

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

УТВЕРЖДАЮ

Заведующий кафедрой
прикладной информатики
и теории вероятностей
д.т.н., профессор
К.Е. Самуйлов

«___» _____ 2020г.

КУРСОВАЯ РАБОТА

на тему

«Исследование взаимодействия пользователей с подвижными базовыми станциями при неполных данных об их местоположении»

02.03.01 «Математика и компьютерные науки»

шифр и наименование учебной дисциплины

Выполнил

Студент группы НКНбд-01-17

Студенческий билет №: 1032172734

Папикян.Г.Т

«___» _____ 2020 г.

Руководитель

Заведующий кафедрой
прикладной информатики
и теории вероятностей
д.т.н., профессор
К.Е. Самуйлов

Москва 2020

Оглавление

Введение.....	3
1. Обзор литературы.....	4
Оптимальное расположение базовых станций с использованием предопределенной плотности пользователей.....	4
Предложенная модель.....	4
Оптимальное расположение базовых станций без использования статистических данных о расположении пользователей.....	11
Предложенная модель.....	11
2. Модель взаимодействия дрона и пользователя.....	14
3. Анализ модели.....	17
4. Заключение.....	19
Источники.....	20

Введение

За последние годы в результате развития индустрии беспилотных летательных аппаратов, дроны получили широкое распространение в различных сферах деятельности: от производства мультимедийного контента и агрокультуры, до телекоммуникаций. Как утверждается в [1], к 2023 году глобальные поставки дронов составят 2.4 миллиона долларов, и телекоммуникации будут являться одной из индустрий, обеспечивающих подобный рост.

Время автономной работы, другими словами, емкость батареи дронов является одним из основных факторов, ограничивающих распространение данной технологии. К счастью, данная область также переживает активное развитие; так, в [2] сообщается о стартапе SolidEnergy, обещающей вдвое увеличить длительность полета дронов применением литий-металлических батарей, вместо литий-ионных.

Что же касается индустрии телекоммуникаций, финская компания Nokia вместе с британским оператором мобильной связи EE провела тестирование технологии летающих базовых станций, призванных улучшить радиопокрытие для отдаленных от города селений в Шотландии. Дроны были оснащены миниатюрными базовыми станциями весом в 2 кг. Таким образом, уже сейчас возможно использование дронов в качестве базовых станций.

Цель данной работы: проведение обзора некоторых исследований касательно летающих базовых станций, а также реализация и численный анализ простой модели взаимодействия дрона и пользователя.

1. Обзор литературы.

Оптимальное расположение базовых станций с использованием предопределенной плотности пользователей

Работа [4] фокусируется на проблеме эффективного расположения летающих базовых станций в городской среде. Авторы предлагают жадный алгоритм, располагающий дроны таким образом, чтобы обеспечить покрытие для максимального числа пользователей.

Предложенная модель

В данной модели рассматривается дискретное трехмерное пространство V с измерениями $[X_{min}, X_{max}]$, $[Y_{min}, Y_{max}]$ и $[0, H_{max}]$. Также вводится S - двумерная плоскость с координатами (X, Y) , где $X \in [X_{min}, X_{max}]$, $Y \in [Y_{min}, Y_{max}]$. Фактически, S моделирует землю. Расстояние между двумя точками $p_1 = (x_1, y_1, z_1)$ и $p_2 = (x_2, y_2, z_2)$ определяется функцией $d(p_1, p_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}$. Предполагается, что дроны находятся в городе, и потому на плоскости S вводится функция $h(x, y)$, значением которой является высота здания в этой точке x, y . В случае, если $h(x, y)$ равен нулю, считается, что здания в данной точке нет. Дроны, разумеется, должны облетать здания, но кроме этого, они должны находиться на безопасном расстоянии от них. Пусть дрон не может подлететь к зданию ближе чем на H_s .

Тогда

$$U_i = \{ p \mid d(P_i, p) < H_s, p \in V \} \quad (1.1)$$

где P_i - позиция дрона i . Полученный шар U_i является безопасной зоной вокруг дрона, и тогда можно сформулировать следующее условие

$$h(X, Y) \leq Z, \quad \forall (X, Y, Z) \in U_i \quad (1.2)$$

Всякая позиция дрона, удовлетворяющая условию (1.2) будет являться безопасной.

Хотя данная работа и не ставит своей целью моделирование энергопотребления дронов, судя по всему просто для полноты картины вводятся некоторые простые ограничения.

Сначала необходимо определить плоскость $G_i = \{ p \mid h(p) < z_i - H_s, p \in S \}$ для дрона i . G_i параллельна плоскости S и представляет собой множество точек, доступных дрону без изменения высоты. Считается, что станции перезарядки расположены на высоте 0 и Q_i - их позиции.

Дрон может либо обслуживать пользователей, паря в определенной точке пространства, либо двигаться между двумя точками пространства: позицией для обслуживания и ближайшей станцией перезарядки. При этом данный путь разделяется на два участка. Первый - когда дрон двигается в плоскости G_i до тех пор, пока не окажется над станцией перезарядки, и второй - когда меняет высоту, уже находясь над ней. Первую часть пути обозначим как $g(P_i, Q_j)$, это будет являться наикратчайшим путем от P_i до Q_j при в проекции плоскости G_i на S . Авторы не рассматривают энергию, потраченную на беспроводную передачу, и фокусируются только на физических перемещениях дрона.

