

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ПРАКТИЧЕСКОЙ РАБОТЕ № 3

дисциплина: Построение и анализ

моделей беспроводных сетей 5G/6G

Студент:

Кубасов Владимир Юрьевич

ст.б. 1132249516

Группа:

НФИмд-01-24

Преподаватель:

Молчанов Дмитрий Александрович

Самуйлов Андрей Константинович

МОСКВА

2024г.

Цель работы:

Исследовать вероятности блокировки прямой видимости и изменения сигнала, связанные с блокировкой

Выполнение работы

Блокирующие объекты фиксированного размера

Рассмотрим сценарий блокировки канала связи между пользовательским устройством, и точкой доступа mmWave, схематично изображенный на рисунке. Пользовательское устройство расположено в точке К, а точка доступа – в точке О. С целью упрощения блокировка канала связи происходит, когда центральная плоскость цилиндра блокирующего объекта пересекает линию между передатчиком (точка А) и приемником (точка В).

1. Предположив, что блокирующие объекты имеют фиксированную высоту, запишите формулу зависимости площади зоны, в рамках которой должен находиться центр блокирующего объекта чтобы он заблокировал линию прямой видимости между передатчиком и приемником. $SB(r, hT, hR, hm, dm) = \dots$, где r – расстояние между передатчиком и приемником, hT – высота передатчика, hR – высота приемника, hm – высота блокирующего объекта, dm – диаметр блокирующего объекта.

2. Предположим, что центры блокирующих объектов расположены на плоскости согласно точечному процессу Пуассона (Poisson Point Process), т.е. случайно и равномерно с фиксированной плотностью λm шт/м. Постройте графики вероятности блокировки линии прямой видимости между передатчиком и приемником в зависимости от расстояния между передатчиком и приемником, положив $hT = 10$ м, $hR = 1.5$ м, $hm = 1.7$ м, $dm = 0.3$ м. Вероятность блокировки линии прямой видимости определяется как обратная вероятность отсутствия блокирующих объектов на фиксированной площади (void probability).

3. Как меняется вероятность блокировки линии прямой видимости при изменении параметров высот передатчика и приемника? На какой высоте вероятность блокировки минимальная? Блокирующие объекты случайной высоты

4. В условиях предыдущей задачи и предположив, что высота блокирующего объекта распределена согласно нормальному распределению с математическим

ожиданием $\mu = 1.7$ м и дисперсией $\sigma^2 = 1$ м, постройте графики вероятности блокировки линии прямой видимости между передатчиком и приемником в зависимости от расстояния между ними. Для случайной высоты блокирующего объекта вероятность блокировки линии прямой видимости определяется как обратная вероятность отсутствия блокирующих объектов на всей рассматриваемой площади с учетом просеивания случайного потока.

5. Сравните полученные результаты с результатами, полученными для блокирующих объектов фиксированной высоты. Сделайте вывод о точности аппроксимации случайной высоты блокирующего объекта фиксированным значением. Оптимизация высоты передатчика

6. В условиях предыдущей задачи предположим, что блокировка прямой видимости приводит к потере в уровне мощности принимаемого сигнала в $b = 20$ дБ. При этом используется модель потерь при распространении радиосигнала FSPL на частоте 28 ГГц с мощностью передатчика 23 дБм и усилениями антенн на передаче и приеме в 10 дБ. Определите оптимальную высоту передатчика, которая максимизирует средний уровень принятого сигнала для нескольких расстояний между передатчиком и приемником $r_1 = 50$ м, $r_2 = 100$ м и $r_3 = 200$ м путем построения графиков зависимости среднего уровня принимаемого сигнала от высоты передатчика. Реалистичны ли полученные результаты?

