

Симуляция движения пешеходов с применением муравьиных алгоритмов с целью проектирования оптимальной сети дворовых дорожек.

Введение

Достаточно немного прогуляться по дворам чтобы заметить, что их проектировка, как правило, далека от идеала. Проведенные по линейке и пересекающиеся под прямыми углами дворовые проезды и пешеходные дорожки, может быть, и выглядят красиво, но при этом не очень удобны для использования.

Людам свойственно сокращать свой путь всеми возможными способами, поэтому прямые углы всячески срезаются (см. рис. 1), а через красивые газоны протаптываются тропинки.



Рисунок 1: Типичная тропинка через газон (Рощинская улица, СПб).

В итоге в дождливую погоду, особенно осенью или весной, жидкая грязь разносится с газона на ногах пешеходов по всему двору или улице, газон же при этом несет невосполнимый урон и превращается в болото.

В редких случаях местные власти принимают адекватные меры по исправлению данной ситуации и покрывают такие тропинки асфальтом или плиткой. В таком случае получается красивый и удобный для жителей двор, показанный на рис.2. В большинстве же дворов эту

проблему либо игнорируют, либо делают прямо противоположное — раз за разом засыпают тропинку свежей землей и засаживают травой, которая тут же снова вытапывается.



Рисунок 2: Пример хорошего двора с удобными тропинками (Московский проспект, СПб).

Когда-то давно автором была услышана история о том, как в некоем наукограде передовое руководство решило провести эксперимент. На первое время дорожек не было сделано вовсе, жители протоптали их сами, после чего их замостили и получили тем самым удобную сеть проходов.

При таком подходе нет никакой нужды, например, дополнительно огораживать заборами газоны и клумбы, так как у жителей просто не будет стимула по ним ходить — ведь до любого нужного места можно добраться по удобной дорожке, ведущей именно туда, куда надо.

На подобные эксперименты, однако, согласится далеко не каждый застройщик, к тому же пока дорожки проложены и замощены не будут, во дворе все равно будет грязно. И вот тут как раз и могут пригодиться методы компьютерного моделирования, рассматриваемые в данной статье.

Постановка задачи

Дана карта местности, содержащая в себе объекты следующих типов:

- Непреодолимые препятствия, такие как дом или забор.
- Источники пешеходов (в дальнейшем будем называть их *генераторами*), это места между которыми пешеходы передвигаются, например подъезд дома, вход в магазин или детская площадка.
- Проходимые области, такие как газон или дорога, сюда же относятся небольшие легко преодолимые препятствия, например невысокие оградки по колено, у каждой области задана сложность прохода через нее.

Необходимо определить те места, по которым надо проложить дороги, чтобы пешеходы, двигаясь между генераторами, как можно меньше сходили с них (в идеале, чтобы даже самые недисциплинированные пешеходы двигались только по проложенным дорогам), при этом построенная сеть дорог должна иметь *наименьшую возможную длину*.

Последнее условие весьма существенно. Без него идеальным решением задачи была бы система дорог, соединяющая напрямую все генераторы пешеходов. Такое решение было бы максимально удобно с точки зрения движения, но оно является наиболее дорогим в реализации (требуется больше покрытия), а так же может вступать в конфликт с другими задачами благоустройства (например, необходимостью оставить некоторый процент площади двора под озеленение).

На рис.3 приведен простой пример, показывающий различия в системе прямых путей (слева) и сокращенной но по-прежнему удобной системе справа. При такой конфигурации генераторов (цветные квадраты) пешеходам нет нужды идти напрямую между верхним и нижними генераторами, так как вполне достаточно дорожки проходящей через центральный.