Таблица 1.1 Используемые обозначения [4]

Обозначение	Описание
V	Трехмерная дискретная модель рассматриваемого пр-ва.
λ	Частота дискретизации измерений x и y пр-ва V .
λ_z	Частота дискретизации измерения z пр-ва V .
S	Плоскость в основании пр-ва V . Моделирует поверхность земли.
$p = (X, Y)$ или (X, Y, Z)	Координаты $p \in S$ или $p \in V$.
$h(p)$	Высота здания в позиции $p \in S$.
$\rho(p)$	Плотность пользователей в позиции $p \in S$.
D	Множество дронов.
n	Количество дронов.
$P_i = (x_i, y_i, z_i)$	Трехмерные координаты дрона i .
H_s	Безопасная дистанция. Т.е. минимальная дистанция между дроном и зданием.
$d(p_1, p_2)$	Евклидова дистанция между $p_1 \in V$ и $p_2 \in V$.
Q_j	Позиция станции перезарядки j .
G_j	Плоскость, которую дрон использует для достижения проекции станции перезарядки. То есть, данную плоскость дрон использует, чтобы оказаться над станцией перезарядки перед тем, как начнет приземление.
$g(P_i, Q_j)$	Длина наикратчайшей пути между P_i и проекцией Q_j на плоскости G_j .
$p_s(z_i)$	Энергия, затрачиваемая дроном в единицу времени, на парение в позиции z_i .
v_i	Скорость перемещения дрона i .
$p_f(v_i)$	Энергия, затрачиваемая дроном i в единицу времени, при движении со скоростью v_i .
t_s	Общее время парения (обслуживания).
$L(P_i, p)$	Path Loss между дроном в позиции $P_i \in V$ и пользователем в позиции $p \in S$.
$SINR(p)$	$SINR$ пользователя в позиции $p \in S$.
N	Максимальное число пользователей, которых может обслуживать один дрон.
$C(P_i)$	Множество позиций, покрываемых дроном i в позиции $P_i \in V$.
$C(P_1, P_2, \dots, P_n)$	Множество позиций, покрываемых всеми дронами.

Итак пусть $p_s(z_i)$ - энергия, потраченная в единицу времени при парении на высоте z_i , а $p_f(v_i)$ на полет со скоростью v_i . При этом, затраты на ускорение не рассматриваются; t_s - время, которое дроны проводят в определенной точке при обслуживании пользователей. Пусть E - доступный уровень энергии, а α - константа, определяющая уровень энергии, которую позволено потратить ($0 < \alpha < 100\%$). Тогда получается, что

$$p_s(z_i) t_s + p_f(v_i) \frac{g(P_i, Q_j) + z_i}{v_i} \leq \alpha E \quad (1.3)$$

где P_i - позиция дрона i , а Q_j - позиция станции перезарядки j .

Считается, что если высота здания в позиции пользователя равна нулю, что данный пользователь находится снаружи. В таком случае между ним и дроном может быть LoS, или NLoS. Авторы используют реалистичную модель 3GPP для path loss'a:

$$L(P_i, p) = \begin{cases} A^{LoS} + B^{LoS} \log(d(P_i, p)), & \text{при LoS} \\ A^{NLoS} + B^{NLoS} \log(d(P_i, p)), & \text{при NLoS} \end{cases}$$

где $A^{LoS}, B^{LoS}, A^{NLoS}, B^{NLoS}$ – заданные константы

Также считается, что пользователь в позиции $p \in S$ находится в зоне покрытия дрона в позиции P_i , если

$$SINR(p) = \frac{r_i(P_t - L(P_i, p))}{\sum_{j \in D \setminus i} r_j(P_t - L(P_j, p)) + N_0} \geq \beta \quad (1.4)$$

Здесь P_t – мощность переданного сигнала, одинаковая для всех дронов,

$P_t - L(P_i, p)$ – мощность принятого пользователем сигнала,

$r_i \sim \exp(i)$, моделирует Рэлеевское затухание сигнала,

$D \setminus i$ – дроны, не обслуживающие пользователя,

N_0 – мощность шума,

β – заданная константа.

Также вводится ограничение на количество обслуживаемых пользователей одним дроном, поскольку качество обслуживания зависит также от пропускной способности летающей базовой станции.

Пусть N – максимальное число пользователей, которых дрон может обслуживать. Тогда формулируется следующее ограничение:

$$\sum_{p \in C(P_i)} \Phi_p \rho(p) dp < N, \quad \forall i \in D \quad (1.5)$$

где $\rho(p)$ – плотность пользователей в позиции p ,

Φ_p – точечный Пуассоновский процесс, моделирующий уровень трафика в точке p ,

$C(P_i)$ – мн-во точек плоскости S , которые покрываются дроном, находящимся в позиции P_i .

Теперь, пусть $C(P_1, P_2, \dots, P_n) = \bigcup_{i \in D} C(P_i)$ – это мн-во точек на S , обслуживаемых дронами в позициях P_1, P_2, \dots, P_n .

Тогда число всех обслуживаемых пользоателей можно получить следующим образом:

$$f(P_1, P_2, \dots, P_n) = \sum_{p \in C(P_1, P_2, \dots, P_n)} \Phi_p \rho(p) dp$$

В результате всех этих рассуждений, далее в работе формулируется основная оптимизиционная задача.

Формулировка задачи:

$$\max_{P_1, P_2, \dots, P_n} \sum_{p \in C(P_1, P_2, \dots, P_n)} \Phi_p \rho(p) dp$$

т.е необходимо подобрать такие значения позиций P_i , чтобы сумма была максимальной.

Разумеется, также должны быть удовлетворены ранее описанные условия:

$$\begin{aligned}
h(X, Y) &\leq Z, \forall (X, Y, Z) \in U_i, \forall i \in D \\
U_i &= \{ p \mid d(P_i, p) < H_s, p \in V \}, \forall i \in D \\
p_s(z_i) t_s + p_f(v_i) \frac{g(P_i, Q_j) + z_i}{v_i} &\leq \alpha E, \forall i \in D \\
\sum_{p \in C(P_i)} \Phi_p \rho(p) dp &< N, \forall i \in D \\
x_i &\in [X_{min}, X_{max}], \forall i \in D \\
y_i &\in [Y_{min}, Y_{max}], \forall i \in D \\
z_i &\in [H_{min}, H_{max}], \forall i \in D
\end{aligned}$$

Далее для проведения анализа алгоритмической сложности, задача упрощается и сводится к известному случаю.

Рассмотрим введенные упрощения:

- убирается условие наличия зданий, что обеспечивает LoS во всех случаях;
- предполагается, что дроны находятся на одной высоте, что превращает задачу из трехмерной в двухмерную;
- предполагается, что все дроны обслуживают пользователей на разной частоте, что позволяет использовать SNR вместо SNIR, т.к. интерференция отсутствует. В действительности, поскольку мы уже рассматриваем случай с LoS, данное условие приводит к тому, что с каждым дроном можно ассоциировать окружность на плоскости S , и рассматривать пользователей внутри нее, как обслуживаемых
- предполагается, что у дронов достаточно заряда на все время полета
- предполагается, что у дронов достаточная пропускная способность

Таким образом, убираются почти все ограничения. Упрощенная задача формулируется так: даны плоскость S , плотность пользователей $\rho(p)$, радиус покрытия дронов.