Листинг программы:

```
ht = 10;
hr = 1.5;
hm = 1.7;
dm = 0.3;
l = 0.1

Sb = lambda r, hm, ht : dm * r * max(0, (hm - hr)) / (ht - hr) + math.pi * dm **
2 / 8;

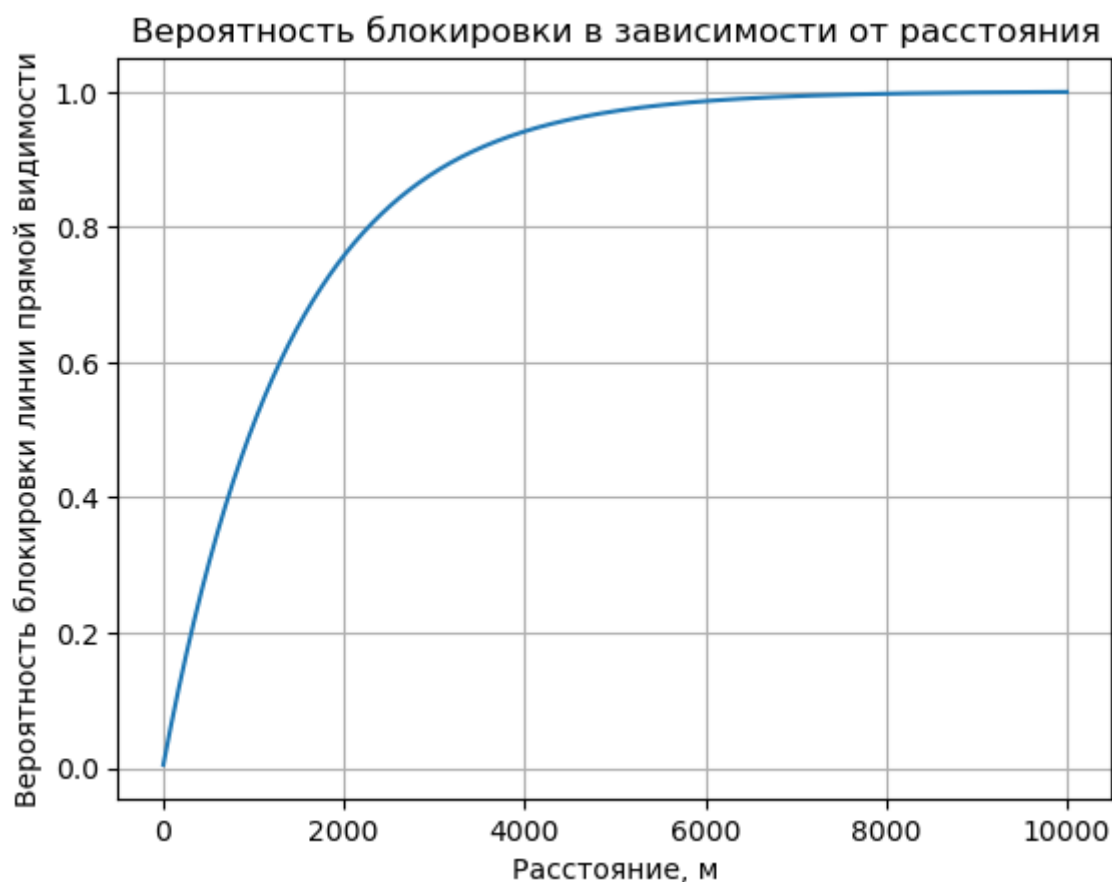
pb = lambda r, hm, ht : 1 - math.exp( - l * Sb(r, hm, ht));
DISTANCE = np.linspace(2, 10000, 100000);
plt.plot(DISTANCE, [pb(i, hm, ht) for i in DISTANCE]);
plt.grid();
plt.title("Вероятность блокировки в зависимости от расстояния");
plt.xlabel("Расстояние, м");
plt.ylabel("Вероятность блокировки линии прямой видимости");
```

```
plt.show();
```

Для начала, т. к. мною не была найдена в лекциях формула площади, в которой должен располагаться центр блокировки для блокировки линии прямой связи, выведено это было самостоятельно.

$$S = d_m \cdot r \cdot \frac{h_m - h_r}{h_t - h_r} + \frac{\pi \cdot d_m^2}{8}$$

После этого по вышеизложенному коду находили вероятности блокировки прямой видимости через данную площадь в зависимости от расстояния между станцией и приёмником:



Зависимость вероятности блокировки от высот станции, блокировки или приёмника следующая: данная зависимость актуальна, если высота блокировки меньше высоты станции и больше высоты приёмника. В условиях, когда высота блокировки меньше высоты приёмника, вероятность блокировки сводится к нулю. При высоте блокировки больше базовой станции, задача сводится к двумерной.

