



Основы вероятностного метода SLAM в робототехнике

Андрей Антонов (robotosha@mail.ru)

В статье излагаются основы метода одновременного определения местоположения мобильного робота и построения карты (SLAM) (данный метод активно развивается и широко используется при создании автономных мобильных платформ), а также рассматриваются решения задачи SLAM с использованием расширенного фильтра Калмана (EKF-SLAM) и с применением частичного фильтра (FastSLAM).

SLAM (Simultaneous Localization and Mapping) можно перевести с английского как «одновременное определение местоположения и построение карты». Задача SLAM возникает в случаях, когда мобильный робот, помещённый в неизвестное место с неизвестной окружающей средой, постепенно строит полную карту местности и одновременно определяет своё местоположение в пределах построенной карты. Данный метод позволяет сделать робота по-настоящему автономным.

Задача SLAM была сформулирована теоретически и решена на практике несколькими способами, что явилось одним из наиболее заметных успехов в области робототехники за последние два десятка лет. Методики SLAM были реализованы для роботов, работающих внутри и вне помещений, под водой, а также для летающих робототехнических систем. В качестве примеров современных роботов, в которых используется SLAM, можно привести самоуправляемые автомобили Google [1] и роботы-пылесосы от Neato Robotics [2].

История задачи SLAM

Впервые вероятностная задача SLAM была представлена в 1986 году на конференции *Robotics and Automation*, проводимой Институтом инженеров электротехники и электроники (IEEE) в Сан-Франциско (США). В то время вероятностные методы только начинали внедряться в области робототехники и искусственного интеллекта. Ряд исследователей искали способы применения теоретико-оценочных мето-

дов в задачах создания карты и определения местоположения. В их числе были Питер Чизман, Джим Кроули и Хью Дюрант-Уайт. В ходе конференции широко обсуждалось последовательное создание карты окружающей среды.

В результате дискуссий было решено, что последовательное вероятностное создание карты является фундаментальной проблемой в робототехнике с серьёзными концептуальными проблемами и сложными вычислительными задачами. В течение следующих нескольких лет работы Смита и Чизмана [3] и Дюранта-Уайта [4] заложили статистическую основу для описания отношений между различными ориентирами на карте и решения задачи геометрической неопределённости. Проделанная учёными работа показала, что между оценками расположения различных ориентиров на карте должна быть высокая степень корреляции и что, на самом деле, эти корреляции растут при проведении последующих наблюдений.

В это же время Аяч и Фоджерас опубликовали работы в области визуальной навигации [5], Кроули, Чатила и Ламонд работали над навигацией мобильных роботов, основанной на ультразвуковых датчиках с использованием фильтровых алгоритмов Калмана [6, 7]. Данные исследования имели много общего, результатом чего стала обобщающая статья Смита [8]. В этой работе было показано, что, если мобильный робот движется в неизвестной среде, производя относительные наблюдения за ориентирами, то все оценки местоположения ориентиров обязательно коррелируют друг с дру-

гом, являясь причиной общей ошибки в расчётах местоположения транспортного средства. Полученные выводы имели большое значение. Оказалось, что полное решение комплексной задачи определения местоположения и построения карты требует совместного определения состояния системы, включающего в себя положение транспортного средства и каждого ориентира. При этом состояние системы изменяется после каждого наблюдения за ориентиром. Решение данной задачи потребовало бы использования огромного вектора состояния и большого объёма вычислений.

Важно отметить, что в рамках этой работы не преследовались цели обеспечения сходимости карты или условий её стационарного поведения. Было принято считать, что ошибки оценки карты не сходятся, а вместо этого имитируют процесс «случайного блуждания» с неограниченным ростом ошибки. Таким образом, учитывая вычислительную сложность задачи построения карты, исследователи сосредоточились на приближённом решении задачи последовательного построения карты, что подразумевало минимизацию или устранение корреляций между ориентирами [9]. По тем же причинам теоретическая работа по решению комбинированной задачи определения местоположения и построения карты была временно приостановлена, и исследования часто сосредотачивались либо на построении карты, либо на определении местоположения.

Концептуальный прорыв произошёл с осознанием того, что комбинированная задача построения карты и определения местоположения, сформулированная как единная задача, была на самом деле сходящейся. Было признано, что корреляции между ориентирами, которые большинство исследователей пытались минимизировать, были критической частью задачи, и что, наоборот, с ростом корреляций решение получа-

ется более точным. Структура задачи SLAM, сходящееся решение и аббревиатура SLAM были впервые представлены в работе, посвящённой исследованию мобильной робототехники, в 1995 году на Международном симпозиуме по исследованию робототехники [10]. После этого несколько групп учёных, работавших над задачами построения карты и определения местоположения, начали серьёзно работать над SLAM.

