

7. Диаграмма Вороного

Проблема Почтового отделения

Предположите, что Вы находитесь на консультативном совете для планирования сети супермаркетов, и есть планы открыть новый магазин в определенном месте. Чтобы предсказать, будет ли новый магазин успешен, вы должны оценить число клиентов, которых он привлечет. Для этого вы должны смоделировать поведение своих потенциальных клиентов: как люди решают, где ходить по магазинам? Подобный вопрос возникает в социальной географии, при изучении деловой активности в стране: какова экономическая область определенных городов? При более абстрактных условиях у нас есть ряд центральных мест (сайтов) – которые обеспечивают определенные товары или услуги, и мы хотим знать для каждого места (сайта), где живут люди, кто получает их товары или услуги от того места (сайта). (В вычислительной геометрии места (сайты) традиционно рассматриваются как почтовые отделения, где клиенты хотят отправить свои письма, смотри подзаголовки этой главы.) Чтобы изучить этот вопрос, мы делаем следующие предположения упрощения:

- цена особой пользы или обслуживания – одинаковая на каждом месте (сайте);
- стоимость приобретения пользы или обслуживания равна цене плюс стоимость транспортировки к сайту;
- потребители пытаются минимизировать стоимость приобретения пользы или обслуживания;

Обычно эти предположения не полностью удовлетворены: товары могут быть более дешевыми на некоторых местах (сайтах) чем в других, и стоимость транспортировки между двумя точками, вероятно, не линейна при Евклидовом расстоянии между ними. Но модель выше может дать грубое приближение торговых областей мест (сайтов). Области, где поведение людей отличается от предсказанного моделью, могут быть подвергнуты дальнейшему исследованию, чтобы видеть то, что вызвало другое поведение.

Наш интерес заключается в геометрической интерпретации модели выше. Предположения в модели вызывают подразделение общей площади на рассмотрении в регионы – торговые области мест (сайтов) – таким образом, что люди, которые живут в том же самом регионе, все идут в то же самое место (сайт). Наши

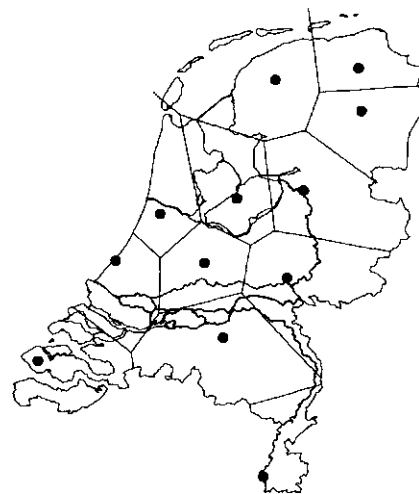


Рисунок 7.1. Торговые области столиц двенадцати областей в Нидерландах, как предсказано моделью назначения Вороного.

предположения подразумевают, что люди просто получают свои товары на самом близком месте - довольно реалистическая ситуация. Это означает, что торговая область для данного места состоит из всех тех точек, для которых то место ближе чем любое другое место. Рисунок 7.1 дает пример. Места на этом рисунке – столицы двенадцати областей в Нидерландах.

Модель, где каждая точка отнесена к ближайшему месту (сайту), называют моделью назначения Вороного. Подразделение, вызванное этой моделью, называют диаграммой Вороного набора мест (сайтов). Из диаграммы Вороного мы можем получить все виды информации о торговых областях мест (сайтов) и их отношений. Например, если у областей двух сайтов будет общая граница тогда, эти два места, вероятно, будут конкурировать за клиентов, которые живут в граничной области.

Диаграмма Вороного – универсальная геометрическая структура. Мы описали применение к социальной географии, но у диаграммы Вороного есть применения в физике, астрономии, робототехнике, и еще многих областях. Она также близко связана с другой важной геометрической структурой, так называемой триангуляцией Делоне, с которой мы столкнемся в Главе 9. В текущей главе мы ограничимся основными свойствами и составлением диаграммы Вороного ряда точечных сайтов в плане.

7.1. Определение и Основные Свойства

Обозначьте Евклидово расстояние между двумя точками p и q как $dist(q, p_i)$. На плоскости мы имеем:

$$dist(p, q) := \sqrt{(p_x + q_x)^2 + (p_y + q_y)^2}$$

Пусть $P := \{p_1, p_2, \dots, p_n\}$ будет множеством n различных точек на плоскости; эти точки сайты. Мы определяем диаграмму Вороного P как разбиение плоскости на n ячеек, по одной для каждого сайта в P , при условии, что точка q находится в ячейке соответствующей сайту p_i если и только если $dist(q, p_i) < dist(q, p_j)$ для каждого $p_i \in P$ и $j \neq i$. Мы обозначаем диаграмму Вороного P как $Vor(P)$. Злоупотребляя терминологией немного, мы будем иногда использовать ' $Vor(P)$ ' или 'диаграмма Вороного', чтобы указать только на грани и вершины разбиения. Например, когда мы говорим, что диаграмма Вороного связана, мы подразумеваем, что союз ее граней и вершин формирует связанный набор. Ячейка $Vor(P)$ которая соответствует сайту p_i обозначается $\mathcal{V}(p_i)$; мы называем это ячейкой Вороного от p_i . (В терминологии вступления к этой главе: $\mathcal{V}(p_i)$ торговая область места (сайта) p_i .)

Мы теперь бросаем более близкий взгляд на диаграмму Вороного. Сначала мы изучаем структуру одной клетки Вороного. Для двух сайтов p и q в плоскости мы определяем среднюю линию (*bisector*) между p и q как перпендикулярная средняя линия от линейного сегмента \overline{pq} . Эта средняя линия раскалывает плоскость на две полуплоскости. Мы обозначаем открытую полуплоскость, которая содержит p как $h(p, q)$ и открытую полуплоскость, которая содержит q как $h(q, p)$. Заметьте что $r \in h(p, q)$ если и только если $\text{dist}(r, p) < \text{dist}(r, q)$. От этого мы получаем следующее наблюдение.

Наблюдение 7.1. $\mathcal{V}(p_i) = \bigcap_{1 \leq j \leq n, j \neq i} h(p_i, p_j)$.

Таким образом $\mathcal{V}(p_i)$ пересечение $n - 1$ полуплоскостей и, следовательно, представляет собой (возможно безграничный) открытый выпуклый многоугольный регион ограниченный максимум $n - 1$ вершин и максимумом $n - 1$ граней.

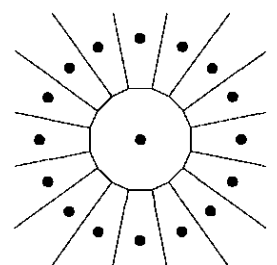
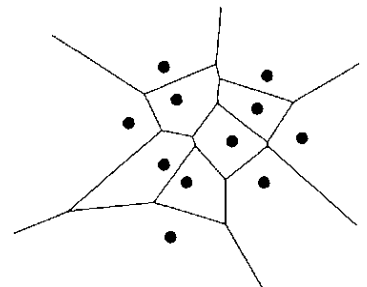
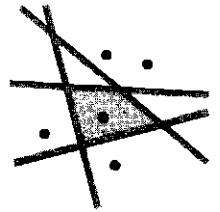
На что похожа полная диаграмма Вороного? Мы только что видели, что каждая клетка диаграммы – пересечение многих полуплоскостей, таким образом, диаграмма Вороного – плоское подразделение, грани которого являются прямыми. Некоторые грани – отрезки, а другие – лучи. Если все сайты будут коллинеарны, то все грани будут прямыми (x-line.)

