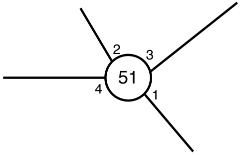
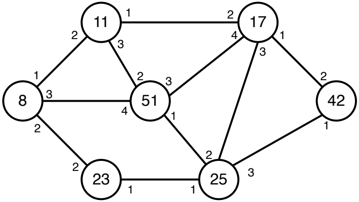
How to route a packet from 51 to 42?

Using Local Information Only . . .

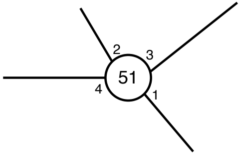


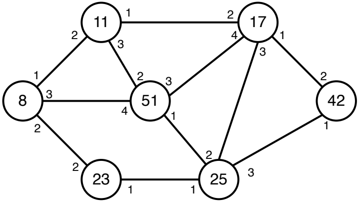


- The process identifier
- Port numbers of incident channels

How to route a packet from 51 to 42?

Using Local Information Only . . .



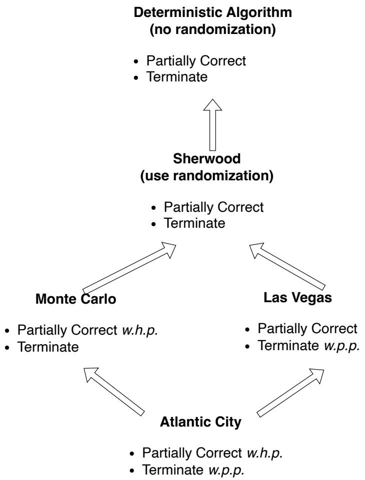


- The process identifier
- Port numbers of incident channels

In an arbitrary connected bidirectional network, without any further information:

only **randomization** can help!

Random Algorithms



w.p.p. = with (strictly) positive probability
w.h.p. = with high probability, *i.e.*, the probability depends on a parameter x such that the probability converges to 1 when x goes to the infinite (*w.h.p.* $\Rightarrow$ *w.p.p.*)

Remark: the Quicksort algorithm where the pivot is randomly chosen is a Sherwood algorithm.

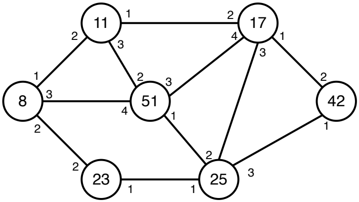
Random Local Algorithm (Las Vegas Algorithm)

Given a packet p with destination label d at node u.

if $d = u$ **then**
    `deliver` p
else
    pick $i \in \{1, \ldots, \delta_u\}$ according to P_u
    send p via port number i
end if

P_u is a probability distribution:

- $\forall i \in \{1, \ldots, \delta_u\}$, $P_u(i)$ gives the probability of picking i
- $P_u : \{1, \ldots, \delta_u\} \to [0, 1]$ such that $\sum_{i \in \{1,\ldots,\delta_u\}} P_u(i) = 1$



Random Local Algorithm (Las Vegas Algorithm)

Given a packet p with destination label d at node u.

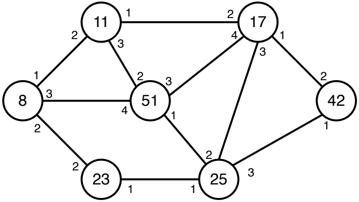
if $d = u$ **then**
  `deliver` p
else
  pick $i \in \{1, \ldots, \delta_u\}$ according to P_u
  send p via port number i
end if

P_u is a probability distribution:

- $\forall i \in \{1, \ldots, \delta_u\}$, $P_u(i)$ gives the probability of picking i
- $P_u : \{1, \ldots, \delta_u\} \to [0, 1]$ such that $\sum_{i \in \{1,\ldots,\delta_u\}} P_u(i) = 1$
- For example, P_u may be a uniform distribution:

$$\forall i \in \{1, \ldots, \delta_u\}, P_u(i) = \frac{1}{\delta_u}$$

E.g., $P_8(1) = P_8(2) = P_8(3) = \frac{1}{3}$



Random Local Algorithm (Las Vegas Algorithm)

Given a packet p with destination label d at node u.

if $d = u$ **then**
    `deliver` p
else
    pick $i \in \{1, \ldots, \delta_u\}$ according to P_u
    send p via port number i
end if

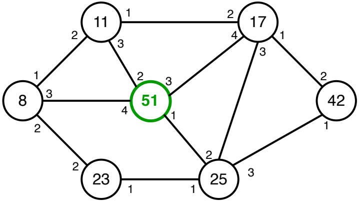
P_u is a probability distribution:

- $\forall i \in \{1, \ldots, \delta_u\}$, $P_u(i)$ gives the probability of picking i
- $P_u : \{1, \ldots, \delta_u\} \rightarrow [0, 1]$ such that $\sum_{i \in \{1,\ldots,\delta_u\}} P_u(i) = 1$
- For example, P_u may be a uniform distribution:

$$\forall i \in \{1, \ldots, \delta_u\}, P_u(i) = \frac{1}{\delta_u}$$

E.g., $P_8(1) = P_8(2) = P_8(3) = \frac{1}{3}$

Routing from 51 to 42
(*e.g.*, using uniform distribution)



Routing path: 51

Random Local Algorithm (Las Vegas Algorithm)

Given a packet p with destination label d at node u.

if $d = u$ **then**
    `deliver` p
else
    pick $i \in \{1, \dots, \delta_u\}$ according to P_u
    send p via port number i
end if

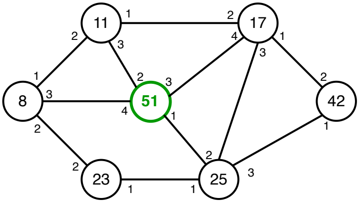
P_u is a probability distribution:

- $\forall i \in \{1, \dots, \delta_u\}$, $P_u(i)$ gives the probability of picking i
- $P_u : \{1, \dots, \delta_u\} \to [0, 1]$ such that $\sum_{i \in \{1,\dots,\delta_u\}} P_u(i) = 1$
- For example, P_u may be a uniform distribution:

$$\forall i \in \{1, \dots, \delta_u\}, P_u(i) = \frac{1}{\delta_u}$$

E.g., $P_8(1) = P_8(2) = P_8(3) = \frac{1}{3}$

Routing from 51 to 42
(*e.g.*, using uniform distribution)



Routing path: 51
Pick 3

Random Local Algorithm (Las Vegas Algorithm)

Given a packet p with destination label d at node u.

if $d = u$ **then**
    `deliver` p
else
    pick $i \in \{1, \ldots, \delta_u\}$ according to P_u
    send p via port number i
end if

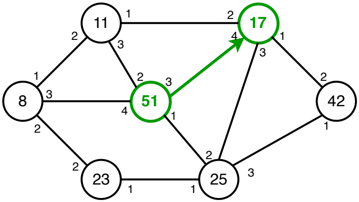
P_u is a probability distribution:

- $\forall i \in \{1, \ldots, \delta_u\}$, $P_u(i)$ gives the probability of picking i
- $P_u : \{1, \ldots, \delta_u\} \to [0, 1]$ such that $\sum_{i \in \{1,\ldots,\delta_u\}} P_u(i) = 1$
- For example, P_u may be a uniform distribution:

$$\forall i \in \{1, \ldots, \delta_u\}, P_u(i) = \frac{1}{\delta_u}$$

E.g., $P_8(1) = P_8(2) = P_8(3) = \frac{1}{3}$

Routing from 51 to 42
(*e.g.*, using uniform distribution)



Routing path: 51,17

Random Local Algorithm (Las Vegas Algorithm)

Given a packet p with destination label d at node u.

if $d = u$ **then**
    `deliver` p
else
    pick $i \in \{1, \ldots, \delta_u\}$ according to P_u
    send p via port number i
end if

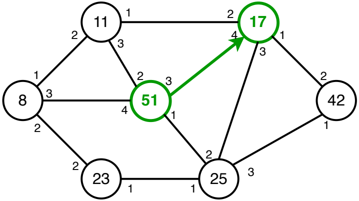
P_u is a probability distribution:

- $\forall i \in \{1, \ldots, \delta_u\}$, $P_u(i)$ gives the probability of picking i
- $P_u : \{1, \ldots, \delta_u\} \rightarrow [0, 1]$ such that $\sum_{i \in \{1,\ldots,\delta_u\}} P_u(i) = 1$
- For example, P_u may be a uniform distribution:

$$\forall i \in \{1, \ldots, \delta_u\}, P_u(i) = \frac{1}{\delta_u}$$

E.g., $P_8(1) = P_8(2) = P_8(3) = \frac{1}{3}$

Routing from 51 to 42
(*e.g.*, using uniform distribution)



Routing path: 51,17
Pick 2

Random Local Algorithm (Las Vegas Algorithm)

Given a packet p with destination label d at node u.

if $d = u$ **then**
    `deliver` p
else
    pick $i \in \{1, \ldots, \delta_u\}$ according to P_u
    send p via port number i
end if

