... + RM_N^{v,u})$$

Local stability of a node is LSTAB

Parameters …

$$Q_y = W_1 * DG_v + W_2 * B_v + W_3 * ILSTAB$$

W_1, W_2 and W_3 are weights associated with various factors affecting the quality

Suitable values can be assigned based on the required application.

The Scenario-based Clustering Algorithm for Mobile Ad Hoc Networks (SCAM)

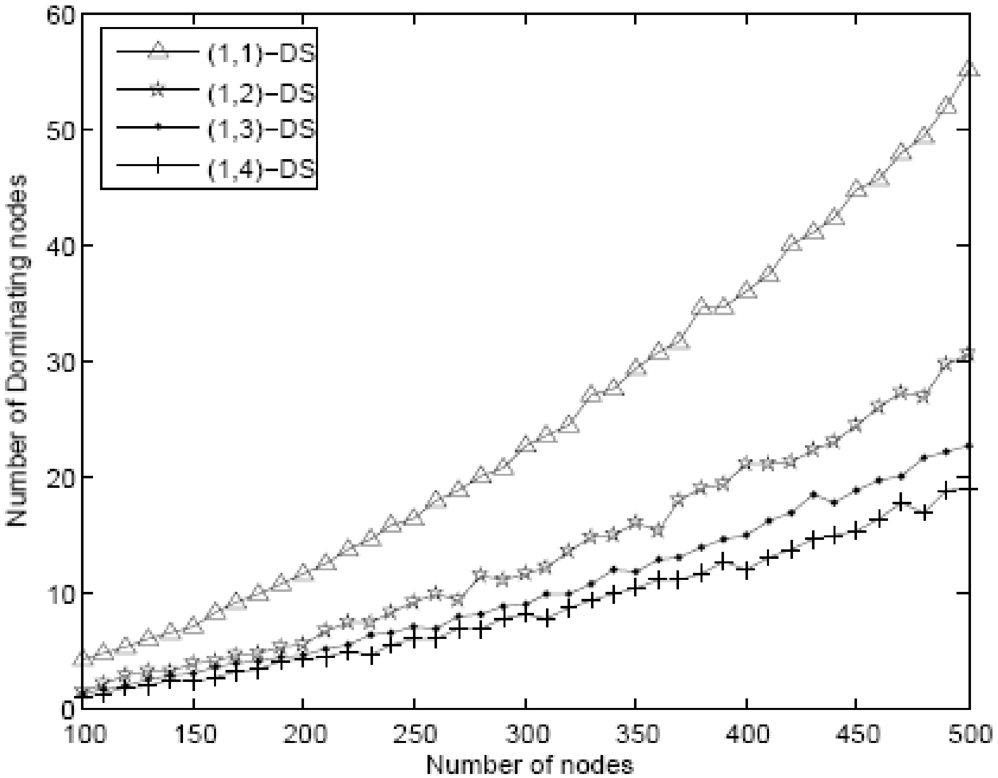
1. Dominating set computation
2. Quality computation
3. Quality communication
4. Cluster head selection
5. Cluster head association
6. Cluster maintenance

Properties of SCAM

- Redundancy property
- Bounded distance property
- Weight based cluster head affiliation
- Cluster maintenance property

Performance Evaluation

Number of dominating nodes for k = 1 and r = 1, 2, 3 & 4



Observations

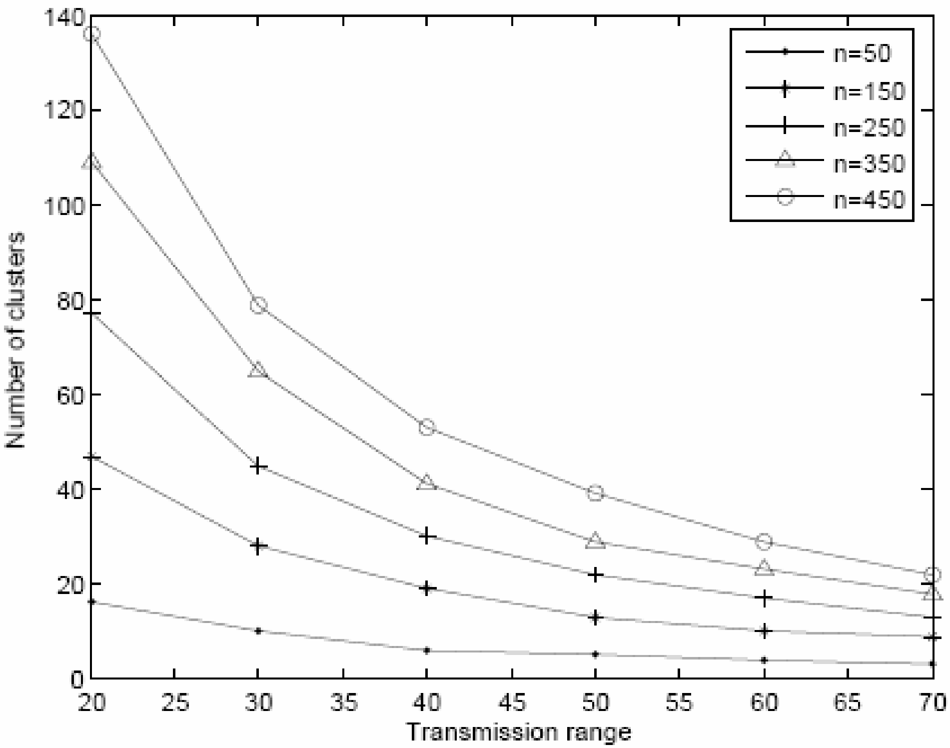
1. Number of dominating nodes created decreases with increase in cluster diameter.
2. When the value of r is increased from 1 to 2, the number of dominating nodes is reduced by 51%.