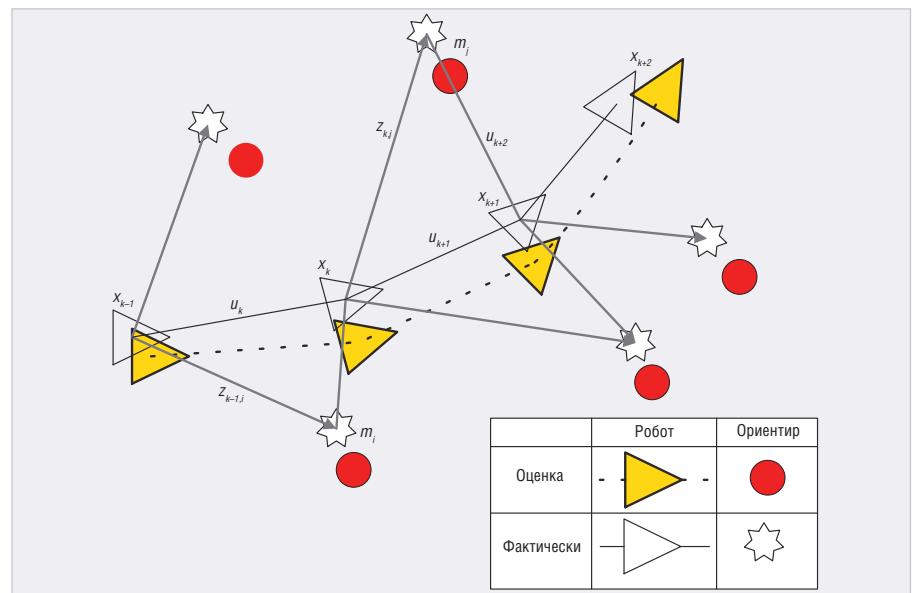
На этот раз усилия специалистов были сконцентрированы на повышении эффективности вычислений и решении проблемы «закрытия петель». Международный симпозиум по исследованию робототехники 1999 года (ISRR'99) [11] стал важным мероприятием, в рамках которого была проведена первая конференция по SLAM. Себастиан Тран представил свою работу [12], в которой соединил воедино методы SLAM, основанные на применении фильтра Калмана, вероятностное определение местоположения и методы построения карты. Проводившийся в 2000 году семинар по SLAM в рамках Международной конференции по робототехнике и автоматизации IEEE (ICRA) привлёк полтора десятка исследователей и был посвящён таким вопросам, как алгоритмическая сложность, сопоставление данных и проблемы реализации метода. В 2002 году семинар по SLAM на конференции ICRA привлёк уже 150 специалистов в различных областях. В последующие годы появилось большое количество научных работ, позволивших решить многие проблемы, возникающие в SLAM.

ФОРМУЛИРОВКА И СТРУКТУРА ЗАДАЧИ SLAM

SLAM является процессом, в рамках которого мобильный робот может построить карту окружающей среды и одновременно использовать эту карту, чтобы сделать выводы о своём местоположении. Траектория движения мобильной платформы и расположение всех ориентиров в SLAM оцениваются в реальном времени, без необходимости какого-либо априорного знания о местоположении.

Предварительные положения

Рассмотрим мобильного робота, движущегося в некоторой окружающей среде и получающего данные о ней посредством наблюдения ряда неизвестных ориентиров с помощью датчика (см. рис. 1). В момент времени определены следующие величины:



Примечание: истинные позиции ориентиров неизвестны и не могут быть прямо измерены

Рис. 1. Одновременная оценка местоположения робота и ориентиров

- x_k – вектор состояния, описывающий местоположение и ориентацию робота;
- u_k – вектор управления, приложенный в момент времени $k-1$ для приведения робота в состояние x_k ;
- m_i – вектор, описывающий местоположение i -го ориентира, чьё истинное местоположение предполагается постоянным;
- z_{ik} – наблюдение i -го ориентира, полученное роботом в момент времени k . При наличии нескольких наблюдений ориентира в момент времени k или когда конкретный ориентир не относится к рассматриваемым, наблюдение будет записываться просто как z_{ik} .

Также определены:

- $X_{0:k} = \{x_0, x_1, \dots, x_k\} = \{X_{0:k-1}, x_k\}$ – история местоположений робота;
- $U_{0:k} = \{u_1, u_2, \dots, u_k\} = \{U_{0:k-1}, u_k\}$ – история управляемых входов;
- $m = \{m_0, m_1, \dots, m_n\}$ – множество всех ориентиров;
- $Z_{0:k} = \{z_1, z_2, \dots, z_k\} = \{Z_{0:k-1}, z_k\}$ – множество всех наблюдений ориентиров.

Вероятностный метод SLAM

В вероятностной форме задача одновременного определения местоположения и построения карты (SLAM) требует, чтобы распределение вероятности

$$P(x_k, m | Z_{0:k}, U_{0:k}, x_0) \quad (1)$$

было вычислено для всех моментов времени k . Это распределение вероятности описывает совместную апостериорную плотность местоположения ори-

ентиров и состояния робота (в момент времени k), определяемую данными наблюдений и управляющих входов. В подобных случаях предпочтительно использовать рекурсивные методы решения задачи. Оценка распределения $(x_{k-1}, m | Z_{0:k-1}, U_{0:k-1})$ в момент времени $k-1$ в соответствии с управляющим воздействием u_k и наблюдением z_k вычисляется с помощью теоремы Байеса. Эти вычисления требуют, чтобы модели перехода состояния и наблюдения описывали воздействие управляющего входа и наблюдение соответственно.

