Geometria Computacional

Cristina G. Fernandes

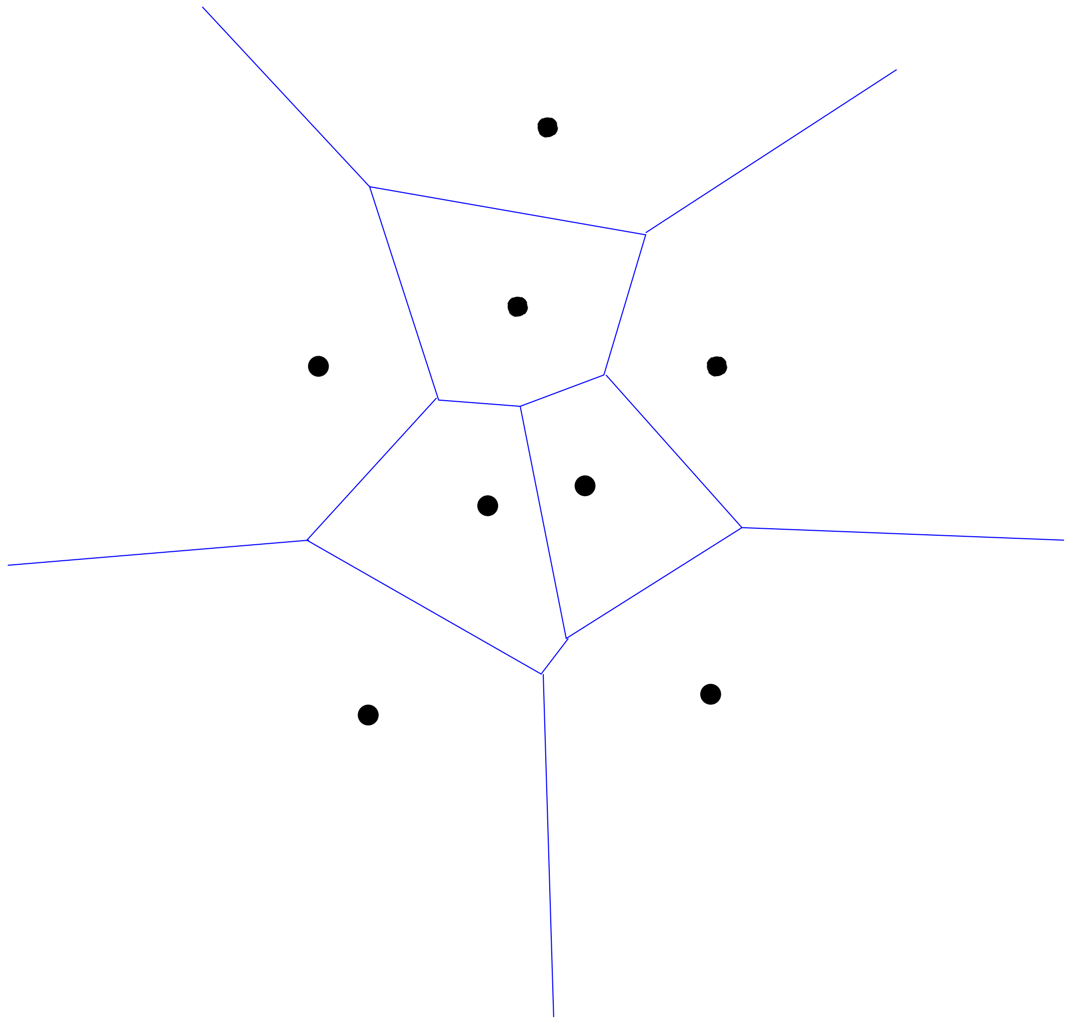
Departamento de Ciência da Computação do IME-USP

`http://www.ime.usp.br/~cris/`

segundo semestre de 2011

Diagrama de Voronoi

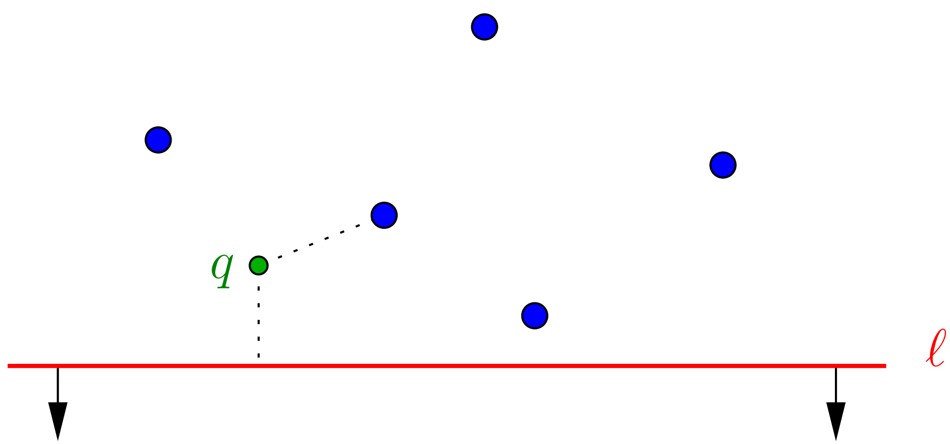
Dados endereços de agências de correio, determinar qual é a região da cidade que fica mais próxima de cada agência.



Algoritmo de Fortune

ℓ^+: semiplano acima da linha de varredura ℓ

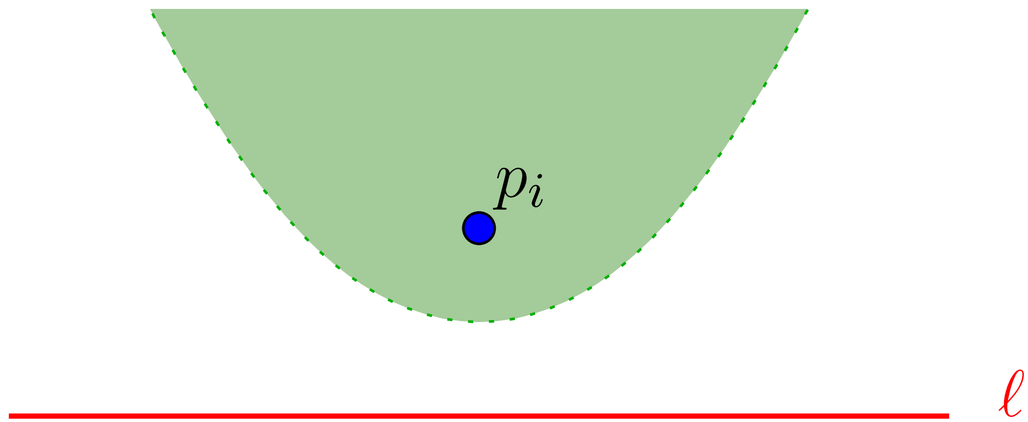
Para quais pontos q em ℓ^+
já conhecemos o ponto de P mais próximo a q?



