

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по технологической практике
аспиранта

Аспирант

направление 09.06.01

направленность 05.13.11



Филатов Антон Юрьевич

Руководитель д.т.н., проф



Экало Александр Владимирович

Санкт-Петербург

2017

ЗАДАНИЕ НА ТЕХНОЛОГИЧЕСКУЮ ПРАКТИКУ АСПИРАНТА

Аспирант Филатов Антон Юрьевич

направление 09.06.01

специальность 05.13.11

Тема практики: технологическая практика аспиранта

Задание на практику:

Выявить отличительные черты в современных алгоритмах одновременного построения карты и определения собственного положения мобильных роботов, а также определить устойчиво работающий алгоритм для максимально вариативных входных данных

Сроки прохождения практики: 01.09.2017 – 10.01.2017

Дата сдачи отчета: 11.01.2018

Дата защиты отчета: 11.01.2018

Аспирант



Филатов А. Ю.

Руководитель



Экало А. В.

АННОТАЦИЯ

Одной из ключевых проблем, препятствующих созданию действительно автономного мобильного робота, является одновременная проблема локализации и определения собственного положения (SLAM). Решение должно оценивать позицию робота и одновременно создавать карту неизвестной среды. Несмотря на существование разных алгоритмов, которые пытаются решить проблему, универсальный пока не нигде предлагается. Лазерный дальномер является широко распространенным датчиком для мобильных платформ, и было решено оценить фактические алгоритмы SLAM на основе использования 2D-сканера. Были рассмотрены следующие алгоритмы: Google Cartographer, GMapping, tinySLAM. Согласно их оценке, Cartographer и GMapping более точны, чем tinySLAM, а Cartographer - самый надежный из алгоритмов.

SUMMARY

One of the key issues that prevents creation of a truly autonomous mobile robot is the simultaneous localization and mapping (SLAM) problem. A solution is supposed to estimate a robot pose and to build a map of an unknown environment simultaneously. Despite existence of different algorithms that try to solve the problem, the universal one has not been proposed yet. A laser rangefinder is a widespread sensor for mobile platforms and it was decided to evaluate actual 2D laser scan based SLAM algorithms on real world indoor environments. The following algorithms were considered: Google Cartographer, GMapping, tinySLAM. According to their evaluation, Cartographer and GMapping are more accurate than tinySLAM and Cartographer is the most robust of the algorithms.

СОДЕРЖАНИЕ

	Введение	5
1.	Описание работы алгоритмов	7
2.	Сравнение	9
	Заключение	12
	Список использованных источников	13

ВВЕДЕНИЕ

Одновременная локализация и сопоставление (SLAM) - это проблема построения карты и в то же время локализации робота внутри этой карты без предварительного знания карты и положения. Существуют различные подходы SLAM, основанные на данных от разных типов датчиков. Популярные сенсоры для мобильных платформ это:

- лазерные сканы - набор диапазонов от платформы до препятствий.
- одометрия - информация о перемещении платформы;

Первый тип входных данных представляет собой 2D сверху вниз среду, которая предоставляет меньше информации, чем трехмерный вид. Поэтому обычно требуется меньшее количество вычислений, которое больше подходит для недорогих мобильных платформ, где ограничены вычислительные ресурсы.

Проблема SLAM хорошо определена, но алгоритм, который решает ее для произвольной среды, пока не предлагается. Предполагается, что некоторые внедренные подходы используются в наружной среде, некоторые другие - в помещениях, где преобладают прямые линии.

Несмотря на различия в 2D алгоритмах SLAM, они используют общие идеи и структуры данных. Информация об окружающей среде хранится как сетка определенности, которая представляет собой двумерный массив идентичных единиц, которые по меньшей мере сохраняют свою вероятность быть занятыми. Положение платформы (поза робота) состоит из координат и ориентации. Комбинация поза робота и соответствующей карты - это мировое состояние. Алгоритмы обычно используют компонент согласования сканирования, который корректирует позу робота, оцениваемую с помощью необработанной одометрии, путем сопоставления данных лазерного сканирования с предыдущими сканированиями. Средство проверки

сканирования может быть реализовано несколькими способами: упорядочивание точек сканирования, поиск наилучшей корреляции между полным сканированием и картой и т.д.