Модель наблюдения описывает вероятность наблюдения z_k при известном местоположении робота и ориентиров:

$$P(z_k | x_k, m). \quad (2)$$

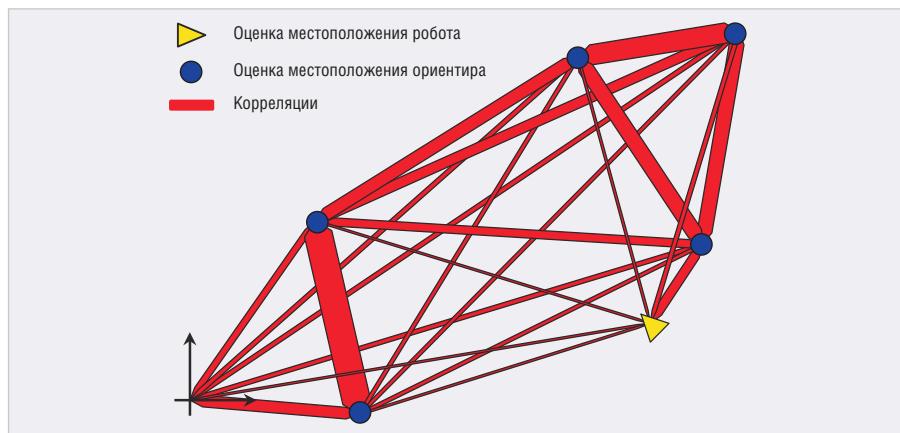
Резонно предположить, что как только местоположение робота и карта определены, наблюдения становятся условно независимыми для данной карты и текущего состояния робота.

Модель движения для робота может быть описана в терминах распределения вероятности перехода состояния в форме

$$P(x_k | x_{k-1}, u_k). \quad (3)$$

Предполагается, что переход состояния является марковским процессом, в котором следующее состояние x_k зависит только от предшествующего состояния x_{k-1} и приложенного управляющего воздействия u_k и не зависит от наблюдения и карты.

Алгоритм SLAM реализован в стандартной двухшаговой рекурсивной форме предсказания (временнóе



Примечание: при движении робота вперёд и назад в окружающей среде жёсткость пружины (корреляция) увеличивается – красные соединения становятся толще

Рис. 2. Аналогия с сетью пружин

обновление) коррекции (обновление измерения).

Временное обновление:

$$P(x_k, m | Z_{0:k-1}, U_{0:k}, x_0) = \int P(x_k | x_{k-1}, u_k) \times \\ \times P(x_{k-1}, m | Z_{0:k-1}, U_{0:k-1}, x_0) dx_{k-1}. \quad (4)$$

Обновление измерения:

$$P(x_k, m | Z_{0:k}, U_{0:k}, x_0) = \\ \frac{P(z_k | x_k, m) P(x_k, m | Z_{0:k-1}, U_{0:k}, x_0)}{P(z_k | Z_{0:k-1}, U_{0:k})}. \quad (5)$$

Уравнения (4) и (5) описывают рекурсивную процедуру для вычисления совместной апостериорной плотности вероятности (1) для состояния робота x_k и карты m в момент времени k , основываясь на всех наблюдениях $Z_{0:k}$ и всех управляющих входах $U_{0:k}$ вплоть до момента времени k . Рекурсия является функцией модели робота (3) и модели наблюдения (2).

Стоит отметить, что задача построения карты может быть сформулирована как задача вычисления условной плотности вероятности $P(x_k | Z_{0:k}, U_{0:k}, m)$. При этом предполагается, что местоположение робота x_k известно (или, по крайней мере, определено) для всех моментов времени. Карта m строится путём объединения данных наблюдений из различных местоположений. С другой стороны, задача определения местоположения может быть сформулирована как вычисление распределения той же вероятности. При этом предполагается, что местоположение ориентиров известно достоверно, и цель состоит в том, чтобы оценить местоположение робота относительно этих ориентиров.

Структура вероятностного метода SLAM

Для упрощения в данном разделе опущены исторические переменные в формуле (1) и совместная апостери-

орная плотность вероятности для карты и местоположения робота будет обозначаться $P(x_k, m | z_k)$ или даже $P(x_k, m)$ (в некоторых случаях).

Модель наблюдения (2) делает очевидной зависимость наблюдений и от местоположения робота, и от расположения ориентиров. Из этого следует, что совместная апостериорная плотность вероятности не может быть разделена обычным способом $P(x_k, m | z_k) \neq P(x_k | z_k) P(m | z_k)$.

Тем не менее, задача SLAM имеет более сложную структуру, чем кажется.

Обратившись снова к рисунку 1, можно увидеть, что ошибка между оценённым и истинным местоположениями ориентира обуславливается в основном ошибкой в определении текущего местоположения робота. Это означает, что ошибки в оценках местоположения ориентиров являются сильно коррелированными. На практике это приводит к тому, что относительное местоположение между любыми двумя ориентирами m_i, m_j может быть известно с высокой точностью, несмотря на то что абсолютное местоположение ориентира m_i в некоторой степени неопределённо. В вероятностной форме это означает, что совместная плотность вероятности для пары ориентиров $P(m_i, m_j)$ имеет высокий пик даже в случаях, когда граничные плотности вероятности $P(m_i)$ обладают достаточно большой дисперсией.

