

# РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра теории вероятностей и кибербезопасности

## ОТЧЕТ

### ПО ПРАКТИЧЕСКОЙ РАБОТЕ №1

дисциплина: Построение и анализ моделей беспроводных сетей 5G/6G

Студент: Быстров Глеб Андреевич

Группа: НФИмд-01-24

Преподаватель: Бегишев Вячеслав Олегович

МОСКВА

2024 г.

**Цель работы:**

Изучение различных моделей распространения радиосигнала, анализ затухания сигнала в зависимости от частоты и расстояния, а также оценка влияния условий среды на дальность и уровень приема сигнала. Построение и сравнение графиков затухания для моделей FSPL, UMa, и InH в условиях прямой и не прямой видимости.

**Список сокращений:**

- FSPL — Free Space Path Loss (Потери на свободном пространстве)
- LoS — Line of Sight (Прямая видимость)
- nLoS — Non-Line of Sight (Непрямая видимость)
- UMa — Urban Macro (Городская макросреда)
- InH — Indoor (Внутренние помещения, офисная среда)

## Выполнение работы

### Задание.

1. Запишите модель распространения сигнала FSPL (Free Space Path Loss) и переведите ее из линейной шкалы в логарифмическую. Напомним, что для логарифмической шкалы децибел используют преобразование  $x_{dB} = 10 \log_{10}(x_{linear})$ .

$$FSPL(d) = \left( \frac{4\pi df}{c} \right)^2,$$

где  $d$  – расстояние,  $f$  – частота радиоволны,  $c$  – скорость света.

2. Постройте графики значений модели FSPL как функцию уровня затухания радиосигнала от расстояния для частот  $f_1 = 900$  МГц,  $f_2 = 1.9$  ГГц и  $f_3 = 28$  ГГц.
3. Объясните удобство использования шкалы децибел. Проанализируйте и сделайте вывод о влиянии расстояния и частоты на потери распространения.

