Teoria das Filas

Introdução

Conceitos básicos: Nomenclatura e métricas

Classificação

Little's law para redes abertas

Little's law para redes fechadas

Notação de Kendall

Processo de Markov

Processo de Nascimento e Morte para uma fila M/M/1

Processo de Nascimento e Morte para uma fila M/M/c e M/M/c/K

Políticas de Escalonamento

Introdução

No contexto de computação:
- muitos jobs
- recurso limitado
- longas filas
- delay na resposta

Precisamos:
* Predizer a performance do sistema
* Encontrar o melhor design

Conceitos básicos p/ uma Fila

* Number of jobs in the system

Service order

* Average Arrival rate: e.g. 3 jobs/sec

* Mean Interarrival rate: e.g. $1/\lambda = 1/3$ sec

* Response time

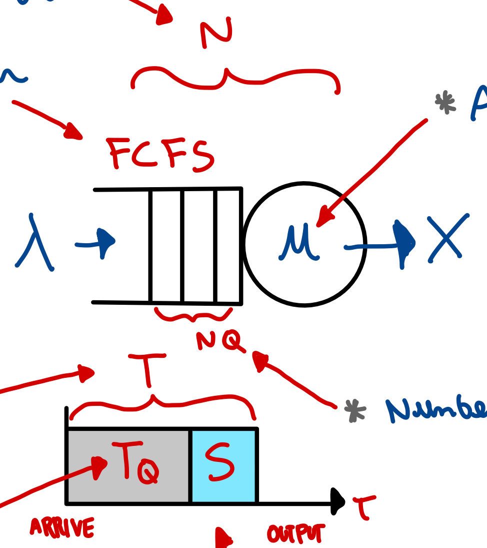
Waiting time (delay)

$(\mu = 1/E[S])$

* Average service rate: e.g. 4 jobs/sec

Throught $(X = \rho\mu)$

* Number of jobs in the queue



* Service requirement (Size): V.A S

* Mean Service Time: E[S]

Classificação

Queue
- Aberta: Single-Server System, Finite buffer

Network
- Fechada
  - Iterativa: Recebe jobs de fora e envia a resposta para fora
  - batch System: População fixa, nunca sai do sistema

* taxa de saída (X):

$$X = \frac{C}{Z}$$

taxa de saída; C: Nº de jobs completed em Z período; Z: Tempo de observação

* Utilização (ρ):

$$\rho = \frac{B}{Z}$$

ρ: fração de tempo que o servidor está busy; B: tempo device busy em Z observação; Z: Tempo de observação

* $\begin{cases} \rho = B/Z \rightarrow Z = B/\rho \\ X = C/Z \rightarrow Z = \frac{C}{X} \end{cases}$

$$\boxed{X = \frac{C}{B}\rho}$$

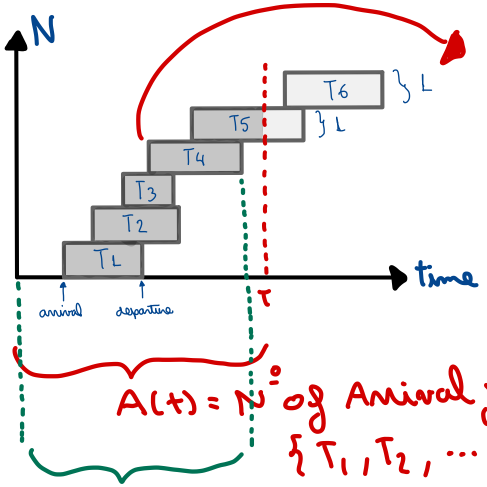
$$\frac{C}{B} = \frac{\text{Nº completed jobs (job)}}{\text{busy time (sec)}} = \mu$$

$$\boxed{X = \mu\rho}$$

Little's Law para Sistemas Abertos

$$\boxed{E[N] = \lambda E[T]}$$

Prova:



$A = \int_0^t N(s)\,ds$

$A(t) =$ Nº of Arrival jobs $\{T_1, T_2, \dots, T_5\}$

$C(t) =$ Nº of completed jobs $\{T_1, T_2, \dots, T_4\}$

A área (A) em cinza escuro é dado por:

$$\sum_{i \in C(t)} T_i \leq \int_0^t N(s)\,ds \leq \sum_{i \in A(t)} T_i$$

$$\sum_{C(t)} \frac{T_i\, C(t)}{t\; C(t)} \leq \int_0^t \frac{N(s)\,ds}{t} \leq \sum_{A(t)} \frac{T_i\, A(t)}{A(t)\, t}$$

$$\underbrace{\lim_{T\to\infty} \frac{\sum T_i}{C(t)}}_{\bar{T}} \cdot \underbrace{\lim_{T\to\infty} \frac{C(t)}{T}}_{X} \leq \underbrace{\lim_{T\to\infty} \int_0^t \frac{N(s)\,ds}{t}}_{\bar{N}} \leq \underbrace{\lim_{T\to\infty} \frac{\sum T_i}{A(t)}}_{\bar{T}} \underbrace{\lim_{T\to\infty} \frac{A(t)}{T}}_{\lambda}$$

