Voronoi Diagrams
The Post Office Problem

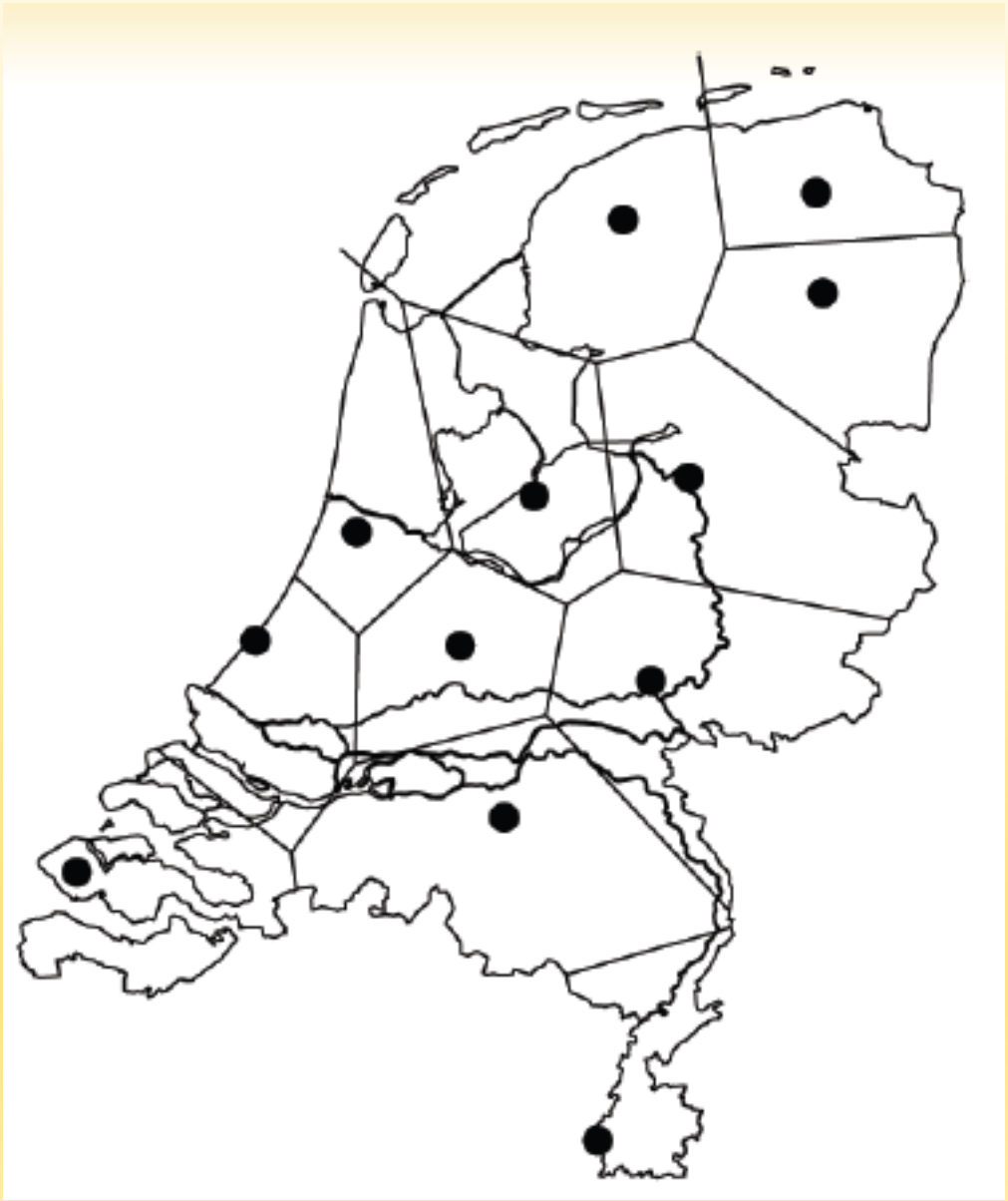
Nafiseh Mohkam
Yazd University

Computer Science 2013

رئوس مطالب

1. تعاریف و خواص پایه
2. محاسبه ورونوی دیاگرام
3. ورونوی دیاگرام پاره خطها
4. Farthest-Point ورونوی دیاگرام‌ها

مقدمه



تعریف

- مدل ورونوی assignment: مدلی که هر نقطه را به نزدیکترین سایت اختصاص می دهد.
- ورونوی دیاگرام: زیربخشی که از مدل ورونوی assignment بدست می آید.

کاربرد

- جغرافیای اجتماعی
- فیزیک
- ستاره شناسی
- رباتیک
- مثلث بندی دلونی

رئوس مطالب

1. تعاریف و خواص پایه
2. محاسبه ورونوی دیاگرام
3. ورونوی دیاگرام پاره خطها
4. Farthest-Point ورونوی دیاگرام‌ها

تعاریف اولیه

یادآوری

فاصله اقلیدسی بین دو نقطه p و q را با dist(p, q) نشان می دهیم:

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

ورونوی دیاگرام (vor(P))

- مجموعه n نقطه سایت مجزا درصفحه داریم. $P = p_1, p_2, \ldots, p_n$
- صفحه طبق قاعده زیر به n سلول تقسیم می‌شود:

$$q \in p_i \quad \text{if and only if} \quad \text{dist}(p, p_i) < \text{dist}(p, p_j) \quad \forall\, p_j \in P,\ j \neq i$$

سلول ورونوی ونیمساز

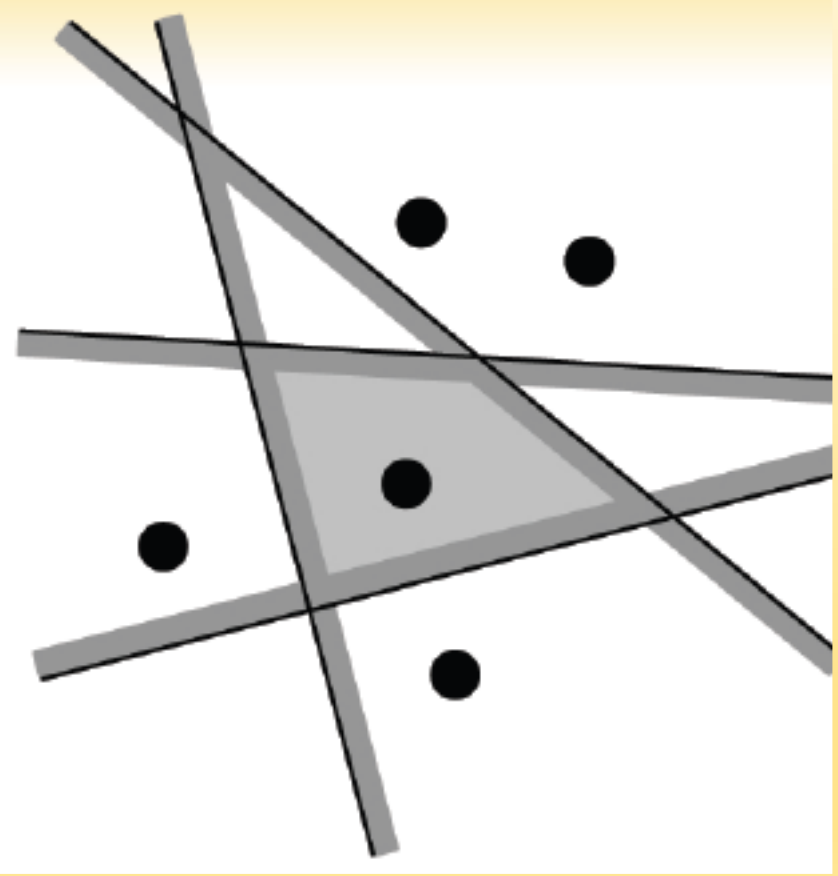
سلول p_i ورونوی

سلول vor(P) که متناظر است با سایت p_i با $V(p_i)$ نمایش داده می‌شود.

نیمساز p وq

- عمودمنصف پاره خط $\overline{pq}$
- h(p, q): نیم‌صفحه باز که شامل p است.
- h(q,p): نیم‌صفحه باز که شامل q است.
- $r \in h(p, q)$ if and only if $dist(r, p) < dist(r, q)$

مشاهده



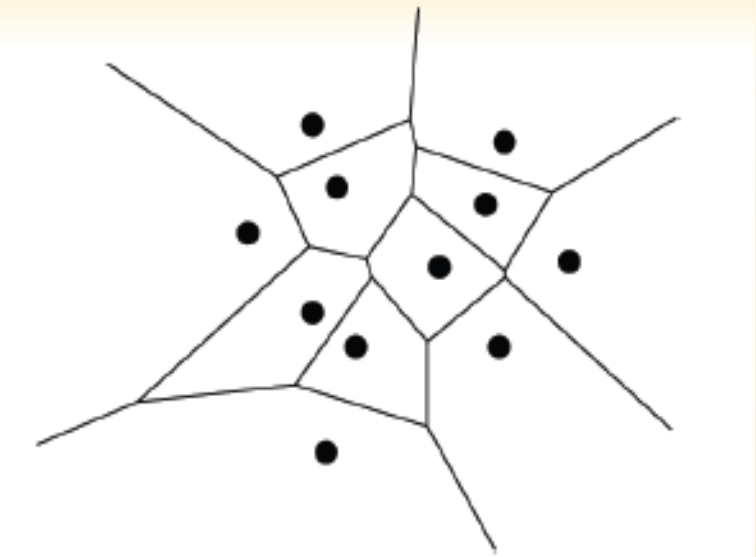
مشاهده 7.1

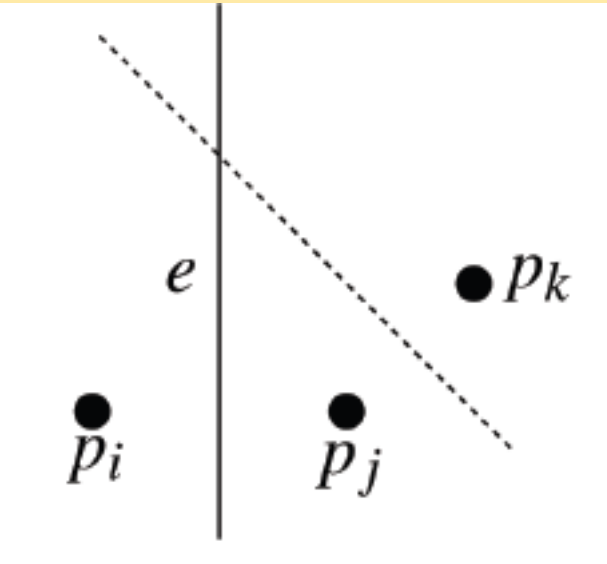
$V(p_i) = \bigcap_{1 \le j \le n, j \ne i} h(p_i, p_j)$

نتیجه

ورونوی دیاگرام زیربخش مسطحی است که یال هایش مستقیم هستند.

پاره خط یا نیم صفحه





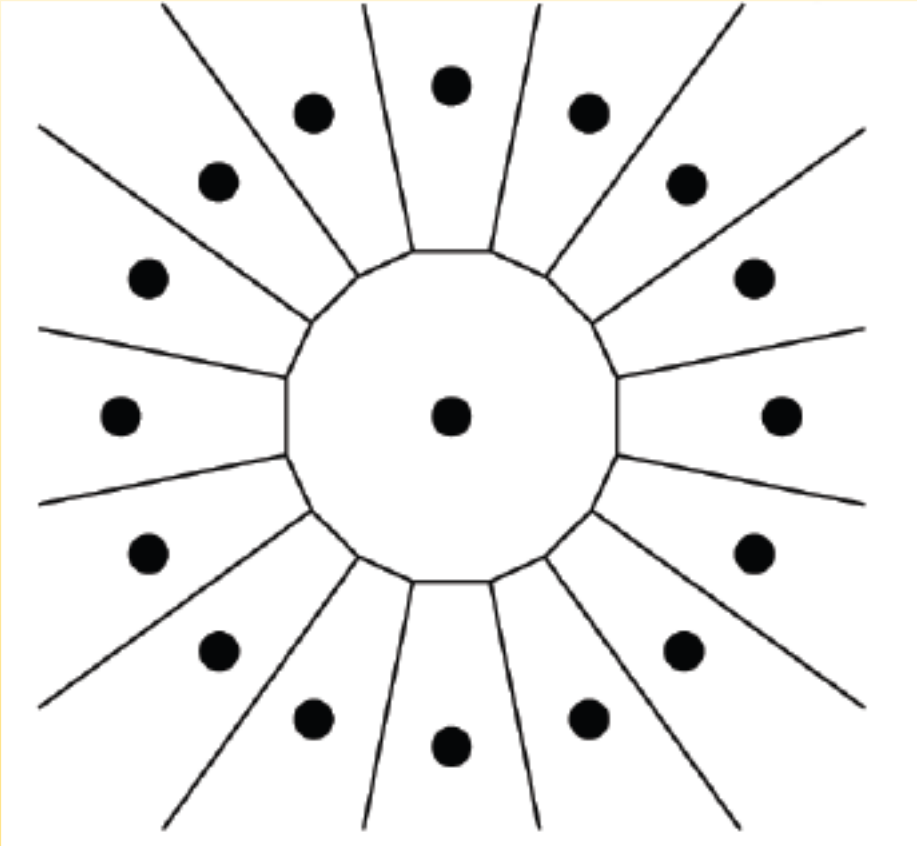
قضیه 7.2

P مجموعه n نقطه سایت در صفحه داده شده‌است. اگر همه سایت‌ها در خط مستقیم واقع شده‌اند پس Vor(P) شامل n-1 نیم صفحه است. در غیر این صورت، Vor(P) پیوسته است و یال‌هایش پاره‌خط یا نیم‌خط هستند.

اثبات

- قسمت اول آسان است.
- یال‌های Vor(P) پاره خط یا نیم خط هستند.
- Vor(P) متصل است.

تعداد یال‌ها و رئوس

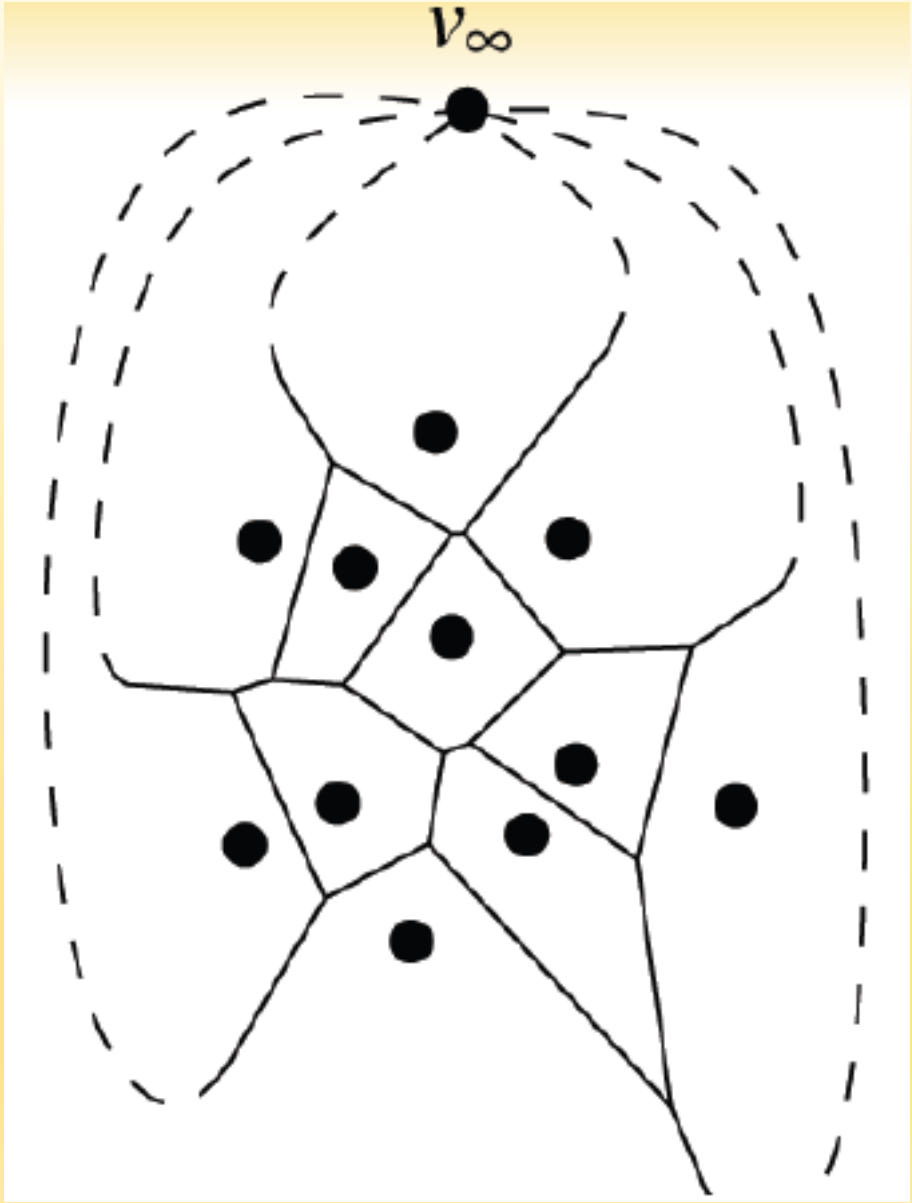


قضیه 7.3

برای $n>3$ رأس در ورونوی دیاگرام مجموعه n سایت نقطه در صفحه حداکثر $2n-5$ و تعداد یال‌ها حداکثر $3n-6$ است.

اثبات (قسمت اول)

همه سایت ها درخط مستقیم واقع شده اند.

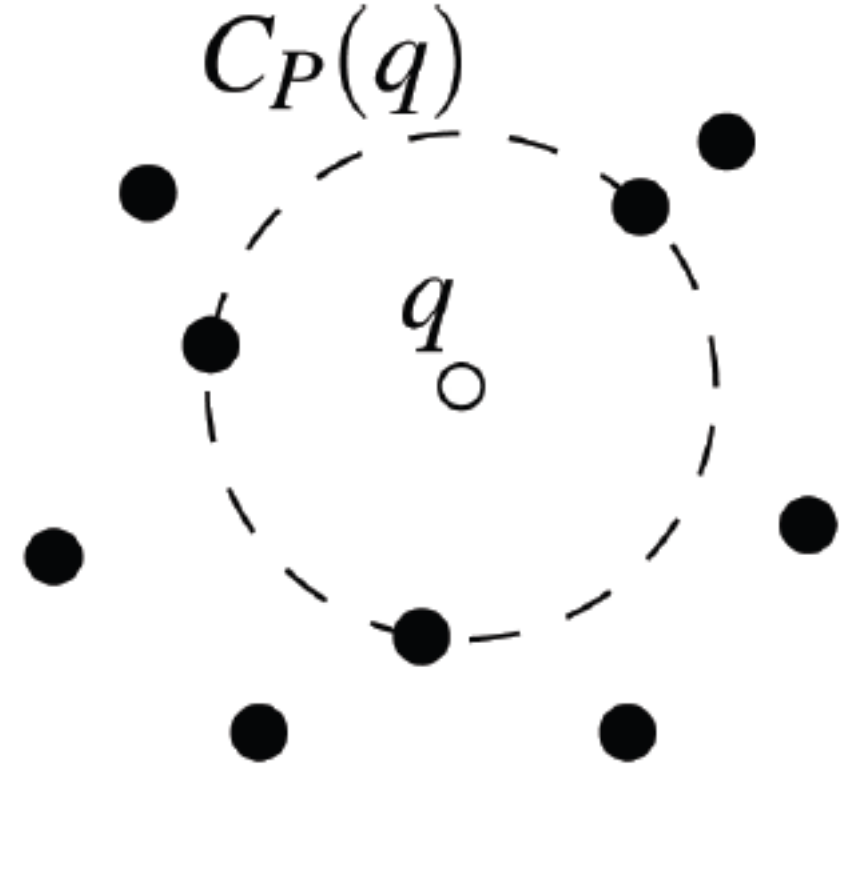


تعداد یال‌ها و رئوس

اثبات (قسمت دوم)

- $m_v - m_e + m_f = 2$
- $(n_v + 1) - n_e + n = 2$
- $2n_e \geq 3(n_v+1)$

بزرگترین دایره خالی

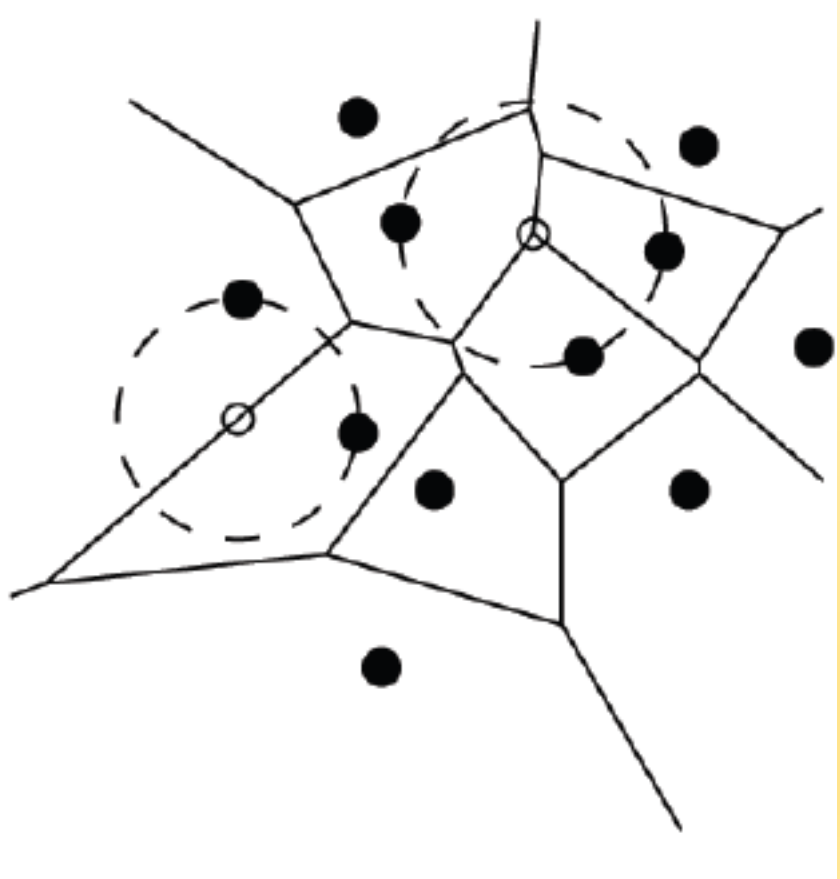


تعریف $(C_p(q))$

بزرگترین دایره خالی از q با توجه به P:

بزرگترین دایره خالی با مرکز q و شامل هیچ سایت دیگری از P در داخلش نیست.

شناخت ورونوی دیاگرام



قضیه 7.4

برای ورونوی دیاگرام Vor(P) از مجموعه نقاط P روابط زیر برقرار است:

I. نقطه q رأس Vor(P) است اگر و تنها اگر بزرگترین دایره خالی $C_p(q)$ آن شامل سه یا بیشتر سایت روی مرز آن باشد.

II. عمودمنصف بین سایت‌های p_i و p_j یال Vor(P) را تعریف می کند اگر و تنها اگر نقطه q روی عمودمنصف وجود داشته باشد چنانچه$C_p(q)$ شامل هر دوتای p_i وp_j، و نه سایت دیگری، روی مرزش است.(یعنی فقط شامل p_i و p_j روی مرزش است)

رئوس مطالب

1. تعاریف و خواص پایه
2. محاسبه ورونوی دیاگرام
3. ورونوی دیاگرام پاره خطها
4. Farthest-Point ورونوی دیاگرام‌ها

روش‌های مختلف

روش‌های محاسبه ورونوی دیاگرام

1. روش ساده ⟸ پیچیدگی: $O(n^2 logn)$
2. الگوریتم Fortune ⟸ پیچیدگی: $O(nlogn)$

چرا الگوریتم Fortune بهینه است؟

- مسئله مرتب سازی n عدد حقیقی به مسئله محاسبه ورونوی دیاگرام کاهش پذیر است.
- بنابراین ورونوی دیاگرام می بایست در بدترین حالت زمان $\Omega(nlogn)$ داشته باشد.

مروری بر الگوریتم Furtune

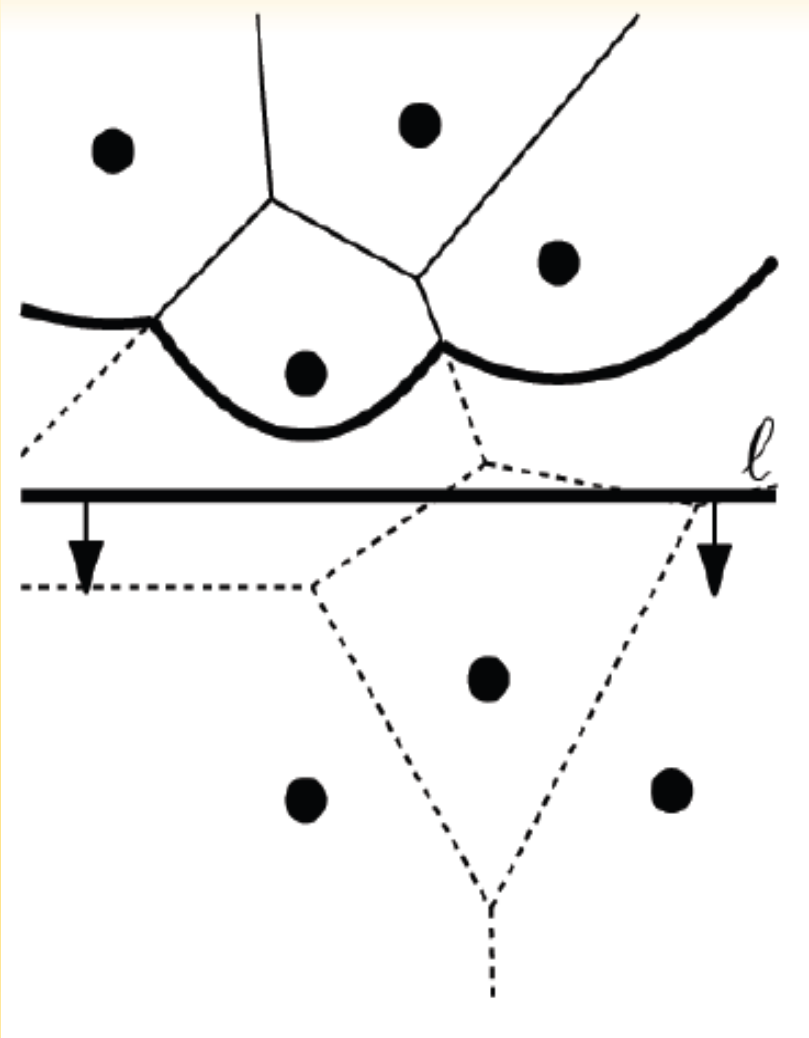
مرور

از استراتژی جاروب صفحه (plane sweep) در الگوریتم Fortune استفاده می‌کنیم.

مسئله

قسمتی از Vor(P) بالای ℓ نه تنها به سایت‌هایی که بالای ℓ قرار می‌گیرند بستگی دارد، بلکه به سایت‌های زیر ℓ نیز بستگی دارد.

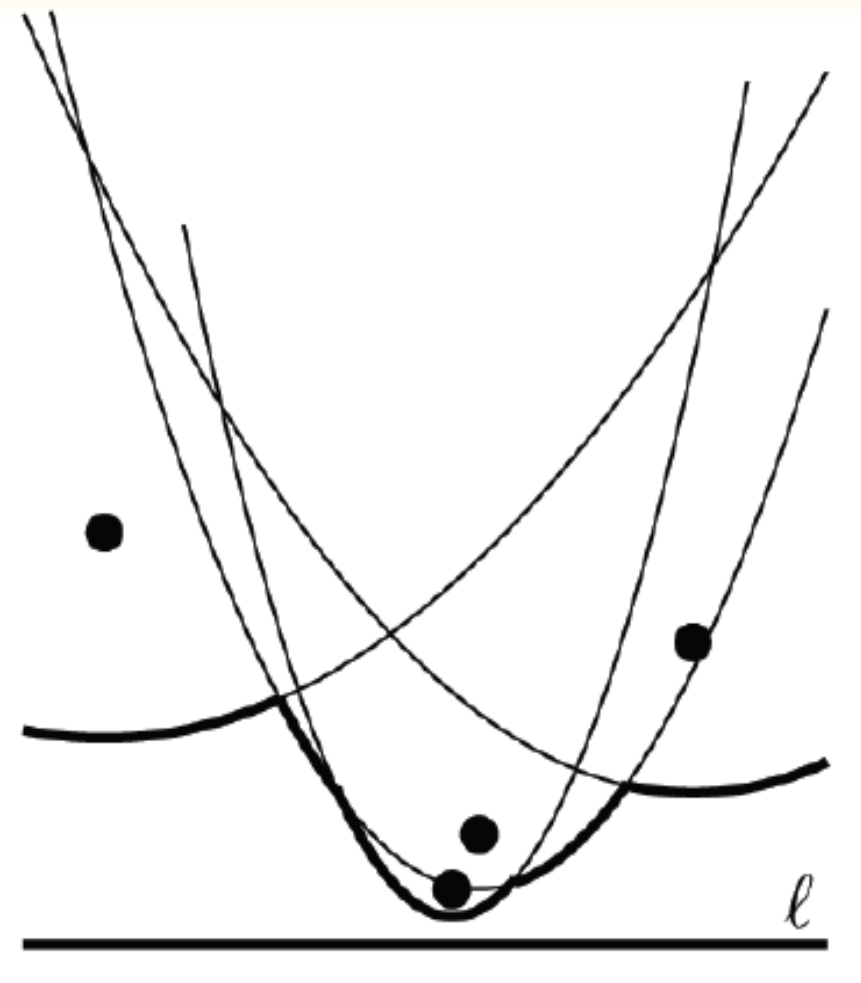
روش‌های مختلف



راه حل

- ℓ^+: نیم‌صفحه بسته بالای ℓ
- سوال: چرا قسمت بالای ورونوی دیاگرام بالای sweep line تغییر نمی‌کند؟ یا به عبارتی برای هر نقطه $q \in \ell^+$ آیا می‌دانیم نزدیکترین سایت‌ها به q کدام است؟
- نزدیکترین سایت به q نمی‌تواند زیر ℓ قرار بگیرد اگرq حداقل نزدیک به سایتی مثل $p_i \in \ell^+$ باشد به‌طوریکه q نسبت به ℓ قرار دارد.

خط ساحلی (Beach line)



خط ساحلی (Beach line)

مکان هندسی نقاطی که از هر سایت بالای ℓ نزدیکتر به خود ℓ هستند با کمان‌های سهمی واری محدود شده‌اند. این دنباله کمان‌های سهمی‌وار را خط ساحلی (Beach line) می‌نامیم.

مشاهده

مشاهده 7.5

خط ساحلی x-مونوتون است، یعنی، هر خط عمودی دقیقا یک نقطه از آن را قطع می‌کند.

نکته

- breakpointهای بین کمان‌های سهمی‌وار متفاوت خط ساحلی روی یال‌های ورونوی دیاگرام قرار می‌گیرند.
- Breakpointها، دقیقا یال‌های ورونوی دیاگرام را دنبال می کنند.

کمان جدید

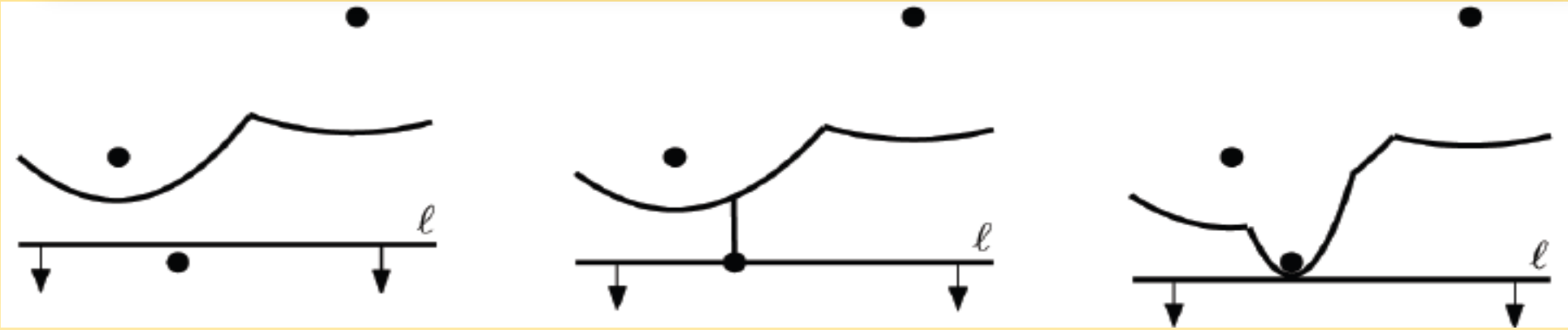
چه موقع کمان جدید روی خط ساحلی ظاهر می‌شود؟

یک مورد: زمانی است که sweep line ℓ به سایت جدید می‌رسد.

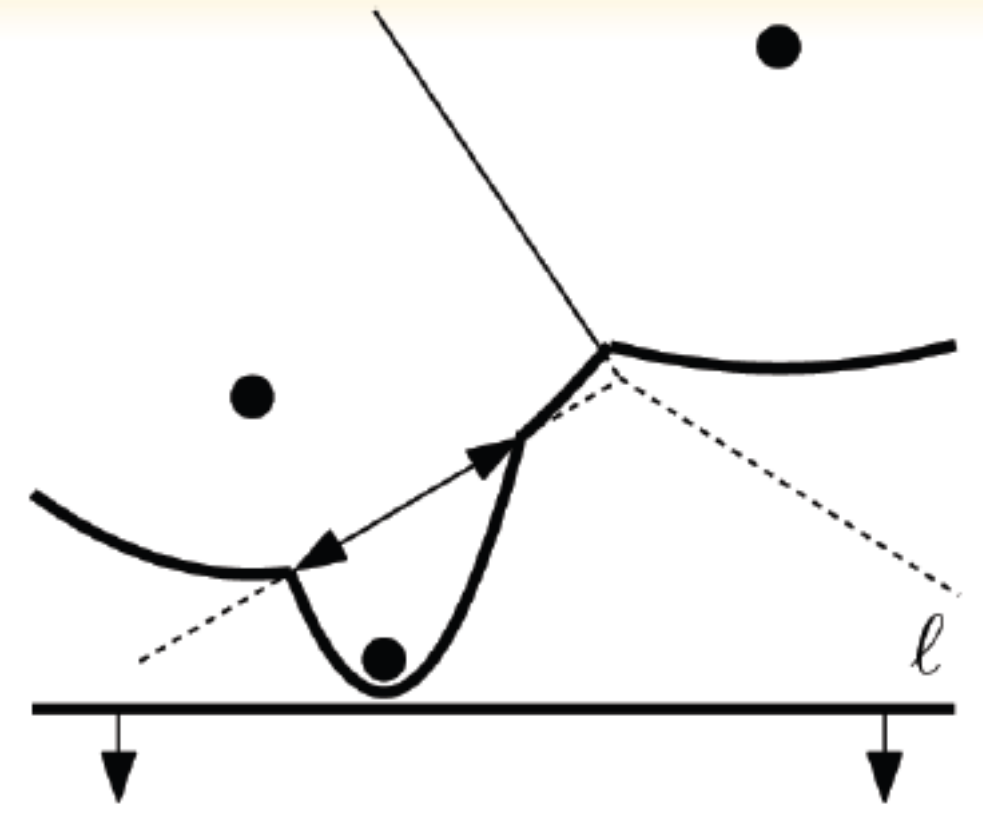
- سهمی با پهنای صفر
- Sweep line ℓ به سمت پایین حرکت می‌کند ⟸ سهمی جدید پهن وپهن تر می‌شود.

Site Event

جایی که با سایت جدید مواجه می‌شود.



کمان جدید



برای ورونوی دیاگرام در Site Event چه اتفاقی می‌افتد؟

1. در ابتدا دو breakpoint جدید بر هم منطبق هستند.
2. سپس در مسیر متفاوتی برای دنبال کردن یال یکسانی حرکت می‌کنند.

نکته

در ابتدا این یال به بقیه ورونوی دیاگرام، بالای sweep line متصل نیست.

لم کمان جدید

لم 7.6

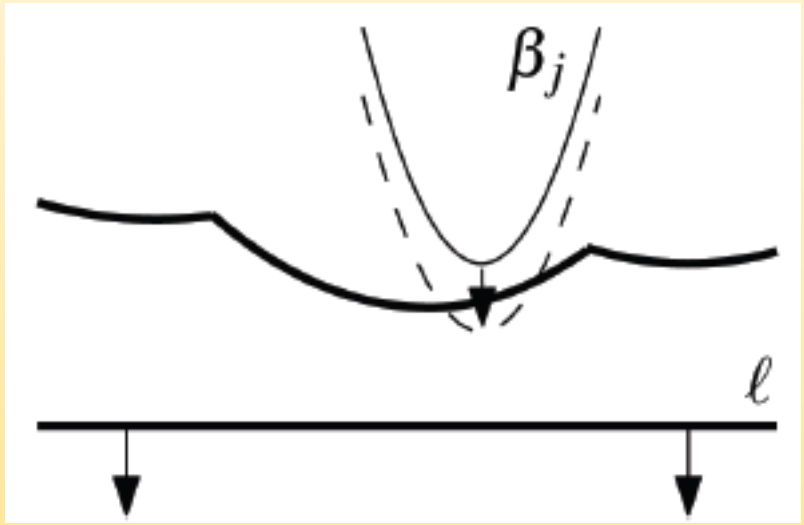
تنها حالتی که کمان جدید می‌تواند روی خط ساحلی به وجود آید، از طریق Site Event است.

اثبات

(برهان خلف) در حال حاضر سهمی β_j توسط سایت p_j تعریف شده‌است که وسط خط ساحلی را می شکند.

دو حالت می‌تواند اتفاق بیفتد:

لم کمان جدید



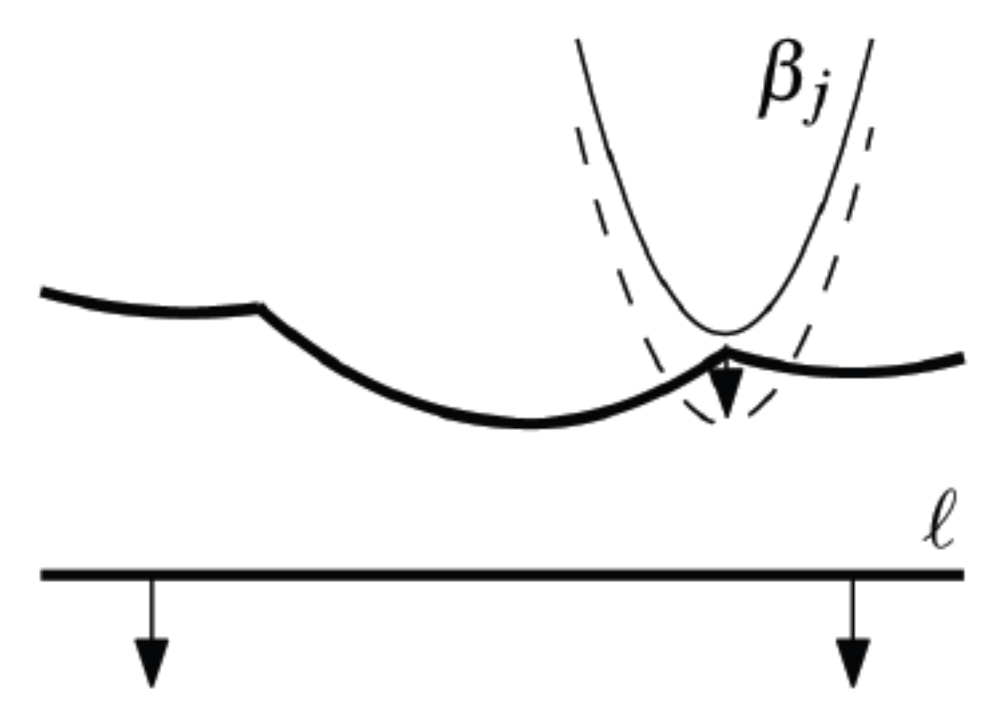
اثبات (قسمت 1)

- β_j میانه کمان سهمی β_i را می‌شکند.
- ℓ_y: مختصات y از sweep line در لحظه مماس را نشان می‌دهد.
- $P_j := (P_{j,x}, P_{j,y})$
- $\beta_j := y = \frac{1}{2(P_{j,y}-\ell_y)}(x^2 - 2P_{j,x}x + P_{j,x}^2 - P_{j,y}^2 - \ell_y^2)$

نتیجه

$P_{j,y}$ و $P_{i,y}$ بزرگتر از ℓ_y هستند $\Leftarrow$ امکان ندارد که β_i و β_j یک نقطه تقاطع داشته باشند.

لم کمان جدید



اثبات (قسمت2)

β_j بین دو کمان β_i و β_k ظاهر شود.

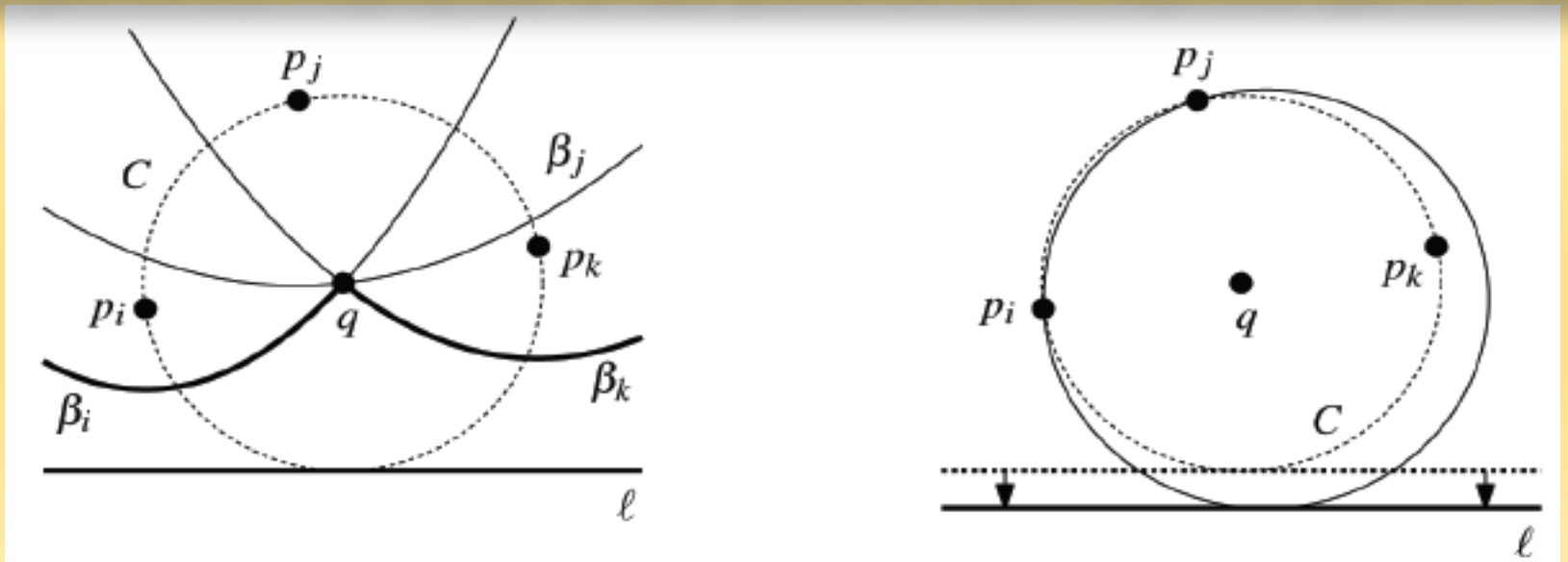
لم کمان جدید

اثبات (قسمت2)

❖ حرکت بی‌نهایت کوچک sweep line به سمت پایین، در حالی که دایره C با ℓ مماس باقی بماند را درنظر داشته باشید. p_j یا p_j به داخل نفوذ می‌کنند.

نتیجه

❖ وقتی seep line به سمت پایین حرکت می‌کند، به طور کامل همسایگی کوچک q سهمی β_j نمی‌تواند روی خط ساحلی دیده شود.



Circle Event

Circle Event

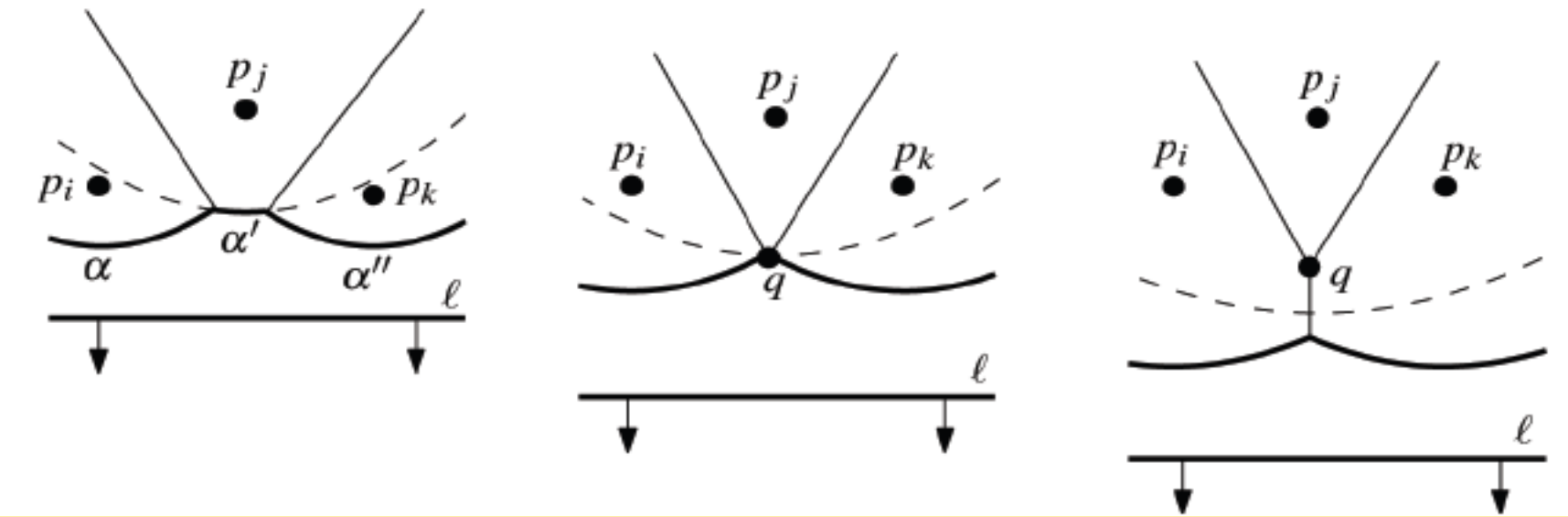
جایی که sweep line به پایین‌ترین نقطه دایره گذرنده از سه سایت تعریف شده کمان‌های متوالی روی خط ساحلی یک circle event می‌رسد.

فرض

α': کمان در حال ناپدید شدن

α و α'': دو کمان همسایه α'

q: فاصله مساوی از ℓ و هر سه کمان دارد.



Circle Event لم

لم 7.7

تنها حالتی که کمان جدید می‌تواند روی خط ساحلی ناپدید شود، ازطریق circle event است.

مرور

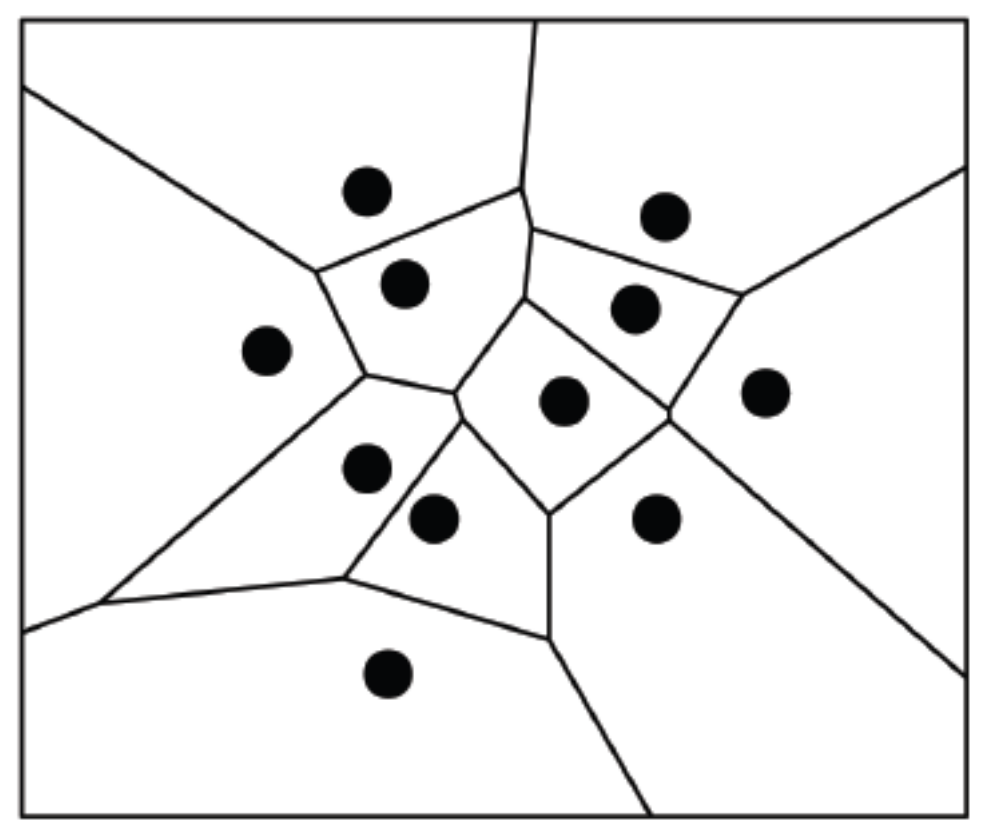
رویدادها (events)

1. **Site event**
   - ❖ کمان جدید پدیدار می‌شود.
   - ❖ یال جدید شروع به بزرگ شدن می‌کند.
2. **Circle event**
   - ❖ یکی از کمان‌های موجود حذف می‌شود.
   - ❖ دو یال در حال رشد در رأس به هم می‌رسند.

سوال

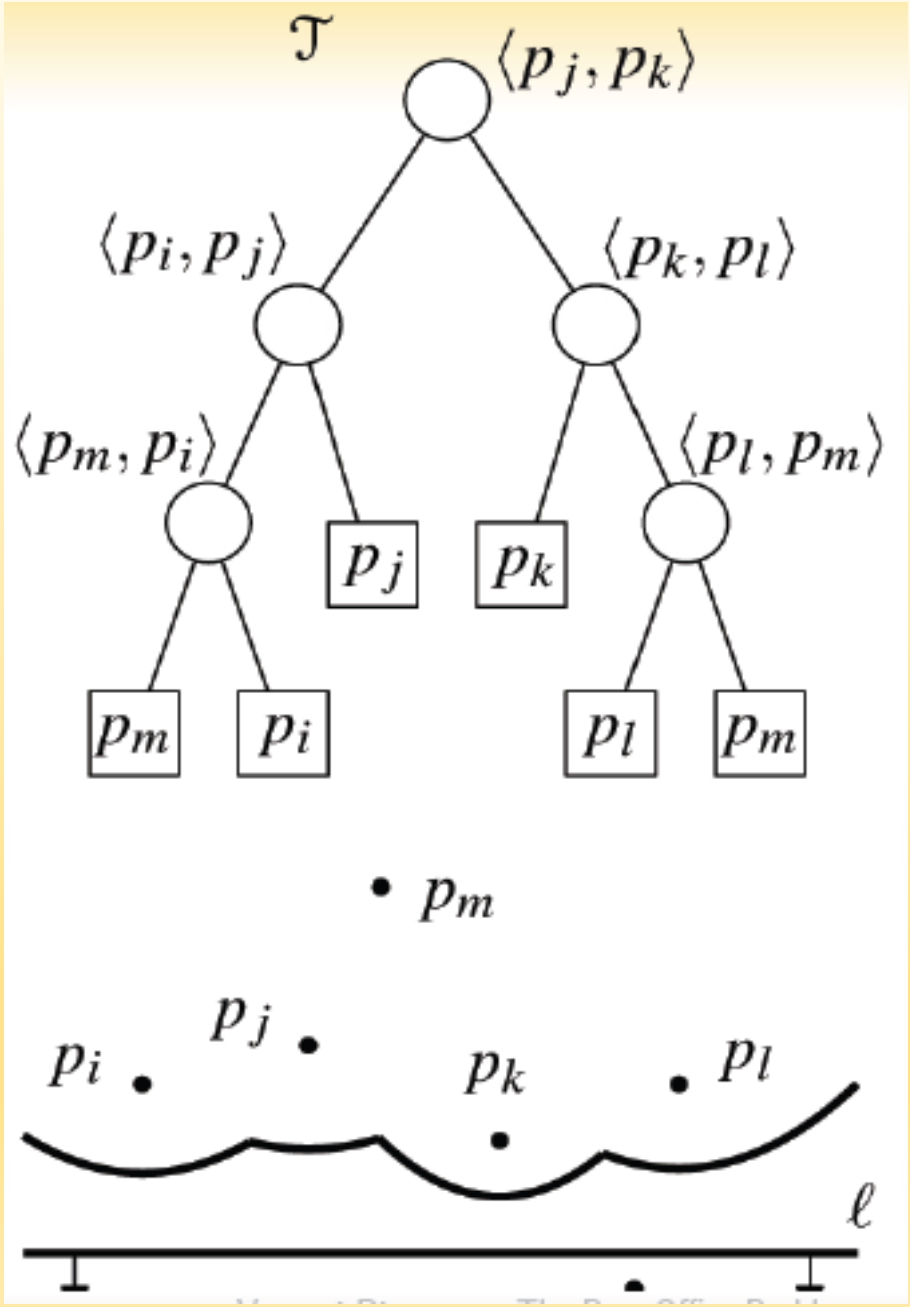
چه چیز دیگری نیاز داریم انجام دهیم؟

ساختمان داده



ورونوی دیاگرام

- ساختمان داده: Doubly-connected edges list (DCEL)
- آیا مشکلی برای ذخیره وجود دارد؟



ساختمان داده

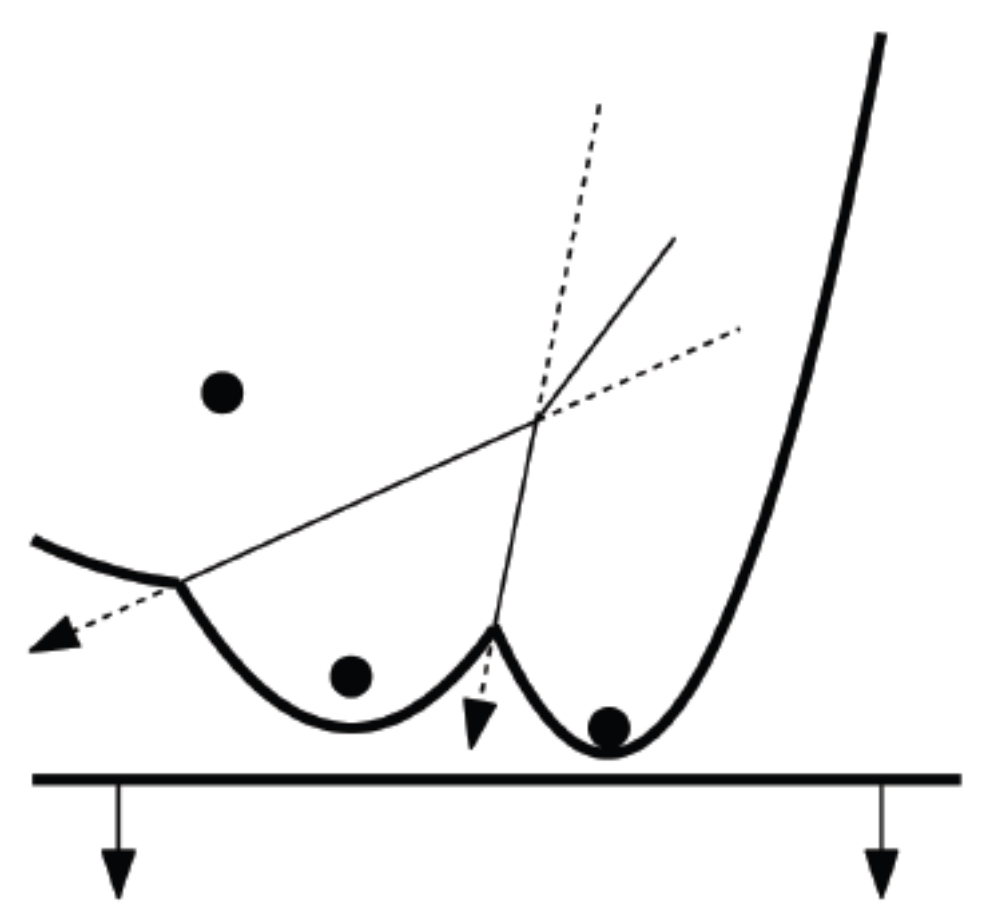
خط ساحلی (Beach line)

- ساختمان داده: درخت جست‌وجوی دودویی T ؛ این ساختار وضعیت (status) است.
- برگ‌های T: کمان‌ها
- نودهای داخلی T: Break pointها

صف رخداد (Event Queue)

ساختمان داده: صف الویت (الویت مختصات y مربوط به event)

لم کمان جدید



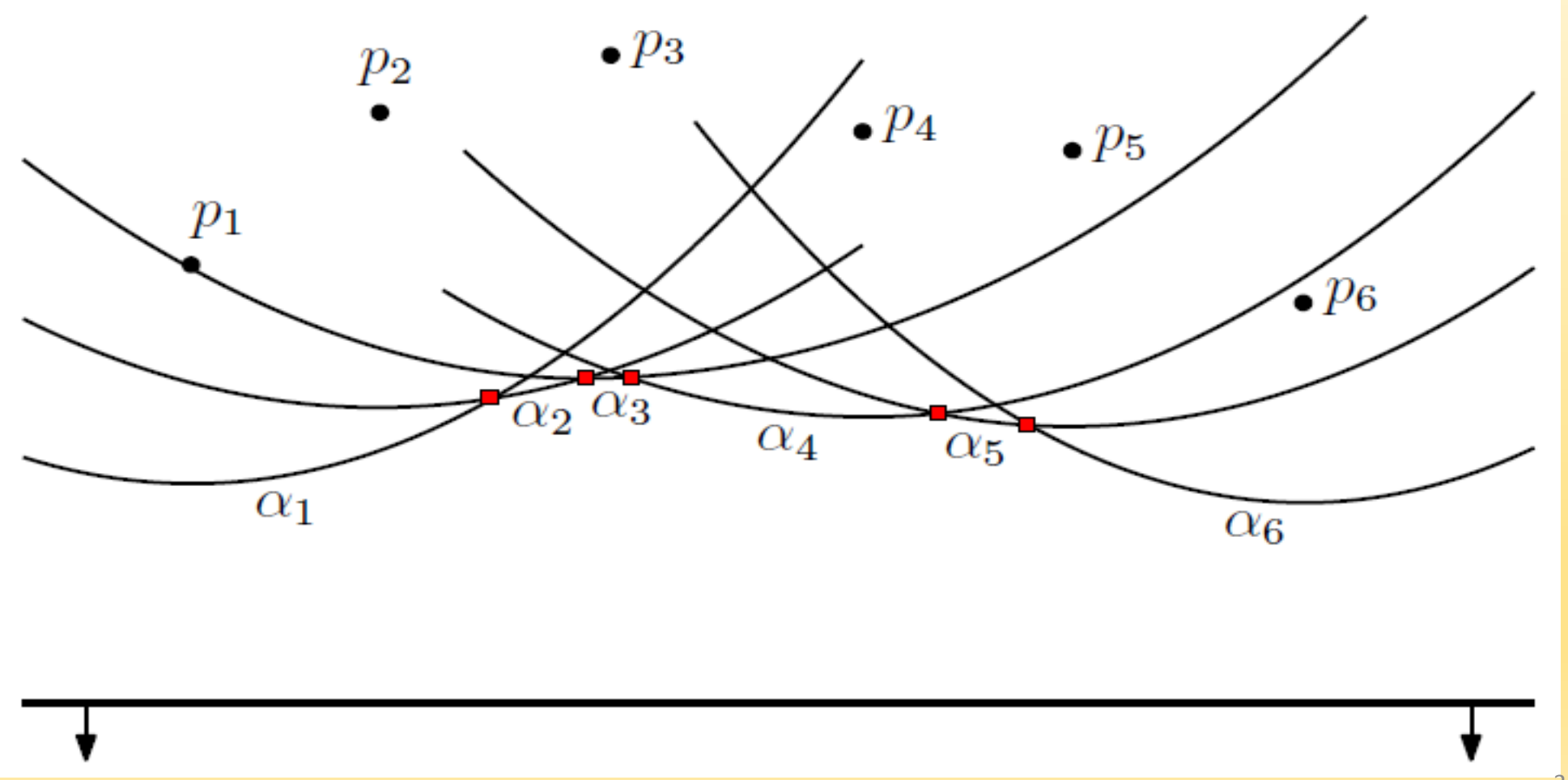
Event Cases

1. سه‌تایی‌هایی متوالی که دو تا Break point دارند که در مسیری حرکت می‌کنند که همگرا نمی‌شوند.
2. Circle event ی که قبل از اینکه به event تعیین شده برسد ناپدید شود.

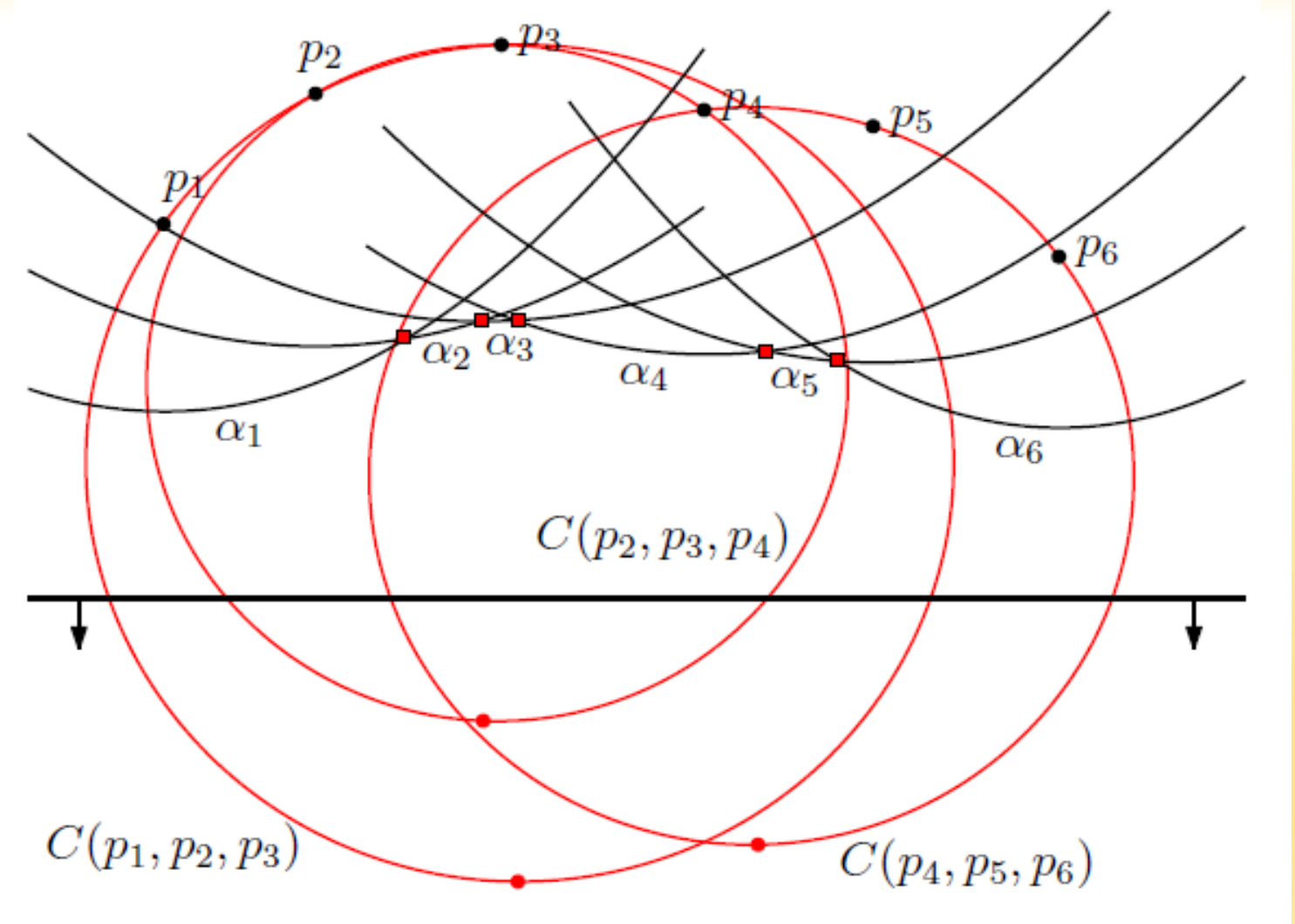
سوال

الگوریتم با Circle event چگونه رفتار می‌کند؟

Circle Event شناسایی



Circle Event شناسایی



شناسایی circle event

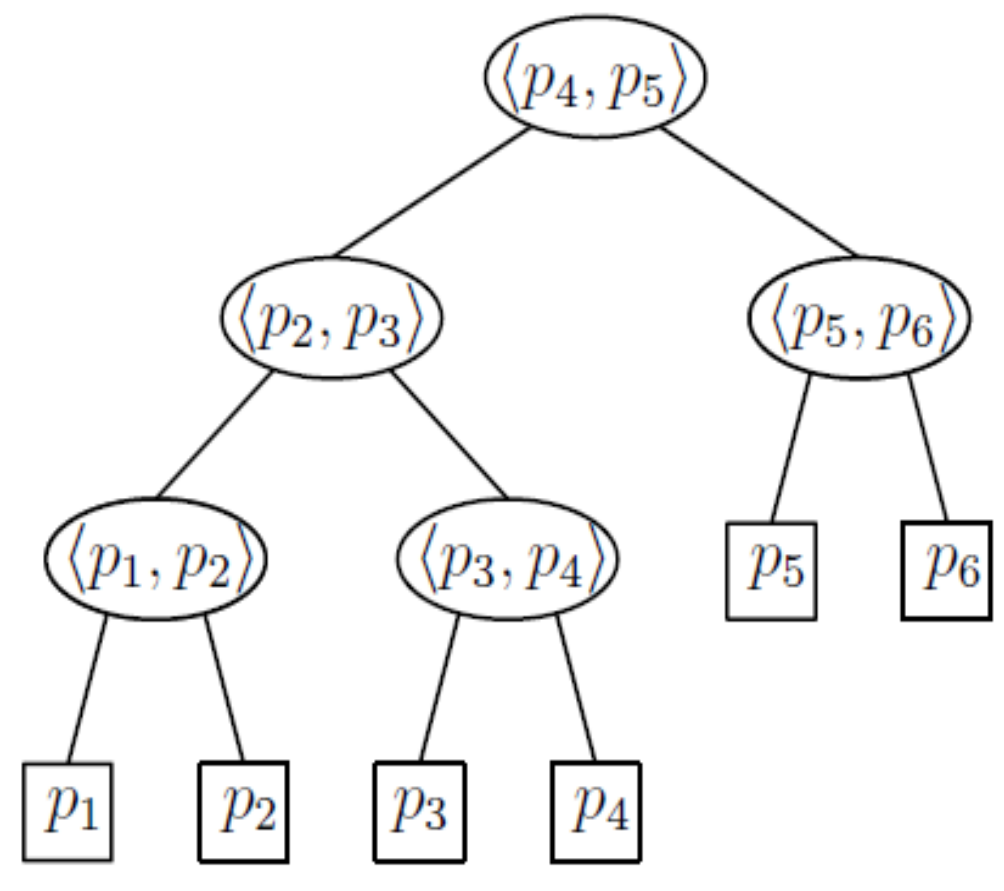
چه موقع circle event رخ می‌دهد؟

اگر sweep line به پایین‌ترین نقطه دایره خالی تعریف شده توسط سه سایت که کمان سهمی متوالی روی خط ساحلی دارند برسد، circle event رخ می‌دهد.

حال باید چه کاری انجام دهیم؟

مطمئن خواهیم شد که پایین ترین نقطه از دایره‌ای که توسط هر سه سایت که سه کمان سهمی متوالی روی خط ساحلی دارند و زیر sweep line قرار می‌گیرد، در صف event ذخیره شود.

شناسایی circle event



circle event در ساختار status

در ساختار status همه سه تایی‌های متوالی کمان‌های سهمی که می‌توانند circle event را بدهند، را می‌توانیم ببینیم.

شناسایی circle event

آیا همه circle eventها صحیح هستند؟

ممکن است circle eventی را در صف event Q ذخیره کنیم ولی هیچ وقت رخ ندهد...

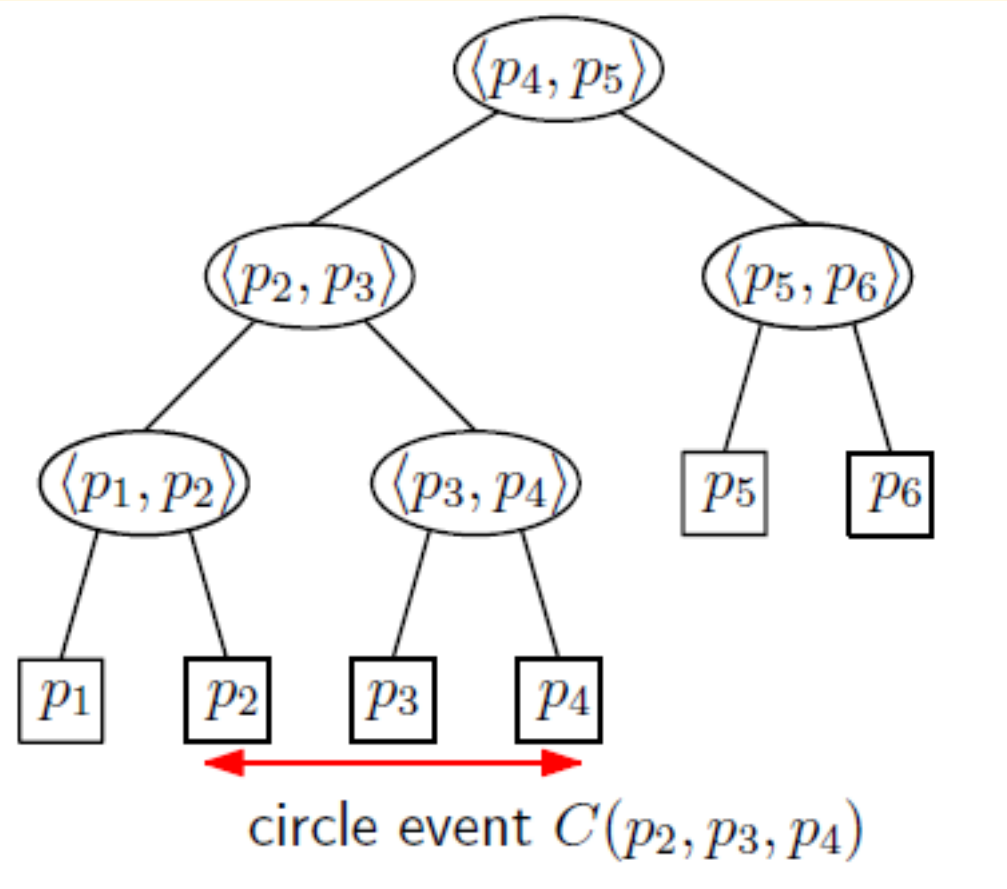
این False Alarm نامیده می‌شود.

دلایل

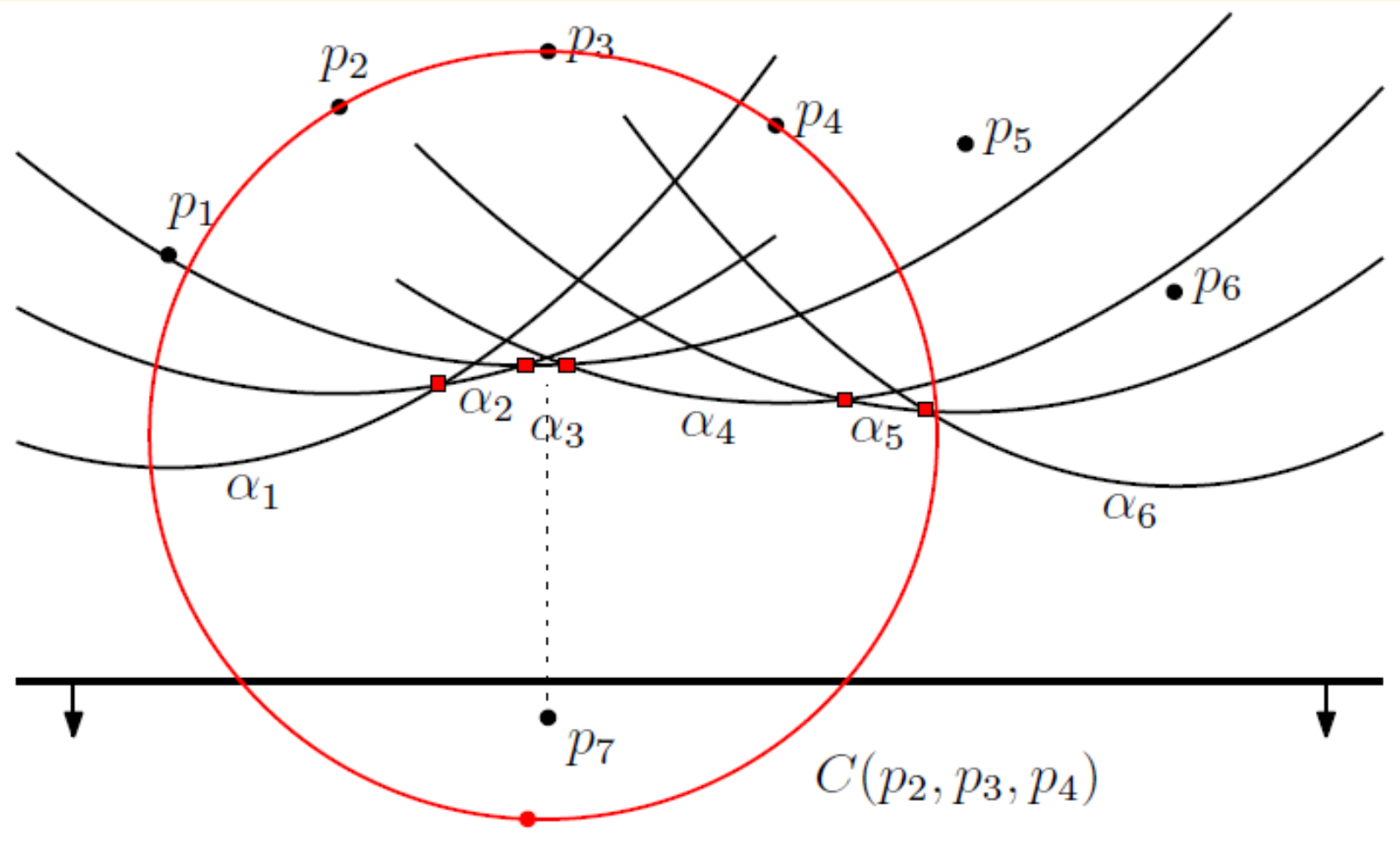
دو دلیل برای false alarm:

1. Site event
2. Circle eventهای دیگر

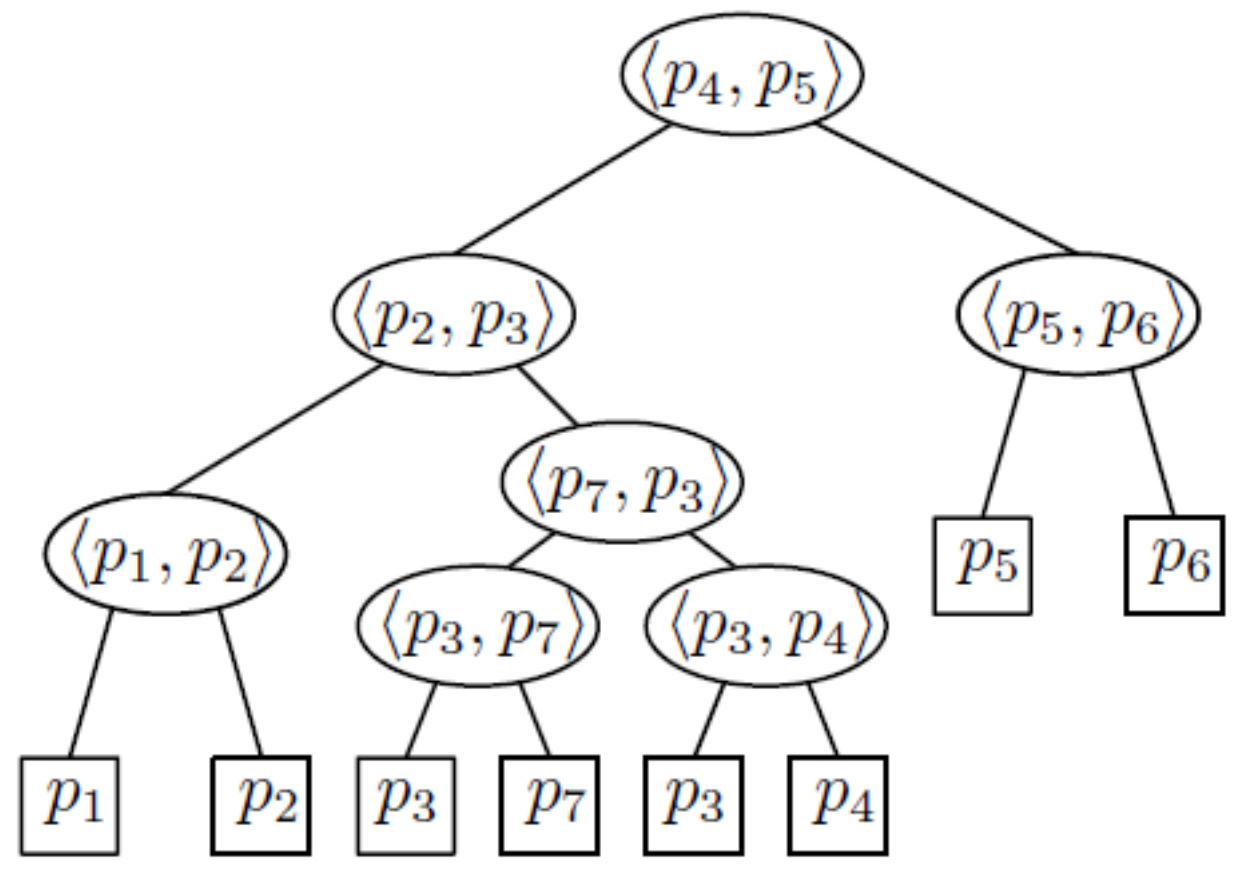
(1) False Alarm دلیل



(1) False Alarm دليل

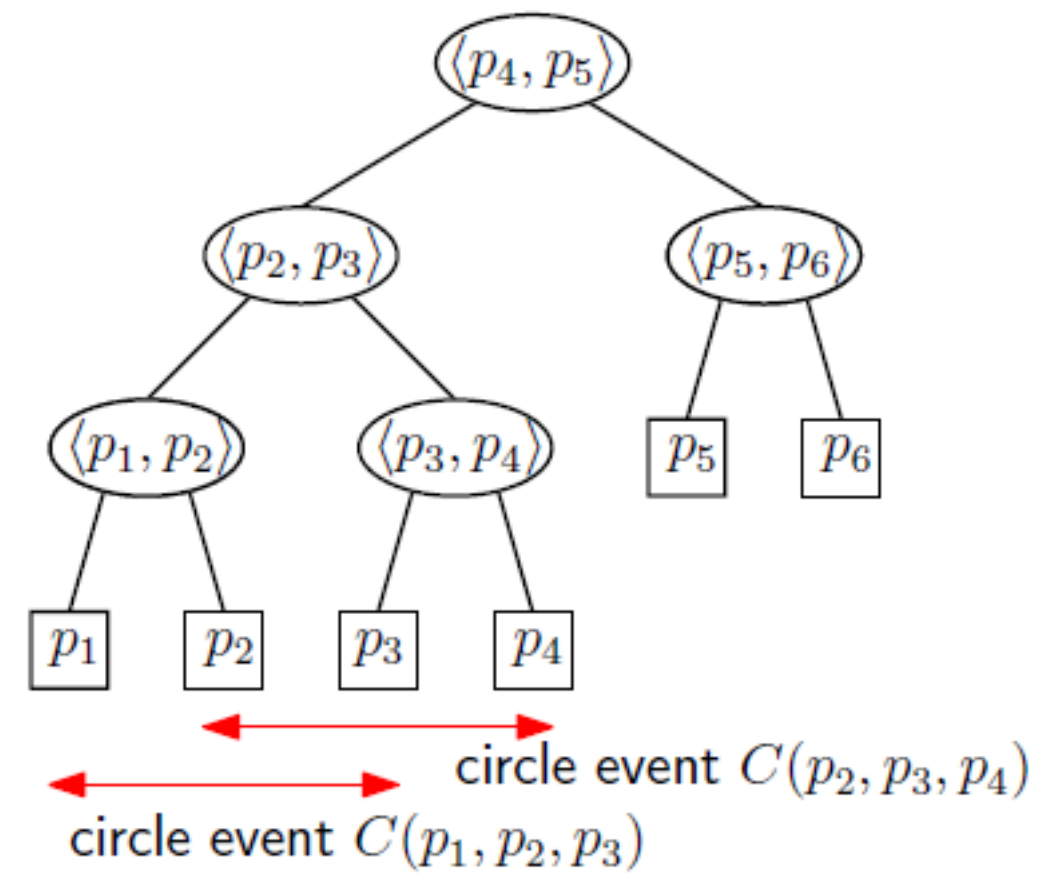


(1) False Alarm دلیل

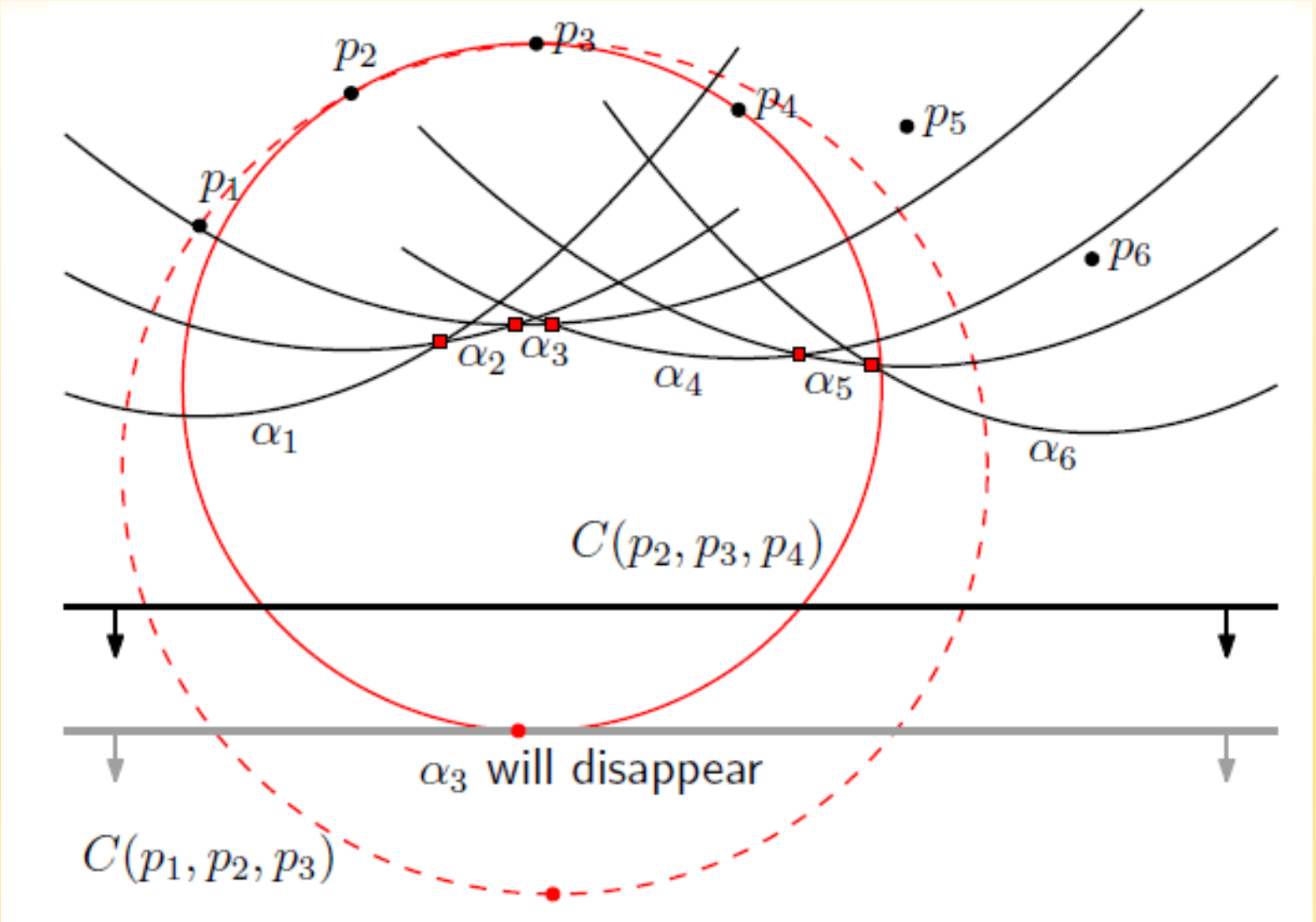


Status (بعد ازسایت p_7)