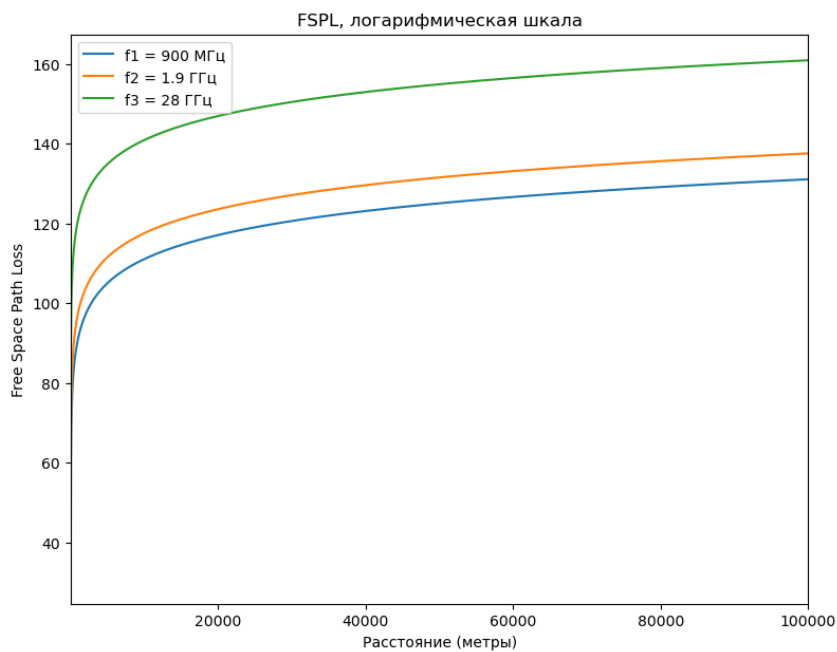
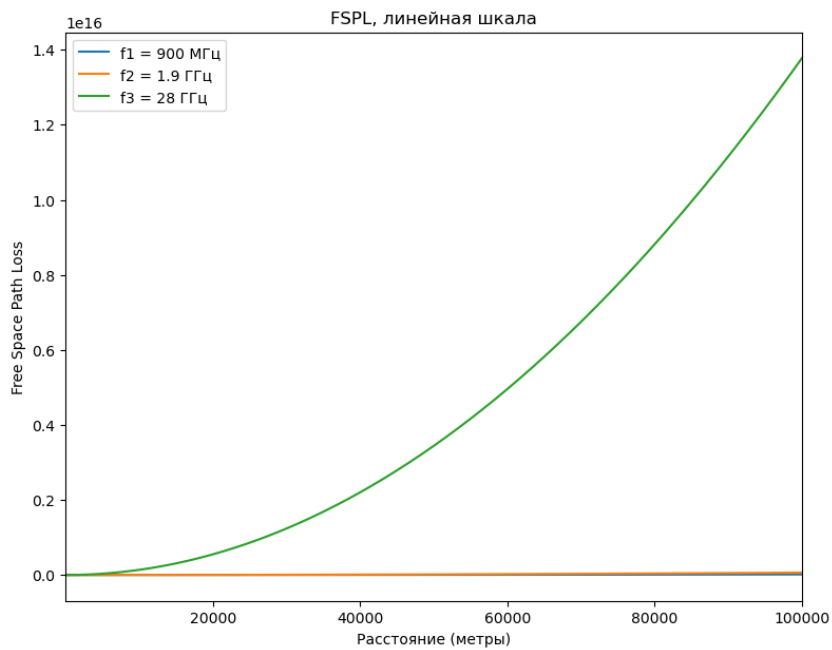
```
import numpy as np
import scipy.constants as spc
import matplotlib.pyplot as plt

def linear(d, f):
    return ((4.0 * np.pi * f * d) / spc.c)**2.0
def log(d, f):
    return 20 * np.log10(d) + 20 * np.log10(f) - 148

distances = np.linspace(1, 100000, 10000)
f1 = 900000000
f2 = 1900000000
f3 = 2800000000

plt.figure(figsize=(9,7))
plt.plot(distances, [linear(i, f1) for i in distances], label='f1 = 900 МГц')
plt.plot(distances, [linear(i, f2) for i in distances], label='f2 = 1.9 ГГц')
plt.plot(distances, [linear(i, f3) for i in distances], label='f3 = 28 ГГц')
plt.title("FSPL, линейная шкала")
plt.xlabel("Расстояние (метры)")
plt.ylabel("Free Space Path Loss")
plt.xlim(xmin=distances[0], xmax=distances[-1])
plt.legend()

plt.figure(figsize=(9,7))
plt.plot(distances, [log(i, f1) for i in distances], label='f1 = 900 МГц')
plt.plot(distances, [log(i, f2) for i in distances], label='f2 = 1.9 ГГц')
plt.plot(distances, [log(i, f3) for i in distances], label='f3 = 28 ГГц')
plt.title("FSPL, логарифмическая шкала")
plt.xlabel("Расстояние (метры)")
plt.ylabel("Free Space Path Loss")
plt.xlim(xmin=distances[0], xmax=distances[-1])
plt.legend()
```



Шкала децибел позволяет компактно отображать большие диапазоны потерь, переводя умножение в сложение, что упрощает анализ. Потери увеличиваются с ростом расстояния. Чем выше частота, тем сильнее затухание. Таким образом, увеличение расстояния и частоты приводит к значительным потерям сигнала.

## Дистанция уверенного приема

4. Рассчитайте максимально возможную дистанцию уверенного приема для передатчика на частотах  $f_1 = 900$  МГц,  $f_2 = 1.9$  ГГц и  $f_3 = 28$  ГГц с мощностью передачи в 23 дБм. Усиления антенн на передаче и приеме считать одинаковыми и равными 10 дБ. Чувствительность приемника, т.е. минимальный уровень сигнала, который приемник может принять и различить, равна -70 дБм.
5. Изменение каких параметров позволит увеличить зону покрытия беспроводной сети связи?

