DESEMPENHO E DIMENSIONAMENTO DE REDES

ASSIGNMENT GUIDE NO. 3

AVAILABILITY PERFORMANCE OF MULTI-HOP WIRELESS NETWORKS WITH MOBILE TERMINALS

Realizado por:

- Fábio Alves, Nmec: 84734
- Sérgio de Aguiar, Nmec: 84831

2. Simulator Development

Funções desenvolvidas:

```
function counter= InitializeCounter(N)
% counter - a row vector with N values to count the number of time slots
%           such that each mobile node has Internet access
% This function creates the row vector 'counter' and initializes it with
zeros
% in all positions.
    counter = zeros(1,N);
end
```

```
function counter= UpdateCounter(C,counter)
% This function increments the values of the 'counter' positions of the
% mobile nodes that have Internet access.
counter=counter+C;
end
```

```
function L= ConnectedList(pos,W,AP)
% Input:
%   pos -  a matrix with N rows and 2 columns; each row identifies the
(x,y)
%          coordinates of each mobile node
%   W -    radio range (in meters)
%   AP -   coordinates of Access Points (APs)
% Output:
%   L -    a matrix with 2 columns; each row identifies a pair of nodes
%          (mobile nodes and AP nodes) with a direct wireless link
%          between them

    L = [];
    W = W^2;
    aux = [pos ; AP];

    for k = 1:size(pos,1)
        for  l = k+1:size(aux,1)
            elem1 = aux(k,:);
            elem2 = aux(l,:);
            dist = (elem1(1)-elem2(1))^2 + (elem1(2)-elem2(2))^2;

            if dist <= W
                L = [L;[l,k]];
            end
        end
    end
end
```

```matlab
function C= ConnectedNodes(L,N,AP)
% Input:
%   L -  a matrix with 2 columns; each row identifies a pair of nodes
%         (mobile nodes and AP nodes) with a direct wireless link
%         between them
%   N -  no. of mobile nodes
%   AP - coordinates of Access Points (APs)
% Output:
%   C - an array with N values (one for each mobile node) that is 1 in
%       position i if mobile node i has Internet access
%
% NOTE: To develop this function, check MATLAB function 'distances' that
%       computes shortest path distances of a graph.
%

    C = false(1, N);

    num_ap = size(AP,1);
    id_last_ap = N + num_ap;

    tmp1= N+1:id_last_ap;
    tmp2= N+1:id_last_ap;

    col1 = [ L(:,1) ; tmp1(:) ];
    col2 = [ L(:,2) ; tmp2(:) ];

    grap = graph(col1,col2);
    dis= distances(grap,1:N,N+1:id_last_ap);

     for i=1:size(dis,1)
        for j=1:num_ap
            if not(dis(i,j)==Inf)
                C(1,i) = true;
            end
        end
     end
end
```

3. Assignment tasks:

Código escrito para recolha de dados:

```
Result = zeros(24,6);
Conf = zeros(24,6);

Case = zeros(10,2);
Aps = [ 250,100;
        150,100;
        350,100;
        50,50;
        250,100;
        450,150];

 for ap = 1:3
     for w = 1:3
         for s = 1:2
             for n = 1:4
                 for k = 1:10
                    [AvgAvail, MinAvail]= simulatorFunction(n*20, s*3,
w*20 + 20, 1, 7200, Aps(2^(ap-1):2^(ap-1)+ap-1,1:2), 0);
                    Case(k,1) = AvgAvail;
                    Case(k,2) = MinAvail;
                 end

                Result((8*w-8)+(4*s-4)+n, 2*ap-1) = mean(Case(:,1));
                Result((8*w-8)+(4*s-4)+n, 2*ap) = mean(Case(:,2));
                Conf((8*w-8)+(4*s-4)+n, 2*ap-1) = 1.645*std(Case(:,
1))/sqrt(10);
                Conf((8*w-8)+(4*s-4)+n, 2*ap) = 1.645*std(Case(:,
2))/sqrt(10);

                Result
                Conf
             end
         end
     end
 end
```

a)

Para 1 AP:

Case	N	S (km/h)	W (meters)	**Average availability**	**90% confidence interval**	**Minimum availability**	**90% confidence interval**
1	20	3	40	0.0791	0.0164	0.0014	0.0018
2	40	3	40	0.1377	0.0104	0.0077	0.0036
3	60	3	40	0.2387	0.0240	0.0462	0.0148
4	80	3	40	0.4145	0.0211	0.1903	0.0434
5	20	6	40	0.0795	0.0066	0.0098	0.0073
6	40	6	40	0.1344	0.0074	0.0303	0.0160
7	60	6	40	0.2415	0.0171	0.0623	0.0231
8	80	6	40	0.4197	0.0129	0.1921	0.0326

9	20	3	60	0.2738	0.0084	0.0825	0.0325
10	40	3	60	0.6052	0.0239	0.3811	0.0382
11	60	3	60	0.8782	0.0108	0.7366	0.0268
12	80	3	60	0.9736	0.0032	0.9102	0.0110
13	20	6	60	0.2973	0.0242	0.1366	0.0481
14	40	6	60	0.6025	0.0108	0.4021	0.0585
15	60	6	60	0.8901	0.0103	0.7859	0.0394
16	80	6	60	0.9759	0.0033	0.9363	0.0085
17	20	3	80	0.6082	0.0235	0.4508	0.0365
18	40	3	80	0.9418	0.0081	0.8574	0.0213
19	60	3	80	0.9944	0.0017	0.9680	0.0099
20	80	3	80	0.9995	0.0002	0.9897	0.0036
21	20	6	80	0.6136	0.0240	0.4230	0.0634
22	40	6	80	0.9298	0.0077	0.8523	0.0299
23	60	6	80	0.9946	0.0013	0.9785	0.0031
25	80	6	80	0.9995	0.0003	0.9933	0.0012

Para 2 APs:

Case	N	S **(km/h)**	W **(meters)**	**Average availability**	**90% confidence interval**	**Minimum availability**	**90% confidence interval**
1	20	3	40	0.1415	0.0072	0.0314	0.0213
2	40	3	40	0.2666	0.0118	0.0730	0.0257
3	60	3	40	0.4255	0.0204	0.2107	0.0325
4	80	3	40	0.6239	0.0110	0.3541	0.0598
5	20	6	40	0.1445	0.0070	0.0496	0.0246
6	40	6	40	0.2637	0.0092	0.1351	0.0384
7	60	6	40	0.4311	0.0114	0.2228	0.0350
8	80	6	40	0.6325	0.0098	0.4031	0.0467
9	20	3	60	0.5027	0.0214	0.2934	0.0646
10	40	3	60	0.7996	0.0057	0.5945	0.0568
11	60	3	60	0.9537	0.0038	0.8488	0.0242
12	80	3	60	0.9877	0.0022	0.9411	0.0120
13	20	6	60	0.4965	0.0090	0.3916	0.0360
14	40	6	60	0.8083	0.0113	0.6852	0.0244
15	60	6	60	0.9464	0.0039	0.8586	0.0361
16	80	6	60	0.9877	0.0008	0.9590	0.0045
17	20	3	80	0.8289	0.0092	0.6731	0.0488
18	40	3	80	0.9778	0.0027	0.9221	0.0130
19	60	3	80	0.9977	0.0007	0.9780	0.0053
20	80	3	80	0.9997	0.0001	0.9926	0.0029
21	20	6	80	0.8351	0.0052	0.7334	0.0277
22	40	6	80	0.9776	0.0017	0.9353	0.0111
23	60	6	80	0.9978	0.0004	0.9845	0.0025
25	80	6	80	0.9997	0.0001	0.9948	0.0017

Para 3 APs:

Case	N	S (km/h)	W (meters)	Average availability	90% confidence interval	Minimum availability	90% confidence interval
1	20	3	40	0.2215	0.0195	0.0717	0.0253
2	40	3	40	0.3408	0.0138	0.1634	0.0269
3	60	3	40	0.5027	0.0140	0.2400	0.0375
4	80	3	40	0.7094	0.0079	0.4718	0.0683
5	20	6	40	0.2206	0.0123	0.0757	0.0281
6	40	6	40	0.3330	0.0112	0.1611	0.0361
7	60	6	40	0.5142	0.0077	0.3357	0.0439
8	80	6	40	0.7187	0.0066	0.5748	0.0321
9	20	3	60	0.5932	0.0190	0.3874	0.0539
10	40	3	60	0.8811	0.0064	0.7352	0.0375
11	60	3	60	0.9748	0.0020	0.9018	0.0155
12	80	3	60	0.9940	0.0006	0.9603	0.0071
13	20	6	60	0.5851	0.0173	0.4178	0.0556
14	40	6	60	0.8844	0.0041	0.8010	0.0202
15	60	6	60	0.9723	0.0016	0.9180	0.0107
16	80	6	60	0.9942	0.0006	0.9709	0.0079
17	20	3	80	0.9048	0.0078	0.8173	0.0173
18	40	3	80	0.9900	0.0015	0.9491	0.0129
19	60	3	80	0.9984	0.0004	0.9806	0.0045
20	80	3	80	0.9998	0.0001	0.9926	0.0054
21	20	6	80	0.9048	0.0044	0.8215	0.0263
22	40	6	80	0.9897	0.0010	0.9635	0.0048
23	60	6	80	0.9987	0.0004	0.9904	0.0024
25	80	6	80	0.9998	0.0001	0.9942	0.0021

a) Não influencia uma vez que a velocidade não tem influencia direta sobre a conexao entre nodes. Esta é apenas alterada devido ao range, à posição e ao número de dispositivos.

b) Quanto maior o range, maior será a disponibilidade média, uma vez que quanto maior for o range, maior será a área na qual os dispositivos se conseguem conectar, ou seja, existirão mais conexões entre estes.

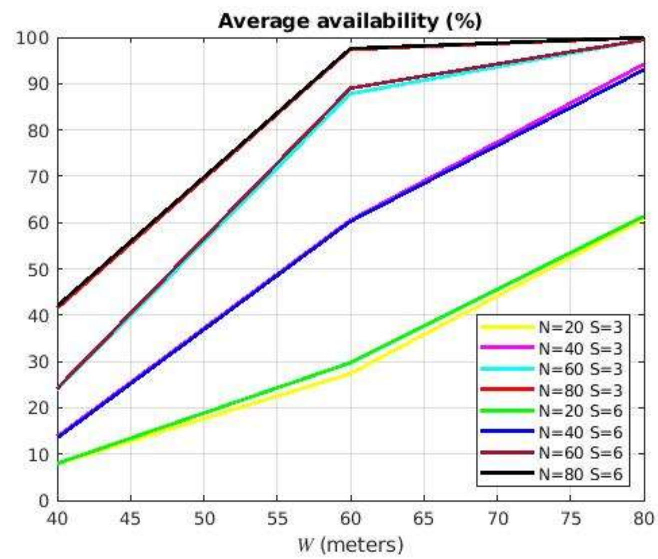

Figura 1 - Disponibilidade média para 1 AP.

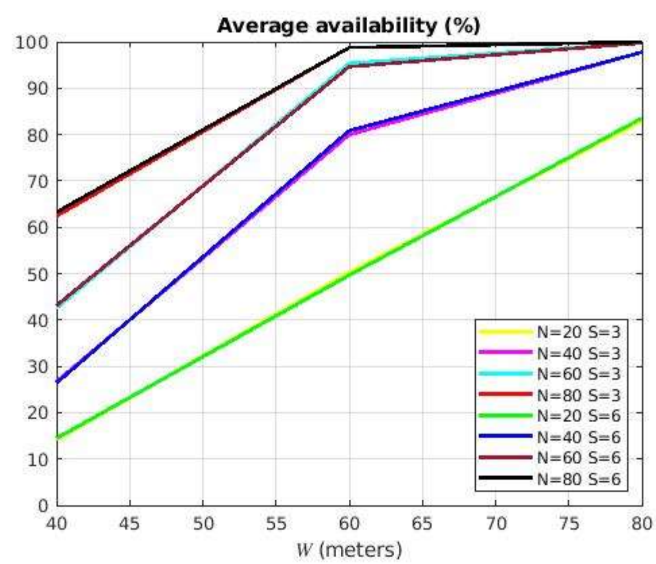

Figura 2 - Disponibilidade média para 2 APs.

