EP2200 Queuing theory and teletraffic systems

Lecture 4-5

Queuing systems
Little's result
M/M/1

Viktoria Fodor
KTH EES

Outline for today and for next lecture

- Recall: Markov chain, birth-death process
- Queuing systems
  - Categories, Kendall notation
  - Markovian queuing systems
- Little's result
- M/M/1 queuing systems

Recall – Continuous time Markov-chains

- Markovian property: next state depends on the present state only
- State transition intensity matrix **Q**
- Stationary solution: $\underline{p}\mathbf{Q}=\underline{0}$, or balance equations

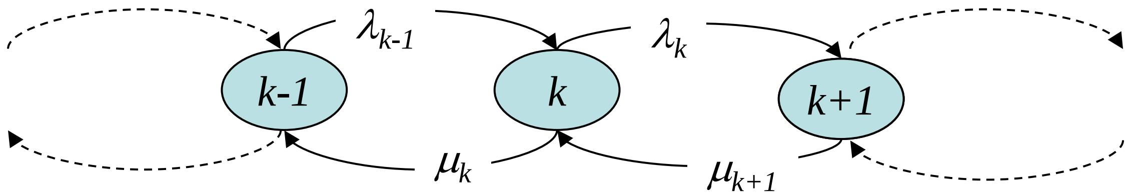
- Pure birth process, Pure death process
  - Transition to neighboring state
  - Exponentially distributed times between events
  - Poisson process

- Birth-death process: Markov chain with transition between neighboring states

Birth-death process

- Continuous time Markov-chain
- Transitions occur only between neighboring states

$i \to i+1$ birth with intensity λ_i
$i \to i-1$ death with intensity μ_i (for $i>0$)
} modells population



$$\mathbf{Q} = \begin{bmatrix} -\sum q_{0j} & q_{01} & q_{02} & \cdots \\ q_{10} & -\sum q_{1j} & q_{12} & \cdots \\ q_{20} & q_{21} & -\sum q_{2j} & \cdots \\ \vdots & & & \ddots \end{bmatrix} = \begin{bmatrix} -\lambda_0 & \lambda_0 & 0 & \cdots \\ \mu_1 & -(\lambda_1+\mu_1) & \lambda_1 & 0 \\ 0 & \mu_2 & -(\lambda_2+\mu_2) & \lambda_2 \\ & & & \ddots \end{bmatrix}$$

- State holding time – length of time spent in a state k
  - Until transition to states $k-1$ or $k+1$
  - Minimum of the times to the first birth or first deaths → minimum of two Exponentially distributed random variables: Exp($\lambda_k+\mu_k$)

B-D process - stationary solution

- Local balance equations, like for general Markov-chains
- Stability: positive solution for <u>p</u> (since the MC is irreducible)

$$\text{Cut 1}: \lambda_{k-1} p_{k-1} = \mu_k p_k \quad \Rightarrow \quad p_k = \frac{\lambda_{k-1}}{\mu_k} p_{k-1}$$

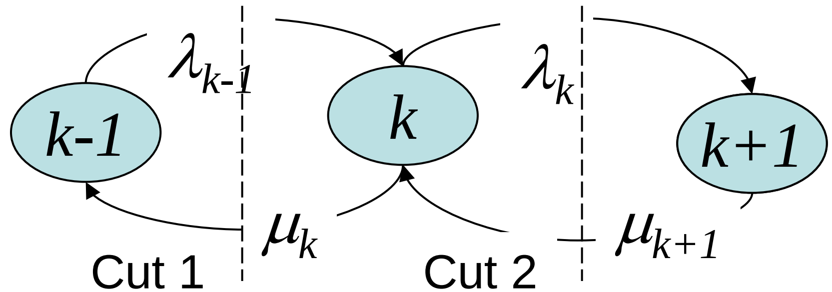
$$\text{Cut 2}: \lambda_k p_k = \mu_{k+1} p_{k+1} \quad \Rightarrow \quad p_{k+1} = \frac{\lambda_k}{\mu_{k+1}} p_k = \frac{\lambda_k \lambda_{k-1}}{\mu_{k+1}\mu_k} p_{k-1}$$

$$\vdots$$

$$\Rightarrow \boxed{p_k = \frac{\lambda_0 \cdots \lambda_{k-1}}{\mu_1 \cdots \mu_k} p_0 = \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}} p_0,}$$

$$\sum p_k = 1 \quad \Rightarrow$$

$$\boxed{p_0 = \frac{1}{1 + \sum_{k=1}^{\infty} \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}}},}$$