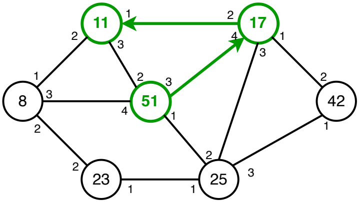
P_u is a probability distribution:

- $\forall i \in \{1, \ldots, \delta_u\}$, $P_u(i)$ gives the probability of picking i
- $P_u : \{1, \ldots, \delta_u\} \to [0, 1]$ such that $\sum_{i \in \{1,\ldots,\delta_u\}} P_u(i) = 1$
- For example, P_u may be a uniform distribution:

$$\forall i \in \{1, \ldots, \delta_u\}, P_u(i) = \frac{1}{\delta_u}$$

E.g., $P_8(1) = P_8(2) = P_8(3) = \frac{1}{3}$

Routing from 51 to 42
(*e.g.*, using uniform distribution)



Routing path: 51,17,11

Random Local Algorithm (Las Vegas Algorithm)

Given a packet p with destination label d at node u.

if $d = u$ **then**
    `deliver` p
else
    pick $i \in \{1, \ldots, \delta_u\}$ according to P_u
    send p via port number i
end if

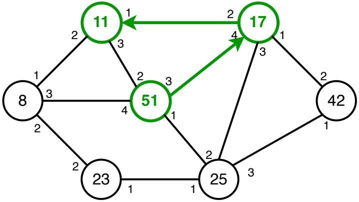
P_u is a probability distribution:

- $\forall i \in \{1, \ldots, \delta_u\}$, $P_u(i)$ gives the probability of picking i
- $P_u : \{1, \ldots, \delta_u\} \to [0, 1]$ such that $\sum_{i \in \{1, \ldots, \delta_u\}} P_u(i) = 1$
- For example, P_u may be a uniform distribution:

$$\forall i \in \{1, \ldots, \delta_u\}, P_u(i) = \frac{1}{\delta_u}$$

E.g., $P_8(1) = P_8(2) = P_8(3) = \frac{1}{3}$

Routing from 51 to 42
(*e.g.*, using uniform distribution)



Routing path: 51,17,11
Pick 3

Random Local Algorithm (Las Vegas Algorithm)

Given a packet p with destination label d at node u.

if $d = u$ **then**
    `deliver` p
else
    pick $i \in \{1, \ldots, \delta_u\}$ according to P_u
    send p via port number i
end if

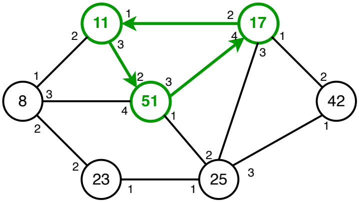
P_u is a probability distribution:

- $\forall i \in \{1, \ldots, \delta_u\}$, $P_u(i)$ gives the probability of picking i
- $P_u : \{1, \ldots, \delta_u\} \to [0, 1]$ such that $\sum_{i \in \{1,\ldots,\delta_u\}} P_u(i) = 1$
- For example, P_u may be a uniform distribution:

$$\forall i \in \{1, \ldots, \delta_u\}, P_u(i) = \frac{1}{\delta_u}$$

E.g., $P_8(1) = P_8(2) = P_8(3) = \frac{1}{3}$

Routing from 51 to 42
(*e.g.*, using uniform distribution)



Routing path: 51,17,11,51

Random Local Algorithm (Las Vegas Algorithm)

Given a packet p with destination label d at node u.

if $d = u$ **then**
  `deliver` p
else
  pick $i \in \{1, \dots, \delta_u\}$ according to P_u
  send p via port number i
end if

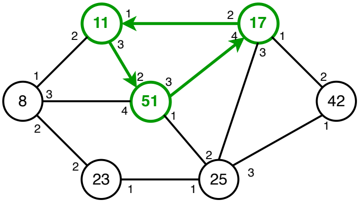
P_u is a probability distribution:

- $\forall i \in \{1, \dots, \delta_u\}$, $P_u(i)$ gives the probability of picking i
- $P_u : \{1, \dots, \delta_u\} \to [0, 1]$ such that $\sum_{i \in \{1,\dots,\delta_u\}} P_u(i) = 1$
- For example, P_u may be a uniform distribution:

$$\forall i \in \{1, \dots, \delta_u\}, P_u(i) = \frac{1}{\delta_u}$$

E.g., $P_8(1) = P_8(2) = P_8(3) = \frac{1}{3}$

Routing from 51 to 42
(*e.g.*, using uniform distribution)



Routing path: 51,17,11,51
Pick 1

Random Local Algorithm (Las Vegas Algorithm)

Given a packet p with destination label d at node u.

if $d = u$ **then**
  `deliver` p
else
  pick $i \in \{1, \ldots, \delta_u\}$ according to P_u
  send p via port number i
end if

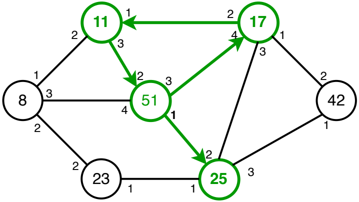
P_u is a probability distribution:

- $\forall i \in \{1, \ldots, \delta_u\}$, $P_u(i)$ gives the probability of picking i
- $P_u : \{1, \ldots, \delta_u\} \to [0, 1]$ such that $\sum_{i \in \{1,\ldots,\delta_u\}} P_u(i) = 1$
- For example, P_u may be a uniform distribution:

$$\forall i \in \{1, \ldots, \delta_u\}, P_u(i) = \frac{1}{\delta_u}$$

E.g., $P_8(1) = P_8(2) = P_8(3) = \frac{1}{3}$

Routing from 51 to 42
(*e.g.*, using uniform distribution)



Routing path: 51,17,11,51,25

Random Local Algorithm (Las Vegas Algorithm)

Given a packet p with destination label d at node u.

if $d = u$ **then**
  `deliver` p
else
  pick $i \in \{1, \dots, \delta_u\}$ according to P_u
  send p via port number i
end if

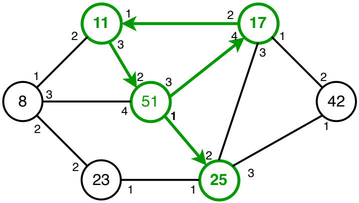
P_u is a probability distribution:

- $\forall i \in \{1, \dots, \delta_u\}$, $P_u(i)$ gives the probability of picking i
- $P_u : \{1, \dots, \delta_u\} \to [0, 1]$ such that $\sum_{i \in \{1,\dots,\delta_u\}} P_u(i) = 1$
- For example, P_u may be a uniform distribution:

$$\forall i \in \{1, \dots, \delta_u\}, P_u(i) = \frac{1}{\delta_u}$$

E.g., $P_8(1) = P_8(2) = P_8(3) = \frac{1}{3}$

Routing from 51 to 42
(*e.g.*, using uniform distribution)



Routing path: 51,17,11,51,25
Pick 3

Random Local Algorithm (Las Vegas Algorithm)

Given a packet p with destination label d at node u.

if $d = u$ **then**
    `deliver` p
else
    pick $i \in \{1, \ldots, \delta_u\}$ according to P_u
    send p via port number i
end if

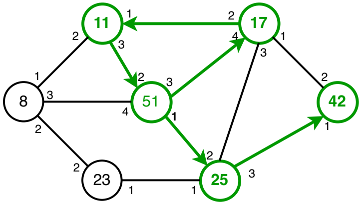
P_u is a probability distribution:

- $\forall i \in \{1, \ldots, \delta_u\}$, $P_u(i)$ gives the probability of picking i
- $P_u : \{1, \ldots, \delta_u\} \to [0, 1]$ such that $\sum_{i \in \{1,\ldots,\delta_u\}} P_u(i) = 1$
- For example, P_u may be a uniform distribution:

$$\forall i \in \{1, \ldots, \delta_u\}, P_u(i) = \frac{1}{\delta_u}$$

E.g., $P_8(1) = P_8(2) = P_8(3) = \frac{1}{3}$

Routing from 51 to 42
(*e.g.*, using uniform distribution)



Routing path: 51,17,11,51,25,42

Random Local Algorithm (Las Vegas Algorithm)

Given a packet p with destination label d at node u.

if $d = u$ **then**
    `deliver` p
else
    pick $i \in \{1, \ldots, \delta_u\}$ according to P_u
    send p via port number i
end if

P_u is a probability distribution:

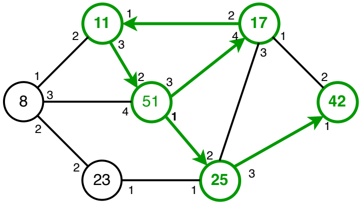
- $\forall i \in \{1, \ldots, \delta_u\}$, $P_u(i)$ gives the probability of picking i
- $P_u : \{1, \ldots, \delta_u\} \rightarrow [0, 1]$ such that $\sum_{i \in \{1,\ldots,\delta_u\}} P_u(i) = 1$
- For example, P_u may be a uniform distribution:

$$\forall i \in \{1, \ldots, \delta_u\}, P_u(i) = \frac{1}{\delta_u}$$

  E.g., $P_8(1) = P_8(2) = P_8(3) = \frac{1}{3}$

- Formally, 51,17,11,51,25,42 = prefix of a **(standard) random walk**

Routing from 51 to 42
(*e.g.*, using uniform distribution)



Routing path: 51,17,11,51,25,42

Routing using Local Information: Random Walks (also called *Drunkard's Walks*)

Réseaux & Communication

Alain Cournier Stéphane Devismes

Université de Picardie Jules Verne

January 13, 2026

Roadmap

Roadmap

Context

Let $G = (V, E)$ be a finite, simple, and connected graph with order $n = |V| \geq 2$ and size $m = |E|$

$\forall u \in V$, let $N(u) = \{v \mid \{u, v\} \in E\}$ be the neighborhood of u.

