



**Desempenho e Dimensionamento de Redes - 2018/2019**

**Engenharia de Computadores e Telemática**

## **3º Guião Prático - Relatório**

**AVAILABILITY PERFORMANCE OF MULTI-HOP WIRELESS  
NETWORKS WITH MOBILE TERMINALS**

**P4G2**

**Diego Hernandez nº 77013**

**Ricardo Pousa nº 80328**

# Simulador

## Descrição

O simulador representa uma área com uma grelha em 2D onde em ela estão representados nós de rede wireless móveis espalhados e movimentando-se nas linhas da grelha apresentada no simulador. Cada secção vertical encontra-se espaçada 50 unidades no eixo x do gráfico do simulador. De forma semelhante, cada secção horizontal encontra-se espaçada 50 unidades no eixo y do gráfico do simulador. Os nós são representados por pequenos círculos/pontos.

A rede wireless fornece internet aos nós móveis através de APs (*Access-Points*), representados no simulador como quadrados, que juntamente encontram-se espalhados na área do simulador intersectando também as linhas da grelha.

Tal como é mencionado no enunciado, o alcance rádio do AP e dos nós móveis é caracterizada por um dado alcance (*range*)  $W$  máximo para o estabelecimento de interligação entre nós móveis e/ou AP. Se a distância entre dois nós for superior a  $W$ , estas não estabelecem uma ligação. Se um dos nós interligados tiver acesso a internet, este lhe dá acesso a internet ao outro nó. Para que uma sequência de nós interligados obtenham conectividade à Internet, pelo menos um dos nós dentro de esse conjunto de nós interligados deve encontrar-se na área de alcance de um dos AP, isto é, o nó móvel tem de se encontrar a uma distância inferior a  $W$  do AP de modo a ter conectividade Internet.

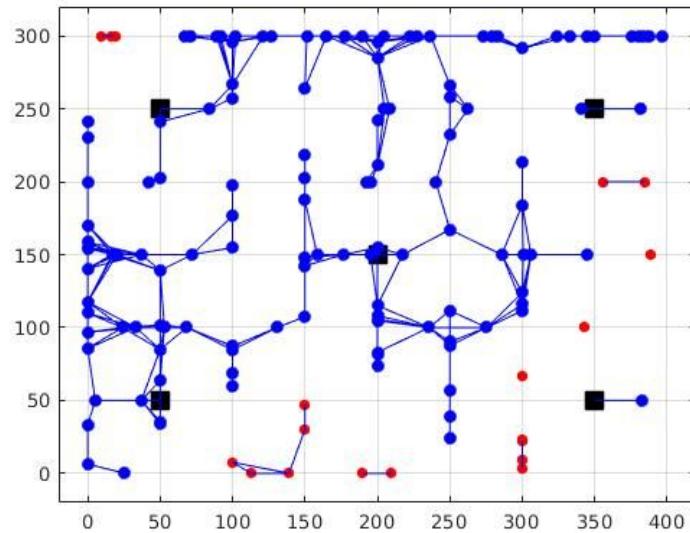
Na figura abaixo está um exemplo da execução do simulador, onde estão representados os nós móveis e os APs representados da forma em que foi explicado anteriormente. As interligações entre nós móveis e APs são representadas por retas azuis. Os nós móveis, durante a simulação, conseguem assumir as cores azul e vermelho em certas situações. Os nós que têm conectividade com a internet encontram-se representados pela cor azul, caso contrário são representados pela cor vermelha.

O simulador disponibilizado estabelece as posições dos nós de forma aleatória sempre que estes se encontram interceptados na grelha e dentro da área do simulador disponibilizada.

Para a avaliação do desempenho da disponibilidade de Internet pelos AP e nós móveis foi requerido a implementação de funções para o simulador do tipo:

- Contador do número de *time-slots* para cada nó móvel em que estes tiveram acesso a internet no simulador.
- Registo atual dos pares de nós que se encontram interligados.
- Registo dos identificadores dos nós móveis com conectividade à internet.
- Calculos da *Average Availability* e *Minimum Availability*.

Estas funcionalidades não foram fornecidas e tiveram de ser implementadas por nós para a monitorização da eficiência da disponibilidade de internet para redes wireless relativamente , tendo em conta o número de nós , Access-Point e alcances distintintos.



## UpdateCounter

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

A seguinte função encarrega-se da contagem, para cada nó móvel, do número de *time-slots* que estes tiveram acesso a internet. O parâmetro *C* representa um vetor com tamanho equivalente ao número de nós existentes na área geográfica do simulador. O índice do vetor representa o ID do nó móvel, enquanto o valor na posição pode ser equivalente a 0 ou 1. É 0 se o nó móvel no dado momento não se encontrar com conectividade à Internet e 1 caso contrário. É dado como parâmetro o *counter* de modo a atualizar os valores de modo a obter o pretendido. O retornado é o valor atualizado do número de *time-slots* que o nó móvel teve conectividade com a internet até o momento atual, obtendo-se pela soma do vetor *counter*, que tem os valores do número de *time-slots* que os nós tiveram acesso a internet até o momento anterior, com o vetor *C* que contém a informação (a '1') sobre os nós estão no momento com acesso a internet.

## ConnectedList

```
function L= ConnectedList(par,map)
% map.pos - a matrix with par.N rows and 2 columns; each row
identifies the (x,y)
%           coordinates of each mobile node
% L -       a matrix with 2 columns; each row identifies a pair of
nodes (mobile
%           nodes and AP nodes) with direct wireless links between
them
% This function computes matrix 'L' based on matrix map.pos.
assist = [map.pos; par.AP];
dist = pdist2(assist, assist);
Lbool = dist < par.W & dist ~= 0;
[I,J] = find(triu(Lbool,1));
L = [I,J];
end
```

Esta função encarrega-se do registo de ligações atuais entre pares de nós. Inicialmente é armazenado numa matriz de duas colunas, com nome *assist*, as coordenadas de todos os nós juntamente com as coordenadas dos APs, sendo obtidas a partir dos parâmetros de entrada *par* e *map*, nas colunas da matriz, sendo a primeira a coordenada *x* e a segunda coluna a coordenada *y*, sendo o número de linhas o número de objetos (nós e APs) apresentados no simulador. É usada a função matlab *pdist2*, de modo a determinar a distância euclidiana entre todos os pares de pontos apresentados no simulador (Nó-nó, Nó-AP e AP-AP). O resultado irá ser uma matriz com um número de linha e coluna equivalente ao número de objetos apresentados no simulador, cada elemento corresponde a distância do objeto com índice de linha *i* e o objeto com índice da coluna *j*. A seguinte matriz é armazenada na variável *dist*.

Como se pretende saber quais são os objetos que se interligam e sabendo que é possível a interligação entre nós desde que a distância entre estes não ultrapasse o alcance *W* mencionado, é feito um mapeamento de todas as distâncias que não são equivalentes a zero (que se for a zero, trata-se de uma distância dum nó com ele mesmo) e não são superiores ao alcance de *W* definido. Passando a matriz *dist* por filtragem, esta é armazenada na variável *Lbool*. Se os elementos estão de acordo com as condições, estas são postas na matriz com valor a ‘1’ na respetiva posição da matriz.

Uma vez que os valores da matriz que se encontram na parte triangular superior são refletidos na parte triangular inferior da matriz, isto é os valores repetem-se (por e.g, a distância entre o nó *i* e *j* é equivalente a distância entre o nó *j* e *i*), e uma vez que não se pretende ter duplicação de valores, fazemos mais outra filtragem: Extraímos apenas a parte triangular superior da matriz *Lbool* de modo a

evitar a duplicação de valores, sendo os valores da parte triangular inferior da mesma matriz posta a zeros.