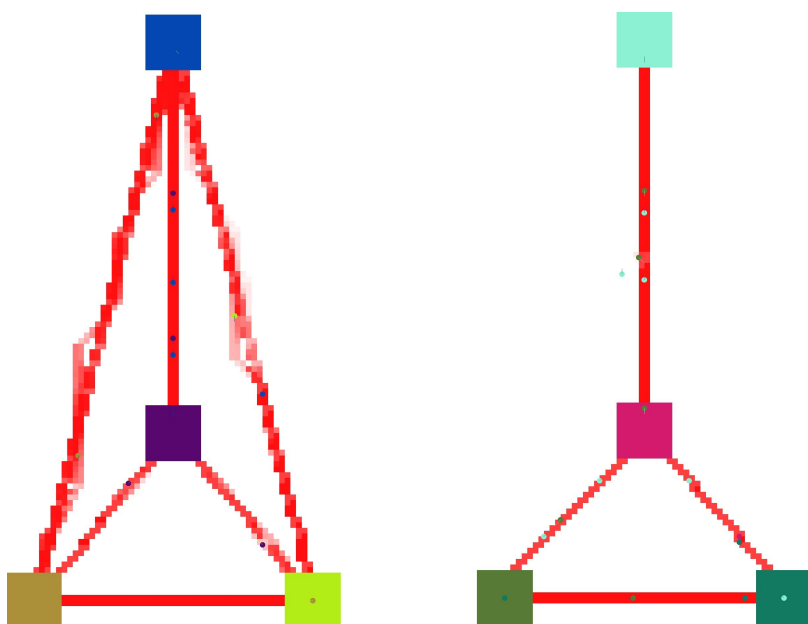


Рисунок 3: Система полных путей (слева) и более оптимальная версия, решающая ту же задачу с меньшими затратами (справа).

Отдельно стоит отметить, что пути, прокладываемые пешеходами, нередко напоминают минимальное остовное дерево (*MST, Minimal Spanning Tree*) графа перемещений, однако реальные результаты, взятые из жизни, все-таки обычно лежат где-то посередине между системой полных путей и MST. Это было показано в работе [3], в которой авторы пытались использовать симуляцию движения пешеходов для поиска этих самых MST.

Поведение пешеходов, схожесть с муравьиными алгоритмами

Рассмотрим поведение типичного пешехода, желающего пройти по двору между двумя точками. Его действия при выборе пути преследуют две цели — локальную и глобальную.

Глобальная цель — это собственно необходимость достичь пункта назначения.

Локальная цель — это выбор направления следующих нескольких шагов таким образом, чтобы обеспечить максимальное *удобство пути*. Данная характеристика в достаточной степени индивидуальна, и может включать в себя пройденное расстояние, качество дорожного покрытия, отсутствие необходимости вытаптывать траву и т. п. В одной и той же ситуации один пешеход предпочтет более длинный, но более чистый путь по асфальту, а другой пойдет напрямик, не обращая внимания на грязь или препятствия. Именно так и появляются тропинки — сперва находится несколько человек, которым геометрическое расстояние важнее удобства передвижения, именно они первыми начинают ходить через газоны. Если таких людей мало — тропинка так и не образуется, не наберет достаточно вытоптанности чтобы стать привлекательной для других, более «порядочных» пешеходов. Если же их много — то вскоре на месте газона появится широкая и надежная тропа, по которой начнут ходить все или почти все местные жители.

Глобальные цели для пешеходов являются постоянными и заданными по условиям задачи, а вот локальное удобство пути может изменяться со временем. Пешеходы вытаптывают тропинки на газонах, повышая тем самым их привлекательность. Там, где на первых порах ходило всего пара человек, со временем начинает ходить все больше и больше, ведь вытоптанная тропинка в хорошую погоду по удобству передвижения не намного хуже заасфальтированного тротуара.

Из указанных особенностей поведения и вытекает схожесть с поведением муравьев в классическом муравьином алгоритме. Чем больше пешеходов проходит через данную местность, тем больше они ее «вытаптывают», тем больше пешеходов пойдет по ней в дальнейшем. В данном случае «вытоптанность» пути играет роль феромонов из муравьиного алгоритма и определяет привлекательность пути для других участников движения.

Помимо увеличения потока пешеходов по вытоптаным путям, есть и второе сходство, а именно растворение со временем: если по тропинке никто не ходит, то она со временем зарастает и возвращается к исходному состоянию. Схожим образом, феромоны в классическом муравьином алгоритме «выветриваются» со временем.

Требования к алгоритму, существующие реализации

Исходя из описанного выше, алгоритм, решающий поставленную задачу, должен одновременно как решать общую задачу поиска пути, так и учитывать изменение со временем удобства каждого конкретного участка пути.