$N[u] = N(u) \cup \{u\}$ is the closed neighborhood of u and $\delta_u = |N(u)|$ is the degree of u

Transition Probability Matrix

We will consider **pure random walks** where the probability distribution at each node is constant[1]

The probability distributions are stored in a **transition probability matrix P for G**:

$$P = (p(u,v))_{u,v \in V} \in [0,1]^{V \times V}$$

- $p(u,v)$ is the probability of moving from u to v

[1] In case the probability distributions evolve along the time, a random walk is biased, *e.g.*, the simulated annealing is a biased random walk in a state space

[2] $u \in N[u]$: to be more general, we allow a walk to stay for sometime at some nodes.

Transition Probability Matrix

We will consider **pure random walks** where the probability distribution at each node is constant[1]

The probability distributions are stored in a **transition probability matrix P for G**:

$$P = (p(u, v))_{u,v \in V} \in [0, 1]^{V \times V}$$

- $p(u, v)$ is the probability of moving from u to v
- $\forall u \in V, \sum_{v \in N[u]} p(u, v) = 1$ and $v \notin N[u] \Rightarrow p(u, v) = 0$, indeed a walk is a graph traversal[2]

[1] In case the probability distributions evolve along the time, a random walk is biased, *e.g.*, the simulated annealing is a biased random walk in a state space

[2] $u \in N[u]$: to be more general, we allow a walk to stay for sometime at some nodes.

Transition Probability Matrix

We will consider **pure random walks** where the probability distribution at each node is constant[1]

The probability distributions are stored in a **transition probability matrix P for G**:

$$P = (p(u,v))_{u,v \in V} \in [0,1]^{V \times V}$$

- $p(u,v)$ is the probability of moving from u to v
- $\forall u \in V, \sum_{v \in N[u]} p(u,v) = 1$ and $v \notin N[u] \Rightarrow p(u,v) = 0$, indeed a walk is a graph traversal[2]

Let $\mathcal{P}(G)$ be the set of all transition probability matrix for G

[1] In case the probability distributions evolve along the time, a random walk is biased, *e.g.*, the simulated annealing is a biased random walk in a state space

[2] $u \in N[u]$: to be more general, we allow a walk to stay for sometime at some nodes.

Example: Uniform Transition Probability Matrix

$\forall u, v \in V$:

- $v \notin N(u) \Rightarrow p(u, v) = 0$
- $v \in N(u) \Rightarrow p(u, v) = \frac{1}{\delta_u}$

Remark: $\forall u, p(u, u) = 0$, so no wait!

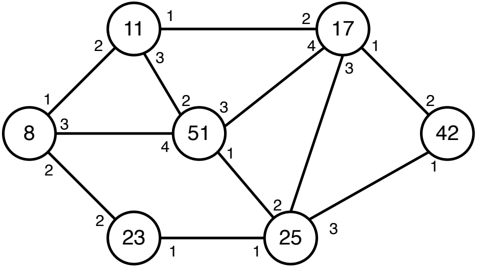
A **random walk** $\omega = (\omega_0, \omega_1, \ldots)$ on G starting at vertex u under $P \in \mathcal{P}(G)$ is an infinite sequence of random variables ω_i whose domain is V such that

- $\omega_0 = u$ with probability 1, and
- $\forall i \in \mathbb{N}$, the (conditional) probability that $\omega_{i+1} = w$, provided that $\omega_i = v$, is $p(v, w)$

Random Walk

Example

The infinite random sequence $(51, 17, 11)^{\omega}$ is a random walk on the graph given below under a uniform transition probability matrix.



Remark: a random walk on a graph under a uniform transition probability matrix is called a **standard random walk**

Roadmap

Correctness of the Random-walk-based Algorithm

Correctness = Partial Correctness + Termination

Partial Correctness: Trivial! The algorithm stops only if the packet has reached its destination.

Termination *almost sure*: Termination with probability one (Las Vegas Algorithm).

I.e., there are infinite executions where the destination is never reached (*e.g.*, $(51, 17, 11)^{\omega}$), yet the overall probability of the occurrence of such executions is 0.

The almost sure termination is due to the fact that any vertex has probability 1 of occurring in any standard random walk on G.

Characterization

Let $S = (V_S, E_S)$ be the digraph such that

- $V_S = V$ and
- $E_S = \{(u, v) \in V^2 \mid p(u, v) > 0\}$

Theorem 1

For every $u, v \in V$, v has probability 1 of occurring in any random walk on G starting at vertex u under $P \in \mathcal{P}(G)$

if and only if

S is strongly connected.

Corollary 2

v has probability 1 of occurring in any standard random walk on G.

Proof of Theorem 1

Necessary Condition

Assume S is not strongly connected and let u, v be two nodes of S such that v is not reachable from u. $(u \neq v)$

Proof of Theorem 1
Necessary Condition

Assume S is not strongly connected and let u, v be two nodes of S such that v is not reachable from u. $(u \neq v)$

Assume, by the contradiction, that v occurs in a random walk on G starting at vertex u under P.

Let SP by the smallest prefix of the walk starting from u and ending with v.

Proof of Theorem 1
Necessary Condition

Assume S is not strongly connected and let u, v be two nodes of S such that v is not reachable from u. ($u \neq v$)

Assume, by the contradiction, that v occurs in a random walk on G starting at vertex u under P.

Let SP by the smallest prefix of the walk starting from u and ending with v.

Every two consecutive nodes w and w' in SP satisfies $p(w, w') > 0$, which in turn implies $(w, w') \in E_S$.

SP is a (directed) path from u to v in S: v is reachable from u in S, a contradiction. □

Proof of Theorem 1
Sufficient Condition

Let ω be any random walk on G starting at vertex u under P. Let $p_{\min} = \min\{p(w, w') \mid (w, w') \in E_S\}$.

Let ω be any random walk on G starting at vertex u under P. Let $p_{\min} = \min\{p(w, w') \mid (w, w') \in E_S\}$.

$p_{\min} > 0$, by definition of S.

Proof of Theorem 1
Sufficient Condition

Let ω be any random walk on G starting at vertex u under P. Let $p_{\min} = \min\{p(w, w') \mid (w, w') \in E_S\}$.

$p_{\min} > 0$, by definition of S.

Since S is strongly connected and $n > 1$, its diameter $\mathcal{D}$ satisfies $1 \leq \mathcal{D} < n$.

Proof of Theorem 1

Sufficient Condition

Let ω be any random walk on G starting at vertex u under P. Let $p_{\min} = \min\{p(w, w') \mid (w, w') \in E_S\}$.

$p_{\min} > 0$, by definition of S.

Since S is strongly connected and $n > 1$, its diameter $\mathcal{D}$ satisfies $1 \leq \mathcal{D} < n$.

In every suffix s of ω, the probability that v occurs among the first $\mathcal{D}$ values of s is at least $0 < (p_{\min})^{\mathcal{D}} \leq 1$. Indeed, there is an elementary path of length at most $\mathcal{D}$ from any vertex to v in S.

Proof of Theorem 1

Sufficient Condition

Let ω be any random walk on G starting at vertex u under P. Let $p_{\min} = \min\{p(w, w') \mid (w, w') \in E_S\}$.

$p_{\min} > 0$, by definition of S.

Since S is strongly connected and $n > 1$, its diameter $\mathcal{D}$ satisfies $1 \leq \mathcal{D} < n$.

In every suffix s of ω, the probability that v occurs among the first $\mathcal{D}$ values of s is at least $0 < (p_{\min})^{\mathcal{D}} \leq 1$. Indeed, there is an elementary path of length at most $\mathcal{D}$ from any vertex to v in S.

So, the probability that v does not occur among the first $k \times \mathcal{D}$ values of ω is at most $(1 - (p_{\min})^{\mathcal{D}})^k$.

Proof of Theorem 1

Sufficient Condition

Let ω be any random walk on G starting at vertex u under P. Let $p_{\min} = \min\{p(w, w') \mid (w, w') \in E_S\}$.

$p_{\min} > 0$, by definition of S.

Since S is strongly connected and $n > 1$, its diameter $\mathcal{D}$ satisfies $1 \leq \mathcal{D} < n$.

In every suffix s of ω, the probability that v occurs among the first $\mathcal{D}$ values of s is at least $0 < (p_{\min})^{\mathcal{D}} \leq 1$. Indeed, there is an elementary path of length at most $\mathcal{D}$ from any vertex to v in S.

So, the probability that v does not occur among the first $k \times \mathcal{D}$ values of ω is at most $(1 - (p_{\min})^{\mathcal{D}})^k$.

