

# Avanços em otimização de redes de filas finitas dependentes do estado

---

Frederico R. B. Cruz e outros  
*Departamento de Estatística,  
Universidade Federal de Minas Gerais,  
31270-901 - Belo Horizonte - MG, Brazil*  
E-mail: `fcruz@ufmg.br`

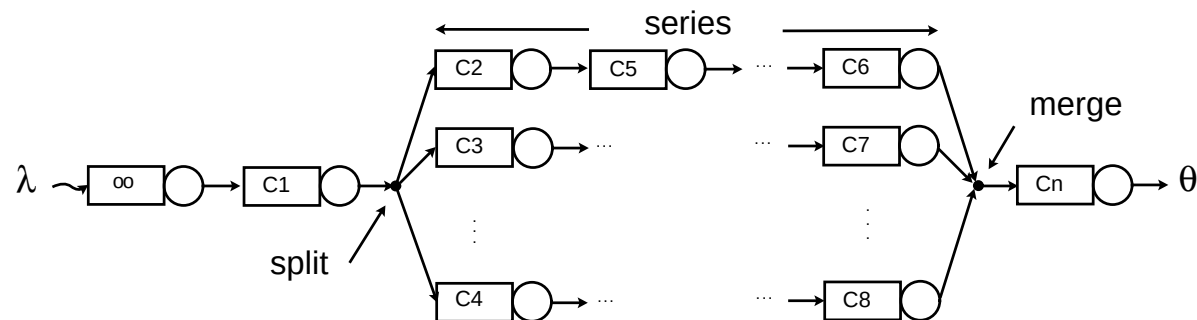
## Resumo

---

Em quase todo sistema físico real há espaço limitado para formação de filas. Por exemplo, em sistemas de manufatura, há limite nas salas de espera entre as estações de trabalho em linhas de montagem, sistemas de manipulação de material e células de manufatura. Em sistemas de telecomunicações, há limite máximo de capacidade de linhas telefônicas e redes de computadores. Finalmente, em sistemas de serviços, há espaço limitado para circulação, atividades e armazenagem. Em todas estas aplicações, é crucial calcular medidas de desempenho. É também importante determinar configurações ótimas, com limitações de espaço em mente. As principais razões desta apresentação são apresentar resultados recentes em sistemas modelados por redes de filas finitas e perspectivas futuras de pesquisa na área.

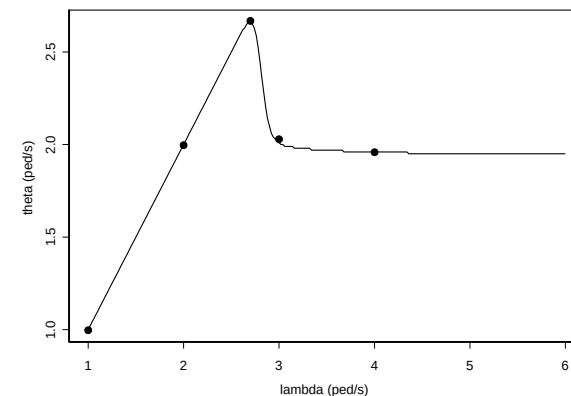
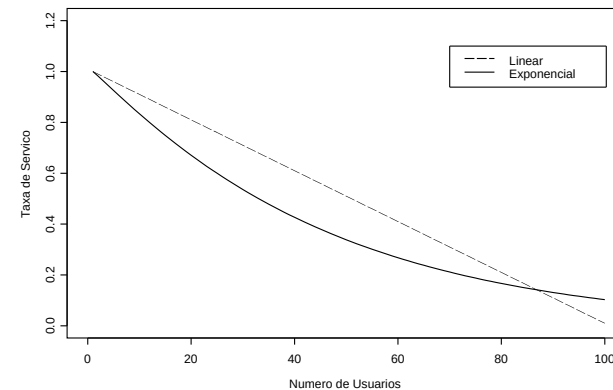
## O quê é uma rede de filas?

- modeladas por grafos  $G = (V, A)$
- $V$  é o conjunto de vértices
- $A$  é o conjunto de arcos



## O quê é uma fila finita dependentes do estado?

- espaço finito para formação de filas, o que leva ao fenômeno de bloqueio (antes ou após serviço)
- a dependência com o estado  $\rightarrow$  congestionamentos
- throughput sui-generis
- úteis em várias situações práticas nas quais ocorre congestionamento



