

## Оценка позиций точек доступа системы обнаружения мобильных объектов в горных выработках

*Д.Д. Ларионов, Р.В. Воронов, Е.А. Питухин, Л.В. Щеголева*

*Петрозаводский государственный университет*

**Аннотация:** В данной работе обзревается ряд алгоритмов для решения задачи построения карты и задачи SLAM – Simultaneous localization and mapping – в различных постановках. Эти задачи решаются на графовой структуре, описывающей горную выработку, в которой установлена система локализации из стационарных точек доступа. Для решения задачи SLAM используется алгоритм FastSLAM в различных модификациях в зависимости от метода оценки позиций точек доступа. Во всех предложенных решениях удалось достичь ошибки оценки позиций точек доступа не более 2 метров. Задача построения карты решается как комбинаторная задача поиска перестановки точек доступа по известным позициям. Предложенный алгоритм поиска перестановки показал ощутимые преимущества по времени выполнения перед методом полного перебора на системах с 5 и более точками доступа.

**Ключевые слова:** граф, горная выработка, локализация, построение карты, SLAM, МСМС, фильтр частиц, фильтр Калмана, метод Байеса.

### Введение

Во многих системах локального позиционирования в качестве модели пространства используются графы. В качестве примера можно привести здания коридорного типа, подземные паркинги, подземные переходы, склады, горные выработки. В графах ребрам соответствуют прямолинейные участки путей, а вершинам – места их соединения, перекрестки, развилки.

Для систем локального позиционирования (далее – локализации) используются различные технологии. Один из подходов основан на использовании точек доступа беспроводной сети, расставленных в различных местах пространства. Мобильные объекты снабжены датчиками и обмениваются сигналами с точками доступа. По времени прохождения сигнала между мобильными датчиками и точками доступа оцениваются расстояния между ними.

Для повышения точности определения местоположения объектов и уменьшения времени расчетов используются специальные алгоритмы.

В некоторых из них гипотеза о местоположении объекта задается в виде отрезка на ребре графа. Несколько отрезков соответствует нескольким гипотезам. Чем больше длина отрезка, тем выше неопределенность знаний о местонахождении объекта.

В ходе развертывания и эксплуатации систем локализации могут возникать различные проблемы. Например, иногда отсутствует достоверная информация о реальном размещении стационарных точек доступа, их координаты могут быть заданы с некоторой погрешностью. Из-за сложности установки и большого объема работы в некоторых случаях данные о нахождении точек доступа могут быть перепутаны между собой. Для выявления таких случаев и уточнения информации о местонахождении датчиков используются методы SLAM – Simultaneous localization and mapping. Они характеризуются одновременным расчетом позиций мобильных объектов и стационарных точек доступа.

Ранее задача SLAM решалась в самых различных условиях. В данной работе рассмотрен случай решения этой задачи в горной выработке, системы локализации которых, как правило, основаны на радиокommunikациях [1].

Как уже было упомянуто выше, в системе локализации позиция мобильного объекта оценивается путем оценки расстояния до точек доступа. Оценка этого расстояния делается на основе времени пролета сигнала (ToF) или по силе получаемого сигнала (RSS). В основе этой оценки могут лежать самые разные радиотехнологии. Вот некоторые их примеры: UWB [2], Bluetooth [3], WiFi [4]. Также в качестве дополнительного источника информации могут быть использованы инерциальные системы, установленные непосредственно на мобильном объекте.

В данной работе дается обзор ряда статей авторов, опубликованных в материалах различных конференций IEEE [5 – 7]. В этих работах были представлены алгоритмы для решения задач SLAM и построения карты для

---

горной выработки с известной структурой, но неизвестными позициями точек доступа.

Далее будут рассмотрены общая модель структуры горной выработки, разные постановки задачи SLAM и задачи построения карты, а также алгоритмы решения этих задач на этой структуре. К каждой задаче приведено описание результатов тестирования предложенных алгоритмов.