Наиболее важным выводом в теории SLAM является то, что корреляции между оценками ориентиров монотонно возрастают с увеличением числа произведённых наблюдений. Другими словами, знание относительного месторасположения ориентиров всегда улучшается и никогда не расходится, независимо от движения робота.

В вероятностной терминологии это означает, что совместная плотность вероятности для всех ориентиров $P(m)$ становится более остроконечной с ростом числа произведённых наблюдений.

Такая сходимость является следствием того, что наблюдения, произведённые роботом, могут рассматриваться как «почти независимые» измерения относительного месторасположения между ориентирами. Снова вернёмся к рисунку 1 и рассмотрим робота в позиции x_k , наблюдающего два ориентира: m_i и m_j . Относительное расположение наблюдаемых ориентиров, очевидно, независимо от координат робота, и последовательные наблюдения из этого фиксированного месторасположения дадут дополнительные независимые измерения относительного расположения ориентиров. Теперь, когда робот перемещается в местоположение x_{k+1} , он вновь наблюдает ориентир m_j . Это позволяет обновить оценённое местоположение робота и ориентира по отношению к предыдущему местоположению x_k . В свою очередь, это справедливо и для обновления местоположения ориентира m_i , пусть даже ориентир и не наблюдается из нового местоположения. Так происходит из-за того, что два ориентира сильно коррелированы (их относительное местоположение хорошо известно) с предыдущими измерениями. Кроме того, тот факт, что те же самые измерения используются для обновления данных двух ориентиров, делает их ещё более коррелированными. Термин «почти независимое измерение» является уместным, потому что ошибки наблюдений будут коррелированными из-за последовательных перемещений робота. Также стоит обратить внимание на то, что в позиции x_{k+1} робот наблюдает два новых ориентира относительно m_j . Эти новые ориентиры, таким образом, сразу же связываются или коррелируют с остальной картой. Последующие обновления этих ориентиров будут также обновлять ориентир m_i и через него m_j , и так далее. Таким образом, все ориентиры в конечном счёте образуют сеть, связывающую относительные местоположения или корреляции, точность которых увеличивается каждый раз при произведении очередного наблюдения.

Данный процесс может быть представлен в виде сети пружин, соединяющих все ориентиры вместе (см. рис. 2). Наблюдение за соседним звеном действует как смещение в пружинной системе таким образом, что его эффект является значимым для ближайших звеньев

и в зависимости от локальной жёсткости (корреляции) уменьшается с расстоянием до других ориентиров. По мере движения робота в окружающей среде и наблюдения за ориентирами, пружины становятся всё более жёсткими, причём монотонно. В конечном итоге получается жёсткая карта ориентиров или точная относительная карта окружающей среды. Как только карта построена, точность местоположения робота ограничена только качеством карты и относительным измерением датчика. В теоретическом пределе точность относительного местоположения робота становится равной точности определения местоположения, достижимого для данной карты.

Решения задачи SLAM

Решения вероятностной задачи SLAM включают в себя нахождение соответствующего представления для модели наблюдения (2) и модели движения (3), которые позволяют эффективно и полно вычислять предыдущие и последующие распределения (4) и (5). Безусловно, наиболее полным представлением является модель пространства состояний с аддитивным гауссовым шумом, что приводит к необходимости использования расширенного фильтра Калмана (EKF). Одним из важных альтернативных подходов к решению является описание модели движения робота в (3) в виде выборок из более общего, отличного от нормального распределения вероятности. В таком случае для решения задачи используется частичный фильтр (фильтр частиц) или алгоритм FastSLAM. В то время как EKF-SLAM и FastSLAM являются наиболее важными методами решения, были предложены и другие современные методы, обладающие большим потенциалом [13].

Метод EKF-SLAM

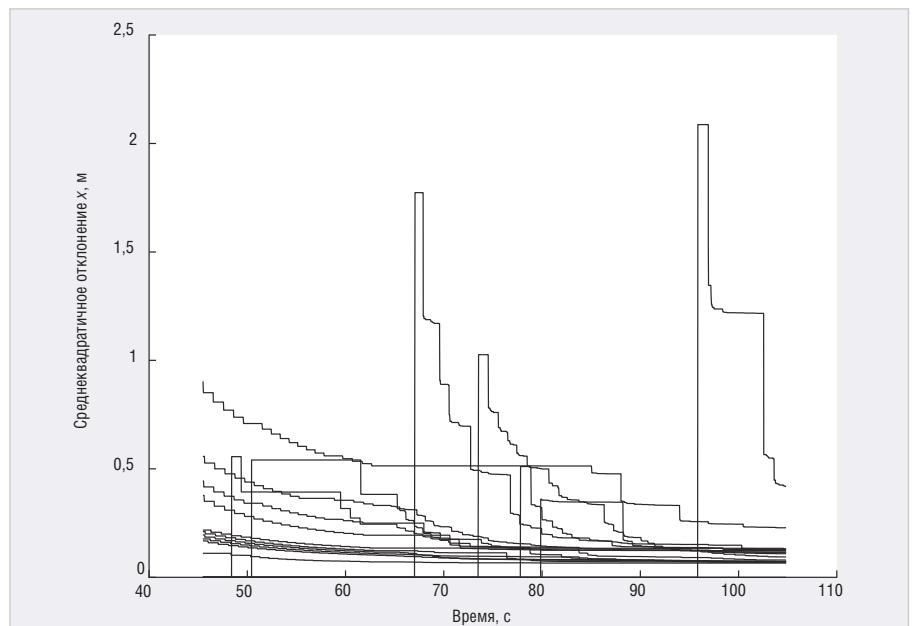
В основе метода EKF-SLAM лежит описание движения робота в форме