Finalmente, com a matriz resultante podemos determinar que nós se interligam, sabendo o número da coluna e da linha dos elementos da matriz que se encontram a ‘1’. Para obter este tipo de informação é utilizada a função *find* de modo a obter as linhas (*I*) e as coluna (*J*) dos elementos a ‘1’ da matriz resultante da função *triu* mencionada.

## ConnectedNodes

```
function C= ConnectedNodes(par,L)
% C - an array with N values (one for each mobile node) that is 1 in
%      position i if mobile node i has Internet access
% This function computes array 'C' based on matrix 'L' with the node pairs
% that have direct wireless links.
%

% NOTE: To develop this function, check MATLAB function 'distances' that
%        computes shortest path distances of a graph.
x = par.N +par.nAP+1;
L_temp = L;
for n = 1:par.nAP
    L_temp = [ L_temp ; x (par.N+n)];
end
G = graph(L_temp(:,1),L_temp(:,2)); % plot(G);
C = distances(G,1:par.N,x).';
C = C < Inf;
end
```

A seguinte função é encarregada de determinar que nós móveis dentro da área geográfica apresentada no simulador apresentam acesso à internet.

Nesta função é necessário decorrer como auxílio a matriz *L* calculada na função *ConnectedList*, como foi mencionado na descrição da mesma função, esta contém todos os pares de nós móveis e/ou APs interligados no último instante registado. Esta matriz contém duas colunas, representando os índices dos pares de objetos interligados, que se encontram na mesma linha da matriz. É criada uma variável que contém a cópia da matriz *L*.

A variável *x* apresentada na função representa o índice da Internet. Os APs têm acesso a internet. O índice da internet irá ter um índice superior aos nós móveis. Logo, o índice dos APs e da Internet será equivalente a:

$$ID\ Internet = Número\ de\ nós\ móveis + Número\ de\ Access-Points + 1$$

Na variável *L\_temp* é necessário mencionar que os APs estão interligados com a Internet, portanto é inserido os Ids dos AP junto com o Id da Internet nas correspondentes linhas, para retificar o mencionado.

Utilizamos a função matlab *graph* para determinar os objetos grafos existentes, isto é, determinar os conjuntos de nós que se encontram interligados.

Estando os grafos determinados, recorremos a uma outra função matlab, *distances*, de modo a verificar o número de *next-hops* mínimo (distância shortest-path de um grafo) necessários para um dado objeto (nó móvel) atingir a Internet, com índice  $x$ , consequentemente a um dos APs apresentados na área geográfica. Se não é possível de a partir um dado nó percorrer o caminho do correspondente grafo até a Internet, a distância resultante é equivalente a infinito (*Inf*). O número de next-hops é armazenado na linha correspondente ao índice do nó. Os valores calculados são armazenados no vetor coluna C.

Finalmente, para determinar quais dos nós se encontra ligado a Internet são selecionados a 1 os elementos do vetor C cujo valor é inferior ao infinito, *Inf*, isto é, os identificadores dos nós que têm acesso a internet, pois no grafo a que eles pertencem existe pelo menos um *Access-Point* que fornece acesso a Internet.

## AverageAvailability, MinimumAvailability, Results

```
function [AverageAvailability, MinimumAvailability]=  
results(T,counter)  
% This function computes the average and the minimum availability  
(values  
% between 0 and 1) based on array 'counter' and on the total number  
of  
% time slots T.  
AverageAvailability = mean(counter)/T;  
MinimumAvailability = min(counter)/T;  
end
```

Esta função é responsável por calcular a *AverageAvailability* e *Minimum Availability* do acesso a internet dos nós móveis que se encontram na área geográfica ao longo da simulação.

O cálculo de *AverageAvailability* consiste no cálculo da média do *counter*, mencionado na explicação da função *UpdateCounter*, e a sua divisão com o número de intervalos de tempo (*time-slots*) usados na simulação.

O cálculo de *MinimumAvailability* consiste no cálculo do valor mínimo apresentado no *counter*, isto é o menor número de acesso a internet que um nó teve durante a simulação, e a sua divisão com o número de intervalos de tempo usados na simulação.

## Intervalo de Confiança

Em estatística, intervalo de confiança é um tipo de estimativa por intervalo de um parâmetro populacional desconhecido. Para o seu cálculo é utilizada a seguinte expressão:

$$\bar{X}(n) \pm z_{1-\alpha/2} \sqrt{s^2(n)/n}$$

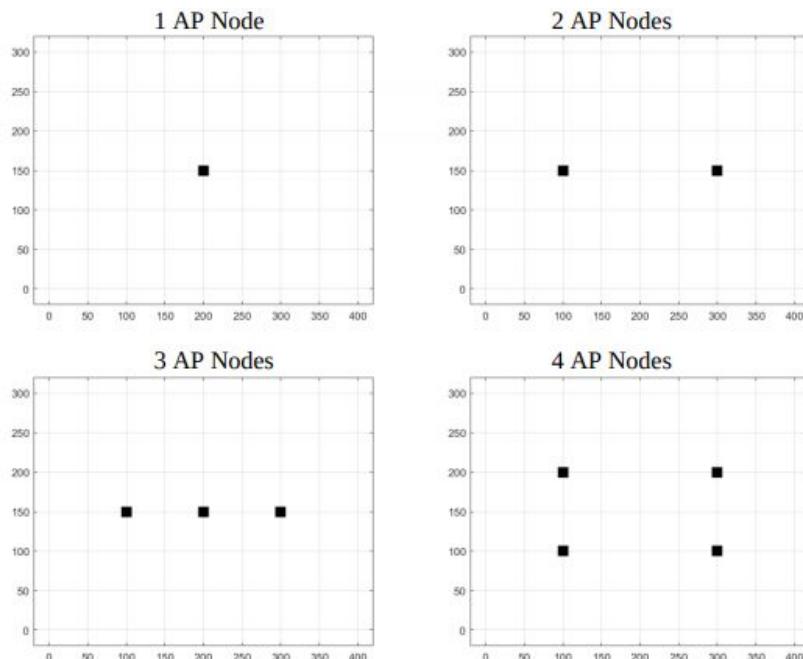
Nó código do programa simulador, o cálculo do intervalo de confiança para valores de *Minimum Availability* e também para *Average Availability* é conseguido da seguinte forma:

```
AverageMatrix(o-1,1) = mean(AverageAvailability);
AverageMatrix(o-1,2) = norminv(1-alfa/2)*sqrt(var(AverageAvailability)/N);
MinimumMatrix(o-1,1) = mean(MinimumAvailability);
MinimumMatrix(o-1,2) = norminv(1-alfa/2)*sqrt(var(MinimumAvailability)/N);
```

## Assignment Tasks

A & B)

Para os distintos casos de configurações de AP apresentadas no código do programa do simulador, foi pedido a execução de 20 vezes do mesmo simulador, para cada caso, de modo a determinar o *Minimum Availability* e *Average Availability* do acesso a Internet dos nós móveis, com um intervalo de confiança equivalente a 90%. Para cada execução foi necessário considerar o tempo delta equivalente a 1 ( $\Delta = 1$ ) e o tempo necessário para a simulação equivalente a 3600 ( $T = 3600$ ).



