

Modelos de Redes de Filas com Múltiplas Classes (Análise de Valores Médios)

Profa. Jussara M. Almeida
1º Semestre de 2014

Modelos com Múltiplas Classes

- Provê estimativas de medidas de desempenho separadamente para cada classe
 - Resultados mais precisos se carga heterogênea (ex: mistura de processos CPU e I/O bound)
- Coleta de informações:
 - Parâmetros de entrada devem ser coletados para cada classe, separadamente
 - Ferramentas disponíveis nem sempre provêm info discretizada por classe: precisão superior pode ser comprometida
- Solução dos modelos: maior custo computacional

Representação da Carga de Trabalho

- Conjunto de demandas + intensidade da carga separadas para cada classe:
 - $D_{c,k}$
 - λ_c , N_c ou N_c e Z_c (classe aberta ou fechada?)
- Política de escalonamento em cada centro deve ser especificada
 - Redes separáveis:
 - FCFS: se $S_{c,k}$ é o mesmo para todo c , e dado centro k ($V_{c,k}$ e $D_{c,k}$ podem variar. Ex: disco)
 - Processor sharing
 - LCFS-preemptive resume
 - Resultados obtidos com modelo serão os mesmos para as três políticas acima

Contraste com Modelos de Classe Única

- Seja um sistema hipotético com dois recursos, uma CPU e um disco e dois componentes de carga, um batch e um interativo. Medições realizadas no sistema fornecem as seguintes informações (tempos em seg):

$$B_{\text{batch,CPU}} = 600$$

$$B_{\text{batch,disk}} = 54$$

$$C_{\text{batch}} = 600$$

$$N_{\text{batch}} = 10$$

$$Z_{\text{batch}} = 0$$

$$B_{\text{interativo,CPU}} = 47.6$$

$$B_{\text{interativo,disk}} = 428.4$$

$$C_{\text{interativo}} = 476$$

$$N_{\text{interativo}} = 25$$

$$Z_{\text{interativo}} = 30$$

Modelo com múltiplas classes:

$$D_{\text{batch,CPU}} = 1.0$$

$$D_{\text{batch,disk}} = 0.09$$

$$D_{\text{interativo,CPU}} = 0.1$$

$$D_{\text{interativo,disk}} = 0.9$$

Contraste com Modelos de Classe Única

- A construção de um modelo de classe única deve assumir que as medições não distinguem requisições pelo tipo da carga

$$Z = 476 \times 30 / (476 + 600) = 13.271 \text{ seg}$$

$$D_{\text{cpu}} = (600 + 47.6) / (476 + 600) = 0.602$$

$$D_{\text{disk}} = (54 + 428.4) / (476 + 600) = 0.448$$

- Como as soluções dos dois modelos se comparam quando são usados para verificar o ganho de desempenho de uma troca da CPU por outra 5 vezes mais rápida?

Contraste com Modelos de Classe Única

	single class		multiple class					
	overall		overall		batch		interactive	
	base	upgrade	base	upgrade	base	upgrade	base	upgrade
X	1.64	2.11	1.66	5.26	.93	4.64	.74	.62
R	8.07	3.32	7.52	3.16	10.79	2.16	3.40	10.57
U_{CPU}	.985	.254	1.000	.943	.926	.928	.074	.015
Q_{CPU}	10.70	.34	10.57	5.28	9.72	5.20	.85	.08
U_{Disk}	.733	.946	.752	.979	.084	.418	.668	.561
Q_{Disk}	2.58	6.63	2.37	11.02	.28	4.80	2.09	6.22

- Modelos de única e múltiplas classes concordam quanto ao sistema base, mas diferem significativamente nas projeções de X e R
 - Enquanto modelo com uma única classe projeta melhora de 60% para R , o modelo de múltiplas classes projeta uma melhora de 80% para os processos batch mas uma *degradação de 200%* para interativos
- Carga média: speedup da CPU é limitado pelo bottleneck secundário do disco, mas
 - Carga batch é CPU-bound: X melhora bastante com upgrade
 - Carga interativa é disk-bound: competição no disco leva a degradação

Contraste com Modelos de Classe Única

- Um modelo pode projetar efeitos que a intuição pode não reconhecer:
 - Upgrade da CPU leva a degradação de performance para carga interativa
- A aplicação de modelos de uma única classe para cargas significativamente heterogêneas pode levar a resultados errados porque:
 - A projeção de desempenho da carga "média" não é realista
 - Não é possível obter projeções para classes específicas a partir dos resultados médios

