Need for Convergence Speed: How do graph metrics influence the convergence speed of Markov chains?

Kalin Doychev
University of Twente

I. Introduction

Markov chains are systems that can describe the probabilities of going from one state to another. Usually, they are drawn using graphs or transition matrices. Markov chains have real-world applications that can help us predict the future based on recent data. Forecast predictions [6] and Google search results [3] are a few examples of where Markov chains are being used.

It has been researched how to converge Markov chains [1, 2] but there has not been any research on *how graph metrics influence the convergence speed of Markov chains*. Discrete-Time Markov Chains (DTMC) models are probabilistic systems, that eventually converge to an equilibrium state. The convergence speed measures how long it takes to reach this equilibrium. This is an important parameter because it also plays a role in the robustness of a system: if the convergence is fast, then the system will return to equilibrium quickly.

To conduct this research I will try to find and analyse real-world examples of DTMC. I will look at these matrices to try and generate matrices that are similar enough to real-world data. I need to generate some data because I was not able to find a lot of publicly available Markov chains and I needed a larger dataset to work with. For all of these matrices, I need to identify and extract important metrics that could influence the convergence speed. Finally, I will use Pearson Correlation Coefficient (PCC) analysis to identify the correlation between the convergence time and the different metrics.

I will first explain some definitions and ground truths about the Markov chains. Then, I will go through the data collection/generation for the experiment. Finally, I will explain what conclusions I was able to make based on my experiments. I found out that the graph metrics with the largest correlation were the diameter, the radius, and the Second-Largest Eigenvalue (SLE). Surprisingly, graph size turned out to play no role in the convergence speed. Overall, this study suggests that the number of states of a Markov chain is not important for its convergence speed, a more important metric to take into account is the longest hop count between any two states and the SLE.

As we will see, this study suggests that some graph-theoretic metrics can have a strong influence on the convergence speed and some do not.

II. Definitions & Theorems

A. Markov Chains

Markov chains are stochastic models that show the possible sequence of events. In these Markov chains, the probability of going from one state to the other only depends on the previous state - Markov property [7]. More formally: a Markov chain consists of a finite number of possible states I (also known as the state space). A Markov chain can be represented by a $|I|\times|I| = n\times n$ transition matrix P that shows the probability of going from one state to another. Each row of the transition matrix sums up to 1 ($i \in 1,\ldots,n; \sum_{j=1}^{n} p_{ij} = 1$, where p_{ij} is the probability of going from state i to state j, and n is the number of states: $|I| = n$)

$$P = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n1} & p_{n2} & \cdots & p_{nn} \end{bmatrix} \tag{1}$$

If $x^{(n)}$ is a probability vector representing the state of the system at time t, then the state of the system at time $t+1$ is given by $x^{(n+1)} = x^{(n)}P$. Then if we have the probability distributions $x^{(0)}, x^{(1)}, \ldots, x^{(n)}, \ldots$ at some time t the distribution $x^{(t)}$ will become a stationary vector (Definition II.1). This is when the Markov chain has converged. This stationary vector is independent of the original input vector. Each column in this final vector represents the average probability of being in that state. The sum of the vector is equal to 1, as the total sum of all probabilities should be 100% (law of total probability).

Definition II.1. A distribution x is called a stationary distribution of a Markov chain P if $xP = x$.

Not all Markov chains can converge. We need to establish whether a Markov chain is convergent or not. A sufficient condition is ergodicity, which is defined as follows:

Definition II.2 (Ergodicity)**.** An ergodic graph is a graph that is both aperiodic and irreducible.

A graph is irreducible if from any state you can reach any other state in any number of transitions. In the case of Markov chains, this means that there is a larger than zero chance to get from any state to any other state. A more formal definition would be:

Definition II.3 (Irreducibility)**.** A Markov chain is irreducible if for all states $i, j \in I$, there exists a $t \geq 0$ such that $p_{ij}^t > 0$, where if p_{ij} is the distribution at time 0, then p_{ij}^t is $p_{ij}P^t$.