На настоящий момент существуют алгоритмы, подходящие для удовлетворения указанных условий по отдельности, что не позволяет использовать их для симуляции движения пешеходов во взятых из реальной жизни ситуациях. Задачей данной работы является их модификация и демонстрация работы на реальном примере достаточно сложного двора.

Active walkers

Авторами работ [1-2] была предложена модель, названная *Active Walkers*. Она представляет собой жадный алгоритм поиска пути, выбирающий направление следующего шага в

зависимости от направления и расстояния до цели пути и градиента функции привлекательности окружающей поверхности в текущей точке. Сама же функция привлекательности для каждой точки пространства задается с учетом рассмотренных выше соображений (увеличение с ростом количества проходящих через нее людей, уменьшение со временем).

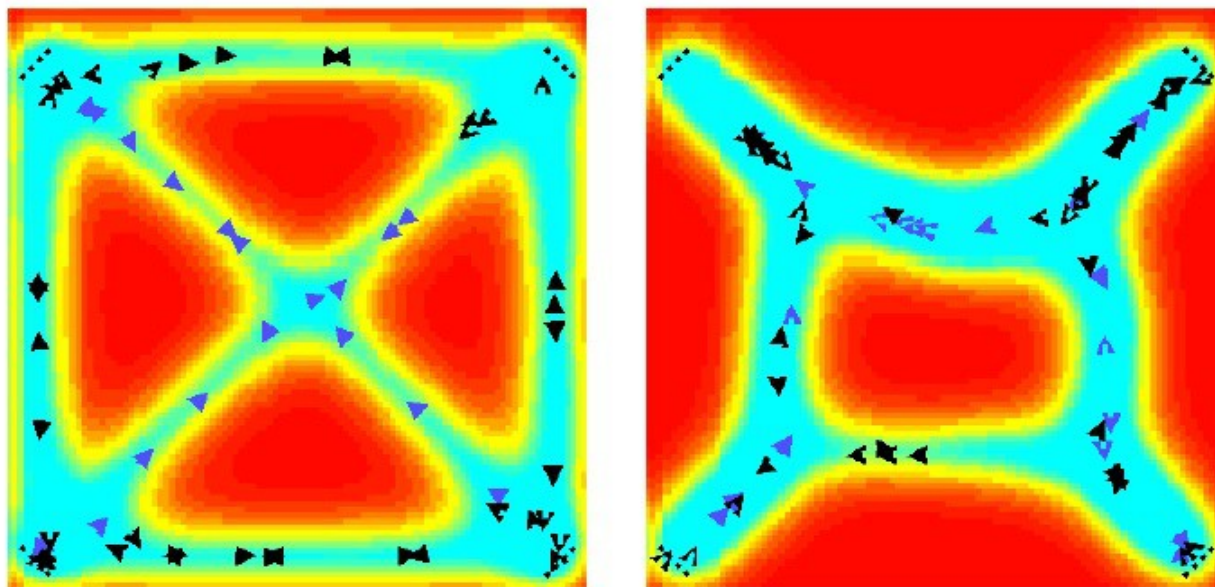


Рисунок 4: Пример результатов симуляции алгоритма *Active Walkers*

На рис.4 приведен пример из статьи [1]. Слева показано состояние после одного из первых шагов симуляции, справа — окончательный результат. Стрелками показаны пешеходы, цветом — интенсивность движения через точку, голубой цвет означает наиболее интенсивное движение. Как можно видеть, со временем простой вариант с прямыми путями между всеми генераторами заменяется менее тривиальным и имеющим меньшую суммарную длину. Форма итоговой сети тропинок зависит от параметров симуляции (на сколько тропинки удобнее газона, на сколько они зарастают и на сколько вытаптываются от каждого пешехода).

Недостаток у такого подхода один, но существенный. Данный подход отвечает только за выполнение локальной цели (удобства следующего шага), при этом плохо решает общую задачу поиска пути, так как работает он только при отсутствии препятствий сложной формы и, в особенности, тупиков. На рис.5 показан типичный пример препятствия, на котором подобные алгоритмы построят путь, совершенно не похожий на оптимальный, так как будут до последнего стараться идти в направлении цели и в итоге зайдут в тупик.