$$P_r(d) = P_t G_t G_r \left( \frac{c}{4\pi d f} \right)^2,$$

где  $d$  – расстояние,  $f$  – частота радиоволны,  $c$  – скорость света,  $G_t$  и  $G_r$  – усиления антенн на передаче и приеме соответственно,  $P_t$  – мощность передатчика.

```
def calc(d, f, P, Gt, Gr):
    c = spc.c
    part1 = (c / (4 * np.pi * d * f))**2
    part2 = Gt * Gr * P
    formula = part1 * part2
    return 10 * np.log10(formula)

def maxd(distances, f, L, P, Gt, Gr):
    d = 0
    for i in distances:
        if calc(i, f, P, Gt, Gr) >= L:
            d = i
        else:
            break
    return d

P = 23
Gr = Gt = 10
L = -70

print("Макс дистанция для f = 900МГц: {}".format(maxd(distances, f1, L, P, Gt, Gr)))
print("Макс дистанция для f = 1.9ГГц: {}".format(maxd(distances, f2, L, P, Gt, Gr)))
print("Макс дистанция для f = 28ГГц: {}".format(maxd(distances, f3, L, P, Gt, Gr)))

Макс дистанция для f = 900МГц: 4011.3609360936093
Макс дистанция для f = 1.9ГГц: 1901.17101710171
Макс дистанция для f = 28ГГц: 121.01080108010801
```

Для увеличения зоны покрытия беспроводной сети можно:

1. Увеличить мощность передатчика
2. Усилить антенны на передаче и приеме.
3. Снизить частоту радиоволны.

Эти изменения приведут к повышению уровня принимаемого сигнала на заданном расстоянии.

## Стандартные модели распространения 3GPP

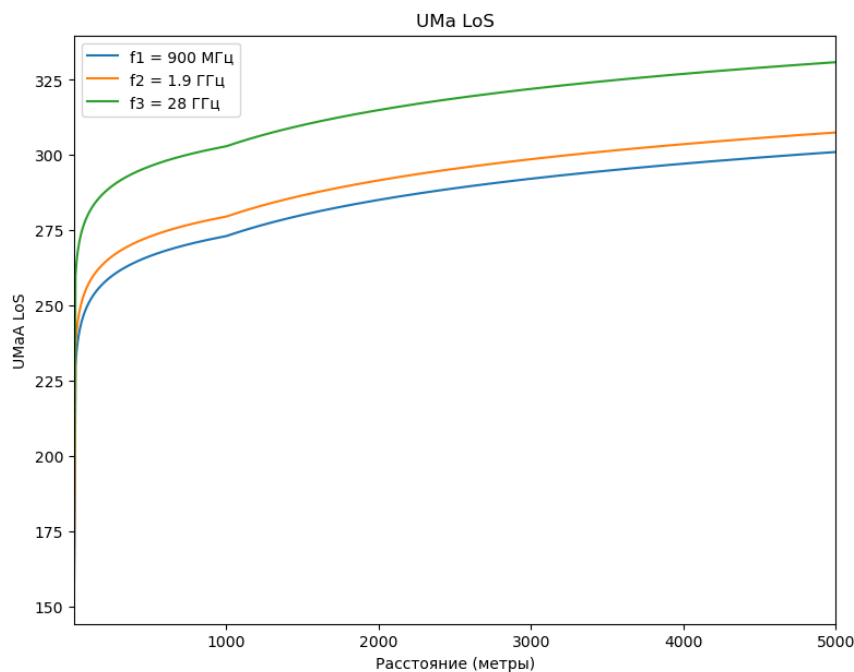
- Построить графики зависимости уровня затухания радиосигнала от расстояния для моделей UMa LoS, UMa nLoS, InH-Office LoS, InH-Office nLoS из стандарта 3GPP TR 38.901.
- Проведите сравнительный анализ качественного поведения данных моделей распространения радиосигнала с моделью FSPL.