The Greatest Common Divisor (GCD) of the time periods, in which the probability of going from state i back to state i is larger than 0, is the period of that state. A state is considered

aperiodic when its period is 1. If all states in a Markov chain are aperiodic then the Markov chain itself is aperiodic. This can be described more formally in the following definitions:

Definition II.4. Let $T(i) = \{t \geq 1 : p_{ii}^t > 0\}$ be the set of all time steps for which a Markov chain can start and end in a state i. Then the period of state i is $\gcd T(i)$.

Definition II.5. If P is irreducible, then the period of all states is equal, or $\gcd T(i) = \gcd T(j), \forall i, j \in I$.

Definition II.6. An irreducible Markov chain is called aperiodic if its period is equal to 1, or equivalently, $\gcd T(i) = 1, \forall x \in I$.

For many Markov chains, convergence can occur for all initial distributions $x^{(0)}$ and the stationary distribution that is reached is the same for all of them. In other words, the stationary distribution is unique. Definition II.7 says that to have a unique stationary distribution the Markov chain must be irreducible [5]. This would be satisfied if the graph is ergodic.

Definition II.7. If P is irreducible, then it has a unique stationary distribution x with $x(i) > 0, \forall i \in I$.

Because convergence can take too long to finish I will consider the process done even without explicitly reaching a stationary vector but a vector within some small error margin ε. I measure the distance between the distribution I have calculated so far $x^{(n)}$, and the stationary distribution $x^{stationary}$ and the first iteration where $\|x^{(n)} - x^{stationary}\| = \sum_{i=1}^{n} |x_i^{(n)} - x_i^{stationary}| < \varepsilon$ is satisfied I consider the process complete and measure it in seconds and the number of iterations. The choice of ε is arbitrary and in my case, I chose $\varepsilon = 1 * 10^{-4}$. If the time required for my process to finish was too short I would have chosen a smaller ε, as this would have helped me visualize the results better. If it was taking too long I would have increased ε.

B. Graph Metrics

Converting from a matrix P to a graph is trivial. From the original matrix P that is of size $n \times n$ we know that the graph will have n states. Then each transition is from row to column, for example, if row 2 of a 3×3 matrix has the values 0, 0.25, and 0.75 in that order we know that the probability of state 2 going to state 1 is 0%, the probability of going back to state 2 is 25%, and the probability of going to state 3 is 75%. So, to go from a matrix to a graph we need to traverse the whole matrix and convert the values in the matrix to transitions.

From the matrices, I extracted the number of nodes and edges, diameter, radius, average deg, max in- and out-deg, and the Second-Largest Eigenvalue (SLE). The number of nodes is the number of states because the states are represented using nodes in the graph version of a Markov chain. The edges are the transitions between the nodes. These two metrics are easy to get from any matrix representation - one dimension of the matrix is the number of states and the number of transitions is the amount of non-zero values in the matrix. The eccentricity of a matrix is the longest hop count from any state to any other state [4]. The diameter and the radius are the longest and the shortest eccentricity, respectively. The average in- and out-degree are the same and so I put them into a single variable called "average degree". It is the sum of all edges divided by the number of states. The max in- and out-degree are the largest in-degree for all of the states and the largest out-degree for all of the states.

I chose the SLE because the largest eigenvalue is always 1 and that would not be an interesting value to consider. Also, the SLE should be the slowest eigenvalue to converge. A lot of the eigenvalues were complex numbers so I normalized them using $\sqrt{a^2 + b^2}$, where $a + i * b$ would be the complex number. Then I compared every graph metric to the time it took the matrix to converge by using a scatter plot. I also made a comparison to the number of iterations it took each matrix to converge. Finally, I calculated the Pearson correlation coefficient between every graph-theoretic metric and the convergence speed/number of iterations.

C. Pearson Correlation Coefficient

To analyze the results, I used Pearson Correlation Coefficient. It is a way to analyze the linear relation between two variables. The PCC is a value between -1 and 1 [10]. If the value is -1, it means that with the increase of the variable x, the variable y decreases. If it is 0, it means that there is no linear correlation between the two variables. If the value is 1 with the increase of variable x, variable y also increases.