### Общая модель

Структура горной выработки представлена в виде неориентированного взвешенного графа  $G = \{V, E\}$ , где множество ребер  $E$  соответствует прямым участками горной выработки, таким как тоннели, а множество вершин  $V$  – соединениям этих участков. Вес ребра  $w(e)$  соответствует длине участка горной выработки в метрах. Одна из вершин ребра выбирается в качестве точки отсчета в системе координат на этом ребре. Таким образом, позиция в горной выработке задается парой  $\{e, l\}$ , где  $e$  – некоторое ребро графа, а  $l \in [0, w(e)]$  – координата на этом ребре. Схожая модель представлена в [8].

Пусть  $x_1 = (e_1, l_1)$  и  $x_2 = (e_2, l_2)$  – некоторые позиции на графе, а ребро  $e_1$  соединяет вершины  $u_1$  и  $v_1$ , а ребро  $e_2$  –  $u_2$  и  $v_2$ . Если  $e_1 = e_2$ , то расстояние между двумя позициями это расстояние между двумя точками в системе координат ребра. В противном случае, расстояние между двумя позициями в горной выработке рассчитывается следующим образом:

$$\delta(x_1, x_2) = \min \left\{ \begin{array}{l} l_1 + \delta(u_1, u_2) + l_2 \\ l_1 + \delta(u_1, v_2) + w(e_2) - l_2 \\ w(e_1) - l_1 + \delta(v_1, u_2) + l_2 \\ w(e_1) - l_1 + \delta(v_1, v_2) + w(e_2) - l_2 \end{array} \right\}.$$

Моделью мобильного объекта служит точка на ребре графа, которую для краткости также будем называть мобильным объектом. Состояние объекта в дискретный момент времени  $t$  описывается в виде кортежа

$x_t = \{e_t, l_t, \theta_t\}$ , где  $\{e_t, l_t\}$  обозначает позицию мобильного объекта, а  $\theta_t$  – его ориентацию относительно ребра. Ориентация объекта используется для определения направления его движения по ребру. Состояние объекта обновляется на основе измерений  $\{s_t, \beta_t\}$ , где  $s_t$  – длина шага, а  $\beta_t$  изменение ориентации. Обновление состояния происходит согласно следующим формулам:

$$\begin{aligned}\theta_t &= (\theta_{t-1} + \beta_t) \bmod 2\pi, \\ l_t &= l_{t-1} + s_t \cos \theta_t.\end{aligned}$$

Если  $l \notin [0, w(e)]$ , то мобильный объект вышел за пределы текущего ребра и перешел на смежное ему. Координата и ориентация на новом ребре вычисляется в соответствии с тем, насколько сильно мобильный объект превысил вышеуказанный предел, а также со стороны какой вершины ребра он зашел.

Помимо мобильного объекта в горной выработке также расположена система локализация, состоящая из стационарных точек доступа с позициями  $a_k$ . В каждый момент времени  $t$  мобильный объект оценивает расстояния до точек доступа, сигнал от которых доступен ему в данный момент. Эта оценка имеет следующую форму:

$$z_{k,t} = \delta(a_k, x_t) + g_{k,t}, \text{ где } g_{k,t} \text{ – гауссовский шум с нулевым средним.}$$

Задача состоит в том, чтобы оценить позиции точек доступа и мобильного объекта в каждый момент времени  $t$ , используя оценки изменения состояния мобильного объекта и оценки расстояний до точек доступа.

Пусть  $m_k$  – оценка позиции точки доступа  $a_k$ . Вероятность получить измерение  $z_{k,t}$  от точки доступа на позиции  $m_k$  мобильным объектом на позиции  $x_t$ :

$$p(z_{k,t} | m_k, x_t) = N(z_{k,t} | \delta(m_k, x_t), \sigma_m^2),$$

где  $N(z_{k,t} | \delta(m_k, x_t), \sigma_m^2)$  – функция плотности вероятности нормально распределенной случайной величины со средним  $\delta(m_k, x_t)$  и дисперсией  $\sigma_m^2$ . Тогда задачу SLAM в общем виде можно сформулировать как оценку следующего распределения:

$$p(x_{1:t,m} | z_{1:t}) = p(x_{1:t} | z_{1:t}) p(m | x_{1:t}, z_{1:t}), \text{ где } p(m | x_{1:t}, z_{1:t}) = \prod_{k=1}^K p(m_k | x_{1:t}, z_{1:t}).$$