Now, $\lim_{k\to\infty}(1 - (p_{\min})^{\mathcal{D}})^k = 0$ since $0 \leq 1 - (p_{\min})^{\mathcal{D}} < 1$.

Hence, v has probability 1 of occurring in ω. □

Roadmap

Roadmap

Relevant Quantities

Hitting Time: informally, the hitting time is the expected time to move to a node v in a random walk

(from a routing point of view, it is the expected length of the routing path)

Cover Time: informally, the cover time is the expected time to visit all nodes in a random walk

Hitting Time

Given a random walk $\omega = (\omega_0, \omega_1, \ldots)$ starting at vertex $u \in V$, the hitting time $H_G(P; u, v)$ from u to v under P is:

$$H_G(P; u, v) = E_P[\inf\{i \geq 1 \mid \omega_i = v\}]$$

i.e., the expectation of the smallest time where ω reaches v after leaving u.

Hitting Time

Given a random walk $\omega = (\omega_0, \omega_1, \ldots)$ starting at vertex $u \in V$, the hitting time $H_G(P; u, v)$ from u to v under P is:

$$H_G(P; u, v) = E_P[\inf\{i \geq 1 \mid \omega_i = v\}]$$

i.e., the expectation of the smallest time where ω reaches v after leaving u.

Remark: $H_G(P; u, u)$ is the expectation of the smallest time for ω to leave and then return to u!

The hitting time $H_G(P)$ of G under P is:

$$H_G(P) = \max_{u,v \in V} H_G(P; u, v)$$

Hitting Time

Given a random walk $\omega = (\omega_0, \omega_1, \ldots)$ starting at vertex $u \in V$, the hitting time $H_G(P; u, v)$ from u to v under P is:

$$H_G(P; u, v) = E_P[\inf\{i \geq 1 \mid \omega_i = v\}]$$

i.e., the expectation of the smallest time where ω reaches v after leaving u.

Remark: $H_G(P; u, u)$ is the expectation of the smallest time for ω to leave and then return to u!

The hitting time $H_G(P)$ of G under P is:

$$H_G(P) = \max_{u,v \in V} H_G(P; u, v)$$

In the following, we will denote by $\mathcal{H}_G(u, v)$ the hitting time from u to v in a standard random walk on G.

Cover Time

Given a random walk $\omega = (\omega_0, \omega_1, \ldots)$ starting at vertex $u \in V$, the cover time $C_G(P; u)$ from u under P is:

$$C_G(P; u) = E_P[\inf\{i \geq 1 \mid \{\omega_0, \ldots, \omega_i\} = V\}]$$

i.e., the expectation of the smallest time for ω to visit all vertices starting from u.

The cover time $C_G(P)$ of G under P is:

$$C_G(P) = \max_{u \in V} C_G(P; u)$$

Roadmap

Definition

A **Markov chain** or Markov process is a stochastic model where the probability of future (next) state only depends on the most recent (current) state.

This memoryless property of a stochastic process is called Markov property.

From a probability perspective, the Markov property implies that the conditional probability distribution of the future state (conditioned on both past and current states) only depends on the current state.

Definition

A **Markov chain** or Markov process is a stochastic model where the probability of future (next) state only depends on the most recent (current) state.

This memoryless property of a stochastic process is called Markov property.

From a probability perspective, the Markov property implies that the conditional probability distribution of the future state (conditioned on both past and current states) only depends on the current state.

A Markov chain is usually represented as a weighted digraph where nodes are states and arcs are possible transitions weighted with their positive probability of occurrence.

Definition

A **Markov chain** or Markov process is a stochastic model where the probability of future (next) state only depends on the most recent (current) state.

This memoryless property of a stochastic process is called Markov property.

From a probability perspective, the Markov property implies that the conditional probability distribution of the future state (conditioned on both past and current states) only depends on the current state.

A Markov chain is usually represented as a weighted digraph where nodes are states and arcs are possible transitions weighted with their positive probability of occurrence.

A Markov chain in which every state can be reached from every other state is called an **irreducible Markov chain**.

Example

A random walk on a graph as the Markov property: it can be modeled by a finite Markov chain.

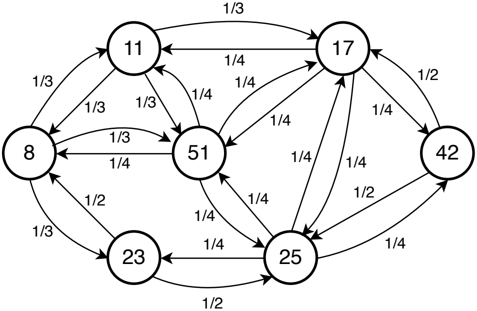
For example, the weighted digraph S in Theorem 1 is a Markov Chain.

Example

A random walk on a graph as the Markov property: it can be modeled by a finite Markov chain.

For example, the weighted digraph S in Theorem 1 is a Markov Chain.

Below, we give the Markov chain corresponding to the standard random walk on our sample graph.



Stationary Distribution of a Markov Chain

The **stationary distribution** $\pi = (\pi_i)_{i \in E}$ of a Markov chain gives the fraction of the time spent in each state i of the state space E of this Markov chain, asymptotically.

Let $S_n(i)$ the time spent in state i after the first n steps.

$$\pi_i = \lim_{n \to \infty} \frac{S_n(i)}{n}$$

Application of the theorem of Perron-Frobenius [5, 8]

Corollary 3

*Any **finite irreducible Markov chain** has a stationary distribution $\pi = (\pi_i)_{i \in E}$ that is the **unique solution** of:*

1. $\sum_{i \in E} \pi_i = 1$, *and*
2. $\forall j \in E$, $\sum_{i \in E} \pi_i p(i,j) = \pi_j$

where $p(i,j)$ are the transition probabilities of the Markov chain.

Intuition:

Application of the theorem of Perron-Frobenius [5, 8]

Corollary 3

Any ***finite irreducible Markov chain*** *has a stationary distribution $\pi = (\pi_i)_{i \in E}$ that is the* ***unique solution*** *of:*

1. $\sum_{i \in E} \pi_i = 1$, *and*
2. $\forall j \in E$, $\sum_{i \in E} \pi_i p(i,j) = \pi_j$

where $p(i,j)$ are the transition probabilities of the Markov chain.

Intuition:

1. In a distribution, the sum of probabilities is equal to 1

Application of the theorem of Perron-Frobenius [5, 8]

Corollary 3

Any ***finite irreducible Markov chain*** *has a stationary distribution* $\pi = (\pi_i)_{i \in E}$ *that is the* ***unique solution*** *of:*

1. $\sum_{i \in E} \pi_i = 1$, *and*
2. $\forall j \in E, \sum_{i \in E} \pi_i p(i,j) = \pi_j$

where $p(i,j)$ *are the transition probabilities of the Markov chain.*

Intuition:

1. In a distribution, the sum of probabilities is equal to 1
2. From i, j is reached in one step with probability $p(i,j)$: it is the fraction of time j is reached from i provided that the walk is in i

Application of the theorem of Perron-Frobenius [5, 8]

Corollary 3

Any ***finite irreducible Markov chain*** *has a stationary distribution* $\pi = (\pi_i)_{i \in E}$ *that is the* ***unique solution*** *of:*

1. $\sum_{i \in E} \pi_i = 1$, *and*
2. $\forall j \in E, \sum_{i \in E} \pi_i p(i,j) = \pi_j$

where $p(i,j)$ *are the transition probabilities of the Markov chain.*

Intuition:

1. In a distribution, the sum of probabilities is equal to 1
2. From i, j is reached in one step with probability $p(i,j)$: it is the fraction of time j is reached from i provided that the walk is in i

   π_i is the fraction of the time spent in i

   So, $\pi_i p(i,j)$ gives the fraction of time j is reached from i

Application of the theorem of Perron-Frobenius [5, 8]

Corollary 3

*Any **finite irreducible Markov chain** has a stationary distribution $\pi = (\pi_i)_{i \in E}$ that is the **unique solution** of:*

1. $\sum_{i \in E} \pi_i = 1$, *and*
2. $\forall j \in E$, $\sum_{i \in E} \pi_i p(i,j) = \pi_j$

where $p(i,j)$ are the transition probabilities of the Markov chain.