LoS	$PL_{\text{UMa-LoS}} = \begin{cases} PL_1 & 10\text{m} \leq d_{2D} \leq d'_{\text{BP}} \\ PL_2 & d'_{\text{BP}} \leq d_{2D} \leq 5\text{km}, \text{ see note 1} \end{cases}$	$\sigma_{\text{SF}} = 4$	$1.5\text{m} \leq h_{\text{UT}} \leq 22.5\text{m}$ $h_{\text{BS}} = 25\text{m}$
	$PL_1 = 28.0 + 22 \log_{10}(d_{3D}) + 20 \log_{10}(f_c)$ $PL_2 = 28.0 + 40 \log_{10}(d_{3D}) + 20 \log_{10}(f_c) - 9 \log_{10}((d'_{\text{BP}})^2 + (h_{\text{BS}} - h_{\text{UT}})^2)$		

```
def UMa_LoS(d, fc):
    PL = 0
    dbp = 1000
    hbs = 25
    hut = 15
    if 10 <= d <= dbp:
        PL = 28.0 + 22*np.log10(d) + 20*np.log10(fc)
    else:
        PL = 28.0 + 40*np.log10(d) + 20*np.log10(fc) - 9*np.log10(dbp**2 + (hbs - hut)**2)
    return PL

distances = np.linspace(1, 5000, 1000)
f1 = 900000000
f2 = 1900000000
f3 = 2800000000

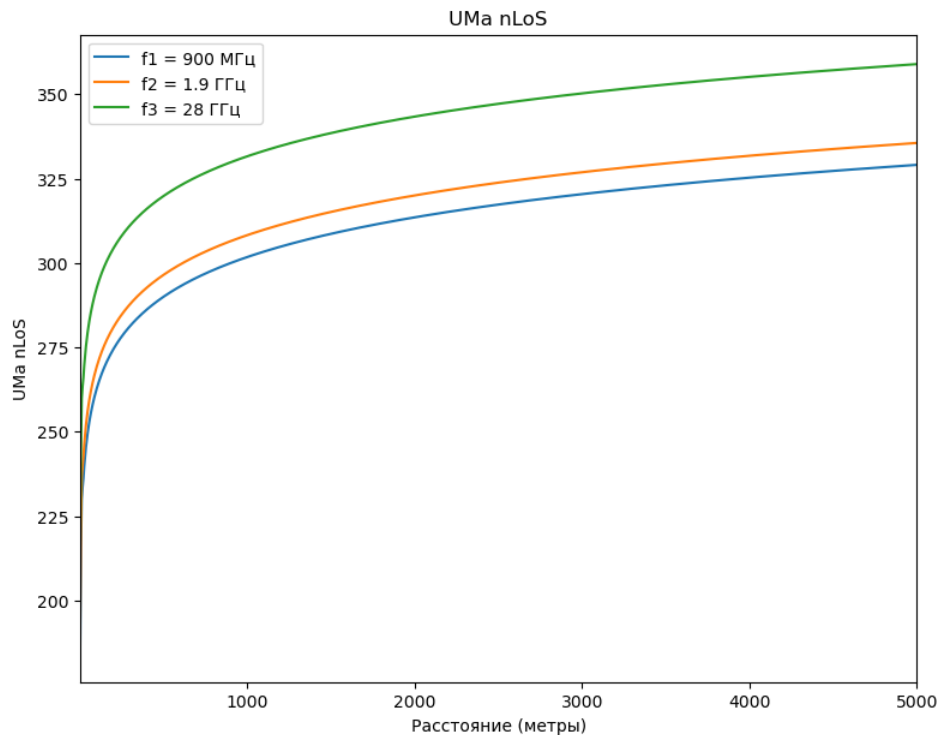
plt.figure(figsize=(9,7))
plt.plot(distances, [UMa_LoS(i, f1) for i in distances], label='f1 = 900 МГц')
plt.plot(distances, [UMa_LoS(i, f2) for i in distances], label='f2 = 1.9 ГГц')
plt.plot(distances, [UMa_LoS(i, f3) for i in distances], label='f3 = 28 ГГц')
plt.title("UMa LoS")
plt.xlabel("Расстояние (метры)")
plt.ylabel("UMa LoS")
plt.xlim(xmin=distances[0], xmax=distances[-1])
plt.legend()
```