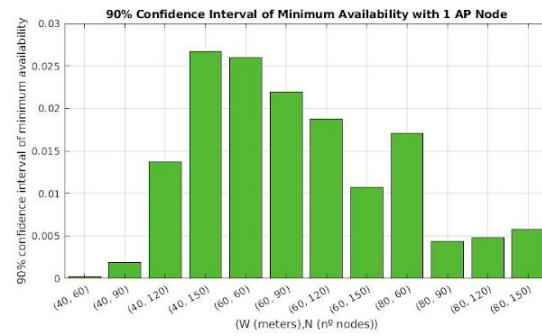
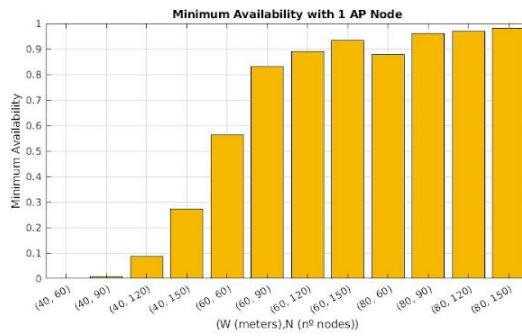
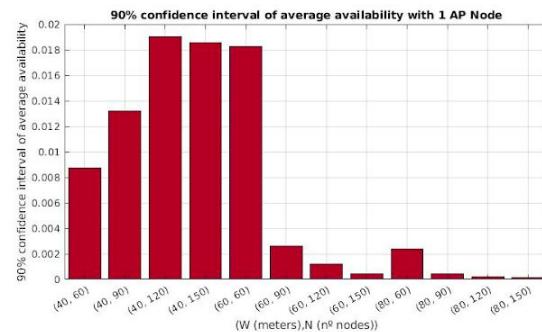
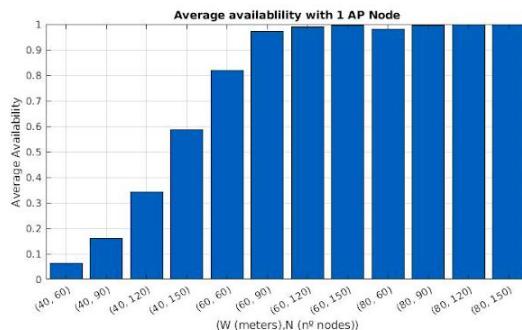
# 1 AP Node

Coordenadas:

AP	x	y
1	200	175

Resultados:

Case	W(meters)	N	Average Availability	90% confidence interval	Minimum Availability	90% confidence interval
A	40	60	0.0634863426	0.0087187276	0.000125	0.0001829868
B	40	90	0.1612868827	0.0131797016	0.0071805556	0.0018824842
C	40	120	0.3431531250	0.0190583678	0.086666666667	0.0137602758
D	40	150	0.5867371296	0.0185491734	0.2728472222	0.0266965701
E	60	60	0.8204481481	0.0182590327	0.5630972222	0.0259741342
F	60	90	0.9712538580	0.0026022281	0.8305694444	0.0219174115
G	60	120	0.9894645833	0.0012002135	0.8904027778	0.0187485470
H	60	150	0.9960137963	0.0004557870	0.9347916667	0.0107485596
I	80	60	0.9798062500	0.0023633089	0.8769861111	0.0171102698
J	80	90	0.9966251543	0.0004196700	0.9590694444	0.0043592491
K	80	120	0.9989561343	0.0001840538	0.9678888889	0.0048227192
L	80	150	0.9995602778	0.0001394780	0.9798611111	0.0057428567



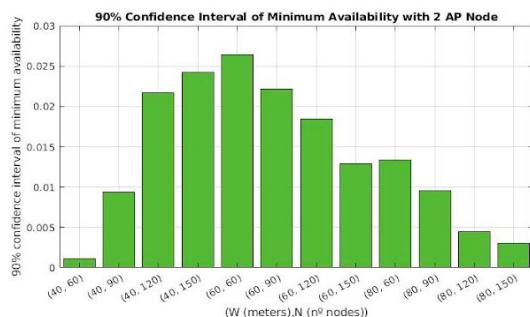
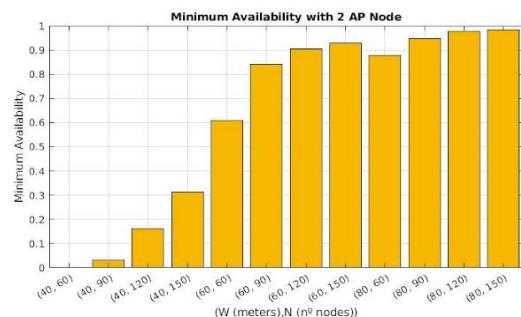
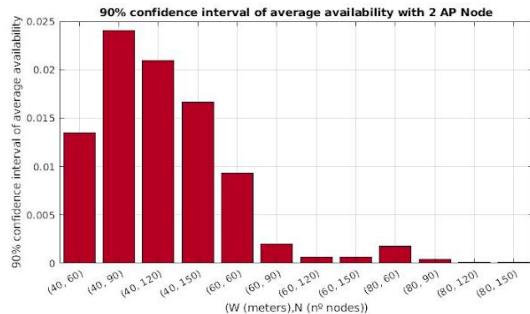
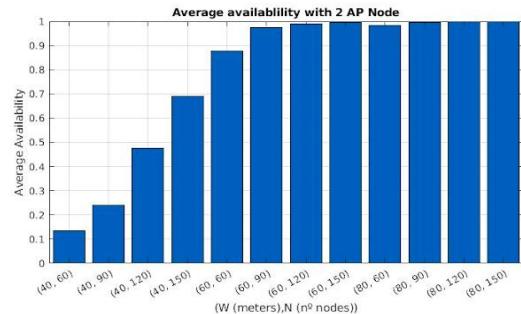
## 2 AP Node

Coordenadas:

AP	x	y
1	100	150
2	300	150

Resultados:

Case	W(meters)	N	Average Availability	90% confidence interval	Minimum Availability	90% confidence interval
A	40	60	0.13554468	0.01351088	0.00191667	0.00109532
B	40	90	0.24015833	0.02402423	0.03288889	0.00937356
C	40	120	0.47612431	0.02096396	0.16137500	0.02173321
D	40	150	0.68983213	0.01666004	0.31150000	0.02421992
E	60	60	0.87654282	0.00929382	0.60883333	0.02640092
F	60	90	0.97354228	0.00199721	0.84231944	0.02218250
G	60	120	0.99057882	0.00067105	0.90481944	0.01844766
H	60	150	0.99585417	0.00065555	0.92954167	0.01295165
I	80	60	0.98344398	0.00180261	0.87633333	0.01333455
J	80	90	0.99694012	0.00043282	0.94605556	0.00959959
K	80	120	0.99918368	0.00012230	0.97615278	0.00450319
L	80	150	0.99963463	0.00007710	0.98477778	0.00302161



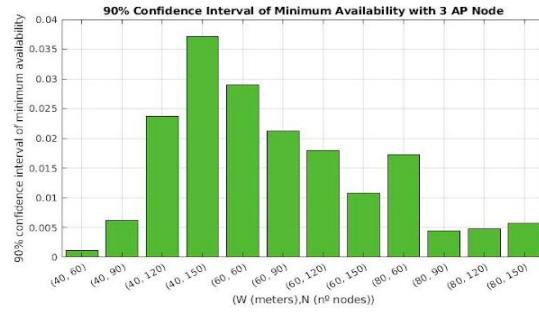
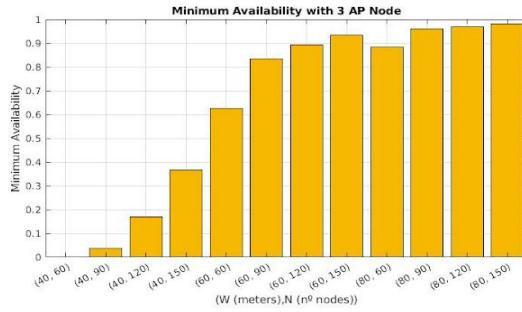
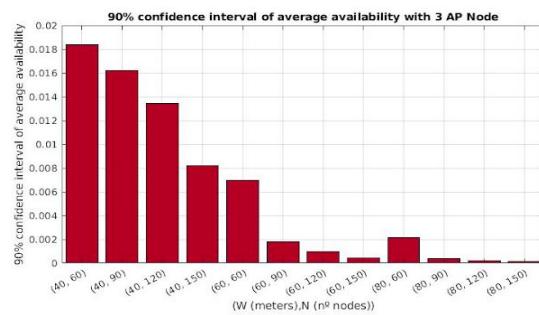
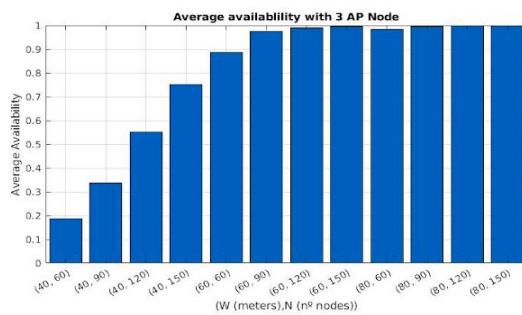
### 3 AP Node

