

Capítulo 5

Implementación en Matlab de Algoritmos aplicables a la cobertura en redes inalámbricas

5.1

Introducción al programa Matlab

MATLAB es un lenguaje de programación de alto nivel, con un enfoque directo hacia la computación científica. Más concretamente, es un programa de cálculo numérico con gran número de instrucciones dirigidas a la resolución de problemas científicos. Desde este punto de vista puede ser considerado entonces como una gran calculadora científica programable y muy potente.

El nombre de MATLAB proviene de las palabras inglesas *Matrix Laboratory* (laboratorio de matrices), que evidentemente dan una idea de la utilidad primordial de dicho programa. Algunas de las aplicaciones del programa MATLAB pueden ser:

- Computación y matemáticas.
- **Desarrollo de algoritmos.**
- Modelado y simulación.
- Exploración, visualización y análisis de datos.
- Creación de gráficas científicas.

Estas características comentadas hacen del programa MATLAB una herramienta de trabajo muy extendida tanto entre estudiantes como entre técnicos e investigadores.

5.2

Utilización del Matlab en este Proyecto

Como hemos visto, el Matlab es un programa de amplio desarrollo que permite una gran variedad en su utilización. Hemos hecho uso de esta herramienta por el gran número de posibilidades que permite. En este caso en concreto, su uso va destinado

mayormente a la representación gráfica de los algoritmos que implementemos. Pero no solo eso, si no que también mostraremos los resultados en tres dimensiones del efecto de la variación de ciertos parámetros como el Área de Cobertura o la Detectabilidad de las grandes redes de sensores, por citar algunos de ellos, y los podremos ver gráficamente.

5.3

Área de Cobertura

Hemos implementado en Matlab el Área de Cobertura de una red de sensores a gran escala, y para ello hemos dispuesto dos casos, según los parámetros de los que depende su función densidad de probabilidad. En primer lugar haremos variar el parámetro α , seguidamente del parámetro λ , y veremos cómo influyen en la representación gráfica de la función.

Así pues, aquí está el código en Matlab:

1) Código fuente en Matlab del Área de Cobertura:

```
%Funcion que calcula el Area de Cobertura

function y=areacobertura()

%La funcion densidad de probabilidad sera FDP

%FDP=((lambda.*pi.*alpha.^(1./2))./2).*(x.^(-3./2)).*exp((-
lambda.^2.*pi.^3.*alpha)./(4.*x));

%Asi pues, el Area de Cobertura sera:

%Elegiremos los valores de:

%alpha=3;

%lambda=4;
```

```
Respuesta=quad('((4.*pi.*3.^(1./2))./2).*(x.^(-3./2)).*exp((-  
4.^2.*pi.^3.*3)./(4.*x))',5,6);
```

```
fprintf('El Area de Cobertura es %.16d\n',Respuesta);
```

%Lo siguiente sera representar el Area de Cobertura en funcion de la variable X,

%para 2 casos distintos:

%1)Primero en funcion de la variable X, y de lambda, que aqui representaremos

% como variable Y:

```
[X,Y]=meshgrid([0.1:0.1:5],[0.1:0.1:5]);
```

```
Flambda=((Y.*pi.*3.^(1./2))./2).*(X.^(-3./2)).*exp((-Y.^2.*pi.^3.*3)./(4.*X));
```

%Contour pinta las curvas de nivel

```
contour(X,Y,Flambda);
```

%Meshgrid hace una grafica en 3D

```
mesh(X,Y,Flambda);
```

%2)Ahora como funcion de la variable Z, y de alpha, que igualmente representaremos

% como variable T:

```
[Z,T]=meshgrid([0.1:0.1:5],[0.1:0.1:5]);
```

```
Falpha=((4.*pi.*T.^(1./2))./2).*(Z.^(-3./2)).*exp((-4.^2.*pi.^3.*T)./(4.*Z));
```