Теорема 7.2. Пусть P множество n сайтов на плоскости. Если все сайты коллинеарны тогда $\text{Vor}(P)$ состоит из $n - 1$ параллельных линий. Иначе $\text{Vor}(P)$ связан и его грани – или отрезки или лучи.

Доказательство. Первую часть теоремы легко доказать, так предположите, что не все сайты в P коллинеарны.

Мы сначала показываем, что грани $\text{Vor}(P)$ являются или сегментами или лучами. Мы уже знаем, что грани $\text{Vor}(P)$ являются частями прямых линий, а именно, части средних линий между парами сайтов. Теперь предположите для противоречия, что есть грань e $\text{Vor}(P)$, которая является сплошной линией. Позвольте e быть на границе ячеек Вороного $\mathcal{V}(p_i)$ и $\mathcal{V}(p_j)$. Позвольте $p_k \in P$ быть сайтом, который не коллинеарен с p_i и p_j . Средняя линия между p_j и p_k не параллельна e и, следовательно, она пересекает e . Но тогда часть e , которая находится внутри $h(p_k, p_j)$ не может быть на границе $\mathcal{V}(p_j)$, потому что она ближе к p_k чем к p_j , что приводит к противоречию.

Остается доказывать, что $\text{Vor}(P)$ связан. Если бы это не имело место, тогда была бы ячейка Вороного $\mathcal{V}(p_i)$, раскалывающая плоскость на две части. Поскольку клетки Вороного выпуклы, $\mathcal{V}(p_i)$ состоял бы из полосы, ограниченной двумя параллельными сплошными линиями. Но мы только что доказа-



ли, что грани данной диаграммы Вороного не могут быть сплошными линиями, что приводит к противоречию.

Теперь, когда мы понимаем структуру диаграммы Вороного, мы исследуем ее сложность, то есть, общее количество ее вершин и граней. С тех пор есть n сайты, и каждая ячейка Вороного имеет в большинстве $n - 1$ вершина и грани, сложность $Vor(P)$ является самое большее квадратной. Не ясно, однако, может ли у $Vor(P)$ фактически быть квадратная сложность: легко построить пример, где у отдельной ячейки Вороного есть линейная сложность, но это может произойти, что у многих клеток есть линейная сложность? Следующая теорема показывает, что дело обстоит не так и что среднее число вершин ячеек Вороного - меньше чем шесть.

Теорема 7.3. Для $n \geq 3$, число вершин в диаграмме Вороного множества n сайтов на плоскости не больше $2n - 5$, и число граней не больше $3n - 6$.

Доказательство. Если все сайты коллинеарны тогда, теорема немедленно следует из Теоремы 7.2, так что предположим, что дело обстоит не так. Мы докажем теорему, используя формулу Эйлера, которая заявляет, что для любого связанного плоского вложенного графа с m_v узлов, m_e дуг, и m_f граней, даёт следующее отношение:

$$m_v - m_e + m_f = 0$$

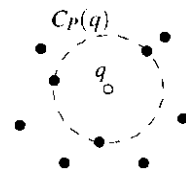
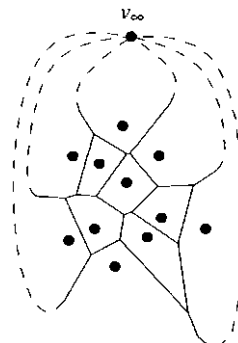
Мы не можем применить формулу Эйлера непосредственно к $Vor(P)$, потому что $Vor(P)$ имеет полубесконечные грани и является поэтому не надлежащим графом. Чтобы исправить ситуацию, мы добавляем одну дополнительную вершину v_∞ "в бесконечности" к набору вершин и мы соединяем все полубесконечные грани $Vor(P)$ с этой вершиной. У нас теперь есть связанный плоский граф, к которому мы можем применить формулу Эйлера. Мы получаем следующее отношение между n_v , числом вершин $Vor(P)$, n_e , числом граней $Vor(P)$, и n , числом сайтов:

$$(n_v + 1) - n_e + n = 2$$

Кроме того, у каждого края в увеличенном графе есть точно две вершины, так, если мы суммируем степени всех вершин, мы получаем дважды число граней. Поскольку у каждой вершины, включая v_∞ , есть степень по крайней мере три, мы добиваемся

$$2n_e \geq 3(n_v + 1)$$

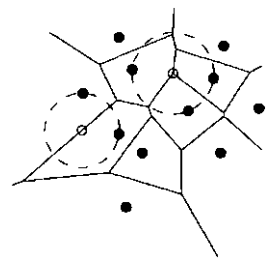
Вместе с уравнением (7.1) это подразумевает теорему.



Мы закрываем эту секцию с характеристикой граней и вершинами диаграммы Вороного. Мы знаем, что грани - части средних линий пар сайтов и что вершины - точки пересечения между этими средними линиями. Есть квадратное число средних линий, тогда как сложность $Vor(P)$ только линейна. Следовательно, не все средние линии определяют грани $Vor(P)$, и не все пересечения вершины $Vor(P)$. Характеризовать, какие средние линии и пересечения определяют особенности диаграммы Вороного, мы делаем следующее определение. Для точки q мы определяем *самый большой пустой круг q относительно P* , обозначенного $Cr(q)$, как самый большой круг с центром в q , который не содержит сайта P в его интерьере. Следующая теорема характеризует вершины и грани диаграммы Вороного.

Теорема 7.4. Для диаграммы Вороного $Vor(P)$ ряда точек P следующее утверждения:

- (i). Точка q является вершиной $Vor(P)$, если и только если её самый большой пустой круг $Cr(q)$ содержит три или больше сайта на его границе.
- (ii). Средняя линия между сайтами p_i и p_j определяет грань $Vor(P)$, если и только если есть точка q на средней линии, такая, что $Cr(q)$ содержит и p_i , и p_j на его границе, но никаком другом сайте.



Доказательство:

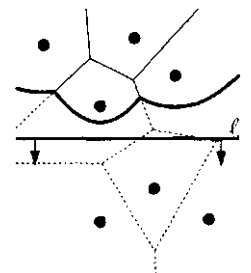
- (i). Предположим, что есть точка q такая, что $Cr(q)$ содержит три или больше сайта на своей границе. Пусть p_i, p_j , и p_k будут тремя из этих сайтов. Начиная с интерьера $Cr(q)$ - пустой q , должен быть на границе каждого $V(p_i), V(p_j), V(p_k)$, и q должен быть вершиной $Vor(P)$. С другой стороны, каждая вершина q $Vor(P)$ является инцидентом по крайней мере к трем граням и, следовательно, по крайней мере к трем клеткам Вороного $V(p_i), V(p_j)$, и $V(p_k)$ - Вершина q должна быть равноудалена от p_i, p_j , и p_k и не может быть другого сайта ближе к q , иначе бы $V(p_i), V(p_j)$, и $V(p_k)$ не встретились бы в q . Следовательно, интерьер круга с p_i, p_j , и p_k на его границе не содержит другого сайта.
- (ii). Предположите, что есть точка q со свойством, заявленной в теореме. Начнём с того, что $Cr(q)$ не содержит сайтов в своем интерьере и p_i , и p_j находятся на его границе, у нас есть $dist(q, p_i) = dist(q, p_j) \leq dist(q, p_k)$ для всех $1 \leq k \leq n$. Из этого следует, что q находится на краю или вершине $Vor(P)$. Первая часть теоремы подразумевает, что q не может быть вершиной $Vor(P)$. Следовательно, q находится на краю $Vor(P)$, который определен средней линией между p_i , и p_j . Наоборот, позвольте средней

линии между p_i и p_j определять грань Вороного. Самый большой пустой круг любой точки q в интерьере этой грани должен содержать p_i и p_j на его границе и никаких других сайтов.

