Advanced Techniques for Combinatorial Algorithms: Randomized Algorithms

Gianluca Della Vedova

Univ. Milano–Bicocca
https://gianluca.dellavedova.org

April 16, 2020

Gianluca Della Vedova

- Advanced Techniques for Combinatorial Algorithms

Gianluca Della Vedova

- Advanced Techniques for Combinatorial Algorithms
- `https://gitlab.com/dellavg/advanced-algorithms`

Gianluca Della Vedova

- Advanced Techniques for Combinatorial Algorithms
- https://gitlab.com/dellavg/advanced-algorithms
- https://gianluca.dellavedova.org

Gianluca Della Vedova

- Advanced Techniques for Combinatorial Algorithms
- `https://gitlab.com/dellavg/advanced-algorithms`
- `https://gianluca.dellavedova.org`
- `gianluca.dellavedova@unimib.it`

Karp-Rabin

Binary alphabet

Karp-Rabin

Binary alphabet

- $H(S) = \sum_{i=1}^{|S|} 2^{|S|-i} H(S[i])$

Karp-Rabin

Binary alphabet

- $H(S) = \sum_{i=1}^{|S|} 2^{|S|-i} H(S[i])$
- m-long sliding window on T

Karp-Rabin

Binary alphabet

- $H(S) = \sum_{i=1}^{|S|} 2^{|S|-i} H(S[i])$
- m-long sliding window on T
- $H(T[i+1 : i+m]) = ((H(T[i : i+m-1]) - T[i]) / 2 + 2^m T[i+m])$

Karp-Rabin

Binary alphabet

- $H(S) = \sum_{i=1}^{|S|} 2^{|S|-i} H(S[i])$
- m-long sliding window on T
- $H(T[i+1:i+m]) = ((H(T[i:i+m-1]) - T[i]) / 2 + 2^m T[i+m])$
- bit operations

Karp-Rabin

Binary alphabet

- $H(S) = \sum_{i=1}^{|S|} 2^{|S|-i} H(S[i])$
- m-long sliding window on T
- $H(T[i+1:i+m]) = ((H(T[i:i+m-1]) - T[i])/2 + 2^m T[i+m])$
- bit operations
- $T[i:i+m-1] = P \Leftrightarrow H(T[i:i+m-1]) = H(P)$

Karp-Rabin: problem

Numbers too large

Karp-Rabin: problem

Numbers too large

- RAM model: all numbers $O(n + m)$

Karp-Rabin: problem

Numbers too large

- RAM model: all numbers $O(n+m)$
- Solution: modulus p, random prime p

Karp-Rabin: problem

Numbers too large

- RAM model: all numbers $O(n+m)$
- Solution: modulus p, random prime p
- $H(T[i+1:i+m]) =$
  $((H(T[i:i+m-1]) - T[i])/2 + 2^m T[i+m]) \mod p$

Karp-Rabin: problem

Numbers too large

- RAM model: all numbers $O(n+m)$
- Solution: modulus p, random prime p
- $H(T[i+1:i+m]) =$
  $((H(T[i:i+m-1]) - T[i])/2 + 2^m T[i+m]) \mod p$
- Horner's formula. $2^{m-1}T[i+m] \mod p$ computed iteratively

Karp-Rabin: false positives

Kinds of error

Karp-Rabin: false positives

Kinds of error

- False positive (FP): reported false occurrence

Karp-Rabin: false positives

Kinds of error

- False positive (FP): reported false occurrence
- False negative (FN): occurrence not found

Karp-Rabin: false positives

Kinds of error

- False positive (FP): reported false occurrence
- False negative (FN): occurrence not found
- $H(T[i : i + m - 1]) = H(P) \Leftrightarrow T[i : i + m - 1] = P$

Karp-Rabin: false positives

Kinds of error

- False positive (FP): reported false occurrence
- False negative (FN): occurrence not found
- $H(T[i : i + m - 1]) = H(P) \Leftrightarrow T[i : i + m - 1] = P$
- $H(T[i : i + m - 1]) \mod p = H(P) \mod p \Leftarrow T[i : i + m - 1] = P$