Оценка данного распределения во всех постановках производится с помощью модификаций алгоритма FastSLAM [9], в основе которого лежит фильтр частиц. Каждая частица с номером  $j=1..N$  хранит независимые оценки позиций мобильного объекта  $x_{1:t}^j$  и точек доступа  $m_{1:k}^j$  и имеет вес  $w^j \propto p(z_t | z_{1:t-1}, x_{1:t}^j)$ . На шаге  $t$  позиция каждой частицы обновляется. При этом при достижении вершины частица равновероятно выбирают одно из ребер, содержащее это вершину.

Для каждой оценки расстояния до точки доступа производится обновление оценки ее позиции. Если точка доступа обнаружена впервые, то позиция инициализируется некоторыми начальными значениями. Если позиция уже инициализирована, то обновляются распределение  $p(m | x_{1:t}, z_{1:t})$  и вес частицы. Далее осуществляется процедура ресемплинга. После этих шагов производится новый цикл со следующими измерениями.

В качестве алгоритма ресемплинга выбран «low variance resampling», описанный в [10].

### Непрерывная постановка задачи

В непрерывной постановке позиция точки доступа оценивается как непрерывное распределение, заданное следующими параметрами для частицы  $j$  и точки доступа  $k$  на шаге  $t$ :  $\mu_{k,t}^j$  – среднее значение,  $\nu_{k,t}^j$  – дисперсия. Для оценки этих величин используется сигма-точечный фильтр

Калмана [11]. Для каждой точки доступа при этом известно единственное ребро графа, но не известна позиция на этом ребре.

Ранее эта модель была представлена в [5].

На первом шаге алгоритма вычисляются сигма-точки:

$$X_0^j = \mu_{k,t-1}^j, \quad X_1^j = \mu_{k,t-1}^j - \sqrt{1+\lambda} \sqrt{v_{k,t-1}^j}, \quad X_2^j = \mu_{k,t-1}^j + \sqrt{1+\lambda} \sqrt{v_{k,t-1}^j}.$$

Параметр  $\lambda$  определяет разброс сигма-точек. Чем он выше, тем больше разброс. Для каждой сигма-точки вычисляется вес:

$$W_0 = \frac{\lambda}{1+\lambda}, \quad W_1 = W_2 = \frac{1}{2(1+\lambda)}.$$

Также необходимо вычислить значение функции расстояния на графе от сигма-точек до позиции частицы:

$$Y_i^j = \delta(x_t^j, \langle e_k, X_i^j \rangle), i = 0, 1, 2.$$

Далее вычисляется ряд вспомогательных величин:

$$a_j = \sum_{i=0}^2 W_i Y_i^j, \quad Q_j = \sum_{i=0}^2 W_i (Y_i^j - a_j)^2 + \sigma_m^2, \quad C_j = \sum_{i=0}^2 W_i (X_i^j - \mu_{k,t-1}^j) (Y_i^j - a_j).$$

С помощью выше описанных вспомогательных величин можно вычислить искомые среднее значение и дисперсию позиции точки доступа:

$$\mu_{k,t}^j = \mu_{k,t-1}^j + C_j (z_{k,t} - a_j) / Q_j, \quad v_{k,t}^j = v_{k,t-1}^j - C_j^2 / Q_j.$$

Также обновляется вес частицы:

$$w^j = N(z_{k,t} | a_j, Q_j).$$

### Дискретная постановка задачи

Задача в дискретной постановке изначально решалась для частного случая, когда для каждой точки доступа известно единственное ребро, но не известна позиция на этом ребре. Позднее эта же задача была решена для общего случая: ребра могут быть не известны или же известно одно или более ребер для отдельных точек доступа. Ниже описана постановка для общего случая.

Решения задачи как для частного, так и общего случая более подробно описаны в [5, 6].