Coordenadas:

AP	x	y
1	100	150
2	200	150
3	300	150

Resultados:

Case	W(meters)	N	Average Availability	90% confidence interval	Minimum Availability	90% confidence interval
A	40	60	0.18572917	0.01839122	0.00291667	0.00111105
B	40	90	0.33574907	0.01622186	0.03659722	0.00614888
C	40	120	0.55107836	0.01346710	0.16823611	0.02371966
D	40	150	0.75006741	0.00823220	0.36695833	0.03709266
E	60	60	0.887778009	0.00697887	0.62526389	0.02904070
F	60	90	0.97489568	0.00178921	0.83519444	0.02126692
G	60	120	0.99025475	0.00098743	0.89268056	0.01795645
H	60	150	0.99608630	0.00043129	0.93490278	0.01078562
I	80	60	0.98293333	0.00214211	0.88319444	0.01718518
J	80	90	0.99692269	0.00035752	0.95995833	0.00444461
K	80	120	0.99896377	0.00018370	0.96788889	0.00482272
L	80	150	0.99956454	0.00013670	0.97986111	0.00574286



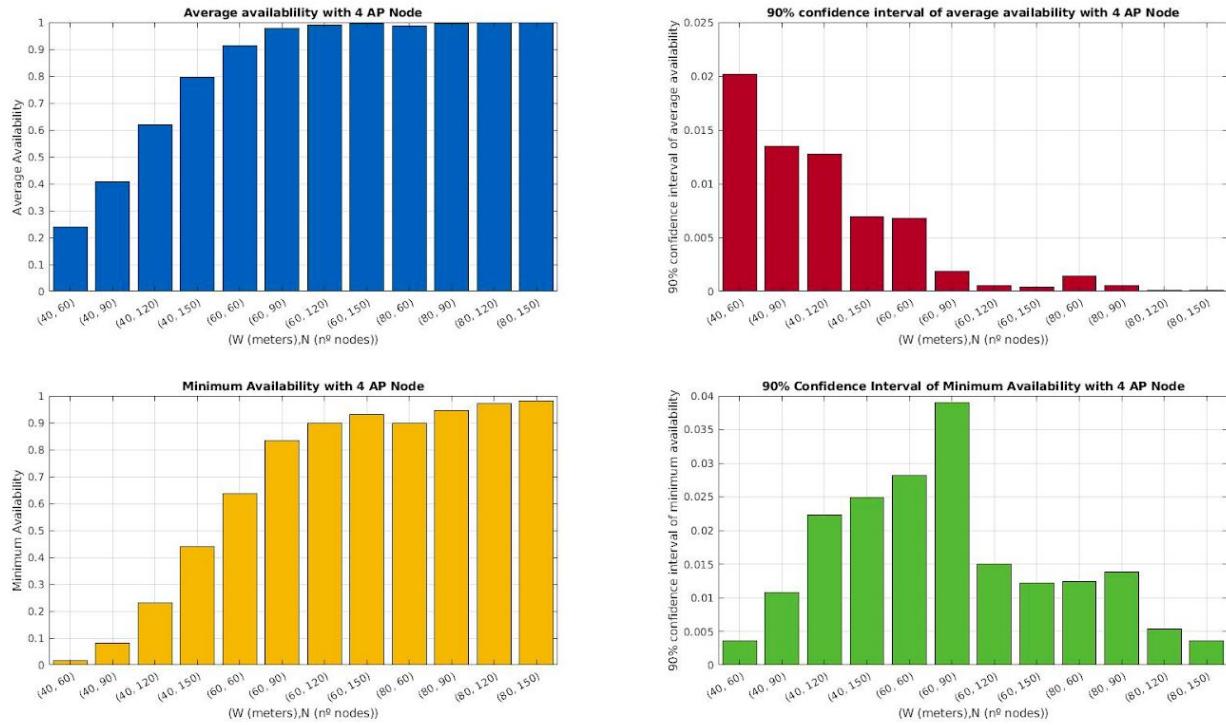
## 4 AP Node

Coordenadas:

AP	x	y
1	100	100
2	300	100
3	100	200
4	300	200

Resultados:

Case	W(meters)	N	Average Availability	90% confidence interval	Minimum Availability	90% confidence interval
A	40	60	0.23939	0.020148	0.015278	0.003614
B	40	90	0.40757	0.013523	0.081319	0.01071
C	40	120	0.62018	0.012723	0.23135	0.022301
D	40	150	0.79639	0.0069607	0.44115	0.024899
E	60	60	0.91179	0.0068211	0.6364	0.028168
F	60	90	0.97933	0.0018716	0.83329	0.039022
G	60	120	0.991	0.00055822	0.89872	0.015043
H	60	150	0.99592	0.00042651	0.93176	0.012212
I	80	60	0.98738	0.0014072	0.89911	0.012442
J	80	90	0.9965	0.0005458	0.94528	0.013833
K	80	120	0.99901	0.0001424	0.9719	0.0053007
L	80	150	0.99956	8.5301e-05	0.9816	0.0036279



C)

De uma forma geral, para os 4 casos apresentados acima, verificamos que a percentagem média de nós com Internet, como também a percentagem de um número mínimo de nós com internet na simulação vai aumentando com o aumento do alcance rádio,  $W$ , dos nós móveis e dos Access Points, como também aumenta com o aumento de número de nós móveis e Access-Points.

De forma geral, a percentagem de nós com internet vai aumentar quando o número de APs aumenta, uma vez que a área de cobertura gerada pelos APs para a área geográfica do simulador aumenta, havendo maiores probabilidades de um dado nó ter em um dado momento acesso a internet devido a encontrar-se ao alcance do AP ou um nó a que se encontra interligado se encontra também interligado com o Access-Point.

Para um mesmo range, a percentagem de nós com Internet aumenta com o aumento de nós móveis na área geográfica, uma vez que outros nós que não se encontram dentro da área de cobertura gerada pelos APs, tem maiores probabilidades de encontrar um nó com acesso a internet a uma distância inferior ao alcance ( $W$ ) para a interligação entre nós.

Quando o número de nós móveis e/ou o alcance dos nós e APs é muito elevado, maior parte dos nós da área geográfica tem uma cobertura de Internet, não sendo o número de APs um fator muito significativo, uma vez que vários nós móveis que têm acesso a internet permitem a outros nós móveis dentro da sua área de interligação dar acesso a internet, quando estes não se encontram a uma distância inferior a  $W$  dos APs da área geográfica apresentada no simulador.

D)

Com a informação obtida pelas várias simulações para os mais variados casos verifica-se que tanto para um aumento de número de nós, de APs e/ou de alcance para a interligação entre nós e Access-Points na rede, menor irá ser o intervalo de confiança.

Por um lado quando o número de nós móveis é elevado, o número de amostras também o é, que dá como consequência o intervalo de confiança ser pequeno devido as variâncias de valores apresentada ser reduzida, em comparação com casos em que o número de amostras é menor. Por outro lado quando a área de cobertura e de ligação entre nós móveis é elevada, maior parte dos nós irão encontrar-se com acesso a Internet durante a simulação, levando a poucas variâncias no número de nós interligados a Internet, visto, na simulação, que maior parte desses encontram-se ligados durante toda simulação.

