計算幾何学セミナー*
第 6 章 点位置決定問題

伊藤剛志† (コンピュータ科学専攻 今井研究室 博士 1 年)

2004 年 6 月 8 日

本 [1] の第 6 章では平面上の点位置決定問題に対する確率的逐次添加アルゴリズムを学ぶ。

6.1 点位置決定問題と台形地図

平面を折れ線によって多角形領域に区切った図 (平面領域分割、詳しくは第 2 章) が与えられる。点が指定されたとき、その点がどの領域に入っているか調べたい。

例えば、地図ソフトウェアで、地図上の点をマウスでクリックしたら国名が表示される、など。

平面上の n 角形 P と点 q が与えられたとき、 q が P の内部・外部・辺上のいずれにあるかを判定するのは $O(n)$ 時間で可能 (問題 6.3)。これを使えば点位置決定問題も $O(n)$ 時間で可能 (問題 6.4)。どんなアルゴリズムも線形時間はかかるので、これが最良……?

これだと n が大きい ($n \sim 10^7$ とか) とき遅い。もっと速く動作させたい。そこで、平面領域分割が与えられたら、与えられた平面領域分割に対して**前処理**をして、後で点 q が与えられたときに短時間で q が属する領域を答えられるような方法を考えることにする (このような方法に対して、前のような一度にすべてが与えられる設定を single shot problem という)。

アルゴリズム 1: スラブ分割

すべての頂点で y 軸に平行な直線を引いて分割する。これを S のスラブ (slab) 分割という。辺数が n のとき、 $\Theta(n^2)$ 個に分割されることがあるため、 $\Theta(n^2)$ 領域が必要になる (図 1)。点が指定されたときの問い合わせにかかる時間は $O(\log n)$ で効率が良い。しかし、 n が大きいと、 $\Theta(n^2)$ 領域では不満。

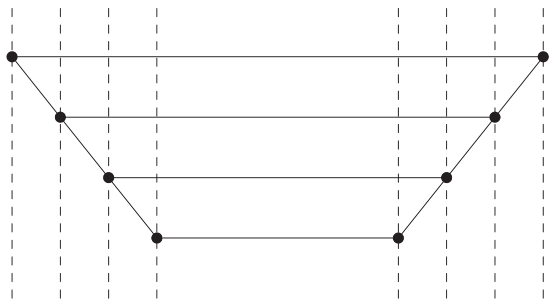

図 1: スラブ分割すると $\Theta(n^2)$ 領域必要になる例

*http://www-ise.is.s.u-tokyo.ac.jp/~kenya/seminar/computational_geometry.html

†tsuyoshi@is.s.u-tokyo.ac.jp

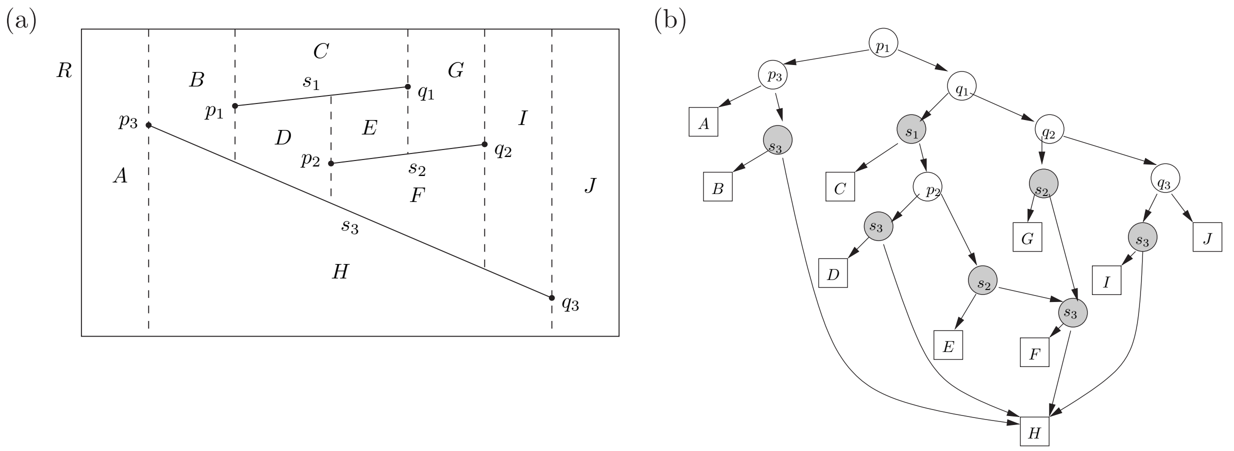

図 2: (a) 一般の位置にある 3 本の線分の台形地図。 (b) それに対する検索構造

アルゴリズム 2: 台形地図+検索構造

しばらく次の仮定をおく。仮定:

1. 平面領域分割において、二つの異なる頂点が同じ x 座標を持つことはない (まったく同じ点が 2 度表れるのは OK)。
2. 問い合わせの点が頂点に一致しないならば、頂点と同じ x 座標を持つことはない。