Найти двумерные позиции для n дронов, которые максимизировали бы число обслуживаемых пользователей.

Можно заметить, что получается задача о покрытии множества, которая является NP-сложной, и это только для упрощенного случая, следовательно, изначальная задача будет не проще.

Рассмотрим предложенный алгоритм.

Алгоритм 1.1

Ввод: \mathcal{V} , $\rho(p)$, N , n

Вывод: \mathcal{P}

1: $\mathcal{P} \leftarrow \emptyset, \mathcal{C} \leftarrow \emptyset, i \leftarrow 1$.

2: **while** $i \leq n$ **do**

3: **if** \mathcal{V} is empty **then**

4: Terminate

5: **end if**

6: **for** $p \in \mathcal{V}$ **do**

7: Составить множество пользователей, покрываемых дроном в позиции p (т.е. составить $C(p)$)

8: Добавить интерференцию от дрона в позиции p к пользователям в \mathcal{C} .
Удалить из \mathcal{C} тех пользователей, чей обновленный SNIR меньше чем

β и обозначить обновленное множество как \bar{C} . Обновить множество пользователей, покрываемых каждым дроном и обозначить их как $C(P_k, p), \forall k < i$

```

9:   if  $|C|(P_k, p) \leq N, \forall k \leq i$  then
10:       $C(p) \leftarrow \bar{C} \cup C(p).$ 
11:   else
12:       $C(p) \leftarrow \emptyset.$ 
13:   end if
14:    $f(p) \leftarrow \sum_{q \in C(p)} \rho(q)$ 
15: end for
16: Найти  $p$  такой, что  $f(p)$  максимальный.
17:  $\mathcal{P} \leftarrow \mathcal{P} \cup p; \mathcal{C} \leftarrow \mathcal{C}(p); C(P_k) \leftarrow C(P_k, p), \forall k \leq i; \mathcal{V} \leftarrow \mathcal{V} \setminus p; i \leftarrow i+1.$ 
18: end while

```

Идея алгоритма заключается в том, чтобы по очереди рассматривать все дроны, и для каждого из них находить позицию, максимизирующую количество покрываемых пользователей. Данный алгоритм является жадным, поскольку он каждый раз рассматривает позиционирование одного дрона. В противном случае, можно было бы рассматривать все дроны одновременно, но в таком случае решение превратилось бы в полный перебор позиций дронов, что было бы крайне неэффективно. В данном же случае получается решение, хоть и не оптимальное, но близкое к оптимальному.

Рассмотрим обозначения: $\mathcal{V} \subset V$ – набор точек, в которых потенциально может находиться дрон. Фактически, это просто все точки, которые соответствуют ранее определенным ограничениям; $\rho(p)$ – плотность пользователей в точке p ; N – уже рассмотренное ограничение на количество пользователей, которые может обслуживать один дрон; n – количество дронов; $C(p)$ – позиции пользователей, которых покрывает дрон в позиции p ; $C(P_k, p)$ – позиции пользователей, которых покрывает дрон k в позиции P_k при добавлении другого дрона в позиции p ;

\mathcal{C} – позиции всех покрываемых пользователей. \bar{C} – обновленное множество \mathcal{C} , с из которого удалены пользователи, SINR которых стал меньше, чем β после добавления очередного дрона. И наконец, $C(p)$ – покрываемые пользователи при добавлении дрона в позиции p , а $f(p)$ – покрываемые пользователи при добавлении дрона в позиции p с учетом плотности пользователей $\rho(p)$.

Как работает алгоритм: Первый цикл, **while**, перебирает все дроны. Далее выбранный дрон размещается во всех доступных позициях $p \in \mathcal{V}$, и из числа этих позиций выбирается такая, которая максимизирует число покрываемых пользователей \mathcal{C} (здесь видно, что алгоритм жадный). Для того, чтобы это сделать, для каждой точки из \mathcal{V} в цикле **for** считается количество покрываемых пользователей, если бы рассматриваемый дрон был добавлен в эту точку. Для этого в строках 7-8 обновляется множество уже обслуживаемых пользователей: оттуда удаляются те, SINR которых стал неприемлимым под влиянием интерференции рассматриваемого дрона. Также там создается множество пользователей, которых покрывал бы рассматриваемый дрон, будь он добавлен в рассматриваемую точку, а также пересчитываются множества покрываемых пользователей для уже размещенных дронов (поскольку некоторые пользователи были удалены из числа покрываемых). Далее, в строке 9 нужно проверить, что в новой конфигурации, с учетом добавленного дрона, ни для

какого дрона максимальный порог N обслуживаемых пользователей не будет превышен. Если будет превышен, то данная конфигурация не рассматривается, и множество покрываемых пользователей при расположении рассматриваемого дрона в данной точке является пустым, алгоритм переходит к рассмотрению следующей точки. Если же это условие выполнено для всех дронов, то можно рассматривать дальше, и $C(p)$ устанавливается равным сумме множеств пользователей, которых покрывает рассматриваемый дрон и тех, которых покрывают все остальные дроны. Далее составляется $f(p)$ – то же самое, что и $C(p)$, но с учетом плотности расположения пользователей (Это необходимо, поскольку пространство дискретное). После выполнения цикла `for`, находится наибольший $f(p)$, и соответствующая ей точка p . На самом деле, предполагается, что это будет происходить во время выполнения цикла `for`.

Далее, когда точка найдена, она добавляется в \mathcal{P} , удаляется из \mathcal{V} , и обновляются переменные, содержащие мн-ва обслуживаемых пользователей. После этого алгоритм переходит к рассмотрению следующего дрона. Таким образом, алгоритм находит оптимальное решение подзадачи на каждом шаге, и предполагается, что вся задача в результате тоже будет иметь оптимальное решение. В случае с некоторыми классическими жадными алгоритмами это так, но в данном случае, получается суб-оптимальное, то есть приближенное к оптимальному, решение, что тоже может являться хорошим результатом, учитывая сложность задачи.