Modelos com Múltiplas Classes Abertas: Soluções

- **Carga:** $\vec{\lambda} \equiv (\lambda_1, \lambda_2, \dots, \lambda_C)$
- **Capacidade de Processamento**

$$\max_k \left[\sum_{c=1}^C \lambda_c D_{c,k} \right] < 1$$

- **Throughput**

$$X_c = \lambda_c$$

$$X_{c,k}(\vec{\lambda}) = \lambda_c V_{c,k}$$

Modelos com Múltiplas Classes Abertas: Soluções

- Utilização

$$U_{c,k}(\vec{\lambda}) = X_{c,k}(\vec{\lambda}) S_{c,k} = \lambda_c D_{c,k} \qquad U_k(\vec{\lambda}) = \sum_c U_{c,k}(\vec{\lambda})$$

- Tempo de Residência

Delay center: $R_{c,k}(\vec{\lambda}) = D_{c,k}$

Centros de serviço: $R_{c,k}(\vec{\lambda}) = D_{c,k} (1 + A_{c,k}(\vec{\lambda}))$

$A_{c,k}(\vec{\lambda})$: # médio de clientes que um cliente da classe c vê ao chegar em um centro k

Modelos com Múltiplas Classes Abertas: Soluções

$$A_{c,k}(\vec{\lambda}) = Q_k(\vec{\lambda}) \quad \text{Teorema da Chegada}$$

$$R_{c,k}(\vec{\lambda}) = D_{c,k} (1 + Q_k(\vec{\lambda}))$$

$$R_{c,k}(\vec{\lambda}) = D_{c,k} \left[1 + \sum_{j=1}^C \lambda_j R_{j,k}(\vec{\lambda}) \right]$$

$$\frac{R_{c,k}(\vec{\lambda})}{R_{j,k}(\vec{\lambda})} = \frac{D_{c,k}}{D_{j,k}} \longrightarrow R_{j,k}(\vec{\lambda}) = \frac{D_{j,k}}{D_{c,k}} R_{c,k}(\vec{\lambda})$$

$$R_{c,k}(\vec{\lambda}) = \frac{D_{c,k}}{1 - \sum_{j=1}^C U_{j,k}(\vec{\lambda})}$$

Modelos com Múltiplas Classes Abertas: Soluções

- Tamanho médio da Fila

$$Q_{c,k}(\vec{\lambda}) = \lambda_c R_{c,k} = \begin{cases} U_{c,k}(\vec{\lambda}) & \text{centro de atraso} \\ \frac{U_{c,k}(\vec{\lambda})}{1 - \sum_{j=1}^C U_{j,k}(\vec{\lambda})} & \text{centro de serviço} \end{cases}$$

- Tempo de Resposta do Sistema

$$R_c(\vec{\lambda}) = \sum_{k=1}^K R_{c,k}(\vec{\lambda})$$

- Número Médio de Clientes no Sistema

$$Q_c(\vec{\lambda}) = \lambda_c R_c(\vec{\lambda}) = \sum_{k=1}^K Q_{c,k}(\vec{\lambda})$$

Modelos com Múltiplas Classes

Abertas: Sumário

$$\text{processing capacity : } \max_k \left\{ \sum_{c=1}^C \lambda_c D_{c,k} \right\} < 1$$

$$\text{throughput : } X_c(\bar{\lambda}) = \lambda_c$$

$$\text{utilization : } U_{c,k}(\bar{\lambda}) = \lambda_c D_{c,k}$$

$$\text{residence time : } R_{c,k}(\bar{\lambda}) = \begin{cases} D_{c,k} & \text{(delay)} \\ \frac{D_{c,k}}{1 - \sum_{j=1}^C U_{j,k}(\bar{\lambda})} & \text{(queueing)} \end{cases}$$

$$\text{queue length : } Q_{c,k}(\bar{\lambda}) = \lambda_c R_{c,k}(\bar{\lambda})$$

$$= \begin{cases} U_{c,k}(\bar{\lambda}) & \text{(delay)} \\ \frac{U_{c,k}(\bar{\lambda})}{1 - \sum_{j=1}^C U_{j,k}(\bar{\lambda})} & \text{(queueing)} \end{cases}$$

$$\text{system response time : } R_c(\bar{\lambda}) = \sum_{k=1}^K R_{c,k}(\bar{\lambda})$$

$$\text{average number in system : } Q_c(\bar{\lambda}) = \lambda_c R_c(\bar{\lambda}) = \sum_{k=1}^K Q_{c,k}(\bar{\lambda})$$

