

1 Diseño antena

El estudiante debe realizar un notebook donde plantee el problema de optimización a partir del enunciado dado y encuentre la solución optima por medio de técnicas de optimización NLP con restricciones.

1.1 Descripción

Se requiere estimar los parámetros del diseño de una antena (la cual esta basada en cables doblados) que permita aumentar la ganancia en la frecuencia de resonancia seleccionada.

Table 1: Descripción de los problemas de optimización.

Item	3 segmentos	5 segmentos
Función Objetivo	Ganancia máxima	Ganancia media
Frecuencia resonancia	2.5 GHz	5 GHz

Con la frecuencia de resonancia asignada, se debe poner como restricción que el conjunto de cables doblados no supere el espacio ocupado por un cubo de $\lambda/2$, donde λ es la longitud de onda (no confundirlo con el λ empleado en multiplicadores de Lagrange), como se muestra en la figura 1.

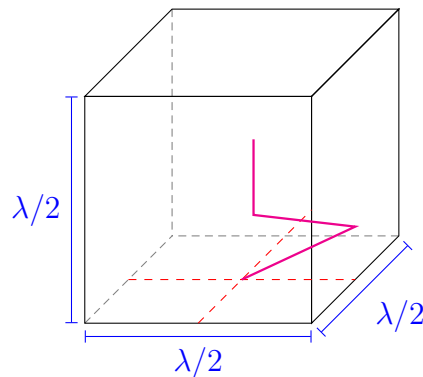


Figure 1: Esquema de diseño de la antena, con restricciones dentro del cubo. En rojo el plano de la tierra, y en magenta un ejemplo de una antena doblada en 3 segmentos.

A continuación se presenta un ejemplo en Python usando la libreria necpp [1].

1.2 Código en Python

```
import necpp

# Parameters
freq = 2400 #MHz
wireThickness = 2 #mm
maxWireLength = 0.35*0.001 #mm
directionality = 2 #0 = maximize max gain, 1 = maximize min gain, 2 = maximize average gain
```

```
def calcGain(params): #Example with 2 wires
x1 = params[0]
y1 = params[1]
z1 = params[2]
x2 = params[3]
y2 = params[4]
z2 = params[5]

#create an object on which to include wires and metal areas
nec = necpp.nec_create()
#we include some interconnected cables
necpp.nec_wire(nec, 1, 15, 0.0, 0.0, 0.0, x1, y1, z1, wireThickness, 1.0, 1.0)
necpp.nec_wire(nec, 1, 15, x1, y1, z1, x2, y2, z2, wireThickness, 1.0, 1.0)
#Finish the model to be simulated
necpp.nec_geometry_complete()

#Test signals set-up
#check: https://pypi.org/project/necpp/
#Define range of frequencies for the simulation from 1 to freq
necpp.nec_fr_card(nec, 0, 1, freq, 0.0)
#Define ground's location
necpp.nec_gn_card(nec, 1, 0, 0, 0, 0, 0, 0, 0, 0)
#Define excitation source parameters
necpp.nec_ex_card(nec, 0, 1, 1, 0, 1.0, 0, 0, 0, 0, 0)
#Define radiation pattern parameters
necpp.nec_rp_card(nec, 0, 17, 45, 0, 5, 0, 0, 0, 0, 5, 8, 0, 0)

# Objective function output
if (directionality == 0):
gain = necpp.nec_gain_max(nec, 0)
elif (directionality==1):
gain = necpp.nec_gain_min(nec, 0)
else:
gain = necpp.nec_gain_mean(nec, 0)

necpp.nec_delete(nec)
return gain
```

1.3 Procedimiento

1. Escribir el código en Python del conjunto de ecuaciones del problema de optimización: función objetivo, restricciones y límites de las variables. Describir las razones por las cuales se escribe cada ecuación.
2. Implementar los métodos vistos en clase para solucionar problemas NLP con restricciones. Específicamente, método de penalización cuadrático (QPM), método del Lagrangiano aumentado (ALM) y método de la barrera logarítmica (LBM). Estos modelos están diseñados para problemas NLP con restricciones de desigualdad.

3. Comparar el desempeño de los métodos de optimización, comparando cantidad de iteraciones y tiempo de ejecución de cada iteración. Probar con diferentes puntos de inicialización y analizar la convergencia de los métodos.
4. Emplear una librería para comparar con los resultados obtenidos en el paso 2.
5. Escribir en cada paso anterior el análisis realizado y al final las conclusiones.

2 Informe

Desarrollar un notebook en Python, que incluya las siguientes secciones:

1. Introducción al problema.
2. Código y desarrollo de la solución.
3. Análisis de resultados.
4. Conclusiones.
5. Bibliografía.

2.1 Videos recomendados

- Introducción a la simulación de antenas. https://www.youtube.com/watch?v=vaLx3_F6-b4.
- Medición de parámetros con 4NEC2. <https://www.youtube.com/watch?v=s-AaloLVbaY>
- Diseño de antenas con Python (antenas evolucionadas/WiFi de 2.4 GHz). <https://www.youtube.com/watch?v=Rn0dtLyNceI>

Referencias

- [1] NEC++ functions description, howpublished = https://tmolteno.github.io/necpp/libnecpp_8h.html, note = Accessed: 2023-09-14.