Данная проблема характерна для всех жадных алгоритмов поиска пути (сравнение и недостатки таких алгоритмов можно найти, например, в [4]), в том числе и для методов градиентного спуска, к которым относится рассматриваемая модель. Схожие проблемы (нахождение извилистых неоптимальных путей и даже заикливание) отмечали авторы работы [5], исследовавшие применение *Active Walkers* к сложным ландшафтам.

В итоге данный алгоритм не может быть применен для достаточно сложных примеров из реальной жизни, с его помощью можно максимум находить пути пешеходов через один отдельно взятый газон.

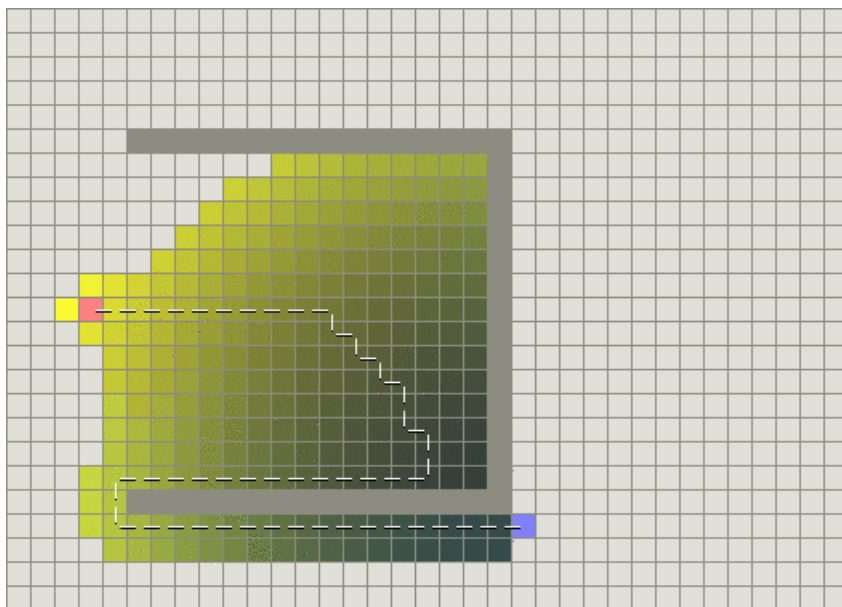


Рисунок 5: Пример «плохого» препятствия для жадного алгоритма поиска пути

Классические алгоритмы поиска пути

Такие алгоритмы, как алгоритм Дейкстры, или его широко используемая в компьютерных играх модификация A^* , хорошо подходят для решения глобальной задачи поиска пути между двумя точками. Но они в своей работе оперируют понятием веса ребра графа, который в простейшем случае соответствует расстоянию между вершинами. В случае же рассматриваемой задачи требуется дополнительное определение веса ребра, которое удовлетворяло бы поставленным условиям и представляло бы как раз то самое *удобство пути*. Этим мы и займемся в следующем разделе.

Алгоритм Дейкстры гарантированно находит оптимальный путь, при этом он является полностью информированным алгоритмом — то есть ищущий путь агент обладает полной информацией обо всей карте местности. Существуют и другие варианты — например, алгоритмы, в которых знания агента об окружающем пространстве ограничены радиусом, или сектором видимости. На первый взгляд может показаться, что подобные алгоритмы лучше соответствуют ситуации поиска пути человеком (так как у человека поле зрения ограничено, он не может увидеть что за углом — стена, пока не зайдет туда).

Однако это так только лишь в случае поиска пути на незнакомой местности. В данной работе рассматривается приложение алгоритма к типичному городскому двору, а через двор, как правило, ходят лишь люди, которые живут поблизости, следовательно хорошо знакомы с территорией. Именно поэтому логично использовать алгоритм, имеющий доступ к полной карте — ведь живущий в этом дворе человек уже знает, что за углом его ждет тупик, и идти надо совсем в другую сторону, так что он будет двигаться глобально оптимальным путем.

Предлагаемый алгоритм

Карта местности представляется в виде навигационного графа, вес ребер которого зависит от типа местности и определяет ее привлекательность для пешеходов. Вес составляется из двух компонент: фиксированной, определяемой типом местности, и изменяемой, показывающую вытоптанность местности в данной точки. У некоторых типов ландшафта (дорожек с жестким покрытием) изменяемой компоненты может не быть.