Number of nodes : 100-500 Network Size : 1000 X 1000 m^2

Transmission range : 20 m

Performance Evaluation …

Transmission range Vs number of clusters



Number of nodes : 100-500 Network Size : $500 \times 500\ m^2$

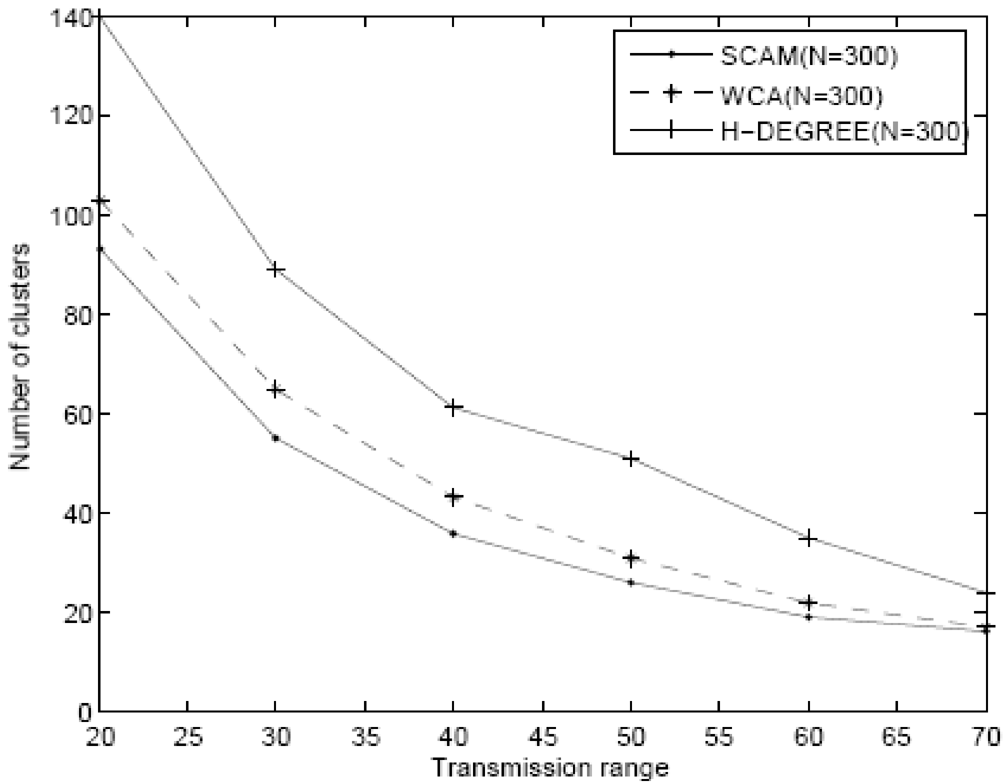
$1000 \times 1000\ m^2$

Observations

1. The average number of clusters is relatively high when the transmission range is high.

Performance Evaluation…

Number of clusters in SCAM, WCA and H_DEGREE



Observations

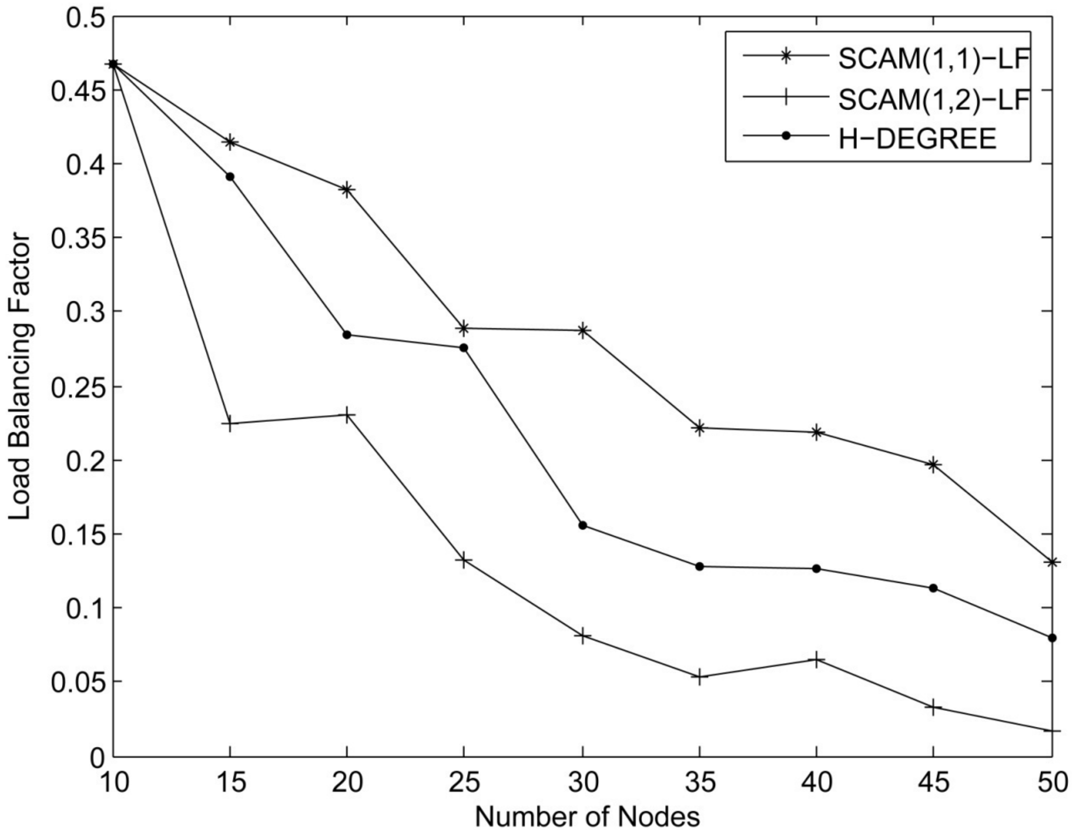
1. SCAM creates less number of clusters with increase in the value of ***r***.
2. Increase in the value of ***r*** leads to increase in cluster size, Which adversely affects the performance.

Number of nodes : 300 Network Size : 1000 X 1000 m^2

K =2, r = 2

Performance Evaluation…

Load balancing factor in SCAN & H_DEGREE



Observations

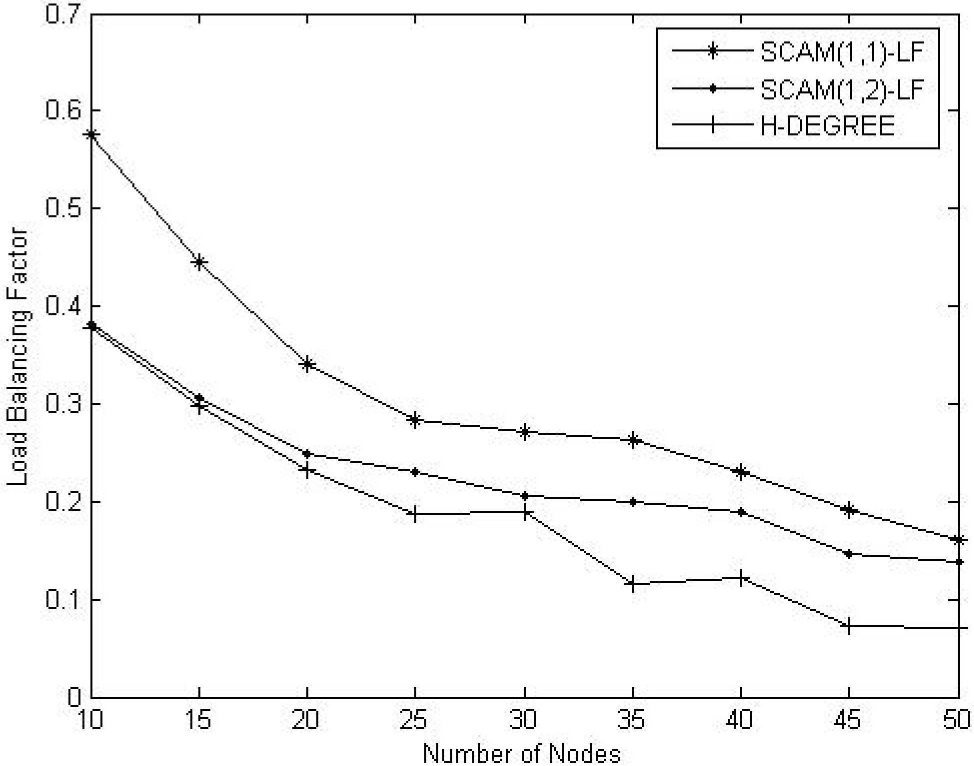
1. Higher value of LBF signifies a better load distribution.
2. SCAM(1,1) gives better load balancing than H-Degree.
3. Result shows LBF without putting a limit on total number of nodes a clusterhead can handle.