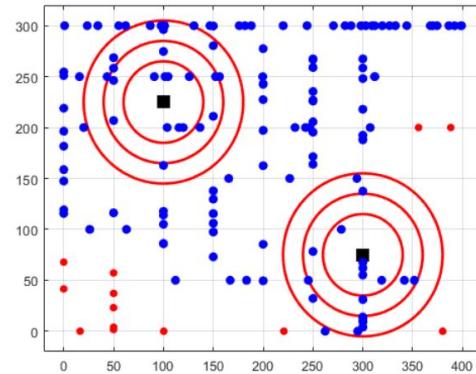
E)

Através de várias experiências, com a alteração dos posicionamento dos *Access-Points*, na área geográfica do simulador, foi verificado que é possível obter melhores resultados no desempenho de disponibilidade de Internet para os nós que se encontram na dada área geográfica. Para os diferentes casos de números de AP, e realizando o mesmo número de simulações que nas experiências efetuadas na alínea a), as seguintes configurações e resultados são as seguintes:

## 2 AP Nodes

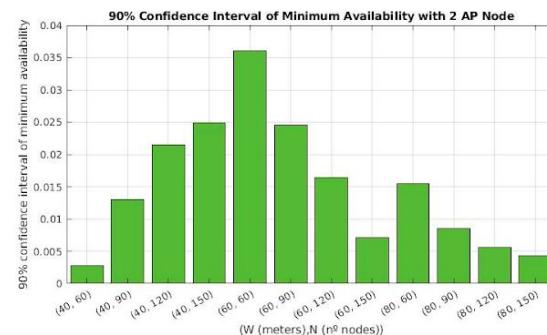
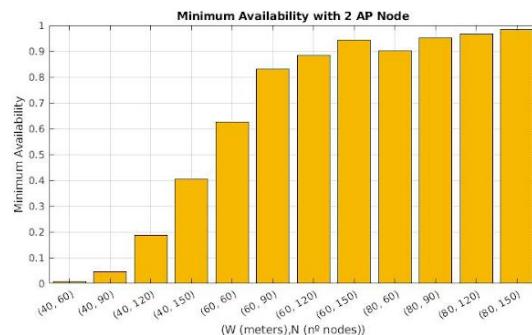
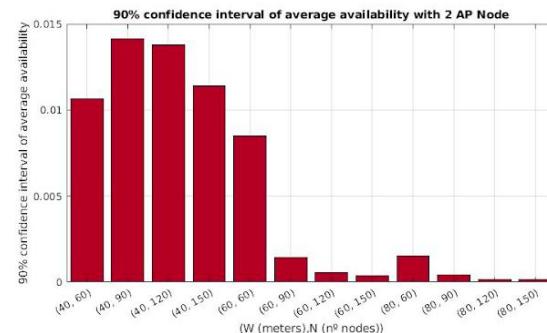
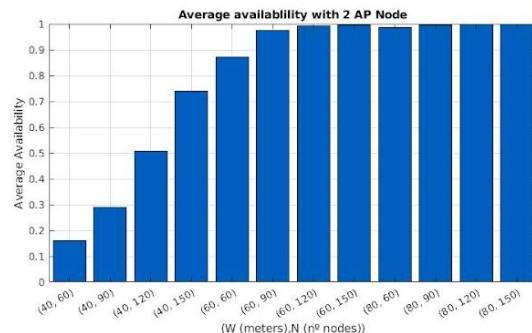
Coordenadas:

AP	x	y
1	100	225
2	300	75



Resultados:

Case	W(meters)	N	Average Availability	90% confidence interval	Minimum Availability	90% confidence interval
A	40	60	0.16003056	0.01067509	0.00708333	0.00277023
B	40	90	0.29089537	0.01412078	0.04656944	0.01298575
C	40	120	0.50633715	0.01378096	0.18741667	0.02152655
D	40	150	0.73975148	0.01141800	0.40479167	0.02485058
E	60	60	0.87226713	0.00850248	0.62470833	0.03607159
F	60	90	0.97520725	0.00142352	0.83134722	0.02451331
G	60	120	0.99122269	0.00056117	0.88501389	0.01638755
H	60	150	0.99606833	0.00038553	0.94395833	0.00710829
I	80	60	0.98631528	0.00150934	0.90200000	0.01542990
J	80	90	0.99664769	0.00039742	0.95077778	0.00852411
K	80	120	0.99910521	0.00013446	0.96681944	0.00564132
L	80	150	0.99954926	0.00015104	0.98286111	0.00423665



### Observações:

A seguinte configuração para dois APs foi o posicionamento de um AP próximo do extremo esquerdo superior e outro próximo do extremo direito inferior. Tendo em comparação os resultados obtidos

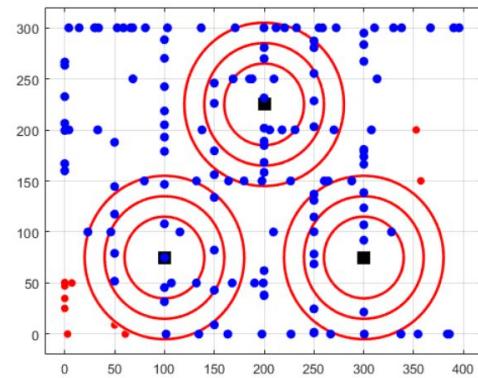
para este caso com os resultados da configuração de dois APs realizados nas simulações para a tarefa da alínea a) verifica-se que tanto o *Average Availability* como também o *Minimum Availability* aumenta para os casos A,B,C e D; ligeiros aumentos para *Average Availability* para os casos F, G ,H, I e valores de *Average Availability* não superiores, mas muito aproximados para os restantes casos.

Uma vez que para este tipo de configuração favorece maior parte dos casos registados, consideramos esta como a ideal para o aumento de percentagem de nós móveis com conectividade à Internet durante o decorrer do tempo de simulação.

### 3 AP Nodes

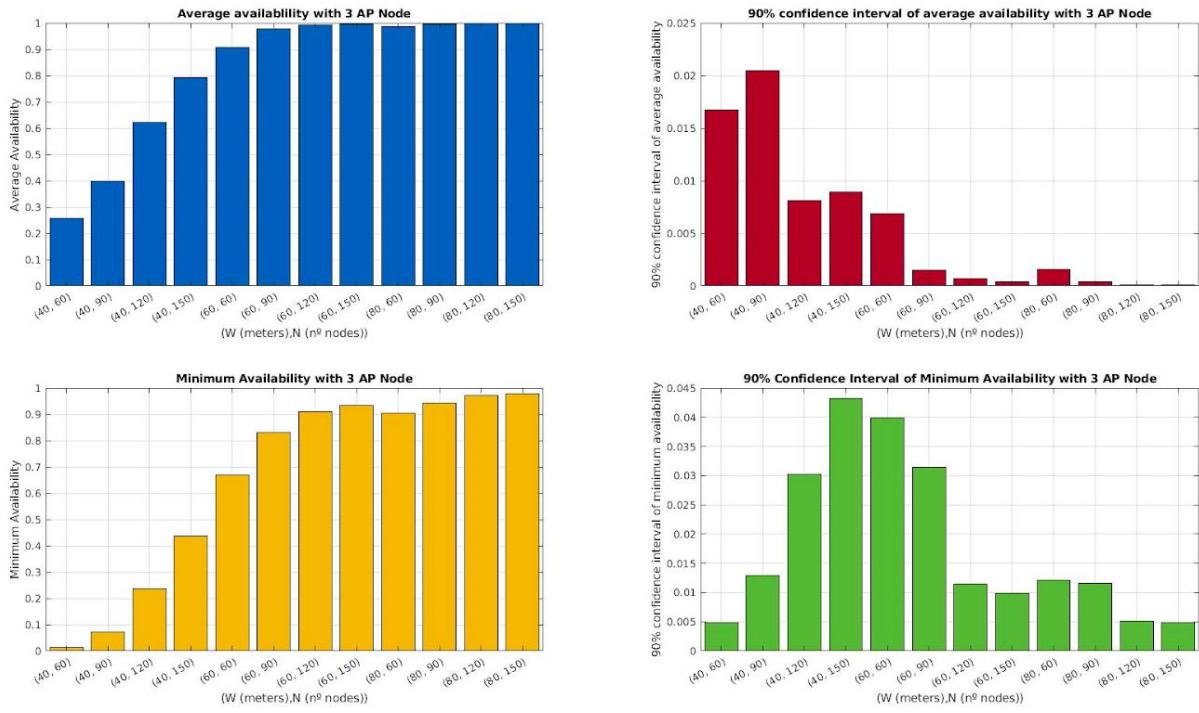
Coordenadas:

AP	x	y
1	100	75
2	200	225
3	300	75



Resultados:

Case	W(meters)	N	Average Availability	90% confidence interval	Minimum Availability	90% confidence interval
A	40	60	0.25591343	0.01676253	0.01227778	0.00481088
B	40	90	0.39821836	0.02045574	0.07211111	0.01291887
C	40	120	0.62144028	0.00811818	0.23570833	0.03021423
D	40	150	0.79311093	0.00896500	0.43719444	0.04322195
E	60	60	0.90860162	0.00688814	0.67034722	0.03990360
F	60	90	0.97708241	0.00150403	0.82966667	0.03143757
G	60	120	0.99139988	0.00069026	0.91031944	0.01143924
H	60	150	0.99566852	0.00037611	0.93247222	0.00982726
I	80	60	0.98597014	0.00155502	0.90520833	0.01210207
J	80	90	0.99672377	0.00038935	0.94197222	0.01161309
K	80	120	0.99908507	0.00013605	0.97070833	0.00515890
L	80	150	0.99961602	0.00007977	0.97816667	0.00480457



### Observações:

Com a disponibilidade de três Access Points, foram conseguidos melhores resultados com o posicionamento dos APs de forma a estes formarem um triângulo. De notar que foi verificado que com este tipo de configuração, os *Access-Points* possibilitam a cobertura de Internet para os nós que se encontram próximos dos dois extremos inferiores, sendo estes, verificado por várias simulações, os mais afetados no que toca a acesso a internet quando os APs se encontram afastados pelo centro da área geográfica.

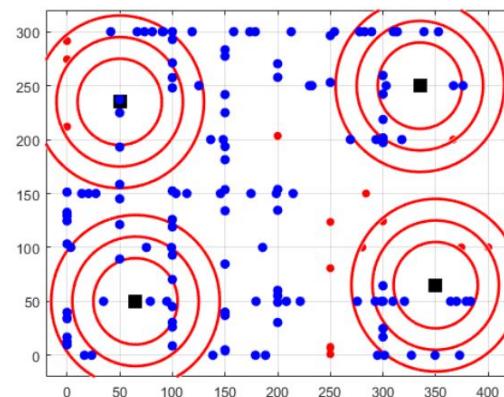
Com este tipo de configuração e comparando os resultados da configuração de 3 *Access-Points* da alínea a), consideramos a obtenção de aumentos significativos tanto no *Average Availability* como no *Minimum Availability* em casos em que o alcance rádio dos nós móveis é pequena, especificamente para os casos em que o *range*,  $W$ , é equivalente a 40 e 60 metros, sendo estes os casos: A,B,C,D,E,F e G. Relativamente aos restantes outros casos, comparando os resultados da alínea a) para a configuração de 3 AP, os valores de *Average Availability* e *Minimum Availability* são bastante próximos, sendo alguns minimamente superiores e outros inferiores, não sendo significante a diferença.

Uma vez que para este tipo de configuração favorece maior parte dos casos registados, consideramos esta como a ideal para o aumento de percentagem de nós móveis com conectividade à Internet durante o decorrer do tempo de simulação.

## 4 AP Nodes

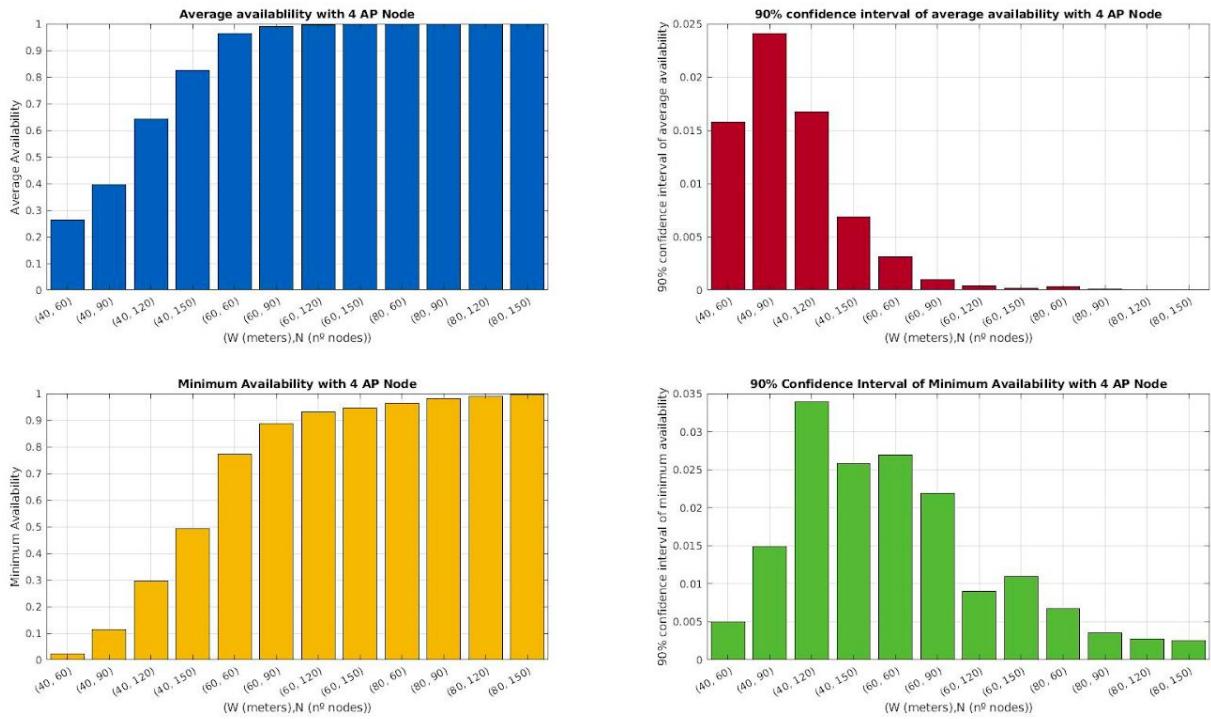
Coordenadas:

AP	x	y
1	50	235
2	335	250
3	65	50
4	350	65



Resultados:

Case	W(meters)	N	Average Availability	90% confidence interval	Minimum Availability	90% confidence interval
A	40	60	0.26185139	0.01578413	0.02087500	0.00497113
B	40	90	0.39625880	0.02406076	0.11336111	0.01484143
C	40	120	0.64374248	0.01670032	0.29472222	0.03394102
D	40	150	0.82548093	0.00684652	0.49193056	0.02582266
E	60	60	0.96201065	0.00312534	0.77205556	0.02687307
F	60	90	0.99021358	0.00097263	0.88631944	0.02183352
G	60	120	0.99604525	0.00041312	0.93002778	0.00897740
H	60	150	0.99803574	0.00020919	0.94581944	0.01091782
I	80	60	0.99815671	0.00031859	0.96440278	0.00670627
J	80	90	0.99955340	0.00008202	0.98118056	0.00353976
K	80	120	0.99984282	0.00004183	0.99061111	0.00271476
L	80	150	0.99993806	0.00004286	0.99547222	0.00249115



### Observações:

No caso de 4 *Access-Points* o posicionamento destes assemelha-se ao posicionamento dos 4 Access-Points da experiência com 4 Access-Points da tarefa a), no entanto, para esta nova configuração estes encontram-se ligeiramente mais próximos dos extremos da área geográfica do simulador, visto que, durante as experiências foi observado que os nós móveis que se encontram a movimentar-se nas zonas das extremidades da área geográfica são aqueles que normalmente possuem menor número de *time-slots* com conectividade com a Internet durante a simulação em casos em que os APs se encontram mais centralizados da área e afastados dos extremos.

