Generating Randomized Maximum Bipartite Matchings Via Swaps And Augmenting Paths Using Markov Chain Monte Carlo

Amit Joshi

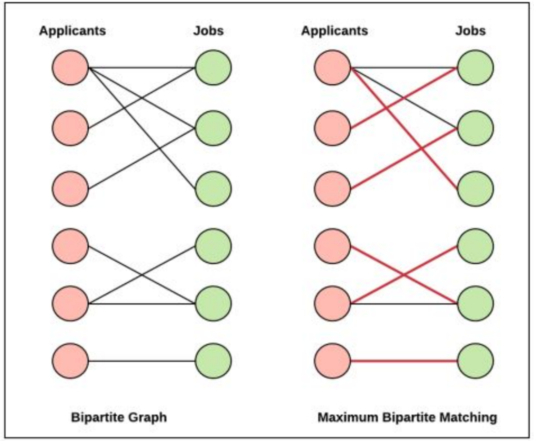
After saying that in one breath:

Abstract

- In this paper I explore generating random maximum bipartite matchings (MBPs) using Markov Chain Monte Carlo.
- I show that given an arbitrary MBP of a given graph, it is possible to generate a random MBP across the space of all MBPs of a graph.
- Markov Chain (M1) - generates each MBP with positive probability
- Markov Chain (M2) - generates each MBP with uniform positive probability
- Prove mixing time of M1 and M2 (how long generation takes)

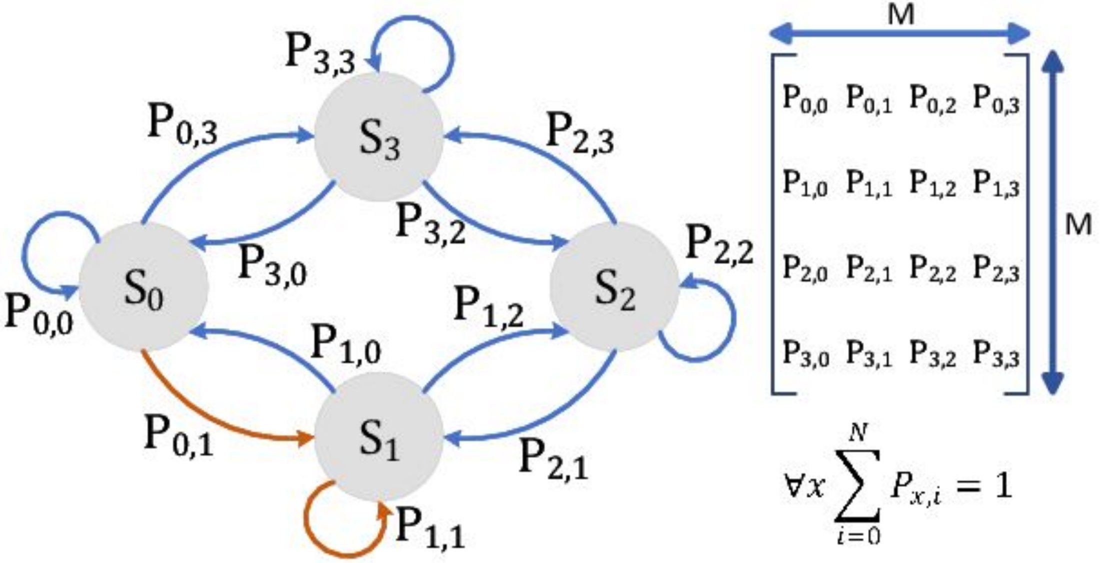
Goal

- Given a bipartite graph G (two columns), and a Maximum Bipartite Matching (MBP), generate a randomized MBP (preferably uniformly random over space of all MBPs)
- Maxflow generates an arbitrary MBP, but know nothing about the probability distribution over the MBPs generated
- Uniform Generation Uses:
    - Task assignment
    - Appoximating total number of MBPs
    - Monomer model in physics



Markov Chains

- Graph over states with transition probabilities
- States/Vertices represent all different MBPs, Transitions/Edges change between MBPs



Markov Chain Terminology

2.3.1 Markov Chains

Let Ω be the finite state space a Markov Chain operates over. Consider a stochastic process/sequence $(X_t)_{t=0}^{\infty}$. Conider the transition matrix P, of dimension $|\Omega| \times |\Omega|$. Note that:

$$\sum_{j\in\Omega} P_{i,j} = 1 \qquad\qquad \forall i \in \Omega$$

Process X is a Markov Chain if the probability that X transitions from i to j is independent of every transition prior to coming to i at that point. Mathematically, this is written as:

$$Pr[X_{t+1} = x_{t+1}|X_t = x_t, X_{t-1} = x_{t-1}, ..., X_1 = x_1] = Pr[X_{t+1} = x_{t+1}|X_t = x_t]$$

Let's define some relevant terms to get a sense of the vocabulary.

- Stationary: A distribution π is called stationary if $P\pi = \pi$, meaning that the transition matrix does not affect the distribution.
- Irreducible: $\forall i, j \in \Omega$, $\exists t$ such that $P^t_{i,j} > 0$. In other words, an irreducible markov chain is one where every state can eventually get to another state with some positive probability.
- Aperiodic: $\forall i \in \Omega$, $gcd(t, P^t_{i,i}) = 1$, where $P^t_{i,j}$ is the t-step transition probability from i to j. One way to show aperiodicity is the existence of self-loops in each state, meaning that with some positive probability a state i stays at itself after applying the transition. We can set $P = \frac{P+I}{2}$ so that each state stays at itself with at least $\frac{1}{2}$ probability.
- Time Reversibility: $\forall i, j$ we have that $\pi_i P_{ij} = \pi_j P_{ji}$. This means that the probability of being at i and transitioning from i to j is the same as the probability of probability of being at j and transitioning from j to i.

A Markov Chain has a *unique* stationary distribution (is ergodic) if it is irreducible and aperiodic by the Fundamental Theorem of Markov Chains (Mitzenmacher and Upfal, 2005).

2.3.3 Mixing Time

The mixing time τ or T_{mix} is the time the Markov Chain takes to converge to its stationary distribution. If a coupling argument is being used, it is the time needed for expected difference between the states of the coupling to be small. Once the difference between X and Y is small ($\mathbb{E}[d(X,Y)] < 1$), I consider X to be converged to the stationary distribution (uniformly random). Mixing Time is found by proving expected bounds on how $d(X,Y)$ changes per each transition. A Markov Chain that converges can either be rapidly (polynomially) mixing or slow (superpolynomially) mixing.

How can Markov Chains help our goal?

- An arbitrary matching means that we are at some node of the Markov Chain graph
- If the Markov Chain is ergodic, then no matter what node was started at, the unique probability distribution of nodes that we end up at after transitioning for Tmix is the same. This is called the stationary distribution.
- We can sample from the stationary distribution by making Tmix transitions, and seeing what state we end up at
    - If the stationary distribution is uniform, then this uniformly samples from the state space (set of all MBPs)

Related Work

- “On the Switch Markov Chain for Perfect Matchings”
  - Not ergodic for all graphs, such as for convex,biconvex
  - Transitions too simple since they are simple switches
- “Approximating the Permanent”
  - Reduces approximating the permanent (number of valid permutations of 1…n) to uniformly sampling perfect matchings (MBPs where all nodes are matched)
  - Broder’s Transitions involve deleting edges, so has a larger state space than just MBPs
  - Jerrum and Sinclair proved mixing time using Conductance

Markov Chain M1

- Generates each MBP with some positive probability
  - Needs to be Ergodic (irreducible and aperiodic). So, no matter initial state, can get to any other state (irreducible), and potentially stay there (aperiodicity/self-loops)
- Don't know much more about stationary distribution; depends on graph
  - Stationary distribution named "Graph-Dependent" distribution
  - M1 ergodic, so unique stationary distribution is the Graph-Dependent distribution
- Mainly used to define M2, Graph-Dependent stationary distribution not that useful

Markov Chain M1 transition P

1. Select vertex $(v, w) \in E$ uniformly at random
2. If $(v, w) \in E^i$ do nothing, or if v is already matched to w (aka $f(v) = w$) then also do nothing
3. If $(v, w) \notin E^i$ match via the protocol below:
   - If w unmatched, simply match v to w by removing $(v, f_i(v))$ which allows for the addition of (v, w) as now both v and w are unmatched (right-swap transition).
   - If v unmatched, simply match v to w by adding (v, w) via removing $(f_i(w), w)$ (left-swap transition). Note that both v and w could not have been unmatched because then we could add edge (v, w) to obtain a larger matching than i, contradicting the fact that it is an MBP.
   - If neither are unmatched initially, do the following procedure. Consider the subgraph $G' = (V', E')$ which is the same as G but does not contain the nodes v and w and as a result does not contain any edges touching v or w (every other node/edge remains the same). Now i without v and w be known as $i(G')$, called here a "subgraph MBP." Now, attempt to find a randomized left augmenting path in $i(G')$, which is guaranteed to be found as long as (v, w) exists in any MBP of G by Lemma 1 (see Proof of Lemmas section). Note that these edges of the augmenting paths are ordered in such a way that it starts with some $l \in L$ and ends with some $r \in R$. Denote the result of finding an augmenting path in $i(G')$ as $i'(G')$. If an augmenting path is not found, then do nothing in this transition, as it means (v, w) can't be in any MBP of G by Lemma 1.

Randomized Augmenting Paths - Residual Graphs

- Augmenting Paths:
  - Used to find more flow in maxflow and more matches
  - Called on the Residual Graph

2.2.1 Residual Graphs

Residual Graphs are directed graphs which are based upon the current matching, and are used in order to find an *augmenting path*, which, if found, will increase the number of edges in the current matching by 1 (Kleinberg and Tardos, 2014).

Definition 2.1 (Left Residual Graph). A left residual graph $Res_L(i)$ on graph G and matching i can be found by doing the following (Kleinberg and Tardos, 2014):

1. Add all nodes in V to $Res_L(i)$
2. For each undirected edge of the form $(l, r) \in E$ where $l \in L$ and $r \in R$, if $(l, r) \in E^i$ then add the directed edge (r, l) to $Res_L(i)$.
3. For each undirected edge of the form $(l, r) \in E$ where $l \in L$ and $r \in R$, if $(l, r) \notin E^i$ then add the directed edge (l, r) to $Res_L(i)$.

Randomized Augmenting Paths

2.2.2 Randomized Augmenting Path Algorithm

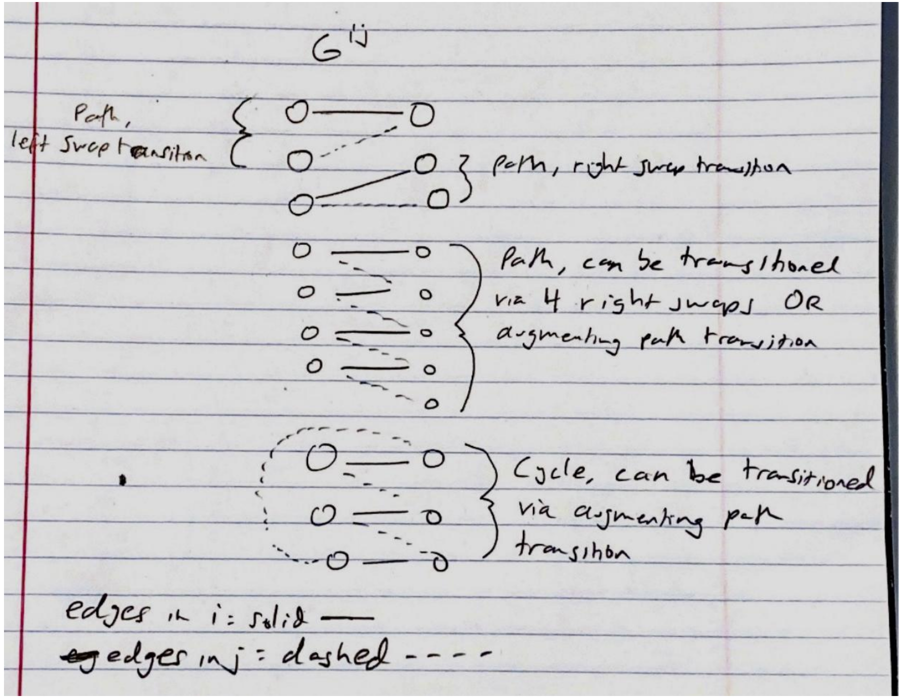
I modify the augmenting path algorithm from (Ibrahim 2020) to randomized. One is a randomized left augmenting path, which takes in a left residual graph and tries to find a path from a (given) unmatched node in L to an unmatched node in R. However, the modification is that the order in which the out-neighbors of all $l \in L$ are processed. This is done in order to ensure that each possible augmenting path has some positive probability of being taken.

Randomized Augmenting Paths

Algorithm 1 find-randomized-left-augmenting-path

```
procedure RANDOMIZED-DFS(Res(G),node,visited,match)
    if node ∈ visited then
        return false
    end if
    add node to visited
    randomize order of neighbors of node
    for neighbor nb of node do
        if nb unmatched OR randomized − dfs(match[nb]) then
            match[node] ← nb
            match[nb] ← node                    ▷ match node and nb to each other
            return true
        end if
    end for
end procedure
procedure FIND-RANDOMIZED-LEFT-AUGMENTING-PATH(G, a)    ▷ Parameters: Graph G,
matching a
    shuffle − order ← random permutation of [1, 2, . . . , |L|]
    visited ← ∅
    Res(G) ← left residual graph of G
    match ← a                    ▷ Mapping from nodes to their matches in a
    for unmatched l ∈ L ordered by shuffle − order do
        found ← dfs(Res(G), l, visited, match)
        if found = true then
            break
        end if
    end for
end procedure
```

Irreducibility/Aperiodicity => Ergodicity of P



Markov Chain M2

- Generates each MBP with positive uniform probability
  - Is ergodic (irreducible and aperiodic). So, no matter initial state, can get to any other state (irreducible), and potentially stay there (aperiodicity/self-loops)
- Stationary distribution is uniform
  - M2 ergodic, so unique stationary distribution is the uniform distribution
- Uniform stationary distribution is quite useful, can be used to sample each MBP with the same probability.
- Can be used to approximate total number of MBPs using a fully polynomial randomized approximation scheme, provided mixing time is polynomial

Markov Chain M2 transition hatP (1)

We can modify this Markov Chain $M1$'s transition matrix P so that it converges to the uniformly random distribution, where each MBP has equal probability of being generated. To do this, let's define a time-reversible (for the uniform distribution) Markov Chain, called $M2$. Let's denote it's transition matrix as $\hat{P}$. $\hat{P}$ is similar to P, but with some added self loops. In particular, it adds self-loops to ensure that $\hat{P}_{ij} = \hat{P}_{ji}$. This is done by setting $\hat{P}_{ij} = \hat{P}_{ji} = min(P_{ij}, P_{ji})$.

- If j is accessible from i via a swap transition:

  Without loss of generality, let's say that the swap transition was a right swap, so matching i has $(l, f_i(l))$ and matching j instead has $(l, f_j(l))$. For the transition from i to j to happen, the (v, w) chosen in step 1 must be $(l, f_j(l))$ which occurs with $\frac{1}{|E|} = P_{ij}$ probability. For the transition from j to i to happen, the (v, w) chosen in step 1 must be $(l, f_i(l))$ which also occurs with $\frac{1}{|E|} = P_{ji}$ probability. Hence, we have that $P_{ij} = P_{ji} = \hat{P}_{ij} = \hat{P}_{ji}$.

Markov Chain M2 transition hatP (2)

- If j is accessible from i via an augmenting path transition:
  - Calculate probability with which the augmenting path was taken
  - Calculate probability that each random edge was taken
  - $q(v)$ "viable" edges out of v (edges that can lead to an unmatched node on the right, calculate which edges are viable via DFS). Probability that any viable edge was taken is $\frac{1}{q(v)}$, as all edges were randomly shuffled in the $Res_L(i(G'))$.
  - As you go along augmenting path, multiply these probabilities together to find the probability of the augmenting path you are taking

Want to set the probability of this transition equal to that of its opposite transition. Can simply calculate probability of opposite augmenting path (Lemma 9) and set both of their hatP probabilities to the min of the two

Markov Chain M2 transition hatP (3)

- Each augmenting path from i to j has a unique opposite augmenting path from j to i
- Calculate probability of opposite augmenting path similarly to last slide
- Let’s say you are at state i and deciding whether to transition to state j …

Markov Chain M2 transition hatP (4)

We can see that P_{ij} may differ from P_{ji} but only in the case where j is accessible via an augmenting path transition due to the degrees of the vertices and the order in which the vertices are traversed. After setting $\hat{P}_{ij} = \hat{P}_{ji} = min(P_{ij}, P_{ji})$, one way to modify the transition defined for $\hat{P}$ is as follows. Once the augmenting path for transitioning from i is found to arrive at j, and the probability (P_{ij}) has been calculated via the find-randomized-augmenting-path-prob algorithm (which finds a randomized augmenting path and returns it and the probability with which it was taken), calculate P_{ji} using the calc-augmenting-path-prob which takes in an augmenting path and calculates the probability with which it is taken.

Case that $P_{ij} \leq P_{ji}$: transition to j. Hence we have that $\hat{P}_{ij} = P_{ij} = min(P_{ij}, P_{ji})$.

Case that $P_{ij} > P_{ji}$, since $\hat{P}_{ij}$ should be equal to $\hat{P}_{ji}$, after finding j, transition to j with probability $p = \frac{P_{ji}}{P_{ij}}$, and stay at i with probability $1 - p$ (in addition to the existing self-loops at i defined in P). Hence, we have that $\hat{P}_{ij} = P_{ij} * p = P_{ij} * \frac{P_{ji}}{P_{ij}} = P_{ji} = min(P_{ij}, P_{ji})$.

Uniform Distribution Stationary for M2 (1)

Theorem 2. *The uniformly random distribution is stationary for M2 and $\hat{P}$, and M2 will eventually converge to a uniformly random state.*

Proof. To see that the uniformly random distribution is stationary, consider any state i in Ω (consider any MBP of G) and consider all the states that are accessible via one transition. Given that i comes from a uniformly random distribution, $\pi_i = \frac{1}{|\Omega|}$. Let j be accessible from i via one transition. Since j also comes from the uniformly random distribution, $\pi_j = \frac{1}{|\Omega|}$.

So, for every probability of leaving state i to go to another state j under transition P, we have equal probability of coming to state i from state j under transition $\hat{P}$. Mathematically, we have that $\hat{P}_{ij} = \hat{P}_{ji}$ (by construction). As $\pi_i = \pi_j$, we have that $\hat{P}$ is also time-reversible.

Consider the definition of a stationary distribution, where the probability of transitioning to any state i is the same as the probability of leaving any state i.

Uniform Distribution Stationary for M2 (1)

$$
\begin{aligned}
&Pr[\text{leaving } i] && && (13)\\
&= \sum_{\forall j \text{ reachable from } i} \pi_i \hat{P}_{ij} && && (14)\\
&= \sum_{\forall j \text{ reachable from } i} \pi_j \hat{P}_{ij} && \text{Since } \pi_i = \pi_j \text{ since uniform} && (15)\\
&= \sum_{\forall j \text{ reachable from } i} \pi_j \hat{P}_{ji} && \text{By construction of } \hat{P} && (16)\\
&= Pr[\text{coming to } i] && && (17)
\end{aligned}
$$

Hence, we have that:

$$\hat{P}\pi = \pi \qquad \text{where } \pi \text{ is the stationary distribution in this case} \qquad (18)$$

Hence, the uniformly random distribution is stationary. By the same argument made in Theorem 1, there is only one stationary distribution for $M2$, I just proved that it is the uniformly random distribution, so eventually, $M2$ will arrive at (or converge to) a uniformly random state. □

Intermission

- We've learned how to transition between MBPs for both M1 (Graph-Dependent stationary distribution) and M2 (uniform stationary distribution)
- We know that both these Markov Chains do eventually converge to their respective unique stationary distributions
- Now, let's try to prove mixing time of these Markov Chains, or how long they take to converge to their stationary distributions
    - Once they have converged to their stationary distributions (with enough transitions), we can sample from them by simply looking at the state after at least a certain number of transitions

Coupling (1)

2.3.2 Coupling

Coupling is a technique to estimate the expected difference between a state and a state from the stationary distribution in a Markov Chain (Mitzenmacher and Upfal, 2005). Consider the tuple of stochastic processes $(X, Y) \in \Omega \times \Omega$. Let Y be a random state from the stationary distribution (which is unknown to us, as that is what we are trying to approximate), and let X be the current known state. Let X_t, Y_t be these states at timestep t. Note that X_0 is the initial state before applying P at all. I attempt to perform the same transition to both X and Y (by applying the same transition matrix P to both), so note that:

- If $X_t = Y_t$ then $X_{t+1} = Y_{t+1}$

Define a distance function $d(X, Y)$ which computes the difference between two states in Ω. The aim of coupling is to decrease the *expected* distance between X and Y. If the expected distance between X and Y decreases, then the Markov Chain is said to *converge*.

2.3.3 Mixing Time

The mixing time τ or T_{mix} is the time the Markov Chain takes to converge to its stationary distribution. If a coupling argument is being used, it is the time needed for expected difference between the states of the coupling to be small. Once the difference between X and Y is small ($\mathbb{E}[d(X, Y)] < 1$), I consider X to be converged to the stationary distribution (uniformly random). Mixing Time is found by proving expected bounds on how $d(X, Y)$ changes per each transition. A Markov Chain that converges can either be rapidly (polynomially) mixing or slow (superpolynomially) mixing.

Coupling (2)

- Coupling arguments for complex transitions have been shown to be hard to get working and/or wrong
- Tried coupling initially for M1
  - Defined distance function between X and Y that obeys triangle inequality
  - Was able to couple swaps easily and show that distance either stayed the same or decreased
  - Coupling augmenting paths was difficult, as I found examples where distance would increase after a transition
  - Perhaps in expectation augmenting path transitions would decrease distance, or maybe in a subclass of graphs like k-regular graphs with perfect matchings this is true - idk
  - Conjecturing the above bullet point, I found Tmix to be $|E|\ln(|V|/2)$

Conductance (1)

- Takes a different approach to bounding mixing time
- Measures how easily the Markov Chain gets stuck in a particular cut (2-partition) of the state space
  - If it leaves any cut easily (with high probability) ⇒ high conductance ⇒ low mixing time
  - Inverse holds true
- Was successfully able to get a provable mixing time for M2 this way :')
  - Perhaps due to not being able to bound values as well as I could have, this led to a superpolynomial mixing time. Next steps are to do more careful analysis to get better, hopefully polynomial bounds on mixing time. Can generate uniformly random MBPs!!
  - This automatically yields a fully polynomial time randomized approximation scheme to counting the number of MBPs of a graph

Conductance (2)

2.3.4 Conductance

The Conductance of a Markov Chain is a measure of how quickly the Markov Chain can leave cut $S \subset \Omega$. The conductance and mixing time are inversely correlated. If conductance is high, then no matter the initial state, the Markov Chain can transition out easily. Let's define some terms that we'll use later (Jerrum and Sinclair 1989).

- Relative Pointwise Distance (r.p.d.):

$$\Delta(t) = max_{i,j\in\Omega} \frac{|P_{ij}^t - \pi_j|}{\pi_j}$$

- Conductance of a time-reversible Markov Chain with transition matrix P, where $w_{ij} = \pi_i P_{ij} = \pi_j P_{ji}$

$$\Phi(P) = min_S \frac{\sum_{i\in S, j\notin S} w_{ij}}{\sum_{i\in S} \pi_i}$$

- Mixing Time τ:

$$\tau(\epsilon) = min\{t \in \mathbb{N} : \Delta(t') \leq \epsilon \text{ for all } t' > t$$

Conductance (3)

By a theorem and corollary in (Jerrum and Sinclair 1989), we have that for a time-reversible ergodic Markov Chain with $\forall i, P_{i,i} \geq \frac{1}{2}$, where π_{min} is the minimum stationary distribution probability:

$$\Delta(t) \leq \frac{(1 - \Phi(P)^2/2)^t}{\pi_{min}}$$

$$\tau(\epsilon) \leq \frac{2}{\Phi(P)^2}(ln(\pi_{min}^{-1}) + ln(\epsilon^{-1}))$$

As we have chosen ϵ to be 1 as an arbitrary value close to 0, we have that:

$$\tau(1) = \tau = T_{mix} = \frac{2}{\Phi(P)^2}(ln(\pi_{min}^{-1})) \tag{1}$$

Conductance (4)

5 Markov Chain Conductance

I included the coupling argument in the paper because it gives a better bound on the mixing time, but it requires that two conjectures hold true. To provably bound mixing time, for $M2$, I can use conductance instead.

Theorem 4. *The mixing time (T_{mix}) of $M2$ is $4|E|\gamma^{|V|}|V|ln(|V|)$, where γ is the maximum degree of any vertex.*

Proof. To define the conductance $\Phi(\hat{P})$, simply apply the formula. We want to find cut S that minimizes the conductance. First, let's rewrite the formula, for conductance:

$$\begin{aligned}
\Phi(\hat{P}) &= min_S \frac{\sum_{i \in S, j \notin S} w_{ij}}{\sum_{i \in S} \pi_i} && (38)\\
&= min_S \frac{\sum_{i \in S, j \notin S} \pi_i \hat{P}_{ij}}{\sum_{i \in S} \pi_i} && (39)\\
&= min_S \frac{\sum_{i \in S, j \notin S} \pi_i \hat{P}_{ij}}{|S|\pi_i} && \text{Since } \pi \text{ is uniform for } M2 \quad (40)\\
&\leq min_S \frac{|S|\pi_i \hat{P}_{ij}}{|S|\pi_i} && \text{At least } |S| \text{ pairings for } i, j \text{ in a cut } S \quad (41)\\
&\leq min \hat{P}_{ij} && (42)\\
&\leq max_{v \in |V|} \frac{1}{2|E|\delta(v)^{|V|}} && \text{By Thm 1, (12), } \hat{P} \text{ stays at itself with half probability} \quad (43)\\
&= \frac{1}{2|E|\gamma^{|V|}} && \gamma \text{ is maximum degree } \delta \text{ of any node} \quad (44)
\end{aligned}$$

Conductance (5) and final algorithm

Using equation (1), we know that:

$$
\begin{aligned}
\tau = T_{mix} &= \frac{2}{\Phi(\hat{P})^2}(ln(\pi_{min}^{-1})) && && (45)\\
&= 4|E|\gamma^{|V|}ln(|V|!) && \text{upper bound on number of MBPs} && (46)\\
&= 4|E|\gamma^{|V|}|V|ln(|V|) && \text{Stirling's approximation} && (47)
\end{aligned}
$$

$\square$

Finally, I can use this result to formally define my algorithm for generating a random maximum matching (from the GD distribution) for bipartite graphs, which is to simply apply the transition to an arbitrary MBP T_{mix} times.

Here is the algorithm I provide to generate/sample an MBP for Graph G from the GD distribution, named after me (the author of this paper) <3. To generate/sample an MBP for Graph G from the uniform distribution, simply replace P with $\hat{P}$.

Algorithm 4 amitjoshi24-algorithm

```
procedure AMITJOSHI24-ALGORITHM(G)                    ▷ Takes in G as input
    apply augmenting path algorithm to generate arbitrary MBP X
    for t ∈ T_mix do
        X ← P(X)                                      ▷ Apply P to X
    end for
    return X
end procedure
```

Conclusion/Contributions

- Show a Markov Chain irreducible over all MBPs and aperiodic, and shows that the state distribution is dependent on the graph G (GD distribution), not on the initial matching given to the algorithm. Furthermore, it shows how the stationary distribution gives positive probability to all MBPs.
- Show how to create a Markov Chain over all MBPs whose unique stationary distribution is uniform.
- I also use a conductance argument to bound the mixing times of the M2 Markov Chain. I also show how to construct a coupling argument between two MBPs, both among swap transitions and augmenting path transitions (using induction) assuming some conjectures hold true.
- Another contribution I have made is giving an algorithm that finds a simple path (path with no repeated nodes) from s to t in such a way that any simple path from s to t is chosen with positive probability and calculates the probability with which that path is chosen.

Future Work

- Future Research should look into determining what the stationary distribution (GD distribution) is.
- For \hat{P}, future research can also look into more efficient ways of seeing if the augmenting path transition be modified to lead to a uniform stationary distribution in the Markov Chain M2 (rather than the O(|V||E|^2) per augmenting path method of calculating probabilities).
- Future research should look into either proving the conjectures described in this paper or look more into other methods of bounding mixing time such as Conductance such as from (Jerrum and Sinclair) or canonical paths such as the one from (Dyer et al., 2017).
- Future work should look to construct a more careful conductance argument to show if/that M2 converges in polynomial time.
- Future work can also apply ideas from this paper to other problems.
- Change MBP to MBPM

Acknowledgements

- I would like to thank my family and friends for being wonderful!
- I would like to thank my advisor Dr. Shuchi Chawla for her great guidance this semester!
- I would like to thank all of you for being here!
- <3