この仮定は 6.3 節で取り除く。

2 本の線分が**交差しない** (non-crossing) とは、2 本の線分が共有点を持たないか、共有点が両方の端点になっていることをいう。十字や T 字になっていたり、線分がちょうど重なっていて重なり部分が正の長さを持っていたりするのは駄目。

どの 2 本も交差しない n 本の線分が**一般の位置にある** (in general position) とは、異なる端点が同じ x 座標を持つことがないことをいう。上の仮定 1 を満たす平面領域分割を考えて、その全部の辺からなる集合を考えると、一般の位置にある線分の集合になる。

S を一般の位置にある n 本の線分の集合とする。 S を囲むような長方形 R を考え、以下ではすべての問い合わせは R の内部の点に対してのみ行われるとする。

S の各端点から上下に鉛直線をのばし、線分か R の上下辺に当たったら止める。こうしてできる R の台形 (三角形を含む) への分割を S の**台形地図**(trapezoidal map; vertical decomposition とか trapezoidal decomposition ともいう) といい、 $\mathcal{T}(S)$ と書く (図 2 (a))。

次のような節点からなる DAG (directed acyclic graph; 閉路を持たない有向グラフ) を**検索構造** (search structure) という (図 2 (b))。

- x 節点: S の端点のうち一つが格納されている。左の子と右の子を持つ。
- y 節点: S の線分のうち一つが格納されている。上の子と下の子を持つ (図では上の子を左に、下の子を右にかく)。
- 葉節点: $\mathcal{T}(S)$ の台形のうち一つが格納されている。子を持たない。

検索構造を使うと、点 q が与えられたとき、 q の属する台形 $\Delta_q(S)$ がわかる。

台形地図 $\mathcal{T}(S)$ において、各台形 $\Delta \in \mathcal{T}(S)$ の上辺と下辺は S の線分か R の上下辺のいずれかになる。これをそれぞれ $top(\Delta), bottom(\Delta)$ と表す。

各台形 Δ の左端の状況は、次の 5 通りのどれかになる (図 3)。それぞれの場合に $leftp(\Delta)$ という点を定義する。

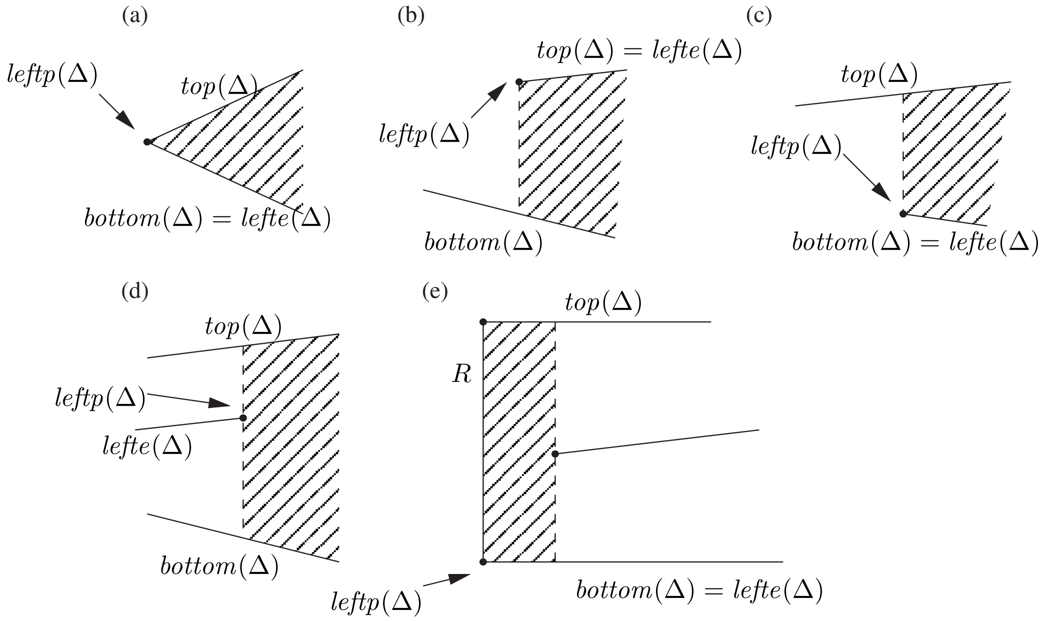

図 3: 台形分割における台形 Δ の左端の状況

(a) Δ の左辺はただ 1 点からなる。この場合、この点を $leftp(\Delta)$ とする。

(b) Δ の左辺は $top(\Delta)$ の左端から下にのびた鉛直線である。この場合、 $top(\Delta)$ の左端を $leftp(\Delta)$ とする。

(c) Δ の左辺は $bottom(\Delta)$ の左端から上にのびた鉛直線である。この場合、 $bottom(\Delta)$ の左端を $leftp(\Delta)$ とする。

