

Здравствуйте!

Начну с неформального описания задачи. Имеется морская навигационная карта, требуется предоставить пользователю возможность найти как можно больше непохожих, локально оптимальных маршрутов по воде. При этом само по себе решение задачи не является конечным результатом, а лишь осуществляет поддержку для принятия решения. Само решение принимается пользователем в голове, поэтому основной принцип подобных задач в том, что они должны решаться в режиме реального времени, чтобы не сбивать пользователя с мыслей.

На картинках проиллюстрировано понятие непохожести маршрутов. Основное требование состоит в том, что маршруты должны отличаться способом обхода существенных препятствий.

Поясню, зачем это нужно.

Задача поиска кратчайшего маршрута при наличии полигональных препятствий хорошо известна и исследована, однако при таком подходе возникают следующие проблемы:

- Кратчайший путь (по какой-либо метрике) не всегда оптимален, а реальные критерии оптимальности могут быть не очевидны и трудно формализуемы.
- Если по каким-то причинам воспользоваться кратчайшим маршрутом не представляется возможным (например, где-то проходят военные учения), то хочется иметь альтернативу.
- Иногда хочется переправить большое число кораблей из одного места в другое. Если все пойдут кратчайшим маршрутом, то могут потратить существенно больше времени, чем если часть пойдёт не кратчайшим путём.

При этом неизвестны алгоритмы `multipath planning` (то есть, поиска нескольких маршрутов), которые бы хорошо работали для поиска маршрутов по воде.

Существующие алгоритмы

Что касается известных алгоритмов поиска нескольких путей в графе, то им присущи следующие проблемы:

- Разрабатывались для других целей: например, поиск маршрутов по дорогам.
- Эти алгоритмы в большинстве своём оперируют абстрактными графами, не привязанными к конкретным координатам в мире.
- Как следствие, попытка применить эти алгоритмы к данной задаче приводит к похожим маршрутам.

Пример: видно, что маршруты почти не имеют общих рёбер, однако, неформально говоря, они очень похожи.

Формальная постановка задачи

На основе сказанного можно формализовать поставленную задачу. Более-менее всё можно просто зачитать. Про критерий непохожести будет сказано на следующем слайде. Поскольку требуется искать маршруты в реальном времени, потребуем не больше секунды на запрос.

Для оценки похожести будем использовать метрики на маршрутах. Сказать, что такое ρ_g и ρ_r .

В первой метрике для каждой вершины первого пути рассматривается длина кратчайшего расстояния до второго пути, и её значение — максимум этих расстояний. В зависимости от отношения её значения к длине кратчайшего из двух путей маршруты считаются либо похожими, либо непохожими, либо «возможно, похожими». Константы, скрытые в этих определениях, являются параметрами. Если первой метрики недостаточно, используется вторая, которая аналогична первой, но графовое расстояние до ближайшей вершины в графе делится на реальное расстояние до ближайшей вершины в мире. Большое значение этой метрики означает, что между путями есть какое-то препятствие, поскольку кратчайший путь в графе существенно больше реального расстояния до ближайшей вершины в мире. При этом не рассматриваются слишком близкие вершины, поскольку между ними может быть незначительное препятствие.

Чтобы искать пути, нужно построить граф.

- Контура в исходной карте упрощаются, чтобы в итоге было не очень много вершин в графе, и склеиваются, получается полигон, описывающий воду.
- Полигон смещается внутрь, потому что корабли не плавают в 12-мильной зоне и так будет лучше видно маршруты.
- Затем строится граф по сетке на плоскости, то есть добавляются рёбра сетки, принадлежащие полигону. Также добавляются вершины полигона рёбра из них во все видимые вершины в некотором радиусе.
- При этом важно ограничить максимальную длину ребра, поскольку путь ищется на сфере, а проверка корректности рёбер осуществляется на плоскости. В этом случае разница между кратчайшей траекторией на сфере и на плоскости будет пренебрежимо мала.
- Также добавляются рёбра, получившиеся в результате схлопывания отрезков при смещении полигона, и рёбра через 180-ый меридиан.

Поиск одного маршрута

Перейдём к описанию поиска. Начнём с описания поиска одного маршрута, поскольку нам нём основан поиск семейств маршрутов.

