Randomness in Computing

CS 537

LECTURE 17

Last time before review

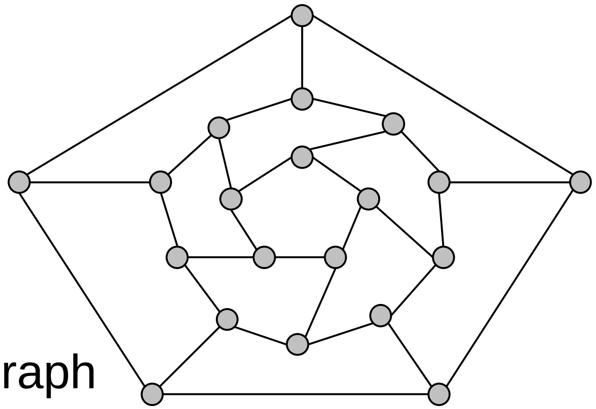
- Review Poisson approximation
- Application: max load
- Application: Coupon Collector
- Random graphs

Today

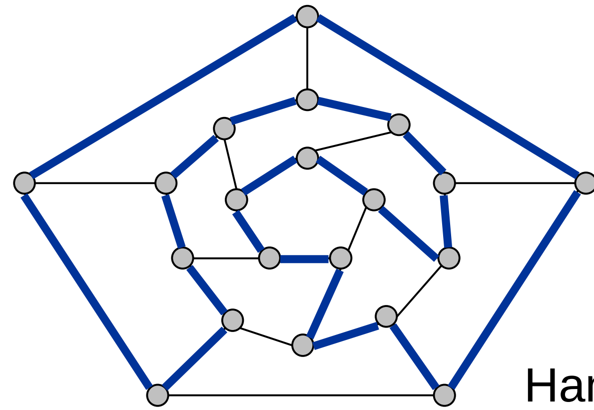
- Finding Hamiltonian cycles in random graphs

Finding Hamiltonian cycles

A Hamiltonian cycle is a cycle that visits each vertex exactly once.



Input graph



Hamiltonian cycle

- Finding Hamiltonian cycles is NP-hard.

Question

The property of having a Hamiltonian cycle is

A. Monotone increasing

B. Monotone decreasing

C. Not monotone

CS 537

Finding Hamiltonian cycles in random graphs

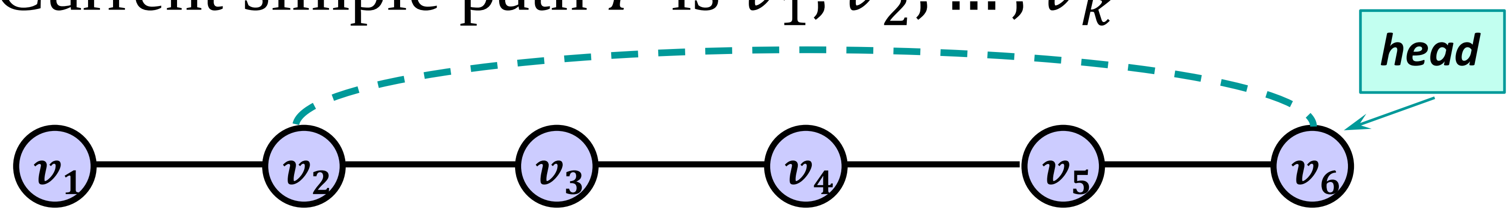
Main Theorem

Suppose $p \geq \frac{40 \ln n}{n}$. There is a polynomial time randomized algorithm that, given p and $G \sim G_{n,p}$, finds a Hamiltonian cycle in G with probability $1 - O\left(\frac{1}{n}\right)$.

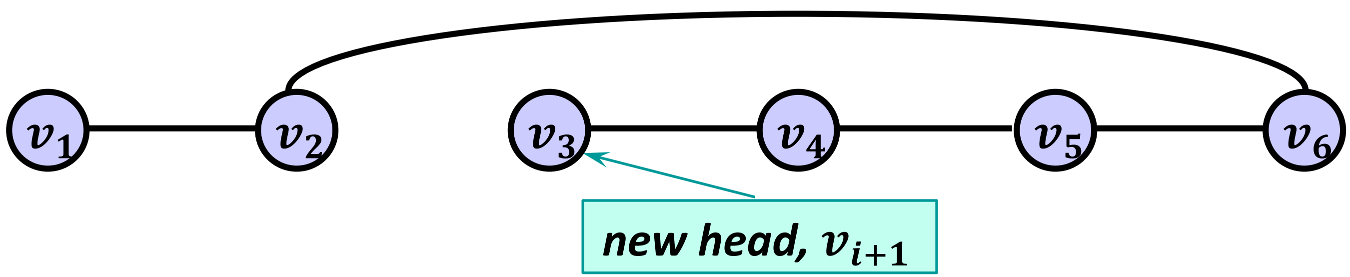
Corollary: Hamiltonian cycle exists in $G \sim G_{n,p}$ w. p. $1 - O\left(\frac{1}{n}\right)$.

Rotation operation

- Current simple path P is $v_1, v_2, \ldots, v_k$



- Rotation of P with an edge (v_k, v_i) in G, $i \in [k-1]$



  – If $i = k - 1$, no change.
  – If $k = n$ and $i = 1$,
  rotation edge (v_k, v_i) closes Hamiltonian path.

Hamiltonian cycle algorithm

Input: undirected n-node graph $G = (V, E)$ represented with adjacency lists
Output: Hamiltonian cycle or FAIL

For each node v, keep used-edge list UE(v) and unused-edge list UUE(v) to keep track which edges have been used in rotations while v was the head.

0. $\forall v$ initialize UE(v)$= \emptyset$ and UUE(v)$=$ adjacency list of v
1. Start with a uniformly random $v_1 \in V$ as head.
2. Repeat until a rotation edge closes Hamiltonian path **or** UUE(head)$= \emptyset$.
   a. Let $P = (v_1, \dots, v_k)$ denote the current path with head v_k
   b. Execute (i), (ii), (iii) w.p. $\frac{1}{n}, \frac{|UE(v_k)|}{n}, 1 - \frac{1}{n} - \frac{|UE(v_k)|}{n}$, respectively.
      i. Reverse P and make v_1 head.
      ii. Rotate P with a uniformly random edge e from UE(v_k).
      iii. Let (v_k, u) be the first edge on UUE(v_k). If $u \neq v_i$ for $i \in [k]$, extend P to $v_1, \dots, v_{k+1}$. O.w., rotate P with (v_k, u).
   c. Update UE and UUE lists.
3. Return Hamiltonian cycle if found; FAIL o.w.

Initial analysis

- **Simplifying assumption**: For all $v \in V$, list $UUE(v)$ is initialized to contain edges $(v, u) \forall u \neq v$ independently w.p. q

  These edges are in a uniformly random order.
- **Caution**: (u, v) could be in $UUE(u)$, but not in $UUE(v)$

Lemma (Head is chosen uniformly at random at every iteration)

Let V_t be a R.V. representing the head vertex at iteration t.

If algorithm has not terminated at iteration $t+1$, then $\Pr[V_{t+1} = u | V_t = u_t, \dots, V_1 = u_1] = 1/n \quad \forall u, u_t, \dots, u_1$

CS 537

Initial analysis

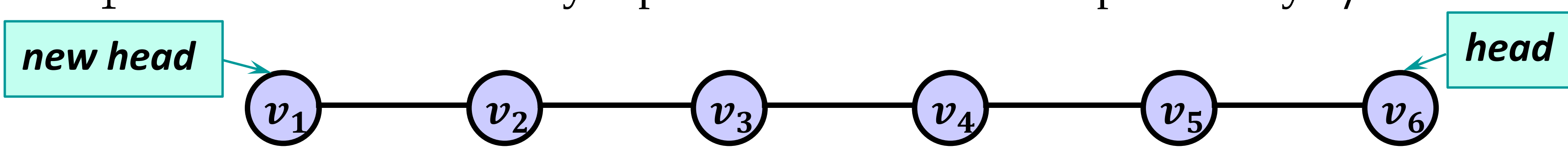
Lemma (Head is chosen uniformly at random from V at every iteration)

Let V_t be head vertex at iteration t.

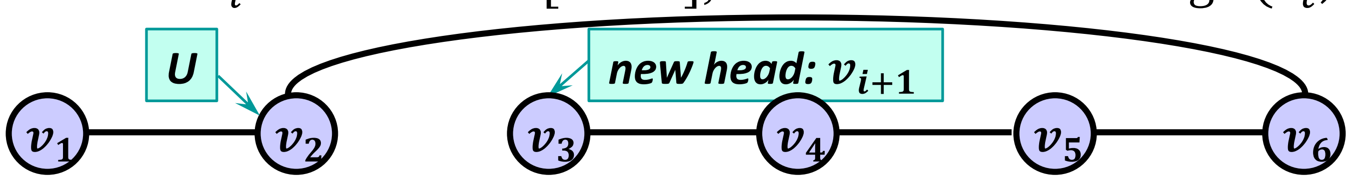
If algorithm has not terminated at iteration $t+1$, then
$\Pr[V_{t+1}=u \mid V_t=u_t,\dots,V_1=u_1]=1/n \quad \forall u,u_t,\dots,u_1$

Proof: Lemma holds for V_1. Now suppose the path P is $(v_1,\dots,v_k)$

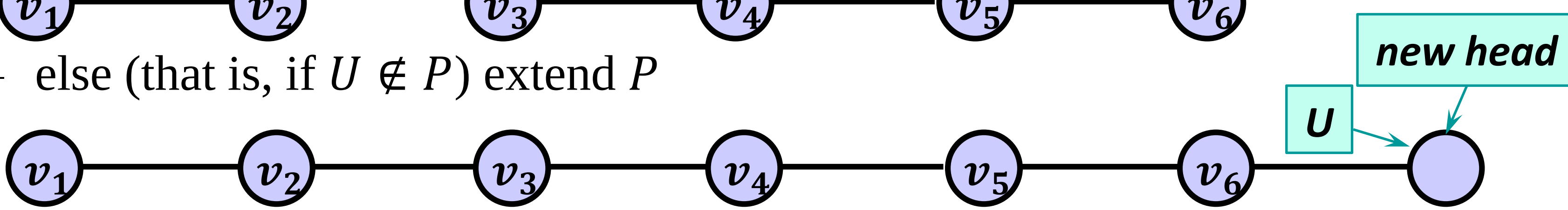
- v_1 becomes new head only if path P is reversed: with probability $1/n$



- With remaining probability, we pick a ``neighbor'' U of V_t **and**
  - if $U=v_i$ for some $i\in[k-1]$, then rotate P with edge (V_t,v_i)



  - else (that is, if $U\notin P$) extend P



Initial analysis (continued)

Lemma (Head is chosen uniformly at random from V at every iteration)

Let V_t be head vertex at iteration t.

If algorithm has not terminated at iteration $t+1$, then
$\Pr[V_{t+1} = u \mid V_t = u_t, \dots, V_1 = u_1] = 1/n \quad \forall u, u_t, \dots, u_1$

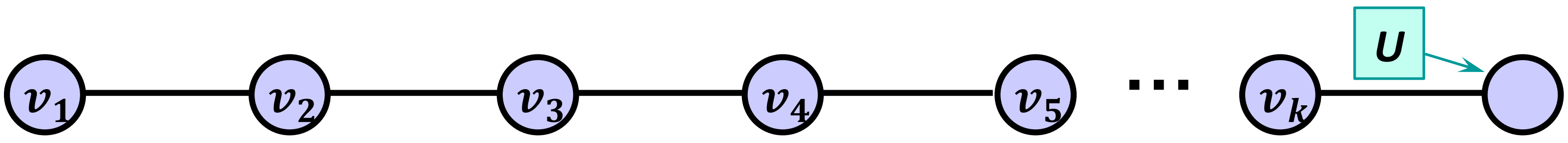
Proof: Remains to prove: every vertex in $V - \{v_k\}$ is chosen to be U w. p. $1/n$

- If $(v_k, u) \in UE(v_k)$, then $\Pr[U = u] = \frac{|UE(v_k)|}{n} \cdot \frac{1}{|UE(v_k)|} = \frac{1}{n}$

- If not, apply Principle of Deferred decisions:
$$\Pr[U = u] = \left(1 - \frac{1}{n} - \frac{|UE(v_k)|}{n}\right) \cdot \frac{1}{n - 1 - |UE(v_k)|} = \frac{1}{n}$$

Obtaining a Hamiltonian path

- What is the probability that we extend the path?



- What problem does it remind you of?

Analysis

Main Theorem (with simplifying assumption)

Suppose $q \geq \frac{20 \ln n}{n}$. Then our algorithm finds a Hamiltonian cycle in $O(n \log n)$ iterations with probability $1 - O\left(\frac{1}{n}\right)$.

Proof: For algorithm to fail one of the following events must occur

- $E_1 =$ no UUE list became empty, but not done after $3n \ln n$ iterations.
- $E_2 =$ at least one UUE list became empty during first $3n \ln n$ iterations.

  $2n \ln n$ iterations for getting a Hamiltonian path

  $n \ln n$ iterations for closing a Hamiltonian path
- Pr[any specific coupon (node) is not collected in $2n \ln n$ iterations] $=$
- Pr[don't close a Hamiltonian path in $n \ln n$ iterations]$=$

Analysis (continued)

Main Theorem (with simplifying assumption)

Suppose $q \geq \frac{20 \ln n}{n}$. Then our algorithm finds a Hamiltonian cycle in $O(n \log n)$ iterations with probability $1 - O\left(\frac{1}{n}\right)$.

Proof: For algorithm to fail one of the following events must occur

- $E_1 =$ no UUE list became empty, but not done after $3n \ln n$ iterations.
- $E_2 =$ at least one UUE list became empty during first $3n \ln n$ iterations.
  - $E_{2a} =$ at least $9 \ln n$ edges were removed from $UUE(v)$ for some $v \in V$ in the first $3n \ln n$ iterations.
  - $E_{2b} =$ initially, $|UUE(v)| < 9 \ln n$ for some $v \in V$

Analysis (continued)

Main Theorem (with simplifying assumption)

Suppose $q \geq \frac{20 \ln n}{n}$. Then our algorithm finds a Hamiltonian cycle in $O(n \log n)$ iterations with probability $1 - O\left(\frac{1}{n}\right)$.

Proof: $E_{2b} =$ initially, $|UUE(v)| < 9 \ln n$ for some $v \in V$

Removing the assumption

Main Theorem

Suppose $p \geq \frac{40 \ln n}{n}$. Then our algorithm *(with appropriately initialized UUE lists)* finds a Hamiltonian cycle in a graph chosen from $G_{n,p}$ in $O(n \log n)$ iterations with probability $1 - O\left(\frac{1}{n}\right)$.

Proof: Select q such that $p = 2q - q^2$. Let $G \leftarrow G_{n,p}$.

For each edge (u, v) of G:

- Put v in $UUE(u)$, but not u in $UUE(v)$
- vice versa
- Put v in $UUE(u)$ **and** u in $UUE(v)$

Randomize the order.