Мировое состояние можно рассматривать как случайную переменную с неизвестной функцией плотности вероятности (PDF), которая обновляется при появлении новых данных датчиков. Простое одиночное отслеживание гипотез подразумевает оценку и обновление только наиболее вероятного мирового состояния. Примерами такого подхода являются алгоритмы `tinySLAM`, `L-SLAM`, `FastSLAM`. Этот метод не является общим, так как среда может содержать шаблоны, которые приводят к PDF с несколькими максимумами. Если карта хранится в виде сетки, позиция сканирования, уже вставленная в нее, не может быть изменена впоследствии.

Цель этой работы - сравнить три алгоритма SLAM: улучшенный `tinySLAM`, оригинальный `GMapping` и новый `Google Cartographer`. Они были выбраны в качестве воплощений описанных выше идей. Кроме того, пределы применения `Google Cartographer` оцениваются с использованием набора входных данных, не упомянутого в статье, которая его описывает.

1. ОПИСАНИЕ РАБОТЫ АЛГОРИТМОВ

Как уже упоминалось выше, `tinySLAM` отслеживает одну гипотезу о реальном мире. Метод основан на прямом цикле коррекции предсказания, поэтому он имеет компактную и простую реализацию. Карта хранится как сетка ячеек. Каждая ячейка сохраняет вероятность быть занятой для части среды, которой она соответствует.

Поза уточняется с помощью набора для сканирования на основе Монте-Карло. Он пытается добавить нормально распределенные (нулевые средние, настраиваемые дисперсии) значения для каждого компонента позы, чтобы найти новую позу, которая ведет к лучшему соответствию между картой и последним полученным сканированием. Соответствие оценивается как сумма занятости ячеек, содержащих точки лазерного сканирования.

Несколько эвристик реализованы `tinySLAM`, так как сопоставление сканирования недостаточно устойчиво. Размытие занятых частей карты увеличивает вероятность появления ячеек в окрестности точки сканирования. Исходная реализация также интерполирует данные лазерного сканирования для увеличения количества показаний диапазона и использует результат для сопоставления сканирования.

Другая эвристика предлагается в другой статье. Он добавляет лазерные сканы к карте с более низким значением «качества», если поза робота была уточнена соглашением сканирования. Измененная модель ячейки, введенная в документе, возвращает среднее значение вероятности заполнения последней ячейки.

GMapping основан на фильтре Rao-Blackwellized частиц (RBPF), который оценивает заднюю вероятность относительно потенциальных траекторий робота с использованием наблюдений и данных одометрии. Задворный аппроксимируется множеством точек (частиц) с соответствующими

вероятностями (весами). Частица с максимальным весом рассматривается как реальное мировое состояние. Вес частицы обновляется мерой соответствия между новым сканированием и картой, оцененной с помощью проверки соответствия. Модель ячейки сетки, используемая GMapping, сохраняет средние значения занятости и положения всех препятствий, обнаруженных лазерным дальномером в этой ячейке.

GMapping использует своего рода метод градиентного спуска для соответствия сканированиям. На каждой итерации проверяется несколько predetermined направлений. В качестве начальной позиции для следующей итерации выбирается позиция вдоль направления с максимальной совпадающей оценкой.

Cartographer Google хранит карту окружения в виде графа, где каждая вершина представляет карту и скан, полученные после создания соответствующего подмапа. Ребра представляют собой преобразования между соответствующими субмапами. Таким образом, шаг оптимизации, чтобы сделать карту последовательной, помимо процесса сопоставления сканирования.

Основная идея подхода при сопоставлении сканирования заключается в минимизации функциональности затрат. Таким образом, процесс согласования сканирования проходит через минимизацию некоторого функционала. Эта минимизация представлена как подход грубой силы с изменением ветвей и границ. Чтобы этот процесс работал быстро, используется библиотека Ceres.

Проблема оптимизации также решена с помощью подхода грубой силы. Требуется искать не только одно значение, но и множество всех подходящих преобразований между вершинами. Библиотека Ceres также используется для обеспечения этих оптимизаций. Эти оптимизации занимают много времени, поэтому эта работа выполняется каждые несколько секунд параллельно и исправляет результаты, которые уже зарегистрированы.