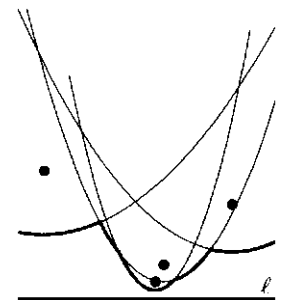
7.2. Вычисление диаграммы Вороного

В предыдущей секции мы изучили структуру диаграммы Вороного. Мы теперь намереваемся вычислять её. Наблюдение 7.1 предлагает простой способ сделать это: для каждого сайта p_i - вычисляют общее пересечение полуплоскостей $h(p_i, p_j)$, с $j \neq i$, используя алгоритм, представленный в Главе 4. Этим путем мы тратим $O(n \log n)$ время за ячейку Вороного, что приводит к $O(n^2 \log n)$ алгоритму, чтобы вычислить целую диаграмму Вороного. Разве мы не можем добиться большего успеха? В конце концов, полная сложность диаграммы Вороного только линейна. Ответ - да: алгоритм заметания плоскости, описанный ниже - обычно известный как алгоритм Форчуна в честь его изобретателя - вычисляет диаграмму Вороного в $O(n \log n)$ время. Вы можете испытать желание искать еще более быстрый алгоритм, например тот, который бежит в линейное время. Это, оказывается, слишком много, чтобы спросить: проблема сортировки n действительных чисел сводима к проблеме вычисления диаграммы Вороного, таким образом, любой алгоритм для того, чтобы вычислить диаграмму Вороного должен взять $\Omega(n \log n)$ время в худшем случае. Следовательно, алгоритм Форчуна оптимален.

Стратегия в алгоритме заметания плоскости состоит в том, чтобы заметать по горизонтальной линии -двигающейся сверху вниз по плоскости. В то время как зачистка - выполненная информация, поддержан относительно структуры, которую каждый хочет вычислить. Более точно, информация поддержана о пересечении структуры с линией зачистки. В то время как линия зачистки перемещается вниз, информация не изменяется, кроме как в определенных специальных точках - точках событий.



Давайте попытаемся применить эту общую стратегию к вычислению диаграммы Вороного набора $P = \{p_1, p_2, \dots, p_n\}$ сайтов на плоскости. Согласно парадигме заметания плоскости мы перемещаем горизонтальную линию заметания ℓ сверху донизу по плоскости. Парадигма вовлекает поддержание пересечения диаграммы Вороного с линией зачистки. К сожалению, это не настолько легко, потому что часть $Vor(P)$ выше ℓ зависит не только от территории, которая лежит выше ℓ , но также и на территории ниже ℓ . Говоря по-другому, когда линия зачистки достигает самой верхней вершины ячейки Вороного, $\mathcal{V}(p_i)$ то она еще не столкну-



лась с соответствующим p_i сайтом. Следовательно, у нас нет всей информации, для определения вершины. Мы вынуждены применить парадигму заметания плоскости немного отличающимся способом: вместо того, чтобы поддержать пересечение диаграммы Вороного с линией заметания, мы сохраняем информацию о части сайтов диаграммы Вороного выше ℓ , которые не могут быть изменены сайтами ниже ℓ .

Обозначьте закрытую полуплоскость выше ℓ как ℓ^+ . Какова часть диаграммы Вороного выше ℓ , которая не может быть больше изменена? Другими словами, для каких точек $q \in \ell^+$ мы знаем наверняка, каков их самый близкий сайт? Расстояние точки $q \in \ell^+$ к любому сайту ниже ℓ больше чем расстояние q к ℓ непосредственно. Следовательно, самый близкий сайт q не может лечь ниже ℓ , если q , по крайней мере, как близко к некоторому сайту $p_i \in \ell^+$, как q к ℓ . Местоположение точек, которые ближе к некоторому сайту $p_i \in \ell^+$ чем к ℓ , ограничено параболой. Следовательно, местоположение точек, которые ближе к любому сайту выше ℓ чем к самому ℓ , ограничено параболическими дугами. Мы называем эту последовательность параболических дуг – *пляжная линия*. Другим способом визуализировать пляжную линию является следующий. Каждый сайт p_i выше линии заметания определяет полную параболу β_i . Пляжная линия – функция, которая для каждой координаты x проходит через самую низкую точку всех парабол.

Наблюдение 7.5. Пляжная линия – монотонна по x , то есть, каждая вертикальная линия пересекает её точно в одной точке.

Легко видеть, что одна парабола может входить больше чем один раз в пляжную линию. Мы будем волноваться позже о том, сколько частей, там может быть. Заметьте, что *контрольные точки* между другими параболическими дугами, формирующими пляжную линию, лежат на гранях диаграммы Вороного. Это не совпадение: контрольные точки точно прослеживают диаграмму Вороного, в то время как линия заметания перемещается сверху донизу. Эти свойства пляжной линии могут быть доказаны, используя элементарные геометрические аргументы.

Так, вместо того, чтобы поддержать пересечение $Vor(P)$ с ℓ мы поддерживаем линию пляжа, поскольку мы перемещаем нашу ℓ линию заметания. Мы не поддерживаем пляжную линию явно, так как она изменяется непрерывно, поскольку ℓ перемещается. В настоящий момент давайте проигнорируем проблему того, как представить пляжную линию, пока мы не понимаем где и как его комбинаторная структура изменяется. Это происходит, когда новая параболическая дуга появляется на этом, и когда параболическая дуга сжимается к точке и исчезает.

Сначала мы рассмотрим событие, когда новая дуга появляется на пляжной линии. Один из вариантов, где это происходит - когда линия заметания ℓ достигает нового сайта. Парабола, определенная этим сайтом, является сначала выродившейся параболой с нулевой шириной: вертикальный линейный сегмент, соединяю-

щий новый сайт с пляжной линией. В то время как линия заметания продолжает понижаться, новая парабола становится всё более и более широкой. Часть новой параболы ниже старой пляжной линии - теперь часть новой пляжной линии. Рисунок 7.2 иллюстрирует этот процесс. Мы называем событие, когда сталкиваемся с новым сайтом - событием сайта.