В статье был проведен анализ временной сложности данного алгоритма, она равняется $O\left(\frac{n^2}{\lambda^4 \lambda_z}\right)$, где λ – частота дискретизации измерений x и y ,
 λ_z – частота дискретизации измерения z ,
 n – количество дронов.

Для анализа предложенной модели авторы вводят метрики:

$$1) \text{ Covered Ratio (CR): } CR = \frac{\sum_{p \in C(P_1, P_2, \dots, P_n)} \Phi_p \rho(p)}{\sum_{p \in S} \Phi_p \rho(p)} - \text{отношение числа покрываемых}$$

пользователей к числу всех пользователей в рассматриваемой местности.

$$2) \text{ Spectral Efficiency (SE): } SE(p) = \log_2(1 + SINR(p)) - \text{мера эффективности}$$

использования ограниченного спектра частот протоколом физического уровня для пользователя, расположенного в позиции $p \in S$.

$$3) \text{ Average Spectral Efficiency: } ASE = \frac{\sum_{p \in C(P_1, P_2, \dots, P_n)} \Phi_p \rho(p) SE(p)}{\sum_{p \in C(P_1, P_2, \dots, P_n)} \Phi_p \rho(p)}.$$

Симуляция проводилась в городской местности размером 1000м на 1000м. Здания моделировались кубами шириной в 60 м и высотой от 10м до 60м, что соответствует зданиями от 3х этажей до 20ти. Ширина улиц между зданиями считалась равной 20м. Высота дронов ограничивалась 50 и 300 метрами. Плотность пользователей генерировалась случайно, между значениями 0 и 5. В статье также говорится, что после генерации плотности пользователей, в некоторых участках она изменялась вручную, чтобы обеспечить области с намного более высокой плотностью. Внутри зданий плотность пользователей, разумеется, считается равной нулю. Суммарное количество пользователей в рассматриваемой области считается равной 3600.

Было проведено сравнение результатов предложенного алгоритма при разных частотах дискретизации, согласно параметрам таблицы 1.2. В результате, при увеличении частот дискретизации CR уменьшается, как показано на рисунке 1.1.

Также было проведено сравнение предложенного алгоритма с алгоритмом полного перебора. Максимальное количество дронов было ограничено пятью, так как при полном переборе авторы не смогли получить результаты за приемлемое время при больших значениях. Частоты дискретизации λ и λ_z , были соответственно равны 50 и 30. Полученный результат показал, что при данных значениях результаты обоих алгоритмов отличаются несущественно, как показано на рисунке 1.2.

Важную роль играют также параметры n и β . Так, например, при увеличении n , CR увеличивается, а ASE уменьшается. Так происходит по той причине, что при увеличении дронов возможно покрыть большее количество пользователей, но вместе с этим увеличивается и взаимная интерференция между дронами. И таким образом, при уменьшении SINR пользователей, ASE также уменьшается. Данные рассуждения отображены на рисунках 1.3 и 1.4.

Также, можно заметить, что при $\beta = 20$ невозможно покрыть всех пользователей.

Таблица 1.2 [4] Параметры, использованные при проведении симуляций

$A^{LoS} = 103.8$	$P_t = 30 \text{ dBm}$	$t_s = 0.5 \text{ h}$
$A^{NLoS} = 145.4$	$\alpha = 98 \%$	$H_s = 10 \text{ m}$
$B^{LoS} = 20.9$	$\beta = 15 \text{ dB}$	$p_s = 140 \text{ W}$
$B^{NLoS} = 37.5$	$E = 89.2 \text{ Wh}^3$	$N = 500$
$N_0 = -91 \text{ dBm}$	$v = 6 \text{ m/s}$	$p_f = 143 \text{ W}$
$\lambda = \text{om } 10 \text{ до } 100$	$\lambda_z = 10, 20, 30$	$n = 10$

