

UNIVERSIDADE ESTADUAL DE CAMPINAS  
INSTITUTO DE MATEMÁTICA, ESTATÍSTICA E COMPUTAÇÃO  
CIENTÍFICA

# Simulação perfeita para redes com perdas

Nevena Marić

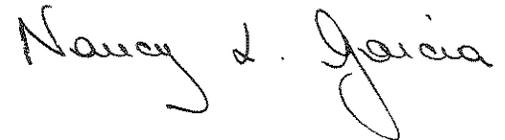
Dissertação de Mestrado orientada pela  
Profa.Dra. Nancy Lopes Garcia

# Simulação perfeita para redes com perdas

Este exemplar corresponde à redação final da dissertação devidamente corrigida e defendida por Nevena Maric e aprovada pela comissão julgadora.

Campinas, 21 de fevereiro de 2002.

Profa. Dra. Nancy Lopes Garcia



Banca Examinadora:

1. Profa. Dra. Nancy Lopes Garcia (IMECC - UNICAMP)
2. Prof. Dr. Pablo Augusto Ferrari (IME - USP)
3. Prof. Dr. Fábio Prates Machado (IME - USP)

Dissertação apresentada ao Instituto de Matemática, Estatística e Computação Científica, UNICAMP, como requisito parcial para obtenção do Título de MESTRE em Estatística.

**UNICAMP**  
BIBLIOTECA CENTI  
SEÇÃO CIRCULAN

UNIDADE B0  
Vº CHAMADA T/UNICAMP  
M337A  
/ EX  
COMBO BCI 49119  
ROC 16-837102  
/ DX  
REÇO R\$ 11,00  
ATA \_\_\_\_\_  
º CPD \_\_\_\_\_

CM00167672-3

B ID 241035

**FICHA CATALOGRÁFICA ELABORADA PELA  
BIBLIOTECA DO IMECC DA UNICAMP**

Maric, Nevena

M337s Simulação perfeita para redes com perdas. / Nevena Maric --  
Campinas, [S.P. :s.n.], 2002.

Orientador : Nancy Lopes Garcia

Dissertação (mestrado) - Universidade Estadual de Campinas,  
Instituto de Matemática, Estatística e Computação Científica.

1. Percolação (Física estatística). 2. Métodos de simulação. 3.  
Processos estocásticos. I. Garcia, Nancy Lopes. II. Universidade  
Estadual de Campinas. Instituto de Matemática, Estatística e  
Computação Científica. III. Título.

Dissertação de Mestrado defendida em 18 de fevereiro de 2002 e aprovada pela  
Banca Examinadora composta pelos Profs. Drs.



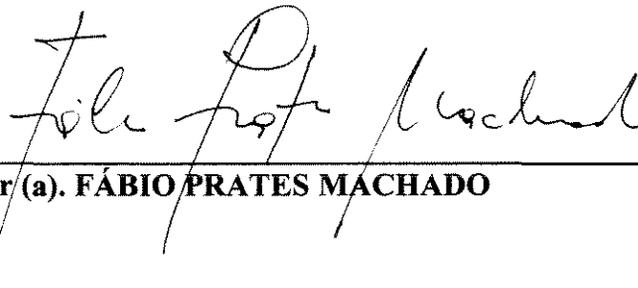
---

Prof (a). Dr (a). NANCY LOPES GARCIA



---

Prof (a). Dr (a). PABLO AUGUSTO FERRARI



---

Prof (a). Dr (a). FÁBIO PRATES MACHADO

100222352

## Resumo

O processo de redes com perdas é um processo espacial de nascimento e morte. Este processo modela, por exemplo, a ocorrência de chamadas numa rede telefônica. O processo possui uma única medida invariante e é ergódico desde que o processo de percolação orientada associado seja sub-crítico. Nós achamos uma condição suficiente para tal sub-criticalidade.

O algoritmo *Backward-Forward* (BFA) é uma das técnicas para simular perfeitamente. Este algoritmo é aplicável para qualquer medida que (1) seja absolutamente contínua com respeito a um processo pontual de Poisson e (2) possa ser obtida como uma medida invariante de um processo espacial de nascimento e morte.

Neste trabalho, aplicamos o BFA a fim de obter uma amostra exata de medida invariante do processo de redes com perdas. As simulações servem também para obter informações adicionais sobre o processo e, além do mais nos levam a sugerir uma condição melhor para sub-criticalidade mencionada.

## Abstract

The loss network process is a spatial birth-and-death process. This process models, for example, the calls occurrence in a telephone network. The process has a unique invariant measure and is ergodic as long as the associated oriented percolation process is sub-critical. We found a sufficient condition for such sub-criticality.

The *Backward-Forward* algorithm (BFA) is a perfect simulation scheme. The algorithm is applicable to any measure which (1) is absolutely continuous with respect to a Poisson point process, and (2) can be obtained as the invariant measure of a spatial interacting birth-and-death process.

In this paper we apply the BFA in order to obtain an exact sample from invariant measure of the loss network process. We also use the simulations to obtain some additional information about the process, and moreover they allow to suggest a better condition for the sub-criticality.

## Agradecimentos

Agradeço especialmente a professora Nancy pelo apoio e pela orientação dedicada e competente deste trabalho.

Agradeço aos professores Hervé e Hotta pelo apoio e ajuda.

Agradeço aos todos professores e colegas de mestrado pelo várias ajudas e paciência.

Agradeço à FAPESP e à CAPES pelo suporte financeiro.

# Índice

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>Processos de interesse</b>	<b>3</b>
2.1	Processos pontuais . . . . .	3
2.1.1	Processo pontual de Poisson . . . . .	4
2.1.2	Processo pontual marcado . . . . .	4
2.2	Processos espaciais de nascimento e morte (em $\mathbb{R}^d$ ) . . . . .	6
2.3	Processo de ramificação multitipo . . . . .	8
2.3.1	Processo de Galton-Watson multitipo . . . . .	8
<b>3</b>	<b>Redes com perdas - caso contínuo, não limitado, unidimensional</b>	<b>11</b>
3.1	Introdução . . . . .	11
3.2	Construção do processo . . . . .	12
3.2.1	Representação gráfica . . . . .	13
3.2.2	Construção em volume finito . . . . .	13
3.2.3	Construção em volume infinito . . . . .	15
3.3	Um processo de ramificação dominando clã de ancestrais. Valor crítico. . .	16
3.4	Um caso particular: $\pi$ é uniforme discreta . . . . .	24
<b>4</b>	<b>Algoritmo Backward-Forward (BFA) &amp; Redes com perdas</b>	<b>28</b>
4.1	O Algoritmo “Backward-Forward”(BFA) . . . . .	28
4.2	Aplicação ao processo de redes com perdas . . . . .	29
4.2.1	Simulações . . . . .	32
4.3	Uma análise sobre o comportamento do clã de ancestrais usando as simu- lações . . . . .	40
4.3.1	Lambda crítico via simulação . . . . .	45