$$P(x_k | x_{k-1}, u_k) \Leftrightarrow x_k = f(x_{k-1}, u_k) + w_k, \quad (6)$$

где $f(\dots)$ описывает кинематическую модель робота и w_k является аддитивным, некоррелированным возмущением гауссовского движения с нулевым средним и ковариацией Q_k . Модель наблюдения описывается формулой

$$P(z_k | x_k, m) \Leftrightarrow z_k = h(x_k, m) + v_k, \quad (7)$$

где $h(\dots)$ описывает геометрию наблюдения, а v_k является аддитив-



Примечание: ориентир изначально наблюдается с неопределенностью, унаследованной от местоположения робота и наблюдения; с течением времени среднеквадратичные отклонения монотонно убывают до нижней границы

Рис. 3. Сходимение неопределённости ориентиров

ной некоррелированной гауссовой ошибкой с нулевым средним и ковариацией R_k .

Таким образом, стандартный метод EKF [14] может быть применён для вычисления среднего

$$\begin{bmatrix} \hat{x}_{k|k} \\ \hat{m}_k \end{bmatrix} = E \begin{bmatrix} x_k \\ m \end{bmatrix} | Z_{0:k}$$

и ковариации совместного апостериорного распределения плотности вероятности (1)

$$P_{k|k} = \begin{bmatrix} P_{xx} & P_{xm} \\ P_{xm}^T & P_{mm} \end{bmatrix}_{k|k} = \\ = E \left[\begin{pmatrix} x_k - \hat{x}_k \\ m - \hat{m}_k \end{pmatrix} \begin{pmatrix} x_k - \hat{x}_k \\ m - \hat{m}_k \end{pmatrix}^T \mid Z_{0:k} \right]$$

Используемые временнéе обновление и обновление наблюдения записываются следующим образом. Временнéе обновление:

$$\hat{x}_{k|k-1} = f(\hat{x}_{k-1|k-1}, u_k), \quad (8)$$

$$P_{xx,k|k-1} = \nabla f P_{xx,k-1|k-1} \nabla f^T + Q_k, \quad (9)$$

где ∇f – якобиан f , вычисляющий оценку $\hat{x}_{k-1|k-1}$. Для стационарных ориентиров временнéе обновление обычно не выполняется, однако оно необходимо при наличии подвижных объектов.

Обновление наблюдения:

$$\begin{bmatrix} \hat{x}_{k|k} \\ \hat{m}_k \end{bmatrix} = \begin{bmatrix} \hat{x}_{k|k-1} \hat{m}_{k-1} \end{bmatrix} + W_k \left[z_k - h(\hat{x}_{k|k-1}, \hat{m}_{k-1}) \right], \quad (10)$$

$$P_{k|k} = P_{k|k-1} - W_k S_k W_k^T, \quad (11)$$

где $S_k = \nabla h P_{(k|k-1)} \nabla h^T + R_k$, $W_k = P_{(k|k-1)} \nabla h^T S_k^{(-1)}$, а ∇h – якобиан вычисляющий $\hat{x}_{k|k-1}$ и \hat{m}_{k-1} .

Данное решение задачи EKF-SLAM, его преимущества и недостатки хорошо известны. Необходимо обратить внимание на ключевые особенности этого метода.

1. В задаче EKF-SLAM сходимость карты проявляется в виде монотонной сходимости к нулю определителя ковариационной матрицы $P_{mm,k}$ карты и всех подматриц пар ориентиров. Отдельные дисперсии ориентиров сходятся к нижней границе, определяемой начальной неопределенностью в положении робота и наблюдения. Типичная сходимость дисперсий положения ориентиров показана на рисунке 3 [14].
2. После каждого наблюдения все ориентиры и совместная ковариационная матрица обновляются. Это означает, что объём вычислений растёт квадратично с ростом числа ориентиров.
3. Стандартное решение EKF-SLAM очень чувствительно к ошибочному сопоставлению наблюдений с ориентирами. Особенно сложной является задача «закрытия петли», когда робот возвращается для повторного наблюдения ориентиров после длительного «путешествия». Проблема сопоставления усугубляется в окружающей среде, где ориентиры не являются точечными объектами, а заметно отличаются при рассмотрении с различных углов обзора.
4. Метод EKF-SLAM использует линеаризованные модели нелинейных пере-