Karp-Rabin: false positives

Error probability

$P[\#FP \geq 1] \leq O(nm/l)$ if p is randomly (w.r.t. uniform distribution) chosen among all prims $\leq l$

Karp-Rabin: false positives

Error probability

$P[\#FP \geq 1] \leq O(nm/I)$ if p is randomly (w.r.t. uniform distribution) chosen among all prims $\leq I$

Some values of I

Karp-Rabin: false positives

Error probability

$P[\#FP \geq 1] \leq O(nm/l)$ if p is randomly (w.r.t. uniform distribution) chosen among all prims $\leq l$

Some values of l

- $l = n^2 m \Rightarrow P[\#FP \geq 1] \leq 2.54/n$

Karp-Rabin: false positives

Error probability

$P[\#FP \geq 1] \leq O(nm/l)$ if p is randomly (w.r.t. uniform distribution) chosen among all prims $\leq l$

Some values of l

- $l = n^2 m \Rightarrow P[\#FP \geq 1] \leq 2.54/n$
- $l = nm^2 \Rightarrow P[\#FP \geq 1] \in O(1/m)$

Karp-Rabin: false positives

Error probability

$P[\#FP \geq 1] \leq O(nm/l)$ if p is randomly (w.r.t. uniform distribution) chosen among all prims $\leq l$

Some values of l

- $l = n^2 m \Rightarrow P[\#FP \geq 1] \leq 2.54/n$
- $l = nm^2 \Rightarrow P[\#FP \geq 1] \in O(1/m)$

Decreasing error probability

Choosing k random primes (independently, without repetitions).

Las Vegas vs. Monte Carlo

Classifying randomized algorithms

Las Vegas vs. Monte Carlo

Classifying randomized algorithms

- Las Vegas:

Las Vegas vs. Monte Carlo

Classifying randomized algorithms

- Las Vegas:
  - Always correct

Las Vegas vs. Monte Carlo

Classifying randomized algorithms

- Las Vegas:
  - Always correct
  - Sometimes not fast

Las Vegas vs. Monte Carlo

Classifying randomized algorithms

- Las Vegas:
  - Always correct
  - Sometimes not fast
  - Example: Quicksort with random pivot

Las Vegas vs. Monte Carlo

Classifying randomized algorithms

- Las Vegas:
  - Always correct
  - Sometimes not fast
  - Example: Quicksort with random pivot
- Monte Carlo:

Las Vegas vs. Monte Carlo

Classifying randomized algorithms

- Las Vegas:
  - Always correct
  - Sometimes not fast
  - Example: Quicksort with random pivot
- Monte Carlo:
  - Always fast

Las Vegas vs. Monte Carlo

Classifying randomized algorithms

- Las Vegas:
  - Always correct
  - Sometimes not fast
  - Example: Quicksort with random pivot
- Monte Carlo:
  - Always fast
  - Sometimes not correct

Las Vegas vs. Monte Carlo

Classifying randomized algorithms

- Las Vegas:
  - Always correct
  - Sometimes not fast
  - Example: Quicksort with random pivot
- Monte Carlo:
  - Always fast
  - Sometimes not correct
  - Karp-Rabin

The probabilistic method

Shows that an object exists

If the probability that an object exists is > 0

The probabilistic method

Shows that an object exists

If the probability that an object exists is > 0

Proposition

Let K_n be the complete graph with n vertices. If $\binom{n}{k} 2^{1-\binom{k}{2}} < 1$ then we can color the edges of K_n so that we have no monochromatic K_k subgraph.

The probabilistic method

Proposition

Let K_n be the complete graph with n vertices. If $\binom{n}{k}2^{1-\binom{k}{2}} < 1$ then we can color the edges of K_n so that we have no monochromatic K_k subgraph.

The probabilistic method

Proposition

Let K_n be the complete graph with n vertices. If $\binom{n}{k}2^{1-\binom{k}{2}} < 1$ then we can color the edges of K_n so that we have no monochromatic K_k subgraph.

Proof

The probabilistic method

Proposition

Let K_n be the complete graph with n vertices. If $\binom{n}{k}2^{1-\binom{k}{2}} < 1$ then we can color the edges of K_n so that we have no monochromatic K_k subgraph.

Proof

1. There are $2^{\binom{n}{2}}$ colorings of K_n.

The probabilistic method

Proposition

Let K_n be the complete graph with n vertices. If $\binom{n}{k}2^{1-\binom{k}{2}} < 1$ then we can color the edges of K_n so that we have no monochromatic K_k subgraph.

Proof

1. There are $2^{\binom{n}{2}}$ colorings of K_n.
2. Color each edge at random

The probabilistic method

Proposition

Let K_n be the complete graph with n vertices. If $\binom{n}{k}2^{1-\binom{k}{2}} < 1$ then we can color the edges of K_n so that we have no monochromatic K_k subgraph.

Proof

1. There are $2^{\binom{n}{2}}$ colorings of K_n.
2. Color each edge at random
3. There are $\binom{n}{k}$ k-cliques

The probabilistic method

Proposition

Let K_n be the complete graph with n vertices. If $\binom{n}{k}2^{1-\binom{k}{2}} < 1$ then we can color the edges of K_n so that we have no monochromatic K_k subgraph.

Proof

1. There are $2^{\binom{n}{2}}$ colorings of K_n.
2. Color each edge at random
3. There are $\binom{n}{k}$ k-cliques
4. Let A_i be the event that k-clique i is monochromatic

The probabilistic method

Proposition

Let K_n be the complete graph with n vertices. If $\binom{n}{k}2^{1-\binom{k}{2}} < 1$ then we can color the edges of K_n so that we have no monochromatic K_k subgraph.

Proof

1. There are $2^{\binom{n}{2}}$ colorings of K_n.
2. Color each edge at random
3. There are $\binom{n}{k}$ k-cliques
4. Let A_i be the event that k-clique i is monochromatic
5. $P[A_i] = 2^{1-\binom{k}{2}}$

The probabilistic method

Proposition

Let K_n be the complete graph with n vertices. If $\binom{n}{k}2^{1-\binom{k}{2}} < 1$ then we can color the edges of K_n so that we have no monochromatic K_k subgraph.

Proof

1. There are $2^{\binom{n}{2}}$ colorings of K_n.
2. Color each edge at random
3. There are $\binom{n}{k}$ k-cliques
4. Let A_i be the event that k-clique i is monochromatic
5. $P[A_i] = 2^{1-\binom{k}{2}}$
6. $P[\bigcup A_i] \leq \sum P[A_i] = \binom{n}{k}2^{1-\binom{k}{2}} < 1$

The probabilistic method

Proposition

Let K_n be the complete graph with n vertices. If $\binom{n}{k}2^{1-\binom{k}{2}} < 1$ then we can color the edges of K_n so that we have no monochromatic K_k subgraph.

Proof

1. There are $2^{\binom{n}{2}}$ colorings of K_n.
2. Color each edge at random
3. There are $\binom{n}{k}$ k-cliques
4. Let A_i be the event that k-clique i is monochromatic
5. $P[A_i] = 2^{1-\binom{k}{2}}$
6. $P[\bigcup A_i] \leq \sum P[A_i] = \binom{n}{k}2^{1-\binom{k}{2}} < 1$
7. $P[\bigcap \bar{A}_i] = 1 - P[\bigcup A_i] > 0$

The expectation method

Shows that an object exists

Let $E[X]$ be the expected value of event X. Then there exists an event with value $\leq E[X]$ and an event with value $\geq E[X]$

The expectation method

Shows that an object exists

Let $E[X]$ be the expected value of event X. Then there exists an event with value $\leq E[X]$ and an event with value $\geq E[X]$

Max Cut

The expectation method

Shows that an object exists

Let $E[X]$ be the expected value of event X. Then there exists an event with value $\leq E[X]$ and an event with value $\geq E[X]$

Max Cut

- Let $G = (V, E)$ be an undirected graph, with $|V| = n$, $|E| = m$.

The expectation method

Shows that an object exists

Let $E[X]$ be the expected value of event X. Then there exists an event with value $\leq E[X]$ and an event with value $\geq E[X]$

Max Cut

- Let $G = (V, E)$ be an undirected graph, with $|V| = n$, $|E| = m$.
- The cut associated with the bipartition (V_1, V_2) of V is $E \cap V_1 \times V_2$.

The expectation method

Shows that an object exists

Let $E[X]$ be the expected value of event X. Then there exists an event with value $\leq E[X]$ and an event with value $\geq E[X]$

Max Cut

- Let $G = (V, E)$ be an undirected graph, with $|V| = n$, $|E| = m$.
- The cut associated with the bipartition (V_1, V_2) of V is $E \cap V_1 \times V_2$.
- There exists a cut with at least $m/2$ edges.

The expectation method

Random cut

The expectation method

Random cut

- For each vertex v, assign v to a side with $p = 1/2$

The expectation method

Random cut

- For each vertex v, assign v to a side with $p = 1/2$

Proof

The expectation method

Random cut

- For each vertex v, assign v to a side with $p = 1/2$

Proof

- $X_i = 1$ if edge i in the cut C, else 0

The expectation method

Random cut

- For each vertex v, assign v to a side with $p = 1/2$

Proof

- $X_i = 1$ if edge i in the cut C, else 0
- Probability of each edge in the random cut C: $1/2$. $E[X_i] = 1/2$

The expectation method

Random cut

- For each vertex v, assign v to a side with $p = 1/2$

Proof

- $X_i = 1$ if edge i in the cut C, else 0
- Probability of each edge in the random cut C: $1/2$. $E[X_i] = 1/2$
- $E[\sum X_i] = \sum E[X_i] = m/2$

The expectation method

Random cut

- For each vertex v, assign v to a side with $p = 1/2$

Proof

- $X_i = 1$ if edge i in the cut C, else 0
- Probability of each edge in the random cut C: 1/2. $E[X_i] = 1/2$
- $E[\sum X_i] = \sum E[X_i] = m/2$

Question

The algorithm?

The expectation method

Random cut

$q = P[|C| \geq m/2]$, for random cut C. Then $q \geq \frac{1}{m/2+1}$

The expectation method

Random cut

$q = P[|C| \geq m/2]$, for random cut C. Then $q \geq \frac{1}{m/2+1}$

Proof

$$\frac{m}{2} = E[C] = \sum_{i \leq m/2-1} i \cdot P[|C| = i] + \sum_{i \geq m/2} i \cdot P[|C| = i] \leq$$

$$\leq (m/2 - 1)(1 - q) + qm \Rightarrow q \geq \frac{1}{m/2 + 1}$$

The expectation method

Random cut

$q = P[|C| \geq m/2]$, for random cut C. Then $q \geq \frac{1}{m/2+1}$

Proof

$$\frac{m}{2} = E[C] = \sum_{i \leq m/2-1} i \cdot P[|C| = i] + \sum_{i \geq m/2} i \cdot P[|C| = i] \leq$$

$$\leq (m/2-1)(1-q) + qm \Rightarrow q \geq \frac{1}{m/2+1}$$

Question

How many random cuts are needed?

The sample and modify method

Independent set

G: undirected graph. Average degree $d = 2m/n$

Algorithm 1: Random Independent Set

$S \leftarrow$ a random sample of V, each vertex is picked with $p = 1/d$;
$I \leftarrow S$;
while *there exists an edge e in* $G|I$ **do**
    Remove an endpoint of e from I

The sample and modify method

Bounding $|I|$

The sample and modify method

Bounding $|I|$

- $E(G) = \frac{nd}{2}$

The sample and modify method

Bounding $|I|$

- $E(G) = \frac{nd}{2}$
- $E[|S|] = \frac{n}{d}$

The sample and modify method

Bounding $|I|$

- $E(G) = \frac{nd}{2}$
- $E[|S|] = \frac{n}{d}$
- $E[E(G|S)] = \frac{nd}{2}\frac{1}{d}\frac{1}{d} = \frac{n}{2d}$

The sample and modify method

Bounding $|I|$

- $E(G) = \frac{nd}{2}$
- $E[|S|] = \frac{n}{d}$
- $E[E(G|S)] = \frac{nd}{2}\frac{1}{d}\frac{1}{d} = \frac{n}{2d}$
- At most $E(G|S)$ are removed in the second step, hence $E[|I|] = E[|S|] - E[E(G|S)] = \frac{n}{2d}$