Запускается симуляция. Заданное количество пешеходов начинает свое движение между случайно выбранными генераторами. На каждом шагу симуляции:

- 1) Пешеходы проходят определенное расстояние.

2) У пройденных ребер графа уменьшаются веса (вытаптываются).

3) Дошедшие до своей цели пешеходы заменяются на новых.

4) Вес всех вытапанных ребер в графе увеличивается на фиксированную величину (зарастание со временем).

Маршрут свой пешеходы прокладывают с помощью алгоритма Дейкстры. При этом для симуляции разных категорий пешеходов (как любящих ходить напрямик, так и предпочитающих придерживаться дорог) используется «коэффициент порядочности». На этот коэффициент домножаются веса вытапываемых ребер, таким образом при значении коэффициента меньше единицы пешеход будет более склонен к срезанию пути через газоны.

Симуляция продолжается заданное число шагов, после ее завершения карта с распределением изменяемой компоненты веса ребер демонстрирует участки, на которых пешеходы чаще всего сходят с дорожек на землю, и которые разумно было бы замостить твердым покрытием.

Работа алгоритма таким образом определяется следующими параметрами:

- Начальные веса разных типов поверхностей.
- Изменение веса при движении пешехода через ребро.
- Изменение веса ребер со временем.

Оптимальные значения параметров подбирались сперва на простом примере с рис. 3, они же дали близкие к реальности результаты, описанные в следующем разделе.

Пример работы

Для примера был рассмотрен двор, находящийся по адресу Санкт-Петербург, проспект Гагарина 14, корпус 6. Это многоквартирный дом, возведенный посередине квартала пятиэтажных панельных «хрущевок», в нем проживает больше человек, чем в нескольких соседних домах вместе взятых, однако планировка пешеходных дорожек вокруг него оставляет желать лучшего. Жители этого дома уже протоптали несколько заметных и достаточно длинных тропинок, которые удобно сравнивать с результатами симуляции.

На первом этапе работы в онлайн-редакторе *geojson.io* была размечена карта местности, как показано на рис. 6.