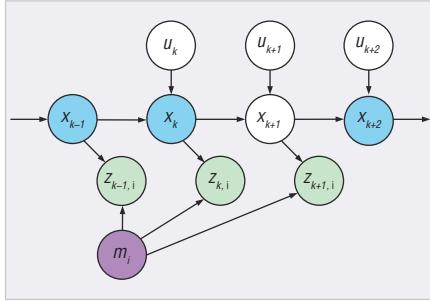
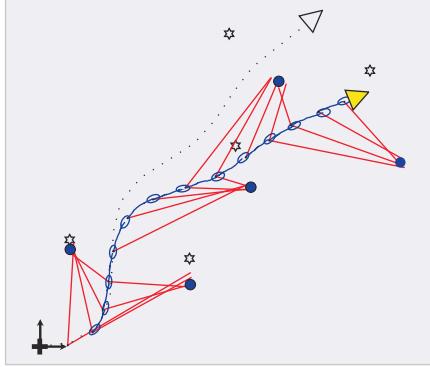


Рис. 4. Графическая модель алгоритма SLAM



Примечание: эллипсами показаны заявленные распределения для каждой стадии обновления, из которых производятся выборка позиций робота и обновление наблюдений ориентиров; множество таких траекторий представляет собой вероятностную модель местоположения робота

Рис. 5. Реализация траектории робота в алгоритме FastSLAM

мещений и модели наблюдения. Нелинейность может быть существенной проблемой в EKF-SLAM и приводит к неустойчивому решению. Сходимость и устойчивость могут быть гарантированы только для линейного случая.

Частичный фильтр

Алгоритм FastSLAM, представленный Монтерерло и коллегами [15], стал концептуальным прорывом в разработке рекурсивного вероятностного метода SLAM. Предыдущие усилия были направлены на повышение производительности EKF-SLAM, сохраняя при этом необходимость в линейных гауссовских предположениях. Метод FastSLAM, основанный на рекурсивной выборке по методу Монте-Карло или частичной фильтрации, был первым алгоритмом, работающим с нелинейной моделью процесса и отличным от нормального распределением. Несмотря на то, что FastSLAM также линеаризует модель наблюдения, это, как правило, является допустимым приближением для азимутально-диапазонных измерений, когда позиция робота известна.

Большая размерность пространства состояний задачи SLAM делает прямое использование частичных фильтров сложно реализуемым с точки зрения вычислительных затрат. Однако, возможно сократить выборочное множество, разделив его на части в соответствии с правилом умножения $P(x_1, x_2) = P(x_2|x_1) \times P(x_1)$, и если $P(x_2|x_1)$ может быть представлена аналитически, то только $P(x_1)$ должна быть сэмплирована – $x_1^{(i)} \sim P(x_1)$. Следовательно, совместное распределение представляет собой множество $\{x_1^{(i)}, P(x_2|x_1) \times P(x_1)\}$ и статистически предел

$$(x_2) \approx \frac{1}{N} \sum_i^N P(x_2 | x_1^{(i)})$$

вычисляется с большей точностью, чем при выборке совместного пространства.

Совместное состояние SLAM может быть факторизовано в компонент робота и компонент условной карты:

$$\begin{aligned} P(X_{0:k}, m | Z_{0:k}, U_{0:k}, x_0) &= \\ &= P(m | X_{0:k}, Z_{0:k}) P(X_{0:k} | Z_{0:k}, U_{0:k}, x_0), \end{aligned} \quad (12)$$

Здесь используется распределение плотности вероятности по траектории $X_{0:k}$, а не в единственном положении x_k , т.к. в случае траектории карта ориентиров становится независимой (см. рис. 4). Это ключевое свойство алгоритма FastSLAM и является причиной высокой скорости его работы. Карта представляется в виде множества независимых гауссиан с линейной сложностью, а не совместной картой ковариации с квадратичной сложностью.

Фундаментальная структура FastSLAM представляет собой состояние, где траектория представлена взвешенной выборкой и карта вычисляется аналитически. Так, совместное распределение в момент времени k представляется множеством $\{w_k^{(i)}, X_{0:k}^{(i)}, P(m | X_{0:k}^{(i)}, Z_{0:k})\}_i^N$, где карта, сопутствующая каждой частице, состоит из независимых нормальных распределений

$$P(m | X_{0:k}^{(i)}, Z_{0:k}) = \prod_j^M P(m_j | X_{0:k}^{(i)}, Z_{0:k}).$$

Частичная фильтрация для состояний положения выполняется рекурсивно, затем для состояний карты применяется EKF. Каждый наблюдаемый ориентир обрабатывается индивидуально как обновление измерения EKF из известной позиции (см. рис. 5). Положение наблюдаемых ориентиров не изменяется.

Рассмотрим обобщённую форму частичного фильтра для SLAM. В момент времени $k-1$ совместное состояние представляется как $\{w_{k-1}^{(i)}, X_{0:k-1}^{(i)}, P(m | X_{0:k-1}^{(i)}, Z_{0:k-1})\}_i^N$. Затем выполняются следующие действия:

- Для каждой частицы вычисляется заявленное распределение, обусловленное историей конкретной частицы, и получается выборочное значение:

$$x_k^{(i)} \sim \pi(x_k | X_{0:k-1}^{(i)}, Z_{0:k}, u_k). \quad (13)$$

Полученная новая выборка является (неявно) присоединённой к истории частицы.