(d) Δ の左辺はある線分の右端から上下にのびた鉛直線である。この場合、この線分の右端を $leftp(\Delta)$ とする。

(e) Δ の左辺は R の左辺である。この場合、 R の左下頂点を $leftp(\Delta)$ とする。

同様に、 Δ の右辺も 5 通り考えられ、それぞれの場合に点 $rightp(\Delta)$ を定義する。

補題 6.2. 一般の位置にある n 本の線分 S の台形地図 $\mathcal{T}(S)$ の頂点数は $6n+4$ 以下、台形の個数は $3n+1$ 以下。

証明 $\mathcal{T}(S)$ の頂点は次のいずれかである。

- R の頂点: 4 個
- S の線分の端点: 高々 $2n$ 個
- S の線分の端点から上にのばした鉛直線の上端: 高々 $2n$ 個
- S の線分の端点から下にのばした鉛直線の下端: 高々 $2n$ 個

よって高々 $6n+4$ 個である。

台形の個数は、頂点の個数と Euler の公式 $f \leq 2v-4$ で評価すると高々 $12n+12$ 個となるが、これよりタイトな評価を考える。

各台形 Δ の左端の状況 (a)–(e) に合わせて、 $lefte(\Delta)$ を次のように定める (図 3)。 $lefte(\Delta)$ は S の線分か R の下辺のいずれかである。

(a) $lefte(\Delta) = bottom(\Delta)$ とする。

(b) $lefte(\Delta) = top(\Delta)$ とする。

(c) $\mathit{lefte}(\Delta) = \mathit{bottom}(\Delta)$ とする。

(d) $\mathit{lefte}(\Delta)$ は $\mathit{leftp}(\Delta)$ を右端に持つ S の線分とする。該当する線分が複数ある場合はどれでもよい。

(e) $\mathit{lefte}(\Delta)$ は R の下辺とする。

こうすると、 R の下辺が対応付けられた台形は 1 個、 S の各線分が対応付けられた台形は (a)(b)(c) で合わせて高々 2 個、 (d) で高々 1 個であり、台形の個数は全部で高々 $3n+1$ 個である (台形の個数の上限 $3n+1$ は平面走査法でも示せる。問題 6.12)。 □

一つの台形 Δ と鉛直線を共有する台形は高々 4 個。 Δ の左辺と上辺を共有する台形 (もしあれば) を Δ の左上隣 (upper left neighbor of Δ) という。同様に、左下隣、右上隣、右下隣を定める。台形地図を覚えるときは、各台形 Δ に対して、 $\mathit{top}(\Delta), \mathit{bottom}(\Delta), \mathit{leftp}(\Delta), \mathit{rightp}(\Delta)$ と、左上隣、左下隣、右上隣、右下隣となる台形を覚えておく。

6.2 確率的逐次添加アルゴリズム

アルゴリズム TrapezoidalMap(S) (図 4)
入力: 一般の位置にある n 本の線分の集合 S
出力: 台形地図 $\mathcal{T}(S)$ と検索構造 $\mathcal{D}$

1. 全体を囲む長方形 R を決定する。
2. 線分をランダムな順に並べ換える。この順序を $s_1, \ldots, s_n$ とする。
3. 台形地図と検索構造を初期化する。
4. 線分を 1 本ずつ追加する。

台形地図と検索構造に線分を追加するには、線分が交わる台形をすべて求めて、その一つ一つに対して変形をすればよい。そのためには、まず線分の左端がどの台形に属するかを決定して、そこから右に進んでいけばよい。

線分を追加する順序によっては、 検索構造 $\mathcal{D}$ のサイズが $\Theta(n^2)$、深さが $\Theta(n)$ になる例がある (図 5; 問題 6.1, 6.2)。この場合、台形地図と検索構造の構築に $\Theta(n^2)$ 時間かかる。

定理 6.3. S を一般の位置にある n 本の線分の集合とする。

(i) アルゴリズム TrapezoidalMap(S) は $O(n \log n)$ 期待時間で台形地図 $\mathcal{T}(S)$ と検索構造 $\mathcal{D}$ を構成する。

(ii) 構成される検索構造 $\mathcal{D}$ のサイズの期待値は $O(n)$

(iii) 任意の点 q に対し、構成された検索構造 $\mathcal{D}$ を使って q の属する台形を $O(\log n)$ 期待時間で求められる。

証明 S に属する n 本の線分を、選ばれた順に $s_1, \ldots, s_n$ とし、 $S_i = \{s_1, \ldots, s_i\}$ とおく。

(iii)(ii)(i) の順に示す。

まず (iii) から。 点 q が与えられたとき、検索構造 $\mathcal{D}$ で q を探す際に辿るパスの長さ X の期待値が $O(\log n)$ であることを示せばよい。