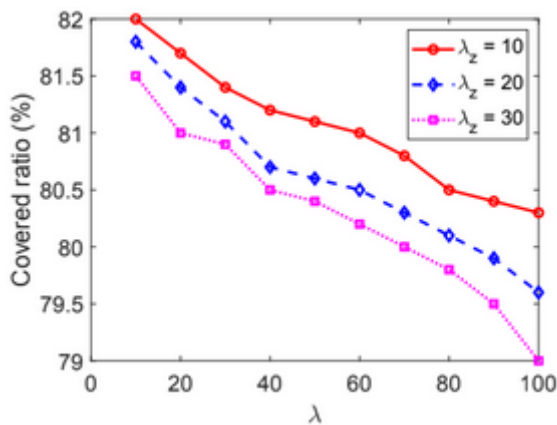


Рисунок 1.1 [4] График CR при различной частоте дискретизации

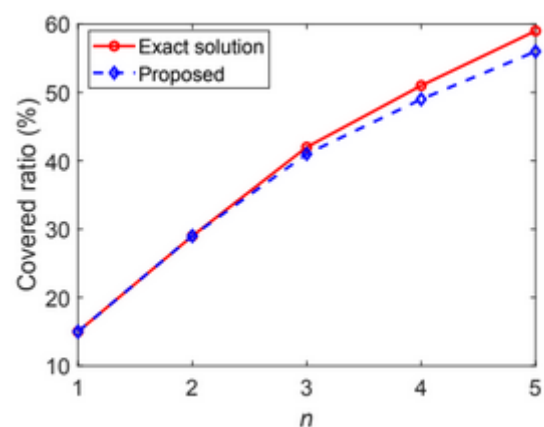


Рисунок 1.2 [4] Сравнение CR для алгоритма полного перебора, и алгоритма 1.1

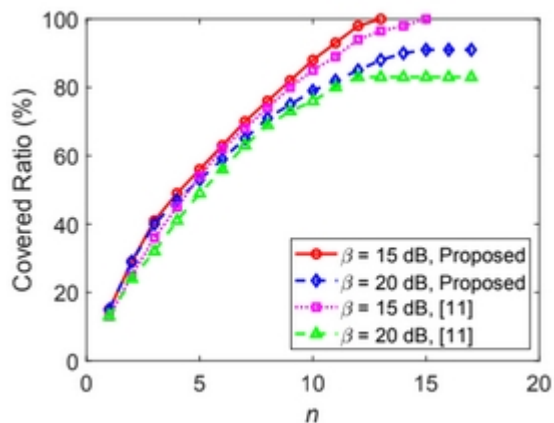


Рисунок 1.3 [4] График CR при различных значениях порога β

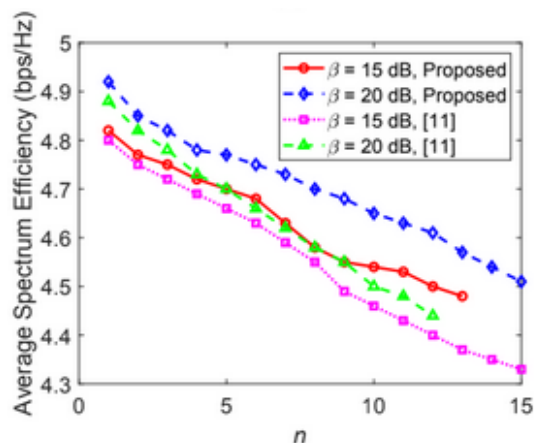


Рисунок 1.4 [4] График ASE при различных значениях параметра β

Оптимальное расположение базовых станций без использования статистических данных о расположении пользователей

Теперь рассмотрим работу [5]. Авторы данной работы также рассматривают проблему расположения летающих базовых станций с целью максимизации обслуживаемых пользователей. Основные отличия от предыдущей работы следующие: модель, предложенная в этой работе, не опирается на изначальные данные о плотности пользователей, более того, здесь рассматривается способ получения расположения пользователей при помощи сканирования местности; рассматривается только двумерное расположение дронов; рассматриваемая местность не обязательно является городской. Также предполагается, что пользователи малоподвижны, или вовсе стационарны. Предложенное решение могло бы использоваться при обеспечении связи во время чрезвычайных происшествий. Здесь, фактически, решаются две задачи: определение расположения пользователей при помощи сканирования местности, оптимальное расположение дронов для обеспечения максимального покрытия.

Предложенная модель.

Рассматривается территория, на которой располагаются n пользователей, о расположении которых нет изначальной информации. Множество пользователей обозначается, как $U = \{U_1, U_2, \dots, U_n\}$. Рассматриваемая область является выпуклым многоугольником с площадью A_p . Для локализации пользователей используется метод Uplink Time Difference of Arrival (UTDOA). Данный метод вычисляет расположение пользователя через разницу во времени между несколькими uplink сигналами, посланными пользователем нескольким базовым станциям. Данный метод предполагает использование как минимум трех базовых станций для получения времени отправленных сигналов. Также данный метод работает с погрешностью в 50 метров в 50% случаев, как говорится в статье, следовательно позиция пользователя в действительности будет являться окружностью с радиусом r_e , а не конкретной точкой.

Группа дронов разделяется на f групп, в каждой группе по три дрона. Все три дрона D1, D2, D3 будут находиться на одной высоте, образуя равносторонний

треугольник со стороной d . Области покрытия дронов будут пересекаться, и у области пересечения будет радиус R_c .

Вводится следующая формула для вычисления площади пересечения областей покрытия трех дронов, A_0 :

$$A_0(d, R_c) = R_c^2(3\alpha - \frac{\pi}{2}) - \frac{3d}{2}\sqrt{R_c^2 - \frac{d^2}{4}} + \frac{\sqrt{3}}{4}d^2, \quad \alpha = \arccos\left(\frac{d}{2R_c}\right) \quad (1.6)$$

Считается, что $A_p \gg f A_0$.

В статье также обращают внимание на то, что уменьшение d приводит к увеличению области A_0 , что положительно сказывается на времени обнаружения пользователей, но поскольку в этом случае дроны окажутся ближе друг к другу, то разница во времени будет меньше, что приведет к большей погрешности. По этой причине, говорят авторы, необходимо с осторожностью подбирать значение d .

Для сканирования рассматриваемой области, она разделяется на под-области, и каждой под-области присваивается одна группа дронов. Далее каждая группа проводит полный обход своей территории и при выявлении пользователей отправляет данные на контрольный центр, который может представлять из себя полноценную наземную базовую станцию. Обход будет производиться зиг-заг'ами, поэтому необходимо также определить оптимальное направление обхода, чтобы уменьшить количество поворотов.

Рассмотрим используемый алгоритм декомпозиции:

Алгоритм 1.2

Ввод: $P, s, \{p1, p2, \dots, ps\}$

Вывод: $DIR, \{S1, S2, \dots, S3\}$

1: Найти функцию диаметра рассматриваемого многоугольника.

2: Найти оптимальное направление обхода, которое будет перпендикулярным направлению наименьшего диаметра.

3: Разделить многогранник на s частей по помощи линий, параллельных оптимальному направлению обхода, учитывая соотношения $\{p1, p2, \dots, ps\}$.

Здесь P – вершины рассматриваемой области-многогранника,

s – количество под-областей, на которые необходимо разделить многогранник P ,

$\{p1, p2, \dots, ps\}$ – соотношения, согласно которым нужно произвести разбиение,

DIR – оптимальное направление обхода,

$\{S1, S2, \dots, S3\}$ – полученные под-области.

Обозначим $C = \{C_1, C_2, \dots, C_n\}$ все окружности, полученные при локализации пользователя, являющиеся их приблизительными местонахождениями. Тогда пусть $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ являются центрами окружностей C .

После нахождения приблизительных позиций пользователей, дроны перестают считаться сгруппированными по три. У каждого дрона есть область покрытия, проекция которой на поверхность земли будет являться окружностью с радиусом R_c .

Обозначим центры этих окружностей $\{(x_D^1, y_D^1), (x_D^2, y_D^2), \dots, (x_D^n, y_D^n)\}$. Пусть $u = \{u_1, u_2, \dots, u_n\}$, $u_i \in \{1, 0\}$, $i \in C$, и пусть $u_i = 1$, когда приблизительная позиция пользователя (диск с радиусом r_e) полностью покрыт дроном с позицией (x_D^i, y_D^i) , и $u_i = 0$ в противном случае.