Number of nodes : 50 Network Size : 100 X 100m^2

K =1, r = 1, 2

Performance Evaluation …

Load balancing factor in SCAN & H_DEGREE



Observations

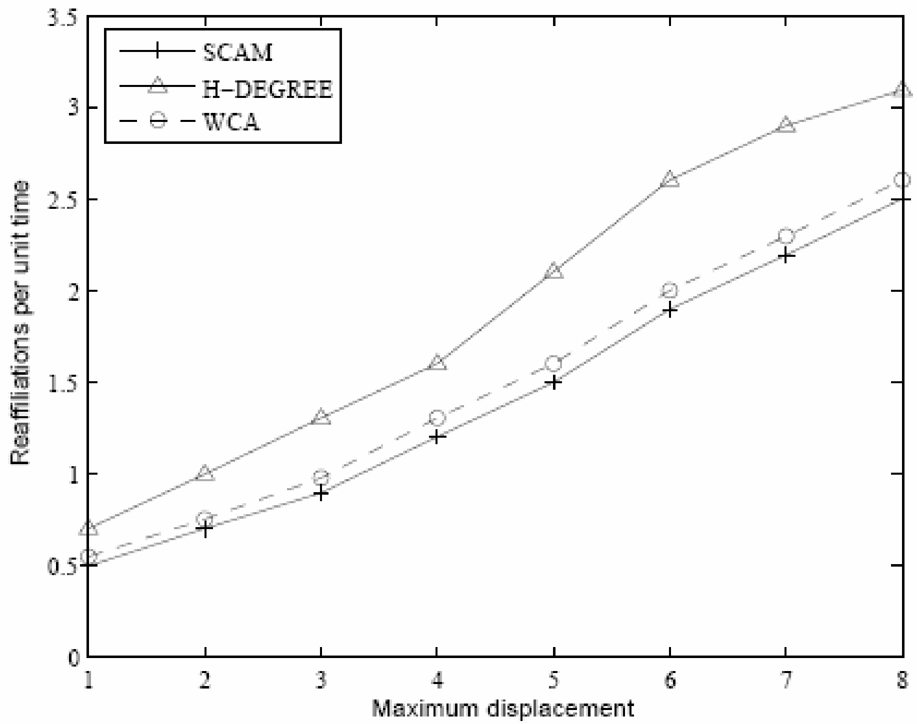
1. Max number of nodes a clusterhead can handle is limited 7.
2. Better LBF for SCAM(1,1) and SCAM(1,2).
3. SCAM is multi-clusterhead bounded distance algorithm

Number of nodes : 50 Network Size : 100 X 100m^2

K =1, r = 1, 2 Max CH degree : 7

Performance Evaluation …

Number of re-affiliations per unit time



Number of nodes : 50 Network Size : 100 X 100m²
Max CH degree : 7

Observations

1. Clusters created by SCAM preserve its structure for a longer period than H-DEGREE
2. The number of re-affiliations increase with increase in number of nodes for higher values of displacement.

Conclusions

- A scenario-based protocol, SCAM, which creates clusters by accepting various parameters and weights, is proposed and its performance is evaluated
- SCAM allows variable diameter clusters with cluster head redundancy to achieve scalability
- SCAM selects the most capable node as cluster head to prolong the lifetime of clusters

Conclusions (cont..)

- Proper load balancing is achieved by putting a limit on the total number of nodes a cluster head can handle
- SCAM uses techniques to maintain the cluster structure as stable as possible with less control messages

Future Work

1. Incorporate security features (this can be done by finding an algorithm for decentralized certification and authentication)

2. Provide the required QoS on demand

3. Enhancement for use in heterogeneous mesh networks

References

[1] Dennis J. Baker and Antony Ephremides, “The Architectural Organization of a Mobile Radio Network via a Distributed Algorithm”, IEEE Transactions on communications, 29(11), 1694-1701, 1981.

[2] Antony Ephremides, Jeffery E. Wieseltheir and Dennis J. Baker, “A Design Concept for Reliable Mobile Radio Networks with Frequency Hopping Signalling”, Proceedings of the IEEE vol. 75 No.1, pages 56-73, 1987.

[3] A. Parekh “Selecting Routers in Ad hoc Networks”, Proceedings of IEEE International Telecommunications symposium, 1994.

[4] Ching-Chuan and Mario Gerla “ Routing in Clustered Multi-hop Mobile Wireless Network with Fading Channel”, Proceedings IEEE SICON’97, 1997

[5] Stefano Basagni “Distributed Clustering for Mobile Ad hoc Network”, Proceedings of ISPAN 99, pp 310-315, 1999.

