Discrete-Time Markov Chains – Applications & the ALOHA case

Examples of DTMCs*

$\{Y_n\}_{n\geq 1}$: set of positive integer, independent random variables that are identically distributed with $\Pr[Y=k]=a_k$

Examples of DTMCs

(a) $X_n = Y_n$ (b) $X_n = \max[Y_1, Y_2, Y_3, \ldots, Y_n]$ (c) $X_n = \sum_{k=1}^{n} Y_k$

(a) $X_n = Y_n$

$$P = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 & \cdots \\ a_1 & a_2 & a_3 & a_4 & \cdots \\ a_1 & a_2 & a_3 & a_4 & \cdots \\ a_1 & a_2 & a_3 & a_4 & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix}$$

All rows are identical and $\Pr[X_{n+1}=j|X_n=i]=a_j$ shows that the states X_{n+1} and X_n are independent from each other.

* Performance Analysis of Communications Networks and Systems (Piet Van Mieghem), Chap. 11

Examples of DTMCs

$\{Y_n\}_{n\geq 1}$: set of positive integer, independent random variables that are identically distributed with $\Pr[Y=k]=a_k$

(b) $X_n = \max[Y_1, Y_2, Y_3, \ldots, Y_n]$

$X_{n+1} = \max[X_n, Y_{n+1}]$ reflects the Markov property

For $j < i$ $\Pr[X_{n+1}=j|X_n=i]=0$

For $j > i$ $\Pr[X_{n+1}=j|X_n=i] = \Pr[Y_{n+1}=j] = a_j$

For $j = i$

$$\Pr[X_{n+1}=j|X_n=i] = \Pr[Y_{n+1}\leq j] = \sum_{k=1}^{j}\Pr[Y_{n+1}=k] = \sum_{k=1}^{j} a_k = A_j$$

$$P = \begin{bmatrix} A_1 & a_2 & a_3 & a_4 & \cdots \\ 0 & A_2 & a_3 & a_4 & \cdots \\ 0 & 0 & A_3 & a_4 & \cdots \\ 0 & 0 & 0 & A_4 & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix}$$

Examples of DTMCs

$\{Y_n\}_{n\geq 1}$: set of positive integer, independent random variables that are identically distributed with $\Pr[Y=k]=a_k$

(c) $X_n = \sum_{k=1}^{n} Y_k$

$X_{n+1} = X_n + Y_{n+1}$ reflects the Markov property

For $j \leq i$ $\Pr[X_{n+1}=j|X_n=i]=0$

For $j > i$ $\Pr[X_{n+1}=j|X_n=i] = \Pr[X_n+Y_{n+1}=j|X_n=i]$

$= \Pr[Y_{n+1}=j-i] = a_{j-i}$

$$P = \begin{bmatrix} 0 & a_1 & a_2 & a_3 & \cdots \\ 0 & 0 & a_1 & a_2 & \cdots \\ 0 & 0 & 0 & a_1 & \cdots \\ 0 & 0 & 0 & 0 & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix}$$

Slotted Aloha

N nodes that communicate via a shared channel using slotted Aloha

Time is slotted, packets are of the same size

A node transmits a newly arrived packet in the next timeslot

If two nodes transmit at the same timeslot (collision) packets must be retransmitted

Backlogged nodes (nodes with packets to be retransmitted) wait for some random number of timeslots before retransmitting

Packet arrivals at a node form a Poisson process with mean rate λ/N, where λ is the overall arrival rate at the network of N nodes

We ignore queuing of packets at a node (newly arrived packets are discarded if there is a packet to be retransmitted)
We assume, for simplicity, that p_r is the probability that a node retransmits in the next time slot

Slotted Aloha

Slotted Aloha constitutes a DTMC with $X_k \varepsilon \{0, 1, 2, \ldots\}$, where

- state j counts the number of backlogged nodes
- k refers to the k-th timeslot

Each of the j backlogged nodes retransmits a packet in the next time slot with probability p_r

Each of the N-j unbacklogged nodes transmits a packet in the next time slot iff a packet arrives in the current timeslot which occurs with probability $p_a = \Pr[A > 0] = 1 - \Pr[A = 0]$