## Análise de desempenho via simulação

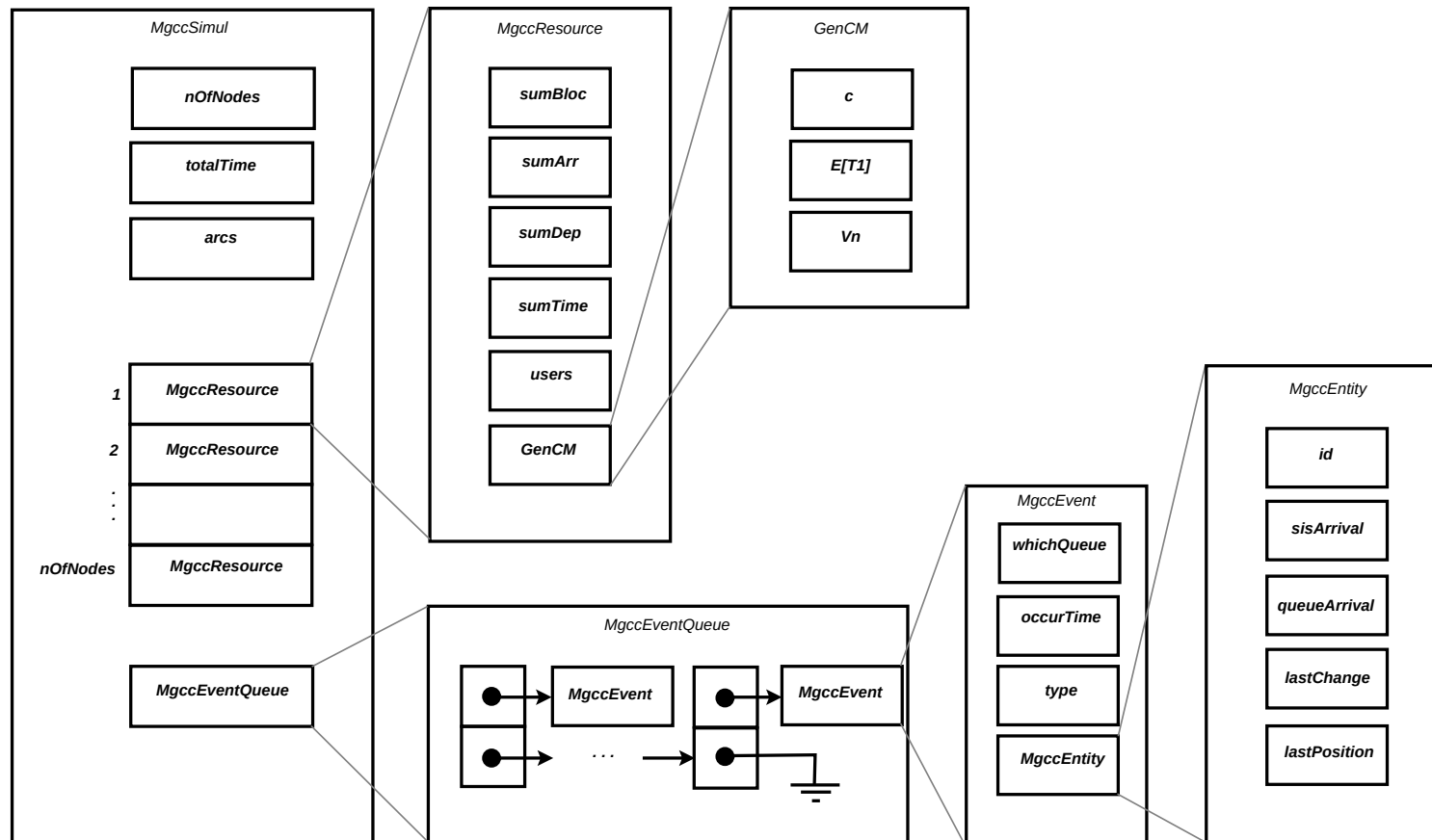
- medidas de desempenho estimadas com precisão via sistemas de simulação a eventos discretos:
  - probabilidade de bloqueio
  - *throughput* (número de atendimentos na unidade de tempo)
  - número médio de usuários
  - tempo médio de serviço

**algorithm** Simulate

```
/* initialize event queue */
Initialize(MgccEventQueue);
/* create and insert 'last' event */
MgccEvent ← new();
MgccEvent.occureTime ← totalTime;
MgccEvent.type ← end_simulation;
Insert(MgccEventQueue, MgccEvent);
/* create and insert 'first' events */
for  $\forall n \mid \lambda_n \neq 0$  do
    MgccEvent ← new();
    MgccEvent.whichQueue ←  $n$ ;
    MgccEvent.occureTime ← 0.0;
    MgccEvent.type ← arrival;
    Insert(MgccEventQueue, MgccEvent);
end for
/* simulate */
MgccEvent ← GetNext(MgccEventQueue);
while MgccEvent.type  $\neq$  end_simulation do
    if MgccEvent.type = arrival then
        ProcessArrival(MgccEventQueue, MgccEvent);
    else if MgccEvent.type = departure then
        ProcessDeparture(MgccEventQueue, MgccEvent);
    end if
    MgccEvent ← GetNext(MgccEventQueue);
end while
print results
end algorithm
```

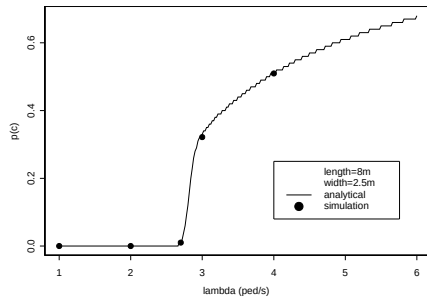
## Análise de desempenho via simul. (cont.)