В дискретной постановке распределение позиции точки доступа оценивается с помощью одномерной решетки [11] на ребрах графа. Каждому узлу этой решетки соответствует вероятность, с которой точка доступа находится в этом узле:

$$p_{k,e,i}^j(t) = p(e_k = e, l_k = i | x_{1:t}^j, z_{1:t}), \text{ где } i \in M_e = \{0, \dots, w(e)\}.$$

Вероятности обновляются через дискретный байесовский фильтр:

$$\tilde{p}_{k,e,i}^j(t) = p_{k,e,i}^j(t-1)p(z_{k,t} | m_k, x_t^j), \quad p_{k,e,i}^j(t) = \frac{\tilde{p}_{k,e,i}^j(t)}{\sum_{e \in E} \sum_{i \in M_e} \tilde{p}_{k,e,i}^j(t)}.$$

Веса частиц вычисляются как сумма вероятностей до нормировки:

$$w^j = \sum_{e \in E} \sum_{i \in M_e} \tilde{p}_{k,e,i}^j(t).$$

Позицию точки доступа можно получить следующим образом:

$$\mu_{k,T}^j = \frac{ip_{k,e_k^*,i}^j(T)}{\sum_{i \in M_e} p_{k,e_k^*,i}^j(T)}, \quad e_k^* = \arg \max_{e \in E} \sum_{i \in M_e} p_{k,e,i}^j(T).$$

Среди всех оценок выбирается оценка той частицы  $j$ , вес которой наибольший. Возможна ситуация, когда нельзя выбрать одно ребро для точки доступа. В этом случае существует несколько ребер, суммы вероятностей на которых равны. Для избегания таких ситуаций необходимо планировать маршрут таким образом, чтобы мобильный объект посещал больше ребер.

### Результаты экспериментов в непрерывной и дискретной постановке

Эксперименты с использованием синтетических данных в обеих постановках, показали возможности предложенных алгоритмов по решению задачи SLAM. Очевидным недостатком алгоритма для непрерывной постановки является необходимость заранее знать для каждой точки доступа единственное ребро, а также подбирать начальное значение дисперсии  $v_{k,0}^j$  –

чем больше среднеквадратическое отклонение шума  $\sigma_m$ , тем больше должно быть начальное значение дисперсии. Эксперименты проводились для  $\sigma_m \in [1.0, 3.0]$ . При подборе оптимальных параметров, оба алгоритма показывают приемлемый результат – среднее значение ошибки оценки позиций точек доступа не превышает 2 метров.

Отдельно также были проведены эксперименты для решения задачи SLAM в дискретной постановки в общем виде. Было оценено влияние других параметров на результат – оптимальным количеством частиц является 50, а для получения стабильного результата нужно знать единственное ребро для 30% точек доступа. В данных экспериментах также применялась процедура Resample Move, которая позволила улучшить отбор частиц после ресемплинга [13 – 15]. Средняя ошибка оценки позиций точки доступа, как и средняя ошибка локализации также не превысили 2 метров.

Более подробный анализ результатов можно увидеть в оригинальных статьях [5, 6].

### **Комбинаторная постановка задачи**

В описанных выше задачах по оценке траектории движения мобильного объекта и позиций точек доступа в основе алгоритма лежали допущения о том, что известна начальная позиция мобильного объекта, а его состояние можно оценить по некоторым измерениям длины шага и изменению ориентации, а на основе оценок расстояний до точек доступа скорректировать результат.

Однако, не всегда есть возможность измерить изменения состояния мобильного объекта, а начальная точка его траектории не всегда известна. Имеются лишь оценки расстояний от точек доступа до неизвестных позиций мобильного объекта. Если заранее известен возможный набор позиций точек доступа, то можно определить их расстановку по этим позициям.

Пусть  $U$  – конечный набор известных возможных позиций точек доступа и  $|U| = |A| = n$ . Задача состоит в поиске биективной функции  $f: N \rightarrow U$ , где  $N = \{1, \dots, n\}$  – номера точек доступа. Данная функция является перестановкой точек доступа с номерами  $N$  по позициям  $U$ . Определим также сужение этой функции –  $f_M$  – перестановку  $M \subset N$  по позициям  $U_M \subset U$ .