2. СРАВНЕНИЕ

Точность алгоритма SLAM может быть оценена путем сравнения выходной траектории с наземной истиной. Это сравнение должно проводиться на наборах данных, собранных в реальных условиях, чтобы оценить применимость метода SLAM.

Набор данных Центра MIT Stata - это один из нескольких наборов данных лазерного сканирования, который обеспечивает наземные траектории истины. Их можно извлечь, выполнив метод локализации на плане этажа с набором данных. Он содержит 84 последовательности, но некоторые из них являются многоэтажными (в отличие от Cartographer, как `tinySLAM`, так и `GMapping` не должны работать в таких средах), а некоторые последовательности не снабжаются соответствующими планами этажей. Таким образом, было выбрано 11 последовательностей для тестирования рассмотренных алгоритмов SLAM.

Набор данных Willow Garage не предоставляет наземных траекторий истины, поэтому точность алгоритмов не может быть оценена количественно. В этом случае алгоритмы SLAM можно качественно сравнить путем визуального сопоставления выходных карт. Авторы Cartographer используют этот набор данных для демонстрации эффективности алгоритма. Таким образом, одна тестовая последовательность была выбрана для тестирования `tinySLAM` и `GMapping`. Насколько нам известно, результаты оценки этого набора данных не являются общедоступными для этих алгоритмов.

Поскольку все алгоритмы являются вероятностными, их выход может варьироваться от одного прогона к другому. Таким образом, среднеквадратичная ошибка (RMSE) между выходными и наземными траекториями истинной правды должна вычисляться для оценки точности алгоритмов SLAM. Эта оценка была выполнена в наборе данных Центра MIT Stata. Траектории сравнивались с инструментом оценки SLAM TUM. Все алгоритмы были протестированы «как есть», т.е. Ни один из параметров алгоритмов не был изменен для определенной последовательности.

Средние и дисперсионные значения RMSE в течение нескольких прогонов представлены в таблице, а на рисунке представлены значения для некоторых последовательностей. Таблица 1 содержит два столбца для Cartographer, поскольку он может выполнять окончательную оптимизацию траектории робота после обработки всей последовательности данных. Столбец `` Cartographer (offline) " содержит значения RMSE после окончательной оптимизации. Столбец `` Cartographer (онлайн) " показывает RMSE данных траектории, собранных во время выполнения. Однако Cartographer не удалось оптимизировать траектории для некоторых последовательностей данных (знак «---» в таблице). Это может быть вызвано ошибками в проверенной реализации алгоритма.

Таблица 1. Сравнение алгоритмов SLAM

Sequence	Length, m	Trajectory RMSE, m			
		GMapping	tinySLAM	Cartographer (online)	Cartographer (offline)
2011-01-19-07-49-38	68	0.216 ± 0.012	1.280 ± 0.640	0.188 ± 0.023	0.191 ± 0.001
2011-01-20-07-18-45	76	0.219 ± 0.012	0.254 ± 0.045	0.219 ± 0.002	0.221 ± 0.001
2011-01-21-09-01-36	87	0.212 ± 0.029	0.242 ± 0.005	0.217 ± 0.003	0.205 ± 0.001
2011-01-24-06-18-27	87	0.290 ± 0.035	0.254 ± 0.006	0.217 ± 0.001	0.217 ± 0.001
2011-01-25-06-29-26	109	0.208 ± 0.008	0.260 ± 0.005	0.232 ± 0.001	0.232 ± 0.002
2011-01-27-07-49-54	94	0.266 ± 0.012	0.620 ± 0.030	0.266 ± 0.004	—
2011-01-28-06-37-23	145	2.388 ± 1.949	2.280 ± 0.750	0.360 ± 0.069	0.354 ± 0.004
2011-03-11-06-48-23	245	0.365 ± 0.208	0.860 ± 0.390	1.152 ± 0.601	1.348 ± 0.001
2011-03-18-06-22-35	80	0.145 ± 0.023	0.103 ± 0.008	0.145 ± 0.021	—
2011-04-06-07-04-17	95	0.190 ± 0.002	0.343 ± 0.025	0.201 ± 0.002	—
2011-10-20-11-38-39	264	0.352 ± 0.003	5.486 ± 2.603	2.217 ± 0.021	—