- proposta de estrutura de dados para sistema de simulação a eventos discretos:

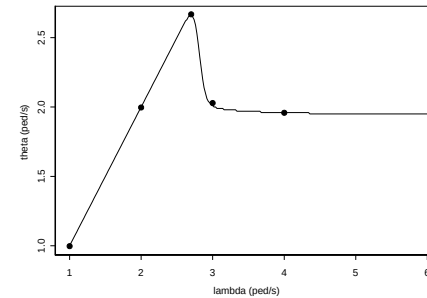


## Análise de desempenho via simul. (cont.)

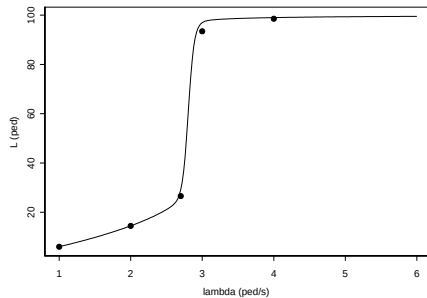
- validação do modelo de simulação:



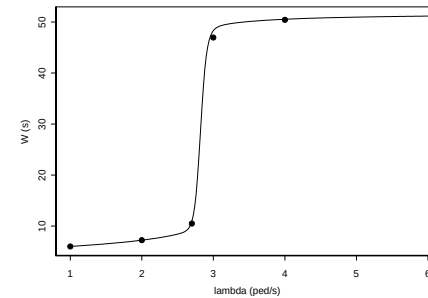
a) probabilidade de bloqueio



b) *throughput*



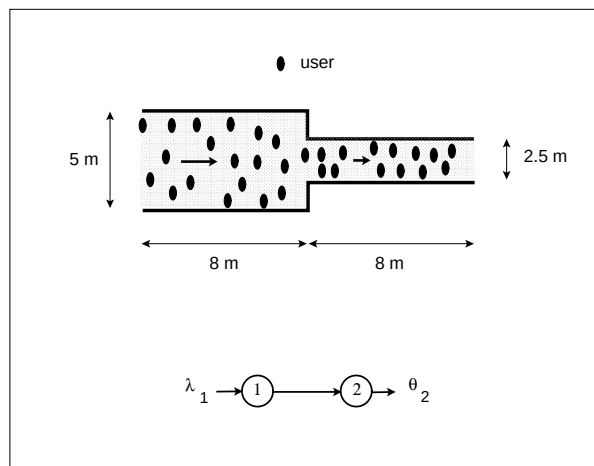
c) número médio de usuários



d) tempo médio de serviço

## Análise de desempenho via simul. (cont.)

- caracterização do tráfego inter-filas:



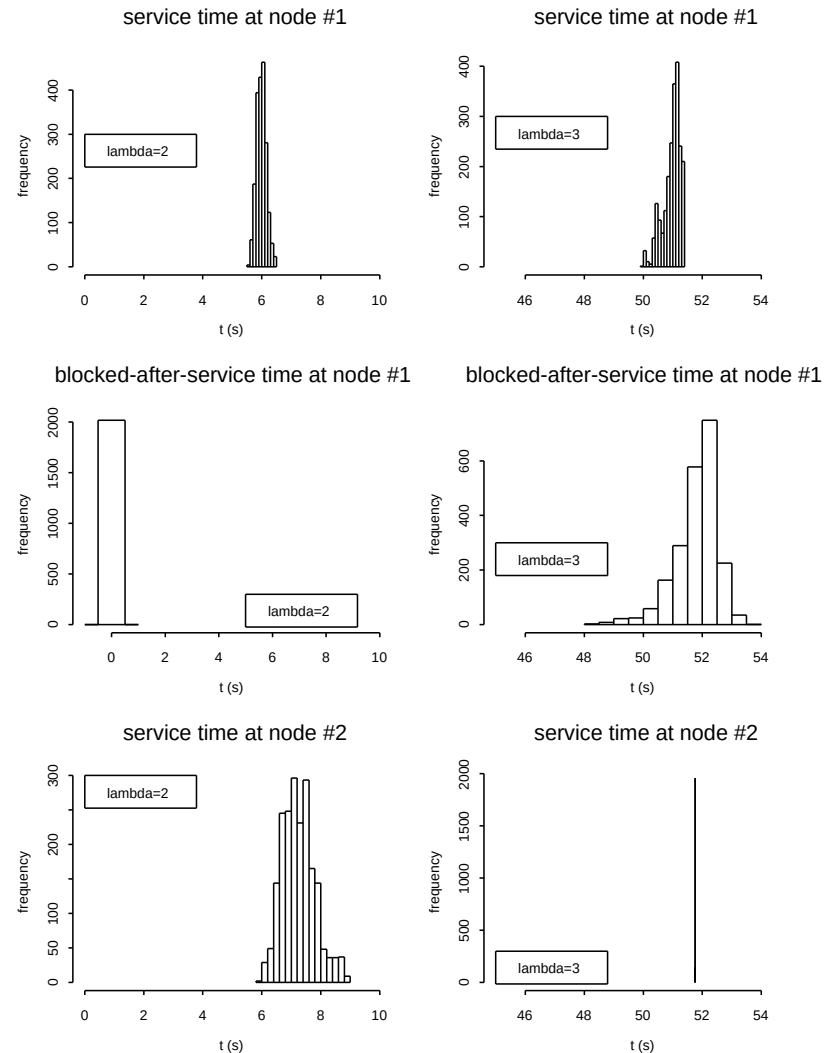
Estatísticas descritivas do tempo entre partidas.

