

Fundamentos de Comunicaciones Inalámbricas - TP N°2

Matías Roqueta - Instituto Balseiro, Centro Atómico Bariloche

1 Introducción

En este trabajo práctico vamos a simular y estudiar los modelos estadísticos de un canal inalámbrico, un ejemplo de tal canal ilustrado en la Figura 1.

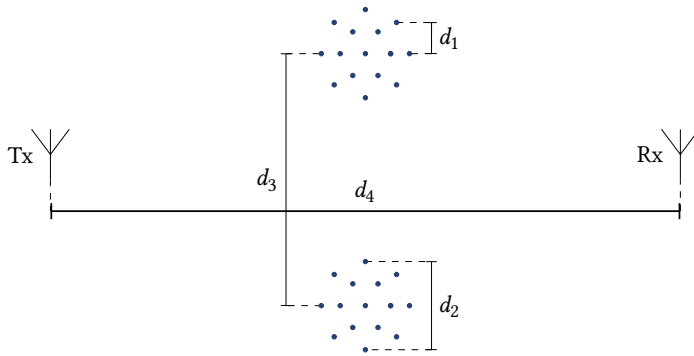


Figura 1: Canal inalámbrico de múltiples caminos entre un transmisor y un receptor en presencia de un número arbitrario de reflectores distribuidos en *clusters*.

Las distancias características del canal, d_1 , d_2 , d_3 , y d_4 , son de los siguientes órdenes de magnitud

- Distancia media entre reflectores en un cluster: $d_1 \sim 1$ m
- Diámetro medio de los clusters: $d_2 \sim 10$ m
- Distancia media entre los clusters: $d_3 \sim 100$ m
- Distancia entre transmisor y receptor: $d_4 \sim 1$ km

Para la simulación se cuenta con una librería de funciones relevantes a la tarea, detalladas en el Apéndice 1 (Sección 3).

2 Implementación del Canal

Se definen primero las posiciones del emisor Tx y el receptor Rx, fijas y del orden de 1 km.

```
posTx = (0, 0)
posRx = (1000, 0)
```

A continuación se definen aleatoriamente las posiciones de 5 clusters dentro de un círculo de radio de 250 m centrado en el punto medio entre Tx y Rx.

```
Cs = [(500, 0).+p.*250
      for p in Lib.genClust(5, 0, 0.75)]
```

Cada cluster tendrá 20 reflectores distribuidos uniformemente en un radio de 25 m.

```
Ns = 20 .+ zeros(size(Cs)) .|> Int
Rs = 25 .+ zeros(size(Cs))

S, Ss = Lib.genScatt(Ns, Rs, Cs)
```

Graficamos el canal resultante, obteniendo la Figura 2.



Figura 2: Realización del canal inalámbrico aleatorio

```
dst = genDists(S, posTx, posRx)
dsts = [genDists(s, posTx, posRx) for s in Ss]
```

3 Apéndice 1 - Documentación

clust = genClust(N; minR, minDist)

Genera un cluster de N posiciones uniformemente distribuidas en un círculo de diámetro unitario. Retorna un vector de tuplas (x,y).

El parámetro opcional minR indica una distancia mínima al centro del cluster que deben tener los elementos.

El parámetro opcional minDist indica una distancia mínima a otros elementos que deben tener los elementos.

S, Ss = genScatt(Ns, Rs, Cs)

Genera una distribución de reflectores organizados en clusters definidos por los vectores N_s , R_s , C_s de la siguiente forma

$N_s[i]$ número de reflectores en el cluster i -ésimo

$R_s[i]$ radio del cluster i -ésimo

$C_s[i]$ coordenadas del centro del cluster i -ésimo

Retorna S un vector de tuplas (x, y) representando las posiciones de los reflectores, y un vector de vectores S_s representando las posiciones de los reflectores agrupados en sus respectivos clusters.

dists = genDists(S , posTx, posRx)

Recibe un vector de tuplas S representando las posiciones (x, y) de reflectores, y dos tuplas posTx y posRx representando las posiciones de un emisor y receptor.

Retorna un vector con las distancias de todos los posibles caminos entre el emisor y el receptor.

$D_s = \text{delaySpread}(D_s)$

Recibe como parametro un vector dists de distancias (en metros) y retorna el delay spread T_d (en segundos) entre esas distancias.