

Извлечение топологической информации из сетки карты для навигации робота

Дэвид Португал и Руи П. Роча

Институт систем и робототехники Университета Коимбры, 3030-290 Коимбра, Португалия
{davidbsp, rprocha} @ isr.uc.pt

Ключевые слова: Работа навигации, графики, топологические карты, диаграммы Вороного.

Аннотация: Во многих работах задачи связанные с навигацией абстрагируются под среду, в которой мобильные роботы выполняют некоторые миссии, поэтому такие задачи могут быть весьма полезными. В частности, извлечение простого топологического графа, как представление более сложной и детализированной карты, часто требуется для пути планирования и навигации пути. В данной работе представлен подход для выполнения таких задач в сетке карты. В центре внимания не только получение диаграммы или визуального представления возможных путей, но и предложен новый способ получения графа информации данных, относящейся к связи окружающей среды, которые могут быть переданы в роботов или для централизованного планирования, с тем чтобы помочь задачам навигации. Предлагаемый подход основан на методах обработки изображений. Эти результаты моделирования доказывают простоту, точность и эффективность.

1 ВВЕДЕНИЕ

В связи с робототехникой, изучения и сохранения модели окружающей среды является очень важным для автономных мобильных роботов. Без внутренней описания окружающей среды, а также информации о своей позиции и ориентации по отношению к этой карте, большинство задач мобильных роботов, как вождение, избегая при этом столкновений и навигации, станет нецелесообразным.

В навигационных задач, она, как правило, предполагается, что среда известна как априори. С другой стороны, исследование, как правило, связано с полностью или частично неизвестными условиями. В обоих случаях, поддержания или создания внутреннего представления инфраструктуры является одним из ключевых вопросов для успешного выполнения задачи.

Две основных различных парадигм для отображения среды на основе сетки карты и топологический карт (Thrun, 1998). В настоящей работе мы ориентируемся на последней.

Grid-методов, основанных на производство точных метрических карт, их легко построить, представлять и поддерживать, но часто страдают от памяти и временной сложности, потому что ее прекрасная ограниченная резолуция, которая приводит к проблемам памяти, а также затрудняет эффективное планирование и навигацию в больших масштабах инфраструктуры. Топологические карты, с другой стороны, производят графики, как карты, которые могут быть использованы более эффективно. Они намного проще и эффективно позволяют планировать и не требуют точного определения положения робота. В этих картах, вершины соответствуют важным местам, памятников архитектуры, которые соединены ребрами, которые представляют собой пути между ними. Тем не менее, его неточность делает это тяжелее для обеспечения согласованности в больших масштабах среды, что приводит к трудностям в признании подобных мест, которые похожи друг на друга. Есть и другие ограничения, связанные с навигация с помощью топологической карты. Некоторые ограничения были выявлены ранее (Циммер и др., 1994) :

обработка неточной позиции и ориентации информации и обнаружения соседней вершины в топологической карте, через них, в отличие от чувствительных или признанных ими. Несмотря на эти ограничения, метод планирования пути, который был использован производит адекватные результаты.

Кроме того, проверка карт и автолокализация в топологической карте также является важным вопросом для правильной навигации робота, который был широко использовано для самодельных взрывных устройств (Дудек и соавт., 1997). Как вы уже поняли, топологические карты могут быть очень важным инструментом для большинства задач с помощью мобильных роботов. Многие произведения использовать эти простые представления используют правильную навигации робота. В Феррейра (Ferreira и соавт., 2007), робот ездит по окружающей среде и самостоятельно локализуется, а с помощью метода место назначения создаются 3D наборы помутнения для мониторинга изменений, которые могут иметь место в окружающей среде.

В данной работе внимание в основном направлено на получение глобальных топологических абстрагирований от существующих метрических представлении среды. Например, для наблюдения, мониторинга и патрулирования задач несколько роботов, обычно полагаются на топологическую карту по вопросам навигации. В предыдущей работе (Португалия и Роча, 2010), мы представили новых мульти-роботов как алгоритм патрулирования, основанный на топологических картах.