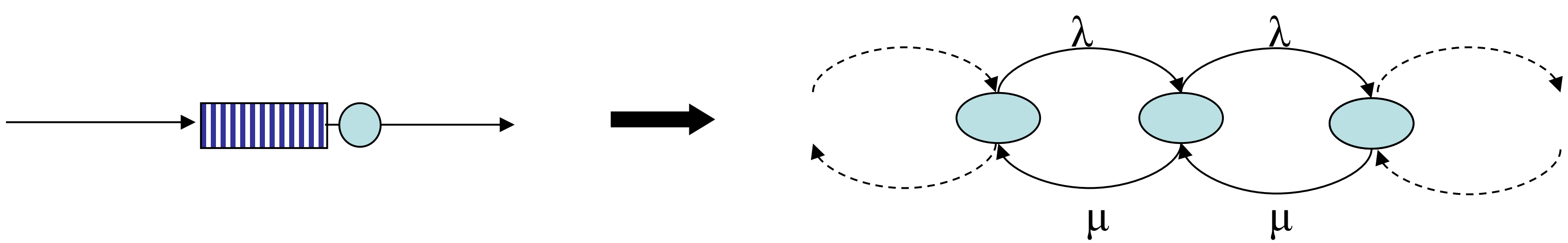


Group work: stationary solution for state independent transition rates:

$$\lambda_i = \lambda, \ \mu_i = \mu.$$

Markov-chains and queuing systems

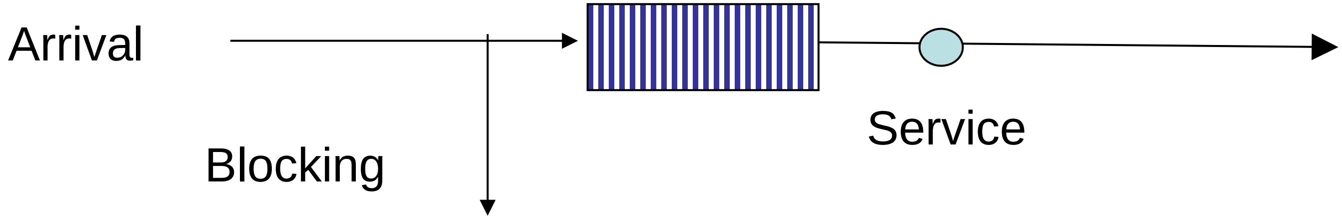
- Why do we like Poisson and B-D processes?
  How are they related to queuing systems?
  - If arrivals in a queuing system can be modeled as Poisson process $\rightarrow$ also as a pure birth process
  - If services in a queuing systems can be modeled with exponential service times $\rightarrow$ also as a (pure) death process
  - Then the queuing system can be modeled as a birth-death process



Outline for today and for next lecture

- Recall: Markov chain, birth-death process
- Queuing systems
  - Categories, Kendall notation
  - Markovian queuing systems
- Little's result
- M/M/1 queuing systems

Queuing system: Kendall's notation A/S/m/c/p/O



- A: arrival process (distribution of interarrival times)
- S: distribution of the service times
- m: number of servers
- c: system capacity – buffer positions and servers included (omitted if infinite)
- p: population generating requests (omitted if infinite)
- O: order of service (omitted if FCFS)

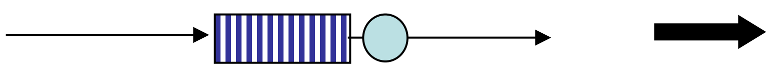
- Inter arrival or service time:
  - *M*: Markovian (exponentially distributed)
  - *D*: Deterministic (same known value)
  - E_r: Erlang with *r* stages (sum of r exponentials)
  - H_k: Hyper exponential with *k* branches (mix of *k* exponentials)
  - *G*: General (but known), some times *GI* for general, independent

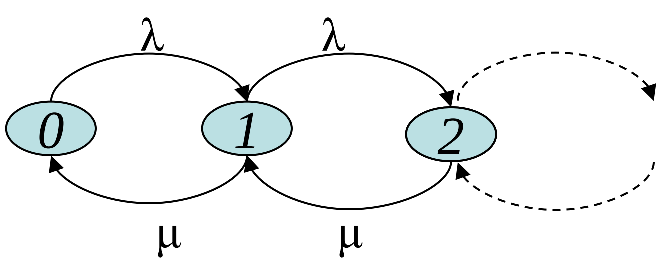
Markovian queuing systems

- State of the queuing system: number of customers in the system
- Markovian queuing system: if the Markovian property holds
  - the next state of the system depends on the present state only