Перестановка или сужение считаются допустимыми, если  $f(i) \in U$  или  $f_M(i) \in U_i$  для всех  $i$ . Перестановка или сужение подтверждаются наблюдениями, если есть такая позиция из  $U$ , что для любого  $j = 1, \dots, n, d_{i,j} < \infty$  существует такая позиция мобильного объекта что расстояние от нее до точки доступа равно  $d_{i,j}$ .

Задача состоит в том, чтобы найти такую  $f$ , чтобы перестановка точек доступа была допустимой и подтверждалась измерениями на каждом шаге  $t$ . Очевидное решение данной задачи состоит в полном переборе всех возможных перестановок точек доступа по позициям  $U$  и проверка каждой из этих перестановок на допустимость и подтверждение наблюдениями. Однако, это потребует проверки  $n!$  перестановок, что в системах локализации с большим числом точек может быть слишком затратно по времени.

Альтернативное решение – применить полный перебор не для перестановок всех точек доступа сразу, а для каждой группы точек доступа, наблюдаемых в отдельные моменты времени  $t$ :

1. Методом полного перебора находятся перестановки точек доступа для каждого отдельного наблюдения. Множество всех таких частичных перестановок –  $A_t$ . Перестановки  $f_1$  и  $f_2$  совместимы, если одна и та же точка доступа в них не находится в разных позициях, а также нет такой позиции, в которой находились бы 2 разных точки доступа.

2. Строится граф  $H = (A, B)$ , где  $A$  – множество вершин, равно объединению всех  $A_i$ , а  $B$  – множество ребер, которое строится следующим образом: если 2 вершины соответствуют совместимым перестановкам, то между ними есть ребро.
3. На графе  $H$  находится максимальная клика  $K \subset A$  – подмножество всех вершин, попарно соединенных между собой. Будем говорить, что клика покрывает позиции  $U$ , если комбинация всех перестановок из  $K$  покрывает  $U$ . Клика находится по алгоритму Брона-Кербоша [16].
4. Решение задачи – объединение всех перестановок из найденной клики.

Ввиду того, что оценка расстояния до точки доступа получена с шумом, вместо расстояния до точки доступа из позиции мобильного объекта  $\{e, l\}$  следует использовать интервал  $[d_{i,k} - s, d_{i,k} + s]$ , где  $2s$  это длина сегмента ребра. Перестановка подтверждается наблюдениями, если существует пересечение интервалов всех наблюдаемых точек доступа.

Данный алгоритм ранее был представлен в [7].

### **Результаты экспериментов в комбинаторной постановке**

Был проведен ряд экспериментов на синтетических данных с разным количеством точек доступа – от 3 до 7. Для каждого набора точек доступа запускался 50 раз предложенный алгоритм и 50 раз алгоритм с полным перебором. В качестве метрики для сравнения алгоритмов использовался доверительный интервал времени выполнения с уровнем значимости 0.95.

Для 3–4 точек доступа предложенный алгоритм и полный перебор выполняются за примерно одинаковое время – (0.004; 0.008) и (0.009; 0.014) секунд. Однако разница между ними начинает резко расти с 5 точек доступа – (0.009; 0.013) и (0.052; 0.056) соответственно. Для 7 точек доступа различия становятся уже существенными – (0.031; 0.038) и (9.145; 9.211).

Описанные выше результаты демонстрируют преимущества предложенного алгоритма по сравнению с полным перебором. Стоит отметить, что это преимущество достижимо только в таких системах, где одно наблюдение в момент времени  $t$  не покрывает все точки доступа. Желательно, чтобы в один промежуток времени было доступно не более 3–4 точек доступа. В случае, если наблюдаются все точки доступа одновременно – предложенный алгоритм сводится к полному перебору их позиций. С другой стороны, в этом случае задача сводится к поиску наибольшего паросочетания в двудольном графе, в котором одна доля – точки доступа, а другая – места возможного их размещения, наличие ребра означает непротиворечивость текущим измерениям размещения точки доступа в данном месте.

Более полное описание можно увидеть в оригинальной статье [7].