References …

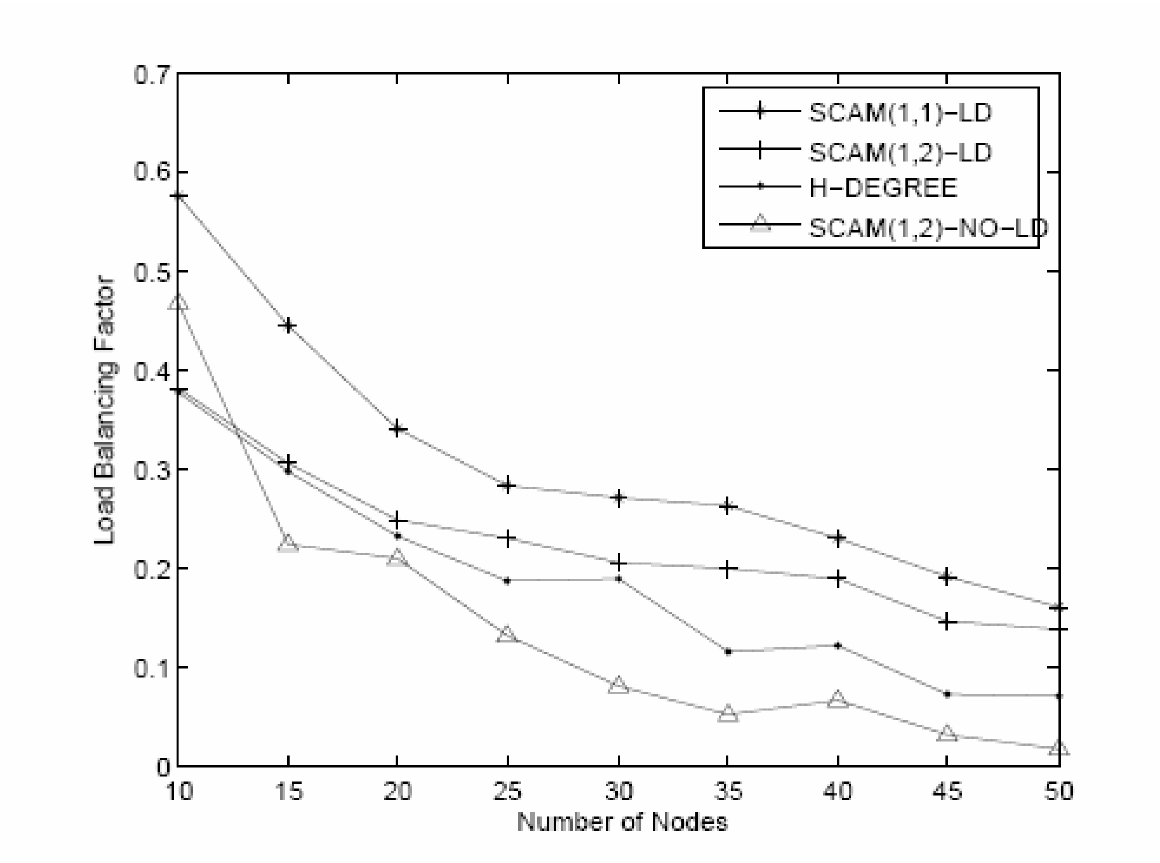
[6] Prithwish Basu, Naved Khan and Thomas D. Little, “A mobility based Metric for Clustering in Mobile Ad hoc Networks”, Proceedings IEEE ICDCW 01, pp 413-418.

[7] F.Dai,Wu, “On constructing k-connected, K-dominating set in wireless networks “. Proceedings IEEE IPDPS, 2005.

[8] Mainak Chaterjee, Sajal K. Das and Dalma Turgut, “ WCA: A weighted Clustering Algorithm for Mobile Ad Hoc Networks”, Cluster Computing, Kluwer Academic Publishers, 2006.

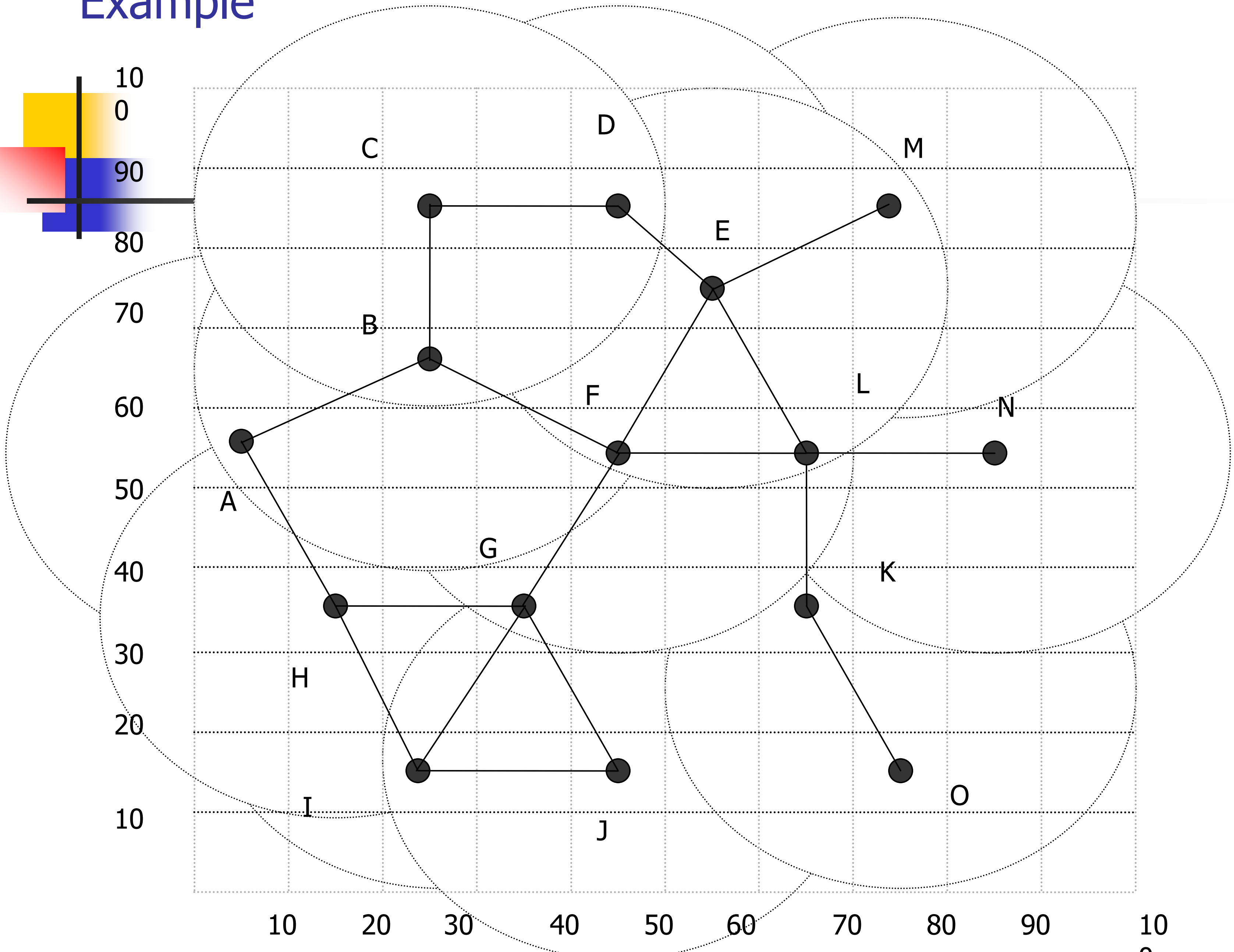
[9] M.A. Spohn and J.J. Garcia-Luna-Aceves, “Bounded-distance multi-clusterhead formation in wireless ad hoc networks”, Elsevier Ad Hoc Networks 5 (2007), pp. 504-530.

