

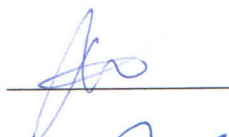
МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по научно-исследовательской деятельности во втором семестре

Аспирант

направление 09.06.01

направленность 05.13.11



Филатов Антон Юрьевич

Руководитель д.т.н., проф



Экало Александр Владимирович

Санкт-Петербург

2017

Цель работы

Провести сравнение существующих алгоритмов многоагентного SLAM и составить описание алгоритма, который бы подходил для стаи мобильных роботов, работающих в помещении

Введение

Задачу одноагентного SLAM (Simultaneous Localization And Mapping) нельзя считать окончательно решённой, не смотря на то, что существуют алгоритмы, работающие с хорошей точностью на некоторых входных данных. Сложность всегда в том, что если во время работы алгоритма происходит сбой, его практически невозможно идентифицировать, не имея некоторого арбитра извне. Решить эту проблему призван многоагентный SLAM, где каждый агент является арбитром для других агентов

Описание проделанной работы

Главным недостатком одноагентного SLAM является его неточность. Ещё не было предложено ни одного алгоритма, который решал бы задачу SLAM на любых входных данных с любой требуемой точностью. Погрешности обусловлены множеством различных факторов, главный из которых — ошибки в процессе работы скан матчера. Поскольку задача скан матчера — выяснить наиболее точное положение робота — не решается обычно методом полного перебора, то невозможно считать, что на каждом шаге алгоритма она выполняется абсолютно точно. А проверить и сопоставить построенную карту на каком-то этапе с некоторым эталоном невозможно. Поэтому построенные карты могут быть приблизительно точными, но они всегда содержат некоторую погрешность.

Решить эту проблему может использование нескольких роботов одновременно. Идеальная работа такой системы выглядит так: в помещение запускают стаю роботов, каждый из которых движется независимо, они обмениваются полученными данными и через некоторое время представляют карту местности и траектории, по которым они двигались.

На этапе планирования такой системы возникают несколько возможных подходов к её реализации.

- 1) Один сервер, который управляет взаимодействием всех роботов и хранит карту местности. По сути именно сервер выполняет все задачи, входящие в состав SLAM по обработке входных данных от роботов, наложению их на построенную карту, корректировка положения роботов. Роботы в этом случае выполняют лишь роль носителей сенсоров. Достоинством такого подхода является дешевизна агентов. При таком подходе от них не требуется никаких вычислений, они должны лишь снимать данные вокруг себя и своевременно посылать их на сервер. Недостатком такого подхода является то, что у сервера нет арбитра, который мог бы проверить его работу. Такая система работает как одноагентный SLAM, только с большим количеством сенсоров

2) Несколько роботов, каждый из которых решает задачу SLAM самостоятельно, при этом они обмениваются картами друг с другом во время встречи. Такой подход является полной противоположностью тому, который был описан выше. В такой системе вовсе нет сервера, который мог бы выполнять сложные математические операции, освободив тем самым агентов. В таком подходе агенты должны выполнять задачу одноагентного SLAM самостоятельно. При этом во время встречи они могут обмениваться построенными картами, причём каждый из них может воспринимать карту второго агента лишь как новые наблюдения. Таким образом, карты, построенные всеми агентами могут различаться в конце работы алгоритма. Это и является недостатком такого подхода. Если какой-то алгоритм «сломался» во время выполнения, то необходимо будет вручную определять такую работу и не учитывать её при построении итоговой карты. Кроме того задача представления карты в виде входных наблюдений для другого агента — это исследовательская задача, которая пока решена только при помощи визуальной детекции агентами друг друга, а этот способ не является очень точным.

3) Лишён указанных выше недостатков графовый метод многоагентного SLAM. Во время работы каждый робот строит граф наблюдений, где каждая вершина графа — это наблюдение, а ребро — это трансформация между наблюдениями. Каждый агент владеет собственным графом, дополняет его и обновляет. Преимущество представления карты в виде графа в том, что когда робот попадает в место, в котором он уже был, он может определить, замыкается ли граф в этом месте, и если нет, то изменить рёбра графа так, чтобы замыкание произошло. Таким образом графовый SLAM может обновлять историю построения карты. Такой механизм можно использовать и в многоагентном SLAM. Как только один из агентов попадает в некоторое место, где уже побывал второй агент, они могут воспользоваться механизмом обновления графа, используя карты, построенные обоими агентами. Это позволит агентам являться арбитрами друг для друга, а также гарантирует, что карты, построенные этими агентами, будут одинаковыми. Недостатком такого

подхода является алгоритмическая сложность. Каждому агенту необходимо выполнять трудоёмкий графовый алгоритм.

Кроме того существуют различные вариации описанных выше алгоритмов. Например, можно воспользоваться хранением нескольких гипотез о положении робота, или использовать сервер лишь как средство передачи информации между агентами, чтобы облегчить нагрузку на их вычислительные модули.

На основании проделанного исследования можно получить алгоритм, который совмещает в себе преимущества всех описанных выше подходов. Общая схема выглядит так. В системе есть сервер, который получает карты от всех агентов в сети. При этом каждый агент строит карту самостоятельно независимо от всех остальных. Карты в таком подходе представляются в виде графов. Задачей сервера является обработка карт от агентов и их слияние, используя алгоритмы оптимизации графа, которые используют и сами агенты при построении карт. При этом пересечение агентов может учитываться как особые точки, в которых необходимо произвести оптимизацию графа.

Заключение

В ходе исследования были изучены существующие алгоритмы, решающие задачу многоагентного SLAM. В качестве алгоритма, который может быть реализован и который совмещает в себе достоинства всех алгоритмов был выбран графовый алгоритм с сервером, занимающимся слиянием карт, построенных агентами. Взаимодействие агентов происходит через сервер, который также занимается обновлением их карт. Траектории движения всех роботов хранятся на каждом агенте, а также дублируются в общей карте, которая строится на сервере.

Список литературы

1. Sebastian Thrun, Wolfram Burgard, Dieter Fox. Probabilistic Robotics / The MIT Press. USA. С. 672
2. Tian, T.; Tomasi, C.; Heeger, D. Comparison of Approaches to Egomotion Computation. IEEE Computer Society Conference on Computer Vision and Pattern Recognition: 315, 2013. С. 221-246.
3. Родионов, С.А. Обработка результатов измерения дисторсии проекционных объективов/С.А. Родионов, Н.Б. Вознесенский, Э.М. Шекольян// Известия ВУЗов СССР Приборостроение. 1991. — Т.34, No7. - С.61-68.
4. GR Bradski, "Computer vision face tracking as a component of a perceptual user interface[C]"/ IEEE Workshop Application of Computer Vision, С. 214-219.
5. E. Rosten, T. Drummond, "Machinelearning for high speed corner detection," / 9th Eupropean Conference on Computer Vision, vol. 1, 2006, pp. 430–443.