| $\lambda$ | Nó    | Min.     | Q1    | Mediana | Média        | Desv. Pad.   | Q3    | Max. |
|-----------|-------|----------|-------|---------|--------------|--------------|-------|------|
| 2         | 1     | 0.000122 | 0.143 | 0.340   | <b>0.489</b> | <b>0.475</b> | 0.686 | 3.51 |
|           | $h_1$ | 0.000122 | 0.143 | 0.340   | <b>0.489</b> | <b>0.475</b> | 0.686 | 3.51 |
|           | 2     | 0.000122 | 0.147 | 0.337   | <b>0.489</b> | <b>0.476</b> | 0.686 | 3.66 |
| 3         | 1     | 0.000732 | 0.151 | 0.359   | <b>0.517</b> | <b>0.483</b> | 0.738 | 3.25 |
|           | $h_1$ | 0.00214  | 0.144 | 0.344   | <b>0.518</b> | <b>0.487</b> | 0.793 | 2.15 |
|           | 2     | 0.00214  | 0.156 | 0.355   | <b>0.517</b> | <b>0.486</b> | 0.781 | 2.15 |



## Análise de desempenho via simul. (final)

- tempo de serviço é dependente do estado:



# Contribuições analíticas à estimativa de medidas de desempenho

- extensão do Método de Expansão Generalizado a redes finitas dependentes do estado:

```

algorithm
  read  $G(V, A)$ 
  read arrival rates  $\lambda_i, \forall i \in V$ 
  read lengths  $l_i, \forall i \in V$ 
  read routing probabilities  $p_{ij}, \forall (i, j) \in A$ 
  /* pre-evaluate every node */
   $P \leftarrow \emptyset$ 
  while  $P \neq V$ 
    choose  $k \in (V \setminus P)$ 
    if  $\{i | (i, k) \in A\} \subseteq P$  then
      /* compute performance measures */
       $E[T_1]_k \leftarrow l_k / V_1$ 
      compute  $\Pr\{N = c_k\}, \theta_k, L_k, W_k$ 
       $P \leftarrow P \cup \{k\}$ 
      /* forward information */
       $\lambda_j \leftarrow \lambda_j + p_{kj} \theta_k, \forall (k, j) \in A$ 
    end if
  end while
  /* evaluate every node */
  to be cont. ...

```

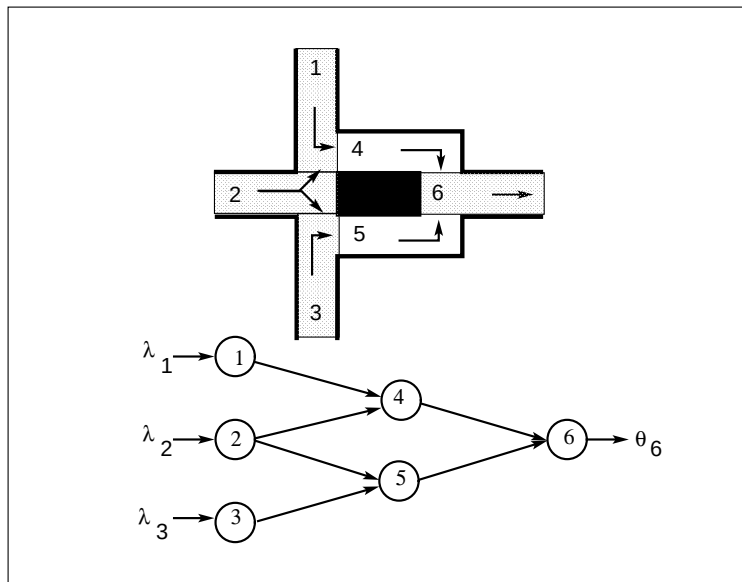
```

algorithm (cont.)
  /* evaluate every node */
   $P \leftarrow \emptyset$ 
   $\theta_i^{\max} \leftarrow \infty, \forall i \in V$ 
  while  $P \neq V$ 
    choose  $k \in (V \setminus P)$ 
    if  $\{j | (k, j) \in A\} \subseteq P$  then
      /* update performance measures */
       $E[T_1]_k \leftarrow \min E[T_1]_k$ 
      s.t.:
         $\theta_k \leq \theta_k^{\max},$ 
         $E[T_1]_k \geq l_k / V_1$ 
      compute  $\Pr\{N = c_k\}, \theta_k, L_k, W_k$ 
       $P \leftarrow P \cup \{k\}$ 
      /* back-propagate information */
      update  $\theta_i^{\max}, \forall (i, k) \in A$ 
    end if
  end while
  write  $\Pr\{N = c_i\}, \theta_i, L_i, W_i, \forall i \in V$ 
end algorithm