В этой статье мы подробно описываем, как полная топологическая информация извлекается из существующих 2D-карт представлением области сетки, которая будет патрулировать, которая, в свою очередь, может быть получена государством самые современные робото-технические отображения, например, (Трун и соавт., 2000).

Следующий раздел представляет собой обзор предыдущих методов для извлечения топологических представлений, как диаграмм и графиков из метрического представления данного априори. Как правило, эти метрики представления размещение сетей, которые являются вероятностные карты, для которых каждая ячейка сетки содержит вероятностные значения, указывающее на соответствующие места свободного пространства или часть препятствий (Эльфы, 1989). Эти решетки, как правило, получены на предыдущем этапе исследования. В разделе III формулируется проблема, которую предстоит решать и в разделе IV ,приводится алгоритм, предложенный для ее решения. Позднее, будут представлены результаты и статья заканчивается выводами и будущей работой.

2 Связанные РАБОТЫ

В литературе есть несколько существующих методов извлечения топологических представлений на основе метрических карт. В частности, диаграммы Вороного (Вороной, 1908) были широко использованы для планирования пути, который остается в стороне от препятствия, насколько это возможно. Кроме того, в обобщенной диаграммы Вороного (ДГС) (Wallgru, 2004) был описан как «втягивание свободного рабочего места пространства на сети одномерных кривых, отражающих связь свободного пространства» и иерархически организованных на основе маршрута графического представление для навигации роботов в помещениях сценариев, был предложен Вороновым.

Кроме того, расширенный график Вороного (ГО) (Beeson и соавт., 2005) был представлен совсем недавно. В этом новом подходе, диаграмма (или скелет) расчетом преодолевает ограничения сенсорными горизонтными роботами, четко определив переход от коридора следования стены в больших залах. Хотя и граф, в фактической схеме нет информации о вершинах или ребрах, представить можно после окончательного расчета.

Кроме того, Колинг и Карпин (Колин и Карпин, 2008) предложили метод для извлечения графического представления в размещении сетки для наблюдения задач, основанных на ДГС. График изменения, называется подрядчиков, которые способствовали упростить задачу наблюдения в направлении ее решения с использованием меньшего количества роботов.

Другой часто используемым методом при извлечении графиков триангуляции Делоне (DT) (Delaunay, 1934). В самом деле, DT из дискретного множества точек в целом соответствует двойственный граф диаграммы Вороного по той же заданной точке.

Катсилиерис соавт. (Катсилиерис и соавт., 2010) извлечь граф из исходной метрике карты в контексте поиска злоумышленников с командой роботов. Во-первых, беспрепятственной области триангуляции и треугольники объединены в большие выпуклые области, где вершины извлечены. Множество ребер состоит из всех пар регионов, которые разделяют стороны.

Кроме того, метод, называемый Снижение ограничением триангуляции Делоне на построения графиков также становится популярным. Этот метод основан на вычислении DT вместе с механизмом снижения графа минимизации расстояния которые каждый робот должен пройти. Он был применен Фазли и соавт. (Фазли и соавт., 2010) в рамках мульти-роботов, зона охвата задач в известных и статических 2D-среде.

Есть другой метод в литературе, чтобы извлечь график среды видимости графа (Лозано; и Уэсли, 1979), который состоит из прямых линий, соединяющих последовательность вершинами, которые, как правило, размещены вблизи границ препятствия в окружающей среде (с учетом размера и позы робота). График создается подключение всех вершин, которые могут «видеть» друг друга, поэтому график называется видимость.

Среди наиболее часто используемых методов является методы истончение (Ко и соавт., 2004) и скелетирование (Сабо, 2004). Эти операции используются для удаления переднего плана пикселей от бинарных изображений, аналогично пилинг лук. Они представляют собой каркас изображения, уменьшая все линии в один пиксель толщиной. Результатом таких методов похож на ДГС. Сабо (Сабо, 2004) также обращается в своей работе некоторые другие методы для извлечения карты из топологических размещение сетей, а именно, редкие подходы пикселей и согласования противоположных контуров. Кроме того, еще один метод, называемый C-клетки можно найти в литературе (Мачадо, 2002). Она имеет принципиальное разделение пространства не занятое препятствия на выпуклые полигоны и вершины графа расположены в самом центре этих многоугольников, имеющих каждая вершина связана по прямой линии до ближайшей видимой вершине.