There is a theorem by Wielandt [9] to verify if a matrix is ergodic:

Theorem 1. *Markov chain is ergodic if and only if all elements of P^m are positive, where $m = (n-1)^2 + 1$, n is the number of states, and P is the transition matrix.*

III. Data Collection & Generation

I want my research to be as useful as it could be for real applications and for that my data analysis should be on as much real-world data as possible. Unfortunately, there is not enough public data on real-world Markov chains. For that reason, I decided to generate matrices myself. The idea is to make a wide variety of matrices with many different metrics so that I can cover any possible case.

A. Real-World Data

I wanted to use real-world DTMC examples as my data from [8], but I was restricted in two ways. First, I was not able to use some of the Markov chains because they required too much memory and time to convert from a file to a Python3 matrix. For example, the largest one is the "Bluetooth Device Discovery" dataset which consists of over 3.4 billion states. The other restriction was that the smaller dataset consisted of matrices that were not ergodic. That meant that I cannot use them for this research.

B. Data Generation

I generated two sets of matrices from size 10×10 up to 200×200. For each matrix size, I also generated 10 matrices,

to generate some randomness. The first set of matrices is generated with a certain percentage of the row containing 0s. For example, I would start the first of the ten matrices of size 100×100 with 25% of the row as 0s. Each new one would have more 0s than the last one up to 75% of the row is 0s. I did this because the first matrices that I generated did not require any number of 0s on a row and this caused most of my matrices to be complete. This means that the complete matrices of the same size had the same values for all graph-theoretic metrics and as such did not provide much information to analyse.

The second set of matrices was generated by first connecting all nodes in a circle. This helped me not have to worry if the generated matrices are irreducible as each node can reach any other node. Then on each row, I added a random number of edges between 1 and 10. This helped me get more diverse values for the radius and diameter. In the previous set, the values for these two metrics were constants (all values were either 2 or 3). All of the data and code can be found at https://github.com/KalinD/ResearchProject.

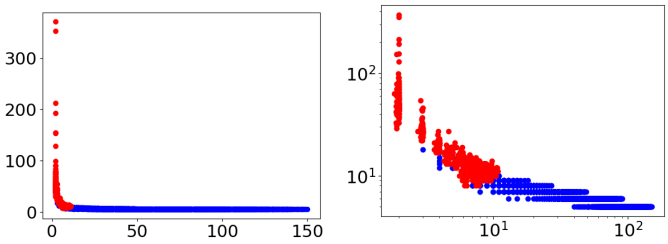
IV. Results

To visualize the results I plotted them using scatter plots. Figure 1 (1k) and (1l) shows that with the increase in the number of nodes, the convergence time almost does not change. This is further shown after checking the PCC for this metric: -0.04 for the number of iterations. A PCC value that is close to 0 means that there is no linear relation between the variables.

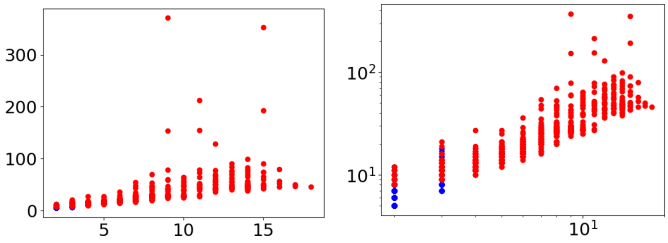
The following metrics turned out to have a high positive (if x increases so does y) correlation - the diameter, the radius, and the normalised Second-Largest Eigenvalue. I expected the SLE because this should be the slowest eigenvalue to converge, and so the larger it is the slower the convergence should be. In this case, this means that more iterations were expected. The PCC for these metrics is between 0.69 and 0.75 and they are the only graph-theoretic metrics with a positive correlation. The high PCC of the radius and diameter can be explained by what they represent. The longer hop count from any state to any other state means that it will take longer for the convergence to reach all states from any random initial state.

The rest of the metrics - edges, average degree, and max in- and out-degree, seem to have a negative (if x increases, then y decreases) correlation. This makes sense as the number of edges is increased so are the other three metrics. Also, if the number of edges is increased the radius and diameter will decrease. This explains the negative correlation compared to the positive one from the radius and diameter. The negative correlation of these metrics is not that big (the largest negative correlation is -0.39). So, they do not seem to have a big influence on the number of iterations.