- Interarrival and service times have to be exponential: M/M/*/*
  - arrival: birth process (intensity: λ_i)
  - service: death process (intensity: μ_i)
  - B-D process to model the queuing system
    - State: number of customers in the system

- E.g. M/M/1, $\lambda_i = \lambda$, $\mu_i = \mu$





Markovian queuing systems (M/M/*/*)

- Markovian property holds:
  - –interarrival times are exponential (Poisson process)
  - –service times are exponential
- Poisson arrival process - motivation
  - –Models a population of independent customers
  - –Each customer access the system at a low rate
  - –The total arrival process tends towards a Poisson process for a large population
- Exponential service time motivation is not that straightforward...
- Queuing system described with a Markov chain (often B-D)

Group-work

- Can we model these queuing systems with a B-D process?

1. Packets of exponential length are multiplexed from a high number of input ports. The arrival processes at the input ports are Poisson.

2. Packets of fixed length are multiplexed at the same router as in 1. The input process in Poisson.

3. Packets of exponential length are multiplexed and the transmission bandwidth is increased as the queue length increases (dedicated bandwidth for this service). The input process is Poisson.

Queuing system –variables

$p_k(t)$: probability of k customers in the system at time t, stationary p_k

λ: arrival intensity, average interarrival time $1/\lambda$ (offered traffic)

x_n: service time requirement of customer n, average x (or $\bar{x}$)

μ: service intensity, $\bar{x} = 1/\mu$

T_n: time customer n spends in the system (system time), average T

W_n: waiting time of customer n, average W

Relation: $T = W + x$

$N(t)$: number of customers in system at time t, average N

$N_q(t)$: number of customers waiting at time t, average N_q

$N_s(t)$: number of customers in service at time t, average N_s

Relation: $N = N_q + N_s$

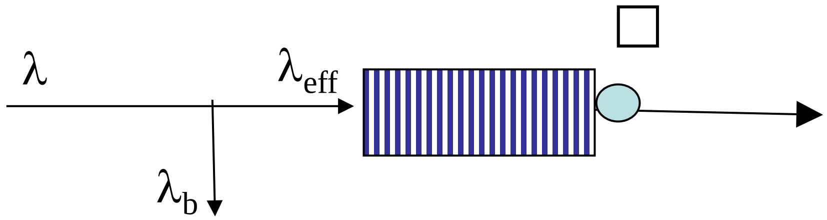
Offered load and utilization

- Offered load: $a = \lambda \bar{x} = \lambda / \mu$, (arrival intensity * length of service)
  - is expressed in Erlang (E) [no unit]
  - sometimes denoted by ρ.
- Server utilization in systems with infinite buffer capacity, m servers

$$\rho = \frac{\text{time server occupied}}{\text{total time}} = \frac{\lambda T \, \bar{x}/m}{T} = \frac{\lambda}{m\mu} = \frac{a}{m}$$

Stability requires ρ<1

- For systems with blocking:
  - Effective traffic: λ_{eff}
  - Blocked traffic: λ_b, $\lambda_{eff} + \lambda_b = \lambda$
  - Effective load: $\lambda_{eff}\bar{x} = \lambda_{eff} / \mu$
  - Server utilization: $\lambda_{eff}\bar{x}/m = \lambda_{eff}/(m\mu)$



Group work: 2 dentists
- 4 arrivals per hour in average
- 20 minutes "service" in average

Offered load?
Part of time the dentist is busy (utilization)?

Outline for today and for next lecture

- Recall: Markov chain, birth-death process
- Queuing systems
  - Categories, Kendall notation
  - Markovian queuing systems
- Little's result
- M/M/1 queuing systems

Little's result

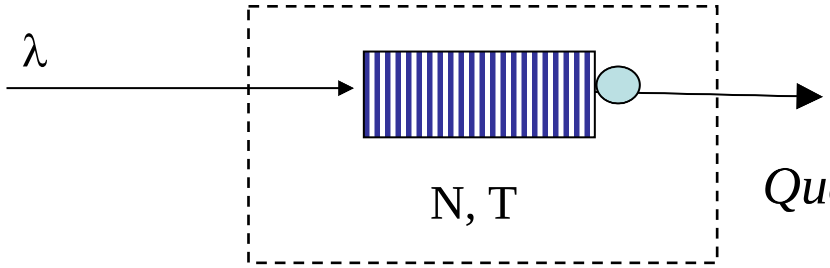
- First for systems without blocking
- The average number of customers in the system is equal to the arrival rate times the average time spent in the system