%Contour pinta las curvas de nivel

```
contour(Z,T,Falpha);
```

%Meshgrid hace una grafica en 3D

```
mesh(Z,T,Falpha);
```

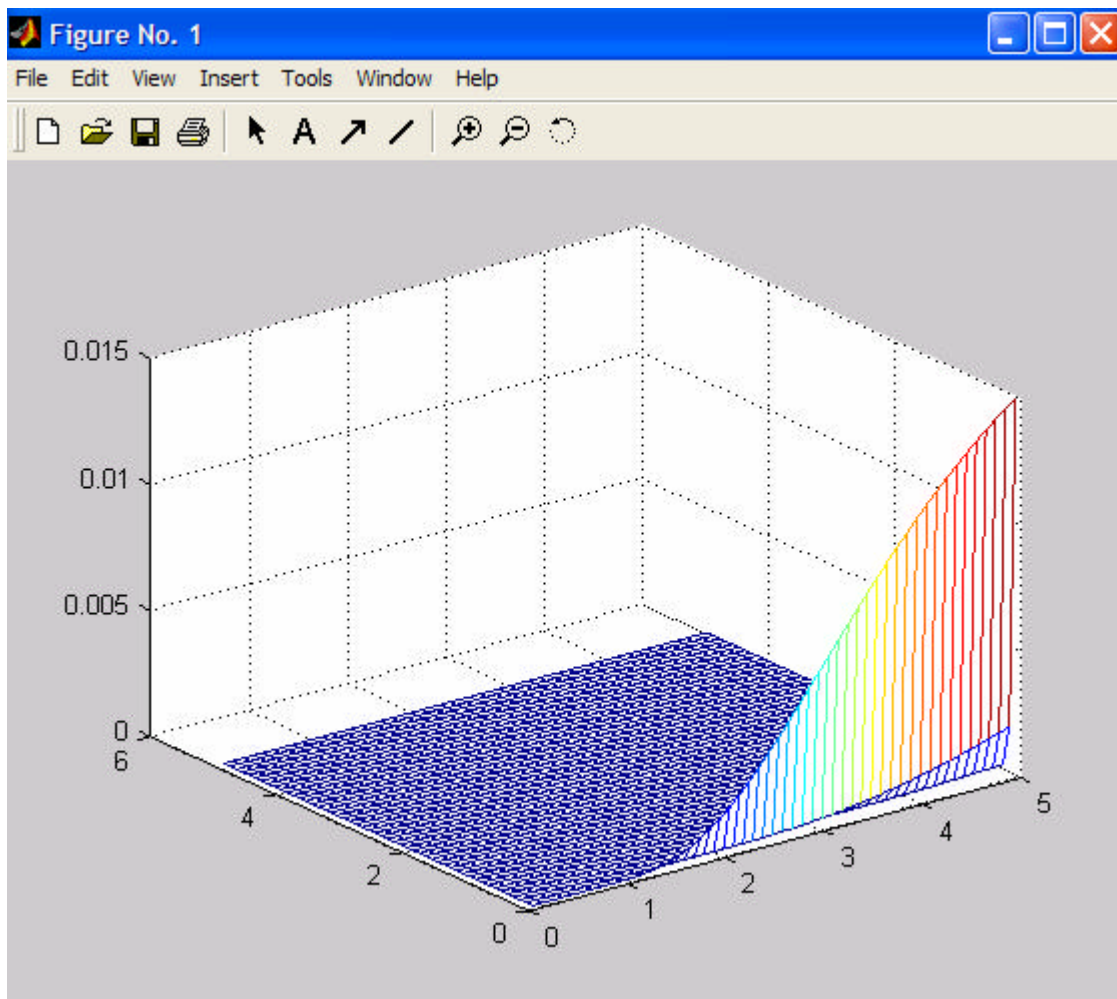
```
end
```

2) Valor de la función:

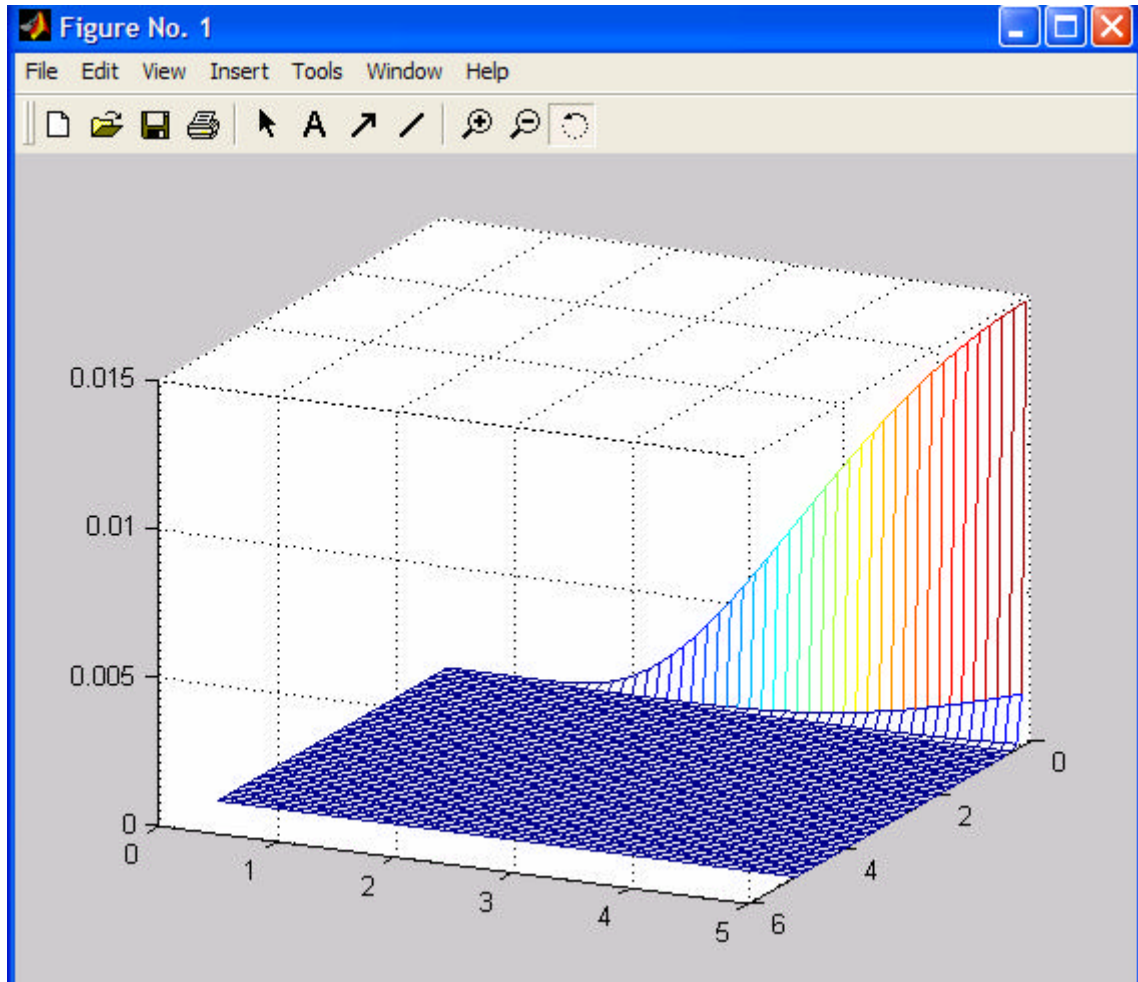
El Área de Cobertura es $8.3210597559471722e-029$