Большинство предыдущих работ, упомянутые выше, представляют современные методы визуального извлечения топологического представление. Тем не менее, многие из них не правильно охарактеризовывают вершины и ребера на графике (например, диаграмм Вороного, EVG, истончение или скелетирование). Что же касается тех, чьи методы, которые на основании триангуляции Делоне нужен ручной определение вершин в среде для расчета топологии и обычно порождают не элегантный и несбалансированный представлениями. В случае видимость графика,в очень неадекватный топологии выпускаются с вершины расположены в непосредственной близости от препятствий, что неестественно для роботов навигации. Обобщенный график Вороного (GVG), который состоит из вершин для каждой встречи и конечной точки ДГС и края отражающего связность ДГС, приводит к очень сложной графики с большим числом вершин и ребер, которые в основном избыточными, особенно при рассмотрении задач навигации для роботов. Кроме того, все эти методы предполагают, что ребера, соединяющих пары соседних вершин должна быть прямой, не всегда верно для роботов с возможностью локально планировать пути между двумя вершинами.

В этой работе мы намерены идти вперед, на первую извлечения элегантный топологическое представление, а затем особым упором на выявление вершин, ребер и их длину (т.е. их вес) и получения соответствующих данных о вершинах и связи между областями на карте, с тем чтобы помочь роботов навигации в задачи, такие как патрулирование, освещение, контроль, надзор, сближения-уклонения и другие.

3 ПОСТАНОВКА ЗАДАЧИ

В этой работе, охраны окружающей среды предполагается априори известно, и метрические представления, как правило, с формой занятости сетки, имеется. Цель состоит в том, чтобы абстрагироваться от окружающей среды за счет топологических репрезентации, т.е. график, и получить всю информацию о связи графа.

Методы, описанные в предыдущем разделе, приведет к различным представлениям графиков, как и размещение вершин вблизи препятствия, поместив их на полпути между препятствиями или помещая их в единый путь по пересеченной местности. Имея это в виду, тщательный выбор должен быть сделан с учетом приложений, в которых добыча не применяется.

Начнем с представления скелета, а затем сосредоточиться на извлечении топологической информации для характеристики связи окружающей среды. Есть несколько методов для расчета начального скелета, как показано ранее в разделе II. В этой работе он получил через EVG вычисления, в основном из-за ее результатов с точки зрения производства визуально привлекательный путь представления, держаться подальше от препятствий и которые считают, чувствительный диапазон роботов в качестве входного метода. Кроме того, это относительно новый метод, который имеет открытый исходный код и легко адаптируется к коду, который был разработан. Имея представление скелета, график получается через методы обработки изображений правильной идентификации вершин и ребер. Топологической карты моделируется как неориентированный граф. Вершины представляют собой места и ребра представляют собой связи (в обоих направлениях) между этими местами.

В следующем разделе мы представляем алгоритм для извлечения полной топологической информации и ее простоты и эффективности становится ясно.

4 Алгоритм

В предыдущей работе (Португалия и Роча, 2010) патрулирование тренажер используется для проверки нескольких роботов патрулирование подхода, основанного на сбалансированном разбиении графа. Здесь мы покажем, как заранее предполагаемый график предоставляется патрулирование симулятор. Алгоритм состоит из 4 шагов, которые описаны ниже.

4.1 Приобретение Скелет

Инструмент под названием ГО-THIN разработанный Патриком Би-сын (Beeson и соавт., 2005) был использован для приобретения скелета из сетки занятости в этом проекте. «EVG» означает расширенный график Вороного, который, хотя и называют граф, не выводят, связанные с информацией о вершин и ребер в графе, давая вместо расширения скелет в верхней части сетки карты, по сравнению с ДГС. «Тонкие» составляет растровых истончение алгоритм, который находит скелеты растровых изображений, которая является быстрым приближением диаграммы Вороного. Его код был написан для применения в режиме реального времени на размещение сетей, где клетки либо заняты, бесплатно, или неизвестно, но он также работает на любых

