The Art of Modeling

Renato Lo Cigno

Simulation and Performance Evaluation 2014-15

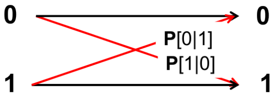
Given a system a model is a mathematical law (function) that describe some of it properties as a function of one or more free parameters

- The Digital Binary Communication Channel (DBCC)
- The bit error probability given the noise on the channel . . . but
  - What is the noise? What modulation is used? What is the "channel"
- The speed of a car given the power (force, torque) yield by the engine
  - What about frictions, air, gears, . . .
- The number and distribution of arcs in a graph (network) given the "arc generation law"
- The completion time of a job on a specific computer
- The time spent in a bank given the operation I have to do (and the other customers?)

- Deterministic
  - Simple (or complex) equations, e.g., $a = \frac{F}{m}$, $v(t) = \int_t \frac{F(t)}{m}\, dt$
- Stochastic
  - Random Variables ...
- Static (does not depend on time)
  - Deterministic, Stochastic
- Dynamic (depends on time)
  - Deterministic, Stochastic (differential equations, random processes)

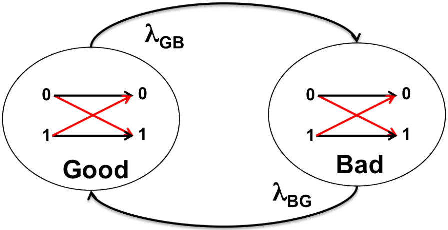
- The system

- The model



- Characterization only requires **P**[1|0] and **P**[0|1]
- But who give us these parameters?

- DBCC can be easily extended with a Markov Chain to model more complex, non-stationary scenarios



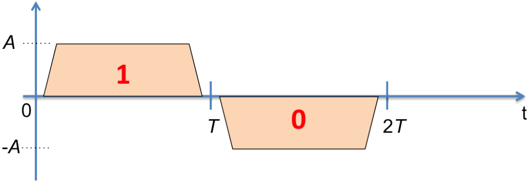
- Now we have more parameters to define:

$$\mathbf{P}_G[1|0]; \mathbf{P}_G[0|1]; \quad \mathbf{P}_B[1|0]; \mathbf{P}_B[0|1]; \quad \lambda_{B,G}; \lambda_{B,G}$$

- Yet we do not know how to set these parameters

- A pretty simple concept, we need it to tune our DBCCs
  ...which is what we use as Computer Scientist to design protocols, networks, distributed applications
- It depends on many characteristics of the transmission system
  - Modulation scheme (amplitude, phase, frequency, No. of bits/symbol, ...)
  - The transmission means (copper, fiber, wireless, central frequency, ...)
  - Receiver characteristics
  - Presence and characteristics of error correcting codes
  - ...
- **Disclaimer:** this is not meant to be a rigorous analysis of Communication Theory!

- PAM: Pulse Amplitude Modulation:
  1 $\rightarrow$ positive amplitude pulse; 0 $\rightarrow$ negative amplitude pulse
- We use a "reasonable" real waveform $w(t)$ (similar to a square wave) of duration T



- The transmitted energy per bit is

$$E_b^T = \int_0^T A w(t)\, dt = A$$

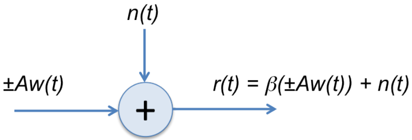
if we assume $w(t)$ energy equal to 1

- Maximum Likelihood Receiver: integrates the received signal over the bit period T and decides based on sign of the integral
  - In practice it evaluates what is the sign of the waveform based on the amount of energy present in the received signal
  - Details are too technical to unfurl here, but in practice we have

$$b_i = \int_{(i-1)T}^{iT} r(t)\, dt$$

where b_i is the i-th bit we decide has been received (1 if $b_i > 0$, 0 if $b_i < 0$), $r(t)$ is the signal received and $w(t)$ is the base waveform

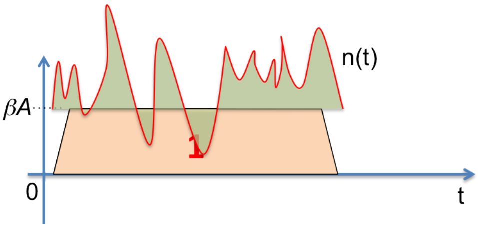
- And the Channel?
- We assume the simplest possible model: only Additive, White (uncorrelated), Gaussian Noise with 0 mean and $\sigma^2 = N_0$; N_0 is called 'spectral noise density'
- An the inevitable attenuation β