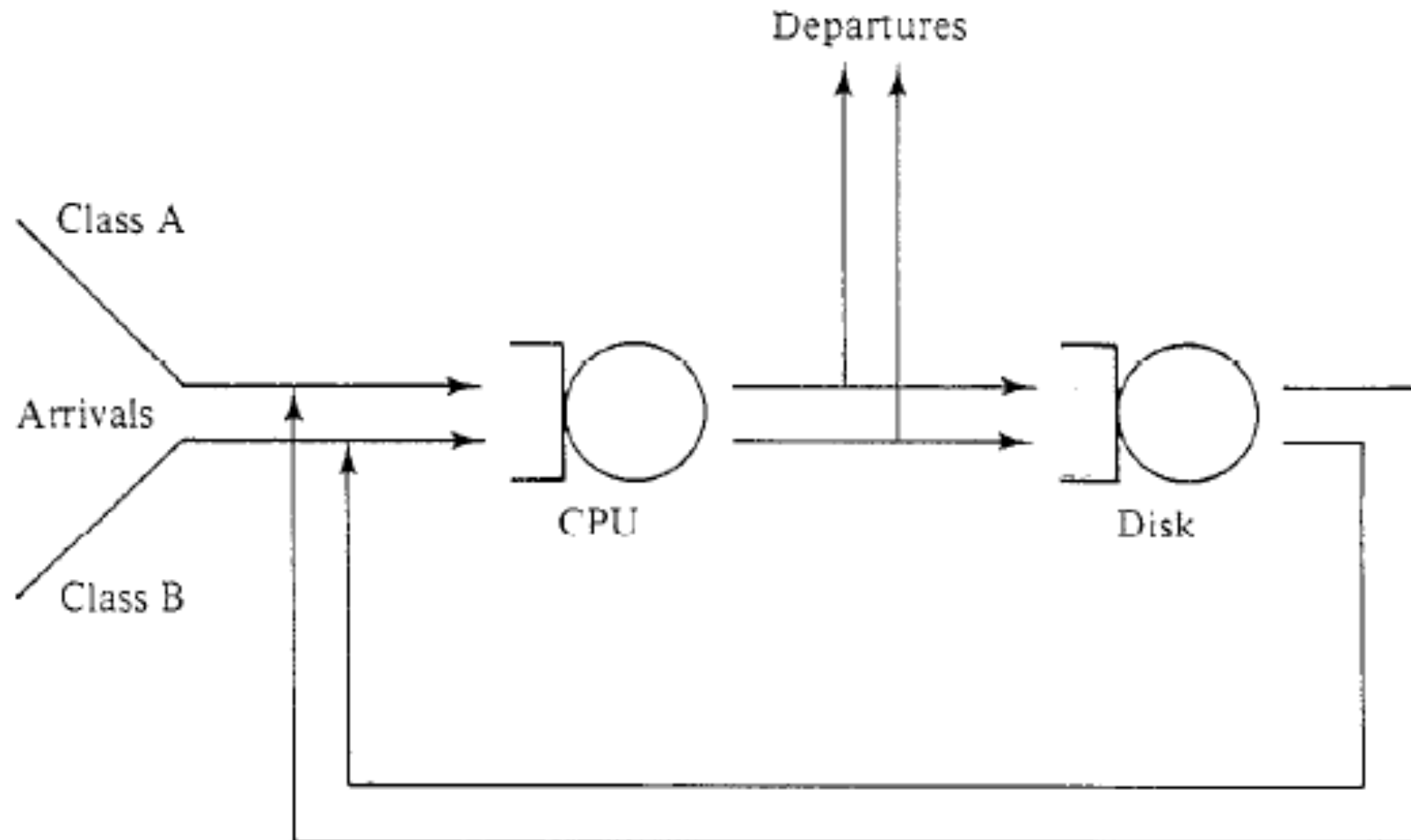
Modelos de Filas Abertos: Exemplo

- $V_{A,CPU} = 10$ $V_{A,disk} = 9$ $V_{B,CPU} = 5$ $V_{B,disk} = 4$
- $S_{A,CPU} = 1/10$ $S_{A,disk} = 1/3$ $S_{B,CPU} = 2/5$ $S_{B,disk} = 1$
- $D_{A,CPU} = 1$ $D_{A,disk} = 3$ $D_{B,CPU} = 2$ $D_{B,disk} = 4$