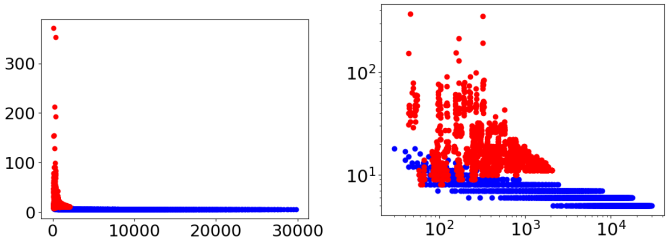
Table I contains the PCC for all graph metrics that I have analysed. The graphs in Fig. 1 visualize some of the relations between the graph-theoretic metrics. For Fig. 1 the y-axis is the number of iterations it took for the matrix to converge and the x-axis is the metric corresponding to the label under the graph.



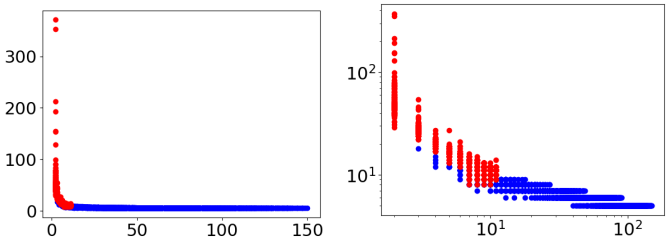
(a) Average Degree (b) Log Scale Average Degree