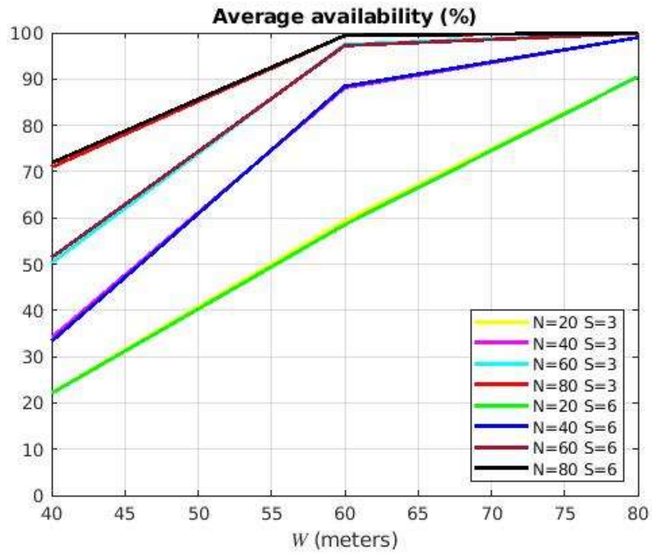

Figura 3 - Disponibilidade média para 3 APs.

```
%1 AP

 x = [40, 60, 80];

 Y1 = [0.0791, 0.2738, 0.6082].*100;
 Y2 = [0.1377, 0.6052, 0.9418].*100;
 Y3 = [0.2387, 0.8782, 0.9944].*100;
 Y4 = [0.4145, 0.9736, 0.9995].*100;
 Y5 = [0.0795, 0.2973, 0.6136].*100;
 Y6 = [0.1344, 0.6025, 0.9298].*100;
 Y7 = [0.2415, 0.8901, 0.9946].*100;
 Y8 = [0.4197, 0.9759, 0.9995].*100;

 figure('Name','Radio range')
 l=plot(x,Y1,'y-',x,Y2,'m-',x,Y3,'c-',x,Y4,'r-',x,Y5,'g-',x,Y6,'b-
',x,Y7,'-',x,Y8,'k-');
 for k=1:8
    l(k).LineWidth = 2;
 end
 title('Average availability (%)')
 xlabel('W (meters)')
 legend('N=20 S=3','N=40 S=3','N=60 S=3','N=80 S=3','N=20 S=6','N=40
S=6','N=60 S=6','N=80 S=6','location','southeast')
 axis([40 80 0 100])
 grid on
```

```
%2 AP

  x = [40, 60, 80];

 Y1 = [0.1415, 0.5027, 0.8289].*100;
 Y2 = [0.2666, 0.7996, 0.9778].*100;
 Y3 = [0.4255, 0.9537, 0.9977].*100;
 Y4 = [0.6239, 0.9877, 0.9997].*100;
 Y5 = [0.1445, 0.4965, 0.8351].*100;
 Y6 = [0.2637, 0.8083, 0.9776].*100;
 Y7 = [0.4311, 0.9464, 0.9978].*100;
 Y8 = [0.6325, 0.9877, 0.9997].*100;

 figure('Name','Radio range')
 l=plot(x,Y1,'y-',x,Y2,'m-',x,Y3,'c-',x,Y4,'r-',x,Y5,'g-',x,Y6,'b-
',x,Y7,'-',x,Y8,'k-');
 for k=1:8
    l(k).LineWidth = 2;
 end
 title('Average availability (%)')
 xlabel('W (meters)')
 legend('N=20 S=3','N=40 S=3','N=60 S=3','N=80 S=3','N=20 S=6','N=40
S=6','N=60 S=6','N=80 S=6','location','southeast')
 axis([40 80 0 100])
 grid on



%3 AP

  x = [40, 60, 80];

 Y1 = [0.2215, 0.5932, 0.9048].*100;
 Y2 = [0.3408, 0.8811, 0.9900].*100;
 Y3 = [0.5027, 0.9748, 0.9984].*100;
 Y4 = [0.7094, 0.9940, 0.9998].*100;
 Y5 = [0.2206, 0.5851, 0.9048].*100;
 Y6 = [0.3330, 0.8844, 0.9897].*100;
 Y7 = [0.5142, 0.9723, 0.9987].*100;
 Y8 = [0.7187, 0.9942, 0.9998].*100;

 figure('Name','Radio range')
 l=plot(x,Y1,'y-',x,Y2,'m-',x,Y3,'c-',x,Y4,'r-',x,Y5,'g-',x,Y6,'b-
',x,Y7,'-',x,Y8,'k-');
 for k=1:8
    l(k).LineWidth = 2;
 end
 title('Average availability (%)')
 xlabel('W (meters)')
 legend('N=20 S=3','N=40 S=3','N=60 S=3','N=80 S=3','N=20 S=6','N=40
S=6','N=60 S=6','N=80 S=6','location','southeast')
 axis([40 80 0 100])
 grid on
```