$$N = \lambda T$$

- Likewise $N_q = \lambda W$

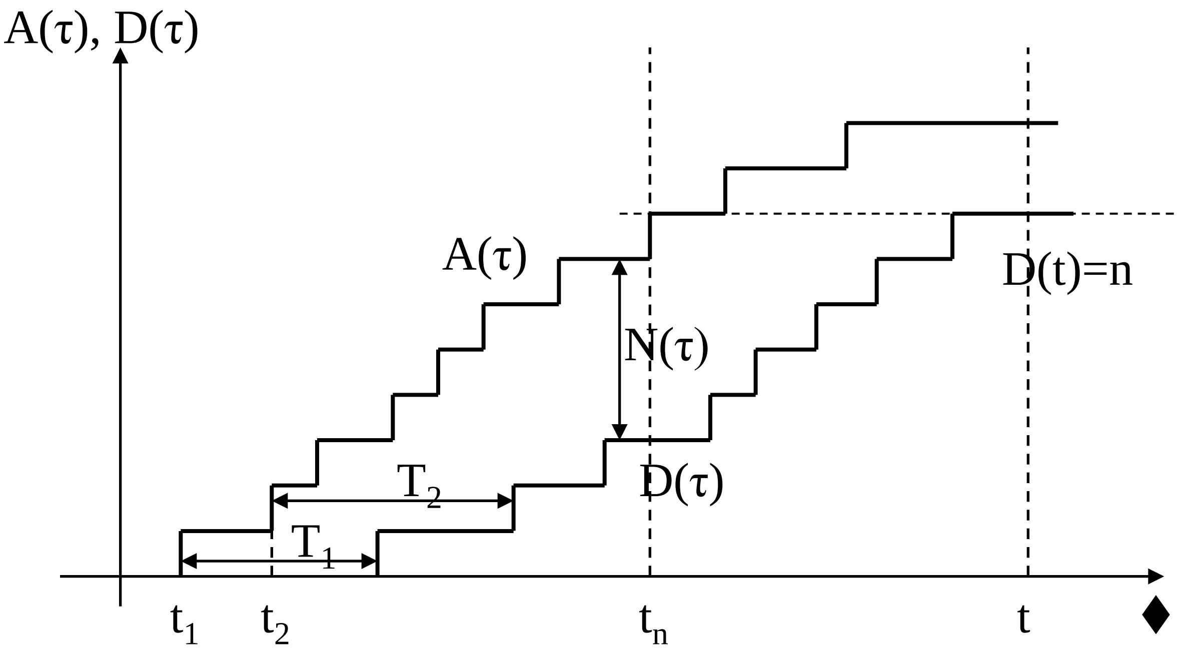
$$N_s = \lambda \overline{x}$$

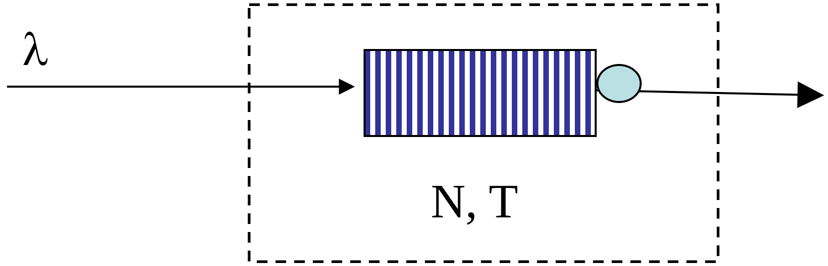
- General result for G/G/m systems
  - applies for all queuing systems we will consider