Intuition:

1. In a distribution, the sum of probabilities is equal to 1
2. From i, j is reached in one step with probability $p(i,j)$: it is the fraction of time j is reached from i provided that the walk is in i

   π_i is the fraction of the time spent in i

   So, $\pi_i p(i,j)$ gives the fraction of time j is reached from i

   Now, the fraction of the time spent in each state j, π_j, is the fraction of time j is reached from all states of E

Fundamental Result
from [6]

Lemma 4

$\forall u \in V$, $\mathcal{H}_G(u, u) = \frac{2m}{\delta_u}$

Intuition:

Fundamental Result

from [6]

Lemma 4

$\forall u \in V$, $\mathcal{H}_G(u,u) = \frac{2m}{\delta_u}$

Intuition:

- Since G has m edges, the Markov chain associated to the standard random walk on G has $2m$ arcs

Fundamental Result

from [6]

Lemma 4

$\forall u \in V$, $\mathcal{H}_G(u, u) = \frac{2m}{\delta_u}$

Intuition:

- Since G has m edges, the Markov chain associated to the standard random walk on G has $2m$ arcs
- Since the random walk is standard, the traversing of any arc is asymptotically equiprobable, *i.e.*, the stationary probability of any arc is $\frac{1}{2m}$

Fundamental Result

from [6]

Lemma 4

$\forall u \in V$, $\mathcal{H}_G(u,u) = \frac{2m}{\delta_u}$

Intuition:

- Since G has m edges, the Markov chain associated to the standard random walk on G has $2m$ arcs
- Since the random walk is standard, the traversing of any arc is asymptotically equiprobable, *i.e.*, the stationary probability of any arc is $\frac{1}{2m}$
- The stationary probability of a node u, π_u, is the sum of the stationary probability of its incoming arcs

Fundamental Result

from [6]

Lemma 4

$\forall u \in V$, $\mathcal{H}_G(u, u) = \frac{2m}{\delta_u}$

Intuition:

- Since G has m edges, the Markov chain associated to the standard random walk on G has $2m$ arcs
- Since the random walk is standard, the traversing of any arc is asymptotically equiprobable, *i.e.*, the stationary probability of any arc is $\frac{1}{2m}$
- The stationary probability of a node u, π_u, is the sum of the stationary probability of its incoming arcs
- Since a node u has δ_u incoming arcs in the Markov chain, we have $\pi_u = \frac{\delta_u}{2m}$

Fundamental Result

from [6]

Lemma 4

$\forall u \in V$, $\mathcal{H}_G(u,u) = \frac{2m}{\delta_u}$

Intuition:

- Since G has m edges, the Markov chain associated to the standard random walk on G has $2m$ arcs
- Since the random walk is standard, the traversing of any arc is asymptotically equiprobable, *i.e.*, the stationary probability of any arc is $\frac{1}{2m}$
- The stationary probability of a node u, π_u, is the sum of the stationary probability of its incoming arcs
- Since a node u has δ_u incoming arcs in the Markov chain, we have $\pi_u = \frac{\delta_u}{2m}$
- Since π_u is the fraction of the time spent in vertex u during the walk, we have $\mathcal{H}_G(u,u) = \frac{1}{\pi_u}$, *i.e.*, $\mathcal{H}_G(u,u) = \frac{2m}{\delta_u}$

Proof of Lemma 4 (1/2)

Consider an arbitrary standard random walk ω on G. Let $\pi = (\pi_v)_{v \in V}$ be the stationary distribution of Markov chain that models ω.

Proof of Lemma 4 (1/2)

Consider an arbitrary standard random walk ω on G. Let $\pi = (\pi_v)_{v \in V}$ be the stationary distribution of Markov chain that models ω.

$\mathcal{H}_G(u, u) = \frac{1}{\pi_u}$.

Proof of Lemma 4 (1/2)

Consider an arbitrary standard random walk ω on G. Let $\pi = (\pi_v)_{v \in V}$ be the stationary distribution of Markov chain that models ω.

$\mathcal{H}_G(u, u) = \frac{1}{\pi_u}$.

Thus, the lemma holds if $\pi_u = \frac{\delta_u}{2m}$.

Proof of Lemma 4 (1/2)

Consider an arbitrary standard random walk ω on G. Let $\pi = (\pi_v)_{v \in V}$ be the stationary distribution of Markov chain that models ω.

$\mathcal{H}_G(u, u) = \frac{1}{\pi_u}$.

Thus, the lemma holds if $\pi_u = \frac{\delta_u}{2m}$.

V is a finite set. G is connected and at each node, the probability of traversing each incident edge is strictly positive. So, the Markov chain modeling ω is finite and irreducible. Hence, Corollary 3 applies: $\forall u \in V$, $\pi_u := \frac{\delta_u}{2m}$ should be the solution of

- $\sum_{u \in V} \pi_u = 1$, and
- $\forall v \in V$, $\sum_{u \in V} \pi_u p(u, v) = \pi_v$

where $p(u, v) = \frac{1}{\delta_u}$ if u and v are neighbors, 0 otherwise.

Proof of Lemma 4 (1/2)

Consider an arbitrary standard random walk ω on G. Let $\pi = (\pi_v)_{v \in V}$ be the stationary distribution of Markov chain that models ω.

$\mathcal{H}_G(u,u) = \frac{1}{\pi_u}$.

Thus, the lemma holds if $\pi_u = \frac{\delta_u}{2m}$.

V is a finite set. G is connected and at each node, the probability of traversing each incident edge is strictly positive. So, the Markov chain modeling ω is finite and irreducible. Hence, Corollary 3 applies: $\forall u \in V$, $\pi_u := \frac{\delta_u}{2m}$ should be the solution of

- $\sum_{u \in V} \pi_u = 1$, and
- $\forall v \in V$, $\sum_{u \in V} \pi_u p(u,v) = \pi_v$

where $p(u,v) = \frac{1}{\delta_u}$ if u and v are neighbors, 0 otherwise.

$$\sum_{u \in V} \pi_u = \sum_{u \in V} \frac{\delta_u}{2m}$$

Proof of Lemma 4 (1/2)

Consider an arbitrary standard random walk ω on G. Let $\pi = (\pi_v)_{v \in V}$ be the stationary distribution of Markov chain that models ω.

$\mathcal{H}_G(u, u) = \frac{1}{\pi_u}$.

Thus, the lemma holds if $\pi_u = \frac{\delta_u}{2m}$.

V is a finite set. G is connected and at each node, the probability of traversing each incident edge is strictly positive. So, the Markov chain modeling ω is finite and irreducible. Hence, Corollary 3 applies: $\forall u \in V$, $\pi_u := \frac{\delta_u}{2m}$ should be the solution of

- $\sum_{u \in V} \pi_u = 1$, and
- $\forall v \in V$, $\sum_{u \in V} \pi_u p(u, v) = \pi_v$

where $p(u, v) = \frac{1}{\delta_u}$ if u and v are neighbors, 0 otherwise.

$$\sum_{u \in V} \pi_u = \sum_{u \in V} \frac{\delta_u}{2m} = \frac{\sum_{u \in V} \delta_u}{2m}$$

Proof of Lemma 4 (1/2)

Consider an arbitrary standard random walk ω on G. Let $\pi = (\pi_v)_{v \in V}$ be the stationary distribution of Markov chain that models ω.

$\mathcal{H}_G(u,u) = \frac{1}{\pi_u}$.

Thus, the lemma holds if $\pi_u = \frac{\delta_u}{2m}$.

V is a finite set. G is connected and at each node, the probability of traversing each incident edge is strictly positive. So, the Markov chain modeling ω is finite and irreducible. Hence, Corollary 3 applies: $\forall u \in V$, $\pi_u := \frac{\delta_u}{2m}$ should be the solution of

- $\sum_{u \in V} \pi_u = 1$, and
- $\forall v \in V$, $\sum_{u \in V} \pi_u p(u,v) = \pi_v$

where $p(u,v) = \frac{1}{\delta_u}$ if u and v are neighbors, 0 otherwise.

$$\begin{aligned}\sum_{u \in V} \pi_u &= \sum_{u \in V} \frac{\delta_u}{2m} \\ &= \frac{\sum_{u \in V} \delta_u}{2m} \\ &= \frac{2m}{2m} = 1 \qquad \text{(handshaking lemma, [2])}\end{aligned}$$

Proof of Lemma 4 (2/2)

Let $v \in V$.

$$\sum_{u \in V} \pi_u p(u, v) = \sum_{u \in N(v)} \frac{\pi_u}{\delta_u}$$

standard random walk

Proof of Lemma 4 (2/2)

Let $v \in V$.

$$\sum_{u \in V} \pi_u p(u,v) = \sum_{u \in N(v)} \frac{\pi_u}{\delta_u} \qquad \text{standard random walk}$$

$$= \sum_{u \in N(v)} \frac{\delta_u}{\delta_u 2m} \qquad \pi_u := \frac{\delta_u}{2m}$$

Proof of Lemma 4 (2/2)

Let $v \in V$.

$$\sum_{u \in V} \pi_u p(u, v) = \sum_{u \in N(v)} \frac{\pi_u}{\delta_u} \qquad \text{standard random walk}$$

$$= \sum_{u \in N(v)} \frac{\delta_u}{\delta_u 2m} \qquad \pi_u := \frac{\delta_u}{2m}$$

$$= \sum_{u \in N(v)} \frac{1}{2m}$$

Proof of Lemma 4 (2/2)

Let $v \in V$.

$$\sum_{u \in V} \pi_u p(u,v) = \sum_{u \in N(v)} \frac{\pi_u}{\delta_u} \qquad \text{standard random walk}$$

$$= \sum_{u \in N(v)} \frac{\delta_u}{\delta_u 2m} \qquad \pi_u := \frac{\delta_u}{2m}$$

$$= \sum_{u \in N(v)} \frac{1}{2m}$$

$$= \frac{\delta_v}{2m} \qquad \delta_v = |N(v)|$$

$$= \pi_v$$

Thus, $\forall u \in V$, $\pi_u := \frac{\delta_u}{2m}$ is the solution! □

Roadmap

Bounds

In [1], the hitting time of the standard random walk is shown to be in

$$\Theta(n^3)$$

Bounds

In [1], the hitting time of the standard random walk is shown to be in

$$\Theta(n^3)$$

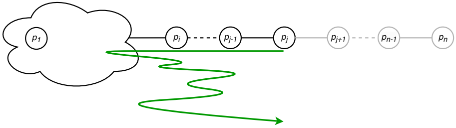
Let us now study the worst case

Basic properties

Let $p_1, \ldots, p_n$ the vertices of V.