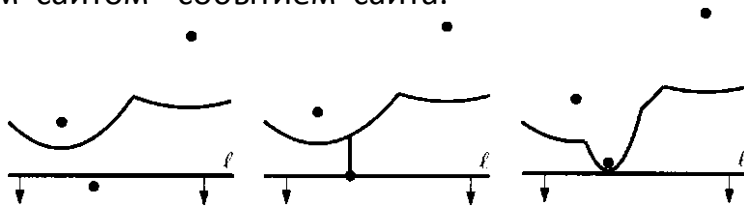
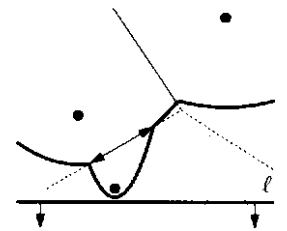


Рисунок 7.2. Новая дуга появляется на пляжной линии, из-за столкновения с сайтом

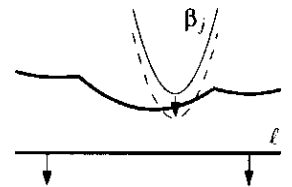
Что происходит с диаграммой Вороного при событии сайта? Вспомните, что контрольные точки на пляжной линии прослеживают грани диаграммы Вороного. При событии сайта появляются две новых контрольные точки, которые начинают прослеживать грани. Фактически, новые контрольные точки совпадают сначала, и затем перемещаются в противоположных направлениях, чтобы проследить один и тот же край. Первоначально, этот край не связан с остальной частью диаграммы Вороного выше линии заметания. Позже - мы будем видеть вскоре точно, когда это произойдет – растущий край столкнется с другим краем, и он становится связанным с остальной частью диаграммы.

Теперь мы понимаем, что происходит при событии сайта: новая дуга появляется на пляжной линии, и новая грань диаграммы Вороного начинает прослеживаться. Действительно ли возможно, чтобы новая дуга появлялась на пляжной линии каким-либо другим способом? Ответ нет:



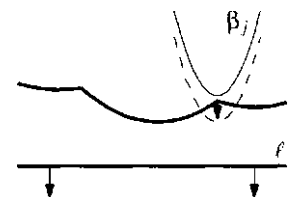
Лемма 7.6. Единственный путь, которым новая дуга может появиться на пляжной линии, через событие сайта.

Доказательство. Предположите для противоречия, что уже существующая парабола β_j , определенная сайтом p_j , прорывается через пляжную линию. Есть два пути, которыми это могло бы произойти.



Первая возможность состоит в том, что β_j прорывается в середине дуги параболы β_i . В момент когда это должно произойти, β_i и β_j соприкасаются, то есть, у них есть точно одна точка пересечения. Позвольте ℓ_y обозначать y -координату линии заметания в момент касания. Если $p_j := (p_{j,x}, p_{j,y})$, тогда парабола β_j даёт

$$\beta_j := y = \frac{1}{2(p_{j,y} - \ell_y)}(x^2 + 2p_{j,x}x + p_{j,x}^2 + p_{j,y}^2 - \ell_y^2)$$



Формула для β_i аналогична, конечно. Используя её для обоих $p_{j,y}$ и $p_{i,y}$ больших чем ℓ_y , легко показать, что это невозможно, что бы у β_i и β_j была только одна точка пересечения. Следовательно, парабола β_j никогда не прорывается в середине дуги другой параболы β_i .

Вторая возможность состоит в том, что β_j окажется между двух дуг. Позвольте этим дугам быть частью парабол β_i и β_k . Позвольте q быть точкой пересечения β_i и β_k , в которой β_j собирается появиться на пляжной линии, и предположить, что β_i находится на пляжной линии слева от q , и β_k находится

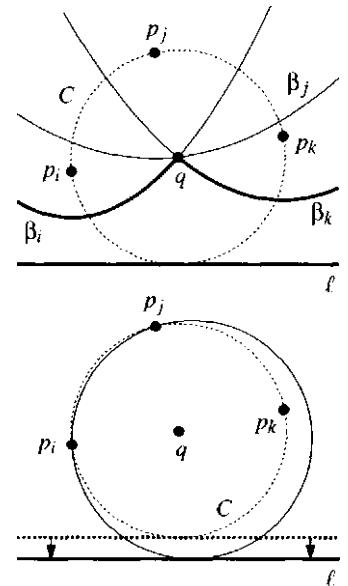


Рисунок 7.3. Ситуация, когда β_j появился бы на пляжной линии, и круг, когда линия заметания продолжилась

справа на пляжной линии от q , как в рисунке 7.3. Тогда есть круг C , который проходит через p_i , p_j , и p_k , сайты, определяющие параболы. Этот круг - также касательная к линии заметания ℓ . Циклический порядок на C , начинающийся в точке касания с ℓ и идущий по часовой стрелке, является p_i , p_j , и p_k , потому

что β_j , как предполагается, появляется промежуточный дуги β_i и β_k . Рассмотрите бесконечно малое движение линии заметания вниз, держа круг C касательной к ℓ ; см. рисунок 7,3. Тогда у C не может быть пустого интерьера и все еще пройти через p_j : или p_i или p_k проникнут через интерьер. Поэтому, по достаточно небольшому соседству q парабола β_j не может появиться на пляжной линии, когда линия заметания понижается, потому что или p_i или p_k будут ближе к ℓ чем p_j .

Непосредственное следствие леммы - то, что пляжная линия состоит из самое большее $2n - 1$ параболических дуг: каждый сайт, с которым сталкиваются, дает начало одной новой дуге и расколу самое большее одной существующей дуги на две, и нет никакого другого способа, которым дуга может появиться на пляжной линии.

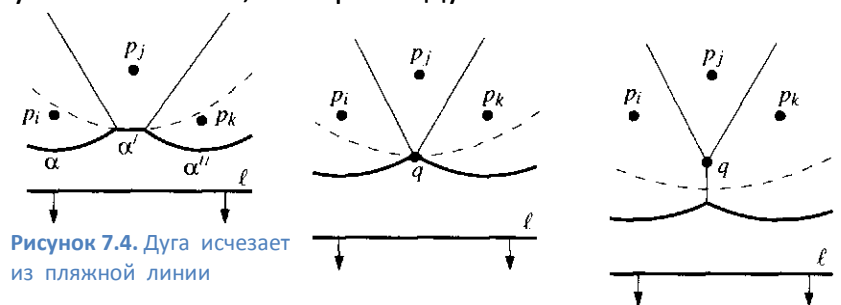


Рисунок 7.4. Дуга исчезает из пляжной линии

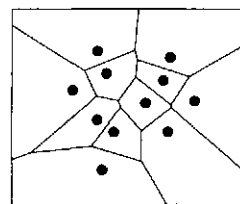
Второй тип события в алгоритме заметания плоскости - то, где существующая дуга пляжной линии сжимается к точке и исчезает, как в рисунке 7.4. Позволим α' быть исчезающей дугой, и позволим α и α'' быть двумя соседними дугами α' прежде, чем она исчезнет. Дуги α и α'' не могут быть частью той же самой параболы; эта возможность может быть исключена таким же образом, как первая возможность в доказательстве Леммы 7.6 была исключена. Следовательно, три дуги α , α' и α'' определены тремя различными сайтами p_i , p_j , и p_k . В момент когда α' исчезает, параболы, определенные этими тремя сайтами, проходят через общую точку q . Точка q равноудалена от ℓ и от каждого из этих трех сайтов. Следовательно, есть круг, проходящий через p_i , p_j , и p_k с q как его центр и чья самая низ-

кая точка находится на ℓ . Не может быть сайта в интерьере этого круга: такой сайт был бы ближе к q , чем q к ℓ , противоречая факту, что q находится на пляжной линии. Из этого следует, что точка q является вершиной диаграммы Вороного. Это не очень удивительно, так как мы заметили ранее, что контрольные точки на пляжной линии прослеживают диаграмму Вороного. Так, когда дуга исчезает из пляжной линии и две контрольные точки встречаются, две грани диаграммы Вороного встречаются также. Мы называем событие, когда линия заметания достигает самой низкой точки круга через три сайта, определяющие последовательные дуги на пляжной линии - *событием круга*. Из вышеупомянутого мы можем вывести следующую лемму.