For Poisson arrival process $p_a = 1 - \exp\left(-\frac{\lambda}{N}\right)$

The probability that *n* backlogged nodes in state *j* retransmit in the next time slot is binomially distributed

$$b_n(j) = \binom{j}{n} p_r^n (1 - p_r)^{j-n}$$

Slotted Aloha

Similarly, the probability that n unbacklogged nodes in state j transmit in the next time slot is

$$u_n(j) = \binom{N-j}{n} p_a^n (1-p_a)^{N-j-n}$$

A packet is transmitted successfully iff

(a) 1 new arrival and 0 backlogged packet is transmitted, or

(b) 0 new arrival and 1 backlogged packet is transmitted

The probability of successful transmission in state j and per slot is

$$p_s(j) = u_1(j) b_0(j) + u_0(j) b_1(j)$$

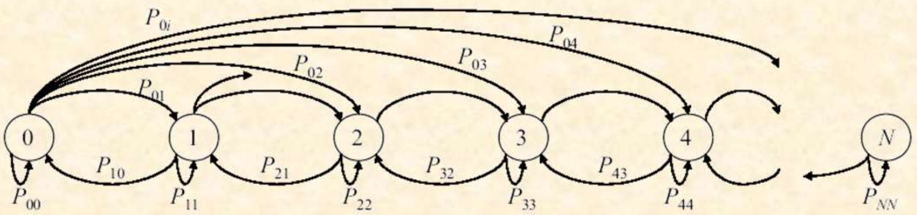
The transition probability $P_{j,j+m}$ equals (see explanation in next slide)

$$P_{j,j+m} = \begin{cases} u_m(j) & 2 \le m \le N-j \\ u_1(j)\left(1-b_0(j)\right) & m = 1 \\ u_1(j) b_0(j) + u_0(j)\left(1-b_1(j)\right) & m = 0 \\ u_0(j) b_1(j) & m = -1 \end{cases}$$

Slotted Aloha

State j (j backlogged nodes) jumps to

- state j-1 if there are no new packets and one retransmission
- state j if
  - there is 1 new arrival and there are no retransmissions or
  - there are no new arrivals and none or more than 1 retransmissions
- state j+1 if there is 1 new arrival (from a non-backlogged node) and at least 1 retransmission (then there are surely collisions)
- state j + m if m new packets arrive from m different non-backlogged nodes, which always causes collisions for m>1

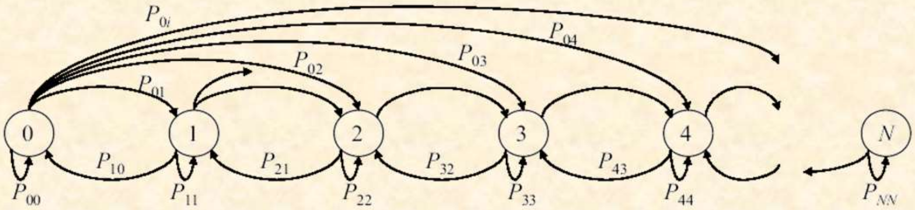


Slotted Aloha

The transition probability $P_{j,j+m}$ equals

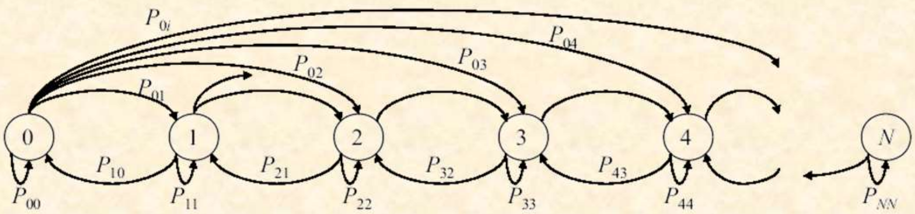
$$P_{j,j+m} = \begin{cases} u_m(j) & 2 \le m \le N-j \\ u_1(j)\,(1-b_0\,(j)) & m=1 \\ u_1\,(j)\,b_0\,(j) + u_0\,(j)\,(1-b_1\,(j)) & m=0 \\ u_0\,(j)\,b_1\,(j) & m=-1 \end{cases}$$

(it is zero for all other m < -1)