оттенках серого растрового изображения в других областях. Это средство командной строки приложения в любой Linux-консоли на определение некоторых параметров, таких как порог пикселей для рассмотрения их на свободу, неизвестно или занята, или минимальное расстояние робота до препятствия для учета геометрический размер робота и срабатывания при навигации на карте. Настройка этих параметров правильно важно, потому что скелет компьютерной топологии и окончательное существенно зависит от этих параметров. Интерфейс Патрулирование Simulator отображается окно, чтобы установить все входные параметры повторного использования и развивается EVG тонкий в коде, который выполняется в бэк-ground1, получение 1 пиксель тонкий скелет окружающей среды, как показано на рисунке 1.



Рисунок 1: EVG-THIN включены в симуляторе.

4.2 Кластера изображения. Удаление

Из-за неровностей, иногда скелет представление, полученных в предыдущем разделе представлены кластерами, т. е. множеств из 4 квадратов пикселей, как показано на рисунке 2. Эти кластеры являются проблематичными, в том смысле, что они вмешиваются в последующем обнаружении вершин графа, в связи с 1 пиксель тонкий скелет предположение. Таким образом, промежуточный этап, чтобы избежать будущих повреждения данных при обнаружении вершин и ребер необходимо: Кластер удаления.



Рисунок 2: обнаружение и исправление кластера. Синие точки соответствуют вершинам впоследствии идентифицированы.

Защитный фильтр был разработан для обнаружения и снова движения таких кластеров. В основном, это проносится образ и когда кластер обнаружены два действия возможным \neg BLE:

- Pixel Удаление: Уберите ненужные пиксели из кластера;
- Pixel переход во: Перемещение пикселей окрестности кластера.

Эти действия осуществляются без ущерба для связности графа. На рисунке 3, на левой стороне, двумя примерами показано их применения. На вершине, 2 пикселем отмечен крестом будут удалены, так как они не являются необходимыми для обеспечения связи между всеми отраслях.

Когда кластер идентифицируется, фильтр проверяет последствия удаления пикселей, а точнее это гарантирует, что нет отключений в результате стирания пикселей будет существовать, путем проверки всех соседних пикселей по всему кластеру.

Стирание пикселей не всегда возможно, так как она может наблюдаться в нижнем левом рисунке 3, где удаление эффекта красных пикселей создаст пробелы в скелете под ним. Таким образом, переключение пикселей стратегии был создан для исправления такой ситуации.

4.3 Обнаружение Вершин

Наличие кластеров без скелета, следующим шагом является извлечение всех вершин графа обработки изображений, которая осуществляется в верхней части ранее созданных скелета. Процесс довольно прост, в связи с 1 пиксель тонкие предположения. Вершины соответствуют отдельных пикселей, которые размещаются в двух определенных местах:

- тупиков: пиксели, которые обычно есть только один сосед пикселей, который также входит в скелет. Они происходят с вершины степени.
- Филиал пересечения: обнаружение пикселей, что соответствует пересечения ветвей скелета. Они обычно имеют три или более соседних пикселей, которые также относятся к скелету, в восемь пикселей вокруг них. Они происходят выше вершины степени.

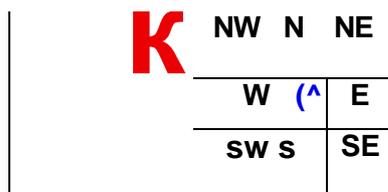


Рисунок 3: На левой стороне: Кластерный анализ - Pixel удаления сверху и переключение пикселей ниже. С правой стороны: возможные направления соседа.

Этот процесс показан на псевдо-код, алгоритм 1 проводится с целью выявления всех вершин графа. В принципе, цель заключается в проверке на 9 пикселей окно по центру проанализированы пикселей скелета и проверить, если это тупиковая ветвь или пересечения. Когда вершина определяется, один синий пиксель рисуется в правильном месте. Обратите внимание, что этот алгоритм основывается на том, что не существует кластеров на графике.