$$\lambda_A = 3/19 \text{ processos/seg}$$

$$\lambda_B = 2/19 \text{ processos/seg}$$

Modelos de Filas Abertos: Exemplo



Modelos de Filas Abertos: Exemplo

$$X_{A,CPU}(\bar{\lambda}) = \lambda_A V_{A,CPU} = \frac{3}{19} \times 10 = 1.58 \text{ jobs/sec.}$$

$$U_{A,CPU}(\bar{\lambda}) = \lambda_A D_{A,CPU} = \frac{3}{19} \times 1 = .158$$

$$R_{A,CPU}(\bar{\lambda}) = \frac{D_{A,CPU}}{1 - \sum_{j=A}^B U_{j,CPU}(\bar{\lambda})} = \frac{1}{12/19} = 1.58 \text{ secs.}$$

$$Q_{A,CPU}(\bar{\lambda}) = \frac{U_{A,CPU}(\bar{\lambda})}{1 - \sum_{j=A}^B U_{j,CPU}(\bar{\lambda})} = \frac{3/19}{1 - (\frac{3}{19} + \frac{4}{19})} = .25 \text{ jobs}$$

$$R_A(\bar{\lambda}) = R_{A,CPU}(\bar{\lambda}) + R_{A,Disk}(\bar{\lambda}) = \frac{19}{12} + \frac{57}{2} = 30.08 \text{ secs.}$$

Modelos com Múltiplas Classes Fechadas: Soluções

- **Carga:** $\vec{N} \equiv (N_1, N_2, \dots, N_C)$

- **Throughput:**
$$X_c(\vec{N}) = \frac{N_c}{Z_c + \sum_{k=1}^K R_{c,k}(\vec{N})}$$

- **Número Médio na Fila:**

$$Q_{c,k}(\vec{N}) = X_c(\vec{N}) R_{c,k}(\vec{N})$$

$$Q_k(\vec{N}) = \sum_{c=1}^C Q_{c,k}(\vec{N})$$

Modelos com Múltiplas Classes Fechadas: Soluções

- Tempo de Residência:

$$R_{c,k}(\vec{N}) = \begin{cases} D_{c,k} \\ D_{c,k} (1 + A_{c,k}(\vec{N})) \end{cases}$$

Soluções exata e aproximada

Modelos com Múltiplas Classes Fechadas: Solução Exata

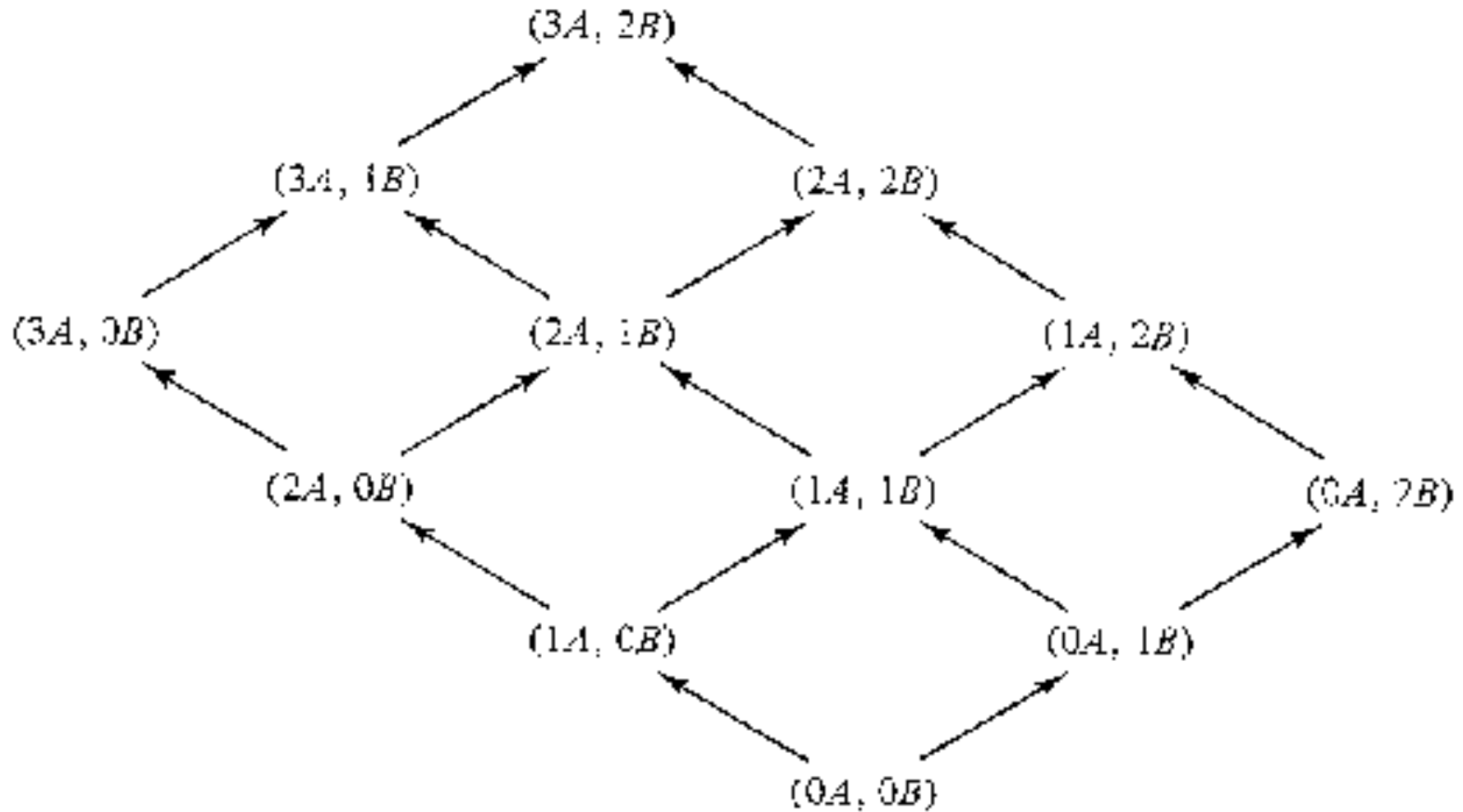
Tempo de Residência:

$$A_{c,k}(\vec{N}) = Q_k(\overrightarrow{N - 1_c})$$

$$R_{c,k}(\vec{N}) = \begin{cases} D_{c,k} \\ D_{c,k} (1 + Q_k(\overrightarrow{N - 1_c})) \end{cases}$$

$\overrightarrow{N - 1_c}$ = população $\vec{N}$ com um cliente da classe c a menos

Precedência de Soluções Intermediárias



Solução Exata para Modelos Fechados

```

for  $k \leftarrow 1$  to  $K$  do  $Q_k(\vec{0}) \leftarrow 0$ 
for  $n \leftarrow 1$  to  $\sum_{c=1}^C N_c$  do
  for each feasible population  $\vec{n} \equiv (n_1, \dots, n_C)$  with  $n$  total
    customers do
    begin
      for  $c \leftarrow 1$  to  $C$  do
        for  $k \leftarrow 1$  to  $K$  do
          
$$R_{c,k} \leftarrow \begin{cases} D_{c,k} & \text{(delay)} \\ D_{c,k} [1 + Q_k(\vec{n-1}_c)] & \text{(queueing)} \end{cases}$$

        for  $c \leftarrow 1$  to  $C$  do  $X_c \leftarrow \frac{n_c}{Z_c - \sum_{k=1}^K R_{c,k}}$ 
        for  $k \leftarrow 1$  to  $K$  do  $Q_k(\vec{n}) \leftarrow \sum_{c=1}^C X_c R_{c,k}$ 
      end
    end
  end
end

```

Modelos de Filas Fechadas: Exemplo de Solução Exata

- $D_{A,CPU} = 1$ $D_{A,disk} = 3$ $D_{B,CPU} = 2$ $D_{B,disk} = 4$
- $N_A = 1$ $N_B = 1$

	population vectors			
	(0A,0B)	(1A,0B)	(0A,1B)	(1A,1B)
$R_{A,CPU}$	-	1	-	4/3 → $D_{A,CPU}(1+Q_{cpu}^{(0A,1B)})$
$R_{A,Disk}$	-	3	-	5
$R_{B,CPU}$	-	-	2	5/2 → $D_{B,CPU}(1+Q_{cpu}^{(1A,0B)})$
$R_{B,Disk}$	-	-	4	7
X_A	-	1/4	-	3/19
X_B	-	-	1/6	2/19
$Q_{A,CPU}$	0	1/4	-	4/19
$Q_{A,Disk}$	0	3/4	-	15/19
$Q_{B,CPU}$	0	-	1/3	5/19
$Q_{B,Disk}$	0	-	2/3	14/19