このパス上の内点のうち、線分 s_i を添加したときにできた内点の個数を X_i とすると、 $X = \sum_{i=1}^{n} X_i$ である。よって、 $\mathrm{E}[X] = \sum_{i=1}^{n} \mathrm{E}[X_i]$ である。

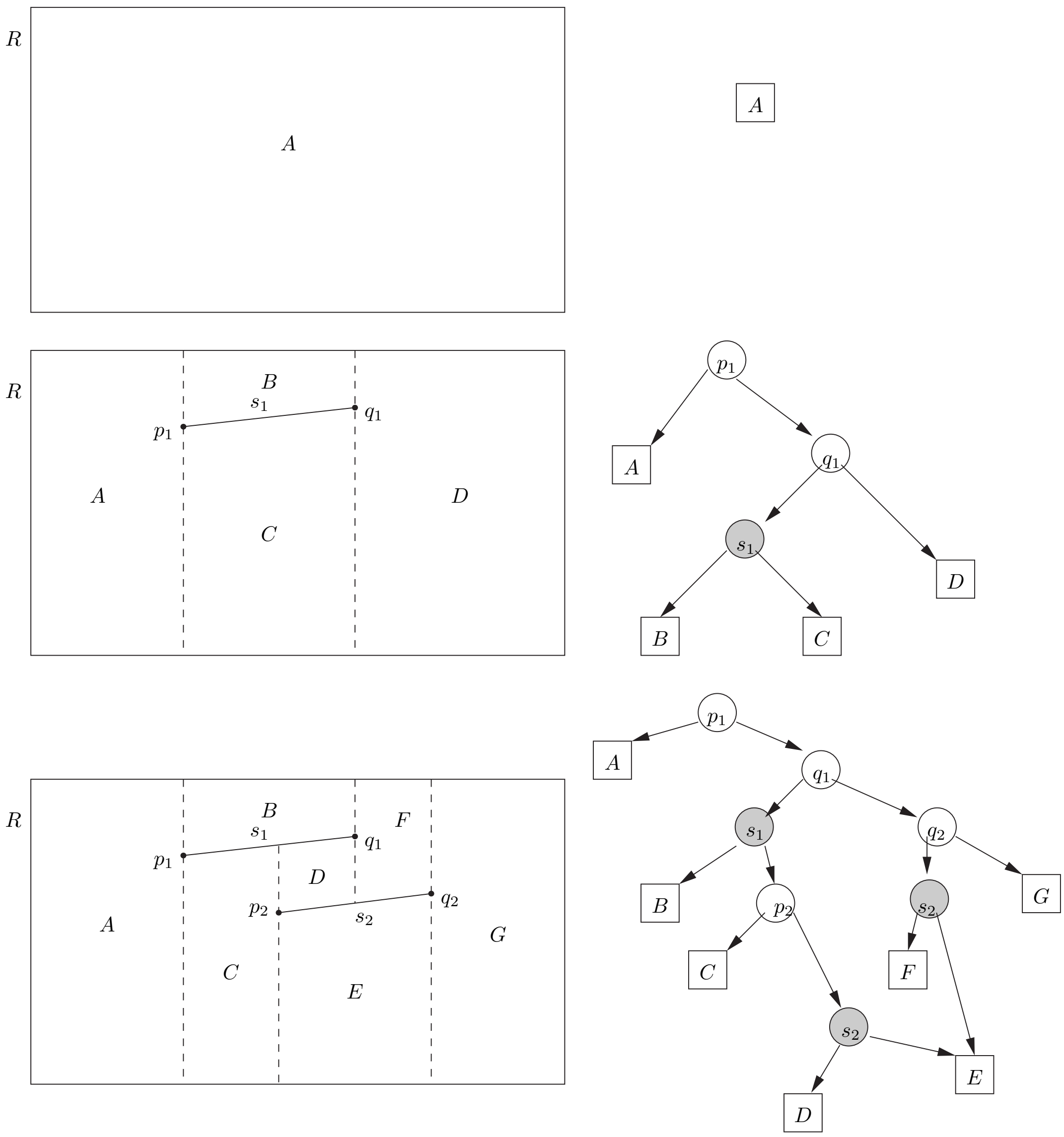

図 4: 台形地図と検索構造の構築

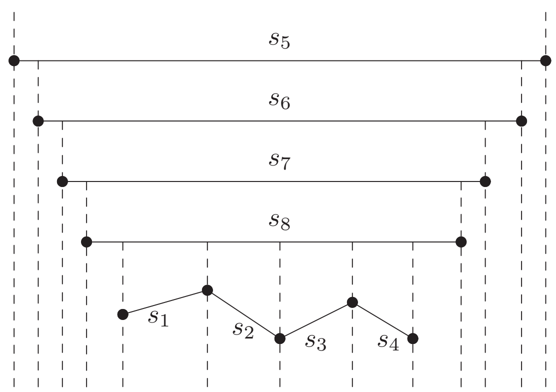

図 5: 検索構造 $\mathcal{D}$ の大きさが $\Theta(n^2)$、深さが $\Theta(n)$ になる例

S_i の台形地図 $\mathcal{T}(S_i)$ で点 q を含む台形を $\Delta_q(S_i)$ と書く ($\mathcal{T}(S_i)$ や $\Delta_q(S_i)$ は S_i のみに依存し、 $s_1, \ldots, s_i$ の順序には依存しないことに注意する)。 $\Delta_q(S_i)$ が線分 s_i を添加したときにできたものである場合、すなわち $\Delta_q(S_i) \neq \Delta_q(S_{i-1})$ である場合、 $X_i \leq 3$ である。そうでない場合、 $X_i = 0$ である。そこで、