Se q está mais próximo de um p_i acima de ℓ do que de ℓ, então q está na célula de p_i.

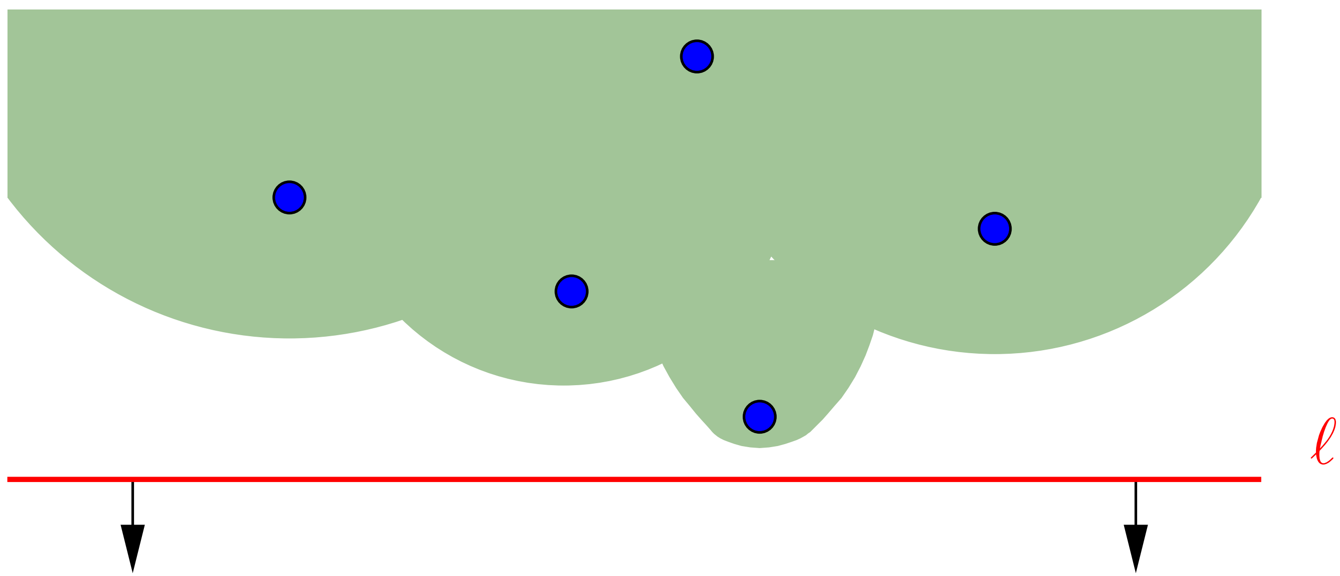
Linha da praia

O conjunto dos pontos mais próximos a p_i do que ℓ é delimitado por uma parábola.



Linha da praia

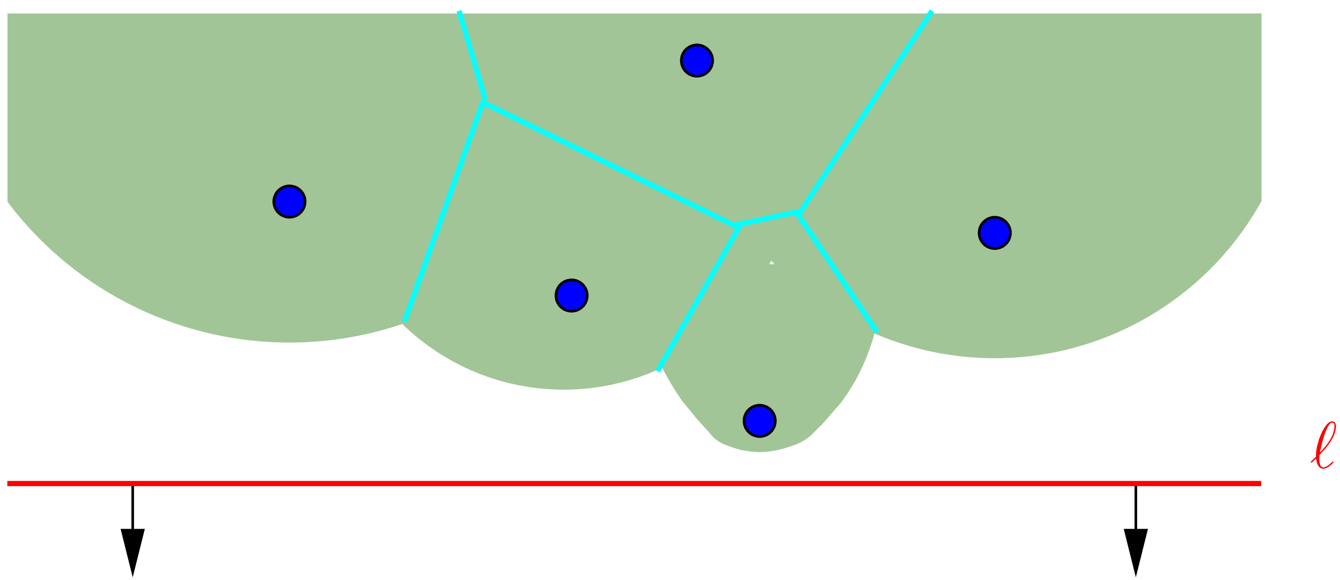
O conjunto dos pontos mais próximos a p_i do que ℓ é delimitado por uma parábola.



Assim, a região de ℓ^+ onde $\mathrm{Vor}(P)$ é conhecido é delimitada por arcos parabolóides, que definem a chamada linha da praia.