Modelos de de Filas Fechadas: Solução Aproximada

- Tamanho da fila no instante de chegada visto por um cliente da classe c , $A_{c,k}(N)$, calculado de forma aproximada

$$A_k(\vec{N}) \approx h_c (Q_{1,k}(\vec{N}), \dots, Q_{C,k}(\vec{N}))$$

$$A_{c,k}(\vec{N}) = \left[\frac{N_c - 1}{N_c} Q_{c,k}(\vec{N}) \right] + \sum_{\substack{j=1 \\ j \neq c}}^C Q_{j,k}(\vec{N})$$

Análise de Valores Médios Aproximada (AMVA): Algoritmo

1. Inicialização: $Q_{c,k}(N) = N_c / K$ para todos os centros k

3. Calcule:

$$R_{c,k} = \begin{matrix} D_{c,k} \\ \left[1 + \frac{N_c - 1}{N_c} Q_{c,k}(\vec{N}) + \sum_{\substack{j=1 \\ j \neq c}}^C Q_{j,k}(\vec{N}) \right] \end{matrix}$$

$$X_c = \frac{N_c}{Z_c + \sum_{k=1}^K R_{c,k}} \quad Q_{c,k} = X_c R_{c,k}$$

4. Se $|Q_{c,k}^{\text{new}}(N) - Q_{c,k}(N)| > 0.1\% Q_{c,k}(N)$:

$Q_{c,k}(N) = Q_{c,k}^{\text{new}}(N)$, retorne ao passo 2

Análise de Valores Médios Aproximada (AMVA): Exemplo

$$D_{A,CPU} = 1 \quad D_{A,disk} = 3$$

$$D_{B,CPU} = 2 \quad D_{B,disk} = 4$$

$$N_A = 1 \quad N_B = 1$$

Análise de Valores Médios Aproximada (AMVA): Exemplo

$$D_{A,CPU} = 1 \quad D_{A,disk} = 3$$

$$D_{B,CPU} = 2 \quad D_{B,disk} = 4$$

$$N_A = 1 \quad N_B = 1$$

iteration	class	performance measure			
		$Q_{c,CPU}$	$Q_{c,Disk}$	X_c	R_c
0	A	.500	.500		
	B	.500	.500		
1	A	.250	.750	.167	6.000
	B	.333	.667	.111	9.000
2	A	.211	.790	.158	6.333
	B	.263	.737	.105	9.500
3	A	.195	.805	.154	6.474
	B	.253	.747	.104	9.579
4	A	.193	.807	.154	6.495
	B	.249	.751	.104	9.610
5	A	.192	.808	.154	6.508
	B	.248	.752	.104	9.614
exact solution	A	.211	.789	.158	6.333
	B	.263	.737	.105	9.500

Modelos com Classes Mistas: Solução

Carga: $\vec{I} \equiv (N_1 \text{ ou } \lambda_1, N_2 \text{ ou } \lambda_2, \dots, N_C \text{ ou } \lambda_C,)$

Seja $\{O\}$ o conjunto de classes abertas e $\{C\}$ o conjunto de classes fechadas

1. Calcule a utilização de cada centro k para cada classe aberta $c \in \{O\}$ e a utilização total para as classes abertas
2. Solucione o modelo fechado considerando *apenas* as classes fechadas
3. Calcule tempos de residência e tamanhos das filas para cada classe aberta

Modelos com Classes Mistas: Solução

1. Calcule a utilização de cada centro k para cada classe aberta $c \in \{O\}$ e a utilização total para as classes abertas

$$U_{c,k}(\vec{I}) = \lambda_c D_{c,k} \quad c \in \{O\}$$

$$U_{\{O\},k}(\vec{I}) = \sum_{c \in \{O\}} \lambda_c D_{c,k}$$

Modelos com Classes Mistas: Solução

2. Solucione o modelo fechado considerando *apenas* as classes fechadas

Calcule throughputs, tempos de residência e número médio de clientes nas filas utilizando os tempos de serviços inflacionados pelo fator $1 - U_{\{O\},k}(I)$