### **Заключение**

В данной работе был рассмотрен ряд алгоритмов для решения задач построения карты и SLAM в различных постановках. Данные задачи решались для системы локализации в горной выработке, которая была описана как граф.

Для задачи SLAM было рассмотрено 2 постановки – непрерывная и дискретная. В непрерывной постановке оценивались такие параметры, как среднее значение и дисперсия позиции точки доступа, а в дискретной оценивалась вероятность нахождения точки доступа в узлах одномерной решетки на ребрах графа. Для решения обеих задач использовались модификации алгоритма FastSLAM. Все модификации алгоритмов показали ошибку оценки позиций точек доступа не более 2 метров, было исследовано влияние некоторых параметров на величину этой ошибки.

Также была рассмотрена комбинаторная задача построения карты по известному набору возможных позиций точек доступа. Был предложен

алгоритм частичного перебора возможных позиций точек доступа для отдельных наблюдений. При сравнении с полным перебором данных алгоритм показывает значительное преимущество во времени выполнения в системах локализации из 5 и более точек доступа.

### Литература (References)

1. Komane B. L. and Mathonsi T. E., A Review of Wireless Sensor Networks: Early Accident Detection Models for South African Mine Industries. Open Innovations (OI), 2019, Cape Town, South Africa, pp. 36–41.
  2. Zhu D. and Yi K. IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC), 2011, Xi'an, China, pp. 1–4.
  3. Viswanathan S. and Srinivasan S., IEEE SENSORS, 2015, Busan, Korea (South), pp. 1–4.
  4. Zhao L., Wang H., Li P. and Liu J., An improved WiFi indoor localization method combining channel state information and received signal strength, 36th Chinese Control Conference (CCC), 2017, Dalian, China, pp. 8964–8969.
  5. Larionov D., Lukashenko O., Moschevikin A., Voronov R. and Sikora A., "Improving the Accuracy for Radio-Based Positioning in Mines Using SLAM", IEEE 5th International Symposium on Smart and Wireless Systems within the Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS), 2020, Dortmund, Germany, pp. 1–7.
  6. Larionov D., Lukashenko O., Moschevikin A., Voronov R. and Sikora A., Correction of Uncertain Access Points Positions in Underground Mines Using SLAM Approach, 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), 2021, Cracow, Poland, pp. 761–766.
  7. Larionov D. and Voronov R. Combinatorial Approach for the Access Points Map Correcting in the Mines Positioning Systems. International Russian
-

Automation Conference (RusAutoCon), 2022, Sochi, Russian Federation, pp. 734–738.

8. Voronov R., Moschevikin A. and Soloviev A., Algorithm for Smoothed Tracking in Underground Local Positioning System. International Russian Automation Conference (RusAutoCon), 2018, Sochi, Russia, pp. 1–6.

9. Montemerlo M., Thrun S., Koller D., and Wegbreit B. American Association for Artificial Intelligence, 2002, USA, pp. 593–598.

10. Thrun S., Burgard W., and Fox D., Probabilistic Robotics (Intelligent Robotics and Autonomous Agents), The MIT Press, 2005, p. 86.

11. Särkkä S., Bayesian Filtering and Smoothing, Cambridge University Press, 2013, pp. 86–92.

12. Burgard W., Fox D., Hennig D., and Schmidt T., Estimating the absolute position of a mobile robot using position probability grids. AAAI/IAAI, 1996, vol. 2, pp. 896–901.

13. Doucet A. and Johansen A. M., A tutorial on particle filtering and smoothing: fifteen years later. The Oxford handbook of nonlinear filtering, 2011, pp. 656–705.

14. Fearnhead P., Journal of Computational and Graphical Statistics, 2002, vol. 11, no. 4, pp. 848–862.

15. Gilks W. and Berzuini C., Journal of The Royal Statistical Society Series B-statistical Methodology, 2001, vol. 63, pp. 127–146.

16. Bron C. and Kerbosch J., “Algorithm 457: finding all cliques of an undirected graph”, Commun. ACM, 1973, vol. 16, no. 9, pp. 575–577.