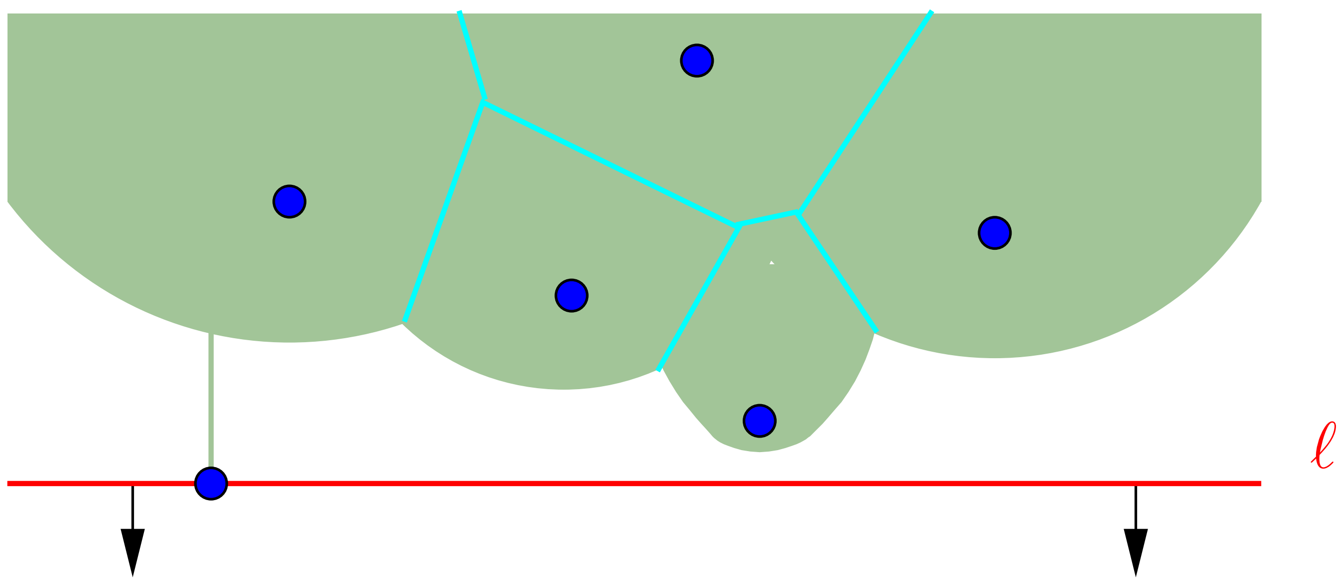
Evento-ponto

Pontos de encontro entre duas parábolas na linha da praia desenham as arestas de $\mathrm{Vor}(P)$.



Evento-ponto

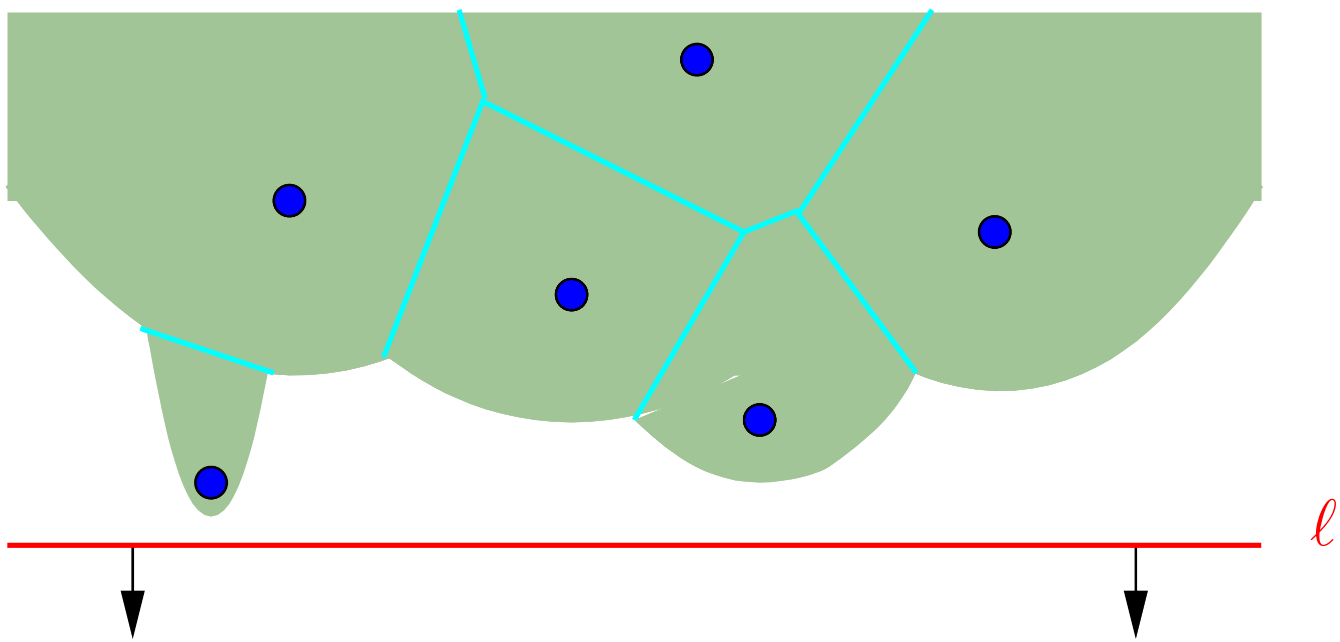
Pontos de encontro entre duas parábolas na linha da praia desenham as arestas de $\mathrm{Vor}(P)$.



Arcos que entram na linha de praia
são arestas de $\mathrm{Vor}(P)$ que começam a ser desenhadas.