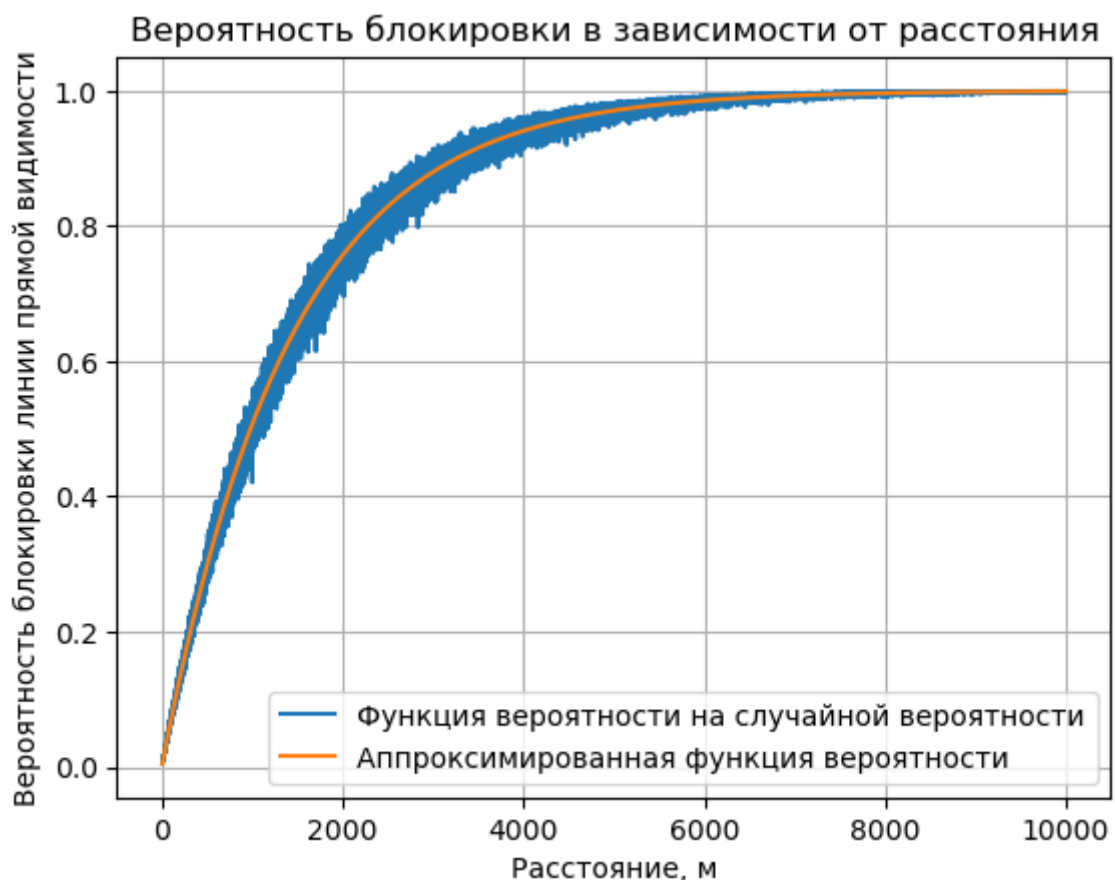
Листинг программы:

```
plt.plot(DISTANCE, [pb(i, np.random.normal(hm, 0.01), ht) for i in DISTANCE],
label = "Функция вероятности на случайной вероятности");

plt.plot(DISTANCE, [pb(i, hm, ht) for i in DISTANCE], label =
"Аппроксимированная функция вероятности");

plt.grid();
plt.legend();
plt.title("Вероятность блокировки в зависимости от расстояния");
plt.xlabel("Расстояние, м");
plt.ylabel("Вероятность блокировки линии прямой видимости");
plt.show();
```

Т.к. необходимые функции уже были введены в первом листинге, здесь остаётся передавать в анонимную функцию случайные величины высот блокировок. Можно справедливо заметить, что я отошёл от условия лабораторной и значение дисперсии взял сильно меньше. Всё из-за того, что при дисперсии около 0.1 высота блокировки становится непредсказуемой и смело зануляет вероятность блокировки даже на больших расстояниях ввиду вычитания из высоты блокировки высоты приёмника.



Причем точность аппроксимации достаточно точно совпадает с

рандомизированными показателями. При увеличении дисперсии график смело закрашивает экран полностью, и говорить о какой-либо аппроксимации бессмысленно.