Assume G has a pending line $L = p_i, \ldots, p_n$ with $i > 1$: $\forall j \in \{i, \ldots, n-1\}, \delta_{p_j} = 2$, $\delta_{p_n} = 1$, and the subgraph $G(L)$ induced by L is a line. Let p_{i-1} the neighbor of p_i such that $p_{i-1} \notin L$.

Assume a random walk starting from p_1

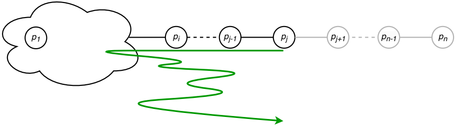


Basic properties

Let $p_1, \ldots, p_n$ the vertices of V.

Assume G has a pending line $L = p_i, \ldots, p_n$ with $i > 1$: $\forall j \in \{i, \ldots, n-1\}, \delta_{p_j} = 2$, $\delta_{p_n} = 1$, and the subgraph $G(L)$ induced by L is a line. Let p_{i-1} the neighbor of p_i such that $p_{i-1} \notin L$.

Assume a random walk starting from p_1



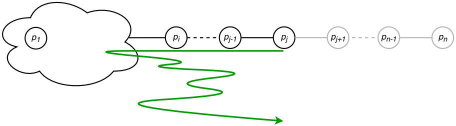
1. A walk that leaves and returns to p_n necessarily first goes to p_{n-1}, so $\mathcal{H}_G(p_n, p_n) = \mathcal{H}_G(p_n, p_{n-1}) + \mathcal{H}_G(p_{n-1}, p_n) = 1 + \mathcal{H}_G(p_{n-1}, p_n)$, so $\mathcal{H}_G(p_{n-1}, p_n) = \mathcal{H}_G(p_n, p_n) - 1 = 2m - 1$, by Lemma 4

Basic properties

Let $p_1, \ldots, p_n$ the vertices of V.

Assume G has a pending line $L = p_i, \ldots, p_n$ with $i > 1$: $\forall j \in \{i, \ldots, n-1\}, \delta_{p_j} = 2$, $\delta_{p_n} = 1$, and the subgraph $G(L)$ induced by L is a line. Let p_{i-1} the neighbor of p_i such that $p_{i-1} \notin L$.

Assume a random walk starting from p_1



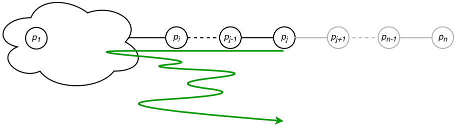
1. A walk that leaves and returns to p_n necessarily first goes to p_{n-1}, so $\mathcal{H}_G(p_n, p_n) = \mathcal{H}_G(p_n, p_{n-1}) + \mathcal{H}_G(p_{n-1}, p_n) = 1 + \mathcal{H}_G(p_{n-1}, p_n)$, so $\mathcal{H}_G(p_{n-1}, p_n) = \mathcal{H}_G(p_n, p_n) - 1 = 2m - 1$, by Lemma 4
2. $\forall j \in \{i, \ldots, n\}$, $\mathcal{H}_G(p_1, p_j) = \mathcal{H}_G(p_1, p_{j-1}) + \mathcal{H}_G(p_{j-1}, p_j)$: a walk from p_1 to p_j necessarily go via p_{j-1}

Basic properties

Let $p_1, \ldots, p_n$ the vertices of V.

Assume G has a pending line $L = p_i, \ldots, p_n$ with $i > 1$: $\forall j \in \{i, \ldots, n-1\}, \delta_{p_j} = 2$, $\delta_{p_n} = 1$, and the subgraph $G(L)$ induced by L is a line. Let p_{i-1} the neighbor of p_i such that $p_{i-1} \notin L$.

Assume a random walk starting from p_1



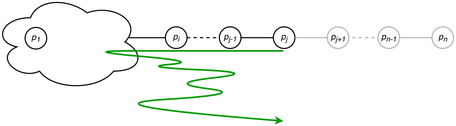
1. A walk that leaves and returns to p_n necessarily first goes to p_{n-1}, so $\mathcal{H}_G(p_n, p_n) = \mathcal{H}_G(p_n, p_{n-1}) + \mathcal{H}_G(p_{n-1}, p_n) = 1 + \mathcal{H}_G(p_{n-1}, p_n)$, so $\mathcal{H}_G(p_{n-1}, p_n) = \mathcal{H}_G(p_n, p_n) - 1 = 2m - 1$, by Lemma 4
2. $\forall j \in \{i, \ldots, n\}$, $\mathcal{H}_G(p_1, p_j) = \mathcal{H}_G(p_1, p_{j-1}) + \mathcal{H}_G(p_{j-1}, p_j)$: a walk from p_1 to p_j necessarily go via p_{j-1}
3. $\forall j \in \{i, \ldots, n\}$, $\mathcal{H}_G(p_{j-1}, p_j) = \mathcal{P}_{G(V \setminus \{p_{j+1}, \ldots, p_n\})}(p_{j-1}, p_j)$: a walk from p_{j-1} hits p_j before any vertex in $p_{j+1}, \ldots, p_n$.

Basic properties

Let $p_1, \ldots, p_n$ the vertices of V.

Assume G has a pending line $L = p_i, \ldots, p_n$ with $i > 1$: $\forall j \in \{i, \ldots, n-1\}, \delta_{p_j} = 2$, $\delta_{p_n} = 1$, and the subgraph $G(L)$ induced by L is a line. Let p_{i-1} the neighbor of p_i such that $p_{i-1} \notin L$.

Assume a random walk starting from p_1



1. A walk that leaves and returns to p_n necessarily first goes to p_{n-1}, so $\mathcal{H}_G(p_n, p_n) = \mathcal{H}_G(p_n, p_{n-1}) + \mathcal{H}_G(p_{n-1}, p_n) = 1 + \mathcal{H}_G(p_{n-1}, p_n)$, so $\mathcal{H}_G(p_{n-1}, p_n) = \mathcal{H}_G(p_n, p_n) - 1 = 2m - 1$, by Lemma 4
2. $\forall j \in \{i, \ldots, n\}$, $\mathcal{H}_G(p_1, p_j) = \mathcal{H}_G(p_1, p_{j-1}) + \mathcal{H}_G(p_{j-1}, p_j)$: a walk from p_1 to p_j necessarily go via p_{j-1}
3. $\forall j \in \{i, \ldots, n\}$, $\mathcal{H}_G(p_{j-1}, p_j) = \mathcal{P}_{G(V \setminus \{p_{j+1}, \ldots, p_n\})}(p_{j-1}, p_j)$: a walk from p_{j-1} hits p_j before any vertex in $p_{j+1}, \ldots, p_n$.
4. $\forall j \in \{i, \ldots, n\}$, $\mathcal{H}_G(p_{j-1}, p_j) = \mathcal{P}_{G(V \setminus \{p_{j+1}, \ldots, p_n\})}(p_{j-1}, p_j) = 2(m - (n - j)) - 1 = 2m - (2n - 2j + 1)$ by Property 1

First attempt: a line L

L: p_1 — p_2 — ... — p_n with $n > 1$

L is p_1 linked to a pending line p_2, ..., p_n so previous properties apply with $i = 2$.

$$\mathcal{H}_G(p_1, p_n) =$$

First attempt: a line L

L: p_1 — p_2 — ... — p_n with $n > 1$

L is p_1 linked to a pending line p_2, ..., p_n so previous properties apply with $i = 2$.

$$\mathcal{H}_G(p_1, p_n) = \sum_{j=2}^{n} \mathcal{H}_G(p_{j-1}, p_j) \qquad \text{by Property 2}$$

First attempt: a line L

L: p_1 — p_2 — $\dots$ — p_n with $n > 1$

L is p_1 linked to a pending line $p_2, \dots, p_n$ so previous properties apply with $i = 2$.

$$\mathcal{H}_G(p_1, p_n) = \sum_{j=2}^{n} \mathcal{H}_G(p_{j-1}, p_j) \qquad \text{by Property 2}$$

$$= \sum_{j=2}^{n} (2m - (2n - 2j + 1)) \qquad \text{by Property 4}$$

First attempt: a line L

L: p_1 — p_2 — $\ldots$ — p_n with $n > 1$

L is p_1 linked to a pending line p_2, ..., p_n so previous properties apply with $i = 2$.

$$\begin{aligned}
\mathcal{H}_G(p_1, p_n) &= \sum_{j=2}^{n} \mathcal{H}_G(p_{j-1}, p_j) && \text{by Property 2} \\
&= \sum_{j=2}^{n} (2m - (2n - 2j + 1)) && \text{by Property 4} \\
&= \sum_{j=2}^{n} (2n - 2 - (2n - 2j + 1)) && m = n - 1
\end{aligned}$$

