**Experiment-7**

**Aim:** To implement concept code for the diffraction loss and Knife –edge diffraction propagation model for large scale propagation loss.

**Activities:**

**A.** To implement and execute a program for analytical calculation of Knife-edge diffraction loss and plot the graph with respect to distance.

**Theory:**

The path loss due to diffraction in the knife edge model is controlled by the Fresnel Diffraction Parameter which measures how deep the receiver is within the shadowed region. A negative value for the parameter shows that the obstruction is below the line of sight and if the value is below -1 there is hardly any loss. A value of 0 (zero) means that the transmitter, receiver and tip of the obstruction are all in line and the Electric Field Strength is reduced by half or the power is reduced to one fourth of the value without the obstruction i.e., a loss of 6dB. As the value of the Fresnel Diffraction Parameter increases on the positive side the path loss rapidly increases reaching a value of 27 dB for a parameter value of 5. Sometimes the exact calculation is not needed and only an approximate calculation, as proposed by Lee in 1985, is sufficient.

**Python Code:**

import math import numpy as np

import matplotlib.pyplot as plt

def knife\_edge\_diffraction(frequency, distance, height\_of\_transmitter, height\_of\_receiver, obstruction\_height):

# Calculate effective Earth radius (Re) using the ITU-R P.526 model earth\_radius = 8500.0 # Earth radius in kilometers k1 = 15.0 # Constant in the model k2 = 0.1 # Constant in the model

Re = earth\_radius / (1 + k1 \* math.exp(-k2 \* height\_of\_receiver))

# Calculate Fresnel zone clearance d1 = math.sqrt(distance \* (2 \* Re + distance)) d2 = math.sqrt(distance \* (2 \* (Re + height\_of\_transmitter) + distance))

d = d1 + d2

# Calculate the knife-edge diffraction loss if height\_of\_transmitter <= obstruction\_height and height\_of\_receiver <= obstruction\_height: diffraction\_loss = 0.0

elif height\_of\_transmitter <= obstruction\_height: diffraction\_loss = 6.9 + 20 \* math.log10(frequency) + 20 \* math.log10(d)

elif height\_of\_receiver <= obstruction\_height:

diffraction\_loss = 6.9 + 20 \* math.log10(frequency) + 20 \* math.log10(d) else:

alpha = math.sqrt(2 \* math.pi \* (obstruction\_height - height\_of\_transmitter) \*

(obstruction\_height - height\_of\_receiver) / frequency) beta = d / (2 \* alpha) diffraction\_loss = 6.9 + 20 \* math.log10(frequency) + 10 \* math.log10(beta\*\*2 + 1) return diffraction\_loss

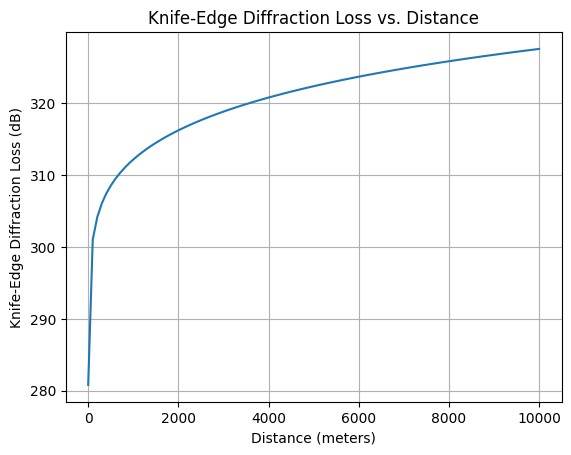
# Parameters frequency = 900e6 # 900 MHz height\_of\_transmitter = 30 # 30 meters height\_of\_receiver = 10 # 10 meters obstruction\_height = 5 # 5 meters

# Range of distances to consider distances = np.linspace(1, 10000, 100) # From 1 to 10,000 meters

# Calculate Knife-edge diffraction loss for each distance losses = [knife\_edge\_diffraction(frequency, d, height\_of\_transmitter, height\_of\_receiver, obstruction\_height) for d in distances]

# Plot the results plt.figure() plt.plot(distances, losses) plt.xlabel('Distance (meters)') plt.ylabel('Knife-Edge Diffraction Loss (dB)') plt.title('Knife-Edge Diffraction Loss vs. Distance') plt.grid(True) plt.show()

Output:



**Conclusion:**

From this graphical visualization, we can understand Fresnel Diffraction clearly.