- Выборки взвешиваются в соответствии с функцией важности:

$$w_k^{(i)} = w_{k-1}^{(i)} \frac{P(z_k | X_{0:k}^{(i)}, Z_{0:k-1}) P(x_k^{(i)} | X_{k-1}^{(i)}, u_k)}{\pi(x_k^{(i)} | X_{0:k-1}^{(i)}, Z_{0:k}, u_k)}. \quad (14)$$

Члены в числителе этого выражения представляют собой модели наблюдения и движения соответственно. Первый отличается от (2) вследствие того, что частичный фильтр требует ограниченности на карте.

$$\begin{aligned} P(z_k | X_{0:k}, Z_{0:k-1}) &= \\ &= \int P(z_k | x_k, m) P(m | X_{0:k-1}, Z_{0:k-1}) dm. \end{aligned} \quad (15)$$

- В случае необходимости формируется новая выборка. В одних реализациях повторная выборка делается на каждой итерации, в других – после фиксированного числа итераций, в третьих – если вес дисперсии превышает некоторое пороговое значение. Повторная выборка осуществляется путём выбора частиц с удалением их из множества $\{X_{0:k}^{(i)}\}_i^N$, включая связанные с ними карты, с вероятностью выбора, пропорциональной $w_k^{(i)}$. Выбранные частицы равномерно взвешиваются на величину $1/N$.

- Для каждой частицы выполняется обновление EKF на наблюдаемых ориентирах, т.е. производится построение карты с известной позицией робота.

В литературе описаны две версии FastSLAM – версии 1.0 [17] и 2.0 [18]. Они различаются только членами в форме их заявленного распределения (шаг 1) и, следовательно, в весах важности (шаг 2). Версия FastSLAM 2.0 является более эффективным решением.

Для FastSLAM 2.0 заявленное распределение включает текущее наблюдение

$$x_k^{(i)} \sim P(x_k | X_{0:k-1}^{(i)}, Z_{0:k}, u_k), \quad (16)$$

где

$$\begin{aligned} P(x_k | X_{0:k-1}^{(i)}, Z_{0:k}, u_k) &= \\ &= \frac{1}{C} P(z_k | x_k, X_{0:k-1}^{(i)}, Z_{0:k-1}) P(x_k | X_{k-1}^{(i)}, u_k) \end{aligned}$$

и C является нормализованной константой. Распределение весов в зависимости от важности в соответствии с (14) есть $w_k^{(i)} = w_{k-1}^{(i)} C$. Преимуществом FastSLAM 2.0 является то, что его заявленное распределение является локально оптимальным. Для каждой частицы это даёт наимень-

шую возможную дисперсию в значимом взвешивании $w_k^{(i)}$, зависящую от доступной информации $Z_{0:k-1}^{(i)}$, $Z_{0:k}$ и $U_{0:k}$.

Статистически обе версии FastSLAM подвержены вырождению из-за их неспособности «забыть прошлое». Ограничение карты в (15) зависит от истории положения и измерения и при повторной выборке истощает эту историю, статистическая точность теряется. Тем не менее, эмпирические результаты FastSLAM 2.0 в реальных условиях внешней среды [18] показали, что алгоритм способен генерировать достаточно точную карту.

Реализация SLAM

Эксперимент «исследуй и возвращайся», проведённый Ньюманом с группой исследователей [19], являлся среднемасштабной реализацией метода внутри помещения, который подтвердил нерасходимость свойств EKF-SLAM возвращением робота точно в исходную точку. Эксперимент был примечателен тем, что возвращение было полностью автономным. Робот управлялся вручную в исследовательской фазе, хотя и без визуального контакта с оператором, который полагался исключительно на построенную в реальном времени роботом карту. На обратном пути робот планировал маршрут и возвращался в исходную точку без вмешательства человека. Гайвант и Небот [20] впервые применили SLAM на большом открытом пространстве. В данной работе были изучены вопросы проведения вычислений в реальном времени при работе с высокоскоростным движением транспортного средства по пересечённой местности с динамическими препятствиями. Представленные результаты особенно интересны, поскольку они сопровождаются точными координатами, полученными с помощью технологии RTK-GPS, и показали практическую работоспособность алгоритма.

В настоящее время приложения SLAM задействованы в самых различных областях и используют такие методы зондирования, как «только азимут» [21] и «только расстояние» [22].

Также следует упомянуть и о методе последовательной оценки положения (Consistent Pose Estimation, CPE) [23], который представляет собой другую парадигму SLAM, основанную на топологическом построении карты и выравнивании данных. Данный метод показывает отличные результаты внутри больших помещений.