Thank you



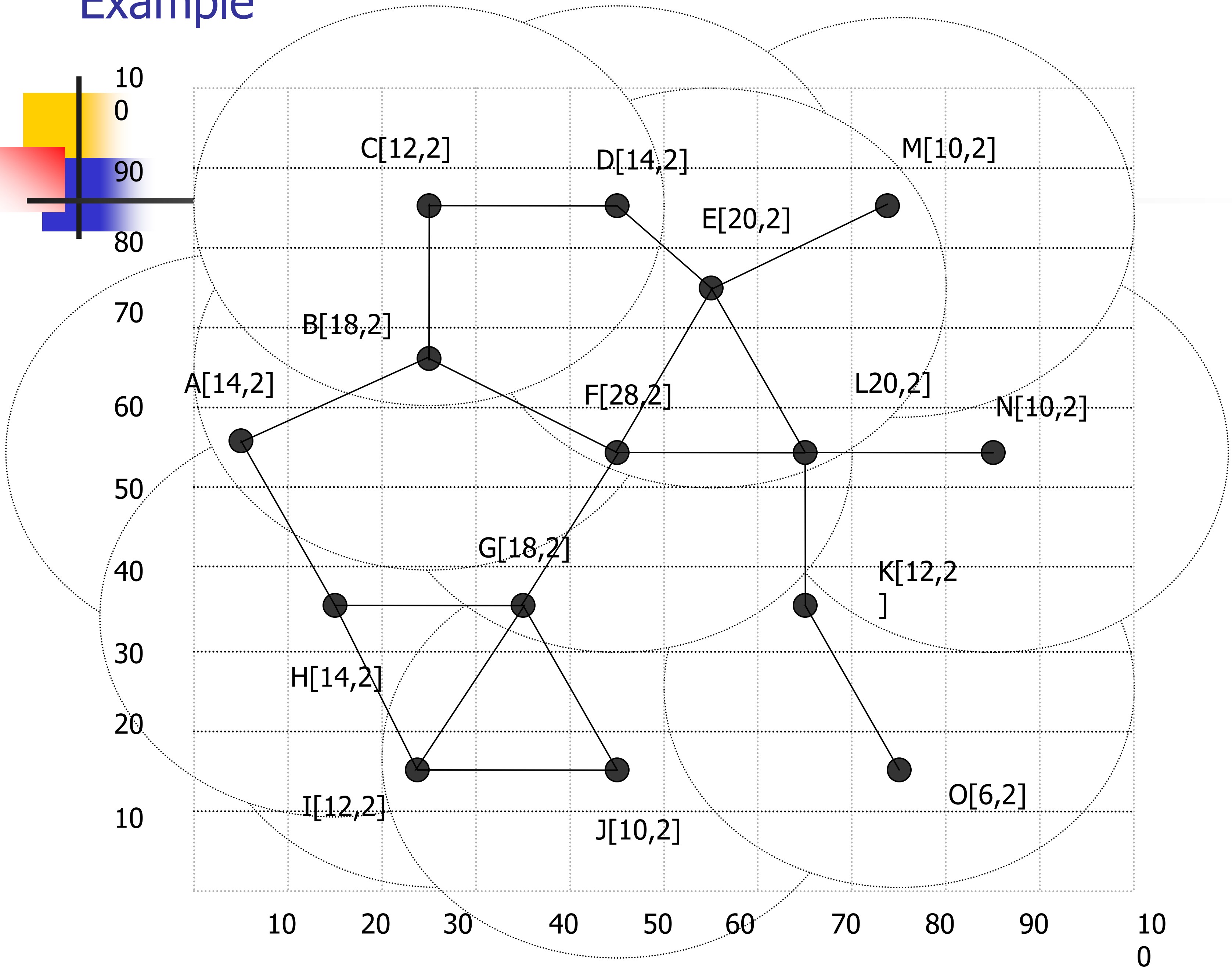
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
10
15
20
25
30
35
40
45
50
Number of Nodes
Load Balancing Factor
SCAM(1,1)−LD
SCAM(1,2)−LD
H−DEGREE
SCAM(1,2)−NO−LD

Example



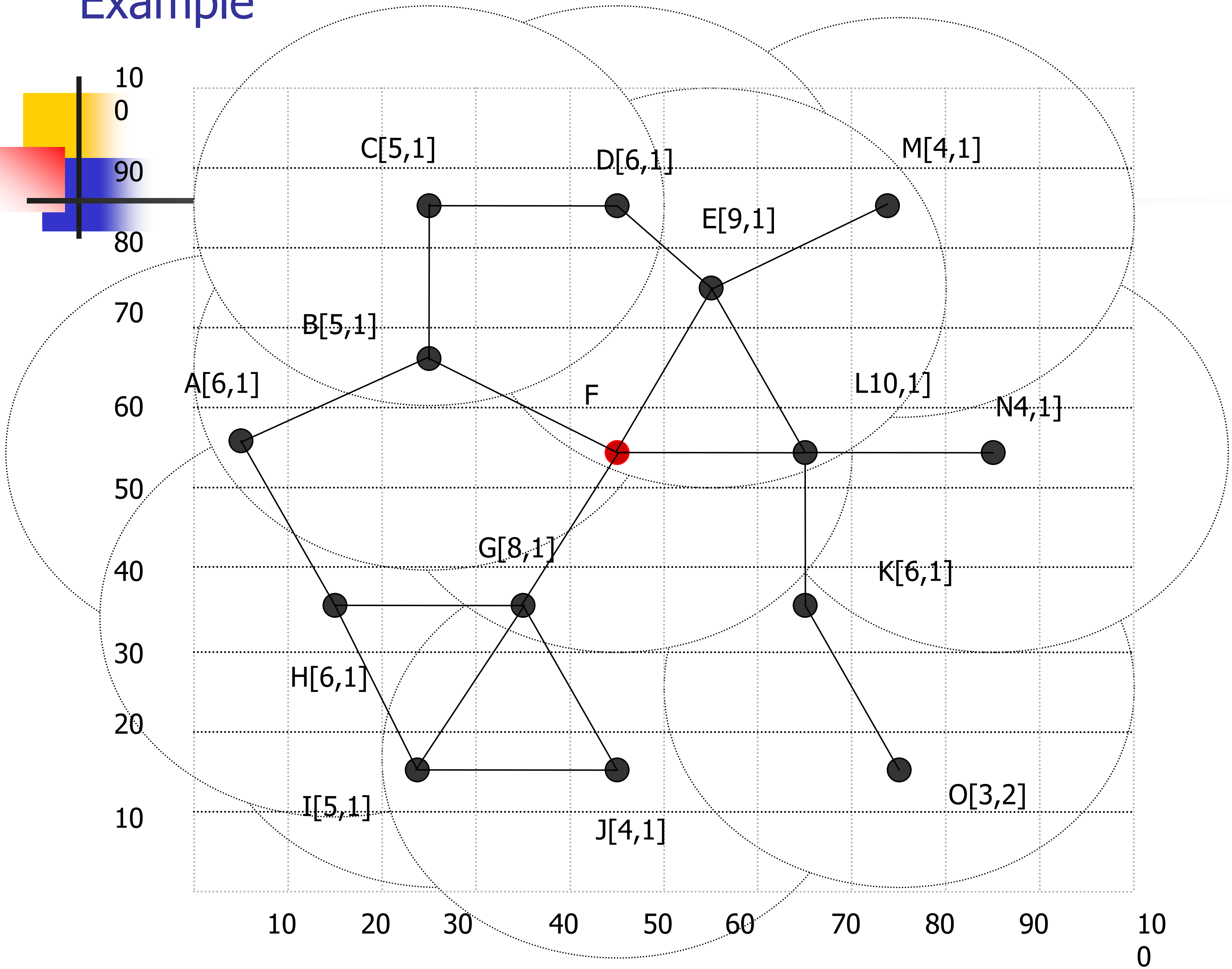
C
D
M
E
B
L
F
N
A
G
K
H
I
J
O
100
90
80
70
60
50
40
30
20
10
10
20
30
40
50
60
70
80
90
100