- The received useful energy per bit is

$$E_b = \int_0^T \beta A w(t)\, dt = \beta A$$

$$r(t) = \pm\beta A w(t) + n(t)$$



Normalizing so that $t = (i-1)T + t$

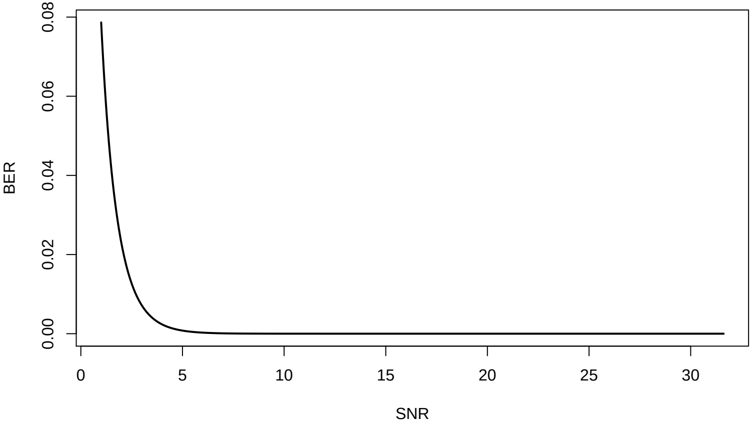
$$b_i = \int_0^T \pm\beta A w(t) + n(t)\, dt$$

Thanks to the central limit theorem b_i is a Gaussian RV with mean $\pm\beta A = \pm E_b$ and standard deviation $\sigma^2 = N_0$

- Computing the BER reduces to evaluate the probability that a b_i has the wrong sign compared to the transmitted signal, i.e., that a Gaussian RV with $\sigma = N_0$ is larger than $\sqrt{E_b}$

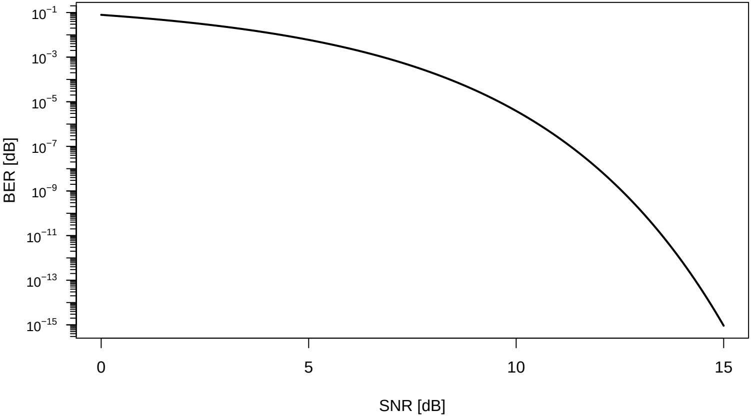
$$\text{BER} = \frac{1}{\sqrt{2\pi}} \int_{\sqrt{E_b/N_0}}^{\infty} e^{-x^2}\, dx = \frac{1}{2}\text{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right)$$

We only have to find a good plot to show its behavior ...



... not this one ...

BER AWGN, PAM – log-log scale



. . . much better!

- What do we have to take into account to get a reasonable model?
- The engine power for sure . . . is it enough?
- That in the end is what we mostly know about our car engine . . .
- What is the torque? And what about frictions and air drag?
- Does the gear have influence? And the weight of the car?
- **Let's make some models**

- **Disclaimer:** these are simplifications of Vehicular Technology for Computer Scientists . . .

We want to model the behavior of a vehicle when we go full throttle. We start from high school physics . . .

$$\begin{cases} \dot{x} = v \\ \dot{v} = a \end{cases} \qquad (1)$$

where x is the position, v is the speed, a is the acceleration

Now we consider three different models for car's acceleration

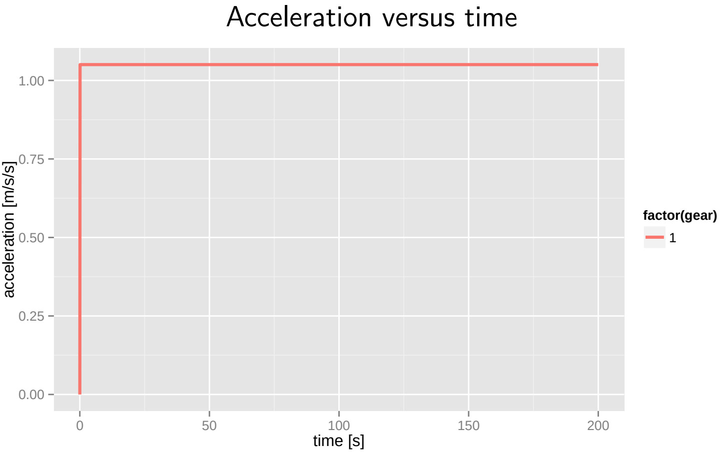
This model assumes constant force (so constant torque) with no RPM limit.

$$\begin{cases} \dot{x} = v \\ \dot{v} = \frac{F_{\text{eng}_1}(r_{\text{gear}})}{m} \end{cases} \tag{2}$$

where F_{eng} is the force generated by the engine, m is the mass of the car, and r_{gear} is the transmission gear ratio. F_{eng} is computed depending on the engine and vehicle parameters. In particular,