Este sería el resultado arrojado por Matlab tras lanzar su implementación. Vemos que es una función exponencial, como era de prever, lo cual va a ser una constante en la representación de todos los parámetros que caracterizan el comportamiento asintótico de una red inalámbrica.

1) Variación gráfica en función del parámetro λ :



2) Variación gráfica en función del parámetro α :



5.4

Detectabilidad

La Detectabilidad depende de cómo estén distribuidos los sensores y del número de ellos. A continuación expondremos el código en Matlab que hemos realizado para implementar esta propiedad que siempre viene asociada a la cobertura en redes inalámbricas de largo alcance.

Tras ver la gráfica representaremos un resultado real para unos parámetros en particular. En la función le pasaremos por parámetro el número de sumatorios n , ya que realmente la función Detectabilidad es un sumatorio de funciones:

1) Código fuente en Matlab de la Detectabilidad:

```
function y=detectabilidad(n)
```

```
%La funcion de Detectabilidad sera la siguiente, siendo
```

```
%1 y n los limites del sumatorio
```

```
%F=SUM((alpha)/((xi-x)^2+(yi-y)^2)^(beta/2),1,n);
```

```
%Inicializamos el valor de la funcion F
```

```
F=0;
```

```
for k=1:n
```

```
%Los valores de lapha y beta seran:
```

```
%alpha=3
```

```
%beta=4(para que el denominador siempre tenga valores positivos
```


%Y los valores de X e Y, seran:

%X=5

%Y=6

%Representarmos los diferentes puntos xi e yi

%con las diferentes iteraciones de k

$F = F + (3) / ((k-5)^2 + (k-6)^2)^{(4/2)}$;

fprintf('La detectabilidad vale 3.0%d\n',F);

pause;

end

%Representaremos la funcion asignando los valores de X y de Y

% a las dos iteraciones de k, que en este caso podrian ser k1 y k2:

[X,Y]=meshgrid([0.1:0.1:5],[0.1:0.1:5]);

$F_{detec} = (3) / ((X-5)^2 + (Y-6)^2)^{(4/2)}$;

%Contour pinta las curvas de nivel

contour(X,Y,Fdetec);

%Meshgrid hace una grafica en 3D

mesh(X,Y,Fdetec);

end

2) Valor de la función:

La detectabilidad vale 3.01.784652e-003

La detectabilidad vale 3.06.584652e-003

La detectabilidad vale 3.02.433613e-002

La detectabilidad vale 3.01.443361e-001

La detectabilidad vale 3.03.144336e+000

La detectabilidad vale 3.06.144336e+000

La detectabilidad vale 3.06.264336e+000

La detectabilidad vale 3.06.282088e+000

La detectabilidad vale 3.06.286888e+000

La detectabilidad vale 3.06.288672e+000

Lo hemos hecho para un número de iteraciones $n=10$. Como vemos la Detectabilidad también es una exponencial que irá aumentando a medida que vayamos realizando las iteraciones.

3) Representación gráfica de la Detectabilidad:

