ANALYZING RANDOMIZED INCREMENTAL ALGORITHMS

Vera Sacristán

Seminar: Geometric Algorithms (MIRI)
Facultat d'Informàtica de Barcelona
Universitat Politècnica de Catalunya

A FRAMEWORK

A FRAMEWORK

General setting

- $X = \{x_1, \ldots, x_n\}$ is a random permutation of the input set.
- Π is the configuration set, i.e., it contains the objects computed by the algorithm.
- $D(\Delta)$ is the set of $x \in X$ defining each $\Delta \in \Pi$.
- $d = \max_{\Delta \in \Pi} |D(\Delta)|$ is constant.
- $K(\Delta)$ is the set of $x \in X$ killing each $\Delta \in \Pi$.
- $K(\Delta) \cap D(\Delta) = \emptyset$ for all $\Delta \in \Pi$.

As the algorithm advances:

- Δ is active for $X_r = \{x_1, \ldots, x_r\}$ if
  - $D(\Delta) \subseteq X_r$
  - $K(\Delta) \cap X_r = \emptyset$.
- $T_r = \{\Delta \mid \Delta \text{ is active for } X_r\}$.

The algorithm computes $T_1, \ldots, T_n$.

A FRAMEWORK

General setting

- $X = \{x_1, \dots, x_n\}$ is a random permutation of the input set.
- Π is the configuration set, i.e., it contains the objects computed by the algorithm.
- $D(\Delta)$ is the set of $x \in X$ defining each $\Delta \in \Pi$.
- $d = \max_{\Delta \in \Pi} |D(\Delta)|$ is constant.
- $K(\Delta)$ is the set of $x \in X$ killing each $\Delta \in \Pi$.
- $K(\Delta) \cap D(\Delta) = \emptyset$ for all $\Delta \in \Pi$.

As the algorithm advances:

- Δ is active for $X_r = \{x_1, \dots, x_r\}$ if
  - $D(\Delta) \subseteq X_r$
  - $K(\Delta) \cap X_r = \emptyset$.
- $T_r = \{\Delta \mid \Delta \text{ is active for } X_r\}$.

The algorithm computes $T_1, \dots, T_n$.

Example 1: Intersecting halfplanes

- $X = \{h_1, \dots, h_n\}$ is a finite set of halfplanes.
- Π is the set of all intersection points of all the lines defining the half-planes.
- $D(\Delta)$ are the two lines defining a given vertex Δ ($d = 2$).
- $K(\Delta)$ are all halfplanes not containing point Δ.
- T_r is the set of vertices of $\cap_{i=1}^{r} h_i$.

A FRAMEWORK

General setting

- $X = \{x_1, \ldots, x_n\}$ is a random permutation of the input set.
- Π is the configuration set, i.e., it contains the objects computed by the algorithm.
- $D(\Delta)$ is the set of $x \in X$ defining each $\Delta \in \Pi$.
- $d = \max_{\Delta \in \Pi} |D(\Delta)|$ is constant.
- $K(\Delta)$ is the set of $x \in X$ killing each $\Delta \in \Pi$.
- $K(\Delta) \cap D(\Delta) = \emptyset$ for all $\Delta \in \Pi$.

As the algorithm advances:

- Δ is active for $X_r = \{x_1, \ldots, x_r\}$ if
  - $D(\Delta) \subseteq X_r$
  - $K(\Delta) \cap X_r = \emptyset$.
- $T_r = \{\Delta \mid \Delta \text{ is active for } X_r\}$.

The algorithm computes $T_1, \ldots, T_n$.

Example 2: Delaunay triangulation

- $X = \{x_1, \ldots, x_n\}$ is a set of points.
- Π is the set of all triangles defined by 3 input points.
- $D(\Delta)$ are the three vertices of the triangle Δ $(d = 3)$.
- $K(\Delta)$ are all input points lying in the interior of the circumcircle of triangle Δ.
- T_r is the set of Delaunay triangles of $\{x_1, |dots, x_r\}$.