5	Conclusão	51
A	Provas	52
A.1	Prova do Lema 3.4.1 . . . . .	52
B	Tabelas	55
C	Programas	64

# Lista de Tabelas

4.1	Lambda crítico obtido via simulações para comprimento de chamada fixo .	48
4.2	Lambda crítico obtido via simulações para distribuições <i>Uniforme</i> e <i>Beta</i>	49
B.1	$E(N)$ estimado para comprimento de chamada fixo $d = 0.3, 0.5, 0.7, 1, 1.2$ . Parte 1. . . . .	56
B.2	$E(N)$ estimado para comprimento de chamada fixo $d = 0.3, 0.5, 0.7, 1, 1.2$ . Parte 2. . . . .	57
B.3	$E(N)$ estimado para comprimento de chamada fixo $d = 1.4, 1.6, 1.8, 2, 2.5$ .	58
B.4	$E(N)$ estimado para comprimento de chamada fixo $d = 3, 3.5, 4, 4.5, 5$ .	59
B.5	$E(N)$ estimado para distribuições <i>Beta</i> . . . . .	60
B.6	$E(N)$ estimado para distribuição $U(0, 1)$ . . . . .	61
B.7	Comprimento fixo $d$ ; Lambda crítico obtido; Grau de polinômio usado para aproximar $1/\log(E(N))$ ; $SQ$ -soma de quadrados de resíduos obtidos usando aproximação polinomial . . . . .	62
B.8	Distribuições <i>Uniforme</i> e <i>Beta</i> ; Lambda crítico obtido; Grau de polinômio usado para aproximar $1/\log(E(N))$ ; $SQ$ -soma de quadrados de resíduos obtidos usando aproximação polinomial . . . . .	62
B.9	Resíduos obtidos ajustando modelos lineares entre $\lambda_c$ e $\lambda_c^*$ , com intercepto e sem intercepto; comprimento fixo $d$ . . . . .	63
B.10	Resíduos obtidos ajustando modelos lineares entre $\lambda_c$ e $\lambda_c^*$ , com intercepto e sem intercepto; distribuições <i>Uniforme</i> e <i>Beta</i> . . . . .	63

# Lista de Figuras

4.1	O clã de ancestrais para $U(0, 1)$ , $\lambda = 0.5$ , $\Lambda = [0, 10]$ . . . . .	33
4.2	Resultado de limpeza para $C = 1$ do clã na Figura 4.1. . . . .	33
4.3	O clã de ancestrais para $U(0, 1)$ , $\lambda = 0.9$ , $\Lambda = [0, 10]$ . . . . .	34
4.4	Resultado de limpeza do clã na Figura 4.3 para a) $C = 1$ b) $C = 2$ . . . . .	35
4.5	O clã de ancestrais para $U(0, 1)$ , $\lambda = 1$ , $\Lambda = [0, 10]$ . . . . .	36
4.6	Resultado de limpeza do clã na Figura 4.5 para a) $C = 1$ b) $C = 2$ . . . . .	37
4.7	O clã de ancestrais para $U(0, 1)$ , $\lambda = 1.2$ , $\Lambda = [0, 8]$ . . . . .	38
4.8	Resultado de limpeza do clã na Figura 4.7 para a) $C = 1$ b) $C = 3$ . . . . .	39
4.9	O número total de retângulos ( $E(N)$ ) no clã de ramificação e de rede com perdas para a) $U(0, 1)$ b) $d = 0.5$ . . . . .	41
4.10	Comprimento Temporal ( $E(TL)$ ) do clã de ramificação e de rede com perdas para a) $U(0, 1)$ b) $d = 0.5$ . . . . .	42
4.11	Largura espacial ( $E(SW)$ ) do clã de ramificação e de rede com perdas para a) $U(0, 1)$ b) $d = 0.5$ . . . . .	43
4.12	Histograma de TL para $U(0, 1)$ e $\lambda = 1$ . . . . .	44
4.13	Histograma de SW para $U(0, 1)$ e $\lambda = 1$ . . . . .	44
4.14	Histograma de N para $U(0, 1)$ e $\lambda = 1$ . . . . .	45
4.15	Número total de retângulos ( $E(N)$ ) para $d = 0.5$ . . . . .	46
4.16	$1/\log(E(N))$ para $d = 0.5$ . . . . .	46
4.17	Lambda crítico obtido via simulações para comprimento de chamada fixo . . . . .	47
4.18	$\lambda_c^*$ vs. $\lambda_c$ para comprimento de chamada fixo . . . . .	47

# Capítulo 1

## Introdução

O modelo de redes com perdas unidimensional, contínuo e não limitado foi introduzido por Kelly (1991). Clientes nesta rede estão arrançados ao longo de um cabo infinitamente longo e cada chamada entre dois pontos  $s_1, s_2 \in \mathbb{R}$  no cabo envolve somente o segmento compreendido entre eles. O cabo tem capacidade limitada, digamos  $C$  chamadas. Isto é, uma chamada entre  $s_1$  e  $s_2 \in \mathbb{R}$ ,  $s_1 < s_2$ , é perdida se ao longo de qualquer ponto entre  $s_1$  e  $s_2$ , o cabo já estiver carregando  $C$  chamadas. Chamadas chegam de acordo com um processo de Poisson homogêneo de taxa  $\lambda$  ao longo do cabo, e sua distância (espaço) segue uma distribuição  $\pi$  com média  $\rho$  e segundo momento  $\nu$ . A duração (tempo) de cada chamada tem distribuição exponencial de taxa 1. A localização da chamada, seu comprimento e sua duração são variáveis aleatórias independentes. Kelly (1991) fez uma conjectura que o processo assim definido sempre possui uma única medida invariante construída a seguir. Seja  $X(s)$  o número de clientes no tempo  $s$  em uma fila  $M/G/\infty$  com taxa de chegada  $\lambda$  e distribuição de serviço  $\pi$ . Condiciona-se o processo  $\{X(s), s \in \mathbb{R}\}$  ao evento  $\{X(s) \leq C, s \in [-T, T]\}$  e deixa-se  $T \rightarrow \infty$ . O processo condicionado no limite seria a única medida invariante para o processo de redes com perdas.

Ferrari e Garcia (1998) usaram o argumento de percolação não orientada para provar que o processo de redes com perdas existe e além disso possui uma única medida invariante desde que terceiro momento de distribuição  $\pi$  seja finito e taxa de chegada  $\lambda$  seja suficientemente pequena. Eles também mostraram que a medida invariante, quando existir, é a mesma sugerida por Kelly.

Este argumento de percolação foi refinado por Fernández, Ferrari e Garcia (2000) para obter como condição para ergodicidade

$$\lambda(\rho + \sup_{L \sim \pi} L) < 1. \tag{1.1}$$

O argumento é baseado em uma representação gráfica do processo e é a base do esquema

de simulação perfeita “*Backward-Forward*”, introduzido no mesmo trabalho de Fernández et al. (2000).

O algoritmo “*Backward-Forward*” construi um processo livre de nascimento e morte em direção ao passado. Este passo produz um “clã de ancestrais” para o qual, então, aplica-se um procedimento de “limpeza”. A possibilidade de aplicar o algoritmo é garantida desde que o “clã de ancestrais” seja finito. A mesma condição garante a existência da única medida invariante do processo em questão.

Neste trabalho nós melhoramos a condição (1.1). Usando a dominação pelo um processo de ramificação multitípo, achamos a condição suficiente

$$\lambda < \frac{1}{\rho + \sqrt{\nu}}. \quad (1.2)$$

Isto amplia a região de aplicabilidade do algoritmo “*Backward-Forward*” para o processo de redes com perdas. Implementamos então este algoritmo e o melhoramos a velocidade da execução .

Foram feitas várias simulações baseadas neste algoritmo as quais nos ajudaram a obter mais informações sobre o clã de percolação orientada associado ao processo. Em particular, nossas simulações mostram falta de percolação para valores de  $\lambda$  significativamente maiores do que o valor obtido teoricamente (1.2). Usando os resultados obtidos pelas simulações , sugerimos a expressão para o valor crítico do processo de percolação orientada. Nossa conjectura é

$$\lambda_c = c \cdot \frac{1}{\rho + \sqrt{\nu}} \quad (1.3)$$

onde  $c$  é uma constante aproximadamente 2.82.

# Capítulo 2

## Processos de interesse

### 2.1 Processos pontuais

Um processo pontual modela a distribuição aleatória de pontos indistinguíveis em algum espaço, por exemplo  $\mathbb{R}^d$ . Processos pontuais podem ser considerados tanto como conjuntos aleatórios de pontos discretos, quanto medidas aleatórias de contagem, contando o número de pontos pertencentes às regiões espaciais.