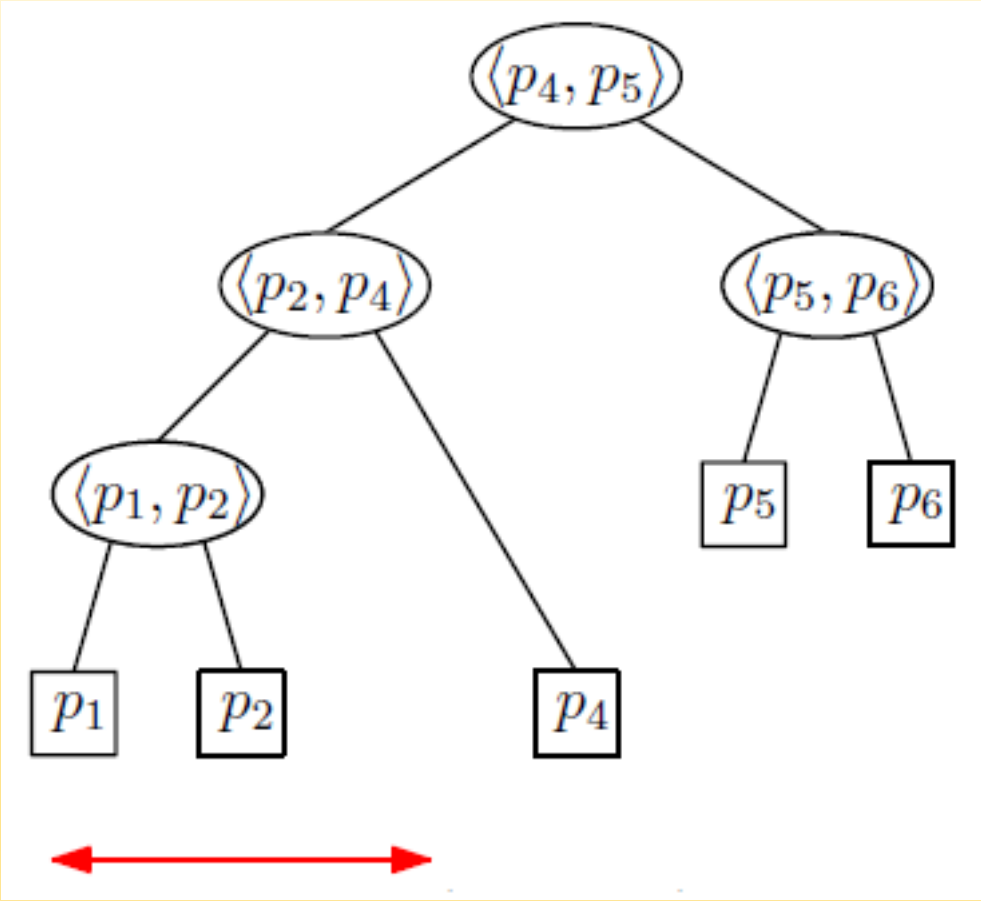
(2) False Alarm دلیل



(2) False Alarm دليل

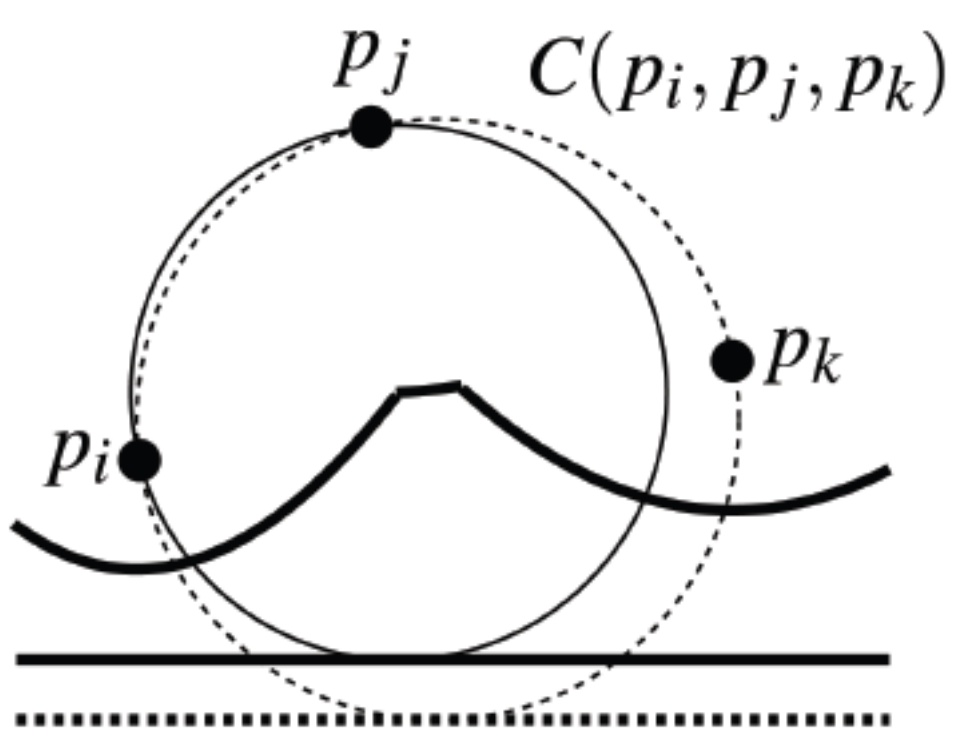


(2) False Alarm دلیل



Status (بعد از حذف p_3)

Circle Event لم



لم 7.8

هر رأس ورونوی توسط circle event تشخیص داده می‌شود.

اثبات

باید نشان دهیم که درست قبل از اینکه sweep line به پایین ترین نقطه $C(p_i,p_j,p_k)$ برسد، سه کمان متوالی α و α' و α'' روی خط ساحلی با سایت‌های p_i و p_j و p_k تعریف شده‌است، وجود دارد.

الگوریتم ورونوی دیاگرام

Algorithm VORONOIDIAGRAM(P)
Input. A set $P := \{p_1, \ldots, p_n\}$ of point sites in the plane.
Output. The Voronoi diagram Vor(P) given inside a bounding box in a doubly-connected edge list $\mathcal{D}$.

1. Initialize the event queue $\mathcal{Q}$ with all site events, initialize an empty status structure $\mathcal{T}$ and an empty doubly-connected edge list $\mathcal{D}$.
2. **while** $\mathcal{Q}$ is not empty
3. **do** Remove the event with largest y-coordinate from $\mathcal{Q}$.
4. **if** the event is a site event, occurring at site p_i
5. **then** HANDLESITEEVENT(p_i)
6. **else** HANDLECIRCLEEVENT(γ), where γ is the leaf of $\mathcal{T}$ representing the arc that will disappear
7. The internal nodes still present in $\mathcal{T}$ correspond to the half-infinite edges of the Voronoi diagram. Compute a bounding box that contains all vertices of the Voronoi diagram in its interior, and attach the half-infinite edges to the bounding box by updating the doubly-connected edge list appropriately.
8. Traverse the half-edges of the doubly-connected edge list to add the cell records and the pointers to and from them.

الگوریتم ورونوی دیاگرام

HANDLESITEEVENT(p_i)

1. If $\mathcal{T}$ is empty, insert p_i into it (so that $\mathcal{T}$ consists of a single leaf storing p_i) and return. Otherwise, continue with steps 2– 5.
2. Search in $\mathcal{T}$ for the arc α vertically above p_i. If the leaf representing α has a pointer to a circle event in $\mathcal{Q}$, then this circle event is a false alarm and it must be deleted from $\mathcal{Q}$.
3. Replace the leaf of $\mathcal{T}$ that represents α with a subtree having three leaves. The middle leaf stores the new site p_i and the other two leaves store the site p_j that was originally stored with α. Store the tuples $\langle p_j, p_i \rangle$ and $\langle p_i, p_j \rangle$ representing the new breakpoints at the two new internal nodes. Perform rebalancing operations on $\mathcal{T}$ if necessary.
4. Create new half-edge records in the Voronoi diagram structure for the edge separating $\mathcal{V}(p_i)$ and $\mathcal{V}(p_j)$, which will be traced out by the two new breakpoints.
5. Check the triple of consecutive arcs where the new arc for p_i is the left arc to see if the breakpoints converge. If so, insert the circle event into $\mathcal{Q}$ and add pointers between the node in $\mathcal{T}$ and the node in $\mathcal{Q}$. Do the same for the triple where the new arc is the right arc.

الگوریتم ورونوی دیاگرام

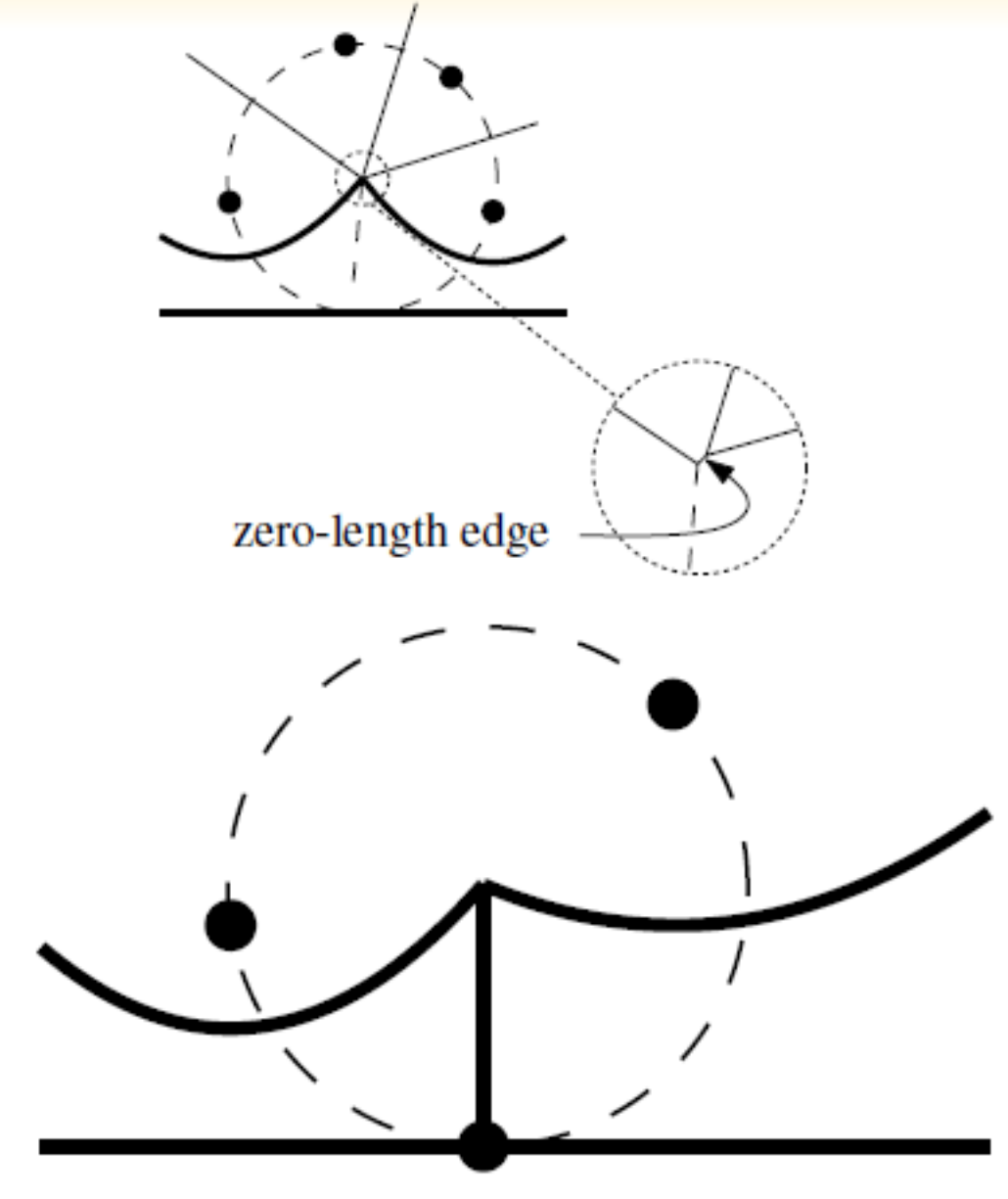
HANDLECIRCLEEVENT(γ)

1. Delete the leaf γ that represents the disappearing arc α from $\mathcal{T}$. Update the tuples representing the breakpoints at the internal nodes. Perform rebalancing operations on $\mathcal{T}$ if necessary. Delete all circle events involving α from $\mathcal{Q}$; these can be found using the pointers from the predecessor and the successor of γ in $\mathcal{T}$. (The circle event where α is the middle arc is currently being handled, and has already been deleted from $\mathcal{Q}$.)
2. Add the center of the circle causing the event as a vertex record to the doubly-connected edge list $\mathcal{D}$ storing the Voronoi diagram under construction. Create two half-edge records corresponding to the new breakpoint of the beach line. Set the pointers between them appropriately. Attach the three new records to the half-edge records that end at the vertex.
3. Check the new triple of consecutive arcs that has the former left neighbor of α as its middle arc to see if the two breakpoints of the triple converge. If so, insert the corresponding circle event into $\mathcal{Q}$. and set pointers between the new circle event in $\mathcal{Q}$ and the corresponding leaf of $\mathcal{T}$. Do the same for the triple where the former right neighbor is the middle arc.

لم الگوریتم

لم 7.9

الگوریتم در زمان O(nlogn) اجرا شده و فضای O(n) را برای ذخیره سازی می‌گیرد.



Degenerate Case

حالت 1 (eventهایی که در یک خط مستقیم قرار دارند)

دویا بیشتر event روی خط افقی مشترک قرار گیرند.

چندین circle event منطبق بر هم وجود داشته باشد، وقتی که چهار یا بیشتر نقطه روی یک دایره مشترک قرار دارند.

حالت 2

وقتی که سایت p_i دقیقا زیر breakpoint بین دو کمان روی خط ساحلی رخ می‌دهد.

نتیجه

قضیه 7.10

ورونوی دیاگرام مجموعه n نقطه سایت در صفحه می‌تواند با الگویتم sweep line در زمان O(nlogn) با فضای O(n) ساخته شود.