Лемма 7.7. *Единственный путь, которым существующая дуга может исчезнуть из пляжной линии, через событие круга.*

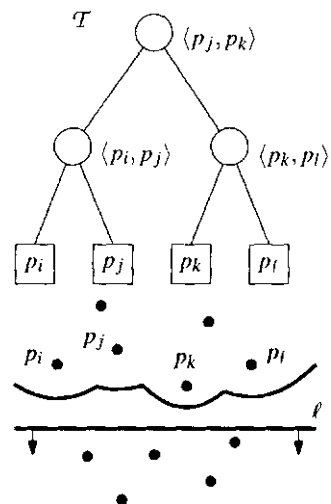
Теперь мы знаем, где и как комбинированное строение пляжной линии изменяется: в событии сайта новая дуга появляется, и в событии круга существующая дуга исключается. Мы также знаем, как это касается диаграммы Вороного в стадии строительства: в событии сайта новая грань начинает расти, и в событии круга две растущих грани встречаются, чтобы сформировать вершину. Остается найти правильные структуры данных, чтобы поддержать необходимую информацию во время развертки. Наша цель состоит в том, чтобы вычислить диаграмму Вороного, таким образом, мы нуждаемся в структуре данных, которая хранит часть диаграммы Вороного, вычисленной к настоящему времени. Мы также нуждаемся в двух 'стандартных' структурах данных для любого алгоритма линии заметания: очередь событий и структуру, которая представляет состояние линии заметания. Здесь последнее строение – представление пляжной линии. Эти структуры данных реализуются следующим образом.

- Мы храним диаграмму Вороного во время строительства в нашей обычной структуре данных для подразбиений, двусвязного граничного списка. Диаграмма Вороного, однако, не является истинным подразбиением как определено в Главе 2: у неё есть грани, которые являются лучами или сплошными линиями, и они не могут быть представлены в двусвязном граничном списке. Во время строительства это не проблема, потому что представление описания пляжной линии затем – позволит получить доступ к соответствующим частям двусвязного граничного списка эффективно во время его строительства. Но после того, как вычисление заканчивается, мы хотим иметь допустимый двусвязный граничный список. С этой целью мы добавляем большой ограничивающий прямоугольник в



нашу сцену, которая является достаточно большой так, чтобы она содержала все вершины диаграммы Вороного. Конечное подразбиение, которое мы вычисляем, затем будет плюсом ограничивающего прямоугольника часть диаграммы Вороного в ней.

- Пляжная линия представлена сбалансированным \mathcal{T} деревом двоичного поиска; это структура состояния. Его листья соответствуют дугам пляжной линии – которая является x -монотонной – упорядоченным способом: крайний левый лист представляет крайнюю левую дугу, следующий лист представляет вторую крайнюю левую дугу, и так далее. Каждый лист μ хранит сайт, который определяет дугу, которую он представляет. Внутренние вершины \mathcal{T} представляют контрольные точки на пляжной линии. Контрольная точка сохранена во внутренней вершине упорядоченным кортежем сайтов $\langle p_i, p_j \rangle$, где p_i определяет параболу, оставленную контрольной точкой, и p_j определяет параболу справа. Используя это представление пляжной линии, мы можем найти за $O(\log n)$ дугу пляжной линии, лежащей выше нового сайта. Во внутренней вершине мы просто сравниваем x координату из нового сайта с x координатой из контрольной точки, которая может быть вычислена из кортежа сайтов и позиции линии заметания в постоянное время. Отметим, что мы явно не храним параболы.

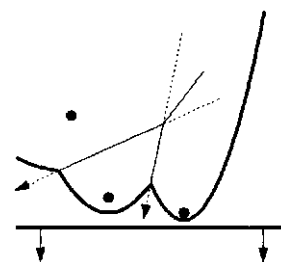


В \mathcal{T} мы также храним указатели на другие две структуры данных, используемые во время развертки. Каждый лист \mathcal{T} , представляет дугу α , хранит один указатель на узел очереди событий, а именно, узел, который представляет событие круга, в котором α исчезает. Этот указатель является nil, если никакого события круга не существует, где α исчезает, или это событие круга еще не было обнаружено. Наконец, у каждого внутреннего узла v есть указатель на полугрань в двусвязном списке граней диаграммы Вороного. Более точно, у v есть указатель на одну из полуграней грани, начерченной контрольной точкой, представленной v .

- Очередь событий Q реализуется как очередь по приоритету, где приоритет события – его y координата. Это позволяет хранить предстоящие события, как уже известные. Для события сайта мы просто храним сайт непосредственно. Для события круга, точка события, которую мы

храним, является самой низкой точкой круга, с указателем на лист в \mathcal{T} , который представляет дугу, которая исчезнет в этом событии.

Все события сайта известны заранее, но события круга - нет. Это приводит нас к одной заключительной проблеме, которую мы должны обсудить, а именно, обнаружение событий круга.

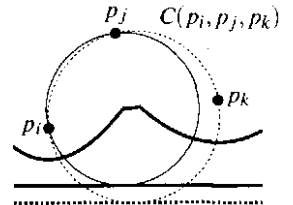


Во время заметания пляжная линия изменяет свою топологическую структуру при каждом событии. Оно может добавить новый триплет (triples) последовательных дуг, которые появятся на пляжной линии, и оно может вызвать исчезновение существующего триплета. Наш алгоритм должен удостовериться, что для каждых трех последовательных дуг на пляжной линии, которые определяют потенциальное событие круга, потенциальное событие сохранено в очереди событий Q . Есть две тонкости в этом. Прежде всего, там может быть последовательный триплет, чьи две контрольные точки не сходятся, то есть, направления, в которые они двигаются, таковы, что они не будут встречаться в будущем; это происходит, когда контрольные точки проходят две средних линии далеко от точки пересечения. В этом случае триплет не определяет потенциальное событие круга. Во-вторых, даже если у триплета есть сходящиеся контрольные точки, соответствующее событие круга не должно иметь место: это может произойти, если триплет исчезает (например из-за появления нового сайта на пляжной линии) прежде, чем событие имело место. В этом случае мы называем событие *ложной тревогой*.

Каким образом алгоритм делает это. При каждом событии, он проверяет все новые триплеты последовательных дуг которые появляются. Например, при событии сайта, мы можем получить три новых триплета: тот, где новая дуга - левая дуга триплета, тот, где это - средняя дуга, и тот, где это - правая дуга. Когда у такого нового триплета есть сходящиеся контрольные точки, событие вставляется в очередь событий Q . Заметьте, что в случае события сайта, триплет с новой дугой, являющейся средним, никогда не может вызывать событие круга, потому что левая и правая дуга триплета прибывает из той же самой параболы, и поэтому контрольные точки должны отклониться. Кроме того, для всех исчезающих триплетов, должно быть проверено, есть ли у них соответствующее событие в Q . Если так, событие - очевидно ложная тревога, и оно должно быть удалено из Q . Это может легко быть сделано, используя указатели, которые мы имеем от листьев в \mathcal{T} к соответствующим событиям круга в Q .

Лемма 7.8. Каждая вершина Вороного обнаружена посредством события круга.

Доказательство. Для вершины Вороного q , позвольте p_i, p_j , и p_k быть этими тремя сайтами, через которые круг $C(p_i, p_j, p_k)$ проходит без сайтов в интерьере. Согласно Теореме 7.4, такой круг и три сайта действительно существуют. Для простоты мы только доказываем случай, где никакие другие сайты не лежат на $C(p_i, p_j, p_k)$, и самая низкая точка $C(p_i, p_j, p_k)$ не является одним из определённых сайтов. Предположите без потери общности, что от самой низкой точки $C(p_i, p_j, p_k)$, по часовой стрелке пересечение $C(p_i, p_j, p_k)$ сталкивается с сайтами p_i, p_j , и p_k в этом порядке.



Мы должны показать, что непосредственно перед тем, как линия заметания достигает самой низкой точки $C(p_i, p_j, p_k)$, есть три последовательных дуги α, α' и α'' на пляжной линии, определенные сайтами p_i, p_j , и p_k . Только тогда будет событие круга иметь место. Считайте линию заметания бесконечно малым количеством прежде, чем она достигнет самой низкой точки $C(p_i, p_j, p_k)$. С тех пор $C(p_i, p_j, p_k)$ не содержит никакие другие сайты внутри или на границе, там существует круг через p_i и p_j который является касательной к линии заметания, и не содержит сайты в интерьере. Таким образом, есть смежные дуги на пляжной линии, определенные p_i и p_j . Точно так же есть смежные дуги на пляжной линии, определенной p_j и p_k , легко видеть, что две дуги, определенные p_j , являются фактически той же самой дугой, и из этого следует, что есть три последовательных дуги на пляжной линии, определенной p_i, p_j , и p_k . Поэтому, соответствующее событие круга находится в Q непосредственно перед тем, как событие имеет место, и вершина Вороного обнаружена.

Мы можем теперь описать алгоритм заметания плоскости подробно. Заметьте, что после того, как все события были обработаны и очередь событий Q пуста, пляжная линия еще не исчезла. Контрольные точки, которые все еще присутствуют, соответствуют полубесконечным граням диаграммы Вороного. Как заявлено ранее, дважды связанный список граней не может представить полубесконечные грани, таким образом, мы должны добавить ограничивающий прямоугольник к сцене, к которой могут быть приложены эти грани. Полная структура алгоритма следующая.

Алгоритм Диаграмма_Вороного(P)

Input. Множество $P := \{p_1, \dots, p_n\}$ точек сайтов на плоскости.

Output. Диаграмма Вороного $Vor(P)$ данная внутри ограничивающего прямоугольника с дважды связанным списком граней \mathcal{D} .

1. Инициализируйте очередь событий Q , со всеми событиями сайта, инициализируйте пустое состояние структуры \mathcal{T} и пустой дважды связанный список граней \mathcal{D} .
2. **while** Q is not empty
3. **do** Получите событие с максимальной y координатой из Q
4. **if** событие – событие сайта, получите сайт p_i
5. **then** HandleSiteEvent(p_i)
6. **else** HandleCircleEvent(γ), где γ это лист \mathcal{T} представляющий дугу, которая должна исчезнуть
7. Внутренние узлы, все еще существующие в \mathcal{T} , соответствуют полубесконечным граням диаграммы Вороного. Вычислите ограничивающий прямоугольник, который содержит все вершины диаграммы Вороного в его интерьере, и приложите полубесконечные грани к ограничивающему прямоугольнику, обновляя дважды связанный список граней соответственно.
8. Пересеките полуграни дважды связанного списка граней, чтобы добавить записи клетки и указатели на и от них.

Процедуры, обработки событий определены следующим образом.

HandleSiteEvent(p_i)

1. **if** \mathcal{T} is empty, **then** вставьте p_i в него (так, чтобы \mathcal{T} состоял из единственного листа - p_i), и возвращайтесь. **else** продолжите с шагов 2 - 5.
2. Ищите в \mathcal{T} дугу α выше p_i , **if** лист, представляющий α имеет указатель на событие круга в Q , **then** это событие круга - ложная тревога и оно должно быть удалено из Q .
3. Замените лист \mathcal{T} который представляет α с поддеревом, имеющим три листа. Средний лист хранит новый сайт p_i , и другие два листа хранят сайт p_j , который был первоначально снабжен α . Сохраните кортежи $\langle p_j, p_i \rangle$ и $\langle p_i, p_j \rangle$ представляющие новые контрольные точки в двух новых внутренних узлах. Выполните операции по перебалансированию \mathcal{T} в случае необходимости.

4. Создайте новую запись полуграни в структуре диаграммы Вороного для грани разделяющей $\mathcal{V}(p_i)$ и $\mathcal{V}(p_j)$, которая будет прослежена двумя новыми контрольными точками.
5. Проверьте триплеты из последовательных дуг, где новая дуга от p_i - левая дуга, чтобы видеть, сходятся ли контрольные точки. Если так, вставьте событие круга в Q , и добавьте указатели между узлом в \mathcal{T} и узлом в Q . Сделайте то же самое для триплета, где новая дуга – правая дуга.

HandleCircleEvent(γ)

1. Удалите лист γ , который представляет исчезающую дугу α от \mathcal{T} . Обновите кортежи, представляющие контрольные точки во внутренних узлах. Выполните операции по перебалансированию \mathcal{T} в случае необходимости. Удалите все события круга, вовлекающие α от Q ; которые могут быть найдены, используя указатели от предшественника и преемника γ в \mathcal{T} . (Событие круга, где α – средняя дуга, в настоящее время обрабатывается, и был уже удален из Q .)
2. Добавьте центр круга, вызывающего событие, поскольку запись вершины в дважды связанный список граней \mathcal{D} хранящий диаграмму Вороного в стадии строительства. Создайте две записи полуграней, соответствующие новой контрольной точке пляжной линии. Установите указатели между ними соответственно. Приложите три новых записи к записям полуграней, которые заканчиваются в вершине.
3. Проверьте новые триплеты из последовательных дуг, у которых есть прежний покинутый сосед α как его средняя дуга, чтобы видеть, сходятся ли две контрольных точки триплета. Если так, вставьте соответствующее событие круга в Q , и указатели набора между новым событием круга в Q , и соответствующим листом \mathcal{T} . Сделайте то же самое для триплета, где прежний правильный сосед – средняя дуга.

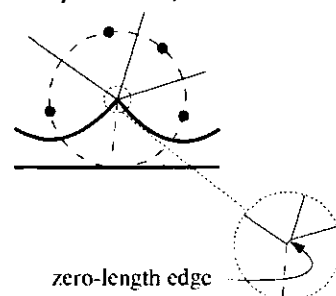
Лемма 7.9. Алгоритм использует $O(n \log n)$ времени и использует $O(n)$ памяти.

Доказательство. Примитивные операции над деревом \mathcal{T} и очередью событий Q , такие как вставка или удаление элемента, берут $O(\log n)$ времени каждый. Примитивные операции с дважды связанным списком граней занимают фиксированное время. Чтобы обращаться с событием, мы делаем фиксированное число таких примитивных операций, таким образом, мы тратим $O(\log n)$ времени, чтобы обработать событие. Очевидно, есть n событий сайта. Что касается числа событий круга, мы замечаем, что каждое такое событие, которое обработано, определяет вершину $Vor(P)$. Отметим, что ложные тревоги удаляются из Q , прежде,

чем они будут обработаны. Они создаются и удаляются, при обработке другого, реального события, и время, которое мы тратим на них, включено в категорию под временем, которое мы тратим, чтобы обработать эти события. Следовательно, число событий круга, которые мы обрабатываем, самое большее $2n - 5$. Время и границы хранения следуют.