Результаты показывают, что, вообще говоря, GMapping и Cartographer имеют сравнимую точность на коротких траекториях (менее 100 м). В то же время Картограф имеет более низкую дисперсию результатов, т. Е. Его оуттут более предсказуем. Однако Cartographer может потерпеть неудачу на последовательностях с длинными безликами залами, например, в последовательности `` 2011-03-11-06-48-23 ". Множественное отслеживание гипотез, реализованное в GMapping, обычно дает гораздо более точные результаты на входе такого рода. Метод tinySLAM также не смог построить

согласованную карту на большинстве таких последовательностей, но иногда он имеет лучшее качество вывода, чем Cartographer. Успешное завершение окончательной оптимизации траектории в алгоритме Картографа оказывает небольшое влияние на значение RMSE, но оно значительно уменьшает дисперсию, которая может быть полезна, если качество карты окружения более важно, чем точность траектории.

Качество карт, построенных по рассматриваемым алгоритмам, оценивалось в последовательности ``2011-08-04-23-46-28 '' из набора данных Willow Garage. Карты и траектории, представленные на рис. , являются лучшими за несколько прогонов алгоритмов. Согласно рисунку, все проверенные алгоритмы способны построить достаточно согласованную карту, но карта, построенная tinySLAM, имеет больше артефактов, чем другие. Стоит отметить, что tinySLAM потребовало наибольшее количество прогонов, чтобы получить согласованную карту, что означает, что она менее надежна. Тем не менее, все траектории вывода достаточно близки друг к другу.

ЗАКЛЮЧЕНИЕ

Сравнивались три реализации популярных алгоритмов SLAM на основе 2D лазерного сканирования: `tinySLAM`, `GMapping` и `Cartographer`. Оценка проводилась на двух наборах данных, собранных в реальных условиях в помещении. Алгоритмы выполнялись с использованием конфигураций по умолчанию, хотя изменение некоторых параметров могло повысить точность, но это противоречит идее универсального (произвольно применимого) метода SLAM.

Было показано, что `tinySLAM` обеспечивает наибольшую среднеквадратичную ошибку при длинных последовательностях данных, поскольку она отслеживает только одну гипотезу и накапливает ошибки во время оценки. Из-за своей стохастической природы `tinySLAM` может неправильно оценивать состояние мира, поэтому это приводит к ошибкам на карте, которая не может быть исправлена. Как и ожидалось, `GMapping` обеспечивает меньшую ошибку, чем `tinySLAM`, поскольку он одновременно поддерживает множественную гипотезу о мировом состоянии и использует нестационарный сканер.

`Cartographer` и `GMapping` имеют сопоставимые значения RMSE для большинства входных последовательностей. Картограф обычно имеет более низкое значение дисперсии ошибки, поэтому он более устойчив. Но он менее точен для других последовательностей, поэтому проверенная реализация не является универсальной.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. W. Burgard, C. Stachniss, G. Grisetti, B. Steder, R. Kummerle, C. Dornhege, M. Ruhnke, A. Kleiner, and J. D. Tardes. A comparison of slam algorithms based on a graph of relations. In 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 2089–2095, Oct 2009.
2. C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. IEEE Transactions on Robotics, 32(6):1309–1332, Dec 2016.
3. M. Fallon, H. Johannsson, M. Kaess, and J. J. Leonard. The mit stata center dataset. The International Journal of Robotics Research, 32(14):1695–1699, 2013.
4. J. Funke and T. Pietzsch. A framework for evaluating visual slam. In Proceedings of the British Machine Vision Conference (BMVC), volume 6, 2009.
5. B. Gerkey. Ros slam gmapping. http://wiki.ros.org/slam_gmapping, 2010. [Accessed 31-July-2017].
6. Google. 2d cartographer backpack deutsches museum, 2016. [Accessed 31-July-2017].
7. C. Harris and M. Stephens. A combined corner and edge detector. In Alvey vision conference, volume 15, pages 10–5244. Manchester, UK, 1988.
8. W. Hess, D. Kohler, H. Rapp, and D. Andor. Real-time loop closure in 2d lidar slam. In Robotics and Automation (ICRA), 2016 IEEE International Conference on, pages 1271–1278. IEEE, 2016.