NLOS	$PL_{\text{UMa-NLOS}} = \max(PL_{\text{UMa-LOS}}, PL'_{\text{UMa-NLOS}})$ for $10\text{m} \leq d_{3D} \leq 5\text{km}$	$\sigma_{\text{SF}} = 6$	$1.5\text{m} \leq h_{\text{UT}} \leq 22.5\text{m}$ $h_{\text{BS}} = 25\text{m}$ Explanations: see note 3
	$PL'_{\text{UMa-NLOS}} = 13.54 + 39.08 \log_{10}(d_{3D}) + 20 \log_{10}(f_c) - 0.6(h_{\text{UT}} - 1.5)$		

```
def UMa_nLoS(d, fc):
    hbs = 25
    hut = 15
    PLL = 13.54 + 39.08 * np.log10(d) + 20*np.log10(fc) - 0.6 * (hut - 1.5)
    PL = UMa_LoS(d, fc)
    if PL > PLL:
        return PL
    else:
        return PLL

plt.figure(figsize=(9,7))
plt.plot(distances, [UMa_nLoS(i, f1) for i in distances], label='f1 = 900 МГц')
plt.plot(distances, [UMa_nLoS(i, f2) for i in distances], label='f2 = 1.9 ГГц')
plt.plot(distances, [UMa_nLoS(i, f3) for i in distances], label='f3 = 28 ГГц')
plt.title("UMa nLoS")
plt.xlabel("Расстояние (метры)")
plt.ylabel("UMa nLoS")
plt.xlim(xmin=distances[0], xmax=distances[-1])
plt.legend()
```

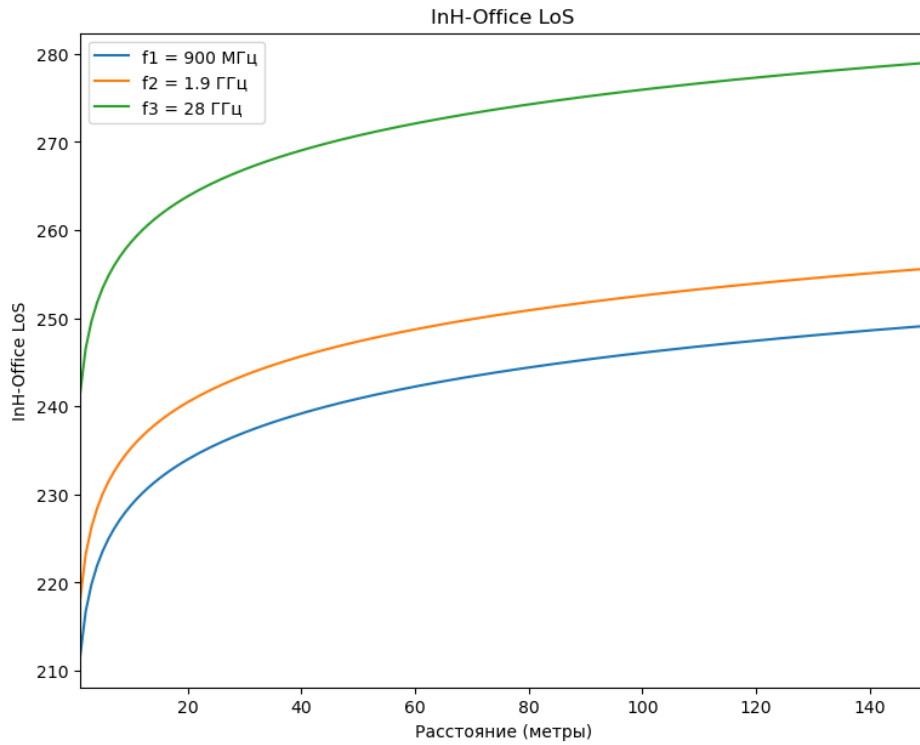


LOS	$PL_{\text{InH-LOS}} = 32.4 + 17.3 \log_{10}(d_{3D}) + 20 \log_{10}(f_c)$	$\sigma_{\text{SF}} = 3$	$1\text{m} \leq d_{3D} \leq 150\text{m}$
-----	---	--------------------------	--

```
def InH_Office_LoS(d, fc):
    PL = 32.4 + 17.3*np.log10(d) + 20*np.log10(fc)
    return PL

distances = np.linspace(1, 150, 150)

plt.figure(figsize=(9,7))
plt.plot(distances, [InH_Office_LoS(i, f1) for i in distances], label='f1 = 900 МГц')
plt.plot(distances, [InH_Office_LoS(i, f2) for i in distances], label='f2 = 1.9 ГГц')
plt.plot(distances, [InH_Office_LoS(i, f3) for i in distances], label='f3 = 28 ГГц')
plt.title("InH-Office LoS")
plt.xlabel("Расстояние (метры)")
plt.ylabel("InH-Office LoS")
plt.xlim(xmin=distances[0], xmax=distances[-1])
plt.legend()
```



