CS 761: Randomized Algorithms Fall 2019

Lecture 7 — October 25, 2019

Prof. Gautam Kamath *By: Amur Ghose, Cameron Seth*
Edited by Vedat Levi Alev

1 Random Walk

Given a graph G, a random walk starts at a node and at each time step moves to a uniformly random neighbour of the current node. We are interested in the following questions:

Question 1. *What is the limiting distribution? (stationary distribution)*

Question 2. *How many steps before approaching the limiting distribution? (mixing time)*

Question 3. *Starting from node s, how long does it take to reach node t? (hitting time)*

Question 4. *How long does it take to visit each node at least once? (cover time)*

2 Markov Chain

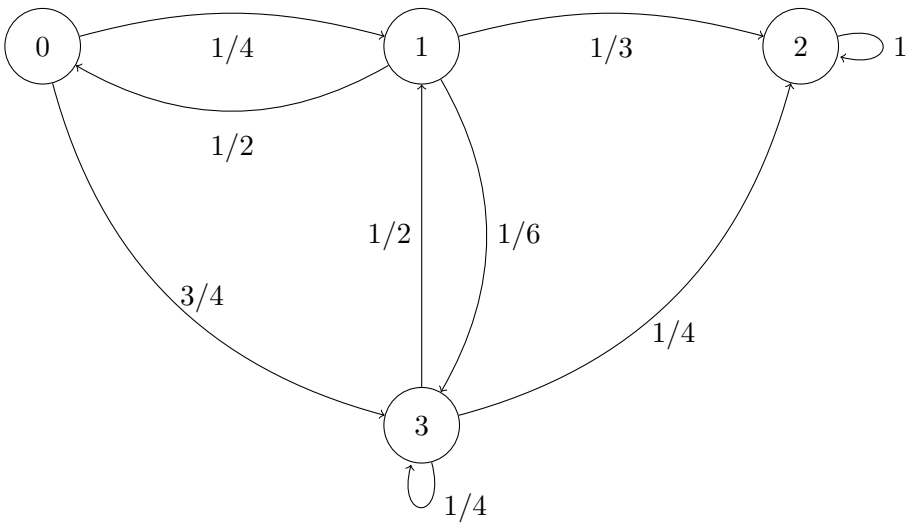
A Markov chain can be represented by a more generalized random walk on a directed graph. In particular, a Markov chain starts at a node and at each time step moves to a neighbour of the current node according to the transition probability matrix P.

$$P_{ij} = \Pr[\text{next state is j given current state i}]$$

Let X_t denote the state of the Markov chain at time t. The defining feature of a Markov chain is that the future behaviour of the Markov chain depends only on its current state. Specifically:

$$\Pr[X_t = a_t | X_{t-1} = a_{t-1}, \ldots, X_0 = a_0] = \Pr[X_t = a_t | X_{t-1} = a_{t-1}] = P_{(a_{t-1})(a_t)}$$

Example 1. *In the following graph the nodes are the possible states of the Markov chain and the edge weights represent the probability of following that edge on the next time step.*



The transition probability matrix is

$$\begin{bmatrix} 0 & \frac{1}{4} & 0 & \frac{3}{4} \\ \frac{1}{2} & 0 & \frac{1}{3} & \frac{1}{6} \\ 0 & 0 & 1 & 0 \\ 0 & \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \end{bmatrix}$$

Let $p_t(i)$ be the probability of being at state i at time t. Then at any time t, $\vec{p}_t$ is a probability distribution over the states. For example, if we start at the first state then $\vec{p}_0 = (1, 0, 0, \ldots, 0)$. If we have a random start then $\vec{p}_0 = (\frac{1}{n}, \ldots, \frac{1}{n})$.

By definition we can calculate the state at time $t+1$ as $p_{t+1}(j) = \sum_{i=0}^{n-1} p_t(i) P_{ij}$. The vector form is $\vec{p}_{t+1} = \vec{p}_t P$ and in general the recurrence simplifies to $\vec{p}_{t+m} = \vec{p}_t P^m$.

2.1 States

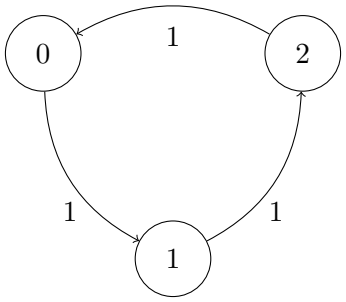
Consider a finite Markov chain with n states and associated graph $G = (V, E)$. Denote the states $s_1, \ldots, s_n$.

Definition 1. *A Markov chain is* ***irreducible*** *if G is strongly connected, i.e. there exists a directed path from i to j for all pairs of states (i, j). In other words $\forall i, j \in V, \exists l > 0$ such that $\Pr[X_{t+l} = s_j | X_t = s_i] > 0$.*

Example 2. *The Markov chain graph in Example 1 is not irreducible because the there exists no path from the 2 node to the other nodes.*

Definition 2. *The* ***period*** *of a Markov chain d is defined as $d(s_i) = gcd(\{t \in \mathbb{N} | (P^t)_{ii} > 0, t > 0\})$. If $d(s_i) = 1$ $\forall i \in V$ then the Markov chain is* ***aperiodic****. Otherwise it is* ***periodic****.*

Example 3. *The following Markov chain is periodic with period 3.*



Theorem 3. *If a Markov chain is finite, irreducible and aperiodic then*

$$\exists T < \infty \text{ such that } (P^t)_{ij} > 0,\ \forall t \geq T,\ \forall i, j \in V$$

Definition 4. *$\vec{\pi}$ is a **stationary distribution** of a Markov chain if $\vec{\pi} = \vec{\pi}P$.*

Definition 5. *Given distribution $\vec{p} = (p_1, \ldots, p_n)$ and $\vec{q} = (q_1, \ldots, q_n)$ the **total variation distance** is*

$$d_{TV}(\vec{p}, \vec{q}) = \frac{1}{2} \sum_{i=1}^{n} |p_i - q_i|$$

In the context of Markov chains, $\vec{p_t}$ converges to $\vec{q}$ if $\lim_{t\to\infty} d_{TV}(\vec{p_t}, \vec{q}) = 0$.

Definition 6. *The **hitting time** from i to j is defined as $H_{ij} = \min(\{t \geq 1 \mid X_t = s_j, X_0 = s_i\})$. The **expected hitting time** is $h_{ij} = E[H_{ij}]$.*

Theorem 7 (Fundamental Theorem of Markov Chains)**.** *For any finite, irreducible and aperiodic Markov chain:*

1. *There exists a stationary distribution $\vec{\pi}$.*
2. *The distribution $\vec{p_t}$ will converge to $\vec{\pi}$ as $t \to \infty$ regardless of the initial state $\vec{p_0}$.*
3. *$\vec{\pi}$ is unique.*
4. *$\pi_i = \lim_{t\to\infty}(P^t)_{ii} = \frac{1}{h_{ii}}$.*

3 PageRank

Consider a directed graph $G = (V, E)$ of the web where a node is a webpage and each directed edge is a link from one webpage to another. We want to rank the pages by importance. Intuitively, a webpage is important if other important pages link to it.

3.1 PageRank Algorithm

1. Assign a pagerank value of $\frac{1}{n}$ to each node.
2. For each node divide the current pagerank value by the out-degree of the node, and send to each out-link equally.
3. The pagerank of each node is updated to be the sum all of values received.

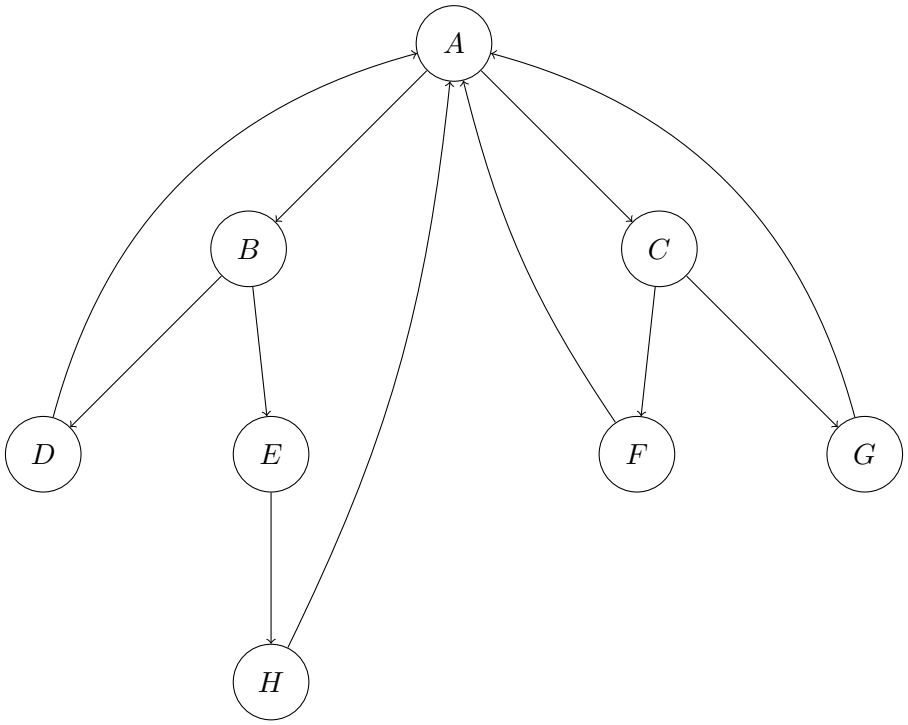
3.2 Equilibrium States

We are interested in the equilibrium state of the algorithm. First, observe that this algorithm resembles a Markov chain with transition probability matrix

$$P_{ij} = \begin{cases} \frac{1}{\deg_{\text{out}}(i)} & \text{if page i links to page j} \\ 0 & \text{otherwise} \end{cases}$$

At each step the new pagerank of a node j is $\sum_{i:(i,j)\in E} \frac{pagerank(i)}{\deg_{\text{out}}(i)}$. Letting $pagerank(i) = p(i)$ it follows that a stationary distribution of the Markov chain is $\vec{p} = \vec{p}P$. By the Fundamental Theorem of Markov Chains, if the web is finite, irreducible and aperiodic then a unique stationary distribution exists and this algorithm will converge to it. The stationary distribution gives the unique equilibrium pagerank values we are interested in.

Example 4. *In the example below an equilibrium pagerank setting is* $(A, B, C, D, E, F, G, H) = (\frac{4}{13}, \frac{2}{13}, \frac{2}{13}, \frac{1}{13}, \frac{1}{13}, \frac{1}{13}, \frac{1}{13}, \frac{1}{13})$. *After one step of the algorithm the pageranks of all the nodes stay the same.*



3.3 Scaled PageRank

In general the web is not irreducible and aperiodic. So the algorithm above will not necessarily find an equilibrium. It is also possible that the algorithm converges to a equilibrium that doesn't make sense (i.e. having all the pagerank concentrated on one or two outlying pages). Instead we modify the algorithm as follows.

1. Assign a pagerank value of $\frac{1}{n}$ to each node.

2. For each node divide the current pagerank value by the out-degree of the node, and send to each out-link equally.
3. The pagerank of each node is updated to be the sum all of values received.
4. Select a scaling factor s with $0 \leq s \leq 1$. Scale down each pagerank by a factor of s.
5. Add $\frac{1-s}{n}$ to the pagerank of every node.

After scaling down the total pagerank, the algorithm redistributes the leftover pagerank to all the other nodes. Hence the total pagerank will still sum to 1. The new transition probability matrix will be

$$P_{ij} = \begin{cases} \frac{s}{\deg_{\text{out}}(i)} & \text{if page i links to page j} \\ \frac{1-s}{n} & \text{otherwise} \end{cases}$$

The graph associated with the new Markov chain is strongly connected and aperiodic so we are guaranteed an equilibrium solution.

4 WalkSAT

It is well known that 3-SAT is a hard problem, but what about 2-SAT? We define the problem as follows with Boolean variables Z_1 to Z_n. We will indicate the corresponding formula in CNF form as

$$\bigwedge_{i=1}^{M} C_i, C_i = A \vee B$$

Where both A, B are either Z_i or $\bar{Z}_i$ for some i. The 2 in 2-SAT arises from having exactly 2 literals per clause C_i.

The algorithm pseudocode is as follows:

- Start at an arbitrary assignment of Z_i values
- For $200n^2$ iterations, repeat the following step, and stop if the formula evaluates to true
- Pick any clause C evaluating to false. Choose a random Z_i appearing in C and flip it.
- If you got a satisfying assignment, return it. Else return `UNSAT`.

Note that you have to evaluate the formula per iteration. Hence this has $O(n^3)$ complexity.

Let S be any configuration of Z_is which satisfies the formula. At iteration i consider the assignment of variables as A_i and the random variable X_i as the number of elements in common between A_i and S. Now, suppose $X_i = 0$. We wish to consider the Markov chain on X_i's values as the states of interest.

If $X_i = 0$ then we are assured of $X_{i+1} = 1$, as regardless of what variable we flip, everything at $X_i = 0$ is "misaligned". Hence

$$\Pr[X_{i+1} = 1 | X_i = 0] = 1$$

We now consider $1 \leq X_i \leq n-1$. Observe that if we pick an unsatisfied clause, at least one of the variables must be set to a value that is 'incorrect' relative to S. Hence we get

$$\Pr[X_{i+1} = j+1 | X_i = j] \geq \frac{1}{2}$$

$$\Pr[X_{i+1} = j-1 | X_i = j] \leq \frac{1}{2}$$

The above is hard to analyze. However, let us create mirror variables Y_i that have the same domain as X_i that obey

$$\Pr[Y_{i+1} = 1 | Y_i = 0] = 1$$

$$\Pr[Y_{i+1} = j+1 | Y_i = j] = \frac{1}{2}$$

$$\Pr[Y_{i+1} = j-1 | Y_i = j] = \frac{1}{2}$$

It is clear that if we take some M steps to find the satisfying assignment in terms of Y i.e. $Y_i = n$ at $i = M$ we'll take $\leq M$ to find it using X_i **in expectation**. Let h_j be the number of steps **in expectation** to reach n if we start at spot j at some Y_i.

With $1/2$ probability we advance, and with $1/2$ we take a step back. Immediately we get the recursion

$$h_j = \frac{1}{2}(h_{j+1} + 1) + \frac{1}{2}(h_{j-1} + 1)$$

Simplify to get

$$h_j - h_{j+1} = h_{j-1} - h_j + 2$$

Recall that at n no time is required, and at step 0 we are bound to move forward

$$h_n = 0, h_0 - h_1 = 1$$

Writing the whole thing as a telescoping sum

$$h_j - h_{j+1} = h_{j-1} - h_j + 2 = h_{j-2} - h_{j-1} + 4 \cdots = 2j + h_0 - h_1 = 2j + 1$$

Now use that to get

$$h_0 = h_0 - 0 = h_0 - h_n = \sum_{i=0}^{n-1}(h_i - h_{i+1}) = \sum_{i=0}^{n-1}(2j+1)$$

The sum of consecutive odd numbers are the square numbers. Hence

$$h_0 = n^2$$

Let's assume we run for $2cn^2$ times, $2 \leq c \in \mathbb{N}$. The probability of not succeeding in the first $2n^2$ steps is $\leq 1/2$ by Markov. The next $2n^2$ steps begin from a step $i \geq 0$ and have an expected time of hitting n which is $\leq n^2$. Therefore, if we run for $200n^2$ steps and repeat this argument, we end up with a success probability which is $\geq 1 - 1/200$ (Markov) but more sharply $\geq 1 - \frac{1}{2^{100}}$ by noting that we can repeat this argument of having a success probability $\geq 1/2$ for every $2n^2$ steps, reapplying Markov's while starting from any i, and that failure implies we chose an outcome with probability $\leq 1/2$ every time.

5 3-SAT WalkSAT

Unlike 2-SAT, we should anticipate an exponential run time in expectation for this problem as it is well known to be hard. In 3-SAT, the CNF form associates clauses with 3 variables. Repeating the previous arguments gives us, for our auxiliary variables Y_i, h_j as before,

$$h_n = 0, h_j = \frac{2}{3}h_{j-1} + \frac{1}{3}h_{j+1} + 1$$

$$h_0 - h_1 = 1, h_j - h_{j+1} = 2(h_{j-1} - h_j) + 3$$

Solve the recursions to get

$$h_j - h_{j+1} = 2^{j+2} - 3 \to h_j = 2^{n+2} - 2^{j+2} - 3(n-j)$$

$$h_0 = 2^{n+2} - 3n - 4$$

To frame the pseudo code, let us try the following: We will repeat each **internal iteration** n times, where within each such, we will generate a random assignment, and perform $3n$ steps where in each step we pick a random unsatisfied clause and flip a variable. We shall return the satisfying assignment - or `UNSAT` - as before.

Now suppose we have $X_0 = n - j$, i.e. j 'mismatches' relative to the satisfying assignment. We can reach n in $3j$ steps should we go j steps right (net), which solves to moving $2j$ steps ahead and j backwards. The probability of that would be

$$\binom{3j}{j}(\frac{2}{3})^j(\frac{1}{3})^{2j}$$

Apply Stirling's approximation

$$\sqrt{2\pi m}(\frac{m}{e})^m \leq m! \leq 2\sqrt{2\pi m}(\frac{m}{e})^m$$

The binomial coefficient above can thus be bounded by

$$\frac{3j!}{2j!j!} \geq \frac{\sqrt{2\pi(3j)}}{4\sqrt{2\pi(2j)}\sqrt{2\pi(j)}} \times (\frac{3j}{e})^{3j}(\frac{e}{2j})^{2j}(\frac{e}{j})^j$$

The RHS simplifies to

$$\frac{\sqrt{3}}{8\sqrt{\pi j}}(\frac{27}{4})^j = \frac{c}{\sqrt{j}}(\frac{27}{4})^j, c = \frac{\sqrt{3}}{8\sqrt{\pi}}$$

So, probability of reaching n starting at j is

$$\geq \frac{c}{\sqrt{j}}(\frac{27}{4})^j(\frac{2}{3})^j(\frac{1}{3})^{2j} = \frac{c}{\sqrt{j}}(\frac{1}{2})^j$$

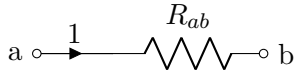
Now we can try to get the probability of reaching n by observing we start with a random assignment. Therefore the probability of exactly j mismatches is trivially $\binom{n}{j}2^{-n}$. We can now get the probability of hitting n by computing

$$\sum_{j=0}^{n} \frac{c}{\sqrt{j}}(\frac{1}{2})^j\binom{n}{j}(\frac{1}{2})^n \geq \frac{c}{\sqrt{n}}(\frac{1}{2})^n\sum_{j=0}^{n}\binom{n}{j}(\frac{1}{2})^j$$

Observe that the term within the summation is just the binomial expansion when expanding $(1 + 1/2)^n$. Therefore, the expression is

$$\frac{c}{\sqrt{n}}(\frac{1}{2})^n(1 + 1/2)^n = \frac{c}{\sqrt{n}}(\frac{3}{4})^n$$

So, on expectation, we'll need $\frac{\sqrt{n}}{c}(\frac{4}{3})^n$ tries to get a success. Overall, the algorithm is $O(n^{1.5}(1.33)^n)$ (the extra n coming from the evaluation per iteration) and the best known algorithm so far is $O(1.308^n)$. As mentioned at the start of the section, this algorithm for 3-SAT has exponential runtime in expectation, unlike the 2-SAT algorithm.



Resistance equation: $R_{ab} = \phi_a - \phi_b$ when $I = 1$

6 Random walks on undirected graphs

Here, we will have a transition probability that is uniform over all neighbours of a vertex $v \in V$ where the graph $G = (V, E)$ i.e. vertex and edge sets in standard notation.

Lemma 8. *The stationary distribution on an undirected connected non bipartite graph for the random walk on its vertices is* $\frac{d(v)}{2m}$*, where* $d(v)$ *is the degree of* $v \in V$ *and* $m = |E|$

To prove this, note that if P be the transition matrix, $P_{uv} = \frac{1}{d(v)}$ as per definition. Take the above stationary distribution as Π and check that $\Pi_v = (\Pi P)_v$, with $\|\Pi\|_1 = 1$ being trivially true.

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

7 Graphs as electrical networks

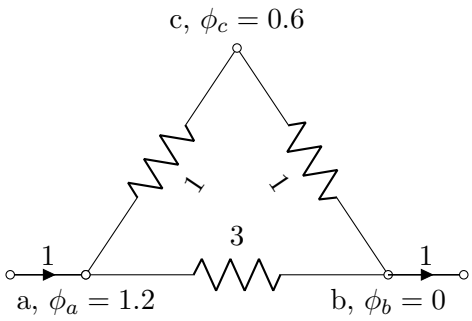
In the case of interpreting graphs as electrical networks, an edge is a resistor of resistance 1 (weighed graphs would be more general and cover all resistors) . We recall the two basic laws of circuits as *Kirchoff's law*: the current entering any node v is equal to leaving it, and *Ohm's law*, which can be stated as $V = RI$, V being the potential difference across a resistor R and I the current across. More formally, given f_{uv}, r_{uv} as the current and resistance from u to v respectively, we can come up with a voltage vector $\phi : V \to \mathbb{R}$ such that

$$\phi_u - \phi_v = f_{uv} r_{uv}, \forall u, v \in V$$

Recall that we have by Ohm's the law relating ϕ and resistance: R values are equivalently definable as potential (ϕ) differences when currents are set to 1. The toy circuit provided shows a simple circuit solved by repeatedly applying the above laws.