$$Y_i = \begin{cases} 1 & \Delta_q(S_i) \neq \Delta_q(S_{i-1}) \text{ のとき} \\ 0 & \text{そうでないとき} \end{cases}$$

とおき $P_i = \Pr[Y_i = 1]$ とおくと、 $X_i \leq 3Y_i$ であり、 $\mathrm{E}[X_i] \leq 3P_i$ である。

$P_i \leq 4/i$ であることを後方解析 (backwards analysis) によって示そう。$S_i \subseteq S$ を一つ決めて固定する。このとき、 $Y_i = 1$ となるのは、 S_i から s_i を除くと台形 $\Delta_q(S_i)$ が消える場合である。こうなるのは、次のいずれかが成り立つ場合に限る。

1. $\mathit{top}(\Delta_q(S_i)) = s_i$ のとき
2. $\mathit{bottom}(\Delta_q(S_i)) = s_i$ のとき
3. $\mathit{leftp}(\Delta_q(S_i))$ が s_i の端点で、かつ $s_1, \ldots, s_{i-1}$ の端点でないとき
4. $\mathit{rightp}(\Delta_q(S_i))$ が s_i の端点で、かつ $s_1, \ldots, s_{i-1}$ の端点でないとき

S_i のうち $\mathit{top}(\Delta_q(S_i))$ が s_i に選ばれる確率は $1/i$ なので、 1 が成り立つ確率は $1/i$ である。同様に、 2 が成り立つ確率も $1/i$ である。また、 3 が成り立つためには

$$\mathit{leftp}(\Delta_q(S_i)) \text{ が 1 本だけの線分の端点であること} \tag{*}$$

が必要で、 (*) が成り立つときに 3 が成り立つ確率は $1/i$ である ((*) が成り立たないときに 3 が成り立つ確率は 0)。よって、 3 が成り立つ確率は $1/i$ 以下である。同様に、 4 が成り立つ確率も $1/i$ 以下である。以上より、1, 2, 3, 4 のいずれかが成り立つ確率は $4/i$ 以下なので、 $Y_i = 1$ となる確率は $4/i$ 以下である。

これは S_i を固定したときの条件付き確率だが、 S_i をどう固定しても $4/i$ 以下なので、条件をなくして $Y_i = 1$ となる確率を考えても $4/i$ 以下である。すなわち $P_i \leq 4/i$ である。

$$\mathrm{E}[X] = \sum_{i=1}^{n} \mathrm{E}[X_i] \leq 3\sum_{i=1}^{n} P_i \leq 12H_i, \quad (\text{ただし } H_i = 1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n})$$

であり、 $H_i \leq 1 + \ln n$ より $\mathrm{E}[X] \leq 12(1 + \ln n) = O(\log n)$ である。

次に (ii) を示す。補題 6.2 より、検索構造 $\mathcal{D}$ の葉節点の個数は高々 $3n+1$ である。 s_i を添加したときに作られる台形の個数を k_i とおく。このとき、作られる内点の個数は $k_i - 1$ である (問題 6.11[1])。よって、検索構造 $\mathcal{D}$ の内点の個数は $\sum_{i=1}^{n}(k_i - 1)$ であり、節点の個数は全部で高々 $3n + 1 + \sum_{i=1}^{n}(k_i - 1) = 2n + 1 + \sum_{i=1}^{n} k_i$ である。 $\mathrm{E}[k_i] = O(1)$ を示そう。

先ほどと同様に S_i を一つ決めて固定する。 $\Delta \in \mathcal{T}(S_i)$ と $s \in S_i$ に対し、確率変数 $\delta(\Delta, s)$ を

$$\delta(\Delta, s) = \begin{cases} 1 & S_i \text{ から } s \text{ を取り除くと } \Delta \text{ が消えるとき} \\ 0 & \text{そうでないとき} \end{cases}$$

と定めると、先ほどの議論から、各 $\Delta \in \mathcal{T}(S_i)$ に対して $\delta(\Delta, s) = 1$ となる s は高々 4 個である。すなわち

$$\sum_{s \in S_i} \delta(\Delta, s) \leq 4$$

[1] $\mathcal{T}(S_{i-1})$ の台形の中で s_i と交わるものの個数を l とする。このとき、 s_i を添加すると、 l 個の葉節点が内点に置き換わり、 k_i 個の新しい葉節点が作られる。この k_i 個の新しい葉節点の入次数の和は $k_i + l - 1$ なので (ここは本当はもっと説明が必要)、内点の増分は $(k_i + l - 1) - l = k_i - 1$ である。