Evento-ponto

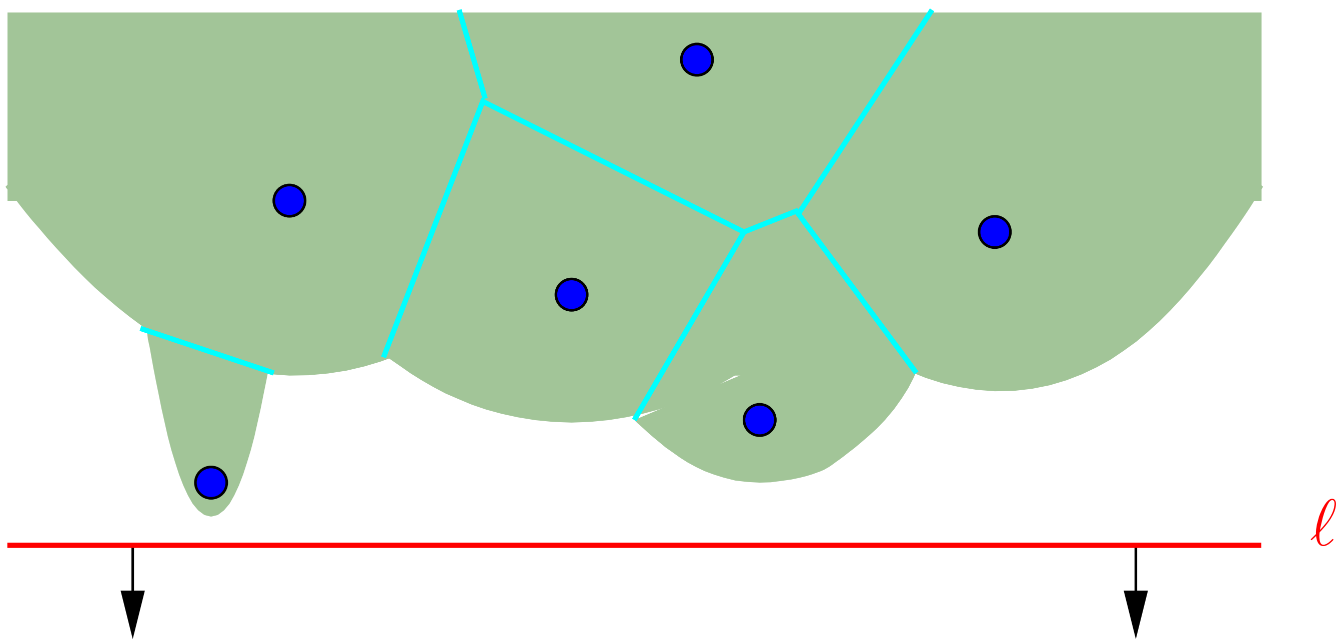
Pontos de encontro entre duas parábolas na linha da praia desenham as arestas de $\mathrm{Vor}(P)$.



Arcos que entram na linha de praia
são arestas de $\mathrm{Vor}(P)$ que começam a ser desenhadas.

Evento-ponto

Pontos de encontro entre duas parábolas na linha da praia desenham as arestas de $\mathrm{Vor}(P)$.

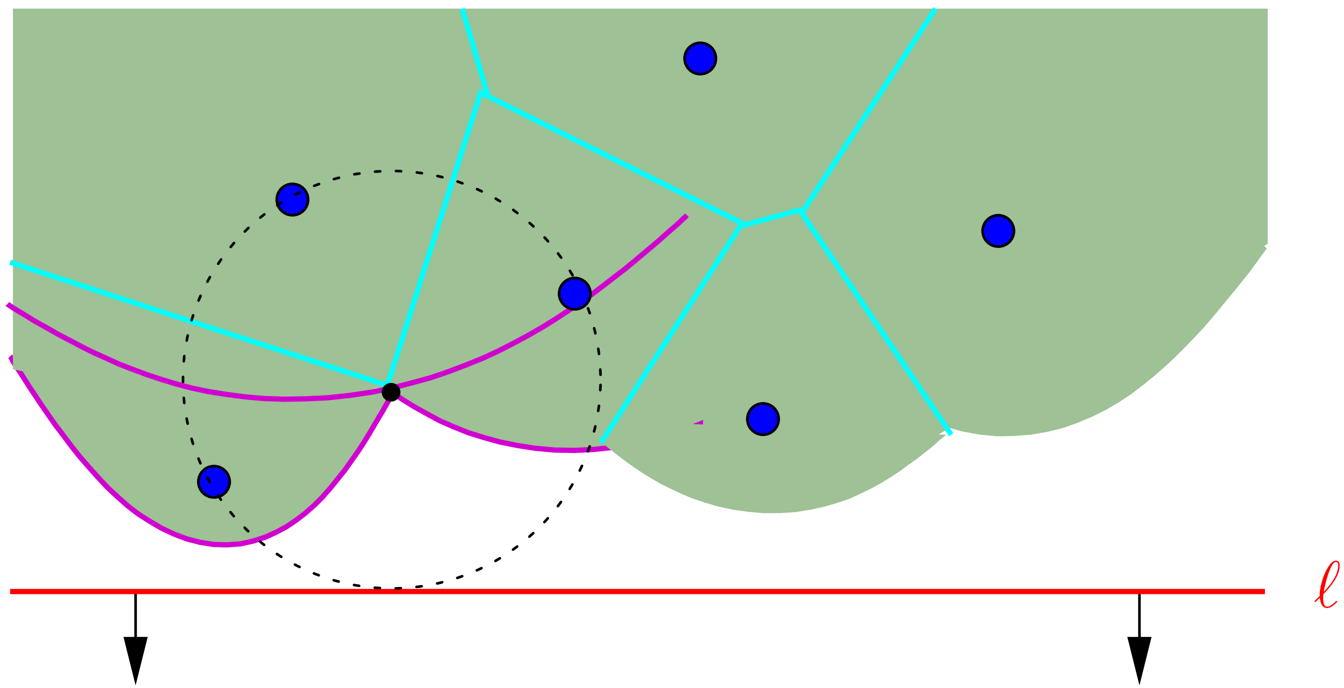


Arcos que entram na linha de praia
são arestas de $\mathrm{Vor}(P)$ que começam a ser desenhadas.

Ponto evento relacionado: um ponto de P (evento-ponto).