Slotted Aloha

$$P = \begin{bmatrix} P_{00} & P_{01} & P_{02} & \cdots & \cdots & P_{0N} \\ P_{10} & P_{11} & P_{12} & \cdots & \cdots & P_{1N} \\ 0 & P_{21} & P_{22} & \cdots & \cdots & P_{2N} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & P_{N-1,N-2} & P_{N-1;N-1} & P_{N-1;N} \\ 0 & 0 & \cdots & 0 & P_{N;N-1} & P_{NN} \end{bmatrix}$$



Slotted Aloha

For small N the steady-state equations can be solved
When N grows, slotted Aloha turns out to be unstable
When N→∞, the steady-state vector π does not exist ($\lim_{N\to\infty} \pi = 0$)

The expected change in backlog per time slot is:

$$E\left[X_{k+1} - X_k | X_k = j\right] = (N-j)\,p_a - p_s(j)$$

drift = expected number of new arrivals - expected number of successful transmissions

Since $p_s(j) \le 1$ and $p_a = 1 - \exp\left(-\frac{\lambda}{N}\right)$ $\qquad \lim_{N\to\infty} E\left[X_{k+1} - X_k | X_k = j\right] = \infty$

The drift tends to infinity, which means that, on average, the number of backlogged nodes increases unboundedly and suggests (but does not prove) that the Markov chain is transient for N→∞

Efficiency of slotted Aloha

For small arrival probability p_a and small retransmission probability p_r, the probability of successful transmission and of no transmission in state j is well approximated by

$$p_s(j) \simeq t(j)\, e^{-t(j)}$$

$$p_{no}(j) \simeq e^{-t(j)}$$

where $t(j) = (N-j)\,p_a + j p_r$ is the expected number of arrivals and retransmissions in state j (=total rate of transmission attempts in state j) and is also called the offered traffic G

for small p_a and p_r, the analysis shows that $p_s(j)$ and $p_{no}(j)$ are closely approximated in terms of a Poisson random variable with rate t(j)

(see next for derivations)

Efficiency of slotted Aloha

The probability of a successful transmission in state j is

$$\begin{aligned} p_s(j) &= u_1(j)b_0(j) + u_0(j)b_1(j) \\ &= (N-j)p_a(1-p_a)^{N-j-1}(1-p_r)^j + jp_r(1-p_r)^{j-1}(1-p_a)^{N-j} \\ &= \left[\frac{(N-j)p_a}{1-p_a} + \frac{jp_r}{1-p_r}\right](1-p_a)^{N-j}(1-p_r)^j \end{aligned}$$

For small arrival probability p_a and small retransmission probability p_r, the probability of successful transmission in state j can be approximated by using the Taylor expansions of $(1-x)^\alpha = e^{\alpha \ln(1-x)} = e^{-\alpha x}(1+o(1))$ and $\frac{x}{1-x} = x + o(x^2)$ as

$$\begin{aligned} p_s(j) &= \left[(N-j)p_a + jp_r + o\left(p_a^2 + p_r^2\right)\right] e^{-[(N-j)p_a + jp_r]}(1+o(1)) \\ &= [(N-j)p_a + jp_r]\exp(-[(N-j)p_a + jp_r])(1+o(1)) \end{aligned}$$

Similarly, the probability that no packet is transmitted in state j equals

$$\begin{aligned} p_{no}(j) &= u_0(j)b_0(j) = (1-p_r)^j(1-p_a)^{N-j} \\ &= \exp(-[(N-j)p_a + jp_r])(1+o(1)) \end{aligned}$$

Efficiency of slotted Aloha

$$p_s(j) \simeq t(j)e^{-t(j)}$$

$$p_{no}(j) \simeq e^{-t(j)}$$

$$t(j) = (N-j)p_a + jp_r \qquad \text{(offered traffic G)}$$

p_s can be interpreted as the throughput $S_{SAloha} = G e^{-G}$, maximized if $G = 1$

The efficiency η_{SAloha} of slotted Aloha with N>>1 is defined as the maximum fraction of time during which packets are transmitted successfully which is $e^{-1} = 36\%$