であり、補題 6.2 より

$$\sum_{\Delta \in \mathcal{T}(S_i)} \sum_{s \in S_i} \delta(\Delta, s) \leq 4|\mathcal{T}(S_i)| \leq 12i + 4$$

一方、 k_i は $\delta(\Delta, s_i) = 1$ となる $\Delta \in \mathcal{T}(S_i)$ の個数なので、

$$k_i = \sum_{\Delta \in \mathcal{T}(S_i)} \delta(\Delta, s_i)$$

である。 S_i の i 本の線分のそれぞれが s_i になる確率は $1/i$ なので、

$$\mathrm{E}[k_i] = \frac{1}{i} \sum_{s \in S_i} \sum_{\Delta \in \mathcal{T}(S_i)} \delta(\Delta, s) = \frac{1}{i} \sum_{\Delta \in \mathcal{T}(S_i)} \sum_{s \in S_i} \delta(\Delta, s) \leq \frac{12i + 4}{i} = 12 + \frac{4}{i} \leq 16$$

よって $\mathcal{D}$ の節点の個数の期待値は高々 $2n + 1 + 16n = 18n + 1 = O(n)$ である。

最後に (i) を示す。 s_i を添加するときにすることは、

- 線分 s_i の左端 p_i を探す: (iii) より $O(\log i)$ 期待時間
- 台形地図と検索構造を更新する: 新しくできる台形 k_i の 1 個 1 個に対して定数時間なので、合計だと (ii) より $O(1)$ 期待時間

よって、合計

$$\sum_{i=1}^{n} \bigl(O(\log i) + O(1)\bigr) = O(n \log n)$$

期待時間でできる。 □

ところで、検索構造 $\mathcal{D}$ の深さの期待値は? ……定理 6.3 だけだと、 $O(n)$ としかわからない。もし深さの期待値が $\Theta(n)$ だったら、どんな順序に対しても嫌な問い合わせ点 q が存在して、 q の問い合わせには $\Theta(n)$ 時間かかる、という可能性も考えられる。 6.4 節で、このアルゴリズムを詳しく解析して、構成される検索構造 $\mathcal{D}$ の深さの期待値が $O(\log n)$ であることを示す。

最後に、もとの点位置決定問題に戻る。平面領域分割が与えられたら、それを構成する辺全体を S だと思って台形地図 $\mathcal{T}(S)$ と検索構造 $\mathcal{D}$ を作り、各台形 $\Delta \in \mathcal{T}(S)$ に対してもとの領域分割における面の番号を記録しておく。こうしておくと、点 q が与えられたとき、 q の属する面が $\mathcal{D}$ での問い合わせをするだけで答えられる。

系 6.4. (i) 仮定 1 を満たす平面領域分割 (辺数 n) が与えられると、このアルゴリズムは $O(n \log n)$ 期待時間で台形地図と検索構造を構成する。

(ii) 構成される検索構造 $\mathcal{D}$ のサイズの期待値は $O(n)$

(iii) 仮定 2 を満たす任意の点 q に対し、構成された検索構造 $\mathcal{D}$ を使って q の属する面を $O(\log n)$ 期待時間で求められる。

6.3 縮退の取り扱い

前節で導入した仮定 1, 2 を取り除く。全体を少しだけ回転あるいは剪断変形 (shear transformation; 図 6) すればいいが、この方法には二つ問題がある。