c) Quanto maior for o numero de nodes, maior será a disponibilidade, uma vez que existirão mais conexões entre estes e é mais provavel que um deles tenha conexão a um dos APs.

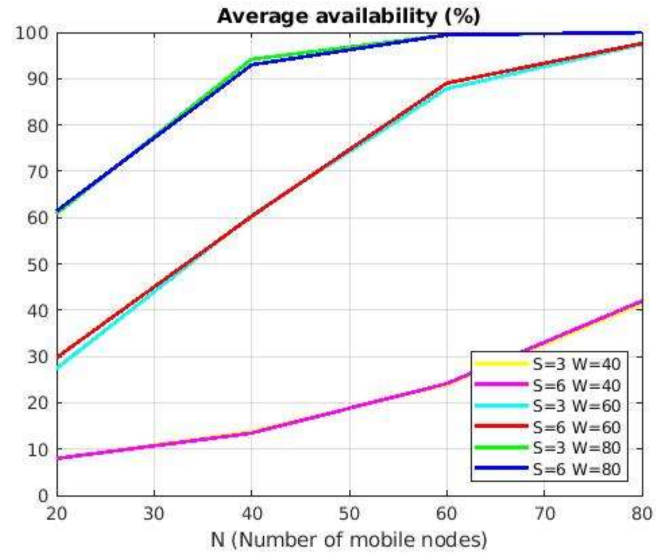

Figura 4 - Disponibilidade média para 1 AP.

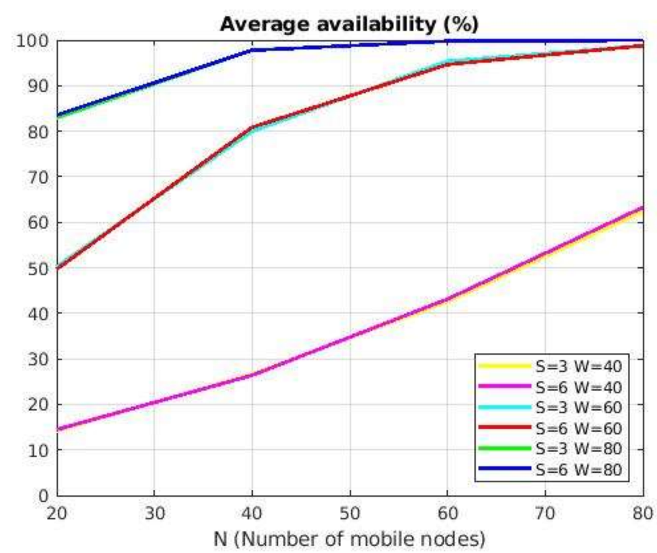

Figura 5 - Disponibilidade média para 2 APs.