First attempt: a line L

L: p_1 — p_2 — $\ldots$ — p_n with $n > 1$

L is p_1 linked to a pending line $p_2, \ldots, p_n$ so previous properties apply with $i = 2$.

$$\begin{aligned}
\mathcal{H}_G(p_1, p_n) &= \sum_{j=2}^{n} \mathcal{H}_G(p_{j-1}, p_j) && \text{by Property 2} \\
&= \sum_{j=2}^{n} (2m - (2n - 2j + 1)) && \text{by Property 4} \\
&= \sum_{j=2}^{n} (2n - 2 - (2n - 2j + 1)) && m = n - 1 \\
&= \sum_{j=2}^{n} (2j - 3)
\end{aligned}$$

First attempt: a line L

L: p_1 — p_2 — $\ldots$ — p_n with $n > 1$

L is p_1 linked to a pending line $p_2, \ldots, p_n$ so previous properties apply with $i = 2$.

$$\begin{aligned}
\mathcal{H}_G(p_1, p_n) &= \sum_{j=2}^{n} \mathcal{H}_G(p_{j-1}, p_j) && \text{by Property 2} \\
&= \sum_{j=2}^{n} (2m - (2n - 2j + 1)) && \text{by Property 4} \\
&= \sum_{j=2}^{n} (2n - 2 - (2n - 2j + 1)) && m = n - 1 \\
&= \sum_{j=2}^{n} (2j - 3) \\
&= 3 - 3n + \sum_{j=2}^{n} 2j = 3 - 3n + 2\sum_{j=2}^{n} j && (n-1). -3 = 3 - 3n
\end{aligned}$$

First attempt: a line L

L: p_1 — p_2 — $\ldots$ — p_n with $n > 1$

L is p_1 linked to a pending line p_2, …, p_n so previous properties apply with $i = 2$.

$$
\begin{aligned}
\mathcal{H}_G(p_1, p_n) &= \sum_{j=2}^{n} \mathcal{H}_G(p_{j-1}, p_j) && \text{by Property 2} \\
&= \sum_{j=2}^{n} (2m - (2n - 2j + 1)) && \text{by Property 4} \\
&= \sum_{j=2}^{n} (2n - 2 - (2n - 2j + 1)) && m = n - 1 \\
&= \sum_{j=2}^{n} (2j - 3) \\
&= 3 - 3n + \sum_{j=2}^{n} 2j = 3 - 3n + 2\sum_{j=2}^{n} j && (n-1). -3 = 3 - 3n \\
&= 3 - 3n + 2\frac{(n+2)(n-1)}{2}
\end{aligned}
$$

First attempt: a line L

L: p_1 — p_2 — ... — p_n with $n > 1$

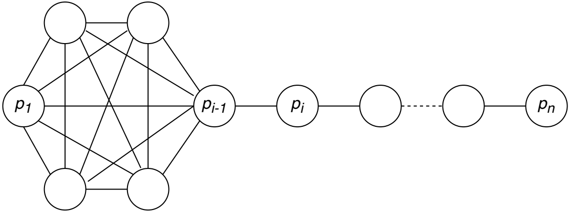
L is p_1 linked to a pending line p_2, ..., p_n so previous properties apply with $i = 2$.

$$
\begin{aligned}
\mathcal{H}_G(p_1, p_n) &= \sum_{j=2}^{n} \mathcal{H}_G(p_{j-1}, p_j) && \text{by Property 2} \\
&= \sum_{j=2}^{n} (2m - (2n - 2j + 1)) && \text{by Property 4} \\
&= \sum_{j=2}^{n} (2n - 2 - (2n - 2j + 1)) && m = n - 1 \\
&= \sum_{j=2}^{n} (2j - 3) \\
&= 3 - 3n + \sum_{j=2}^{n} 2j = 3 - 3n + 2\sum_{j=2}^{n} j && (n-1).-3 = 3 - 3n \\
&= 3 - 3n + 2\frac{(n+2)(n-1)}{2} \\
&= n^2 - 2n + 1 \in \Theta(n^2)
\end{aligned}
$$

Second attempt: Lollipop

A lollipop consists of a clique linked by a bridge to a line

Let us consider a lollipop made of vertices $p_1, \ldots, p_n$ with $n > 2$ where p_1, p_{i-1} is the clique with $i > 2$ and a standard random walk starting from p_1



$$m = \frac{(i-1)(i-2)}{2} + n - (i-1) = \frac{i^2 - 5i}{2} + n + 2$$

Analysis (1/3)

Until reaching p_{i-1}, the probability of hitting p_{i-1} at the next step is $\frac{1}{i-2}$: it is a geometric law. Thus,

$$\mathcal{H}_G(p_1, p_{i-1}) = i - 2$$

We now compute $\mathcal{H}_G(p_1, p_n)$

Analysis (2/3)

$\mathcal{H}_G(p_1, p_n) =$

Analysis (2/3)

$$\mathcal{H}_G(p_1, p_n) = \mathcal{H}_G(p_1, p_{i-1}) + \sum_{j=i}^{n} \mathcal{H}_G(p_{j-1}, p_j)$$ by Property 2

Analysis (2/3)

$$\mathcal{H}_G(p_1, p_n) = \mathcal{H}_G(p_1, p_{i-1}) + \sum_{j=i}^{n} \mathcal{H}_G(p_{j-1}, p_j) \qquad \text{by Property 2}$$

$$= i - 2 + \sum_{j=i}^{n} \mathcal{H}_G(p_{j-1}, p_j)$$

Analysis (2/3)

$$\begin{aligned}
\mathcal{H}_G(p_1, p_n) &= \mathcal{H}_G(p_1, p_{i-1}) + \sum_{j=i}^{n} \mathcal{H}_G(p_{j-1}, p_j) && \text{by Property 2} \\
&= i - 2 + \sum_{j=i}^{n} \mathcal{H}_G(p_{j-1}, p_j) \\
&= i - 2 + \sum_{j=i}^{n} (2m - (2n - 2j + 1)) && \text{by Property 4}
\end{aligned}$$

Analysis (2/3)

$$\begin{aligned}
\mathcal{H}_G(p_1, p_n) &= \mathcal{H}_G(p_1, p_{i-1}) + \sum_{j=i}^{n} \mathcal{H}_G(p_{j-1}, p_j) && \text{by Property 2} \\
&= i - 2 + \sum_{j=i}^{n} \mathcal{H}_G(p_{j-1}, p_j) \\
&= i - 2 + \sum_{j=i}^{n} (2m - (2n - 2j + 1)) && \text{by Property 4} \\
&= i - 2 + (n - i + 1).(2m - 2n - 1) + 2\sum_{j=i}^{n} j
\end{aligned}$$

Analysis (2/3)

$$
\begin{aligned}
\mathcal{H}_G(p_1, p_n) &= \mathcal{H}_G(p_1, p_{i-1}) + \sum_{j=i}^{n} \mathcal{H}_G(p_{j-1}, p_j) && \text{by Property 2} \\
&= i - 2 + \sum_{j=i}^{n} \mathcal{H}_G(p_{j-1}, p_j) \\
&= i - 2 + \sum_{j=i}^{n} (2m - (2n - 2j + 1)) && \text{by Property 4} \\
&= i - 2 + (n - i + 1).(2m - 2n - 1) + 2\sum_{j=i}^{n} j \\
&= i - 2 + (n - i + 1).(2m - n + i - 1)
\end{aligned}
$$

Analysis (2/3)

$$\begin{aligned}
\mathcal{H}_G(p_1, p_n) &= \mathcal{H}_G(p_1, p_{i-1}) + \sum_{j=i}^{n} \mathcal{H}_G(p_{j-1}, p_j) && \text{by Property 2} \\
&= i - 2 + \sum_{j=i}^{n} \mathcal{H}_G(p_{j-1}, p_j) \\
&= i - 2 + \sum_{j=i}^{n} (2m - (2n - 2j + 1)) && \text{by Property 4} \\
&= i - 2 + (n - i + 1).(2m - 2n - 1) + 2\sum_{j=i}^{n} j \\
&= i - 2 + (n - i + 1).(2m - n + i - 1) \\
&= i - 2 + (n - i + 1).(i^2 - 4i + n + 3)
\end{aligned}$$

Analysis (3/3)

Let $i := \frac{n}{4}$.

$$\mathcal{H}_G(p_1, p_n) = \frac{n}{4} - 2 + (\frac{3n}{4} + 1).(\frac{n^2}{16} + 3)$$
$$= \frac{3n^3}{64} + \frac{n^2}{16} + \frac{10n}{4} + 1 \in \Theta(n^3)$$

Actually, the lollipop graph is shown to be the worst case in [6]: precisely the lollipops with a clique of $\frac{2n}{3}$ vertices

Roadmap

Bounds

From [4, 3], we know that the cover time of the standard random walk is also in $\Theta(n^3)$.

Again, the worst-case graph is the lollipop with a clique of $\frac{2n}{3}$ vertices!

Let $C \geq C_G(P)$.

Assume u has a data d to broadcast.