Efficiency of pure Aloha

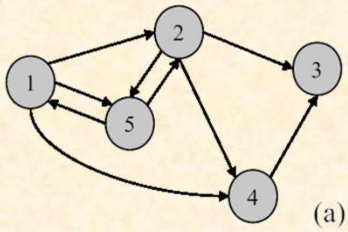
Pure Aloha: the nodes can start transmitting at arbitrary times
Performs half as efficiently as slotted Aloha with $\eta_{PAloha} = 18\%$
A transmitted packet at t is successful if no other is sent in (t-1, t+1) which is equal to two timeslots and thus $\eta_{PAloha} = 1/2\ \eta_{SAloha}$

In pure Aloha, $p_{no}(j) \simeq e^{-2t(j)}$ because in (t-1, t+1) the expected number of arrivals and retransmissions is twice that in slotted Aloha

The throughput S roughly equals the total rate of transmission attempts G (which is the same as in slotted Aloha) multiplied by $p_{no}(j) \simeq e^{-2t(j)}$, hence, $S_{\text{PAloha}} = Ge^{-2G}$

Ranking of webpages

Websearch engines apply a ranking criterion to sort the list of pages related to a query
PageRank (the hyperlink-based ranking system used by Google) exploits the power of discrete Markov theory
Markov model of the web: directed graph with N nodes
- Each node in the webgraph represents a webpage and
- the directed edges represent hyperlinks



(a)

$$P = \begin{bmatrix} 0 & P_{12} & 0 & P_{14} & P_{15} \\ 0 & 0 & P_{23} & P_{24} & P_{25} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & P_{43} & 0 & 0 \\ P_{51} & P_{52} & 0 & 0 & 0 \end{bmatrix}$$

Ranking of webpages

Assumption:

importance of a webpage ~ number of times that this page is visited

Consider a DTMC with transition probability matrix P that corresponds to the adjacency matrix of the webgraph

- P_{ij} is the probability of moving from webpage i (state i) to webpage j (state j) in one time step
- The component $s_i[k]$ of the state vector s[k] denotes the probability that at time k the webpage i is visited

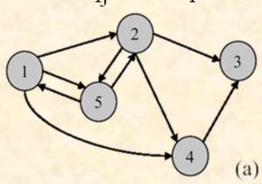
The long run mean fraction of time that webpage i is visited equals the steady-state probability π_i of the Markov chain

This probability π_i is the ranking measure of the importance of webpage i used in Google

Ranking of webpages

Uniformity assumption (if web usage information is not available): given we are on webpage i, any hyperlink on that webpage has equal probability to be clicked on

Thus, $P_{ij} = 1/d_i$ where the d_i is the number of hyperlinks on page i



(a)

$$P = \begin{bmatrix} 0 & \frac{1}{3} & 0 & \frac{1}{3} & \frac{1}{3} \\ 0 & 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \end{bmatrix} \quad \bar{P} = \begin{bmatrix} 0 & \frac{1}{3} & 0 & \frac{1}{3} & \frac{1}{3} \\ 0 & 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \\ 0 & 0 & 1 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \end{bmatrix}$$

Problem: a node may not contain outlinks (dangling node)

$\Rightarrow$the corresponding row in P has only zero elements (P non-stochastic)

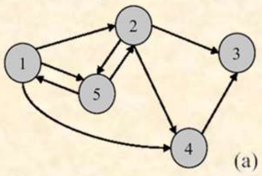
Solution: each zero row is replaced by a non-zero row vector v^T that obeys $\|v\|_1 = v^T u = 1$ where $u^T = [1 \ 1 \ \cdots \ 1]$

simplest case: (uniformity) $v^T = \frac{u^T}{N}$

Ranking of webpages

Uniformity assumption (if web usage information is not available):
given we are on webpage i, any hyperlink on that webpage has equal

When solely adopting the adjacency matrix of the webgraph as underlying structure of P, we cannot assure that P is stochastic



$$P = \begin{bmatrix} 0 & \frac{1}{3} & 0 & \frac{1}{3} & \frac{1}{3} \\ 0 & 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \end{bmatrix} \quad \bar{P} = \begin{bmatrix} 0 & \frac{1}{3} & 0 & \frac{1}{3} & \frac{1}{3} \\ 0 & 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \\ 0 & 0 & 1 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \end{bmatrix}$$