```

## Contribuições analíticas à estimativa de medidas de desempenho (cont.)

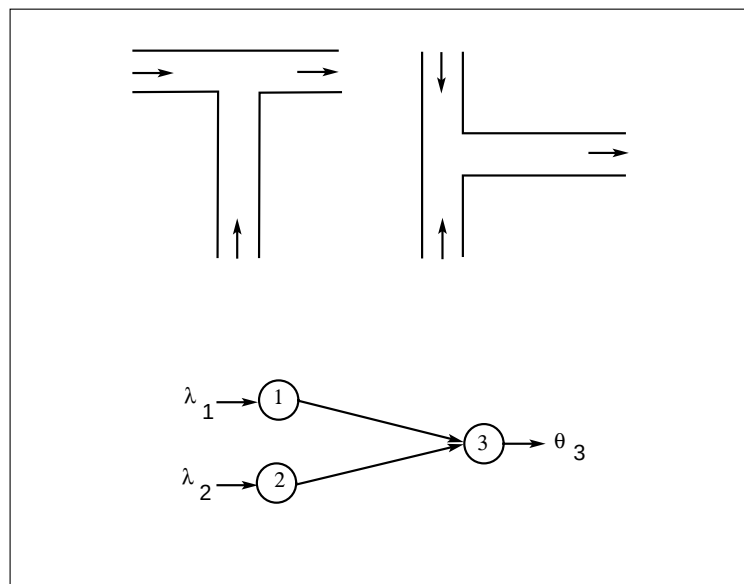
- topologia exemplo:



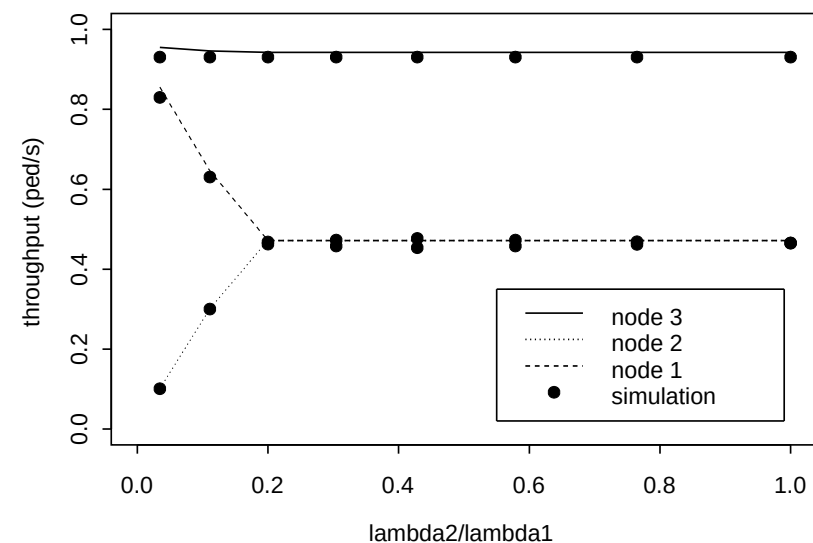
- sequência de pré-avaliação:  
 $1 \rightarrow 2 \rightarrow 4 \rightarrow 3 \rightarrow 5 \rightarrow 6$
- sequência de avaliação:  
 $6 \rightarrow 4 \rightarrow 5 \rightarrow 1 \rightarrow 2 \rightarrow 3$

## Contribuições analíticas à estimativa de medidas de desempenho (cont.)

- topologia de interesse:



- resultados obtidos:

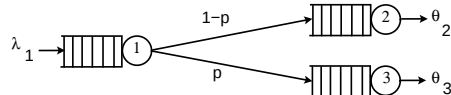


## Contribuições analíticas ... (cont.)

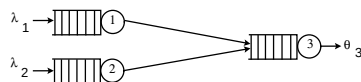
- análise de topologias básicas:



| Medida   | nó 1  |       |                 | nó 2  |       |                 | nó 3  |       |                 |
|----------|-------|-------|-----------------|-------|-------|-----------------|-------|-------|-----------------|
|          | Anal. | Sim.  | 95% CI          | Anal. | Sim.  | 95% CI          | Anal. | Sim.  | 95% CI          |
|          |       |       |                 |       |       |                 |       |       |                 |
| $p(c)$   | 0.33  | 0.33  | [ 0.32 ; 0.33 ] | 0.00  | 0.01  | [ 0.00 ; 0.01 ] | 0.00  | 0.01  | [ 0.00 ; 0.03 ] |
| $\theta$ | 2.01  | 2.02  | [ 2.01 ; 2.03 ] | 2.01  | 2.02  | [ 2.01 ; 2.02 ] | 2.01  | 2.02  | [ 2.01 ; 2.02 ] |
| $L$      | 96.96 | 95.87 | [95.35;96.40]   | 14.56 | 16.44 | [15.00;17.88]   | 14.56 | 16.51 | [15.23;17.79]   |
| $W$      | 48.31 | 47.53 | [47.10;47.96]   | 7.26  | 8.15  | [ 7.44 ; 8.86 ] | 7.26  | 8.19  | [ 7.56 ; 8.81 ] |



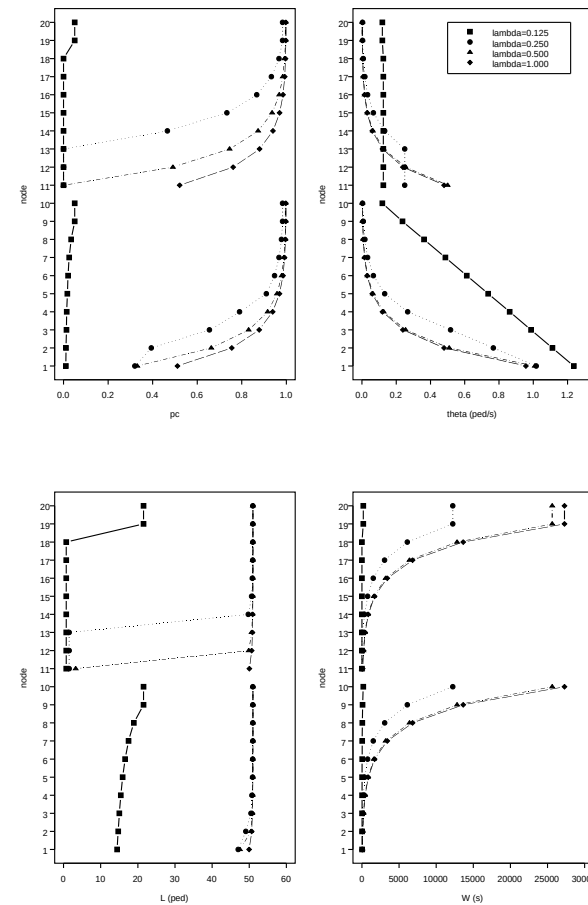
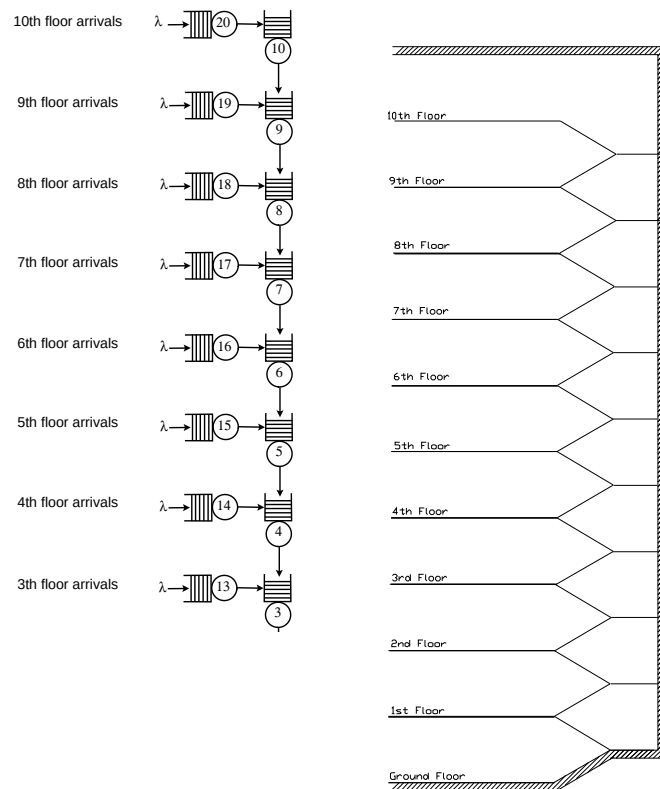
| Medida   | nó 1   |       |                 | nó 2   |      |             | nó 3   |      |             |
|----------|--------|-------|-----------------|--------|------|-------------|--------|------|-------------|
|          | Anal.  | Sim.  |                 | Anal.  | Sim. |             | Anal.  | Sim. |             |
|          | 95% CI |       |                 | 95% CI |      |             | 95% CI |      |             |
|          |        |       |                 |        |      |             |        |      |             |
| $p(c)$   | 0.33   | 0.32  | [ 0.32 ; 0.33 ] | 0.00   | 0.00 | [0.00;0.00] | 0.00   | 0.00 | [0.00;0.00] |
| $\theta$ | 2.01   | 2.03  | [ 2.02 ; 2.04 ] | 1.20   | 1.22 | [1.21;1.23] | 0.80   | 0.81 | [0.81;0.82] |
| $L$      | 96.96  | 95.04 | [94.23;95.84]   | 7.48   | 7.61 | [7.55;7.67] | 4.70   | 4.76 | [4.73;4.80] |
| $W$      | 48.31  | 46.82 | [46.16;47.49]   | 6.21   | 6.24 | [6.23;6.26] | 5.86   | 5.87 | [5.86;5.88] |