A FRAMEWORK

General setting

- $X = \{x_1, \ldots, x_n\}$ is a random permutation of the input set.
- Π is the configuration set, i.e., it contains the objects computed by the algorithm.
- $D(\Delta)$ is the set of $x \in X$ defining each $\Delta \in \Pi$.
- $d = \max_{\Delta \in \Pi} |D(\Delta)|$ is constant.
- $K(\Delta)$ is the set of $x \in X$ killing each $\Delta \in \Pi$.
- $K(\Delta) \cap D(\Delta) = \emptyset$ for all $\Delta \in \Pi$.

As the algorithm advances:

- Δ is active for $X_r = \{x_1, \ldots, x_r\}$ if
  - $D(\Delta) \subseteq X_r$
  - $K(\Delta) \cap X_r = \emptyset$.
- $T_r = \{\Delta \mid \Delta \text{ is active for } X_r\}$.

The algorithm computes $T_1, \ldots, T_n$.

Example 3: Convex hull in 3D

- $X = \{x_1, \ldots, x_n\}$ is a set of points.
- Π is the set of all possible flaps (x_i, x_j, x_k, x_l).
- $D(\Delta)$ are the are the four points defining the flap Δ ($d = 4$).
- $K(\Delta)$ are all input points that can see exactly one of the two facets of the flap Δ.
  In other words, if $\Delta = (p, q, s, t)$, then $K(\Delta)$ is the set of input points for which $\overline{pq}$ is a horizon edge.
- T_r is the set of all possible flaps in $CH(x_1, \ldots, x_r) = CH(X_r)$.
  In other words, $\Delta = (p, q, s, t) \in T_r$ iff $\overline{pq}$ and $\overline{ps}$ are edges of a facet, and $\overline{pq}$ and $\overline{qt}$ are edges of another (different) facet of $CH(X_r)$.

A FRAMEWORK

Theorem 1 (9.14 in reference 1)

The expected size of the structural change at the r-th step of the randomized incremental algorithm, i.e., the expected number of Δ in $T_r \setminus T_{r-1}$, is bounded above as follows:

$$E[|T_r \setminus T_{r-1}|] \leq \frac{d}{r} E[|T_r|].$$

A FRAMEWORK

Theorem 1 (9.14 in reference 1)

The expected size of the structural change at the r-th step of the randomized incremental algorithm, i.e., the expected number of Δ in $T_r \setminus T_{r-1}$, is bounded above as follows:

$$E[|T_r \setminus T_{r-1}|] \leq \frac{d}{r} E[|T_r|].$$

Proof: By backwards analysis.

Consider $X_r = \{x_1, \ldots, x_{r-1}, x_r\}$ and its corresponding T_r. Remove x_r from X_r. If Δ disappears, it means that $x_r \in D(\Delta)$. We know there are at most $d|T_r|$ such pairs (x, Δ). Therefore,

$$\sum_{x \in X_r} |\{\Delta \in T_r : x \in D(\Delta)\}| \leq d|T_r|.$$

Since there exist r equally probable points $x \in X_r$ which can be x_r, we obtain the following bound for the expected number of Δ disappearing from X_r when removing x_r: $\leq \frac{d}{r}|T_r|$.
The average among all possible sets X_r gives the desired bound:

$$E[|T_r \setminus T_{r-1}|] \leq \frac{d}{r} E[|T_r|].$$

A FRAMEWORK

Theorem 2 (9.15 in reference 1)

The overall expected cost of all the location steps, i.e., the expected number of killing done along the entire algorithm, is bounded above as follows:

$$E\left[\sum_{\Delta} |K(\Delta)|\right] \leq \sum_{r=1}^{n} d^2 \; \frac{n-r}{r} \; \frac{E[|T_r|]}{r}.$$

A FRAMEWORK

Proof:

$$E\left[\sum_{\Delta} |K(\Delta)|\right] = E\left[\sum_{r=1}^{n} \sum_{\Delta \in T_r \setminus T_{r-1}} |K(\Delta)|\right] = E\left[\sum_{r=1}^{n} \sum_{y \in X \setminus X_r} k(X_r, y, x_r)\right]$$

Sum over all configurations Δ created by the algorithm along all steps.

A FRAMEWORK

Proof:

$$E\left[\sum_{\Delta}|K(\Delta)|\right] = E\left[\sum_{r=1}^{n}\sum_{\Delta\in T_r\setminus T_{r-1}}|K(\Delta)|\right] = E\left[\sum_{r=1}^{n}\sum_{y\in X\setminus X_r}k(X_r,y,x_r)\right]$$

where $k(X_r, y, x_r)$ is the number of $\Delta \in T_r$ such that $y \in K(\Delta)$ and $x_r \in D(\Delta)$

A FRAMEWORK

Proof:

$$E\left[\sum_{\Delta} |K(\Delta)|\right] = E\left[\sum_{r=1}^{n} \sum_{\Delta \in T_r \setminus T_{r-1}} |K(\Delta)|\right] = E\left[\sum_{r=1}^{n} \sum_{y \in X \setminus X_r} k(X_r, y, x_r)\right]$$

where $k(X_r, y, x_r)$ is the number of $\Delta \in T_r$ such that $y \in K(\Delta)$ and $x_r \in D(\Delta)$

$$E\left[\sum_{y \in X \setminus X_r} k(X_r, y, x_r)\right] \leq \frac{d}{r} \sum_{y \in X \setminus X_r} k(X_r, y) = d\,\frac{n-r}{r}\,E\left[k(X_r, x_{r+1})\right] = d\,\frac{n-r}{r}\,E\left[|T_r \setminus T_{r+1}|\right]$$

The probability that $x \in D(\Delta)$ for a given $\Delta \in T_r$ is $\leq \frac{d}{r}$.

Therefore $E[k(X_r, y, x_r)] \leq \frac{d}{r} k(X_r, y)$, where $k(X_r, y)$ is the number of Δ such that $y \in K(\Delta)$.

A FRAMEWORK

Proof:

$$E\left[\sum_{\Delta} |K(\Delta)|\right] = E\left[\sum_{r=1}^{n} \sum_{\Delta \in T_r \setminus T_{r-1}} |K(\Delta)|\right] = E\left[\sum_{r=1}^{n} \sum_{y \in X \setminus X_r} k(X_r, y, x_r)\right]$$

where $k(X_r, y, x_r)$ is the number of $\Delta \in T_r$ such that $y \in K(\Delta)$ and $x_r \in D(\Delta)$

$$E\left[\sum_{y \in X \setminus X_r} k(X_r, y, x_r)\right] \leq \frac{d}{r} \sum_{y \in X \setminus X_r} k(X_r, y) = d\,\frac{n-r}{r} E\left[k(X_r, x_{r+1})\right] = d\,\frac{n-r}{r} E\left[|T_r \setminus T_{r+1}|\right]$$

Every $y \in X - X_r$ has the same probability $\frac{1}{n-r}$ of being x_{r+1}

Therefore

$$E[k(X_r, x_{r+1})] = \frac{1}{n-r} \sum_{y \in X \setminus X_r} k(X_r, y)$$

A FRAMEWORK

Proof:

$$E\left[\sum_{\Delta} |K(\Delta)|\right] = E\left[\sum_{r=1}^{n} \sum_{\Delta \in T_r \setminus T_{r-1}} |K(\Delta)|\right] = E\left[\sum_{r=1}^{n} \sum_{y \in X \setminus X_r} k(X_r, y, x_r)\right]$$

where $k(X_r, y, x_r)$ is the number of $\Delta \in T_r$ such that $y \in K(\Delta)$ and $x_r \in D(\Delta)$

$$E\left[\sum_{y \in X \setminus X_r} k(X_r, y, x_r)\right] \leq \frac{d}{r} \sum_{y \in X \setminus X_r} k(X_r, y) = d\,\frac{n-r}{r}\,E\left[k(X_r, x_{r+1})\right] = d\,\frac{n-r}{r}\,E\left[|T_r \setminus T_{r+1}|\right]$$

Recall that $k(X_r, y)$ is the number of Δ such that $y \in K(\Delta)$

Proof:

$$E\left[\sum_{\Delta}|K(\Delta)|\right] = E\left[\sum_{r=1}^{n}\sum_{\Delta\in T_r\setminus T_{r-1}}|K(\Delta)|\right] = E\left[\sum_{r=1}^{n}\sum_{y\in X\setminus X_r}k(X_r,y,x_r)\right]$$

$$E\left[\sum_{y\in X\setminus X_r}k(X_r,y,x_r)\right] \leq \frac{d}{r}\sum_{y\in X\setminus X_r}k(X_r,y) = d\,\frac{n-r}{r}\,E\left[k(X_r,x_{r+1})\right] = d\,\frac{n-r}{r}\,E\left[|T_r\setminus T_{r+1}|\right]$$

A FRAMEWORK

Proof:

$$E\left[\sum_{\Delta}|K(\Delta)|\right] = E\left[\sum_{r=1}^{n}\sum_{\Delta\in T_r\setminus T_{r-1}}|K(\Delta)|\right] = E\left[\sum_{r=1}^{n}\sum_{y\in X\setminus X_r}k(X_r,y,x_r)\right]$$

$$E\left[\sum_{y\in X\setminus X_r}k(X_r,y,x_r)\right] \leq \frac{d}{r}\sum_{y\in X\setminus X_r}k(X_r,y) = d\,\frac{n-r}{r}\,E\left[k(X_r,x_{r+1})\right] = d\,\frac{n-r}{r}\,E\left[|T_r\setminus T_{r+1}|\right]$$

A FRAMEWORK

Proof:

$$E\left[\sum_{\Delta}|K(\Delta)|\right] = E\left[\sum_{r=1}^{n}\sum_{\Delta\in T_r\setminus T_{r-1}}|K(\Delta)|\right] = E\left[\sum_{r=1}^{n}\sum_{y\in X\setminus X_r}k(X_r,y,x_r)\right]$$

$$= \sum_{r=1}^{n} d\,\frac{n-r}{r}\,E\left[|T_r\setminus T_{r+1}|\right] = \sum_{\Delta} d\,\frac{n-(j(\Delta)-1)}{j(\Delta)-1}$$

$$E\left[\sum_{y\in X\setminus X_r}k(X_r,y,x_r)\right] \le \frac{d}{r}\sum_{y\in X\setminus X_r}k(X_r,y) = d\,\frac{n-r}{r}\,E\left[k(X_r,x_{r+1})\right] = d\,\frac{n-r}{r}\,E\left[|T_r\setminus T_{r+1}|\right]$$

A FRAMEWORK

Proof:

$$E\left[\sum_{\Delta} |K(\Delta)|\right] = E\left[\sum_{r=1}^{n} \sum_{\Delta \in T_r \setminus T_{r-1}} |K(\Delta)|\right] = E\left[\sum_{r=1}^{n} \sum_{y \in X \setminus X_r} k(X_r, y, x_r)\right]$$

$$= \sum_{r=1}^{n} d\, \frac{n-r}{r}\, E\left[|T_r \setminus T_{r+1}|\right] = \sum_{\Delta} d\, \frac{n-(j(\Delta)-1)}{j(\Delta)-1}$$

Sum over all Δ first created and later on killed along the algorithm

$j(\Delta)$ is the iteration where Δ is killed

A FRAMEWORK

Proof:

$$E\left[\sum_{\Delta}|K(\Delta)|\right] = E\left[\sum_{r=1}^{n}\sum_{\Delta\in T_r\setminus T_{r-1}}|K(\Delta)|\right] = E\left[\sum_{r=1}^{n}\sum_{y\in X\setminus X_r}k(X_r,y,x_r)\right]$$

$$= \sum_{r=1}^{n} d\,\frac{n-r}{r}\,E\left[|T_r\setminus T_{r+1}|\right] = \sum_{\Delta} d\,\frac{n-(j(\Delta)-1)}{j(\Delta)-1}$$

$$= \sum_{\Delta} d\left(\frac{n}{j(\Delta)-1}-1\right) \leq \sum_{\Delta} d\left(\frac{n}{i(\Delta)}-1\right) = \sum_{\Delta} d\,\frac{n-i(\Delta)}{i(\Delta)}$$

A FRAMEWORK

Proof:

$$E\left[\sum_{\Delta} |K(\Delta)|\right] = E\left[\sum_{r=1}^{n} \sum_{\Delta \in T_r \setminus T_{r-1}} |K(\Delta)|\right] = E\left[\sum_{r=1}^{n} \sum_{y \in X \setminus X_r} k(X_r, y, x_r)\right]$$

$$= \sum_{r=1}^{n} d \, \frac{n-r}{r} \, E\left[|T_r \setminus T_{r+1}|\right] = \sum_{\Delta} d \, \frac{n-(j(\Delta)-1)}{j(\Delta)-1}$$

$$= \sum_{\Delta} d \left(\frac{n}{j(\Delta)-1} - 1\right) \leq \sum_{\Delta} d \left(\frac{n}{i(\Delta)} - 1\right) = \sum_{\Delta} d \, \frac{n-i(\Delta)}{i(\Delta)}$$

$i(\Delta)$ is the iteration where Δ is created

$i(\Delta) \leq j(\Delta) - 1$

A FRAMEWORK

Proof:

$$E\left[\sum_{\Delta}|K(\Delta)|\right] = E\left[\sum_{r=1}^{n}\sum_{\Delta\in T_r\setminus T_{r-1}}|K(\Delta)|\right] = E\left[\sum_{r=1}^{n}\sum_{y\in X\setminus X_r}k(X_r,y,x_r)\right]$$

$$= \sum_{r=1}^{n} d\,\frac{n-r}{r}\,E\left[|T_r\setminus T_{r+1}|\right] = \sum_{\Delta} d\,\frac{n-(j(\Delta)-1)}{j(\Delta)-1}$$

$$= \sum_{\Delta} d\left(\frac{n}{j(\Delta)-1}-1\right) \leq \sum_{\Delta} d\left(\frac{n}{i(\Delta)}-1\right) = \sum_{\Delta} d\,\frac{n-i(\Delta)}{i(\Delta)}$$

A FRAMEWORK

Proof:

$$E\left[\sum_{\Delta}|K(\Delta)|\right] = E\left[\sum_{r=1}^{n}\sum_{\Delta\in T_r\setminus T_{r-1}}|K(\Delta)|\right] = E\left[\sum_{r=1}^{n}\sum_{y\in X\setminus X_r}k(X_r,y,x_r)\right]$$

$$= \sum_{r=1}^{n} d\,\frac{n-r}{r}\,E\left[|T_r\setminus T_{r+1}|\right] = \sum_{\Delta} d\,\frac{n-(j(\Delta)-1)}{j(\Delta)-1}$$

$$= \sum_{\Delta} d\left(\frac{n}{j(\Delta)-1}-1\right) \le \sum_{\Delta} d\left(\frac{n}{i(\Delta)}-1\right) = \sum_{\Delta} d\,\frac{n-i(\Delta)}{i(\Delta)}$$

$$\le \sum_{r=1}^{n} d\,\frac{n-r}{r}\,E\left[|T_r\setminus T_{r-1}|\right] \le \sum_{r=1}^{n} d^2\,\frac{n-r}{r}\,\frac{E[|T_r|]}{r}.$$

Sum over all Δ created at some iteration, no matter whether they end up being killed or not

A FRAMEWORK

Proof:

$$E\left[\sum_{\Delta}|K(\Delta)|\right] = E\left[\sum_{r=1}^{n}\sum_{\Delta\in T_r\setminus T_{r-1}}|K(\Delta)|\right] = E\left[\sum_{r=1}^{n}\sum_{y\in X\setminus X_r}k(X_r,y,x_r)\right]$$

$$= \sum_{r=1}^{n} d\,\frac{n-r}{r}\,E\left[|T_r\setminus T_{r+1}|\right] = \sum_{\Delta} d\,\frac{n-(j(\Delta)-1)}{j(\Delta)-1}$$

$$= \sum_{\Delta} d\left(\frac{n}{j(\Delta)-1}-1\right) \leq \sum_{\Delta} d\left(\frac{n}{i(\Delta)}-1\right) = \sum_{\Delta} d\,\frac{n-i(\Delta)}{i(\Delta)}$$

$$\leq \sum_{r=1}^{n} d\,\frac{n-r}{r}\,E\left[|T_r\setminus T_{r-1}|\right] \leq \sum_{r=1}^{n} d^2\,\frac{n-r}{r}\,\frac{E[|T_r|]}{r}.$$

By Theorem 1, $E[|T_r\setminus T_{r-1}|] \leq \frac{d}{r}E[|T_r|]$

Theorem 2 (9.15 in reference 1)

The overall expected cost of all the location steps, i.e., the expected number of killing done along the entire algorithm, is bounded above as follows:

$$E\left[\sum_{\Delta} |K(\Delta)|\right] \leq \sum_{r=1}^{n} d^2 \, \frac{n-r}{r} \, \frac{E[|T_r|]}{r}.$$

Application: Convex hull in 3D

$$\sum_{r=1}^{n} d^2 \, \frac{n-r}{r} \, \frac{E[|T_r|]}{r} = \sum_{r=1}^{n} 4^2 \, \frac{n-r}{r} \, \frac{6r-12}{r} = \sum_{r=1}^{n} 96 \, \frac{n-r}{r} \, \frac{r-2}{r} < 96 \sum_{r=1}^{n} \frac{n-r}{r}$$

$$< 96 \sum_{r=1}^{n} \frac{n-r}{r} < 96 \sum_{r=1}^{n} \frac{n}{r} = 96 \, n \sum_{r=1}^{n} \frac{1}{r} = O(n \log n)$$

APPLICATION TO CH IN 3D

Theorem 2 (9.15 in reference 1)

The overall expected cost of all the location steps, i.e., the expected number of killing done along the entire algorithm, is bounded above as follows:

$$E\left[\sum_{\Delta} |K(\Delta)|\right] \leq \sum_{r=1}^{n} d^2 \; \frac{n-r}{r} \; \frac{E[|T_r|]}{r}.$$

Application: Convex hull in 3D

$$\sum_{r=1}^{n} d^2 \; \frac{n-r}{r} \; \frac{E[|T_r|]}{r} = \sum_{r=1}^{n} 4^2 \; \frac{n-r}{r} \; \frac{6r-12}{r} = \sum_{r=1}^{n} 96 \; \frac{n-r}{r} \; \frac{r-2}{r} < 96 \sum_{r=1}^{n} \frac{n-r}{r}$$

$$< 96 \sum_{r=1}^{n} \frac{n-r}{r} < 96 \sum_{r=1}^{n} \frac{n}{r} = 96 \; n \sum_{r=1}^{n} \frac{1}{r} = O(n \log n)$$

$$\sum_{r=1}^{2^k-1} \frac{1}{r} = 1 + \left(\frac{1}{2} + \frac{1}{3}\right) + \cdots + \left(\frac{1}{2^{k-1}} + \cdots + \frac{1}{2^{k+1}-1}\right) \leq 1 + 2\,\frac{1}{2} + \cdots + 2^{k-1}\,\frac{1}{2^{k-1}} \leq \sum_{i=0}^{k-1} 1 = k$$

FURTHER READING

1. M. de Berg, O. Cheong, M. van Kreveld, M. Overmars

   Computational Geometry – Algorithms and Applications

   Springer-Verlag, 3rd edition, 2008.

   Please recall that the first and the second editions describe the authors in a different way: M. de Berg, M. van Kreveld, M. Overmars, O. Schwarzkopf.

2. K. Mulmuley

   Computational Geometry. An Introduction through Randomized Algorithms

   Prentice Hall, 1994.

3. R. Motwani, P. Raghavan

   Randomized Algorithms

   Cambridge University Press, 1995.