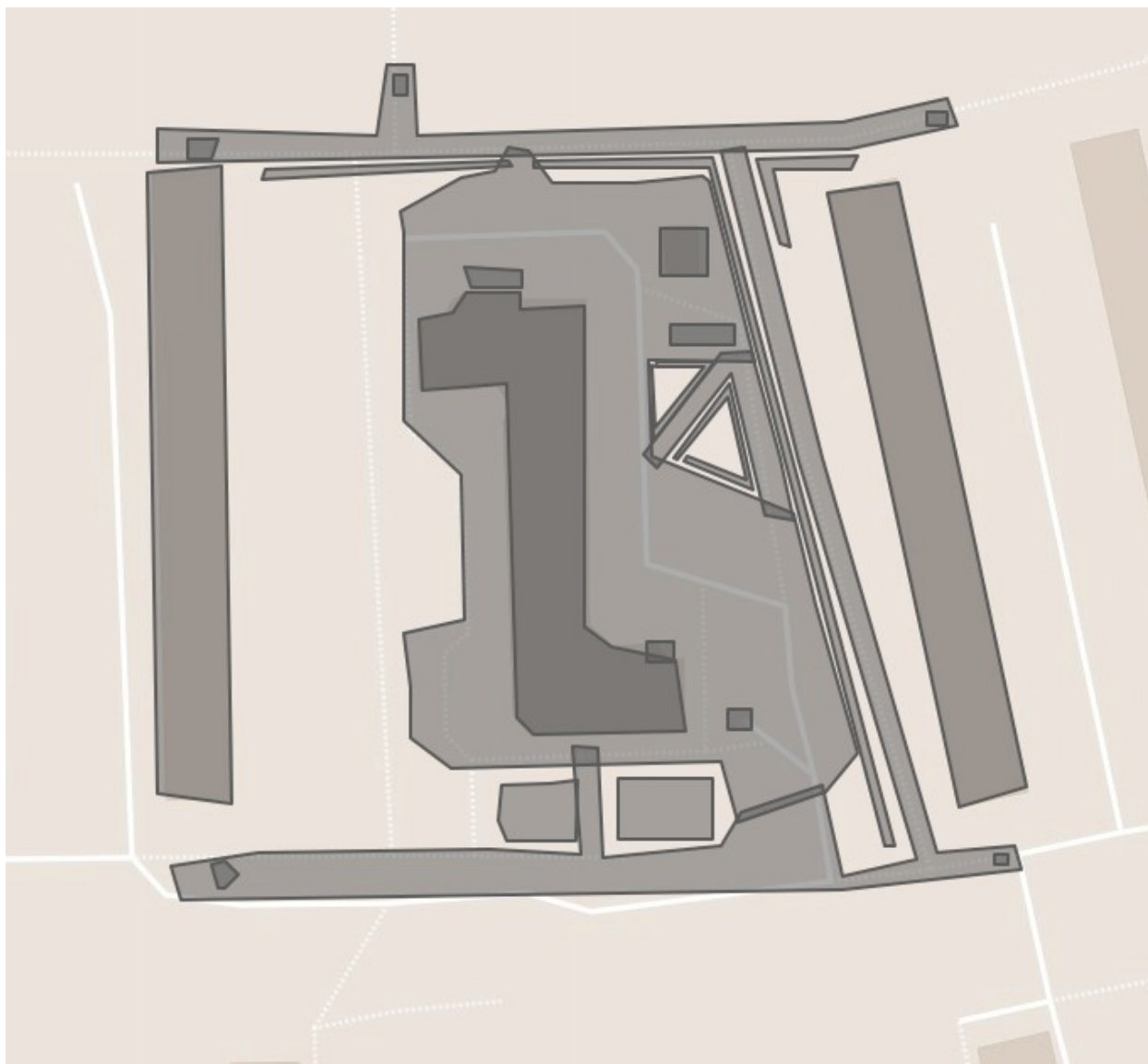


Рисунок 6: Размеченная карта местности

На втором этапе полученный файл в формате *GeoJSON* [6] загружается в приложение. На карту накладывается сетка, координаты ее линий – это координаты точек (то есть каждая точка многоугольников, обозначающих объекты, лежит на пересечении одной горизонтальной и вертикальной линии сетки). Дополнительные линии добавляются, чтобы обеспечить заданную максимальную ширину интервалов при необходимости.

Данная сетка служит основой для построения навигационного графа. Каждый узел сетки становится вершиной графа, ребра добавляются при проходимости данного участка, начальный вес ребра задается как длина, умноженная на вес территории, на которой лежит ребро (для дорог вес минимальный, для газонов, соответственно, значительно более высокий, наиболее похожий на реальность результат был получен при весе газона в 4 раза большем, чем у дороги).

На третьем этапе запускается собственно симуляция. На рис. 7 показаны ее результаты. Цветные кружки — пешеходы, цвет показывает, из какого генератора они идут, линии — их маршруты движения. Красным отмечены вытопанные участки карты.