C_i полностью покрыт, если его центр расположен не дальше, чем $(R_c - r_e)$ от центра окружности области покрытия дрона i : (x_D^i, y_D^i) .

Получается следующее ограничение:

$$u_i((x_i - x_D^i)^2 + (y_i - y_D^i)^2) \leq (R_c - r_e)^2 \quad (1.7)$$

Авторы предлагают переписать (2) в следующем виде, чтобы убедиться, что оно выполняется при $u_i = 0$:

$$(x_i - x_D^i)^2 + (y_i - y_D^i)^2 \leq (R_c - r_e)^2 + P(1 - u_i) \quad (1.8)$$

Здесь P – большая константа, удовлетворяющая уравнению при $u_i = 0$. А при $u_i = 1$ уравнение (1.8) сводится к (1.7).

Формулировка оптимизационной задачи:

$$\max_{x_D^j, y_D^j, u_i} \sum_{i \in C} u_i, \quad (1.9)$$

$$\text{при } (x_i - x_D^i)^2 + (y_i - y_D^i)^2 \leq (R_c - r_e)^2 + P(1 - u_i), \forall i \in C$$

Задачу (1.9) придется решить для всех дронов j . Таким образом, дроны будут позиционироваться поочередно. Так, если для одного дрона из множества всех дронов решается (1.9), то в результате получается множество $C^{cov} \subseteq C$ приблизительных позиций пользователей, которых этот дрон покрыл. И для следующего дрона C будет равняться $C = C - C^{cov}$, и так далее для всех оставшихся дронов, как показано в алгоритме 1.3.

Алгоритм 1.3

Ввод: $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$, R_c , r_e

Вывод: $\{(x_D^1, y_D^1), (x_D^2, y_D^2), \dots, (x_D^m, y_D^m)\}$

1: **for** $j=1:m$

2: Получить (x_D^j, y_D^j) и C^{cov} решением (2.4)

3: $C = C - C^{cov}$

4: **end**

2. Модель взаимодействия дрона и пользователя

Рассмотрим декартовое трехмерное пространство, в котором располагаются дрон D и пользователь U. Пусть Дрон имеет фиксированные координаты. Координаты пользователя x, y будут являться случайными величинами, распределенными равномерно: $x = \xi, y = \eta \sim \text{unif}(a,b)$. Описанная система изображена на рисунке 2.1.

Поскольку между дроном и пользователем будет LoS, нас интересует случайное расстояние d . Но посмотрим для начала на r .

Функция распределения (ФР) для r может быть выражена следующим образом:

$$F_r(r) = P\{\sqrt{\xi^2 + \eta^2} < r\} = \iint_{\sqrt{x^2 + y^2} < r} f_\xi(x) f_\eta(y) dx dy = \int_{-\infty}^{\infty} f_\xi(x) \int_{-\sqrt{r^2 - x^2}}^{\sqrt{r^2 - x^2}} f_\eta(y) dy dx \quad (2.1)$$

где $f_\xi(x) = f_\eta(y) = \frac{1}{a+b}$, - функции плотности вероятности для координат пользователя.

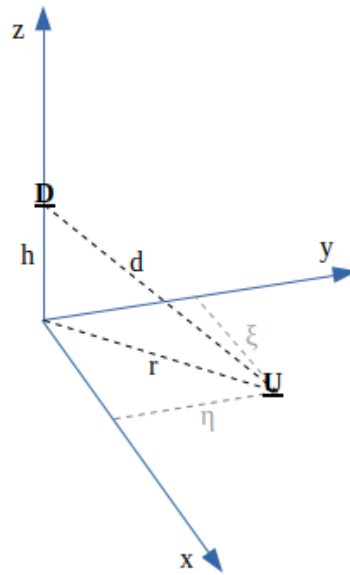


Рисунок 2.1. Изображение рассматриваемой системы

Функция распределения для d может быть выражена следующим образом:

$$\begin{aligned} F_d(d) &= P\{\sqrt{(h^2 + x^2 + y^2)} < d\} = \\ &= \iint_{\sqrt{(h^2 + x^2 + y^2)} < d} f_\xi(x) f_\eta(y) dx dy = \int_{-\infty}^{\infty} f_\xi(x) \int_{-\sqrt{(d^2 - x^2 - h^2)}}^{\sqrt{(d^2 - x^2 - h^2)}} f_\eta(y) dy dx \end{aligned} \quad (2.2)$$

где h - высота дрона.

Заметим, что F_d можно выразить через F_r : $F_d(d) = \begin{cases} F_r(\sqrt{d^2 - h^2}), d > h \\ 0, d < h \end{cases}$

Будем считать, что пользователь передает постоянный поток данных на приемник, расположенный на дроне. Мощность передающей антенны на устройстве пользователя обозначим за P_t , причем передача данных ведется на частоте ν .

Поскольку мы рассматриваем ситуацию, когда дрон находится в области LOS пользователя, будем использовать формулу для расчета Free Space Path Loss (FSPL) из источника [6]:

$$FSPL(d) = 20 \log_{10} \left(\frac{4 \pi d}{\lambda} \right) \quad (2.3)$$

где d - расстояние между антеннами, λ - длина волны.

Тогда согласно [7],

$$SNR(d) = P_t - FSPL(d) - N \quad (2.4)$$

где d - расстояние между антеннами, P_t - мощность переданного сигнала в db, N - шум в db.

Тогда F_{snr} можно выразить следующим образом:

$$\begin{aligned} F_{snr}(z) &= P\{SNR \leq z\} = \\ &= P\{P_t - N - 20 \log_{10} \left(\frac{4 \pi d}{\lambda} \right) \leq z\} = P\left\{d \geq \frac{\lambda}{4 \pi} 10^{\frac{P_t - N - z}{20}}\right\} = \\ &= 1 - P\left\{d < \frac{\lambda}{4 \pi} 10^{\frac{P_t - N - z}{20}}\right\} = 1 - F_d\left(\frac{\lambda}{4 \pi} 10^{\frac{P_t - N - z}{20}}\right) \end{aligned} \quad (2.5)$$

где d - расстояние между антеннами, P_t - мощность переданного сигнала в db, N - шум в db, λ - длина волны.