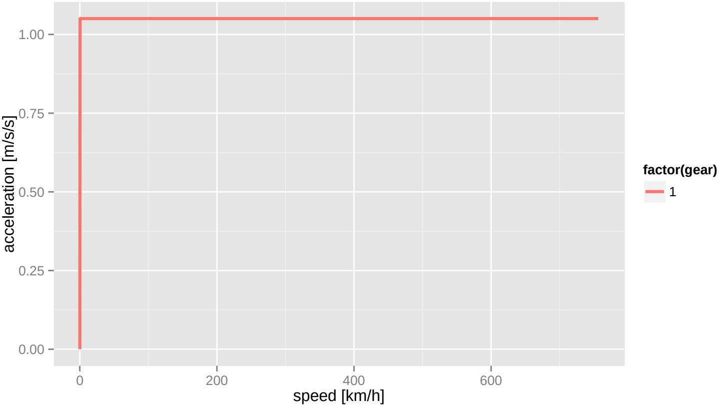
$$F_{\text{eng}_1}(r_{\text{gear}}) = \frac{T \cdot r_{\text{gear}}}{d_{\text{wheel}} \cdot \pi}. \tag{3}$$

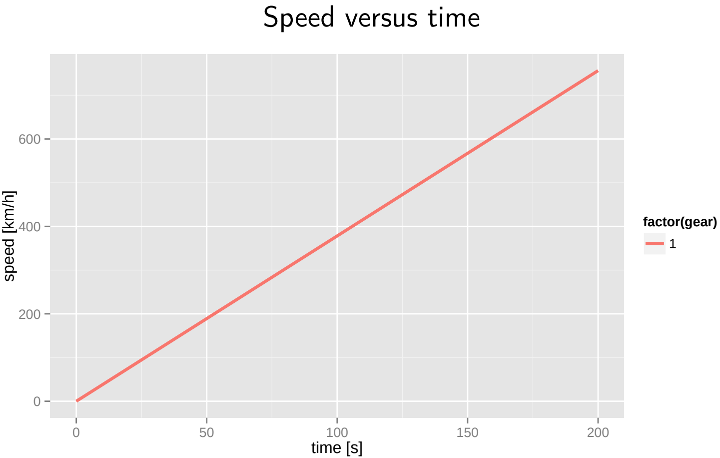
T is the torque in Nm, d_{wheel} is the tracting wheels diameter in m. We assume only one gear, and engine RPM limit . . .



Acceleration versus time
acceleration [m/s/s]
1.00
0.75
0.50
0.25
0.00
0
50
100
150
200
time [s]
factor(gear)
1

Acceleration versus speed




Speed versus time
speed [km/h]
600
400
200
0
0
50
100
150
200
time [s]
factor(gear)
1

This model assumes constant torque, but a maximum number of engine RPM. When we reach this number of RPM, we change gear. In this example, we have four gears.
First we define a function which gives us the engine RPM as function of the speed:

$$RPM(v) = \frac{60 \cdot r_{\text{gear}} \cdot v}{d_{\text{wheel}} \cdot \pi} \tag{4}$$

$$r_{\text{gear}}(v) = \begin{cases} r_1 & \text{if} \quad 0 \le v < v_1 \\ r_2 & \text{if} \quad v_1 \le v < v_2 \\ r_3 & \text{if} \quad v_2 \le v < v_3 \\ r_4 & \text{if} \quad v_3 \le v \end{cases} \tag{5}$$

$$F_{\text{eng}} = \frac{T \cdot r_{\text{gear}}(v)}{d_{\text{wheel}} \cdot \pi}. \tag{6}$$

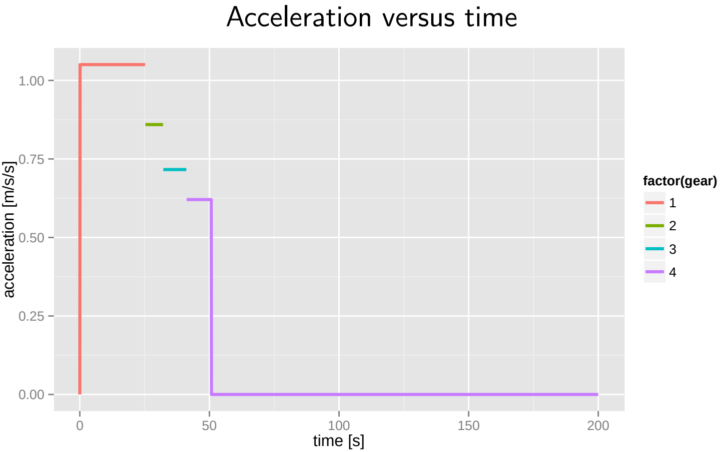
$$F_{\text{eng}_2}(v) = \begin{cases} F_{\text{eng}_1}(r_1) & \text{if} \quad 0 \leq v < v_1 \\ F_{\text{eng}_1}(r_2) & \text{if} \quad v_1 \leq v < v_2 \\ F_{\text{eng}_1}(r_3) & \text{if} \quad v_2 \leq v < v_3 \\ F_{\text{eng}_1}(r_4) & \text{if} \quad v_3 \leq v < v_4 \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

To compute v_i, we can use the following formula which computes the speed of the vehicle given the RPMs and the gear ratio r_i:

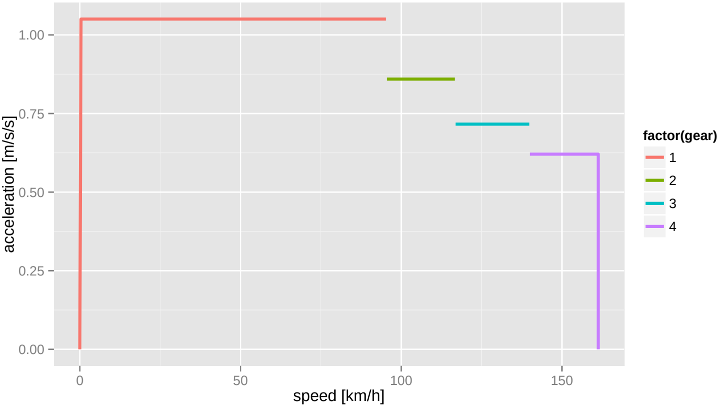
$$v_i = \frac{d_{\text{wheel}} \cdot \pi}{60 \cdot r_i \cdot RPM_{\max}} \tag{8}$$

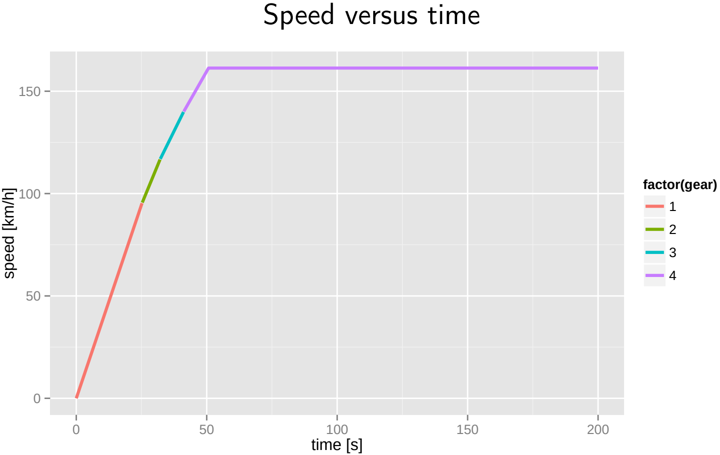
The model now becomes

$$\begin{cases} \dot{x} = v \\ \dot{v} = \dfrac{F_{\mathrm{eng}_2}(v)}{m} \end{cases} \qquad (9)$$


Acceleration versus time
1.00
0.75
0.50
0.25
0.00
acceleration [m/s/s]
0
50
100
150
200
time [s]
factor(gear)
1
2
3
4

Acceleration versus speed




Speech versus time
speed [km/h]
time [s]
150
100
50
0
0
50
100
150
200
factor(gear)
1
2
3
4

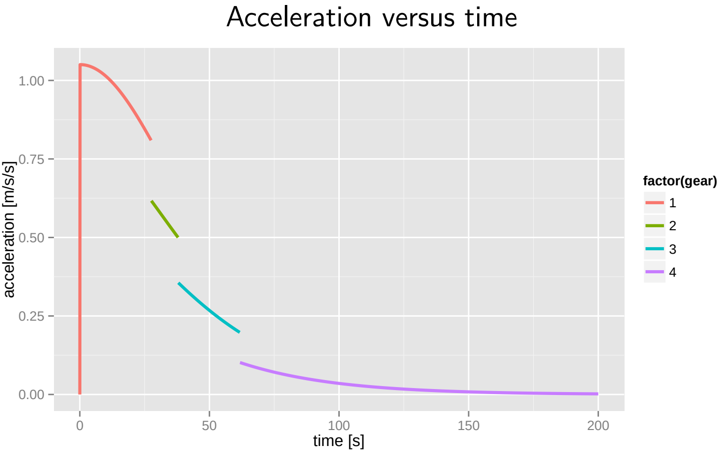
This model assumes the limited RPM engine model, gears, plus air friction

$$\begin{cases} \dot{x} = v \\ \dot{v} = \frac{F_{\text{eng}_2}(v) - F_{\text{air}}(v)}{m} \end{cases} \tag{10}$$

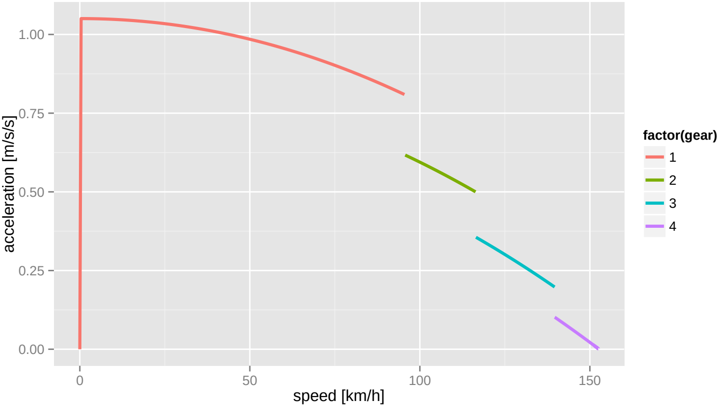
where $F_{\text{air}}(v)$ is the force due to air friction and is defined as

$$F_{\text{air}}(v) = \frac{1}{2} c_{\text{air}} A_L \rho_a v^2 \tag{11}$$

where c_{air} is the drag coefficient, A_L is the maximum vehicle cross section area, ρ_a is the air density, and v the vehicle's speed

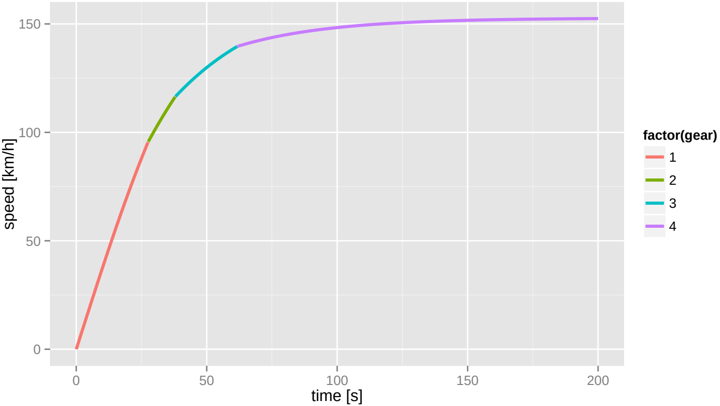

Acceleration versus time
acceleration [m/s/s]
1.00
0.75
0.50
0.25
0.00
0
50
100
150
200
time [s]
factor(gear)
1
2
3
4

Acceleration versus speed



Model 3

Speed versus time



- The BER model is a static stochastic model
- The Car model is a dynamic (differential equations) deterministic model
- The DBCC model is stochastic, and either static or dynamic depending if there is a single error probability model or if we use a Markov Chain to embed different models ...

- Markov Models are one of the most powerful (yet simple) technique to design models

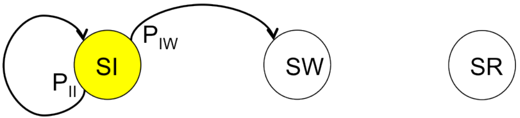
- We have seen that a Markov Chain (DT or CT) is a simple time-varying SP
- It is a suitable means to model dynamic systems with non-deterministic behavior
- We have to identify a set of variables that represent the state of the system
- We have to identify a set of transition probabilities (rates) that govern the evolution of the system ...
- ... We have to find a method to solve it

- Time is slotted: natural modeling with DT
- Note that slots need not be of the same length, they can depend, e.g., on the state
- The protocol can only be in 3 states:
  - **I**dle: there is nothing to transmit, you can sleep
  - **W**ait: one packet is in transmission, waiting for the acknowledgement
  - **R**e-transmit: a packet has not been ack-ed, we have to re-transmit it
  - $S = \{I, W, R\}$

The States of the Model

- States alone are not enough
- We need the transition probabilities

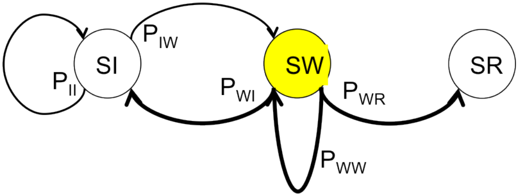
Transition probabilities from State I



$\mathbf{P}_{II}$ Probability that when Idle no packets arrive

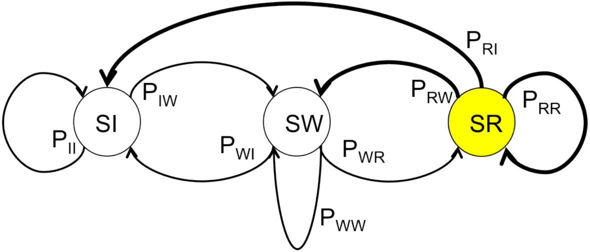
$\mathbf{P}_{IW}$ Probability that when Idle one or more packets arrive

Transition probabilities from State *W*



$\mathbf{P}_{WI}$	Probability that the transmission is successful and there are no other packets to transmit
$\mathbf{P}_{WR}$	Probability that the transmission fails the packet must be re-transmitted
$\mathbf{P}_{WW}$	Probability that the transmission is successful and there are other packets to transmit

Transition probabilities from State *R*

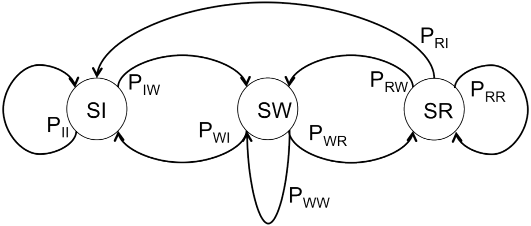


P $_{RI}$ Probability that the re-transmission is successful and there are no other packets to transmit

P $_{RW}$ Probability that the transmission is successful and there are other packets to transmit

P $_{RR}$ Probability that the transmission fails the packet must be re-transmitted (again)