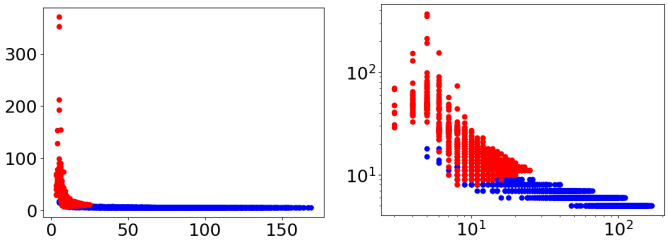
(c) Diameter (d) Log Scale Diameter



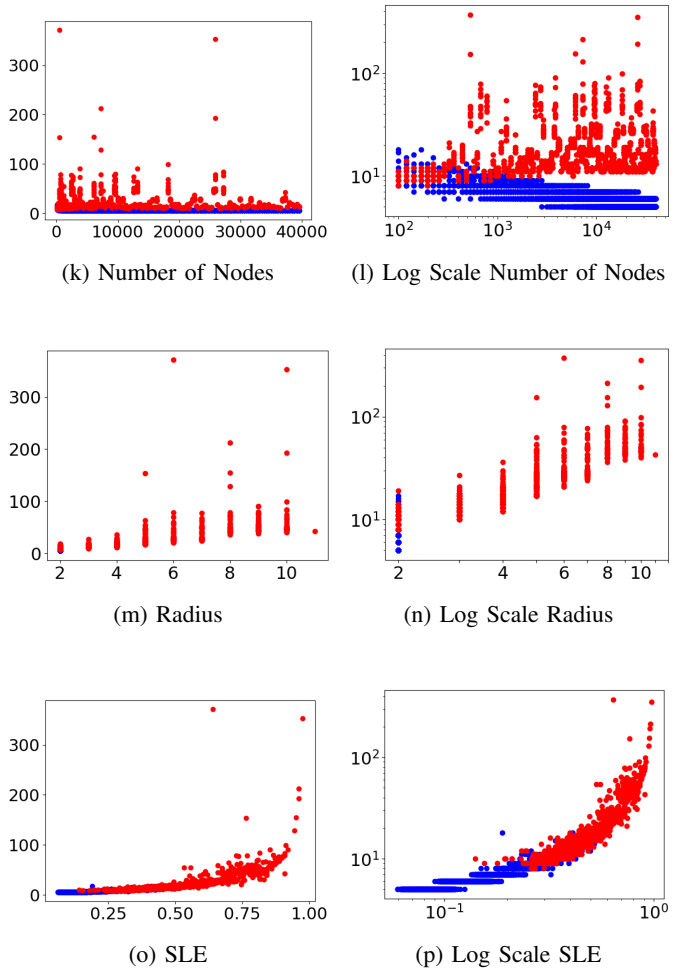
(e) Number of Edges (f) Log Scale Edges



(g) Max In-Degree (h) Log Scale Max In-Degree



(i) Max Out-Degree (j) Log Scale Max Out-Degree



(k) Number of Nodes (l) Log Scale Number of Nodes

(m) Radius (n) Log Scale Radius

(o) SLE (p) Log Scale SLE

Fig. 1: Relation Between Graph Metrics and Number of Iterations to Converge (the blue dots are from the first dataset and the red ones are from the second dataset)

For the graphs and analysis, I decided to focus on the number of iterations rather than the convergence time because the number of iterations is a more consistent variable. Convergence speed can depend on other factors such as the processor, whether there are other processes active, etc. Even running the same matrix on the same machine produced differences in timing. Meanwhile, the number of iterations for a specific matrix will be the same on any machine under any circumstances.

TABLE I: Pearson Correlation Coefficients

	# Nodes	# Edges	Diameter	Radius
Conv. Speed	0.657474	0.927393	−0.258087	−0.220808
# Iterations	−0.040890	−0.305138	0.751935	0.738339
	Avrg Deg	Max In-Deg	Max Out-Deg	Norm. SLE
Conv. Speed	0.850529	0.850400	0.855699	−0.471324
# Iterations	−0.380435	−0.381875	−0.392037	0.690957

V. Conclusion

In conclusion, when wanting to return fast to an equilibrium state for Markov chains a few important graph-theoretic metrics to consider are the diameter, the radius, and the Second-Largest Eigenvalue. The smaller these three metrics are, the faster the system can return to an equilibrium state.

VI. Future Work

A downside to my research is that I only used generated matrices. Real-world Markov chains could have slightly different properties and results. To further develop this paper one might find and analyze Markov chains that were generated from real-world events and compare their findings to mine. Another option is to try and determine the parameter values of benchmark DTMCs to generate similar DTMCs that are also ergodic.

References

[1] Fredrik Backåker. *The Google Markov Chain: convergence speed and eigenvalues*. June 2012. URL: https://uu.diva-portal.org/smash/get/diva2:536076/FULLTEXT01.pdf.

[2] Brandon Franzke and Bart Kosko. "Noise can speed convergence in Markov chains". In: *Phys. Rev. E* 84 (4 Oct. 2011), p. 041112. DOI: 10.1103/PhysRevE.84.041112. URL: https://link.aps.org/doi/10.1103/PhysRevE.84.041112.

[3] Ravi Teja Gundimeda. "Google Page Rank and Markov Chains - Analytics Vidhya". In: (Dec. 15, 2021). URL: https://medium.com/analytics-vidhya/google-page-rank-and-markov-chains-d65717b98f9c.

[4] Javier M Hernández and Piet Van Mieghem. "Classification of graph metrics". Nov. 2011. URL: https://www.nas.ewi.tudelft.nl/people/Piet/papers/TUDreport20111111_MetricList.pdf.

[5] David Levin and Yuval Peres. *Markov Chains and Mixing Times*. 2nd Revised edition. American Mathematical Society, Oct. 31, 2017.

[6] Sourabh Mehta. "5 real-world use cases of the Markov chains". In: (May 3, 2022). URL: https://analyticsindiamag.com/5-real-world-use-cases-of-the-markov-chains/.

[7] J. R. Norris. *Markov Chains*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 1997. DOI: 10.1017/CBO9780511810633.

[8] *QComp - Quantitative Verification Benchmark Set - qcomp.org*. URL: https://qcomp.org/benchmarks/.

[9] H. Wielandt. *Unzerlegbare, Nicht Negativen Matrizen.* Vol. 52. Mathematische Zeitschrift. 1950, pp. 642–648.

[10] Charles Zaiontz. "Basic Concepts of Correlation". In: (June 28, 2022). URL: https://real-statistics.com/correlation/basic-concepts-correlation/.