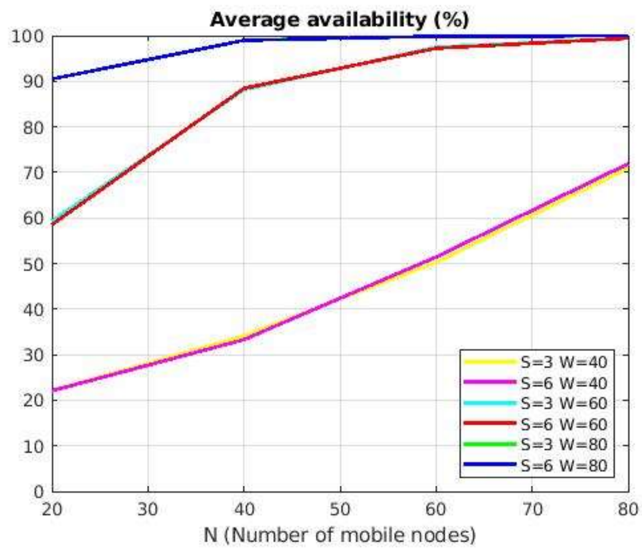

Figura 6- Disponibilidade média para 3 APs.

```
%1 AP

 x = [20, 40, 60, 80];

 Y1 = [0.0791, 0.1377, 0.2387, 0.4145].*100;
 Y2 = [0.0795, 0.1344, 0.2415, 0.4197].*100;
 Y3 = [0.2738, 0.6052, 0.8782, 0.9736].*100;
 Y4 = [0.2973, 0.6025, 0.8901, 0.9759].*100;
 Y5 = [0.6082, 0.9418, 0.9944, 0.9995].*100;
 Y6 = [0.6136, 0.9298, 0.9946, 0.9995].*100;

 figure('Name','Number of mobile nodes')
 l=plot(x,Y1,'y-',x,Y2,'m-',x,Y3,'c-',x,Y4,'r-',x,Y5,'g-',x,Y6,'b-');
 for k=1:6
    l(k).LineWidth = 2;
 end
 title('Average availability (%)')
 xlabel('N (Number of mobile nodes)')
 legend('S=3 W=40','S=6 W=40','S=3 W=60','S=6 W=60','S=3 W=80','S=6
W=80','location','southeast')
 axis([20 80 0 100])
 grid on
```

```
%2 AP

 x = [20, 40, 60, 80];

 Y1 = [0.1415, 0.2666, 0.4255, 0.6239].*100;
 Y2 = [0.1445, 0.2637, 0.4311, 0.6325].*100;
 Y3 = [0.5027, 0.7996, 0.9537, 0.9877].*100;
 Y4 = [0.4965, 0.8083, 0.9464, 0.9877].*100;
 Y5 = [0.8289, 0.9778, 0.9977, 0.9997].*100;
 Y6 = [0.8351, 0.9776, 0.9978, 0.9997].*100;

 figure('Name','Number of mobile nodes')
 l=plot(x,Y1,'y-',x,Y2,'m-',x,Y3,'c-',x,Y4,'r-',x,Y5,'g-',x,Y6,'b-');
 for k=1:6
    l(k).LineWidth = 2;
 end
 title('Average availability (%)')
 xlabel('N (Number of mobile nodes)')
 legend('S=3 W=40','S=6 W=40','S=3 W=60','S=6 W=60','S=3 W=80','S=6
W=80','location','southeast')
 axis([20 80 0 100])
 grid on

 %3 AP

  x = [20, 40, 60, 80];

 Y1 = [0.2215, 0.3408, 0.5027, 0.7094].*100;
 Y2 = [0.2206, 0.3330, 0.5142, 0.7187].*100;
 Y3 = [0.5932, 0.8811, 0.9748, 0.9940].*100;
 Y4 = [0.5851, 0.8844, 0.9723, 0.9942].*100;
 Y5 = [0.9048, 0.9900, 0.9984, 0.9998].*100;
 Y6 = [0.9048, 0.9897, 0.9987, 0.9998].*100;

 figure('Name','Number of mobile nodes')
 l=plot(x,Y1,'y-',x,Y2,'m-',x,Y3,'c-',x,Y4,'r-',x,Y5,'g-',x,Y6,'b-');
 for k=1:6
    l(k).LineWidth = 2;
 end
 title('Average availability (%)')
 xlabel('N (Number of mobile nodes)')
 legend('S=3 W=40','S=6 W=40','S=3 W=60','S=6 W=60','S=3 W=80','S=6
W=80','location','southeast')
 axis([20 80 0 100])
 grid on
```

d) Quantos mais APs, maior será a disponibilidade média uma vez que existirão mais pontos com ligação à internet, permitindo que nós que não conseguiam alcançar um AP agora possam alcançar outro. Porém, existe uma grande dependência da posição em que são colocados (Pouca melhoria caso estejam aglomerados; Muita melhoria caso estejam dispersos).