Данную модель можно также рассмотреть в случае с радиальной системой координат. В таком случае координаты пользователя будут определяться углом ϕ радиусом

$$r, \text{ распределенными равномерно: } f_{\phi}(\phi) = \frac{1}{2\pi}, f_r(r) = \frac{1}{r_{max}}$$

Тогда ФР для r и d примут следующий вид:

$$F_r(X_r) = \int \int_{(r < X_r)} f_r f_{\phi} d\phi dr = \int_0^{X_r} f_{\phi} \int_0^r f_r dr d\phi = \frac{1}{(2\pi)} \frac{1}{R} 2\pi X_r \quad (2.6)$$

$$F_d(X_d) = F_r(\sqrt{(X_d^2 - h^2)}) \quad (2.7)$$

Можно также ввести зависимость координат пользователя от времени. Тогда, ФР дистанции от дрона d будет изменяться в зависимости от прошедшего времени. Т.е, область, в которой пользователь потенциально может находиться, будет увеличиваться с течением времени.

В таком случае, получим

$$f_{\xi}(x, t) = \begin{cases} \frac{1}{(a+b) \frac{t}{t_{max}}}, & x \in \left[\frac{t}{t_{max}} a, \frac{t}{t_{max}} b \right] \\ 0, & x \notin \left[\frac{t}{t_{max}} a, \frac{t}{t_{max}} b \right] \end{cases} \quad (2.8)$$

$$f_{\eta}(y,t) = \begin{cases} \frac{1}{(a+b) \frac{t}{t_{max}}}, y \in [\frac{t}{t_{max}}a, \frac{t}{t_{max}}b] \\ 0, y \notin [\frac{t}{t_{max}}a, \frac{t}{t_{max}}b] \end{cases} \quad (2.9)$$

где $t_{max} = \frac{y_{max}}{v}$ в случае с (2.9), и $t_{max} = \frac{x_{max}}{v}$ в случае с (2.8) - предельное время, т.е время, за которое пользователь потенциально сможет достичь до границы рассматриваемой области, v - скорость пользователя, x_{max} и y_{max} - максимальные координаты пользователя.

Соответственно, ФР для r и d изменятся следующим образом:

$$F_r(r,t) = \int_{-\infty}^{\infty} f_{\xi}(x,t) \int_{-\sqrt{(r^2-x^2)}}^{\sqrt{(r^2-x^2)}} f_{\eta}(y,t) dy dx \quad (2.10)$$

$$F_d(d,t) = \begin{cases} F_r(\sqrt{d^2-h^2}, t), d > h \\ 0, d < h \end{cases} \quad (2.11)$$

3. Анализ модели

Графики F_d и F_{snr} для модели с декартовой системой координат построены на рисунках 3.1, 3.2, а для радиальной – на 3.3, 3.4. При этом для случая декартовой системы в качестве были выбраны следующие начальные данные: пользователь перемещается в диапазоне от $a = -100$ до $b = 100$ метров от положения дрона; сигнал передается на частоте $\nu = 7,5$ МГц, мощность передачи сигнала $P_t = 20$ дБм, уровень шума взят $N = -100$ дБм. На рисунке 3.1 рассмотрены три случая, когда дрон находится на высоте 25, 30 и 100 метров, на рисунке 3.2 приведен график для первого из этих трех случаев.

Для модели радиальной системой координат пользователь передвигается в круге радиуса $R = 100$, остальные параметры совпадают со случаем декартовой системы.

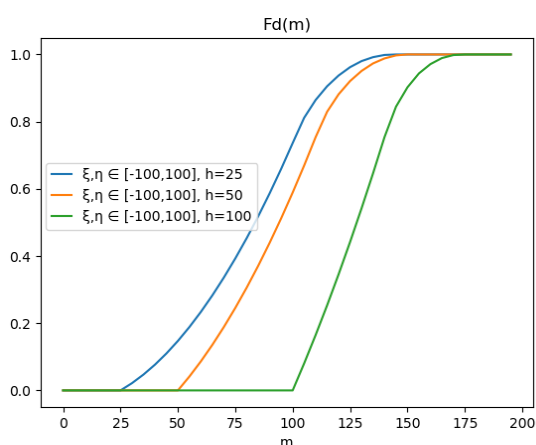


Рисунок 3.1 ФР для дистанции d в модели с декартовой системой координат

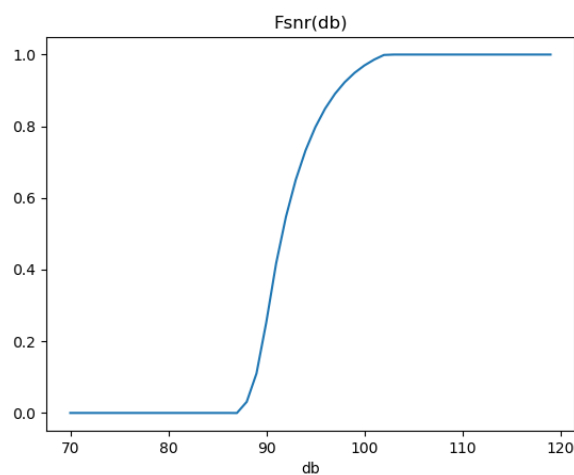


Рисунок 3.2 ФР для SNR в модели с декартовой системой координат

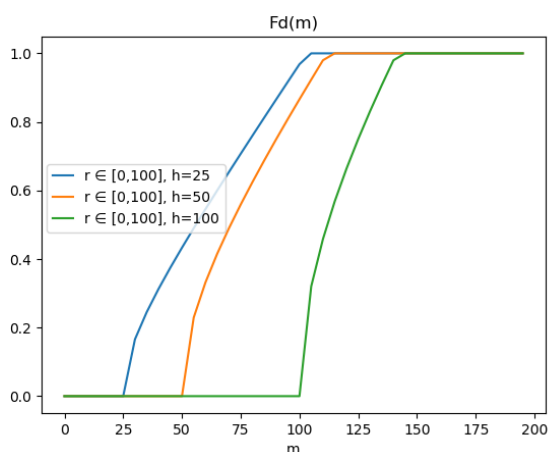


Рисунок 3.3 ФР для дистанции d в модели с радиальной системой координат
 $R = [0, 100]; h = 25, 30, 100; \lambda = 40; N = -100;$
 $P_t = 20;$