Initialization

```
deliver d
```
pick $i \in \{1, \ldots, \delta_u\}$ according to P_u

send $\langle d, 1 \rangle$ via port number i

v **receives** $\langle d, i \rangle$

```
deliver d
```
if $i < C$ **then**

  pick $i \in \{1, \ldots, \delta_v\}$ according to P_v

  send $\langle d, i+1 \rangle$ via port number i

end if

Termination in C hops and partial correctness *w.h.p.* (works in anonymous networks; yet, duplicates ...).

Roadmap

What is the issue with the standard random walk?

What is the issue with the standard random walk?

Lemma 4 claims that **the more the degree of a node is the more often it is visited!**

It is an issue!

Indeed

- In the lollipop, we have both very high degree nodes and very low degree nodes: the hitting time is in $\Theta(n^3)$
- In a line, degrees are almost equal (either 1 or 2): the hitting time is in $\Theta(n^2)$ although the diameter is maximal!

What is the issue with the standard random walk?

Lemma 4 claims that **the more the degree of a node is the more often it is visited!**

It is an issue!

Indeed

- In the lollipop, we have both very high degree nodes and very low degree nodes: the hitting time is in $\Theta(n^3)$
- In a line, degrees are almost equal (either 1 or 2): the hitting time is in $\Theta(n^2)$ although the diameter is maximal!

Solution: load balance the probability distributions

Probability Distributions proposed in [7]

$$p(u,v) = \begin{cases} \dfrac{\delta_v^{-1/2}}{\sum\limits_{w \in N(u)} \delta_w^{-1/2}} & \text{if } v \in N(u) \\ 0 & \text{otherwise} \end{cases}$$

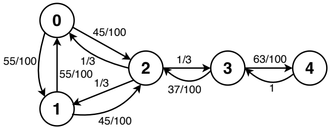
A minor drawback is that each node should know the degree of its neighbors

(but, it is still local information)

Probability Distributions proposed in [7]

$$p(u,v) = \begin{cases} \dfrac{\delta_v^{-1/2}}{\sum_{w \in N(u)} \delta_w^{-1/2}} & \text{if } v \in N(u) \\ 0 & \text{otherwise} \end{cases}$$

Markov chain of the random walk given in [7] on a lollipop



A minor drawback is that each node should know the degree of its neighbors

(but, it is still local information)

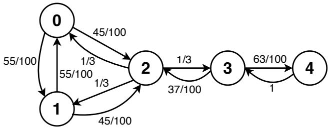
Probability Distributions proposed in [7]

$$p(u,v) = \begin{cases} \dfrac{\delta_v^{-1/2}}{\sum_{w \in N(u)} \delta_w^{-1/2}} & \text{if } v \in N(u) \\ 0 & \text{otherwise} \end{cases}$$

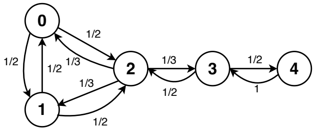
A minor drawback is that each node should know the degree of its neighbors

(but, it is still local information)

Markov chain of the random walk given in [7] on a lollipop



Markov chain of the standard random walk on the same graph



Bounds

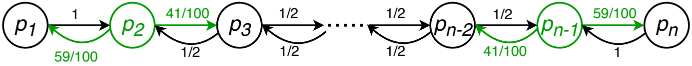
Hitting Time: $\Theta(n^2)$

It is the **optimal** distribution for the pure random walk

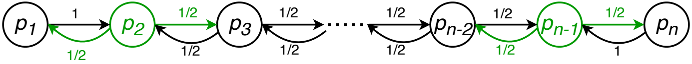
Cover Time: $O(n^2 \log n)$

A few more details

The lower bound is natural: in a line, only two vertices (p_2 and p_{n-1}) have distributions that differ from the standard random walk



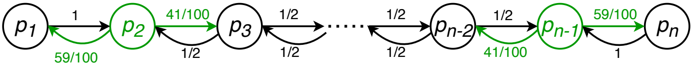
Markov chain of the random walk given in [7] on a line



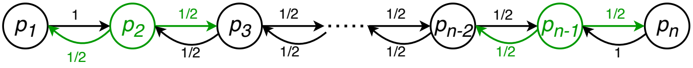
Markov chain of the standard random walk on a line

A few more details

The lower bound is natural: in a line, only two vertices (p_2 and p_{n-1}) have distributions that differ from the standard random walk



Markov chain of the random walk given in [7] on a line

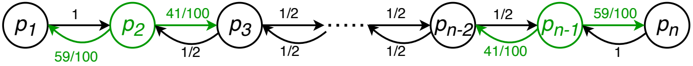


Markov chain of the standard random walk on a line

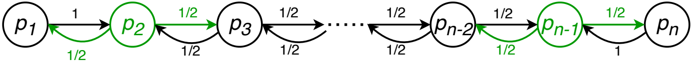
Intuition: With an arbitrary large line, the difference between the standard random walk and the one of [7] becomes negligible, thus we have $\Omega(n^2)$.

A few more details

The lower bound is natural: in a line, only two vertices (p_2 and p_{n-1}) have distributions that differ from the standard random walk



Markov chain of the random walk given in [7] on a line



Markov chain of the standard random walk on a line

Intuition: With an arbitrary large line, the difference between the standard random walk and the one of [7] becomes negligible, thus we have $\Omega(n^2)$.

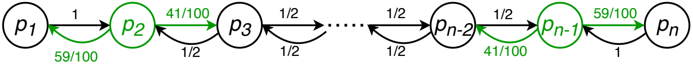
Proof: Assume G is a line $p_1 — \ldots — p_n$ with $n > 1$. Let $\mathfrak{H}_G(p_i, p_j)$ be the hitting time from p_i to p_j under the transition probability matrix of the random walk of [7].

$$\mathfrak{H}_G(p_1, p_n) > \mathfrak{H}_G(p_1, p_{n-1}) > \mathcal{H}_G(p_1, p_{n-1}) \in \Omega(n^2)$$

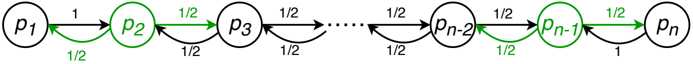
($n.b.$, $\mathfrak{H}_G(p_1, p_{n-1}) > \mathcal{H}_G(p_1, p_{n-1})$ since $\frac{41}{100} < \frac{1}{2}$) $\square$

A few more details

The lower bound is natural: in a line, only two vertices (p_2 and p_{n-1}) have distributions that differ from the standard random walk



Markov chain of the random walk given in [7] on a line



Markov chain of the standard random walk on a line

Intuition: With an arbitrary large line, the difference between the standard random walk and the one of [7] becomes negligible, thus we have $\Omega(n^2)$.

Proof: Assume G is a line $p_1 — \ldots — p_n$ with $n > 1$. Let $\mathfrak{H}_G(p_i, p_j)$ be the hitting time from p_i to p_j under the transition probability matrix of the random walk of [7].

$$\mathfrak{H}_G(p_1, p_n) > \mathfrak{H}_G(p_1, p_{n-1}) > \mathcal{H}_G(p_1, p_{n-1}) \in \Omega(n^2)$$

($n.b.$, $\mathfrak{H}_G(p_1, p_{n-1}) > \mathcal{H}_G(p_1, p_{n-1})$ since $\frac{41}{100} < \frac{1}{2}$) □

The upper bound is more complex! (see [7])

Roadmap

Pros and Cons of Random-walk-based Routing

Pros.

- Partially Correct
- Robust
- Adaptive
- Fair
- Messages: low message overhead and no control message
- Low memory at each process

Cons.

- Termination *almost sure* only
- Slow: $\Omega(n^2)$

  In many large-scale networks, the diameter is logarithmic in n, *e.g.*, IPv6, which allows for up to 2^{128} machines, assumes the diameter is at most 255!
- Not FIFO

Roadmap

References

[1] G. R. Brightwell and P. Winkler.
Maximum hitting time for random walks on graphs.
Random Struct. Algorithms, 1(3):263–276, 1990.

[2] L. Euler.
Solutio problematis ad geometriam situs pertinentis.
Commentarii Academiae Scientiarum Imperialis Petropolitanae, 8:128–140, 1736.

[3] U. Feige.
A tight lower bound on the cover time for random walks on graphs.
Random Struct. Algorithms, 6(4):433–438, 1995.

[4] U. Feige.
A tight upper bound on the cover time for random walks on graphs.
Random Struct. Algorithms, 6(1):51–54, 1995.

[5] G. Frobenius.
Über Matrizen aus nicht negativen Elementen.
Preussische Akademie der Wissenschaften Berlin: Sitzungsberichte der Preußischen Akademie der Wissenschaften zu Berlin. Reichsdr., 1912.

[6] F. Gobel and A. Jagers.
Random walks on graphs.
Stochastic processes and their applications, 2(4):311–336, 1974.

[7] S. Ikeda, I. Kubo, and M. Yamashita.
The hitting and cover times of random walks on finite graphs using local degree information.
Theor. Comput. Sci., 410(1):94–100, 2009.

[8] O. Perron.
Zur theorie der matrices.
Mathematische Annalen, 64:248–263, 1907.