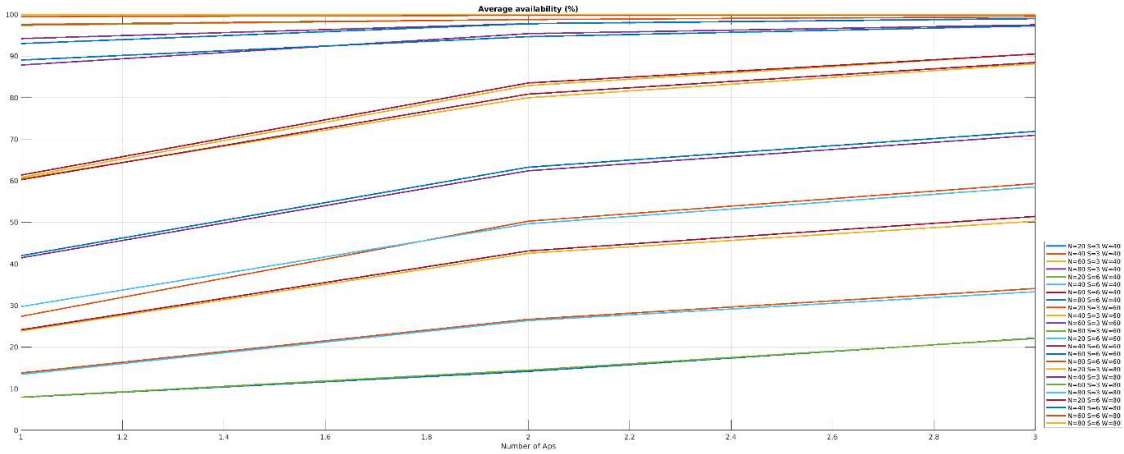

Figura 7- Disponibilidade média por número de APs.

```
x = [1, 2, 3];

 Y1 =  [0.0791, 0.1415, 0.2215].*100;
 Y2 =  [0.1377, 0.2666, 0.3408].*100;
 Y3 =  [0.2387, 0.4255, 0.5027].*100;
 Y4 =  [0.4145, 0.6239, 0.7094].*100;
 Y5 =  [0.0795, 0.1445, 0.2206].*100;
 Y6 =  [0.1344, 0.2637, 0.3330].*100;
 Y7 =  [0.2415, 0.4311, 0.5142].*100;
 Y8 =  [0.4197, 0.6325, 0.7187].*100;
 Y9 =  [0.2738, 0.5027, 0.5932].*100;
 Y10 = [0.6052, 0.7996, 0.8811].*100;
 Y11 = [0.8782, 0.9537, 0.9748].*100;
 Y12=  [0.9736, 0.9877, 0.9940].*100;
 Y13 = [0.2973, 0.4965, 0.5851].*100;
 Y14 = [0.6025, 0.8083, 0.8844].*100;
 Y15 = [0.8901, 0.9464, 0.9723].*100;
 Y16 = [0.9759, 0.9877, 0.9942].*100;
 Y17 = [0.6082, 0.8289, 0.9048].*100;
 Y18 = [0.9418, 0.9778, 0.9900].*100;
 Y19 = [0.9944, 0.9977, 0.9984].*100;
 Y20 = [0.9995, 0.9997, 0.9998].*100;
 Y21 = [0.6136, 0.8351, 0.9048].*100;
 Y22 = [0.9298, 0.9776, 0.9897].*100;
 Y23 = [0.9946, 0.9978, 0.9987].*100;
 Y24 = [0.9995, 0.9997, 0.9998].*100;
```

```
figure('Name','Radio range')

l=plot(x,Y1,x,Y2,x,Y3,x,Y4,x,Y5,x,Y6,x,Y7,x,Y8,x,Y9,x,Y10,x,Y11,x,Y12,x,Y
13,x,Y14,x,Y15,x,Y16,x,Y17,x,Y18,x,Y19,x,Y20,x,Y21,x,Y22,x,Y23,x,Y24);
 for k=1:24
    l(k).LineWidth = 2;
 end
 title('Average availability (%)')
 xlabel('Number of Aps')
 legend('N=20 S=3 W=40','N=40 S=3 W=40','N=60 S=3 W=40','N=80 S=3
W=40','N=20 S=6 W=40','N=40 S=6 W=40','N=60 S=6 W=40','N=80 S=6
W=40','N=20 S=3 W=60','N=40 S=3 W=60','N=60 S=3 W=60','N=80 S=3
W=60','N=20 S=6 W=60','N=40 S=6 W=60','N=60 S=6 W=60','N=80 S=6
W=60','N=20 S=3 W=80','N=40 S=3 W=80','N=60 S=3 W=80','N=80 S=3
W=80','N=20 S=6 W=80','N=40 S=6 W=80','N=60 S=6 W=80','N=80 S=6
W=80','location','southeast')
 axis([1 3 0 100])
 grid on
```