После выполнения алгоритма, визуальное представление, tation графа в верхней части сетки, так как оно будет показано на результатах раздела, в котором версии \neg tices были выделены для лучшей идентификации.

```

1 foreach pixel p of the image do
2   if p is red then
3     count ← number of red pixels inside
       9-pixel window centered on p;
4   else
5     count = 0;
6   end
7   if count > 3 then // Exclude pixels with 3 or
       more neighbors that do not correspond to
       branch intersections.
8     foreach two adjacent red pixels around p
       do
9       count = count -1;
10    end
11   foreach three adjacent red pixels
       around p do
12     count = count -2;
13   end
14   else if count = 3 then // Dead-end (with more
       than one neighbor pixel)
15     if two adjacent red pixels around p then
16       count = count +1;
17     end
18   end
19   if count > 3 ∨ count = 2 then
20     p is a vertex ⇒ blue pixel in p;
21   end
22 end

```

Алгоритм 1: псевдо-код вершины процесс обнаружения.

4.4 Извлечение Соответствующей Топологической информации

На последнем шаге алгоритма для вычисления остальных топологической информации, чтобы полностью охарактеризовать графике (края, связь и т.д.). В этой работе, каждая вершина структуры данных будут иметь последующей ING областях, связанных с ним:

- ID
- Координаты (x, y)
- Степень
- Соседи (Vertex степени) = {ID, направление края вес}



Рисунок 4: Извлечение графике и соответствующих топологической информации. Пример простого древовидного графа с 12 вершинами.

На этом этапе основной задачей является преобразование зрительной информации в графе данных, которые могут быть использованы роботы абстрактной среде.

ID и координатами (x, y) каждой вершины можно легко извлечь, глядя на синие пиксели на картинке и присвоения им удостоверения личности и сохранения их (x, y) координаты пикселей.

Остальные поля получены впоследствии, а также по обработке изображений, следуя всем отраслям оставляя вершины по маршрутам красные пиксели. Когда синий пиксель находится в конце пути, сосед вершины признается, поэтому степень начального пикселя увеличивается, и новую запись в таблице соседей начальной вершины заполнен: идентификатор вновь обнаружил сосед вершины, направления, связанные с начальной вершины и края весом от 2 вершины.

ID получается, касающиеся координат вновь обнаружены вершины ID списка ранее вычислен. Направление определяется аналогично компаса обозначения, как показано на правой части рисунка 3, например, если направление трассы, ведущей к вершине сосед начинается в пиксель выше обследованных вершин, чем направление определяется как север (N). Кроме того, край веса между вершинами определяется как стоимость путешествия из одной вершины к другой в пикселях, в частности, количество красных пикселей на краю между двумя вершинами.

Каждая вершина определяется один синий пиксель, как показано выше на рисунке 2. После получения всех полей данных, каждая вершина будет иметь таблицу соседей с размером его степень и все ребра и связи между соседними вершинами будет хорошо описана. Это видно на рисунке 4. Имея всю эту информацию, рамки для навигации агентов в Патрулирование симулятора, наконец будет создана.

5 РЕЗУЛЬТАТЫ И ОБСУЖДЕНИЕ

Алгоритм, который представлен в этой статье, была протестирована с помощью большого разнообразия карт, создавшего элегантно и сбалансированного представления, которые варьируются от простых топологий сложных. Важным аспектом является то, что сложности топологической карты извлечь не имеет никакого отношения к размеру окружающей среды (в декартовых расстояния). Алгоритм достигнуты четкие и визуально привлекательный график представления-ных с метрической карты любого размера и формы, не представляют вершины с самостоятельной петли, держаться подальше от препятствий, а также с учетом срабатывания из роботов, показывая, что он хорошо масштабируется с точки зрения окружающей среды, как показано на рисунках 5-8. За точный результат получен, алгоритм показал, что вычислительно эффективным. C++ язык программирования использовался и графический интерфейс пользователя (GUI) для патрулирования Simulator был реализован с помощью Qt Open Source Edition 4.4.3, которая также базируется на C++ Language3. Система работала на Ubuntu 8.10 Intrepid на AMD Athlon 64 3500+, 2.21GHz, 1GB RAM с.