Evento-círculo

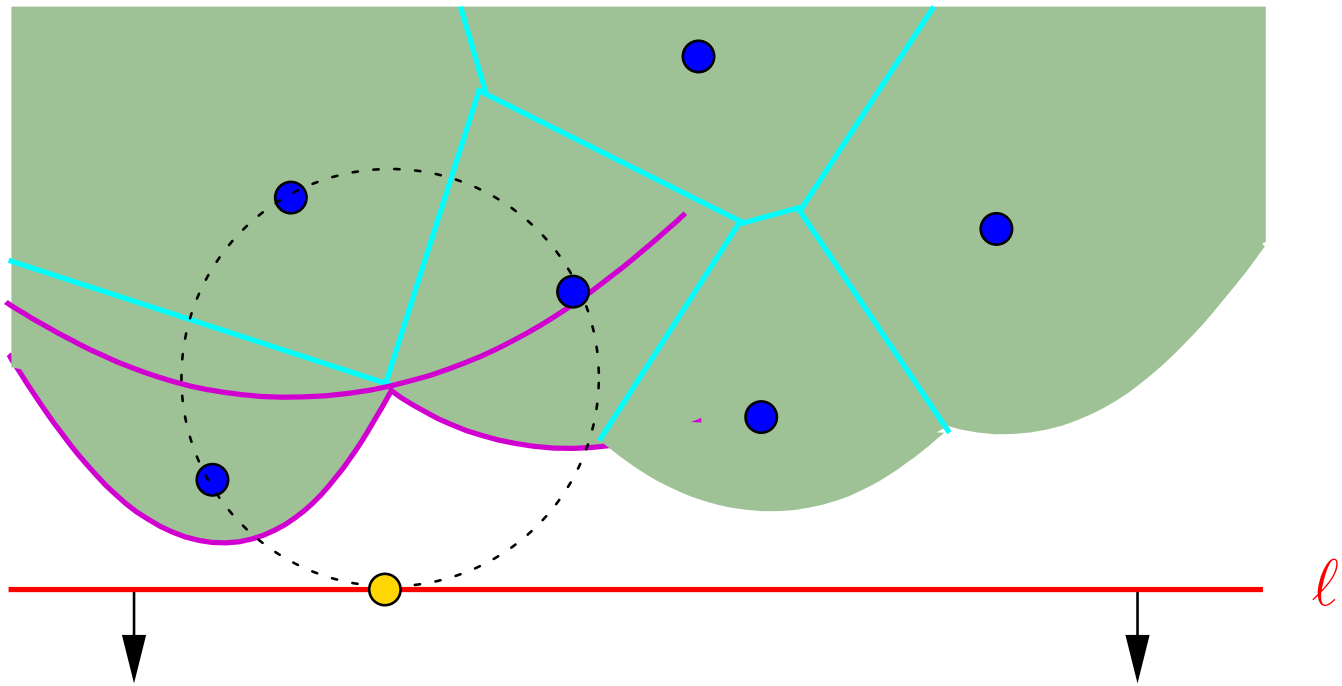
Quando dois pontos de quebra entre arcos se encontram, o arco entre eles sai da linha de praia.



Arcos que saem da linha de praia correspondem a vértices de $\mathrm{Vor}(P)$.

Evento-círculo

Quando dois pontos de quebra entre arcos se encontram, o arco entre eles sai da linha de praia.



Arcos que saem da linha de praia correspondem a vértices de $\mathrm{Vor}(P)$.

Ponto evento relacionado: o mais baixo do círculo com os pontos de P associados aos três arcos (evento-círculo).

Estruturas de dados

Para $\mathrm{Vor}(P)$: listas de arestas duplamente ligadas
(como na partição de polígono em partes monótonas).

Estruturas de dados

Para $\mathrm{Vor}(P)$: listas de arestas duplamente ligadas (como na partição de polígono em partes monótonas).

Para a fila de eventos: uma fila de prioridade, que começa com os pontos de P, ordenados por Y-coordenada.

Durante o algoritmo, inserção e remoção de candidatos a eventos-círculo.

Estruturas de dados

Para $\mathrm{Vor}(P)$: listas de arestas duplamente ligadas (como na partição de polígono em partes monótonas).

Para a fila de eventos: uma fila de prioridade, que começa com os pontos de P, ordenados por Y-coordenada.

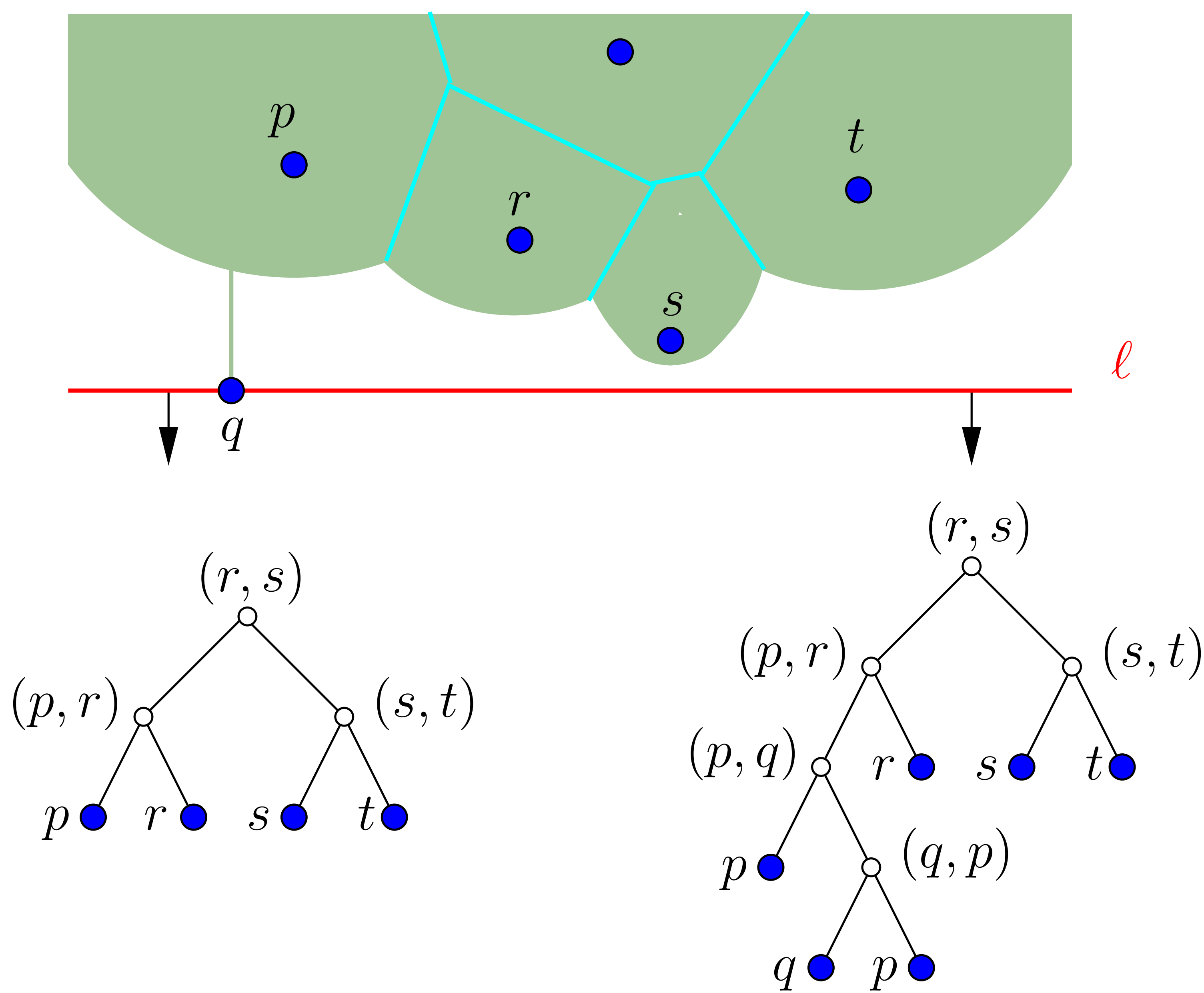
Durante o algoritmo, inserção e remoção de candidatos a eventos-círculo.