Example



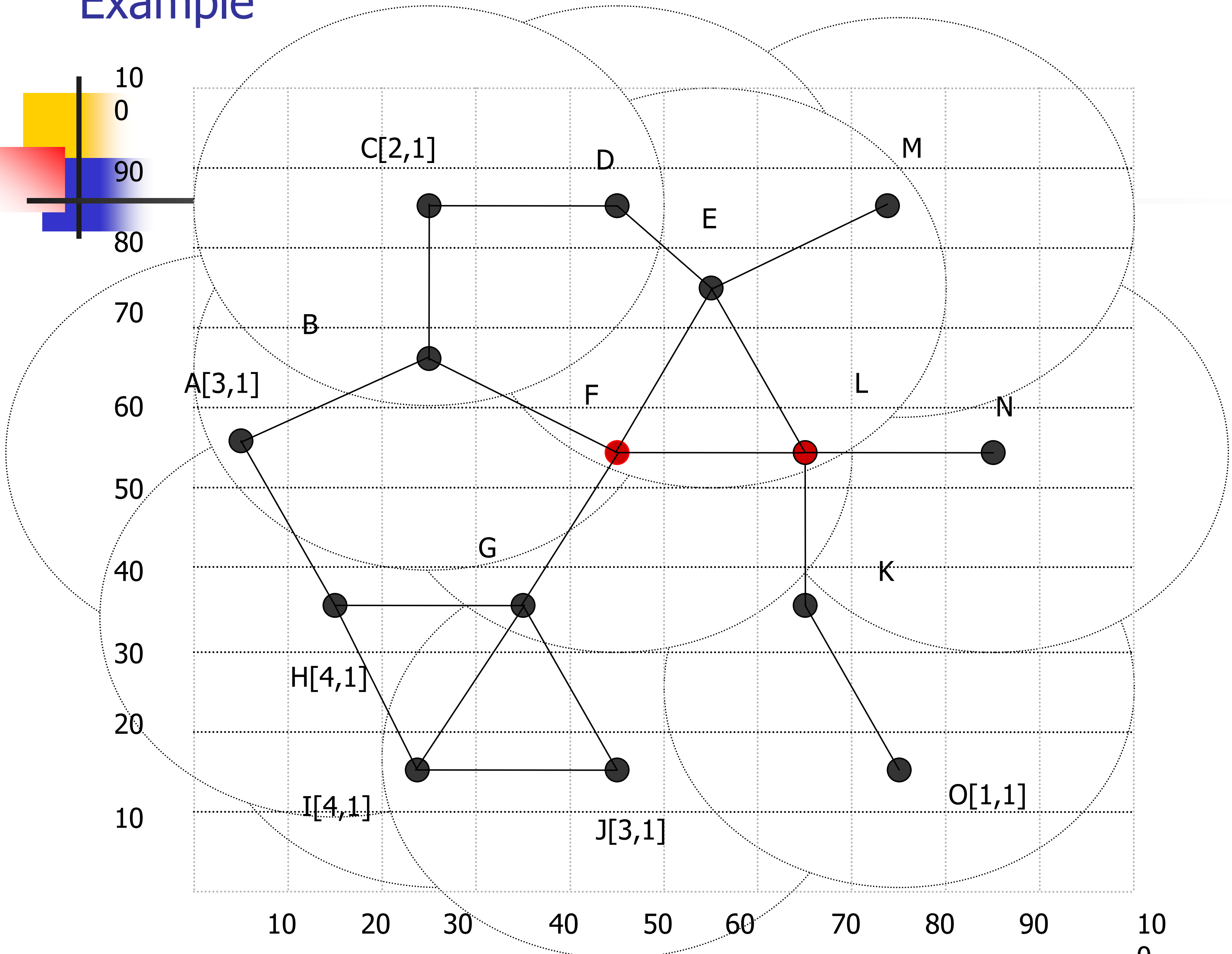
100
90
80
70
60
50
40
30
20
10
10
20
30
40
50
60
70
80
90
100
C[12,2]
D[14,2]
M[10,2]
E[20,2]
B[18,2]
A[14,2]
F[28,2]
L20,2]
N[10,2]
G[18,2]
K[12,2
]
H[14,2]
I[12,2]
J[10,2]
O[6,2]