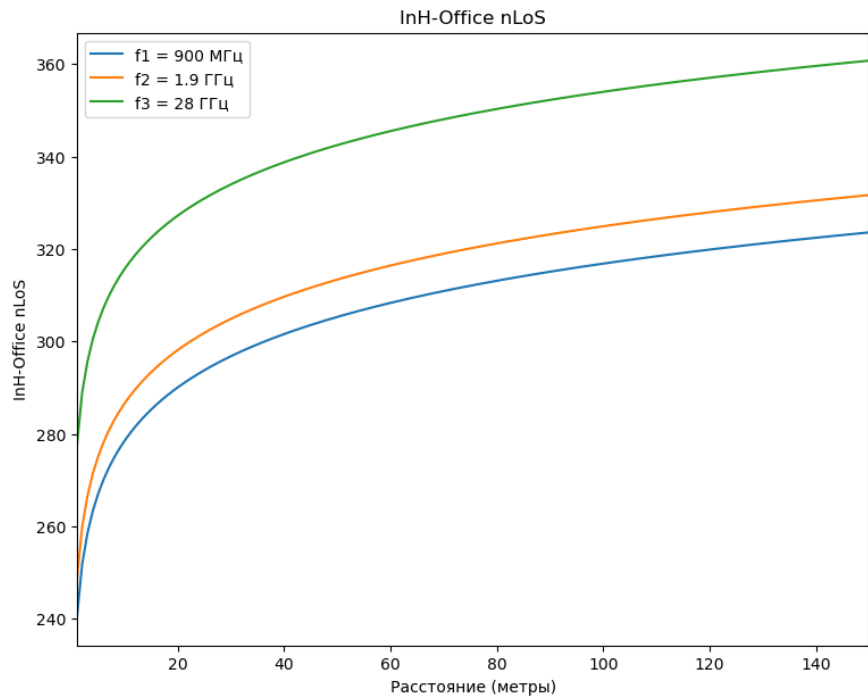
NLOS	$PL_{\text{InH-NLOS}} = \max(PL_{\text{InH-LOS}}, PL'_{\text{InH-NLOS}})$	$\sigma_{\text{SF}} = 8.03$	$1\text{m} \leq d_{3\text{D}} \leq 150\text{m}$
	$PL'_{\text{InH-NLOS}} = 38.3 \log_{10}(d_{3\text{D}}) + 17.30 + 24.9 \log_{10}(f_c)$		
	Optional $PL'_{\text{InH-NLOS}} = 32.4 + 20 \log_{10}(f_c) + 31.9 \log_{10}(d_{3\text{D}})$		

```
def InH_Office_nLoS(d, fc):
    PL = InH_Office_LoS(d, fc)
    PLL = 38.3*np.log10(d) + 17.3 + 24.9*np.log10(fc)
    if PL > PLL:
        return PL
    else:
        return PLL

distances = np.linspace(1, 150, 150)

plt.figure(figsize=(9,7))
plt.plot(distances, [InH_Office_nLoS(i, f1) for i in distances], label='f1 = 900 МГц')
plt.plot(distances, [InH_Office_nLoS(i, f2) for i in distances], label='f2 = 1.9 ГГц')
plt.plot(distances, [InH_Office_nLoS(i, f3) for i in distances], label='f3 = 28 ГГц')
plt.title("InH-Office nLoS")
plt.xlabel("Расстояние (метры)")
plt.ylabel("InH-Office nLoS")
plt.xlim(xmin=distances[0], xmax=distances[-1])
plt.legend()
```