Para a linha da praia, usamos uma ABBB,
com arcos nas folhas e pontos de quebra nos nós internos.

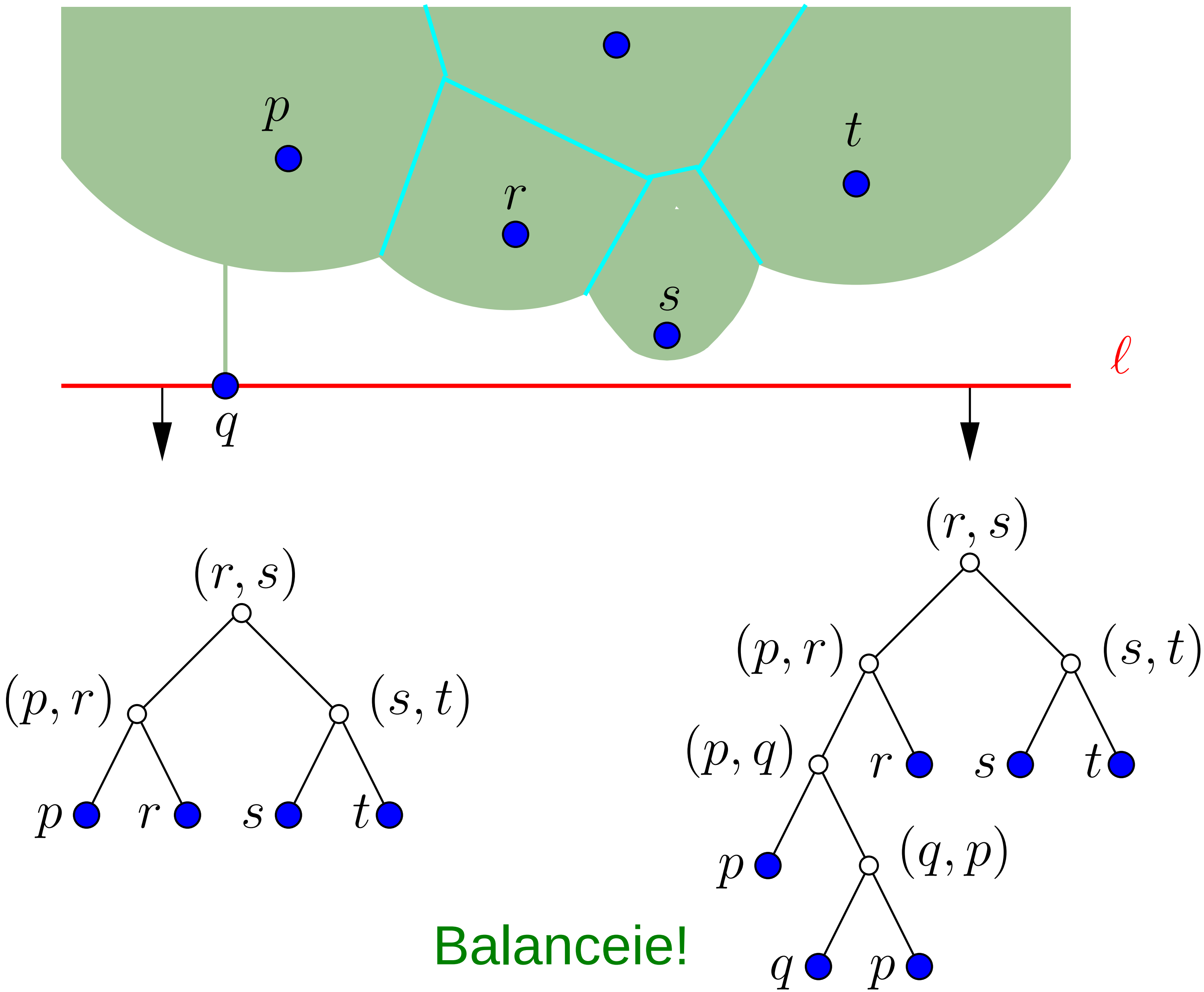
Um arco é representado pelo ponto p_i que o determina.

Um ponto de quebra é representado por um par de pontos (p_i, p_j) cujos arcos o determinam, e está associado a uma aresta de $\mathrm{Vor}(P)$.

ABBB da linha da praia



ABBB da linha da praia



Balanceie!