e) À medida que adicionamos mais nodes, usando um baixo range, verifica-se uma maior variação nos resultados (intrevalo de confiança aumenta). Por outro lado, se o range for mais elevado, esta tendencia inverte-se (intrevalo de confiança diminui).

f)

1 AP - Encontra-se numa das melhores localizações (centro).

Ex: (80,3,40)
[250,100] - A 41% M 19%
[260,100] - A 41% M 16%
[240,100] - A 39% M 12%
[250,150] - A 36% M 19%
[250,50] - A 40% M 20%

(20,3,80)
[250,100] - A 61% M 45%
[300,100] - A 60% M 40%
[200,100] - A 58% M 48%
[250,150] - A 57% M 41%
[250,50] - A 57% M 39%

Outras localizações possuem resultados inferiores ou que se encontram dentro da margem de erro de aproximadamente 2%.

2 AP - Deve ser feita uma translação do ponto onde existe mais uniformdade da distribuição das áreas sem cobertura para ranges baixos, para o que possui maior uniformidade para ranges elevados.

Ex: (20,3,80)
[150,100; 350,100] - A 83% M 67%
[125,100; 375,100] - A 88% M 77%

(20,3,40)
[150,100; 350,100] - A 14% M 3%
[125,100; 375,100] - A 14% M 2%

Ou seja, novas posições [125,100; 375,100].

3 AP - O AP central encontra-se bem localizado. Os outros 2 deviam ser deslocados um para a posiçãoo [80,100] e o outro para a posição [420,100] para permitir maximizar a area de cobertura dos AP. É melhor ter duas pequenas áreas no meio de 2 AP sem cobertura do que quatro metades das duas áreas, estando duas delas apenas na fronteira de 1 AP e as outras 2 no meio de 2 AP. Ou seja, é melhor [80,100] e [420,100] que [85,100] e [425,100].

Ex: (20,3,80)
[50,50; 250,100; 450,150] - A 90% M 82%
[80,100; 250,100; 420,100] - A 94% M 84%

(20,3,40)
[50,50; 250,100; 450,150] - A 22% M 7%
[80,100; 250,100; 420,100] - A 22% M 8%

Ou seja, novas posições [80,100; 250,100; 420,100].

g) O número mínimo encontrado de Aps foi 10, sendo eles os seguintes:
[50,50; 270,50; 375,50; 470,50; 235,150; 345,150; 450,150; 45,150;165,45;135,145].

Modelo usado para alcançar a um resultado base:

```
theta = linspace(0,2*pi);

x1 = 60*cos(theta) + 50;
y1 = 60*sin(theta) + 50;

x2 = 60*cos(theta) + 270;
y2 = 60*sin(theta) + 50;

x3 = 60*cos(theta) + 375;
y3 = 60*sin(theta) + 50;

x4 = 60*cos(theta) + 470;
y4 = 60*sin(theta) + 50;

x5 = 60*cos(theta) + 235;
y5 = 60*sin(theta) + 150;

x6 = 60*cos(theta) + 345;
y6 = 60*sin(theta) + 150;

x7 = 60*cos(theta) + 450;
y7 = 60*sin(theta) + 150;

x8 = 60*cos(theta) + 45;
y8 = 60*sin(theta) + 150;

x9 = 60*cos(theta) + 165;
y9 = 60*sin(theta) + 45;

x10 = 60*cos(theta) + 135;
y10 = 60*sin(theta) + 145;

plot (x1,y1,x2,y2,x3,y3,x4,y4,x5,y5,x6,y6,x7,y7,x8,y8,x9,y9,x10,y10)
axis([0 500 0 200])
```