- 変形の大きさを決めるのが大変 (問題 6.9)
- 数値計算の誤差の対策が必要

そこで、 symbolic perturbation を用いるとよい。

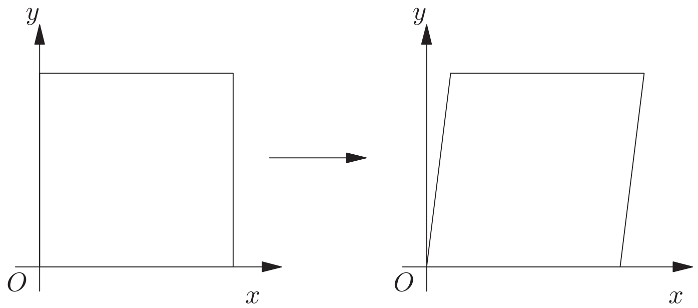

図 6: 剪断変形 $(x, y) \mapsto (x + \varepsilon y, y)$

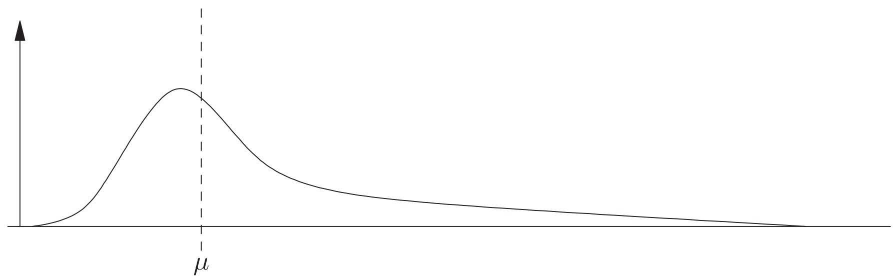

図 7: 尻尾の長い確率分布

6.4 確率分布の尻尾の評価

これまで検索構造のサイズや問い合わせに答えるのにかかる時間の期待値を評価してきたが、期待値の評価だけではわからないことがある。期待値から外れてとても時間がかかる確率はどのくらいだろうか (図 7)。

Markov の不等式　非負の値をとる確率変数 Z の期待値が μ のとき、任意の $\lambda > 0$ に対して

$$\Pr[Z \geq \lambda\mu] \leq \frac{1}{\lambda}$$

これを使うと、定理 6.3 (3) より、次のことはすぐにわかる。

系　S を一般の位置にある n 本の線分の集合、q を点とする。また $\lambda > 0$ とする。アルゴリズム TrapezoidalMap(S) が構成する検索構造 $\mathcal{D}$ を使って q の問い合わせを実行したとき、パスの長さが $12\lambda(1 + \ln n)$ 以上である確率は高々 $1/\lambda$ である。

もっと詳しく解析すると、次が示せる。

補題 6.6. S を一般の位置にある n 本の線分の集合、q を点とする。また $\lambda > 0$ とする。アルゴリズム TrapezoidalMap(S) が構成する検索構造 $\mathcal{D}$ を使って q の問い合わせを実行したとき、パスの長さが $3\lambda\ln(n+1)$ 以上である確率は高々

$$\frac{1}{(n+1)^{\lambda \ln 1.25 - 1}}$$

である。

証明の流れ　$\mathcal{D}$ を使って点 q の問い合わせを実行したときのパスの長さを X とおく。定理 6.3 の証明に出てきた Y_i は、

$$X \leq 3\sum_{i=1}^{n} Y_i, \quad Y_i \in \{0, 1\}, \quad \Pr[Y_i = 1] \leq 4/i$$

という性質を満たした。ここで Y_i と少し違う確率変数 Y_i' を考えることにより[2]、次の性質を満たすようにできる (詳細は略)。

$$Y_i \leq Y_i', \quad Y_i' \in \{0,1\}, \quad \Pr[Y_i'=1] \leq 4/i, \quad Y_1',\ldots,Y_n' \text{ は独立}$$

$Y' = \sum_{i=1}^n Y_i'$ とおく。 Y_i と違って Y_i' たちは独立であるので、任意の正の実数 α に対して

$$\mathrm{E}[\alpha^{Y'}] = \mathrm{E}[\prod_{i=1}^n \alpha^{Y_i'}] = \prod_{i=1}^n \mathrm{E}[\alpha^{Y_i'}]$$

が成り立つ[3]。ここで $\alpha = 1.25$ とおくと

$$\mathrm{E}[1.25^{Y_i'}] \leq 1 \cdot \left(1 - \frac{4}{i}\right) + 1.25 \cdot \frac{4}{i} = \frac{i+1}{i}$$

と評価でき、これより

$$\mathrm{E}[1.25^{Y'}] \leq \prod_{i=1}^n \frac{i+1}{i} = n+1$$

となる。よって、 Markov の不等式より、

$$\Pr[Y' \geq \lambda \ln(n+1)] = \Pr[1.25^{Y'} \geq 1.25^{\lambda \ln(n+1)}] \leq \frac{n+1}{1.25^{\lambda \ln(n+1)}} = \frac{1}{(n+1)^{\lambda \ln 1.25 - 1}}$$

であり、所望の不等式

$$\Pr[X \geq 3\lambda \ln(n+1)] \leq \Pr[Y' \geq \lambda \ln(n+1)] \leq \frac{1}{(n+1)^{\lambda \ln 1.25 - 1}}$$

が得られる。 □

補題 6.7. S を一般の位置にある n 本の線分の集合とする。また $\lambda > 0$ とする。アルゴリズム TrapezoidalMap(S) が構成する検索構造 $\mathcal{D}$ の深さが $3\lambda \ln(n+1)$ 以上である確率は高々

$$\frac{2}{(n+1)^{\lambda \ln 1.25 - 3}}$$

である。

証明 検索構造 $\mathcal{D}$ に対し、二つの点 q と q' の問い合わせが同じパスを辿るとき、 q と q' は等価 (equivalent) という。 S のスラブ分割における台形の個数を N とおくと、 $N \leq 2(n+1)^2$ である[4]。 S のスラブ分割における各台形の内部から 1 点ずつ選んできて $q_1,\ldots,q_N$ とおく。スラブ分割において同じ台形に含まれる点は、アルゴリズム TrapezoidalMap(S) でどんな順序が選ばれても等価になるので、 $\mathcal{D}$ の深さが $3\lambda \ln(n+1)$ より大きいなら、 $q_1,\ldots,q_N$ の中に $\mathcal{D}$ でのパスの長さが $3\lambda \ln(n+1)$ より長いものが存在する。補題 6.6 より、その確率は高々

$$2(n+1)^2 \times \frac{1}{(n+1)^{\lambda \ln 1.25 - 1}} = \frac{2}{(n+1)^{\lambda \ln 1.25 - 3}}$$