Algoritmo de Fortune

Fortune(P, n)

$Q \leftarrow$ FILADEEVENTOS(P, n) ▷ P ord. por Y-coordenada
CRIE(T) ▷ ED para a linha da praia
CRIE$(\mathcal{V})$ ▷ ED para $\mathrm{Vor}(P)$
enquanto não VAZIA(Q) **faça**
  $q \leftarrow$ REMOVAMIN(Q)
  se q é um evento-ponto
    então TRATAEVENTOPONTO$(q, T, Q, \mathcal{V})$
    senão TRATAEVENTOCÍRCULO$(q, T, Q, \mathcal{V})$
FINALIZEVORONOI$(\mathcal{V}, T)$ ▷ adiciona o vértice ∞
devolva $\mathcal{V}$

Algoritmo de Fortune

Fortune(P, n)

$Q \leftarrow$ FilaDeEventos(P, n) $\triangleright$ P ord. por Y-coordenada
Crie(T) $\triangleright$ ED para a linha da praia
Crie$(\mathcal{V})$ $\triangleright$ ED para $\mathrm{Vor}(P)$
enquanto não Vazia(Q) **faça**
  $q \leftarrow$ RemovaMin(Q)
  se q é um evento-ponto
    então TrataEventoPonto$(q, T, Q, \mathcal{V})$
    senão TrataEventoCírculo$(q, T, Q, \mathcal{V})$
FinalizeVoronoi$(\mathcal{V}, T)$ $\triangleright$ adiciona o vértice ∞
devolva $\mathcal{V}$

Há no máximo $2n - 1 = \mathrm{O}(n)$ arcos em T,
logo $\mathrm{O}(n)$ pontos evento em Q.

Algoritmo de Fortune

Fortune(P, n)

$Q \leftarrow$ FilaDeEventos(P, n) $\triangleright$ P ord. por Y-coordenada
Crie(T) $\triangleright$ ED para a linha da praia
Crie$(\mathcal{V})$ $\triangleright$ ED para $\mathrm{Vor}(P)$
enquanto não Vazia(Q) **faça**
    $q \leftarrow$ RemovaMin(Q)
    se q é um evento-ponto
        então TrataEventoPonto$(q, T, Q, \mathcal{V})$
        senão TrataEventoCírculo$(q, T, Q, \mathcal{V})$
FinalizeVoronoi$(\mathcal{V}, T)$ $\triangleright$ adiciona o vértice ∞
devolva $\mathcal{V}$

FinalizeVoronoi$(\mathcal{V}, T)$: adiciona o vértice ∞ como extremo das arestas dos nós internos que restam em T.

Tratamento de evento-ponto

TrataEventoPonto$(q, T, Q, \mathcal{V})$

se $T = \emptyset$
  então Insira(T, q)
  senão $f \leftarrow$ Busque(T, q) $\triangleright$ folha de T do arco acima de q
    $i \leftarrow evento\_circ(f)$
    se $i \neq -1$
      então Remova(Q, i)
    $(u, f, v) \leftarrow$ Quebre_e_Insira(T, f, q)
    NovaAresta$(\mathcal{V}, u, \text{NIL}, v, \text{NIL})$
    AtualizaEventos(Q, T, f)

Tratamento de evento-ponto

TRATAEVENTOPONTO$(q, T, Q, \mathcal{V})$

se $T = \emptyset$
    então INSIRA(T, q)
    senão $f \leftarrow$ BUSQUE(T, q) ▷ folha de T do arco acima de q
        $i \leftarrow \mathit{evento\_circ}(f)$
        se $i \neq -1$
            então REMOVA(Q, i)
        $(u, f, v) \leftarrow$ QUEBRE_E_INSIRA(T, f, q)
        NOVAARESTA$(\mathcal{V}, u, \text{NIL}, v, \text{NIL})$
        ATUALIZAEVENTOS(Q, T, f)

$\mathit{evento\_circ}(f)$: índice de Q para o evento-círculo (se existir) associado ao arco em f.

Tratamento de evento-ponto

TRATAEVENTOPONTO$(q, T, Q, \mathcal{V})$

se $T = \emptyset$
  então INSIRA(T, q)
  senão $f \leftarrow$ BUSQUE(T, q) $\triangleright$ folha de T do arco acima de q
    $i \leftarrow evento\_circ(f)$
    se $i \neq -1$
      então REMOVA(Q, i)
    $(u, f, v) \leftarrow$ QUEBRE_E_INSIRA(T, f, q)
    NOVAARESTA$(\mathcal{V}, u, \textsc{nil}, v, \textsc{nil})$
    ATUALIZAEVENTOS(Q, T, f)

QUEBRE_E_INSIRA(T, f, q): substitua f por árvore com três folhas, a do meio para o arco de q e as outras duas para o arco de $p = ponto(f)$. Balanceie T se necessário. Devolva apontadores para os nós internos novos e folha de q.

Tratamento de evento-ponto

TRATAEVENTOPONTO$(q, T, Q, \mathcal{V})$

se $T = \emptyset$
  então INSIRA(T, q)
  senão $f \leftarrow$ BUSQUE(T, q) ▷ folha de T do arco acima de q
    $i \leftarrow evento\_circ(f)$
    se $i \neq -1$
      então REMOVA(Q, i)
    $(u, f, v) \leftarrow$ QUEBRE_E_INSIRA(T, f, q)
    NOVAARESTA$(\mathcal{V}, u, \text{NIL}, v, \text{NIL})$
    ATUALIZAEVENTOS(Q, T, f)

NOVAARESTA$(\mathcal{V}, u, x, v, y)$: cria aresta nova em $\mathrm{Vor}(P)$, com uma gêmea do nó interno u de T, indo para o vértice x de $\mathrm{Vor}(P)$, e outra, de v, indo para y. (Se x ou y são NIL, tal vértice ainda está indefinido.)

Tratamento de evento-ponto

TRATAEVENTOPONTO$(q, T, Q, \mathcal{V})$
se $T = \emptyset$
  então INSIRA(T, q)
  senão $f \leftarrow$ BUSQUE(T, q) $\triangleright$ folha de T do arco acima de q
    $i \leftarrow evento\_circ(f)$
    se $i \neq -1$
      então REMOVA(Q, i)
    $(u, f, v) \leftarrow$ QUEBRE_E_INSIRA(T, f, q)
    NOVAARESTA$(\mathcal{V}, u, \text{NIL}, v, \text{NIL})$
    ATUALIZAEVENTOS(Q, T, f)

ATUALIZAEVENTOS(Q, T, f): calcule o evento-círculo das duas novas triplas de arcos consecutivos em T; se a X-coordenada tal ponto é menor que q_X, então acrescente-o a Q.