**Definição 1** *Um processo pontual  $N$  em  $\mathbb{R}^d$  é uma função mensurável do espaço de probabilidade  $(\Omega, \mathcal{A}, P)$  ao espaço mensurável  $(\mathcal{N}_{\mathbb{R}^d}, \mathcal{B}(\mathcal{N}_{\mathbb{R}^d}))$*

onde  $\mathcal{N}_{\mathbb{R}^d}$  é o espaço de todas as medidas finitas em conjuntos limitados ( medidas de Radon) e com valores em inteiros, *medidas de contagem*, definidas nos Borelianos,  $\mathcal{B}(\mathbb{R}^d)$ . Por outro lado, pode-se considerar  $\mathcal{N}_{\mathbb{R}^d}$  como uma família de todos os subconjuntos  $N = \{x_1, x_2, \dots\}$  de  $\mathbb{R}^d$ , os quais satisfazem as seguintes condições:

- o conjunto  $N$  é *localmente finito* ( um subconjunto de  $\mathbb{R}^d$ , limitado, pode conter apenas um número finito de pontos )
- o conjunto  $N$  é *simples* (  $x_i \neq x_j$ , se  $i \neq j$  )

Intuitivamente, de acordo com Stoyan, Kendall e Mecke (1987) um processo pontual pode ser considerado como uma escolha aleatória entre os elementos de  $\mathcal{N}_{\mathbb{R}^d}$ . Aqui, identificamos um processo pontual  $N$  com uma medida de contagem  $N$ . A distribuição do processo  $N$  é determinada pelas distribuições de v.as  $N(A)$  (número de pontos de  $N$  em  $A$ ) , para todo  $A \in \mathcal{B}(\mathbb{R}^d)$  limitado.

### 2.1.1 Processo pontual de Poisson

Na teoria dos processos pontuais o processo de Poisson é um dos mais importantes. Além de modelar vários fenômenos naturais ele é bem simples sob o ponto da vista computacional.

**Definição 2** *Seja  $\nu$  uma medida de Radon em  $\mathbb{R}^d$ . Um processo pontual  $N_\nu$  no  $\mathbb{R}^d$  é um processo de Poisson com medida média  $\nu$  se o espaço de estados é  $\mathcal{N} = \{N \in \{0,1\}^{\mathbb{R}^d} \mid N(x) = 1 \text{ somente para um conjunto enumerável de } x \in \mathbb{R}^d\}$ , e, definindo  $N_\nu(A) = \int_A N_\nu(dx)$ , vale:*

1. Para quaisquer conjuntos, disjuntos,  $A_1, \dots, A_k \in \mathcal{B}(\mathbb{R}^d)$ , as variáveis aleatórias  $N_\nu(A_1), \dots, N_\nu(A_k)$  são independentes e
2. Para qualquer  $A \in \mathcal{B}(\mathbb{R}^d)$  e  $k \geq 0$

$$P[N_\nu(A) = k] = \frac{e^{-\nu(A)} \nu(A)^k}{k!}. \quad (2.1)$$

O processo de Poisson é dito ser  $\lambda$ -homogêneo se  $\nu = \lambda m_d$ , onde  $\lambda$  é um constante e  $m_d$  a medida de Lebesgue no  $\mathbb{R}^d$ .

#### Algoritmo PP- Simulação de um processo de Poisson $\lambda$ -homogêneo

1. Para cada janela finita  $\Lambda$ , gerar uma variável aleatória  $K$ , com distribuição  $Poisson(\lambda m_d(\Lambda))$ .
2. Dado  $K = k$ , gerar  $U_1, \dots, U_k$  v.a's independentes e distribuídas uniformemente em  $\Lambda$ .
3. Repetir independentemente para janelas disjuntas.

### 2.1.2 Processo pontual marcado

Um processo pontual se torna um processo pontual marcado juntando uma característica (marca) a cada ponto do processo.

**Definição 3** *Processo pontual marcado em  $\mathbb{R}^d$  com marcas em  $\mathcal{M}$  (um espaço métrico, completo e separável), é um processo pontual em  $\mathbb{R}^d \times \mathcal{M}$  tal que o processo marginal das locações  $\{N(A \times \mathcal{M}) \mid A \in \mathcal{B}(\mathbb{R}^d)\}$  seja um processo pontual em  $\mathbb{R}^d$ .*

Também podemos representar um processo pontual marcado em  $\mathbb{R}^d$  como uma seqüência aleatória  $\{[x_n; m(x_n)]\}$  onde os pontos  $x_n$  constituem um processo pontual em  $\mathbb{R}^d$ , e  $m(x_n)$  são as marcas corespondentes a cada  $x_n$ . Exemplos:

1.  $x$ - o centro da partícula,  $m(x)$ - o volume da partícula
2.  $x$ - a locação do conjunto aleatório,  $m(x)$ - o próprio conjunto aleatório

Um exemplo importante são *processos pontuais marcados completamente independentes*. Diz-se que um processo pontual marcado exhibe a propriedade de independência completa quando  $n$  variáveis do conjunto

$$\{N(A_i \times B_i) \mid A_i \in \mathcal{B}(\mathbb{R}^d), B_i \in \mathcal{B}(\mathcal{M}) \text{ limitados } (i = 1, \dots, n)\}$$

são mutuamente independentes, sempre que os  $A_i$  sejam disjuntos. Esse processo é totalmente especificado por dois componentes:

- um processo de Poisson das locações  $N(\cdot \times \mathcal{M})$
- uma família das distribuições de probabilidade  $\{P(\cdot|x), x \in \mathbb{R}^d\}$  dando a distribuição da marca em  $\mathcal{M}$ .

As marcas podem ser bem complicadas como no segundo exemplo acima, ou seja na interpretação do processo pontual marcado do modelo “*germ-grain*”. O modelo “*germ-grain*” é uma generalização natural do modelo Booleano, onde o processo de Poisson de *germs* é substituído por um processo pontual qualquer.

### O modelo Booleano

Seja  $N = \{x_1, x_2, \dots\}$  um processo pontual de Poisson,  $\lambda$ -homogêneo, em  $\mathbb{R}^d$ . Seja  $\{S_1, S_2, \dots\}$  uma coleção de v.a.'s iid (e independente do processo  $N$ ) com valores em  $\mathcal{B}(\mathbb{R}^d)$ , isto é,  $S_i$  é um conjunto de Borel aleatório em  $\mathbb{R}^d$ . O modelo Booleano agora será construído usando “*germs*”  $x_n$  e “*grains*”  $S_n$

$$\mathcal{S} = \bigcup_{n=1}^{\infty} (x_n + S_n) \tag{2.2}$$

ou, de acordo com Hall (1988), como um processo de cobertura em  $\mathbb{R}^d$

$$\mathcal{S} = \{(x_i + S_i); i = 1, 2, \dots\} \tag{2.3}$$

onde  $x + S = \{(x + s) | s \in S\}$ . Observando que neste modelo, os pontos de  $\mathbb{R}^d$  podem ser classificados em duas categorias, cobertos e não cobertos (1 e 0), assim entenderemos o atributo “Booleano”. Uma propriedade diz que o número de conjuntos de  $\mathcal{S}$  os quais cobrem um ponto fixo  $x \in \mathbb{R}^d$  tem distribuição de Poisson com taxa  $\lambda E(\text{vol}(S))$  (Hall (1988), pag. 201). O modelo Booleano também é chamado “Poisson grain model”.

## 2.2 Processos espaciais de nascimento e morte (em $\mathbb{R}^d$ )

Um processo espacial de *nascimento e morte* é um processo Markoviano a tempo-contínuo. O espaço de estados  $\mathcal{N}$  é a família de configurações pontuais. Esses processos fornecem modelos de processos pontuais espaço-temporais (“spatio-temporal”), os quais envolvem tanto dispersão espacial quanto dispersão temporal dos pontos. Então, um processo espacial de nascimento e morte é um processo pontual no espaço que evolui em instantes do tempo tanto pelo *nascimento* de um ponto novo, o qual será adicionado à configuração, quanto pela *morte* de um ponto já existente, o qual será apagado da configuração. Além do mais, ele tem a propriedade *Markoviana* no tempo; a probabilidade da mudança depende só da configuração atual do processo. Os nascimentos são controlados pelo *taxa de nascimento*  $\mathbf{b}$ , uma função mensurável e não negativa:

$$\mathbf{b} : \mathbb{R}^d \times \mathcal{N} \rightarrow [0, \infty)$$

satisfazendo

$$\int_B \mathbf{b}(x, \eta) dx < \infty$$

para cada  $B$ , conjunto de Borel limitado, e para todos  $\eta \in \mathcal{N}$ . A probabilidade de um nascimento ocorrer, dentro do conjunto  $B$ , durante tempo  $[t, t + s)$ , dada a configuração  $\eta$  do processo no tempo  $t$ , seria

$$s \cdot \int_B \mathbf{b}(x, \eta) dx + o(s).$$

A ocorrência de morte é controlada pela *taxa de morte*  $\mathbf{d}$ , também uma função mensurável e não negativa

$$\mathbf{d} : \mathbb{R}^d \times \mathcal{N} \rightarrow [0, \infty).$$

A probabilidade de um ponto ser apagado da configuração  $\eta$  durante o intervalo de tempo  $[t, t + s)$ , dado que  $\eta$  seja a configuração do processo no tempo  $t$  e que  $x \in \eta$ , seria:

$$s \cdot \mathbf{d}(x, \eta) + o(s).$$

As funções  $\mathbf{b}$  e  $\mathbf{d}$  caracterizam o processo de nascimento e morte, cujo gerador infinitesimal é dado por:

$$Af(\eta) = \int \mathbf{b}(x, \eta)[f(\eta \cup \{x\}) - f(\eta)]dx + \int \mathbf{d}(x, \eta \setminus \{x\})[f(\eta \setminus \{x\}) - f(\eta)]\eta(dx). \quad (2.4)$$

para funções  $f$  “adequadas”.

Para nossos fins, vamos considerar um caso particular do processos de nascimento e morte cuja descrição segue. Seja  $\mathbf{G}$  uma família de objetos  $\gamma$  ( $\gamma \subseteq \mathbb{R}^d$ ), os quais chamaremos de *indivíduos*, e consideramos um espaço de estados  $\mathcal{N} = \{\eta \in \mathbb{N}^{\mathbf{G}} \mid \eta(\gamma) \neq 0 \text{ somente para um conjunto enumerável de } \gamma \in \mathbf{G}\}$ . O processo de nascimento e morte  $\eta_t$  é definido por um processo de Poisson marcado caracterizado por uma função mensurável e não negativa  $\mathbf{b}(\gamma, \eta)$  (mesmo sentido que acima); as marcas incluem tempo de vida - distribuído exponencialmente com taxa um (i.e. taxa de morte é constante, igual a um). Se as taxas de nascimento são independentes da configuração atual, i.e. existe uma função  $\omega : \mathbf{G} \rightarrow [0, \infty)$  tal que

$$\mathbf{b}(\gamma, \eta) = \omega(\gamma)$$

este processo é chamado de processo *livre*. Esse processo é apenas um processo de Poisson espaço-temporal marcado, o qual existe e é ergódico qualquer que seja  $\omega$ . No caso particular onde  $\omega(\gamma) = \lambda$ , a medida invariante é a de um processo de Poisson  $\lambda$ -homogêneo. A taxa de nascimento, desde que seja uniformemente limitada, pode ser decomposta da seguinte maneira:

$$\mathbf{b}(\gamma, \eta) = \omega(\gamma)M(\gamma, \eta) \quad (2.5)$$

onde,  $0 \leq M(\gamma, \eta) \leq 1$ . O primeiro fator representa a taxa de nascimento básica, devido a um relógio Poissoniano interno, e o segundo representa a chance de um indivíduo realmente nascer uma vez quando chegou a sua hora. Essa “chance”, a probabilidade não normalizada, seria aumentada ou diminuída de acordo com a configuração  $\eta$ . Introduzimos a função  $I : \mathbf{G} \times \mathbf{G} \rightarrow \{0, 1\}$

$$I(\gamma, \theta) = \mathbf{1}\{\sup_{\eta} \{|M(\gamma, \eta) - M(\gamma, \eta + \delta_{\theta})|\} > 0\}. \quad (2.6)$$

onde  $\delta_{\theta}(\gamma) = \mathbf{1}\{\gamma = \theta\}$ , isto é a configuração com somente um elemento,  $\theta$ . O supremum é sobre todas configurações tais que  $\eta$  e  $\eta + \delta_{\theta}$  pertençam ao conjunto de configurações permitidas ( $\{0, 1\}^{\mathbf{G}}$  ou  $\mathbb{N}^{\mathbf{G}}$ ). A função  $I(\gamma, \theta)$  indica os indivíduos que podem ter influência na taxa de nascimento do  $\gamma$ , ou seja se  $I(\gamma, \theta) = 1$ , presença (ou ausência) do  $\theta$  modifica a taxa de nascimento do  $\gamma$  e então dizemos que  $\theta$  é *incompatível* com  $\gamma$ .

## 2.3 Processo de ramificação multitypo

Processos de ramificação, em geral, modelam o desenvolvimento de uma população. Um conjunto inicial de indivíduos o qual vamos chamar de 0-ésima geração, tem filhos chamados de primeira geração, os filhos dos filhos são chamadas de segunda geração, e assim por diante. Cada um dos indivíduos se reproduz independentemente de todos os outros indivíduos, mas seguindo a mesma lei estocástica. Um processo de ramificação é definido pela lei estocástica chamada de lei da distribuição de prole, ou somente distribuição de prole (“offspring distribution”).

Quando todos os indivíduos na população são idênticos, com relação à característica de interesse, dizemos que eles são de um tipo só e o processo de ramificação associado é bem conhecido tanto no caso discreto quanto no caso contínuo (para um estudo detalhado veja Harris (1989)).

Se indivíduos diferem com relação à característica de interesse, ou seja, pertencem a vários tipos, tratamos com *processo de ramificação multitypo*. Neste trabalho nós precisamos de um processo de ramificação multitypo em tempo discreto chamado de processo de Galton-Watson multitypo.

### 2.3.1 Processo de Galton-Watson multitypo

Seja  $\chi = \{(x_1, \dots, x_m) \mid x_i \text{ inteiro não negativo}\}$  e seja  $e_i$ ,  $1 \leq i \leq m$ , o vetor cuja  $i$ -ésima coordenada é 1 e todas outras são 0.

O *processo de Galton-Watson multitypo* é uma cadeia de Markov homogênea  $\{\mathbf{Z}_n\}_{n \geq 0}$  com espaço de estados  $\chi$ . A  $i$ -ésima componente de  $\mathbf{Z}_n$ ,  $Z_n^i$  interpreta-se como o número de indivíduos de tipo  $i$  na  $n$ -ésima geração.

As leis da reprodução são dadas por  $p^i(x_1, \dots, x_m)$ ,  $i = 1, \dots, m$ , o que seria a probabilidade de um indivíduo do tipo  $i$  tenha  $x_1$  filhos de tipo 1, ...,  $x_m$  filhos do tipo  $m$ . Se  $\mathbf{Z}_0 = e_i$ , então a função geratriz de momentos de  $\mathbf{Z}_1$  é dada por

$$f^i(s_1, \dots, s_m) = \sum_{x_1, \dots, x_m=0}^{\infty} p^i(x_1, \dots, x_m) s_1^{x_1} \cdots s_m^{x_m}, \quad |s_1|, \dots, |s_m| \leq 1. \quad (2.7)$$

Usaremos também a notação  $\mathbf{s} = (s_1, \dots, s_m)$  e  $\mathbf{f}(\mathbf{s}) = (f^1(\mathbf{s}), \dots, f^m(\mathbf{s}))$ . Em geral, se  $\mathbf{Z}_n = (x_1, \dots, x_m) \in \chi$ , então  $\mathbf{Z}_{n+1}$  é a soma de  $x_1 + \dots + x_m$  vetores aleatórios independentes,  $x_1$  tendo a função geratriz  $f^1$ ,  $x_2$  tendo a função geratriz  $f^2$ , ...,  $x_m$  tendo a função geratriz  $f^m$ . Se  $\mathbf{Z}_n = \mathbf{0}$  então  $\mathbf{Z}_{n+1} = \mathbf{0}$ .

Definimos de  $\mathbf{M} = [m_{ij}]$  a matriz de primeiros momentos

$$m_{ij} = E(Z_1^j \mid \mathbf{Z}_0 = e_i) = \frac{\partial f^i(1, \dots, 1)}{\partial s_j} \quad i, j = 1, \dots, m. \quad (2.8)$$

Esta matriz tem um papel importante na teoria deste tipo do processo. Particularmente,  $\mathbf{M}$  serve para obter informações sobre a sobrevivência do processo. As informações são contidas em  $\mathbf{q} = (q^1, \dots, q^m)$ , onde  $q^i$  é a *probabilidade de extinção* da população, dado que o indivíduo inicial era do tipo  $i$

$$q^i = P(\mathbf{Z}_n = 0 \text{ para algum } n \mid \mathbf{Z}_0 = e_i) \quad i = 1, \dots, m. \quad (2.9)$$

Denotamos de  $\mathbf{1}$  o vetor  $(1, 1, \dots, 1)$ .

Se  $\mathbf{q} = \mathbf{1}$  o processo  $\mathbf{Z}_n$  é dito ser *sub-crítico* enquanto se  $q^i < 1$ , para todos  $i = 1, \dots, m$ , o processo  $\mathbf{Z}_n$  é dito ser *super-crítico*.

Antes de estabelecer o resultado principal relacionando  $\mathbf{M}$  e  $\mathbf{q}$  precisamos de alguns noções de algebra linear:

**Definição 4** *Uma matriz  $M_{m \times m}$  é dita ser positivamente regular se todos seus elementos são finitos e existe  $n \in \mathbb{N}$  tal que todos os elementos de  $M^n$  são estritamente positivos*

**Teorema 2.3.1** *Se  $\mathbf{M}$  é uma matriz positivamente regular então ela possui um autovalor positivo e simples  $\rho$ , e se  $\lambda$  for qualquer outro autovalor de  $M$  então  $|\lambda| < \rho$ .*

O autovalor  $\rho$  desta teorema chama-se de *autovalor de Peron-Frobenius*. Se  $M$  é positivamente regular então vamos dizer que o processo  $(\mathbf{Z}_n)$  é positivamente regular. Vamos dizer, também que o processo é *singular* se, com probabilidade um, cada indivíduo tem exatamente um filho. Este caso é considerado trivial e não vamos tratá-lo. O teorema seguinte dá um critério para verificação se o processo é sub ou super-crítico.

**Teorema 2.3.2** *Seja o processo  $\{\mathbf{Z}_i\}_{i \geq 0}$  positivamente regular e não singular e seja  $\rho$  o autovalor descrito no Teorema 2.3.1. (i) Se  $\rho \leq 1$ , então  $\mathbf{q} = \mathbf{1}$ . (ii) Se  $\rho > 1$ , então  $q^i < 1$ , para todos  $i = 1, \dots, m$  e (iii) qualquer seja o caso,  $\mathbf{q}$  é a menor raiz não-negativa da equação*

$$\mathbf{q} = \mathbf{f}(\mathbf{q}). \quad (2.10)$$

Observa-se que o Teorema 2.3.2 é aplicável somente no caso de número finito de tipos. Os casos de número infinito de tipos são menos estudados na literatura, especialmente quando o número de tipos não é contável. Para o caso contável, oferece-se uma condição

suficiente, um critério de sub-criticalidade do processo, verificando se o número médio de filhos, de um indivíduo do tipo  $i$ , de todos os tipos em todas as gerações seja finito

$$\sum_{n=1}^{\infty} \sum_{j=1}^{\infty} m_{ij}^{(n)} < \infty \quad (2.11)$$

onde  $m_{ij}^{(n)}$  representa o número médio de prole (de um indivíduo do tipo  $i$ ) do tipo  $j$  na  $n$ -ésima geração e é dado indutivamente por

$$m_{ij}^{(n)} = \sum_k m_{ik}^{(n-1)} m_{kj}. \quad (2.12)$$

Supondo agora que o número de tipos é não-contável, denotamos de  $\mathbf{G}$  o conjunto de indivíduos de todos os tipos. De acordo com Fernández, Ferrari e Garcia (2000), uma condição suficiente de sub-criticalidade do processo é dada por

$$\sum_{n=1}^{\infty} \int_{\mathbf{G}} m^{(n)}(G, dH) < \infty \quad (2.13)$$

para todos os indivíduos  $G \in \mathbf{G}$ , onde  $m^{(n)}(G, dH)$  é dado indutivamente por

$$m^{(n)}(G, dH) = \int_{\mathbf{G}} m^{(n-1)}(G, dK) m(K, dH). \quad (2.14)$$

# Capítulo 3

## Redes com perdas - caso contínuo, não limitado, unidimensional

### 3.1 Introdução

O modelo de redes com perdas unidimensional, contínuo e não limitado foi introduzido por Kelly (1991) e estudado por Ferrari e Garcia (1998). Clientes nesta rede estão arranjados ao longo de um cabo infinitamente longo e cada chamada entre dois pontos  $s_1, s_2 \in \mathbb{R}$  no cabo envolve somente o segmento compreendido entre eles. O cabo tem capacidade limitada, digamos  $C$  chamadas. Isto é, uma chamada entre  $s_1$  e  $s_2 \in \mathbb{R}$ ,  $s_1 < s_2$ , é perdida se ao longo de qualquer ponto entre  $s_1$  e  $s_2$ , o cabo já estiver carregando  $C$  chamadas. Chamadas chegam de acordo com um processo de Poisson homogêneo de taxa  $\lambda$  ao longo do cabo, e sua distância (espaço) segue uma distribuição  $\pi$  com média  $\rho$  e segundo momento  $\nu$ . A duração (tempo) de cada chamada tem distribuição exponencial de taxa 1. A localização da chamada, seu comprimento e sua duração são variáveis aleatórias independentes. Ferrari e Garcia (1998) usaram o argumento de percolação não orientada para provar que o processo de redes com perdas existe e além disso possui uma única medida invariante desde que terceiro momento de distribuição  $\pi$  seja finito e taxa de chegada  $\lambda$  suficientemente pequena. Nos refinamos este argumento de percolação, obtendo como condição para ergodicidade  $\nu < \infty$  e  $\lambda(\rho + \sqrt{\nu}) < 1$ . O argumento é baseado em uma representação gráfica da rede com perda e é a base para o esquema de simulação perfeita a ser utilizada neste trabalho.

## 3.2 Construção do processo

A primeira questão a responder sobre redes com perdas é a existência do processo em *volume infinito*. Fernández, Ferrari e Garcia (2000),(2001) obtiveram as condições para existência de processos espaciais de nascimento e morte, construindo graficamente o processo e determinando as condições para que esta construção seja possível. Aqui, descrevemos essa construção para um caso particular de processos em questão, processo de rede com perdas.