Библиотеки, реализующие алгоритмы SLAM, на сегодняшний день доступны

для множества языков программирования, а также в специализированной операционной системе для роботов ROS. Реализации различных алгоритмов SLAM с использованием различных языков программирования и программных сред, а также наборы данных, полученные с реальных датчиков, можно найти в свободном доступе [25]. Реальные данные являются ценным ресурсом для оценки и сравнения с эталоном различных алгоритмов SLAM.

ЗАКЛЮЧЕНИЕ

В настоящее время, помимо представленных в статье базовых методов, используются усовершенствованные алгоритмы, позволяющие повысить точность и сократить требуемые вычислительные затраты. В последующих публикациях планируется рассмотреть различные особенности метода SLAM, широко применяемого в автономных мобильных робототехнических системах.

ЛИТЕРАТУРА

1. <http://robotosha.ru/robotics/how-it-works-driverless-car-google.html>
2. www.neatorobotics.com
3. Smith R., Cheesman P. On the representation of spatial uncertainty. *Int. J. Robot. Res.*, vol. 5, no. 4, pp. 56–68, 1987.
4. Durrant-Whyte H.F. Uncertain geometry in robotics. *IEEE Trans. Robot. Automat.*, vol. 4, no. 1, pp. 23–31, 1988.
5. Ayache N., Faugeras O. Building, registering, and fusing noisy visual maps. *Int. J. Robot. Res.*, vol. 7, no. 6, pp. 45–65, 1988.
6. Crowley J. World modeling and position estimation for a mobile robot using ultrasonic ranging. *Proc. IEEE Int. Conf. Robot. Automat.*, 1989, pp. 674–681.
7. Chatila R., Laumond J.P. Position referencing and consistent world modeling for mobile robots. *Proc. IEEE Int. Conf. Robot. Automat.*, 1985, pp. 138–143.
8. Smith R., Self M., Cheesman P. Estimating uncertain spatial relationships in robotics. *Autonomous Robot Vehicles*, I.J. Cox and G.T. Wilfson, Eds. New York: Springer-Verlag, pp. 167–193, 1990.
9. Renken W.D. Concurrent localization and map building for mobile robots using ultrasonic sensors. *Proc. IEEE Int. Workshop Intell. Robots Syst. (IROS)*, 1993.
10. Durrant-Whyte H., Rye D., Nebot E. Localization of automatic guided vehicles. *Robotics Research: The 7th International Symposium (ISRR'95)*, G. Giralt and G. Hirzinger, Eds. New York: Springer-Verlag, pp. 613–625, 1996.
11. Hollerbach J., Koditscheck D. Robotics Research, The Ninth International Symposium (ISRR'99). New York: Springer-Verlag, 2000.
12. Thrun S., Fox D., Burgard W. A probabilistic approach to concurrent mapping and localization for mobile robots. *Mach. Learning*, vol. 31, no. 1, pp. 29–53, 1998.
13. Thrun, S., Liu, Y., Koller, D., Ng, A., Durrant-Whyte, H. Simultaneous localization and mapping with sparse extended information filters. *Int. J. Robot. Res.*, vol. 23, no. 7–8, pp. 693–716, 2004.
14. Dissanayake G., Newman P., Durrant-Whyte H.F., Clark S., Csobra M. A solution to the simultaneous localization and mapping (SLAM) problem. *IEEE Trans. Robot. Automat.*, vol. 17, no. 3, pp. 229–241, 2001.
15. Montemerlo M., Thrun S., Koller D., Wegbreit, B. Fast-SLAM: A factored solution to the simultaneous localization and mapping problem. *Proc. AAAI Nat. Conf. Artif. Intell.*, 2002, pp. 593–598.
16. Doucet A. On sequential simulation-based methods for Bayesian filtering. Dept. Eng., Cambridge Univ., Tech. Rep., 1998.
17. Montemerlo V., Thrun S., Koller D., Wegbreit B. Fast-SLAM: A factored solution to the simultaneous localization and mapping problem. *Proc. AAAI Nat. Conf. Artif. Intell.*, 2002, pp. 593–598.
18. Montemerlo V., Thrun S., Koller D., Wegbreit B. Fast-SLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. *Proc. Int. Joint Conf. Artif. Intell.*, 2003, pp. 1151–1156.
19. Newman P.M., Leonard J.J., Neira J., Tardos J. Explore and return: Experimental validation of real time concurrent mapping and localization. *Proc. IEEE Int. Conf. Robot. Automat.*, 2002, pp. 1802–1809.
20. Guivant J.E., Nebot E.M. Optimization of the simultaneous localization and map-building algorithm for real-time implementation. *IEEE Trans. Robot. Automat.*, vol. 17, no. 3, pp. 242–257, 2001.
21. Deans M., Hebert M. Experimental comparison of techniques for localization and mapping using a bearing-only sensor. *Proc. Int. Symp. Experimental Robot.*, 2000, pp. 395–404.
22. Leonard J.J., Rikoski R.J., Newman P.M., Bosse M.C. Mapping partially observable features from multiple uncertain vantage points. *Int. J. Robot. Res.*, vol. 21, no. 10–11, pp. 943–975, 2002.
23. Konolig K. Large-scale map-making. *Proc. Nat. Conf. AI (AAAI)*, 2004, pp. 457–463.
24. www.ros.org
25. www.openslam.org