Tratamento de evento-círculo

TrataEventoCírculo$(q, T, Q, \mathcal{V})$

$f \leftarrow folha(q)$ $\triangleright$ folha de T do arco associado a q

$(pred, suc, novo) \leftarrow$ Remova(T, f)

AtualizaEventos$(Q, T, novo)$

$c \leftarrow centro(q)$ $\triangleright$ centro do círculo associado a q

$u \leftarrow$ NovoVértice$(\mathcal{V}, c)$

AdicionaExtremo$(\mathcal{V}, u, aresta(pred), aresta(suc))$

NovaAresta$(\mathcal{V}, novo, \text{NIL}, \text{NIL}, u)$

Remova(T, f): remova f e devolva os dois nós internos de T associados ao arco de f, e o seu substituto.

Tratamento de evento-círculo

TRATAEVENTOCÍRCULO$(q, T, Q, \mathcal{V})$

1 $f \leftarrow folha(q)$ ▷ folha de T do arco associado a q
2 $(pred, suc, novo) \leftarrow$ REMOVA(T, f)
3 ATUALIZAEVENTOS$(Q, T, novo)$
4 $c \leftarrow centro(q)$ ▷ centro do círculo associado a q
5 $u \leftarrow$ NOVOVÉRTICE$(\mathcal{V}, c)$
6 ADICIONAEXTREMO$(\mathcal{V}, u, aresta(pred), aresta(suc))$
7 NOVAARESTA$(\mathcal{V}, novo, \text{NIL}, \text{NIL}, u)$

REMOVA(T, f): remova f e devolva os dois nós internos de T associados ao arco de f, e o seu substituto.

ADICIONAEXTREMO$(\mathcal{V}, u, aresta(pred), aresta(suc))$: põe u como extremo das gêmeas correspondentes aos pontos de quebra associados a q.

Casos degenerados

Pontos evento com mesma Y-coordenada:

- se X-coordenadas são distintas,
  podem ser tratados numa ordem arbitrária,

Casos degenerados

Pontos evento com mesma Y-coordenada:

- se X-coordenadas são distintas,
  podem ser tratados numa ordem arbitrária,
  exceto quando estes são os primeiros,
  quando é necessário um tratamento diferente.

Casos degenerados

Pontos evento com mesma Y-coordenada:

- se X-coordenadas são distintas,
  podem ser tratados numa ordem arbitrária,
  exceto quando estes são os primeiros,
  quando é necessário um tratamento diferente.
- se X-coordenadas coincidem,
  podem ser tratados numa ordem arbitrária,
  exigindo eventualmente uma limpeza final.

Casos degenerados

Pontos evento com mesma Y-coordenada:

- se X-coordenadas são distintas,
  podem ser tratados numa ordem arbitrária,
  exceto quando estes são os primeiros,
  quando é necessário um tratamento diferente.
- se X-coordenadas coincidem,
  podem ser tratados numa ordem arbitrária,
  exigindo eventualmente uma limpeza final.

Ponto de P exatamente abaixo de um ponto de quebra:

- qual arco está acima dele?
  pode-se escolher qualquer um dos dois candidatos.

Casos degenerados

Pontos evento com mesma Y-coordenada:

- se X-coordenadas são distintas,
  podem ser tratados numa ordem arbitrária,
  exceto quando estes são os primeiros,
  quando é necessário um tratamento diferente.
- se X-coordenadas coincidem,
  podem ser tratados numa ordem arbitrária,
  exigindo eventualmente uma limpeza final.

Ponto de P exatamente abaixo de um ponto de quebra:

- qual arco está acima dele?
  pode-se escolher qualquer um dos dois candidatos.

Tripla de arcos de pontos colineares:

- definem um círculo?
  não, logo não geram um evento-círculo.

Consumo de tempo

TRATAEVENTOPONTO$(q, T, Q, \mathcal{V})$
se $T = \emptyset$
  então INSIRA(T, q)
  senão $f \leftarrow$ BUSQUE(T, q) ▷ folha de T do arco acima de q
    $i \leftarrow evento\_circ(f)$
    se $i \neq -1$
      então REMOVA(Q, i)
    $(u, f, v) \leftarrow$ QUEBRE_E_INSIRA(T, f, q)
    NOVAARESTA$(\mathcal{V}, u, \text{NIL}, v, \text{NIL})$
    ATUALIZAEVENTOS(Q, T, f)

Consumo de tempo: $\mathrm{O}(\lg n)$.

Consumo de tempo

TRATAEVENTOPONTO$(q, T, Q, \mathcal{V})$

Consumo de tempo: $\mathrm{O}(\lg n)$.

TRATAEVENTOCÍRCULO$(q, T, Q, \mathcal{V})$

$f \leftarrow folha(q)$ $\quad\triangleright$ folha de T do arco associado a q
$(pred, suc, novo) \leftarrow$ REMOVA(T, f)
ATUALIZAEVENTOS(Q, T, f)
$c \leftarrow centro(q)$ $\quad\triangleright$ centro do círculo associado a q
$u \leftarrow$ NOVOVÉRTICE$(\mathcal{V}, c)$
ADICIONAEXTREMO$(\mathcal{V}, u, aresta(pred), aresta(suc))$
NOVAARESTA$(\mathcal{V}, novo, \text{NIL}, \text{NIL}, u)$

Consumo de tempo: $\mathrm{O}(\lg n)$.

Consumo de tempo

TRATAEVENTOPONTO$(q, T, Q, \mathcal{V})$
TRATAEVENTOCÍRCULO$(q, T, Q, \mathcal{V})$

Consumo de tempo: $\mathrm{O}(\lg n)$.

Fortune(P, n)

$Q \leftarrow$ FILADEEVENTOS(P, n) ▷ P ord. por Y-coordenada
CRIE(T) ▷ ED para a linha da praia
CRIE$(\mathcal{V})$ ▷ ED para $\mathrm{Vor}(P)$
enquanto não VAZIA(Q) **faça**
    $q \leftarrow$ REMOVAMIN(Q)
    se q é um evento-ponto
        então TRATAEVENTOPONTO$(q, T, Q, \mathcal{V})$
        senão TRATAEVENTOCÍRCULO$(q, T, Q, \mathcal{V})$
FINALIZEVORONOI$(\mathcal{V}, T)$ ▷ adiciona o vértice ∞
devolva $\mathcal{V}$

Consumo de tempo: $\mathrm{O}(n \lg n)$.