Rede com perdas é um processo espacial de nascimento e morte (2.2), os indivíduos agora são *chamadas* ( $\gamma = (x, x + u)$ ,  $x, u \in \mathbb{R}$ ) e nascimentos são regulados pelo princípio da exclusão, dependendo da capacidade da rede. O gerador do processo é dado por

$$Af(\eta) = \int (f(\eta + \delta_\gamma) - f(\eta))b(\gamma, \eta)d\gamma + \int (f(\eta - \delta_\gamma) - f(\eta))\eta(d\gamma) \quad (3.1)$$

onde  $\eta \in \{0, 1\}^{\mathcal{B}(\mathbb{R})}$ . A taxa de morte, igual a 1, é omitida do segundo fator. Na taxa do processo livre associado, entra o fator  $\pi(u)$  (seguimos a notação da Seção 2.2):

$$\omega((x, x + u)) = \lambda\pi(u) \quad (3.2)$$

A taxa de nascimento, segundo (2.5) será:

$$\mathbf{b}((x, x + u), \eta) = \lambda \pi(u) M((x, x + u), \eta). \quad (3.3)$$

onde, para capacidade  $C = 1$ , será

$$M(\gamma, \eta) = \prod_{\theta: \eta(\theta) \neq 0} (1 - I(\gamma, \theta)) \quad (3.4)$$

$$I(\gamma, \theta) = \begin{cases} 1 & \gamma \cap \theta \neq \emptyset \\ 0 & \text{caso contrário} \end{cases} \quad (3.5)$$

onde  $\gamma, \theta$  são da forma  $(x, x + u)$ . Para  $C > 1$ , a expressão é menos simples

$$M((x, x + u), \eta) = \begin{cases} 1 & \text{caso contrário} \\ 0 & \text{existe } y \in (x, x + u) \text{ e} \\ & \theta_1, \dots, \theta_C \text{ tais que } \eta(\theta_i) = 1 \\ & \text{e } y \in \theta_i \text{ para todos } i = 1, \dots, C. \end{cases}$$

Observamos que

$$\mathbf{b}((x, x + u), \eta) \leq \lambda, \text{ para todos } x, u, \eta \Rightarrow \sup_{x, u, \eta} \mathbf{b}((x, x + u), \eta) \leq \lambda. \quad (3.6)$$

### 3.2.1 Representação gráfica

Seja  $N = \{(\xi_1, T_1), (\xi_2, T_2), \dots\}$  um processo de Poisson homogêneo de taxa  $\lambda$  em  $\mathbb{R} \times [0, \infty)$ , e sejam  $S_1, S_2, \dots$  variáveis aleatórias independentes, exponencialmente distribuídas com média 1 e sejam  $U_1, U_2, \dots$  variáveis aleatórias independentes e identicamente distribuídas com distribuição comum  $\pi$ . As variáveis  $S$ 's,  $U$ 's e o processo de Poisson são independentes. Considere os retângulos aleatórios

$$R_i = \{(x, y); \xi_i \leq x \leq \xi_i + U_i, T_i \leq y \leq T_i + S_i\}.$$

então  $\{R_i, i \geq 1\} = \{(\xi_i, T_i) + D_i, i \geq 1\}$  é um modelo Booleano em  $\mathbb{R}^2$  onde  $D_i = [0, U_i] \times [0, S_i]$  e representa o processo independente de tentativas de chamadas.

Agora, para cada retângulo  $R_i$  associamos uma marca independente, uma variável aleatória  $Z_i$  com distribuição  $U(0, 1)$ , e cada retângulo marcado identificamos com o ponto marcado  $(\xi_i, T_i, S_i, U_i, Z_i)$ . Reconhecemos no processo pontual marcado  $\mathbf{R} = \{(\xi_i, T_i, S_i, U_i, Z_i), i = 1, 2, \dots\}$  uma representação gráfica do processo de nascimento e morte com taxa de nascimento constante  $\lambda$ , e taxa de morte constante, igual a 1 (o processo livre  $\alpha$ ) tal que  $Z_i$  servirá como indicadora dos nascimentos permitidos. Chamando de  $\mathbf{R} = (\xi, \tau, s, u, z)$ , usamos a notação:

$$\text{Basis}(\mathbf{R}) = (\xi, \xi + u), \quad \text{Birth}(\mathbf{R}) = \tau, \quad \text{Life}(\mathbf{R}) = [\tau, \tau + s], \quad \text{Flag}(\mathbf{R}) = z.$$

Também definimos, para dois retângulos  $\mathbf{R}$  e  $\mathbf{R}'$ ,

$$\begin{aligned} R' &\approx R, \text{ se } R' \cap R \neq \emptyset \\ R' &\sim R, \text{ caso contrário} \end{aligned}$$

### 3.2.2 Construção em volume finito

Vamos construir um processo de rede com perdas em uma janela finita  $\Lambda = [a, b]$  e portanto usaremos apenas um conjunto finito de retângulos marcados, começando de uma configuração inicial das chamadas  $\eta_0 = \{(\xi_{01}, \xi_{01} + U_{01}), (\xi_{02}, \xi_{02} + U_{02}), \dots\}$ . Esta configuração tem que ser uma realização legal do processo a qual proíbe mais de  $C$  chamadas cobrindo o mesmo ponto.

Seja  $\mathbf{R}^\Lambda = \{\mathbf{R} \in \mathbf{R} | \text{Basis}(\mathbf{R}) \cap \Lambda \neq \emptyset, \text{Birth}(\mathbf{R}) > 0\}$ . Para cada elemento  $(\xi_{0i}, \xi_{0i} + U_{0i})$  da configuração inicial  $\eta_0$  associamos uma variável aleatória independente, um tempo  $S_i \sim \text{Exp}(1)$  e retângulo  $(\xi_{0i}, 0, S_i, U_{0i}, 0)$ . Chamamos essa coleção de retângulos iniciais de  $\mathbf{R}_0^\Lambda$ . Seja

$$\mathbf{R}^\Lambda[0, t] = \{(\xi, \tau, s, u, z) \in \mathbf{R}^\Lambda \cup \mathbf{R}_0^\Lambda | 0 \leq \tau, \tau + s \leq t\} \quad (3.7)$$

Considere  $0 < t_1 < t_2 < \dots < t_N$  como as marcas de nascimento e morte  $T_i, S_i$  pertencendo a  $[0, t]$ . Pela propriedade do processo de Poisson todos esses pontos são distintos, ou seja

$$\{t_1, \dots, t_N\} = (0, t] \cap \{\tau, \tau + s \mid (\xi, \tau, s, u, z) \in \mathbf{R}^\Lambda[0, t]\}. \quad (3.8)$$

A construção do processo  $\eta_t^\Lambda$  segue:

FV.1. Para  $\eta_\varpi^\Lambda$  já definido e  $t_{i-1} \leq \varpi < t_i$  ( $t_0 \equiv 0$ ), por

$$\eta_\tau^\Lambda = \eta_\varpi^\Lambda, \text{ para todos } \varpi \leq \tau < t_i. \quad (3.9)$$

Se  $\varpi \geq t_N$  então

$$\eta_\tau^\Lambda = \eta_\varpi^\Lambda, \text{ para todos } \tau \geq \varpi. \quad (3.10)$$

FV.2. Se  $t_i$  é um tempo de morte, ie.  $t_i = \tau + s$  para algum  $(\xi, \tau, s, u, z) \in \mathbf{R}^\Lambda[0, t]$  então a chamada  $(\xi, \xi + u)$  será apagada

$$\eta_{t_i}^\Lambda = \eta_{t_i-}^\Lambda \setminus \{(\xi, \xi + u)\}. \quad (3.11)$$

Voltar a FV.1.

FV.3. Se  $t_i$  é um tempo de nascimento, i e.  $t_i = \tau$  para algum  $R = (\xi, \tau, s, u, z) \in \mathbf{R}^\Lambda[0, t]$ , então verifica-se a condição

$$z < M((\xi, \xi + u), \eta_{t_i-}). \quad (3.12)$$

Caso a resposta seja SIM, incluímos a chamada  $\eta_{t_i}^\Lambda = \eta_{t_i-}^\Lambda \cup \{(\xi, \xi + u)\}$ ; caso NÃO a configuração permanece a mesma  $\eta_{t_i}^\Lambda = \eta_{t_i-}^\Lambda$ .

Voltar a FV.1.