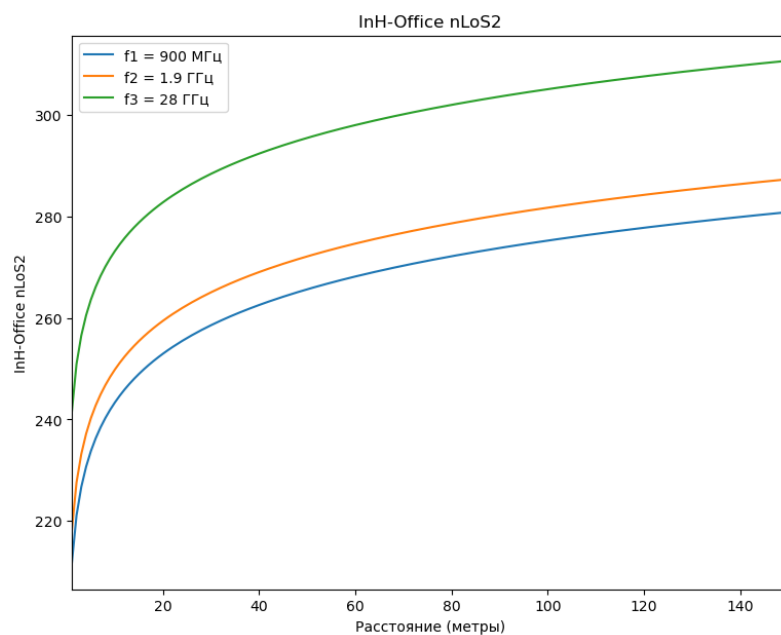




NLOS	$PL_{\text{InH-NLOS}} = \max(PL_{\text{InH-LOS}}, PL'_{\text{InH-NLOS}})$	$\sigma_{\text{SF}} = 8.03$	$1\text{m} \leq d_{3\text{D}} \leq 150\text{m}$
	$PL'_{\text{InH-NLOS}} = 38.3 \log_{10}(d_{3\text{D}}) + 17.30 + 24.9 \log_{10}(f_c)$		
	Optional $PL'_{\text{InH-NLOS}} = 32.4 + 20 \log_{10}(f_c) + 31.9 \log_{10}(d_{3\text{D}})$		

```
def InH_Office_nLoS2(d, fc):
    PL = InH_Office_LoS(d, fc)
    PLL = 32.4 + 20*np.log10(fc) + 31.9*np.log10(d)
    if PL > PLL:
        return PL
    else:
        return PLL

plt.figure(figsize=(9,7))
plt.plot(distances, [InH_Office_nLoS2(i, f1) for i in distances], label='f1 = 900 МГц')
plt.plot(distances, [InH_Office_nLoS2(i, f2) for i in distances], label='f2 = 1.9 ГГц')
plt.plot(distances, [InH_Office_nLoS2(i, f3) for i in distances], label='f3 = 28 ГГц')
plt.title("InH-Office nLoS2")
plt.xlabel("Расстояние (метры)")
plt.ylabel("InH-Office nLoS2")
plt.xlim(xmin=distances[0], xmax=distances[-1])
plt.legend()
```



Модель FSPL описывает затухание сигнала на свободном пространстве, где затухание зависит только от расстояния и частоты.

Другие модели (UMa и InH) учитывают особенности среды:

- UMa (Urban Macro): затухание сильнее, особенно в условиях не прямой видимости (nLoS), из-за препятствий в городской среде.
- InH (Indoor Office): затухание высокое даже на малых расстояниях, особенно без прямой видимости, из-за множества отражений и преград в офисной среде.

Таким образом, модели UMa и InH показывают более сильное затухание по сравнению с FSPL из-за влияния окружающих условий.

## **Заключение.**

Анализ графиков и моделей распространения радиосигнала показывает, что затухание сигнала значительно зависит от расстояния, частоты и окружающих условий. При увеличении частоты и расстояния потери сигнала возрастают, что особенно заметно в условиях без прямой видимости. Модель FSPL демонстрирует наименьшие потери, так как учитывает только свободное пространство. В отличие от неё, модели UMa и InH показывают более значительное затухание из-за влияния препятствий в городской и офисной средах.

В офисной среде (InH) потери на коротких расстояниях высоки из-за множества препятствий, в то время как в городской среде (UMa) затухание возрастает на больших расстояниях и также зависит от прямой или не прямой видимости.

Эта лабораторная работа позволяет понять принципы распространения радиосигнала в различных условиях и необходимость учитывать особенности среды для корректной оценки потерь.

Знания, полученные в результате лабораторной работы, важны для проектирования и оптимизации беспроводных сетей в реальных условиях, что помогает улучшить качество связи и оптимизировать покрытие.