Example



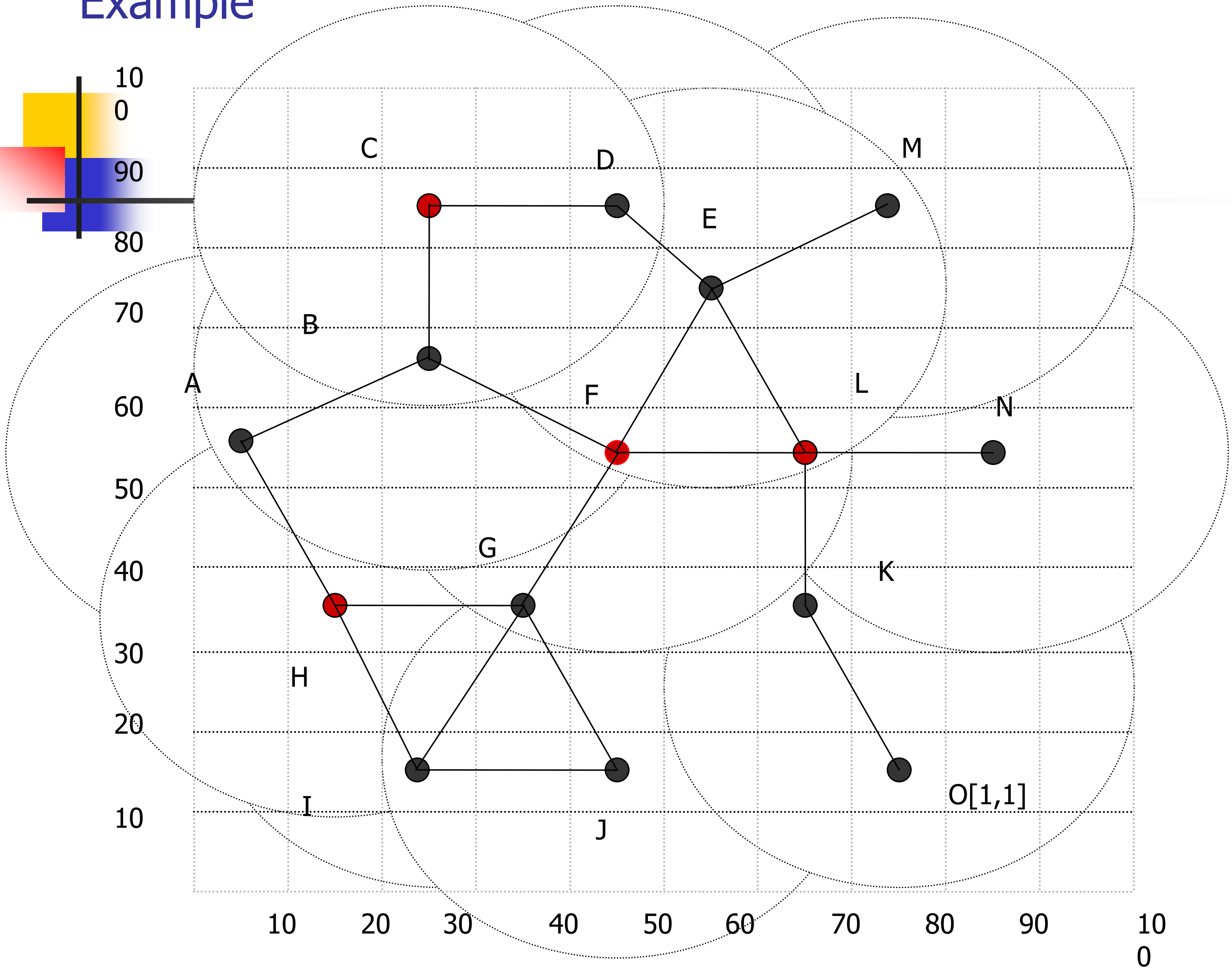
10
0
90
80
70
60
50
40
30
20
10
C[5,1]
D[6,1]
M[4,1]
E[9,1]
B[5,1]
A[6,1]
F
L10,1]
N4,1]
G[8,1]
K[6,1]
H[6,1]
I[5,1]
J[4,1]
O[3,2]
10
20
30
40
50
60
70
80
90
10
0

Example



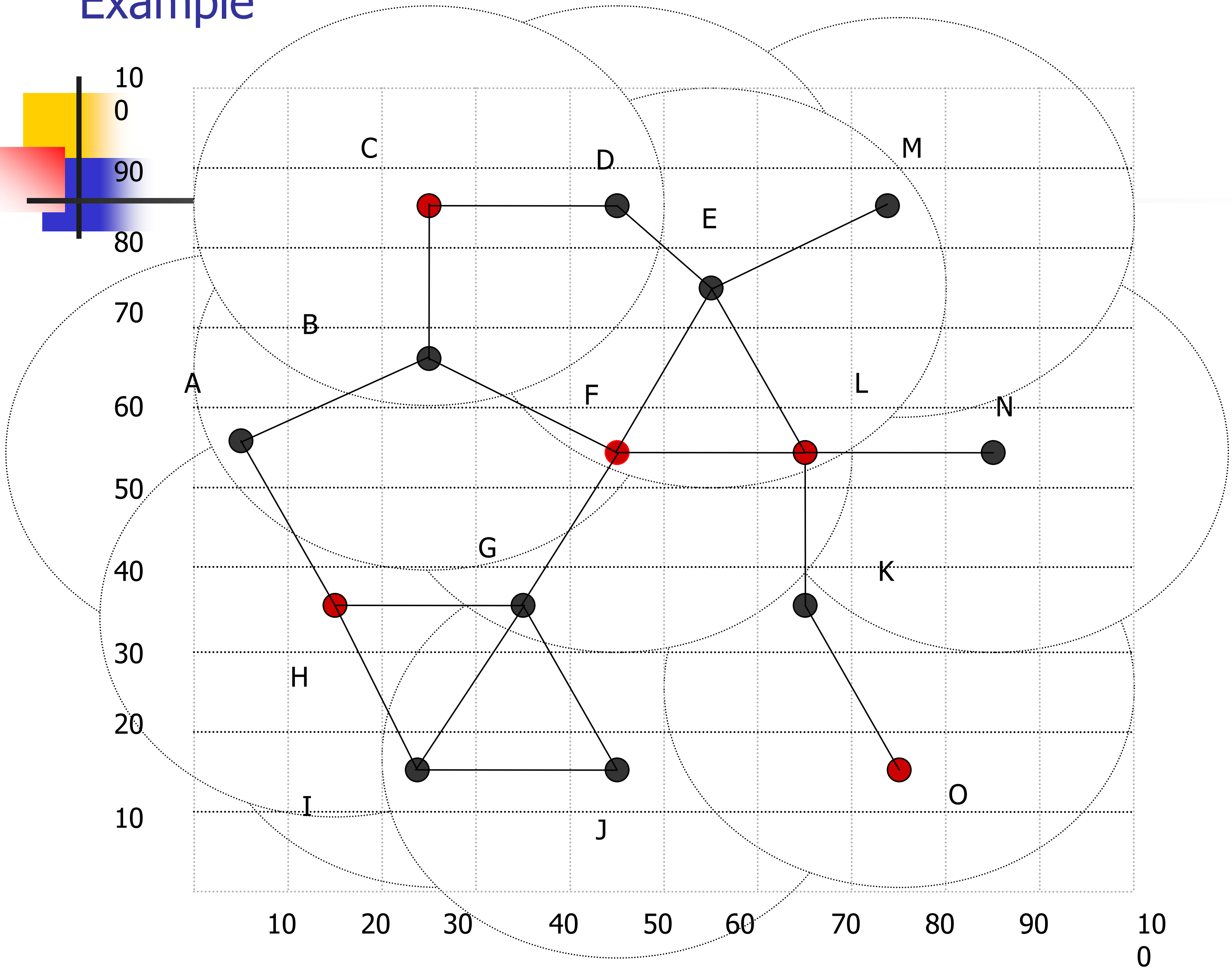
100
90
80
70
60
50
40
30
20
10
C[2,1]
D
M
E
B
A[3,1]
F
L
N
G
K
H[4,1]
I[4,1]
J[3,1]
O[1,1]
10
20
30
40
50
60
70
80
90
100