A condição em FV.3. pode ser interpretada da seguinte maneira, dependendo do capacidade da rede:

- $C = 1$ :  $R$  não intersecta nenhum retângulo cuja base pertence a configuração  $\eta_{t_i-}^\Lambda$ .
- $C > 1$ : Nenhum ponto dentro de  $R$  pertence a mais de  $C - 1$  retângulos cujas bases pertencem a configuração  $\eta_{t_i-}^\Lambda$ .

### 3.2.3 Construção em volume infinito

Para aplicar construção análoga em volume infinito enfrentamos a não-existência da primeira marca. A idéia é usar o fato de que o processo livre  $\alpha_t$  sempre existe e achar as condições para aplicar construção em volume finito. Seguindo Fernández et al. (2001) introduzimos uma série de definições. Para um ponto arbitrário  $(x, t) \in \mathbb{R}^2$  definimos o conjunto de todos retângulos aos quais o ponto pertence:

$$A_1^{(x,t)} = \{R \in \mathbf{R} \mid x \in \text{Basis}(R), t \in \text{Life}(R)\} \quad (3.13)$$

Para cada retângulo  $R$  definimos o conjunto de ancestrais:

$$A_1^R = \{R' \in \mathbf{R} \mid \text{Birth}(R') \leq \text{Birth}(R), R' \approx R\} \quad (3.14)$$

Agora, definimos recursivamente as gerações ( $n > 1$ ) dos conjuntos acima ou seja n-ésimas gerações de ancestrais:

$$A_n^{(x,t)} = \{R'' \mid R'' \in A_1^{R'} \text{ para algum } R' \in A_{n-1}^{(x,t)}\} \quad (3.15)$$

$$A_n^R = \{R'' \mid R'' \in A_1^{R'} \text{ para algum } R' \in A_{n-1}^R\} \quad (3.16)$$

Vamos dizer que há percolação orientada para trás (“backward oriented percolation”) se existe um ponto  $(x, t)$  tal que  $A_n^{(x,t)} \neq \emptyset$  para qualquer  $n$ , ou seja, se existe um ponto com um número infinito de ancestrais. Chamamos de clã do ponto  $(x, t)$  a união de seus ancestrais:

$$A^{(x,t)} = \bigcup_{n \geq 1} A_n^{(x,t)} \quad (3.17)$$

e  $\mathbf{R}[0, t] = \{R \in \mathbf{R} \mid \text{Birth}(R) \in [0, t]\}$ .

Para estimar o “tamanho” do clã  $A^{(x,t)}$  introduzimos duas variáveis aleatórias representando o comprimento temporal (“time-length”) e a largura espacial (“space-width”) do clã.

$$TL(A^{(x,t)}) = t - \sup\{s \mid s \in \text{Life}(R) \text{ para algum } R \in A^{(x,t)}\} \quad (3.18)$$

$$SW(A^{(x,t)}) = m_1(\cup_{R \in A^{(x,t)}} \text{Basis}(R)) \quad (3.19)$$

A existência do processo em volume infinito, em qualquer intervalo finito de tempo é garantida contanto que o processo não exploda, ou seja, nenhum retângulo desenvolva infinitamente muito ancestrais em tempo finito.

**Teorema 3.2.1** *Se  $A^{(x,t)} \cap \mathbf{R}[0, t]$  é finito com probabilidade um, para qualquer  $x \in \mathbb{R}$  e  $t \geq 0$ , então para qualquer  $\Lambda \subseteq \mathbb{R}$  o processo de rede com perdas em  $\Lambda$ ,  $\eta_t^\Lambda$  é bem definido e possui pelo menos uma medida invariante  $\mu^\Lambda$*

Para existência do processo em tempo infinito, precisa-se que todos clãs de ancestrais sejam finitos com probabilidade um, ou seja, a falta de percolação orientada para trás. A fim de achar medidas invariantes, para processos de Markov é usual construir-se o processo desde  $-\infty$  com uma configuração dada e parar o processo no tempo 0. Se o processo não depende da configuração inicial então no tempo 0 terá uma medida invariante. Na construção gráfica acima a falta da percolação nos permite construir o processo  $\eta_t$  “emagrecendo” o processo livre  $\alpha_t$  para todo  $t \in \mathbb{R}$ . Além do mais, o mesmo fato mostra que a distribuição de  $\eta_t$  não depende do passado, ou seja é a distribuição invariante. O próximo teorema sumariza os resultados mais importantes sobre o processo em volume e tempo infinito segundo Fernández et al. (2000), (2001) e Garcia (2000).

**Teorema 3.2.2** *Se com probabilidade um não há percolação orientada para trás em  $\mathbf{R}$ , então o processo de rede com perdas  $\eta_t$  existe em  $(-\infty, \infty)$ , para qualquer  $t \in \mathbb{R}$ , e possui uma única medida invariante para a qual converge exponencialmente rápido.*

Uma maneira de achar condições garantindo a falta da percolação é a dominação pelo um processo de ramificação. Estabelecendo as condições de sub-criticalidade do processo de ramificação obteremos condições suficientes para falta de percolação no processo de ancestrais. Olhando para trás, os ancestrais serão consideradas como os ramos. O tempo real de morte apresentará o tempo de nascimento neste ponto da vista. Vamos primeiramente observar que o processo de ancestrais não é um processo de Galton-Watson por causa de possibilidade de alguns retângulos pertencerem à várias gerações e assim não temos independência dos ramos.

Portanto no conjunto de retângulos introduzimos um processo de Galton-Watson,  $B_n$  tal que a distribuição de prole de um retângulo  $R$  segue a mesma lei (marginal) que a distribuição de  $A_1^R$ , mas os ramos se comportam independentemente. O processo  $B_n$  induz naturalmente no conjunto de chamadas um outro processo de ramificação multitypo o que vamos ver na seção ao seguir.

### 3.3 Um processo de ramificação dominando clã de ancestrais. Valor crítico.

Seja  $R$  um retângulo com base  $\gamma = (x, x + u)$  de comprimento  $u$ , nascido em tempo 0. Definimos  $b_n^u(v)$  como o número de retângulos na  $n$ -ésima geração de ancestrais de  $R$  tendo a base de comprimento  $v$ :

$$b_n^u(v) = |\{R' \in B_n^R \mid |Basis(R')| = v\}|. \quad (3.20)$$

O processo  $b_n$  é um processo de Galton-Watson multitypo, descrito na Seção 2.3.1, onde cada comprimento de chamada representa um tipo. Número de tipos depende de distribuição  $\pi$ , e pode ser:

1. finito, caso  $\pi$  seja simples
2. infinito enumerável, caso  $\pi$  seja discreta
3. infinito não enumerável, caso  $\pi$  seja contínua

Para estabelecer as condições de sub-criticalidade deste processo precisamos das seguintes quantidades:  $m(u, v)$  - o número médio de filhos do tipo  $v$ , quando o tipo do pai é  $u$ . Vamos mostrar que as distribuições de prole (“offspring distributions”) de  $b_n$  são Poisson com médias

$$m(u, v) = \lambda \pi(v) (u + v) \quad (3.21)$$

Observação : na análise a seguir usamos noções “pai” e “ancestral” em sentido original.

Seja  $\gamma$  o indivíduo inicial do tipo  $u$ . Sem perda de generalização suponhamos que  $\gamma = (0, u)$  e consideramos o retângulo  $R$  nascido em tempo 0 tal que  $Basis(R) = \gamma$ . É fácil ver que um retângulo  $(x, x + v) \times (y, y + s)$  pode ser um pai de  $R$  somente se  $x \in (-v, u)$  e será um pai mesmo se  $y + s > 0$ .