Queuing system as a black box

Little's result - justification





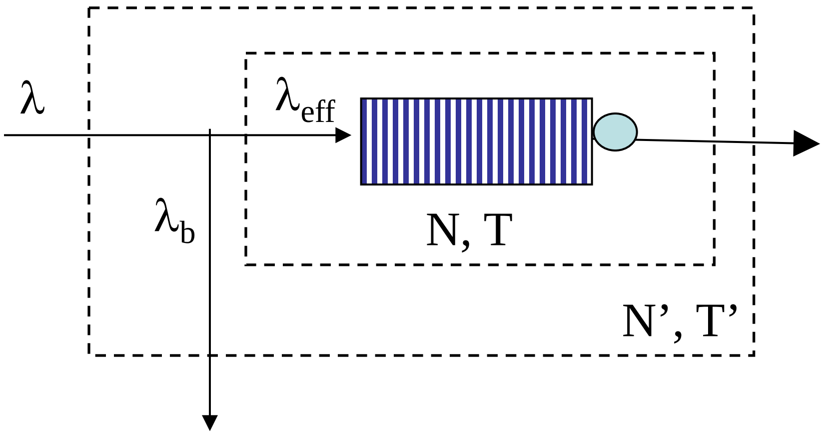
To prove: $N = \lambda T$

- $A(\tau)$: num. of arrived requests until τ
- $D(\tau)$: num. of served requests until τ
- $N(\tau)$: num. of requests in the system at τ
- T_i: system time of customer i
- t_i: arrival time of customer i
- N_t: number of customers in the system averaged until time t
- λ_t: arrival rate until time t
- T_t: system time averaged until time t

(Here we show for m=1)

Little's result for loss systems

- Some of the requests get blocked
- Little's result holds
  1. for the effective traffic (considering the accepted costumers): N=λ_{eff}T
     - Since Little's result holds for the arrival process "after" the blocking
  2. for the offered traffic (considering both the accepted and the blocked costumers: N=λT', where T' is the average system time, including 0 system time for blocked requests, T' ≠ T
     - Proof below



$$1: N = \lambda_{eff} T$$

$$2:$$

$$N' = N$$

$$T' = \frac{\lambda_b}{\lambda} 0 + \frac{\lambda_{eff}}{\lambda} T$$

$$\Rightarrow \quad \lambda T' = \lambda_{eff} T = N$$

Queuing systems - summary

- Kendall notation A/S/m/c/p/O
- Markovian (M/M/*/*) systems and B-D processes
- Offered load and utilization
- Little's result: $N=\lambda T$

M/M/1 queuing systems

- Single server, infinite waiting room
- Service times are exponentially distributed (μ)

- Arrival process Poisson (λ)

- The queuing system can be modeled by a homogeneous (time-independent) birth-death process

- Here basic case: state independent arrival and service
- On the recitation: M/M/1 with state dependent arrival and service (λ_i, μ_i)

M/M/1 queuing systems - performance

1. State transition diagram
2. Stationary state probabilities
   – Condition of stability
3. Average number of customers in the system
4. Other average measures
5. Scheduling discipline?
6. Distribution of system time (and waiting time)

- *The derivation of these expressions *is* exam material. See your lecture notes, or parts of the Virtamo notes.*

M/M/1 queuing systems

- State transition diagram: BD process
- What is the lifetime of a state?
  - Also called holding time
  - Process leaves a state if there is an arrival or a service
  - Exponential interarrival and service time
  - Lifetime: minimum of two independent exponential random variables:

$$P(\tau < t) = 1 - e^{-(\lambda+\mu)t}, \quad \bar{\tau} = \frac{1}{\lambda+\mu}$$

  - For state 0: only arrival, no service

$$P(\tau_0 < t) = 1 - e^{-\lambda t}, \quad \bar{\tau} = \frac{1}{\lambda}$$

Markovian queuing systems

State probability

- $p_k(t)=P(N(t)=k)=$
  P(number of customers in the system is k at time t)

- Stationary state probabilities p_k
  - fraction of processes in state k
  - fraction of time the system is in state k (due to ergodicity)
  - P(a random observer finds the system in state k)

- PASTA (Poisson Arrivals See Time Average)
  - define $a_k(t)$=P(arriving customer finds the system in state k at time t, given that a customer arrives at time t)
  - for Poisson arrivals $a_k(t)= p_k(t)$
    (The arrival rate has to be state independent!)

  - not true for all arrival processes! E.g., deterministic arrivals

Performance results

- The system is in state k with probability $p_k=(1-\rho)\rho^k$
- An arriving customer finds k customers in the system with probability p_k *(PASTA)*
- Expected number of customers in the system is $N=\rho/(1-\rho)$
  – Time measures by Little's law
- Service discipline: state probability and average performance measures do not depend on the service discipline
- System time and waiting time distribution under FIFO

$$P(T<t)=T(t)=1-e^{-(\mu-\lambda)t},\ t\geq 0$$

$$P(W<t)=W(t)=1-\rho e^{-(\mu-\lambda)t},\ t\geq 0$$

- Terminology in the Virtamo notes:
  – System time = sojourn time (M/M/* p7-10)

Summary

- Queuing systems
  - Categories, Kendall notation
- Little’s result, without and with blocking
- M/M/1 queuing systems and performance results

- Continuation: Markovian queuing systems
  - With blocking
  - With more servers