Таблица 1 показывает время, с помощью алгоритма, чтобы извлечь все данные из различных метрических карт, который зависит не только от количества пикселей в зоне свободной каждой метрической карты, а также сложность созданного графом. Около 95% времени значения, показанные проводит вычисления скелета окружающей среды (первый шаг алгоритма), который несет ответственность за EVG-тонкий. Три другие шаги подход вычисляются в несколько сотых долей секунды, что очень быстро рассмотрении заявки, для которых он был разработан.

Пикселей изображения представлены в таблице не связаны таким же образом, в каждом конкретном случае, на свободный пикселей обнаружены в каждом изображении. Ясно также, что время вычислений зависит не только от этого. Он также зависит от размера графа, как, например, четвертая и пятая строка в таблице моментов.

Эти результаты свидетельствуют о применимости метода, предложенного в этом документе. Он является ценным методом для получения топологической карты окружающей среды либо априори известно метрических карта или карта метрических строятся на лету.

6 Выводы и будущая работа

В этой статье, новый способ извлечения топологических карт.



Рисунок 5: Результаты комплекс граф (268 вершин) с высоким подключения.

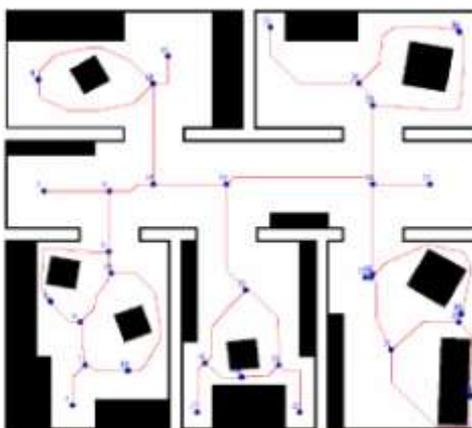


Рисунок 6: Результаты для средних граф (41 вершин) с определенной связи.

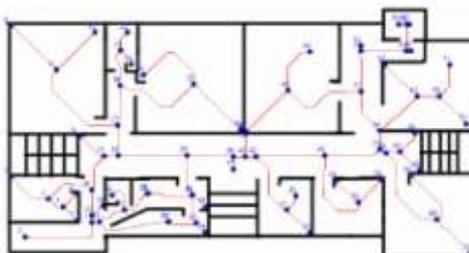


Рисунок 7: Результаты для средних граф (74 вершин) с определенной связи.



Рисунок 8: Результаты для графа с 70 вершинами и несколько ограничений подключения.

Таблица 1: Расчет времени для различных метрических карт.

| Image Pixels | # Vertices | Time (s) |
|--------------|------------|----------|
| 90K | 12 | 0.40 |
| 341K | 74 | 1.38 |
| 379K | 20 | 1.42 |
| 343K | 66 | 1.57 |
| 509K | 206 | 1.61 |
| 544K | 268 | 1.63 |
| 815K | 135 | 3.62 |

Подключения информации из стандартной сетки, описано как оттенки серого представлений. Подход, представленный прибыли выгоду от своей простоте, точности и производительности. Одним из возможных недостатков использования EVG тонкий метод для вычисления скелета сетки карты зависимость от правильной параметризации, которая не просто в большинстве случаев. Тем не менее, подход, изложенный в этой статье не ограничивается использованием EVG тонкий, чтобы извлечь скелет, другие методы, как упомянутые в разделе II, также может быть использован.

В отличие от большинства предыдущих работ в этой области, здесь цель не только представить представление графа в верхней части сетки, но и дать еще один шаг вперед, предложив способ преобразования визуальной информации в структурах данных, с помощью метода обработки изображений, как описано.

Предлагаемый подход позволяет, как выход, полную характеристику топологических аспектов окружающей среды, которая обладает способностью помогать навигации робота в широком спектре мероприятий, особенно тех, которые включают в себя планирование маршрута.

Что касается будущей работы, то было бы интересно проверить этот подход с использованием различных методов в литературе, чтобы получить основной схеме, чтобы проверить, можно ли ускорить первый шаг алгоритма без потери качества на топологическом представлении. Кроме того, некоторые вопросы все еще остаются открытыми, как быстро решения обновление диаграммы Вороного данный динамические изменения в окружающей среде, а также с учетом 3D-моделей и работать с топологическими навигация с помощью мобильных роботов в сценарии реального мира.