DTMC of the Model



- The slot times include the transmission time and its Ack
- We have external events (arrival of packets from the upper protocol layers that drive the model
- We have complex transitions that account for external arrivals and loss/error probabilities
- We have self-transitions that tells us, e.g., the distribution of the number of re-transmissions per packet

- We know that the evolution of a Markov Chain depends only on the state . . . and we assume a time-homogeneous DTMC to make things simpler
- States are numerable, so without loss of generality we can set $S = \{0, 1, 2, 3, 4, \dots\}$
- p_{jk} denotes the transition probability from state j to state k
- The matrix

$$P = [p_{ij}] = \begin{bmatrix} p_{00} & p_{01} & p_{02} & \cdot & \cdot \\ p_{10} & p_{11} & p_{12} & \cdot & \cdot \\ p_{20} & p_{21} & p_{22} & \cdot & \cdot \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

completely characterized a DTMC

- P is a *stochastic matrix*, i.e., it has the following properties

$$0 \leq p_{ij} \leq 1, \quad \forall i, j \in S$$

$$\sum_{j \in S} p_{ij} = 1, \quad \forall i \in S$$

- P elements are all non-negative
- P rows **must** sum to 1 for the theorem of total probability (i.e., the sum of the probabilities of disjoint events covering S must be 1)
- Representing a DTMC with P or with the state diagram is exactly the same

- Let $\mathbf{p}(n) = [p_0(n), p_1(n), \ldots, p_j(n), \ldots]$ be the vector of the probability of being in a given state at step n
- Clearly $\sum_{i \in S} p_i(n) = 1, \quad \forall n$
- It is immediate to see that

$$\mathbf{p}(n+1) = \mathbf{p}(n)P$$

- If we have an initial state distribution (e.g., $\mathbf{p}(0) = [1, 0, 0, 0, \ldots]$) with a simple recursion we have

$$\mathbf{p}(1) = \mathbf{p}(0)P; \;\; \mathbf{p}(2) = \mathbf{p}(1)P = \mathbf{p}(0)P^2; \;\; \ldots; \;\; \mathbf{p}(n) = \mathbf{p}(0)P^n$$

- Another way to see the evolution of a DTMC is computing the transition probabilities in n steps $\forall n$
- This imply computing the sum of the probabilities of all possible paths to go from state i to state j in exactly n-steps
- For $n = 1$ this is trivially p_{ij} entry of the transition matrix P
- Recall that for a time-homogeneous DTMC by definition

$$p_{ij}(n) = \mathbf{P}[X_{m+n} = j | X_m = i], \quad \forall m$$

so we can drop the dependence on m

$$p_{ij}(n) = \mathbf{P}[X_n = j | X_0 = i]$$

- The equation above tells us that we have to compute all the conditional probabilities of going from state i to state k in h steps times the probability of going from state k to state j in $n-h$ stpes
- Formally

$$p_{ij}(n) = \sum_{h=1}^{n} \sum_{k \in S} p_{ik}(h) p_{kj}(n-h)$$

which are the Chapman-Kolmogorov equations that can be rewritten in the simple matrix form of

$$P(n) = P \cdot P(n-1) = P^n$$

in case of homogeneous DTMC

- We can ask the question if it is possible (and meaningful) to compute

$$\mathbf{v} = [v_0, v_1, \ldots v_i, \ldots]$$

where

$$v_i = \lim_{n\to\infty} p_i(n)$$

- As $\mathbf{p}(n) = \mathbf{p}(0)P^n$, it equivalent to ask if $\lim_{n\to\infty} P^n$ exists and is meaningful
- If these limits exists and are meaningful, as P is a stochastic matrix and $\mathbf{v}$ is a stochastic vector $\mathbf{v}$ is the left eigenvector of P associated to the eigenvalue $\lambda = 1$ and can be found as

$$\mathbf{v} = \mathbf{v}P$$

- Every vector **v**that satisfies

$$\mathbf{v} = \mathbf{v}P; \quad \sum_{i \in S} v_i = 1$$

is called a *stationary distribution* (or probability) of the DTMC

- If **v** exists, it is unique and independent from the initial state **p**(0) of the DCMC, then it is called the *steady-state* of the DTMC

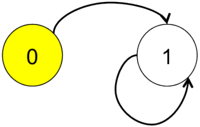
- Question: Under which conditions the steady-state of a DTMC exists?