Comparando o resultados de *Average Availability* e *Minimum Availability* com os posicionamentos dos *Access-Points* da experiência da alínea a), verificamos que para este novo posicionamento existem melhorias para a grande maioria dos casos apresentados, sendo o caso B o único caso que apresenta um valor ligeiramente menor ao valor de *Average Availability* do caso B da experiência realizada na alínea a) para 4 *Access-Points*.

Uma vez que para este tipo de configuração favorece maior parte dos casos registados, consideramos esta como a ideal para o aumento de percentagem de nós móveis com conectividade à Internet durante o decorrer do tempo de simulação.

f)

Para um número de nós equivalente a 150, e para cada alcance de rádio, W, equivalente a 40, 60 e 80 metros, foi determinado, através de várias execuções com o simulador, o número mínimo de *Access-Points* necessários e a sua localização de modo a que o valor mínimo de disponibilidade de acesso a Internet não seja inferior a 99%.

i.

$W = 40$  metros;

Número mínimo de *Access-Points*: 40

Coordenadas:

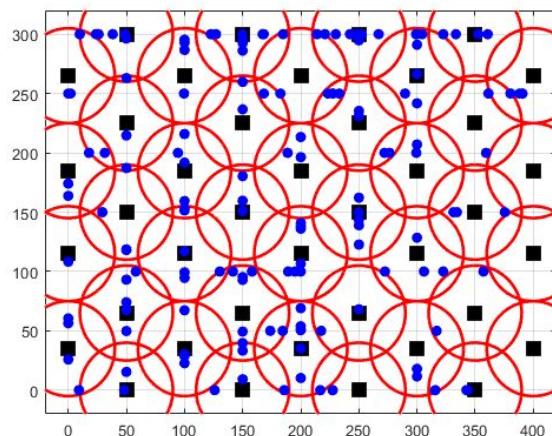
AP	x	y									
1	50	300	11	150	300	21	250	300	31	350	300
2	0	265	12	100	265	22	200	265	32	300	265
3	50	225	13	150	225	23	250	225	33	350	225
4	0	185	14	100	185	24	200	185	34	300	185
5	50	150	15	150	150	25	250	150	35	350	150
6	0	115	16	100	115	26	200	115	36	300	115
7	50	65	17	150	65	27	250	65	37	350	65
8	0	35	18	100	35	28	200	35	38	300	35
9	50	0	19	150	0	29	250	0	39	350	0
10	400	265	20	400	185	30	400	115	40	400	35

Average: 0.99993037

90ConfAverage: 0.00002142

Minimum: 0.99573611

90ConfMinimum: 0.00142852



ii.

$W = 60$  metros;

Número mínimo de *Access-Points*: 4

Coordenadas:

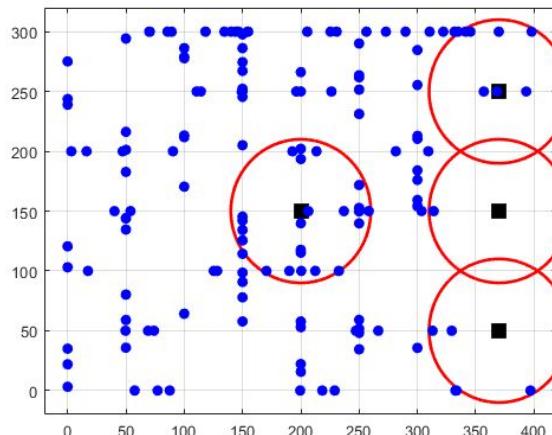
AP	x	y
1	175	150
2	370	250
3	370	150
4	370	50

Average : 0.99977352

90ConfAverage : 0.00012457

Minimum:0.99286111

90ConfMinimum: 0.0003570



### Observação:

Verificou-se durante às experiências, realizadas com o simulador nesta tarefa, que existe menor concentração de nós móveis à direita. Os nós que se encontram mais afastados da zona de grande concentração de nós móveis são os mais afetados no que toca ao acesso a Internet, uma vez que se torna difícil a possibilidade de estes obterem conectividade a partir da interligação com um outro nó, quando não se encontram dentro de um alcance de um *Access-Point*. Portanto, para obter os resultados desejados, utilizando o menor número de *Access-Points* possíveis, é necessário a consideração desta situação e possibilitar acesso a Internet a maioria destes nós através da ligação direta destes com os Access-Points. Logo, foram colocados Access-Points de modo a cobrir a área da direita. É colocado um AP central de modo a dar conectividade internet aos nós que se encontram no seu alcance, e de seguida estes nós móveis conseguirem fornecer conectividade à internet aos que não se encontram no alcance do AP em consideração. É possível fornecer conectividade de todos estes nós devido uma vez que se encontram em uma zona bastante concentrada de nós móveis, existindo facilidade de fornecimento de internet, não só diretamente a partir do AP, mas sim através da interligação entre nós, não sendo requerido muitos APs na zona concentrada de nós móveis.

Relativamente à situação apresentada no simulador, consideramos este cenário mais realista. Podemos por exemplo considerar que : A zona mais concentrada de nós poderá simular uma cidade apresentando grande concentração de pessoas, e as zonas mais afastadas da cidade ser normalmente aquelas cuja população é reduzida.

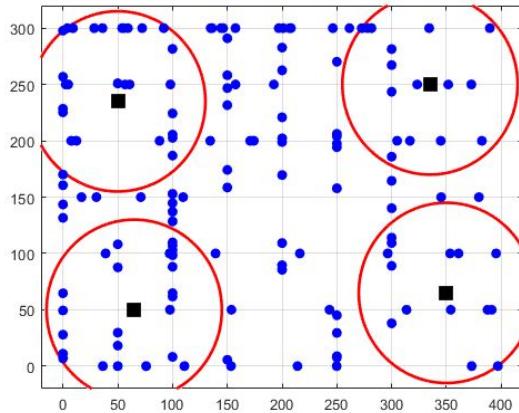
iii.

APs com alcance de 80 .

Número mínimo de *Access-Points*: 4

Coordenadas:

AP	x	y
1	50	235
2	65	50
3	335	250
4	350	65



Average: 0.99982670

90ConfAverage: 0.0000452

Minimum: 0.99923540

90ConfMinimum: 0.0023570

## Código

```
close all;
clear all;
format shortG
```

```
NODES = [60,90,120,150];
RANGE = [40,60,80];
CASES = ['A','B','C','D','E','F','G','H','T','J','K','L'];
AverageMatrix = zeros(12,2);
MinimumMatrix = zeros(12,2);
%Parameters initialization:

par.delta= 1; % Time slot (in seconds)
%par.AP = [200 150]; % Coordinates of each AP
%par.AP = [100 150;300 150]; % Coordinates of each AP
%par.AP = [100 150;200 150;300 150]; % Coordinates of each AP
%par.AP = [100 100;300 100;100 200;300 200]; % Coordinates of each AP1
%f)
% par.AP = [50 235;335 250;... 80 range 99 minimum
%       65 50;350 65];
% RANGE = [80,80,80];
% par.AP = [175 150;... 60 range rework 99 minimum
%       370 250;370 150; 370 50];
% RANGE = [60,60,60];
```

```

par.AP = [ 50 300; 150 300; 250 300; 350 300;... 40 range 99 minimum
          00 265;100 265;200 265;300 265;400 265;...
          50 225;150 225;250 225;350 225;...
          00 185; 100 185; 200 185; 300 185;400 185;...
          50 150;150 150;250 150;350 150; ...
          00 115;100 115 ;200 115;300 115;400 115;
          50 65;150 65;250 65;350 65; ...
          00 35;100 35; 200 35; 300 35; 400 35;...
          50 00;150 00;250 00;350 00;
];
RANGE = [40,40,40];
size(par.AP)