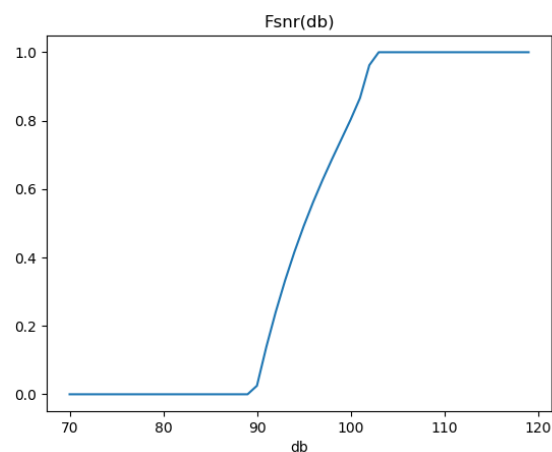


Рисунок 3.4 ФР для SNR в модели с радиальной системой координат
 $a = -100; b = 100; h = 25; \lambda = 40;$
 $N = -100; P_t = 20;$

На рисунках 3.1, 3.3 можно видеть, что в зависимости от высоты дрона, дистанция до дрона лежит либо в интервале [25,130], либо [50,150], либо [100,170]. Причем минимальная дистанция определяется высотой h , так как это тот случай, когда пользователь находится непосредственно под дроном. Графики на рисунках 3.2 и 3.4 показывают, что snr находится в интервале [85,100].

Для модели с декартовой системой координат и дополнительным параметром t , был построен график зависимости вероятности $P\{\text{SNR}(t) < \text{SNR}^*\}$ от t и SNR^* (Рисунок 3.5). Здесь SNR^* является порогом, который определяет допустимое качество соединения.

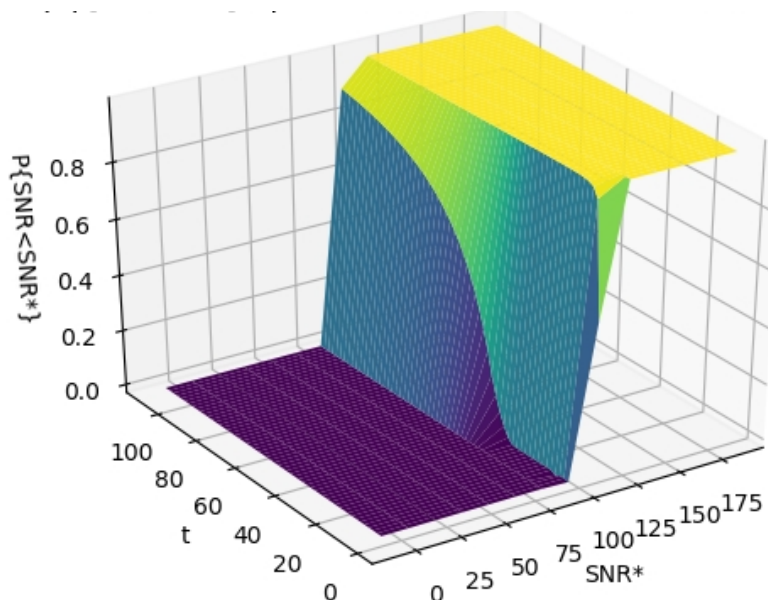


Рисунок 3.5 Зависимость $P\{\text{SNR}(t) < \text{SNR}^*\}$ от t и SNR^* для модели с декартовой системой координат.

$v=1.5 \text{ m/c}$, $a = -150$; $b = 150$; $h = 10$; $\lambda = 40$; $N = -100$; $P_t = 20$;

Таким образом, при фиксированном значении SNR^* , этот график будет показывать зависимость вероятности $P\{\text{SNR}(t) < \text{SNR}^*\}$ от времени, то есть, насколько вероятно, что качество соединения станет неприемлемым в определенный момент времени. Можно заметить, что на рисунке 3.5 при SNR^* от 0 до 80, $P\{\text{SNR}(t) < \text{SNR}^*\} = 0$. Это означает, что за рассматриваемый промежуток времени пользователь не сможет уйти так далеко, чтобы мощность соединения была ниже, чем 80db.

4. Заключение

В данной работе был проведен обзор литературы, касаемо технологии базовых летающих станций, а точнее, касаемо проблемы их позиционирования для обеспечения максимального покрытия пользователей.

Также была построена простая модель взаимодействия летающей базовой станции и пользователя, при условии наличия LoS; были построены графики, описывающие данную модель.

Источники

- [1] “Drone market outlook: industry growth trends, market stats and forecast”, Business Insider Intelligence, Mar 4, 2020. URL: <https://www.businessinsider.com/drone-industry-analysis-market-trends-growth-forecasts>
- [2] “Breakthrough in new type of battery for drones”, unify.aero, Feb 20, 2018. URL: <https://www.unify.aero/news/breakthrough-in-new-type-of-battery-for-drones>
- [3] Mary-Ann Russon, “Nokia and EE trial mobile base stations floating on drones to revolutionise rural 4G coverage”, International Business Times, August 15, 2016. URL: <https://www.ibtimes.co.uk/nokia-ee-trial-mobile-base-stations-floating-drones-revolutionise-rural-4g-coverage-1575795>
- [4] Hailong Huang, Andrey V Savkin, Ming Ding, Mohamed Ali Kaafar , “Optimized deployment of drone base station to improve user experience in cellular networks”, July 2019, DOI:10.1016/j.jnca.2019.07.002
- [5] Xiaohui Li, Optimal Deployment of Drone Base Stations for Cellular Communication by Network-based Localization, July 2018, DOI: 10.23919/ChiCC.2018.8483710
- [6] Path Loss Wikipedia page, URL: https://en.wikipedia.org/wiki/Path_loss
- [7] Signal to Noise Ratio Wikipedia page, URL: https://en.wikipedia.org/wiki/Signal-to-noise_ratio