Литература

- Beeson, P., Jong, N., and Kuiper, B. (2005). Towards autonomous topological place detection using the extended voronoi graph. In *Int. Conf. on Robotics and Automation (ICRA'05)*, pages 4373–4379, Barcelona.
- Delaunay, B. (1934). Sur la sphere vide. *Izvestia Akademii Nauk SSSR, Otdelenie Matematicheskikh i Estestvennykh Nauk*, 7(6):793–800.
- Dudek, G., Jenkin, M., Milios, E., and Wilkes, D. (1997). Map validation and robot self-location in a graph-like world. *Robotics and Autonomous Systems*, 22(2):159–178. Elsevier, Ltd.
- Elfes, A. (1989). Using occupancy grids for mobile robot perception and navigation. *Computer*, 22(6):46–57.
- Fazli, P., Davoodi, A., Pasquier, P., and Mackworth, A. (2010). Fault-tolerant multi-robot area coverage with limited visibility. In *2010 IEEE Int. Conf. on Robotics and Automation (ICRA'10)*, Anchorage, Alaska, USA.
- Ferreira, F., Davim, L., Rocha, R., Dias, J., and Santos, V. (2007). Presenting a technique for registering images and range data using a topological representation of a path within an environment. *J. of Automation, Mobile Robotics & Intelligent Systems*, 1(3):47–55.
- Katsilieris, F., Lindhe', M., Dimarogonas, D., O'gren, P., and Johansson, K. (2010). Demonstration of multi-robot search and secure. In *Int. Conf. on Robotics and Automation (ICRA'10)*, Anchorage, Alaska, USA.

- Ko, B., Song, J., and Lee, S. (2004). Realtime building of thinning-based topological map. In *Int. Conf. on Intelligent Robots and Systems*, Sandal, Japan.
- Kolling, A. and Carpin, S. (2008). Extracting surveillance graphs from robot maps. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'08)*, pages 2323–2328, Nice, France.
- Lozano-Pe´rez, T. and Wesley, M. (1979). An algorithm for planning collision-free paths among polyhedral obstacles. *Communications of the ACM*, 22(10):560–570.
- Machado, A. (2002). *Patrulha multiagente: Uma ana´lise emp´rica e sistema´tica*. Master’s thesis, Centro de Informa´tica, Universidade Federal de Pernambuco (UFPE), Recife, Brasil.
- Portugal, D. and Rocha, R. (2010). Msp algorithm: Multi-robot patrolling based on territory allocation using balanced graph partitioning. In *25th ACM Symposium on Applied Computing (SAC’2010)*, pages 1271–1276, Sierre, Switzerland.
- Reinhard, D. (2010). *Graph Theory*. Springer-Verlag Heidelberg, New York, electronic edition.
- Szabo´, R. (2004). Topological navigation of simulated robots using occupancy grid. *Int. J. of Advanced Robotic Systems*, 1(3):201–206.
- Thrun, S. (1998). Learning maps for indoor mobile robot navigation. *Artificial Intelligence*, 99:21–71. Elsevier Science Ltd.
- Thrun, S., Bugard, W., and Fox, D. (2000). A real-time algorithm for mobile robot mapping with applications to multi-robot and 3d mapping. In *Int. Conf. on Robotics and Automation*, pages 321–328, San Francisco.
- Voronoi, G. (1908). Nouvelles applications des parametres continus a` la the´orie des formes quadratiques. *Journal fu¨r die Reine und Angewandte Mathematik*, 134:198–287.
- Wallgru¨n, J. (2004). Hierarchical voronoi-based route graph representations for planning, spatial reasoning, and communication. In *4th Int. Cognitive Robotics Workshop (CogRob-2004)*, pages 64–69.
- Zimmer, U., Fischer, C., and Puttkamer, E. V. (1994). Navigation on topologic feature-maps. In *3rd International Conference on Fuzzy Logic, Neural Nets and Soft Computing*, pages 131–132, Fukuoka, Japan.