Example



10
0
90
80
70
60
50
40
30
20
10
C
D
M
E
B
A
F
L
N
G
K
H
I
J
O[1,1]
10
20
30
40
50
60
70
80
90
10
0

Example



100
90
80
70
60
50
40
30
20
10
10
20
30
40
50
60
70
80
90
100
A
B
C
D
E
F
G
H
I
J
K
L
M
N
O

Cluster head Selection

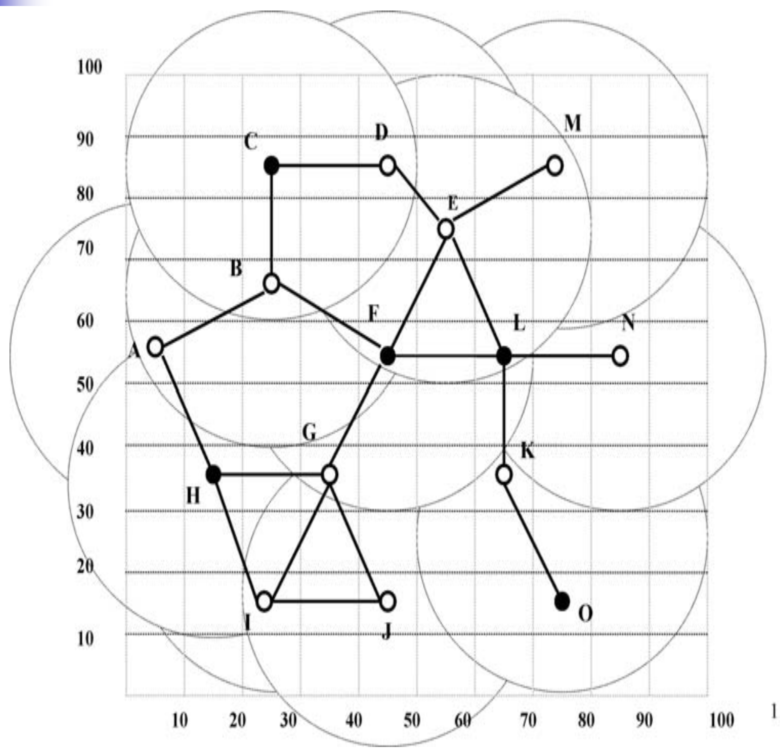
C	2.47	0.41
F	1.07	0.93
H	1.52	0.66
L	1.3	0.77
O	1	1

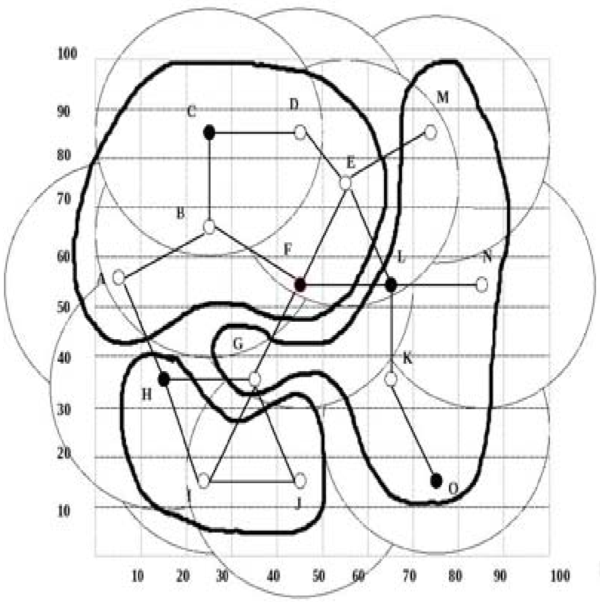
Quality of dominating nodes

Nodes	Chs	T1	T2	T3	T4
A	H,C,F	F	H	H	H
B	H,C,F,L	F	L	L	H
C	F	F			
D	C,F,L	F	L	L	C
E	C,F,L	F	L	L	C
F			L	L	C
G	H,F,L	F	L	L	H
H	F	F			
I	H,F	F	H	H	H
J	H,F	F	H	H	H
K	F,L,O	O	O	L	-
L	O	O			-
M	F,L	O	O	L	-
N	F,L,O	O	O	L	-

Clusterhead association of various nodes

Cluster formation





Algorithm 1: A greedy algorithm (KR) to find the Dominating Set

```
Algorithm- (k,r) DS algorithm
Data: G = ( V, E ) and parameters r and k
Result: D, Dominating Set
begin
    initialization;
    DS = φ;
    for each u ∈ V do
        n =
        Find_Weighted_Numberof_Rhop_Neighbour(u);
        /* 1+Number of single hop nodes*r+Number of
        2-hop nodes*(r-1)+....Number of r-hop nodes*1 */
        u.weight = k * n;
        u.Required_Ch = k;
        Insert_Priority_Heap( H, u);
    end
    while ( H ≠ ∅ ) do
        d = del_root(H);
        /* Root node has the largest weight and is the
        best choice for Dominating node */
        d.Required_Ch = 0;
        DS = DS ∪ d;
        /*Append dominating set*/
        Find_Rhop_Neighbous(d, G, V_rd);
        for ( u ∈ V_rd) and (u ∈ H) do
            u.Required_Ch = u.Required_Ch − 1;
            if u.Required_Ch == 0 then
                delete(H, u)
                /* A node is said to be (k,r) dominated if
                that node has at least k neighbours within
                r-hop distance */
            end
        end
        Find_2rhop_Neighbour(d, G, V_2rd);
        /*Required_Ch of all nodes within r-hop distance
        from dominating node d is reduced by one and
        that affects the weight value of nodes within 2r
        distance from d*/
        for each (u ∈ H) and ( u ∈ V_2rd) do
            u.weight = 0;
            Find_Rhop_Neighbous(u, G, V_rd);
            for each ( w ∈ V_rd) do
                u.weight = u.weight + w.Required_Ch *
                h;
            end
        end
        makeheap(H);
    end
end
```

Algorithm analysis

Let G = (V, E) where |V| = no. of vertices

The first for loop takes O(|V| log|V|)

Initialization - O(|V|) times and heap insertion – O(log|V|) times.

While loop executes |V| times

For loop in line 12 – O(k) where k<=n-1

For loop in line 16 – O(k)

For loop in line 19- O(k)

i.e. Weight recalculation takes $O(k^{2})$

Hence complexity is $O(n^3)$ (k=n)

Correctness of algorithm

Initially heap is empty. After initialization all nodes are inserted into the heap and hence the size of the heap is |V|.

Heap is created in such a way that the root node has the highest weight. This node is a dominating node and is deleted.(Let the set of such nodes be D1). Nodes other than root nodes are also deleted if it has enough number of clusterheads within r-hop, let that set be D2.

Algorithm repeat till the heap become empty ie. V= D1 U D2.

Therefore when the program terminates we get two sets of nodes one set representing the dominating nodes and other set representing dominated nodes.