Прежде, чем мы заявим окончательный результат этой секции, мы должны сказать несколько слов о выродившихся случаях.

Алгоритм обращается с событиями сверху донизу, таким образом есть вырождение, когда два или больше события лежат на общей горизонтальной линии. Это происходит, например, когда есть два сайта с той же самой y координатой. Эти события могут быть обработаны в любом порядке, когда их x координаты отличны, таким образом, мы можем сломать связи между событиями с той же самой y координатой, но с различными x координатами произвольно. Однако, если это происходит прямо в начале алгоритма, то есть, если y второго события сайта есть та же самая y координата как первое событие сайта, то специальный код необходим, потому что еще нет никакой дуги выше второго сайта. Теперь предположите, что есть точки событий, которые совпадают. Например, будет несколько совпадающих событий круга, когда будет четыре или больше со-круглых сайта, такие, что интерьер круга через них пуст. Центр этого круга – вершина диаграммы Вороного. Степень этой вершины - по крайней мере четыре. Мы могли написать специальный код, чтобы обращаться с такими выродившимися случаями, но нет никакой потребности делать так. Что произойдет, если мы позволим алгоритму обращаться с этими событиями в произвольном порядке? Вместо того, чтобы произвести вершину со степенью четыре, это только произведет две вершины со степенью три в том же самом местоположении, с гранью нулевой длины между ними. Эти выродившиеся грани могут быть удалены при последующей обработке, если потребуется.



Помимо этих вырождений в выборе порядка событий мы можем также столкнуться с вырождениями, обращаясь с событием. Это происходит когда сайт p_i который мы обрабатываем, оказывается, расположен точно ниже контрольной точки между двумя дугами на пляжной линии. В этом случае алгоритм раскалывает любую из этих двух дуг и вставляет дугу для p_i между этими двумя, у одной из которых есть нулевая длина. Эта часть нулевой длины теперь - средняя дуга тройного, которое определяет событие круга. Самая низкая точка этого круга совпадает с p_i . Алгоритм вставляет это событие круга в очередь событий Q , потому что есть три последовательных дуги на пляжной линии, которые определяют это. Когда это событие круга обработано, вершина диаграммы Вороного правильно создана, и дуга нулевой длины может быть удалена позже. Другое вырождение встречается, когда три последовательных



дуги на пляжной линии определены тремя коллинеарными сайтами. Тогда эти сайты не определяют круг, ни событие круга.

Мы приходим к заключению, что вышеупомянутый алгоритм обращается с выродившимися случаями правильно.

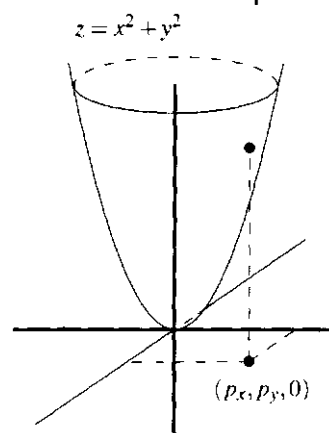
Теорема 7.10. Диаграмма Вороного множества n точек сайтов на плоскости может быть вычислена с алгоритмом линии заметания за время $O(n \log n)$, используя $O(n)$ памяти.

7.3. Примечания и Комментарии

Хотя это вне области этой книги, чтобы дать обширный обзор истории диаграмм Вороного, уместно сделать несколько исторических замечаний. Диаграммы Вороного часто приписываются Дирихле [124] – отсюда название *составления мозаики Дирихле*, который иногда используется – и у Вороного [331, 332]. Они могут уже быть найдены в обработке Декарта космической фрагментации в части III его *Первоначала философий*, изданных в 1644. Также в этом столетии диаграмма Вороного была открыта вновь несколько раз. В биологии это даже произошло дважды за очень короткий период. В 1965 Браун [54] изучил интенсивность деревьев в лесу. Он определил область, *потенциально доступную дереву*, которое было фактически клеткой Вороного того дерева. Один год спустя Мед [241] использовал ту же самую концепцию для растений, называя ячейки Вороного *многоугольниками растения*. К настоящему времени есть внушительное количество литературы относительно диаграмм Вороного и их применений во всех видах областей исследования. Книга Окабе и др. [264] содержит вполне достаточную обработку диаграмм Вороного и их приложений. Мы ограничиваемся в этой секции обсуждением различных аспектов диаграмм Вороного, с которыми сталкиваются в литературе по вычислительной геометрии.

В этой главе мы доказали некоторые свойства диаграммы Вороного, но у этого есть еще много. Например, если Вы соединяете все пары сайтов, со смежными ячейкам Вороного, получающийся набор сегментов формирует триангуляцию множества точек, называемую триангуляцией Делоне. Эта триангуляция, у которой есть некоторые очень хорошие свойства, является темой Главы 9.

Есть красивая связь между диаграммами Вороного и выпуклыми многогранниками. Рассмотрим преобразование, которое наносит на карту точку $p = (p_x, p_y)$, в \mathbb{E}^2 к невертикальной плоскости $h(p): z = 2p_x x + 2p_y y - (p_x^2 + p_y^2)$ в \mathbb{E}^3 . Геометрически, $h(p)$ - плоскость, которая является касательной к параболоиду единицы $\mathcal{U}: z = x^2 + y^2$ в точке вертикально выше $(p_x, p_y, 0)$. Для



множества P точек сайтов на плоскости, позвольте $H(P)$ быть множеством плоскостей, которые являются изображениями сайтов в P . Теперь рассмотрите выпуклый многогранник \mathcal{P} , который является пересечением всех положительных полуплоскостей, определенных плоскостями в $H(P)$, то есть, $\mathcal{P} := \bigcap_{h \in H(P)} h^+$, где h^+ обозначает полуплоскость выше h . Удивительно, если мы проектируем грани и вершины многогранника вертикально вниз на x -плоскость, мы получаем диаграмму Вороного P [140]. См. Главу 11 для более обширного описания этого преобразования.

Мы изучили диаграммы Вороного в их самой элементарной установке, а именно, для множества точек сайтов в Евклидовой плоскости. Первый оптимальный по времени $O(n \log n)$ алгоритм для этого случая был алгоритмом разделяй-и-властвуй, представленным Shamos и Horey [314]; с тех пор много других оптимальных алгоритмов были развиты. Алгоритм зачистки плоскости, который мы описали, придумал Fortune [155]. Оригинальное описание алгоритма Fortune немного отличается от нашего, который следует за интерпретацией алгоритма, данного Guibas и Stolfi [176].

Диаграммы Вороного могут быть обобщены разными способами [16, 264]. Одно обобщение должно указать множество в более многомерных местах. В \mathbb{E}^d максимальная комбинаторная сложность диаграммы Вороного множества точек сайтов n (максимальное количество вершин, граней, и так далее, диаграммы) является $\Theta(n^{\lceil d/2 \rceil})$ [207], и это может быть вычислено в $O(n \log n + n^{\lceil d/2 \rceil})$ оптимальное время [72, 111, 312]. Факт, что двойной из диаграммы Вороного является триангуляция множества сайтов, и связь между диаграммами Вороного и выпуклыми многогранниками как обсуждалось выше, все еще держится в более высоких измерениях.