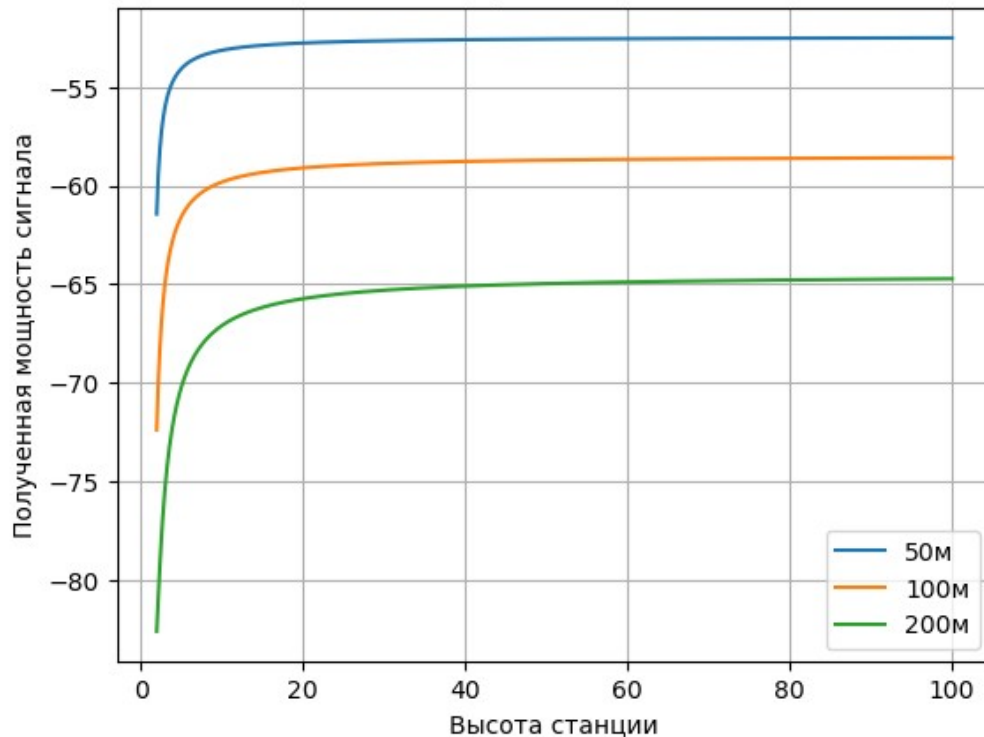
Листинг программы:

```
SPEED_OF_LIGHT = 299_792_458;
PREPARED_CONST = 20 * math.log10(4 * math.pi / SPEED_OF_LIGHT);
heights = np.linspace(2, 100, 1000);
def FSPL(r, f = 28e9):
    return PREPARED_CONST + 20 * math.log10(r * f);
def findAvgByR(r):
    return [23 + 10 + 10 - FSPL(r) - 20 * pb(r, hm, i) for i in heights];
plt.plot(heights, findAvgByR(50), label="50м");
plt.plot(heights, findAvgByR(100), label="100м");
plt.plot(heights, findAvgByR(200), label="200м");
plt.xlabel("Высота станции");
plt.ylabel("Полученная мощность сигнала");
plt.legend();
plt.grid();
plt.title("Полученная мощность сигнала на приёмнике в зависимости от высоты станции");
plt.show();
```

Далее было необходимо найти оптимальную высоту базовой станции. Задача, как оказалось, не сводилась к поиску глобального экстремума, т. к. итоговая мощность всегда росла. При введении, например, затрачиваемых ресурсов, на единицу высоты, можно было бы говорить уже о глобальном экстремуме.

Здесь была введена функция FSPL, модифицированная под наши нужды. Оптимальное значение высоты исследовалось в области от 2 до 100 метров. На выходе программы получаем график итоговой мощности сигнала на приёмнике в зависимости от высоты станции:

Полученная мощность сигнала на приёмнике в зависимости от высоты станции



В данном случае видно, что глобальный экстремум здесь не существует, однако, в вопросе оптимального значения высоты, можно отметить, что высота станции примерно в 35 метров является достаточно оптимальным значением, т. к. далее мы не получаем заметного прироста мощности сигнала.

Выводы:

В данной лабораторной работе исследовали:

- нахождение блокировок в зоне покрытия и на линии прямой видимости с приёмником;
- вероятностные характеристики блокировки.