$1 - U_{\{O\},k}(I)$ = % tempo recurso está disponível para classes fechadas (ex: velocidade efetiva da CPU)

$$D_{c,k}^* = \frac{D_{c,k}}{1 - U_{\{O\},k}(\vec{I})} \quad c \in \{C\} \quad \begin{array}{l} \text{load} \\ \text{concealment} \end{array}$$

Calcule as utilizações dos dispositivos usando as demandas originais

Modelos com Classes Mistas: Solução

3. Calcule tempos de residência e tamanhos das filas para cada classe aberta

$$R_{c,k} = \frac{D_{c,k} (1 + Q_{\{C\},k}(\vec{I}))}{1 - U_{\{O\},k}(\vec{I})} \quad c \in \{O\}$$

$$Q_{c,k}(\vec{I}) = \lambda_c R_{c,k}(\vec{I}) \quad c \in \{O\}$$

onde

$$Q_{\{C\},k}(\vec{I}) = \sum_{c \in \{C\}} Q_{c,k}$$

Exemplo

$$\lambda_A = 1 \quad \lambda_B = \frac{1}{2}$$

$$N_C = 1 \quad N_D = 1$$

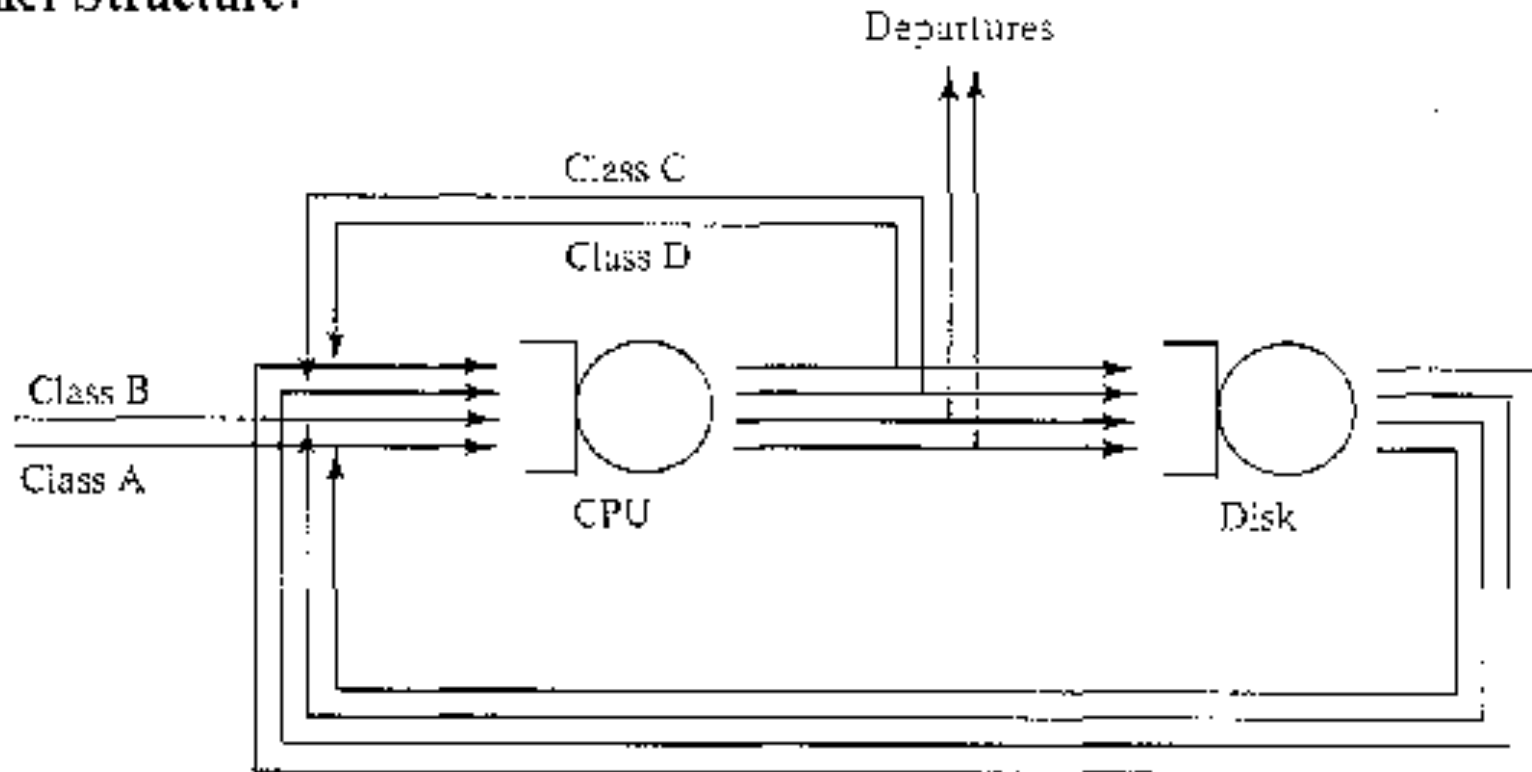
$$D_{A,CPU} = 1/4 \quad D_{A,disk} = 1/6$$