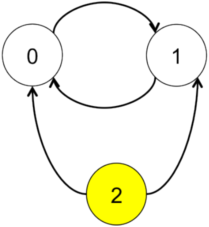
Definition: Transient State

- A state i is said to be transient if there is a positive probability that the process will never return to i after leaving it
- Formally this is equivalent to state that

$$\lim_{n \to \infty} p_{ji}(n) = 0; \quad \forall j \in S$$

Transient States (yellow)





Definition: Recurrent State

- A non-transient state is said recurrent
- A state is recurrent if yhe probability of visiting i after leaving it for $n \to \infty$ is 1

$$\sum_{n=1}^{\infty} p_{ii}(n) = \infty$$

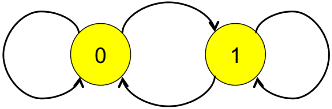
Definition: Recurrent State

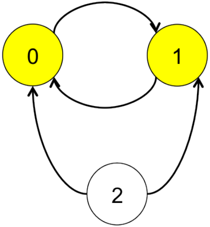
- Let $f_{ij}(n)$ be the conditional probability that the first visit to j after leaving i occurs in exactly n steps
- Then the probability of ever visiting j from i is

$$f_{ij} = \sum_{n=1}^{\infty} f_{ij}(n)$$

- A state is recurrent if $f_{ii} = 1$; if $f_{ii} \leq 1$ it is transient

Recurrent States (yellow)





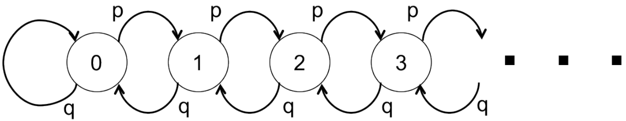
Definition: Recurrent Positive State

- For a recurrent state *i* it is interesting to know the distribution of the recurrence time, i.e., after how many steps the DTMC returns to *i* after leaving it
- We define the *mean recurrence time* of state *i*

$$\mu_i = \sum_{n=1}^{\infty} n f_{ii}(n)$$

- A state is said *recurrent positive* (non=null) if $\mu_i \leq \infty$
- A state is said *recurrent null* if $\mu_i = \infty$

Recurrent Null/Positive States

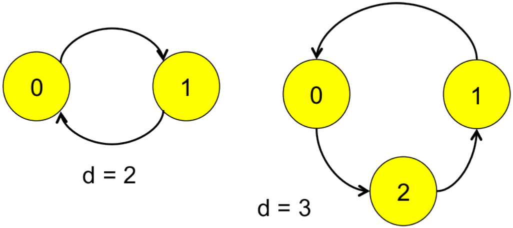


- If $p < q$ all states are recurrent positive
- If $p \geq$ all states are recurrent null

Definition: Periodic State

- Let d_i be the greatest common divisor of the set of positive integers n such that $p_{ii}(n) > 0$
- A state is said *periodic* if $d_i > 1$; the value d_i is called the period
- A state is said *aperiodic* if $d_i = 1$

Periodic States (yellow)

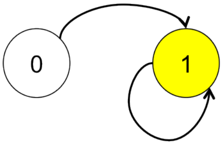


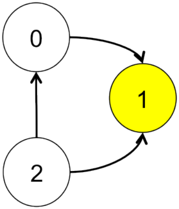
Definition: Absorbing and Communicating States

- A state i is said *absorbing* if $p_{ii} = 1$
- Once the DTMC enters i it will never leave it
- This notion can be extended to a set of states

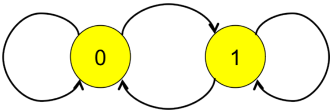
- Given two states i and j they are said *communicating* if directed paths exist from i to j and viceversa $p_{ij}(n) > 0$ for some n and $p_{ji}(m) > 0$ for some m

Absorbing States (yellow)

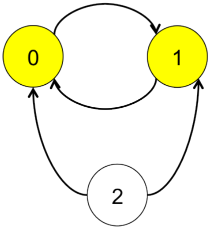




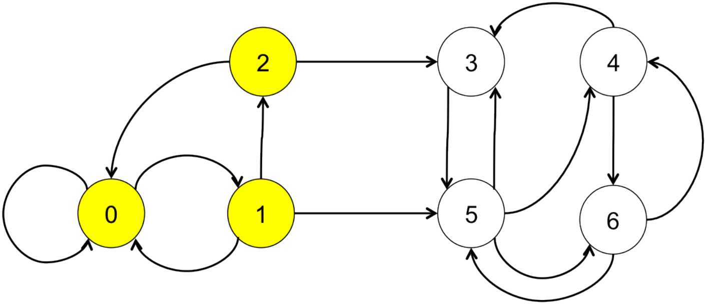
Communicating States
(yellow)



Non Communicating States
(yellow, 0 and 1 do not communicate with 2)



DTMC with Transient States (yellow) and a set of absorbing states (white) that do not communicate with the Transient ones

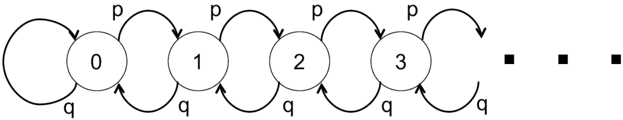


- A MC (not only DT) is said **Irreducible** if every state i is reachable from any other state j in finite time: $\forall i, j \in S$ there exists $n \geq 1$ such that $p_{ij}(n) > 0$
- An irreducible MC does not have Transient or recurrent-null states, i.e., they are all recurrent positive states
- All states in an irreducible MC are of the same type: Periodic or Aperiodic