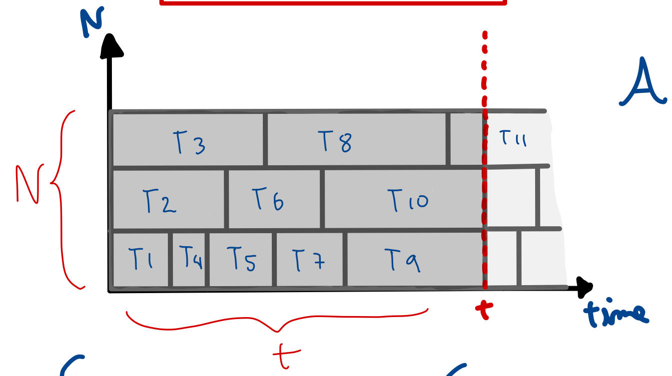
$$\bar{T}X \leq \bar{N} \leq \bar{T}\lambda$$

Tipicamente, $X = \lambda$, logo: $\boxed{\bar{N} = \bar{T}\lambda}$

Little's Law para Sistemas Fechados

$$\boxed{N = X E[T]}$$

Prova:



$A = Nt$

$$\sum_{i, C(t)} T_i \leq Nt \leq \sum_{i, A(t)} T_i$$

$$\underbrace{\lim_{T\to\infty} \frac{\sum_{C(t)} T_i}{C(t)}}_{\bar{T}} \underbrace{\lim_{T\to\infty} \frac{C(t)}{t}}_{X} \leq N \leq \underbrace{\lim_{T\to\infty} \frac{\sum_{i, A(t)} T_i}{A(t)}}_{\bar{T}} \cdot \underbrace{\lim_{t\to\infty} \frac{A(t)}{t}}_{\lambda}$$

Tipicamente $X = \lambda$, então temos:

$$\boxed{N = X\bar{T}}$$

Notação de Kendall:

$$A / S / m\ (/\ B\ /\ K\ /\ SD)$$

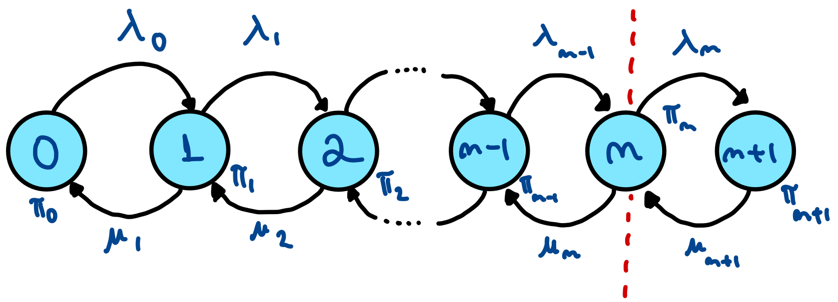
- * Arrival Process → A
- * Service Time Distribution → S
- * Number of servers → m
- * Number of buffer → B
- * Population size → K
- * Service Discipline → SD

Processo de Markov

* As filas podem ser modeladas como um processo de Markov
* O tempo gasto por uma job em uma fila é um processo de Markov
* O número de jobs em uma fila é um processo de Markov

Birth and Death Process (M/M/1)

CTMC



Como o processo está em long-run, consideramos que a cadeia está em steady state.

Balance equation: $\boxed{\text{Flow IN} = \text{Flow out}}$

State	rate IN = rate OUT	π
0	$\lambda_0\pi_0 = \mu_1\pi_1$	π_0
1	$\lambda_0\pi_0 + \mu_2\pi_2 = \lambda_1\pi_1 + \mu_1\pi_1$	$\lambda_0/\mu_1\, \pi_0$
⋮	⋮	⋮
m		$\left(\frac{\lambda_{m-1}\lambda_{m-2}\cdots\lambda_2\lambda_1\lambda_0}{\mu_m\mu_{m-1}\cdots\mu_3\mu_2\mu_1}\right)\pi_0$

Sabendo que λ e μ são independentes do estado. concluimos que

$$\boxed{\pi_m = \left(\frac{\lambda}{\mu}\right)^m \pi_0}$$

Ainda, sabemos que $\sum_{i=0}^{\infty}\pi_i = 1 \rightarrow \sum_{i=0}^{\infty}\left(\frac{\lambda}{\mu}\right)^i\pi_0 = 1$

$$\sum_{m=0}^{\infty}\pi_m = 1$$

$$\sum_{m=0}^{\infty}\rho^m\pi_0 = 1$$

$$1 = \pi_0 + \rho\pi_0 + \rho^2\pi_0 + \dots + \rho^m\pi_0$$

$$1 = \pi_0(\underbrace{1+\rho+\rho^2+\dots+\rho^m}_{SG})$$

$$\boxed{\pi_0 = 1-\rho} \rightarrow \boxed{\pi_0 = 1 - \frac{\lambda}{\mu}}$$

$$N = \sum_{m=0}^{\infty} m\pi_m = \sum_{m=0}^{\infty} m\rho^m\pi_0$$

$$N = \pi_0\sum_{m=0}^{\infty} m\rho^m = \pi_0\underbrace{\sum_{m=0}^{\infty} m\rho\rho^{m-1}}_{\rho\sum_{m=0}^{\infty} m\rho^{m-1}}$$

$$N = \pi_0\rho\frac{d}{d\rho}\sum\rho^m \quad PG$$

$$N = \pi_0\rho\frac{d}{d\rho}\left(\frac{1}{1-\rho}\right)$$

$$\boxed{N = \frac{\lambda}{\mu-\lambda}}$$

* $N = N_Q + N_S$

$N_Q = N - N_S$

Little's Law
$E[N_S] = \lambda E[S]$
e
$E[S] = \frac{1}{\mu}$

$N_Q = N - \frac{\lambda}{\mu}$

$$\boxed{N_Q = \frac{\lambda^2}{\mu(\mu-\lambda)}}$$

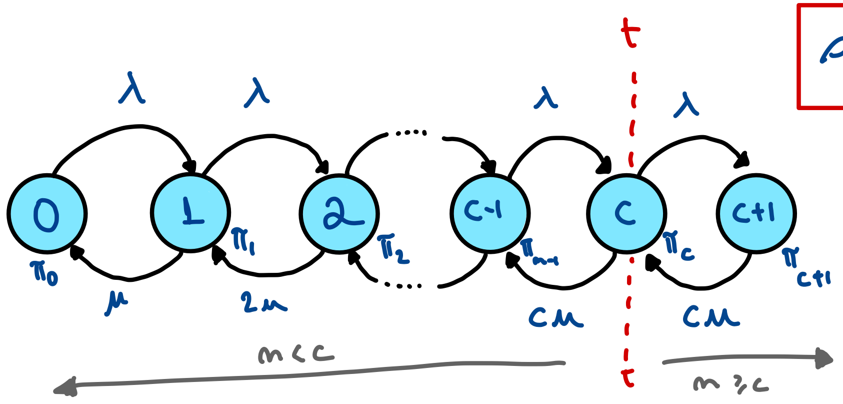
* T_Q: Little's Law $N_Q = \lambda T_Q$

$N_Q = \lambda T_Q$

$$\boxed{T_Q = \frac{\lambda}{\mu(\mu-\lambda)}}$$

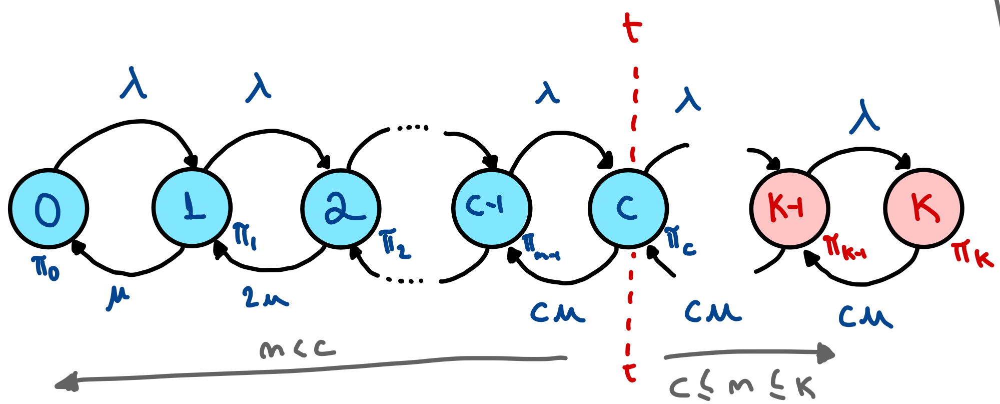
Birth and Death Process (M/M/c)

$\boxed{\rho = \frac{\lambda}{c\mu}}$



Basta resolver para os dois casos.

Birth and Death Process (M/M/c/K)



Basta resolver as equações de balanço para os dois casos.

Políticas de Escalonamento

- Política
  - Não Preemptiva (Não para o job)
    - First Come First Served (FCFS)
    - Shortest Job First (SJF)
    - Priority Scheduling
  - Preemptiva (para o job)
    - Priority Scheduling
    - Round Robin (RR)
    - Multilevel Queue Scheduling

Round Robin: * Cada processo é alocado por um período de tempo de CPU (quantum)

* time-sharing
* É um "real time" algoritmo

Priority Scheduling:

* Cada job tem uma prioridade (integer) definida por uma regra de negócio
* Problema: estagnação. Solução é usar o tempo de vida do job p/ aumentar sua prioridade.

SJF

* Semelhante ao priority scheduling, sendo que a prioridade é o burst time do job