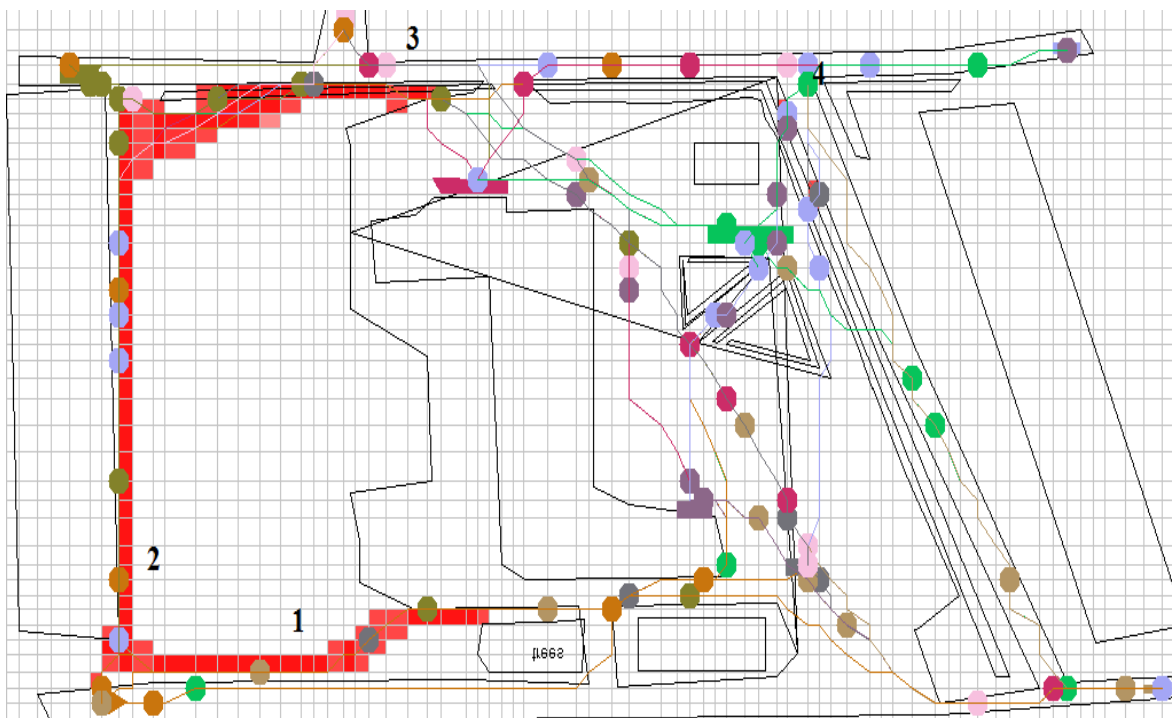


Рисунок 7: Результаты работы симуляции

Цифрами отмечены найденные алгоритмом вытоптанные участки, соответствующие им фотографии реального двора приведены в приложении. Как можно увидеть, соответствие достаточно точное.

На результатах симуляции можно увидеть четыре основных просчета в планировке двора:

- 1) Отсутствие дорожки между дорогой на юге и углом дома.
- 2) Отсутствие прохода с севера на юг слева от дома (при этом справа от дома такой проход сделан).
- 3) Напротив северного подъезда нет выхода с придомовой площадки на идущую с востока на запад дорожку, выход сделан восточнее, метрах в 7. В итоге пешеходы идут напрямик по газону. В этом месте их даже не может остановить сделанная там ограда, которую регулярно ломают и снова чинят.
- 4) Аналогичная проблема с отсутствием выхода с угла придомовой парковки на северную дорожку, в итоге люди там лазают через невысокую ограду и полосу газона.

Применение рассматриваемого алгоритма на этапе проектирования двора могло бы заранее выявить такие недоработки и помочь их избежать, тем самым заметно повысив уровень комфорта перемещений для местных жителей.

Заключение

Предложенный алгоритм симуляции движения оказался способен хорошо предсказать поведение пешеходов из реальной жизни и смог построить близкую к реальности схему тропинок. При этом данный алгоритм лишен недостатков уже существующих подходов, не позволяющих применять их для решения сложных задач.

Основанное на данном алгоритме программное средство могло бы помочь избежать типичных ошибок планировки дворовых территорий и сократить затраты на благоустройство и исправление этих ошибок после завершения строительства.

Источники

1. D. Helbing, J. Keltsch, P. Molnar. «Modelling the Evolution of Human Trail Systems». Nature. 1997.
2. S. Gilks, J. Hague. «Mountain trail formation and the active walker model». International Journal of Modern Physics C (IJMPC) 2009, Vol. 20, pp869 – 890
3. R. Goldstone, M. Roberts. «Self-organized Trail Systems in Groups of Humans». Proceedings of «Understanding Complex Systems» conference, 2006.
4. C.Wilt, J.Thayer, W.Ruml. «A Comparison of Greedy Search Algorithms», Proceedings of the Third Annual Symposium on Combinatorial Search (SOCS-10)
5. A. Girdhar, J. Antonaglia. «Investigation of Trail Formation with the Active Walker Model». 2013
6. <http://geojson.org> Описание формата ГИС-данных *GeoJSON*.

Приложение. Фотографии двора из рассмотренного примера.



Рисунок 8: Тропа №1, от угла дома к дорожке.



Рисунок 9: Тропа №2, вдоль пятиэтажного дома (он слева за деревьями)



Рисунок 10: Тропа №3, ведущая наискосок от подъезда к дорожке, справа она пересекается с тропой номер 2.



Рисунок 11: Тропа №4, от угла парковки возле трансформаторной будки, через ограду, к пешеходной дорожке.