Problem: a node may not contain outlinks (dangling node)

$\Rightarrow$the corresponding row in P has only zero elements

Solution: each zero row is replaced by a non-zero row vector v^T that obeys $\|v\|_1 = v^T u = 1$ where $u^T = [1 \; 1 \; \cdots \; 1]$

simplest case: (uniformity) $v^T = \frac{u^T}{N}$

Ranking of webpages

The existence of a steady-state vector π must be ensured

If the Markov chain is irreducible, the steady-state vector exists

In an irreducible Markov chain any state is reachable from any other

By its very nature, the WWW leads almost surely to a reducible MC

Brin and Page have proposed (to create an irreducible matrix)

$$\bar{\bar{P}} = \alpha \bar{P} + (1 - \alpha) u v^T$$

where 0<a<1 and v is a probability vector (each component of v is non-zero in order to guarantee reachability)

Brin and Page have called v^T the *personalization* vector

For $v^T = \left[\frac{1}{16} \;\; \frac{4}{16} \;\; \frac{6}{16} \;\; \frac{4}{16} \;\; \frac{1}{16} \right]$ and $\alpha = \frac{4}{5}$

$$\bar{P} = \begin{bmatrix} 0 & \frac{1}{3} & 0 & \frac{1}{3} & \frac{1}{3} \\ 0 & 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \\ 0 & 0 & 1 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \end{bmatrix} \quad \bar{\bar{P}} = \begin{bmatrix} \frac{1}{80} & \frac{19}{60} & \frac{3}{40} & \frac{19}{60} & \frac{67}{240} \\ \frac{1}{80} & \frac{1}{20} & \frac{41}{120} & \frac{19}{60} & \frac{67}{240} \\ \frac{1}{16} & \frac{1}{4} & \frac{3}{8} & \frac{1}{4} & \frac{1}{16} \\ \frac{1}{80} & \frac{1}{20} & \frac{7}{8} & \frac{1}{20} & \frac{1}{80} \\ \frac{33}{80} & \frac{9}{20} & \frac{3}{40} & \frac{1}{20} & \frac{1}{80} \end{bmatrix}$$

Computation of the PageRank steady-state vector

A more effective (avoiding creating a dense matrix as before) way to implement the described idea is to define a special vector r whose component $r_j = 1$ if row j in P is a zero-row or node j is dangling node

Then, $\bar{P} = P + rv^T$ is a rank-one update of P and so is $\bar{\bar{P}}$ because

$$\bar{\bar{P}} = \alpha\left(P + rv^T\right) + (1-\alpha)u.v^T = \alpha P + (\alpha r + (1-\alpha)u)\, v^T$$

Brin and Page propose to compute the steady-state vector from the following (instead of solving the standard set of linear eqns, complex)

$$\pi = \lim_{k\to\infty} s[k]$$

Specifically, for any starting vector s[0] (usually s[0] = $\frac{u^T}{N}$), we iterate the equation s[k+1]=s[k] P m-times and choose m sufficiently large such that $\|s[m] - \pi\| \le \epsilon$ where ϵ is a prescribed tolerance

Computation of the PageRank steady-state vector

It holds $s\,[k+1] = s[k]\bar{\bar{P}} = s[k]\left(\alpha P + (\alpha r + (1-\alpha)u)\, v^T\right)$

Since $s[k]u = 1$, we find

$$* \quad s\,[k+1] = \alpha s[k]P + (\alpha s[k]r + (1-\alpha))\, v^T$$

Thus, only the product of s[k] with the (extremely) sparse matrix P needs to be computed and $\bar{P}$ and $\bar{\bar{P}}$ are never formed nor stored

For any personalization vector, the second largest eigenvalue of $\bar{\bar{P}}$ is $\alpha \lambda_2$, where λ_2 is the second largest eigenvalue of $\bar{P}$

Brin and Page report that only 50 to 100 iterations of * for α= 0.85 are sufficient

A fast convergence is found for small α, but then the characteristics of the webgraph are suppressed

Langville and Meyer proposed to introduce one dummy node that is connected to all other nodes and to which all other nodes are connected to ensure overall reachability. Such approach changes the webgraph less