$$D_{C,CPU} = 1/2 \quad D_{C,disk} = 1$$

$$D_{B,CPU} = 1/2 \quad D_{B,disk} = 1$$

$$D_{D,CPU} = 1 \quad D_{D,disk} = 4/3$$

Model Structure:



Exemplo

1. Compute the total utilization of the devices by the open classes:

$$U_{\{O\},CPU}(\bar{I}) = \lambda_A D_{A,CPU} + \lambda_B D_{B,CPU} = .5$$

$$U_{\{O\},Disk}(\bar{I}) = \lambda_A D_{A,Disk} + \lambda_B D_{B,Disk} = .667$$

2. Solve the closed model obtained by deleting the open classes and inflating the service demands of the closed classes:

$$D_{C,CPU}^* = \frac{.5}{1-.5} = 1 \qquad D_{D,CPU}^* = \frac{1}{1-.5} = 2$$

$$D_{C,Disk}^* = \frac{1}{1-.667} = 3 \qquad D_{D,Disk}^* = \frac{1.333}{1-.667} = 4$$

This model is equivalent to the closed model solved in the example of Section 7.4.2, so the same performance measures will result, e.g., CPU queue lengths are .211 and .789 for classes *C* and *D*.

3. Using the queue lengths of the closed classes, compute the performance measures of the open classes. For example:

$$R_{A,CPU} = \frac{.25 (1+1.0)}{1-.5} = 1.0 \qquad R_{B,CPU} = \frac{.5 (1+1.0)}{1-.5} = 2.0$$

Exemplo

Um servidor de banco de dados tem uma CPU e 2 discos. A carga do servidor pode ser dividida em 3 classes: consultas simples (S), consultas complexas (C) e processamento batch (B). Sabe-se também que os arquivos tipicamente consultados estão no disco 1. As demandas por serviço de cada tipo de consulta são dadas na tabela abaixo:

Classe	D_{CPU}	D_{disk1}
S	70 ms	100 ms
C	200 ms	300 ms

Sabe-se que as demandas por CPU e pelos discos 1 e 2 de um processo batch são, em média, 400 ms, 80 ms e 240 ms. Calcule os throughputs e tempos de resposta médios para cada classe quando as taxas de chegadas de consultas simples e complexas são 4 tps e 0.9 tps, respectivamente, e o processamento batch está em pico de operação com um nível de multiprogramação igual a 3.

Exemplo

Um servidor de arquivos, com uma CPU e um disco, serve um número de clientes diskless. A carga de trabalho do servidor consiste de requisições que podem ser agrupadas em três classes: *leitura*, *escrita* e *outros*. Durante 1 hora, foram feitas as seguintes medições

	# req	U_{CPU}	S_{CPU}	U_{disk}	V_{disk}	S_{disk}
Leitura	18000	9	0.006	20	2	0.020
Escrita	7200	18	0.015	20	5	0.020
Outros	3600	5	0.010	8	4	0.020

- (a) Quais são os tempos de resposta e os tempos de espera (na fila) para cada classe?
- (b) Quais os tamanhos médios das filas para cada classe?
- (c) Qual o impacto nos tempos de respostas dos upgrades:
 - (1) Cache de leitura com taxa de acerto de 70%
(assuma atraso no cache desprezível)
 - (2) Adicionar um segundo disco e balancear carga

Exemplo

Um servidor de arquivos com uma cpu e 4 discos serve um ambiente distribuído formado de um número de clientes diskless. Dada a carga imposta por estes clientes, o servidor está balanceado com uma demanda média por dispositivo de 2 segundos. Sabendo que cada cliente executa tarefas locais por 10 segundos, em média, após cada resposta do servidor, qual o máximo de clientes você permitiria no ambiente?