である。 □

定理 6.8. 一般の位置にある n 本の線分の集合 S に対し、

(i) 次のアルゴリズムは $O(n \log n)$ 期待時間で台形地図 $\mathcal{T}(S)$ と検索構造 $\mathcal{D}$ を構成する。

(ii) 構成される検索構造 $\mathcal{D}$ のサイズは常に $O(n)$

---

[2] この資料でいう Y_i' を本では X_i と呼んでいる。定理 6.3 のときと記号の使いかたが違うのが嫌だったので、この資料では名前を変えた。

[3] 一般に、独立な確率変数の和 Y' について、期待値からの外れ具合を評価したい場合、 α を正定数として $\alpha^{Y'}$ に Markov の不等式を適用するのは常套手段である。 cf. Chernoff bound. 問題 6.16 は α としてもっと良い値を選ぶことでできると思う。

[4] N の最大値は $(n+1)^2$ だと思うのだが……(図 1)。

(iii) 任意の点 q に対し、構成された検索構造 $\mathcal{D}$ を使って q の属する台形を $O(\log n)$ 最悪時間で求められる。

アルゴリズム TrapezoidalMap(S) を実行する。ただし、台形地図と検索構造 $\mathcal{D}$ を作っている途中で $\mathcal{D}$ のサイズが $76n$ を超えるか、 $\mathcal{D}$ の深さが $81\ln(n+1)$ を超えたら中止して、 TrapezoidalMap(S) の最初からやり直す。サイズが $76n$ 以下で深さが $81\ln(n+1)$ 以下の検索構造 $\mathcal{D}$ が構成できたら成功して終了する。成功するまで繰り返す。

証明のアイデア アルゴリズムが終了すれば正しいことや、 (ii)(iii) は明らかだろう。このアルゴリズムが $O(n\log n)$ 期待時間で終わることを示そう。 1 回の試行でかかる時間は成功しても失敗しても $O(n\log n)$ 時間なので、失敗する回数の期待値が $O(1)$ となることを示せばよい。

定理 6.3 (ii) より、 $\mathcal{D}$ のサイズの期待値は高々 $18n+1 \le 19n$ なので、節点数が多過ぎて失敗する確率は Markov の不等式より高々 1/4 である。一方、 $\mathcal{D}$ の深さが大き過ぎて失敗する確率は、補題 6.7 で $\lambda = 27$ とおくと、高々

$$\frac{2}{(n+1)^{27\ln 1.25-3}} \le \frac{2}{2^{27\ln 1.25-3}} < \frac{1}{4} \quad (\because \ln 1.25 = 0.2231\cdots > 2/9)$$

である。よって、失敗する確率は高々 $1/4+1/4=1/2$ である。これより、失敗する回数の期待値は高々

$$\frac{1}{2} + \left(\frac{1}{2}\right)^2 + \left(\frac{1}{2}\right)^3 + \cdots = 1$$

となる。 □

発展

6.5 節 と章末問題より:

- 動的版 (辺が増えたり減ったりしても大丈夫なようにする)
- 高次元
- 形を限定した場合 (凸の場合: 問題 6.5, 6.6; 長方形領域の場合; fat な領域の場合)
- Ray shooting (第 8 章; vertical ray shooting なら点位置決定と同じ手間, 問題 6.15)

また、 [2] の第 11 章に点位置決定問題を $O(n)$ 領域、 $O(n)$ 前処理時間、 $O(\log n)$ 問い合わせ時間で答える決定性アルゴリズムが紹介されている。

参考文献

[1] M. de Berg, O. Schwarzkopf, M. van Kreveld, M. Overmars: "Computational Geometry: Algorithms and Applications," Springer-Verlag, 1997. ISBN 3-540-61270-X. 日本語訳: 浅野哲夫訳、コンピュータ・ジオメトリ—計算幾何学: アルゴリズムと応用—、近代科学社、 2000。 ISBN 4-7649-0277-X.

[2] H. Edelsbrunner: "Algorithms in Combinatorial Geometry," Volume 10 of EATCS Monographs on Theoretical Computer Science, Springer-Verlag, 1987. ISBN 0-387-13722-X. 日本語訳: 今井浩・今井桂子訳、組合せ幾何学のアルゴリズム、共立出版、 1995。 ISBN 4-320-02724-8.