Другое обобщение касается метрики, которая используется. В L_1 -метрике или манхэттенской метрике, расстоянии между двумя точками p и q определен как

$$\text{dist}_1(p, q) := |p_x - q_x| + |p_y - q_y|;$$

сумма абсолютных разностей в x - и y -координатах. В диаграмме Вороного в L_1 -метрике все грани являются горизонтальными, вертикальными, или диагональными (угол 45° с координатными осями). В более общей L_p -метрике, расстояние между двумя точками p и q определено как

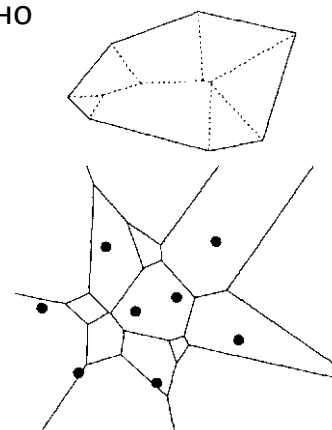
$$\text{dist}_p(p, q) := \sqrt[p]{|p_x - q_x|^p + |p_y - q_y|^p}.$$

Отметьте, что L_2 -метрика - просто Евклидова метрика. Есть несколько статей, имеющих дело с диаграммами Вороного в этих метриках [96, 217, 221]. Можно также определить функцию расстояния, назначая вес на каждый сайт. Теперь расстояние сайта к точке - Евклидово расстояние до точки, плюс ее совокупный вес. Получаю-

щиеся диаграммы называют взвешенными диаграммами Вороного [155]. Вес может также использоваться, чтобы определить расстояние сайта к точке как Евклидовы времена расстояния вес. Диаграммы, основанные на мультипликативно взвешенном расстоянии, также называют взвешенными диаграммами Вороного [17]. Диаграммы мощности [13, 14, 15, 18] являются другим обобщением диаграмм Вороного, где используется другая функция расстояния. Даже возможно пропустить функцию расстояния в целом и определить диаграмму Вороного с точки зрения средних линий между любыми двумя сайтами только. Такие диаграммы называют абстрактными диаграммами Вороного [208, 209, 210, 243].

Другие обобщения касаются формы сайтов. Такие обобщения встречаются, когда диаграмма Вороного используется в целях планирования движения. Важный особый случай - диаграмма Вороного граней простого многоугольника, интерьера к многоугольнику непосредственно. Эта диаграмма Вороного также известна как средняя ось или скелет, и у этого есть применения в анализе формы. Средняя ось может быть вычислена вовремя линейная в числе граней многоугольника [101].

Вместо того, чтобы делить пространство на регионы согласно самым близким сайтам, можно также разделить это согласно k самым близким сайтам, приблизительно для $1 \leq k \leq n - 1$. Диаграммы, полученные таким образом, называют диаграммами Вороного высшего порядка, и для данного k диаграммы назван порядком - k : Диаграмма Вороного [1, 19, 50, 77]. Отметьте, что диаграмма Вороного порядка-1 - не что иное как стандартная диаграмма Вороного. Порядок - $(n - 1)$ диаграмму Вороного также называют диаграммой Вороного самой дальней точки (самого дальнего пункта), потому что ячейка Вороного точки p_i , теперь область точек, для которого p_i , самый дальний сайт. Максимальная сложность диаграммы Вороного порядка- k множества n точек сайтов на плоскости $\Theta(k(n - k))$ [218]. В настоящее время самый известный алгоритм для того, чтобы вычислить диаграмму Вороного порядка- k за время $O(n \log^3 n + k(n - k))$ [1].



7.4. Упражнения

- 7.1. Докажите, что для любого $n > 3$ есть множества точек сайтов на плоскости, таким образом, что у одной из клеток $Vor(P)$ есть $n - 1$ вершин.
- 7.2. Покажите, что Теорема 7.3 подразумевает, что среднее число вершин клетки Вороного - меньше чем шесть.
- 7.3. Покажите, что $\Omega(n \log n)$ является нижней границей в вычислении диаграмм Вороного, сокращая проблему сортировки до проблемы вычисления диаграммы Вороного. Вы можете предположить, что алгоритм диаграммы Вороного

должен быть в состоянии вычислить для каждой вершины диаграммы Вороного свои грани инцидента в циклическом порядке вокруг вершины.

- 7.4. Докажите, что контрольные точки пляжной линии, как определено в Разделе 7.2, прослеживают грани диаграммы Вороного, в то время как линия заметания перемещается сверху донизу.
- 7.5. Дайте пример, где парабола, определенная некоторым сайтом p_i , вносит больше чем одну дугу в пляжную линию. Вы можете дать пример, где это вносит линейное число дуг?
- 7.6. Дайте пример шести сайтов, таким образом, что алгоритм зачистки плоскости сталкивается с шестью событиями сайта перед любым из событий круга. Сайты должны лечь в общем положении: никакие три сайта на линии и никакие четыре сайта на круге.
- 7.7. Контрольные точки пляжной линии всегда перемещаются вниз, когда линия заметания перемещается вниз? Докажите это или дайте контрпример.
- 7.8. Напишите процедуру, чтобы вычислить достаточно большой ограничивающий прямоугольник из неполного дважды связанного списка граней и дерева \mathcal{T} после того, как зачистка закончена. Коробка должна содержать все сайты и все вершины Вороного.
- 7.9. Напишите процедуру, чтобы добавить все записи ячеек и соответствующие указатели на неполный дважды связанного списка граней после того, как ограничивающий прямоугольник был добавлен. Таким образом, заполните детали линии 8 из Алгоритма `VoronoiDiagram`.
- 7.10. Пусть P будет множеством n точек на плоскости. Дайте $O(n \log n)$, алгоритм времени, чтобы найти две точки в P , которые являются самыми близкими вместе. Покажите, что Ваш алгоритм правилен.
- 7.11. Пусть P будет множеством n точек на плоскости. Дайте $O(n \log n)$ алгоритм времени, чтобы найти для каждой точки p в P другая точка в P , который является самым близким к этому.
- 7.12. Пусть диаграмма Вороного множества точек P быть сохраненными во дважды связанного списка граней в ограничивающем прямоугольнике. Дайте алгоритм, чтобы вычислить все точки P , которые лежат на границе выпуклого корпуса P , ввремя линейного в размере выхода. Предположите, что ваш алгоритм получает как его вход указатель на запись некоторой полуграни, происхождение которого находится на ограничивающем прямоугольнике.
- 7.13. * В назначении Вороного моделируют товары или услуги, которые потребители хотят приобрести, имеют ту же самую рыночную цену на каждом сайте. Предположите дело обстоит не так, и что цена пользы на сайте p_i это w_i , Торговые области сайтов теперь соответствуют ячейкам во взвешенной диаграмме Вороного сайтов (смотри Раздел 7.3), где сайт p_i , имеет совокупный вес w_i . Обобщите алгоритм линии заметания Раздела 7.2 к этому случаю.



7.14. * Предполагают, что нам дают подразделение плоскости в n выпуклые регионы. Мы подозреваем, что это подразделение – диаграмма Вороного, но мы не знаем сайты. Развейте алгоритм, который находит ряд n точек сайтов, диаграмма Вороного которых – точно данное подразделение, если такой набор существует.