Now, for such graphs, we can talk about h_{uv} - the hitting time - the usual way, but we can also define the symmetric quantity of **commute time** as $C_{uv} = h_{uv} + h_{vu}$. Observe that h_{uv} might be quite different from h_{vu}. For a concrete example, construct:

- The complete graph on n vertices K_n as G
- Take any fixed $v \in G$ and add a vertex connected only to v as v_1. Now add another vertex connected only to v_1, and so on, till you have a "line" till v_n.
- Take any $v' \in G$. Consider the hitting time starting from v' till v_n. h_{v',v_n} is $O(n^3)$. However, $h_{v_n,v'} = O(n^2)$.



Toy circuit

- Intuition: Starting from the "rod's edge", we can only travel towards the clique-shaped graph. This'll take $O(n^2)$ steps (recall analysis of 2-SAT WalkSAT) . Escaping the clique requires being at the escape spot, therefore an additional factor of n appears. This kind of structure is called the **Lollipop graph**.

8 Relating commute time and resistance

Theorem 9. *Given an undirected unweighed graph $G = (V, E)$, $\forall u, v, \in V$, we have that $C_{uv} = 2mR_{uv}$ where $m = |E|$ and R_{uv} is the effective resistance given all edges have resistance* 1*.*

To prove this, pick u, v and fix them, and $\forall x \in V$, inject $deg(x)$ of current into it. Remove $2m$ current from v. Note that net flow of current into the graph as a whole is zero as $\sum_{x \in V} deg(x) = 2m$. By *Kirchoff's law* above, we have that for all nodes, current into the node equals current out. Let f_{ij} be the directed flow of current $i \to j$, and, writing $N(x)$ as the neighbourhood of a node x with $d(x) = deg(x)$:

$$d(x) = \sum_{w \in N(x)} f_{xw} = \sum_{w \in N(x)} \phi_x - \phi_w = \sum_{w \in N(x)} \phi_{xv} - \phi_{wv}$$

The second equation follows since ϕ_v cancels out from both terms. Observe that we have a recursion for h_{xv} as we can go to any of x's neighbours and add 1 to our total time taken to hit.

$$h_{xv} = \sum_{w \in N(x)} (h_{wv} + 1)\frac{1}{d(x)} \to d(x)h_{xv} = d(x) + \sum_{w \in N(x)} h_{wv}$$

And since $d(x)h_{xv} = \sum_{w \in N(x)} h_{xv}$, we get

$$d(x) = \sum_{w \in N(x)} (h_{xv} - h_{wv})$$

Compare this equation to the previous one for ϕ. For it to hold identically, we'll need $\phi_{xv} = h_{xv}$, for all $x \in V$. By the definition of the potential function ϕ we have $h_{vv} = \phi_{vv} = 0$.

We now carry out the reverse process. Instead of v, we will now inject $2m$ at u, and take $d(x)$ out from each $x \in V$. We'll get $h_{xu} = \phi_{xu}$, off of this by repeating the analysis for v. Now, we add the two cases, by the principle of superposition. We took $d(x)$ in and out so except u, v there will be no current. Between u, v we'll have a current flow of $2m$.

$$\phi_{uv} + \phi_{vu} = h_{uv} + h_{vu} = C_{uv}$$

But we got these potential values for a current flow (I) of $2m$. The potential differences (V) on the LHS are writable as $V = RI$, via Ohm's law. This gives us the sought result:

$$C_{uv} = 2mR_{uv}$$

References

[1] David Easley and Jon Kleinberg. *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*. Cambridge University Press, New York, NY, USA, 2010.

[2] David A Levin and Yuval Peres. *Markov chains and mixing times*, volume 107. American Mathematical Soc., 2017.

[3] Rajeev Motwani and Prabhakar Raghavan. *Randomized Algorithms*. Cambridge University Press, New York, NY, USA, 1995.