- Сначала происходит добавление вершин в граф и рёбер до ближайших вершин, находимых по сетке.
- Сам поиск пути осуществляется с помощью алгоритма Дейкстры.
- Кратчайший путь в графе не обязан быть действительно кратчайшим маршрутом между двумя точками, обычно его можно сократить, тем самым сделав маршрут локально оптимальным. Далее по тексту.
- В связи со структурой графа возможно появление слишком острых углов в маршруте, которые выглядят неестественно, поэтому выполняется их сглаживание. При сглаживании слишком острые углы аппроксимируются дугой эллипса, если при этом маршрут не начинает пересекать исходные препятствия.

Про рисунок всё понятно: слева сократилось, справа сгладилось.

Поиск нескольких маршрутов

Поиск нескольких маршрутов проиллюстрируем псевдокодом.

Сначала находится кратчайший путь описанным только что способом, он добавляется в результат, запоминается длина. Затем, пока не найдено слишком много маршрутов, делаем следующее: обновляем веса в графе (об этом будет сказано на следующем слайде), находим очередной маршрут. Если его длина слишком большая, то заканчиваем. Также заканчиваем, если среди имеющихся маршрутов имеется похожий на только что найденный в смысле описанных ранее метрик. Наконец, если маршрут хороший, то он добавляется в результат.

Теперь об обновлении весов. Для каждой вершины храним потенциалы, изначально равные единице. Чем больше потенциал вершины, тем сильнее будет увеличен вес инцидентных ей рёбер. При обновлении сначала построим новый граф с фиктивной вершиной, из которой проведены рёбра нулевого веса во все вершины пути. Затем с помощью алгоритма Дейкстры найдём кратчайшие расстояния из фиктивной вершины до всех остальных в этом графе в радиусе равном длине пути, поделённой на константу. Для всех вершин посчитаем новый потенциал как число от одного до максимального потенциала. Чем больше расстояние до пути, тем меньше потенциал. Обновим потенциалы всех вершин, взяв максимум. Обновим веса рёбер, взяв начальный вес, умноженный на среднее геометрическое потенциалов смежных вершин.

Для работы в режиме реального времени важно уметь быстро вычислять метрики. При вычислении значения первой метрики между двумя маршрутами сначала построим граф с фиктивной вершиной и рёбрами нулевого веса до вершин первого пути. Из фиктивной вершины запустим обход Дейкстры, пока не посетим все вершины второго пути. Таким образом, для каждой вершины второго маршрута будет известна длина кратчайшего пути до вершин первого маршрута. Аналогично найдём длины для вершин первого маршрута. Наконец, максимум посчитаем перебором, потому что всего вершин в маршрутах не очень много. При этом важно закончить поиск, если для какой-то вершины найденное значение позволяет утверждать, что маршруты точно непохожи. Это позволяет существенно ускорить время работы. Для вычисления второй метрики также нужно находить расстояние до ближайшей вершины в мире. Это было сделано простым перебором. На практике при таком подходе всё стало быстро работать.

Результаты

Типа вот вам результаты, всё круто, я хорош.

Спасибо за внимание!

До свидания!

Ниже фигня какая-то написана для поржать.

1. Рассмотрим такую ситуацию, вершины находятся довольно близко, поэтому их потенциалы примерно равны какому-то C . При этом, если прибавлять среднее арифметическое, то стоимость верхнего пути получится на C больше, хотя он лучше. Поэтому будем домножать на среднее геометрическое. В таком случае вклад потенциалов будет примерно равен, и будет найден верхний маршрут.

2. Слева показана ситуация, где потенциалы аккумулируются, а справа — где берётся максимум. Во первом случае почти весь вклад потенциалов сосредоточен в начале и конце. И в результате эксперимента во втором случае оказалось найдено на один маршрут больше.

3. Если потенциал монотонно убывает с ростом кратчайшего расстояния от найденного маршрута, то на маршруте будут самые большие потенциалы, поэтому следующий маршрут может отличаться в местах, где он ожидается таким же. Для получения более естественного результата немного уменьшим потенциалы вершин маршрута.