Any Irreducible Aperiodic Markov Chain admits a Steady-State that can be computed (for DTMCs) as

$$\mathbf{v} = \mathbf{v}P$$

- If $|S|$ is infinite, then the steady state can be found only if P has some special structure that allows a recursive solution
- Example: DT Birth-Death Process with $p < q$



$$P = \begin{bmatrix} q & p & 0 & 0 & 0 & \cdot & \cdot & \cdot \\ q & 0 & p & 0 & 0 & \cdot & \cdot & \cdot \\ 0 & q & 0 & p & 0 & \cdot & \cdot & \cdot \\ 0 & 0 & q & 0 & p & \cdot & \cdot & \cdot \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

- To solve the system we simply have to solve this system of recursive equations

$$\begin{cases} pv_0 & = & q(v_0 + v_1) \\ (p+q)v_i & = & pv_{i-1} + qv_{i+1} \quad \forall i > 0 \\ \sum_{i=0}^{\infty} v_i & = & 1 \end{cases}$$

- Whose solution yields the well known geometric distribution of customers in a queue:

$$v_0 = \left(1 - \frac{p}{q}\right); \quad v_i = \left(1 - \frac{p}{q}\right)\left(\frac{p}{q}\right)^i \quad \forall i > 0$$

- The DT Birth-Death Process models any (single server, single customer class) DT queueing system given that p is known and $q = (1 - p)$ is a reasonable assumption

- Consider a simple processor with two cores and a L1 cashe memory
- If processes running on different cores need to access the cashe there is a conflict and one must wait slowing the processing

- The state of the system is simply $S = \{I, C_1, C_2, W\} = \{0, 1, 2, 3\}$ Idle: no core is accessing the cashe; C_1 (C_2) core 1 (or 2) is accessing alone; or one is accessing and the other is Waiting
- Assume the probability of accessing are p_1 and p_2 respectively in any time slot and the time to retrieve the content of the cash is exactly one slot time, while retrieving the content the core is blocked and cannot generate other requests. Then the model is

$$P = \begin{bmatrix} 1-(p_1+p_2) & p_1 - 0.5p_1p_2 & p_2 - 0.5p_1p_2 & p_1p_2 \\ 1-p_2 & 0 & p_2 & 0 \\ 1-p_1 & p_1 & 0 & 0 \\ 0 & 0.5 & 0.5 & 0 \end{bmatrix}$$

- Solve the model
- Extend the model to 4 cores and content retrieve time uniformly distributed between 1 and 4 slots and solve it (if it is too complex to solve it in close form, program the solution as a function of $p_1 \cdots p_4$)

- The real performance of the system can be normally derived from the state distribution (sometimes from transitions, but we do not consider this case for the time being
- We can associate **rewards** r_i to any state that measure it performance
- The performance of the system is associated to the average reward r

$$r = \sum_{i=0}^{\infty} r_i \cdot v_i$$

- If we are interested in the transient reward until step K we can compute

$$r(K) = \sum_{k=0}^{K} \sum_{i=0}^{\infty} r_i \cdot p_i(k)$$

- Back to the cashe memory model
- The performance of the system is given by its efficiency, so we can assume the following reward distribution:
  $r_0 = 1, r_1 = r_2 = 0.5; r_3 = 0$
- Compute the "surface" (p_1, p_2) that guarantees that $r > r_t$, where r_t is the target efficiency of your system
- This result tells you what are the characteristics of the workload that your 2-core processor can accept to
- Extend this result to the 4 cores case
- Make a comparison between a 4 core processor and two 2-cores one with the same processing power and cashe memory capacity

- Some systems cannot be modeled in discrete time ...
  - When “human time” is involved
  - When the evolution of the system is intrinsically analogic
- ... but we know there are CTMC
- Classification of CTMC states is similar to DTMC, but Periodic states do not exist
- The condition for steady state existence is similar to DTMCs (we do not make the whole analysis again)

- Recall that all transitions in a CTMS are exponentially distributed (implied by the fact that dwell times must be exponentially distributed)
- A CTMC is fully described by a matrix

$$Q = [q_{ij}] = \begin{bmatrix} q_0 & q_{01} & q_{02} & \cdot & \cdot \\ q_{10} & q_1 & q_{12} & \cdot & \cdot \\ q_{20} & q_{21} & q_2 & \cdot & \cdot \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

called the **infinitesimal generator**

- q_{ij} are the transition rates from stat i to state j
- $q_i = - \sum_{j=0, j \neq i}^{\infty} q_{ij}$
- Neither q_{ij}, nor q_i are probabilities, but the relation above stems for a simple conservation law "on average whatever goes in must come out"
- State probabilities are normally called π and not v $\ldots \pi(t) = [\pi_0(t), \pi_1(t), \pi_2(t), \ldots]$

- The steady state of a CTMC exists under the same conditions (with the due changes!) of a DTMC
- The Chapman-Kolmogorov equations can be found first writing time-dependent probabilities and then taking the limit for δt going to zero, obtaining differential equations
- Finally, solving these equations we find that the steady-state state probability vector π as solution of the linear system

$$\pi Q = 0; \quad \sum_{i=0}^{\infty} \pi_i = 1$$