| Medida   | nó 1   |       |                 | nó 2   |       |                 | nó 3   |       |                 |
|----------|--------|-------|-----------------|--------|-------|-----------------|--------|-------|-----------------|
|          | Anal.  | Sim.  |                 | Anal.  | Sim.  |                 | Anal.  | Sim.  |                 |
|          | 95% CI |       |                 | 95% CI |       |                 | 95% CI |       |                 |
| $p(c)$   | 0.67   | 0.68  | [ 0.68 ; 0.68 ] | 0.67   | 0.68  | [ 0.68 ; 0.68 ] | 0.51   | 0.50  | [ 0.50 ; 0.50 ] |
| $\theta$ | 0.98   | 0.97  | [ 0.97 ; 0.97 ] | 0.98   | 0.97  | [ 0.97 ; 0.97 ] | 1.96   | 1.93  | [ 1.93 ; 1.93 ] |
| $L$      | 99.51  | 98.63 | [98.60;98.67]   | 99.51  | 98.61 | [98.56;98.67]   | 99.02  | 99.76 | [99.75;99.76]   |
| $W$      | 101.6  | 102.0 | [101.8;102.2]   | 101.6  | 101.9 | [101.8;102.1]   | 50.54  | 51.70 | [51.70;51.71]   |

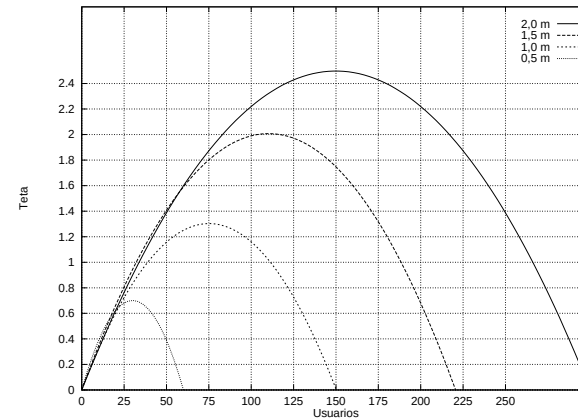
# Contribuições analíticas à estimativa de medidas de desempenho (final)

- resultados obtidos para um rede de evacuação:



## Alocação de recursos em redes de filas finitas

- problema de programação matemática estocástica
- throughput dependente do número de usuários no sistema
- desconhecida expressão analítica para estimação do throughput



# Experiência com alocação de capacidades

- formulação matemática:

$$z = \min \left[ g(\mathbf{c}) = \sum_{\forall i \in V} f_i c_i \right],$$

**S.a.:**

$$p_{c_i} \leq \varepsilon, \quad \forall i \in V,$$

$$c_i \in \{1, \dots\}, \quad \forall i \in V.$$

- algoritmo:

```

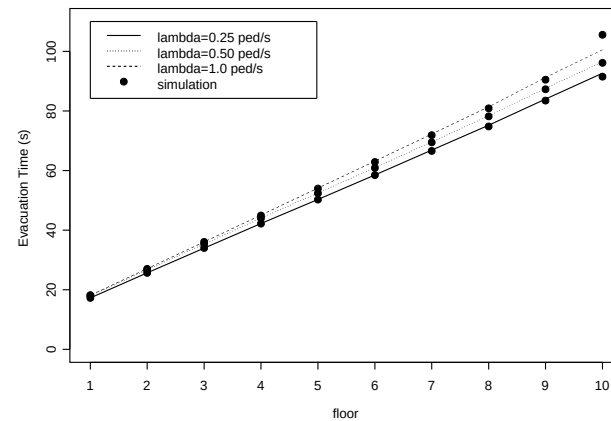
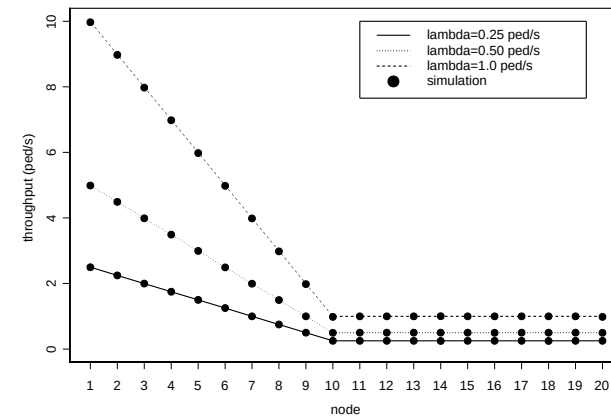
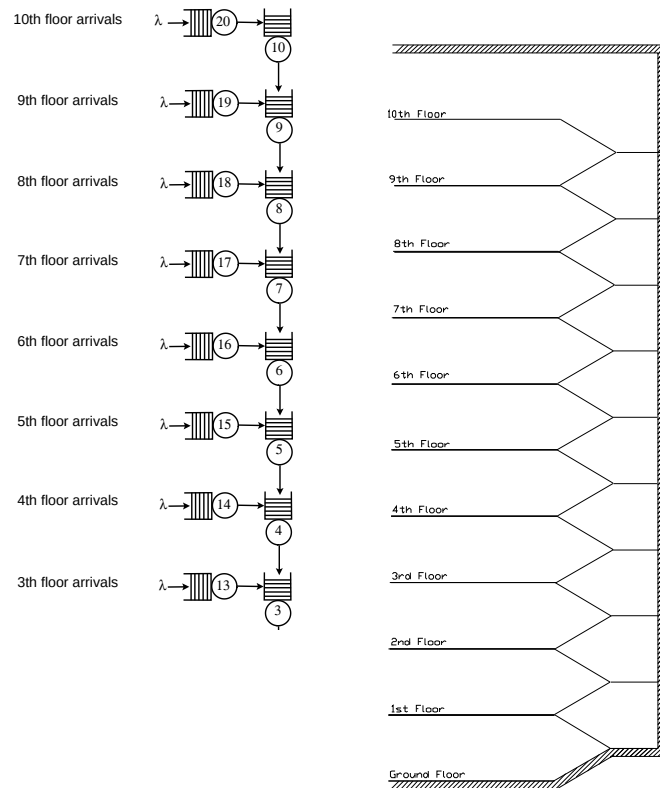
algorithm
  read  $G(V, A)$ 
  read routing probabilities  $p_{ij}, \forall (i, j) \in A$ 
  read arrival rates  $\lambda_i$  and  $f_i, \forall i \in V$ 
  /* find an initial feasible vector  $\mathbf{c}$  */
  for  $\forall i \in V$  do
     $c_i^{\text{opt}} \leftarrow c_i^{\text{sup}} \leftarrow 2^M$ 
  end for
  /* search optimum solution */
  iter  $\leftarrow 0$ 
  repeat
    iter  $\leftarrow$  iter + 1
    for  $\forall i \in V$  do
      /* optimizes  $i$ th queue */
      OptQueue( $i, \mathbf{c}^{\text{sup}}$ )
    end for
    /* update best solution */
    if  $g(\mathbf{c}^{\text{sup}}) < g(\mathbf{c}^{\text{opt}})$  then
       $\mathbf{c}^{\text{opt}} \leftarrow \mathbf{c}^{\text{sup}}$ 
      unmark all nodes
    else
      exit
    end if
  end repeat
  write  $\mathbf{c}^{\text{opt}}$ 
end algorithm

```



# Experiência com alocação de capacidades

- resultados obtidos:



## Equipe de trabalho

---

- Estão envolvidas no projeto alunos de graduação e de pós-graduação:
  - aplicação em redes de telefonia móvel
  - caracterização de tráfego em redes de filas finitas dependentes do estado
- Trabalho futuros incluem:
  - extensão a outras aplicações, tais como tráfego urbano de veículos e a estudo de área de circulação em sistemas produtivos
  - problemas de alocação de tráfego
  - alocação de capacidade multi-objetivo
  - ...

## Conclusões e observações finais

---

- apresentados problemas de pesquisa em redes de filas finitas dependentes do estado:
  - métodos de simulação e analíticos para análise de desempenho em redes de filas finitas dependentes do estado
  - alocação de serviço e capacidades em redes de filas
- reforçada a aplicabilidade dos modelos
- propostas de trabalhos futuros

---

Dúvidas, críticas, sugestões?

---