```

```

%own coordinates e)
%par.AP = [100 225;300 75]; % Best of 2
%par.AP = [100 75;200 225;300 75]; %Best for 3 APs
%par.AP = [75 50;325 50;75 250;325 250]; % Best for 4 APs

```

```
par.nAP = size(par.AP,1); %Number of APs
```

```
T= 3600; % No. of time slots of the simulation
```

```

plotar = 0; % if plotar = 0, there is no visualization
            % if plotar = 1, node movement is visualized
            % if plotar = 2, node movement and connectivity are visualized
%h= waitbar(0,'Running simulation...');
alfa = 0.1; % 90%
N = 20;
tic;
% Simulation cycles:

```

```

%full sim
% for o = 2:13
%   par.N= NODES(rem((o+2),4)+1); %Number of nodes
%   par.W= RANGE(int16(o/4));
%for f)
for o = 2:4
    par.N = 150;
    par.W= RANGE(o-1); % Radio range (in meters)

```

```

parfor n = 1:N
    map= InitializeState(par);% Initialization of mobile node positions and auxiliary matrices:
    counter= InitializeCounter(par,N); % Initialization of statistical counters
    L= [];
    C= [];
    for iter= 1:T
        %waitbar(((n-1)*T+iter)/(N*T),h,sprintf('Running sim: %10.d of %8.d',(n-1)*T+iter, T*N ));
        % Update of mobile node positions and auxiliary matrices:
        map= UpdateState(par,map);

        % Compute L with the node pairs with direct wireless links:
        L= ConnectedList(par,map);
        % Compute C with the nodes with Internet access:
        C= ConnectedNodes(par,L);
        % Update of statistical counters:
        counter= UpdateCounter(counter,C);
        % Visualization of the simulation:
        if plotar>0
            visualize(par,map,L,C.',plotar);
        end

        end
        % Compute the final result:
        size(C);
        size(counter);

        [AverageAvailability(n), MinimumAvailability(n)]= results(T,counter);
    end

    %delete(h);
    toprint = fprintf("Simulating case %s where N = %i and W = %i ",CASES(o-1),par.N,par.W);
    disp(toprint);
    AverageMatrix(o-1,1) = mean(AverageAvailability);
    AverageMatrix(o-1,2) = norminv(1-alfa/2)*sqrt(var(AverageAvailability)/N);
    MinimumMatrix(o-1,1) = mean(MinimumAvailability);
    MinimumMatrix(o-1,2) = norminv(1-alfa/2)*sqrt(var(MinimumAvailability)/N);
    toprint = fprintf("Average,90ConfAverafe,Minimum,90ConfMinimum %1.8f %1.8f%1.8f%1.8f...
        ,AverageMatrix(o-1,1), AverageMatrix(o-1,2),MinimumMatrix(o-1,1),MinimumMatrix(o-1,2));
    disp(toprint);

end
toc;

```

```

function map= InitializeState(par)
N= par.N;
X= 400;
Y= 300;
pos= [50*randi([0 floor(Y/50)],N/2,1) Y*rand(N/2,1)];
pos= [pos; X*rand(N/2,1) 50*randi([0 floor(X/50)],N/2,1)];
vel_abs= 2*rand(N,1);
vel_cont= randi(100,N,1);
vel_angle= pi*randi([0 1],N/2,1) - pi/2;
vel_angle= [vel_angle; pi*randi([0 1],N/2,1)];
vel= [vel_abs.*cos(vel_angle) vel_abs.*sin(vel_angle)];
map.pos= pos;
map.vel= vel;
map.vel_cont= vel_cont;
end

function counter= InitializeCounter(par,N)
% counter - an array with par.N values (one for each mobile node) to count
%       the number of time slots each node has Internet access
% This function creates the array 'counter' and initializes it with zeros
% in all positions.
counter = zeros(1,par.N);
end

function map= UpdateState(par,map)
X= 400;
Y= 300;
N= par.N;
pos= map.pos;
vel= map.vel;
vel_cont= map.vel_cont;
delta= par.delta;
max_pos= [X*ones(N,1) Y*ones(N,1)];
continuar= [vel_cont>0 vel_cont>0];
pos= pos + delta*continuar.*vel;
vel(pos<=0)= -vel(pos<=0);
pos(pos<0)= 0;
vel(pos>=max_pos)= -vel(pos>=max_pos);
pos(pos>max_pos)= max_pos(pos>max_pos);
map.pos= pos;
map.vel= vel;
aux= zeros(N,1);

```

```

for j=find(vel_cont==1)
    aux(j)= -randi(40);
end
for j=find(vel_cont===-1)
    aux(j)= randi(100);
end
aux(vel_cont>1)= vel_cont(vel_cont>1)-1;
aux(vel_cont<-1)= vel_cont(vel_cont<-1)+1;
map.vel_cont= aux;
end

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

function L= ConnectedList(par,map)
% map.pos - a matrix with par.N rows and 2 columns; each row identifies the (x,y)
% coordinates of each mobile node
% L - a matrix with 2 columns; each row identifies a pair of nodes (mobile
% nodes and AP nodes) with direct wireless links between them
% This function computes matrix 'L' based on matrix map.pos.
assist = [map.pos; par.AP];
dist = pdist2(assist, assist);
Lbool = dist < par.W & dist ~= 0;
[I,J] = find(triu(Lbool,1));
L = [I,J];
end

function C= ConnectedNodes(par,L)
% C - an array with N values (one for each mobile node) that is 1 in
% position i if mobile node i has Internet access
% This function computes array 'C' based on matrix 'L' with the node pairs
% that have direct wireless links.
%
% NOTE: To develop this function, check MATLAB function 'distances' that
% computes shortest path distances of a graph.
x = par.N +par.nAP+1;
L_temp = L;
for n = 1:par.nAP
    L_temp = [ L_temp ; x (par.N+n)];

```

```

end
G = graph(L_temp(:,1),L_temp(:,2)); % plot(G);
C = distances(G,1:par.N,x).';
C = C < Inf;
end

function [AverageAvailability, MinimumAvailability]= results(T,counter)
% This function computes the average and the minimum availability (values
% between 0 and 1) based on array 'counter' and on the total number of
% time slots T.

AverageAvailability = mean(counter)/T;
MinimumAvailability = min(counter)/T;
end

function visualize(par,map,L,C,plotar)
N= par.N;
X= 400;
Y= 300;
pos= map.pos;
AP= par.AP;
nAP= par.nAP;
plot(AP(1:nAP,1),AP(1:nAP,2),'s','MarkerEdgeColor','k','MarkerFaceColor','k','MarkerSize',12)
hold on
plot(pos(1:N,1),pos(1:N,2),'o','MarkerEdgeColor','r','MarkerFaceColor','r','MarkerSize',5)
hold on
%rad = ones(par.N,1).*par.W;
%viscircles(pos,rad)

if plotar==2
pos=[pos;AP];
for i=1:size(L,1)
    plot([pos(L(i,1),1) pos(L(i,2),1)],[pos(L(i,1),2) pos(L(i,2),2)],'b')
end
plot(pos(C,1),pos(C,2),'o','MarkerEdgeColor','b','MarkerFaceColor','b')
end
axis([-20 X+20 -20 Y+20])
grid on
set(gca,'xtick',0:50:X)
set(gca,'ytick',0:50:Y)
drawnow
hold off
end

```