Seja  $\beta_{uv}(t)$  o número de pais de  $R$  do tipo  $v$  nascidos depois do tempo  $-t$ . Então

$$b_1^u(v) = \lim_{t \rightarrow \infty} \beta_{uv}(t) \quad q.c. \quad (3.22)$$

Vamos chamar de  $\Delta$  a área  $[-v, u] \times [-t, 0]$ , e de  $N(\Delta)$  o processo de Poisson homogêneo com taxa  $\lambda$  no  $\Delta$ . Então, para  $k = 0, 1, \dots$

$$P(\beta_{uv}(t) = k) = \sum_{n \geq k} P(N(\Delta) = n \text{ e entre } n \text{ rectâng. } k \text{ são pais de } R \text{ do tipo } v) \quad (3.23)$$

Sejam  $(x_1, y_1), \dots, (x_n, y_n)$  uma realização de  $N(\Delta)$ . A cada ponto são associadas duas marcas independentes  $w$ , comprimento de chamada, com distribuição  $\pi$ , e  $s$  comprimento da vida, com distribuição  $\text{Exp}(1)$ . Dado  $N(\Delta) = n$ , os pontos  $(x_i, y_i)$  são distribuídos uniformemente em  $\Delta$  i.e.  $x_i \sim U(-v, u)$  e  $y_i \sim U(-t, 0)$ . Consideramos, então, os retângulos  $R_i = [x_i, x_i + w_i] \times [y_i, y_i + s_i]$ . Assim,

$$P(R_i \text{ é um pai de } R \text{ do tipo } v) = \pi(v) P(y_i + s_i > 0). \quad (3.24)$$

e vale

$$P(y_i + s_i > 0) = \int_{-t}^0 P(s_i > -y) \frac{1}{t} dy = \frac{1 - e^{-t}}{t}. \quad (3.25)$$

Para facilitar os cálculos que seguem usamos a seguinte notação :  
 $\alpha_t = \lambda (u + v) t$ ,  $p_t = \pi(v) (1 - e^{-t})/t$ .

De (3.23),(3.24) e (3.25) temos

$$P(\beta_{uv}(t) = k) = \sum_{n \geq k} \binom{n}{k} (p_t)^k (1 - p_t)^{n-k} e^{-\alpha_t} \frac{(\alpha_t)^n}{n!} = e^{-p_t \alpha_t} \frac{(p_t \alpha_t)^k}{k!}. \quad (3.26)$$

Observamos que

$$\lim_{t \rightarrow \infty} p_t \alpha_t = \lim_{t \rightarrow \infty} \lambda \pi(v) (u + v) (1 - e^{-t}) = \lambda \pi(v) (u + v). \quad (3.27)$$

Agora, de (3.22) segue que  $\beta_{uv}(t)$  converge a  $b_1^u(v)$  em distribuição

$$P(b_1^u(v) = k) = \lim_{t \rightarrow \infty} P(\beta_{uv}(t) = k), \quad k = 0, 1, \dots \quad (3.28)$$

Daqui se conclui que  $b_1^u(v)$  tem distribuição de Poisson com média  $\lambda \pi(v) (u + v)$  o que é exatamente (3.21).

Nos interessa quando o processo  $b_n$  é sub-crítico e portanto, pela Seção 2.3.1, consideramos a média do número total de filhos de todos os tipos em todas as gerações, de um indivíduo inicial do qualquer tipo, digamos  $u$ . A pergunta é então quando converge a série

$$\sum_{n \geq 1} \sum_v m^{(n)}(u, v) \quad (3.29)$$

onde  $m^{(n)}(u, v)$  é o número médio da prole do  $u$  do tipo  $v$  na  $n$ -ésima geração e é dado inductivamente

$$m^{(n)}(u, v) = \sum_w m^{(n-1)}(u, w) m(w, v). \quad (3.30)$$

$$\sum_v m^{(n)}(u, v) = \sum_v \sum_{v_1} \dots \sum_{v_{n-1}} \lambda^n \pi(v_1) (u + v_1) \pi(v_2) (v_1 + v_2) \dots \pi(v) (v_{n-1} + v) \quad (3.31)$$

A fim de facilitar a leitura, neste ponto enfatizamos que

- $\rho$  é o primeiro momento da distribuição  $\pi$  ( $\rho = Eu$ )
- $\nu$  é o segundo momento da distribuição  $\pi$  ( $\nu = Eu^2$ ).

Observamos que

$$\sum_v \pi(v)(v_{n-1} + v) = v_{n-1} \sum_v \pi(v) + \sum_v \pi(v)v = v_{n-1} + \rho = f_1 + v_{n-1}g_1 \quad (3.32)$$

onde  $f_1 = \rho$ ,  $g_1 = 1$ . Também

$$\begin{aligned} & \sum_{v_{n-1}} \pi(v_{n-1})(v_{n-2} + v_{n-1})(f_1 + v_{n-1}g_1) = \sum_{v_{n-1}} \pi(v_{n-1})(v_{n-2} + v_{n-1})(v_{n-1} + \rho) \\ &= \sum_{v_{n-1}} v_{n-1}^2 \pi(v_{n-1}) + v_{n-1} \pi(v_{n-1})(v_{n-2} + \rho) + \pi(v_{n-1})(v_{n-2}\rho) \\ &= \nu + \rho(v_{n-2} + \rho) + v_{n-2}\rho = \nu + \rho^2 + v_{n-2}2\rho = f_2 + v_{n-2}g_2 \end{aligned} \quad (3.33)$$

onde  $f_2 = \nu + \rho^2$ ,  $g_2 = 2\rho$ .

Vamos estabelecer a relação entre  $f_j, g_j e f_{j+1}, g_{j+1}$

$$\sum_{v_{n-j}} \pi(v_{n-j})(v_{n-j-1} + v_{n-j})(f_j + v_{n-j}g_j) \quad (3.34)$$

$$\begin{aligned} &= \sum_{v_{n-j}} g_j v_{n-j}^2 \pi(v_{n-j}) + v_{n-j} \pi(v_{n-j})(f_j + v_{n-j-1}g_j) + \pi(v_{n-j})(v_{n-j-1}f_j) \\ &= g_j \nu + f_j \rho + v_{n-j-1}(g_j \rho + f_j) = f_{j+1} + v_{n-j-1}g_{j+1} \end{aligned} \quad (3.35)$$

Então

$$f_{j+1} = \rho f_j + \nu g_j, \quad g_{j+1} = f_j + \rho g_j \quad (3.36)$$

ou, em forma matricial

$$\begin{bmatrix} f_{j+1} \\ g_{j+1} \end{bmatrix} = \begin{bmatrix} \rho & \nu \\ 1 & \rho \end{bmatrix} \cdot \begin{bmatrix} f_j \\ g_j \end{bmatrix} = \begin{bmatrix} \rho & \nu \\ 1 & \rho \end{bmatrix}^j \cdot \begin{bmatrix} \rho \\ 1 \end{bmatrix} \quad (3.37)$$

Agora, de (3.31) temos que

$$\sum_v m^{(n)}(u, v) = \lambda^n (f_n + u g_n). \quad (3.38)$$

A próxima tarefa é achar explicitamente valores de  $f_n$  e  $g_n$

### Cálculo de $f_n$ e $g_n$

Precisamos de  $T^n$ , onde  $T = \begin{bmatrix} \rho & \nu \\ 1 & \rho \end{bmatrix}$ . Vamos achar os autovalores ( $\epsilon_1, \epsilon_2$ ) e os autovetores correspondentes ( $\mathbf{x}_1, \mathbf{x}_2$ ) a fim de decompor  $T$ .

$$\det(T - \epsilon I) = \epsilon^2 - 2\rho\epsilon + \rho^2 - \nu \quad (3.39)$$

então os autovalores são

$$\epsilon_1 = \rho + \sqrt{\nu}, \quad \epsilon_2 = \rho - \sqrt{\nu}. \quad (3.40)$$

Os autovetores à direita achamos resolvendo a equação  $T\mathbf{x} = \epsilon\mathbf{x}$ . Dois autovetores (colunas) normalizados são:

$$\mathbf{x}_1 = \frac{1}{\sqrt{\nu+1}} \begin{bmatrix} \sqrt{\nu} \\ 1 \end{bmatrix}, \quad \mathbf{x}_2 = \frac{1}{\sqrt{\nu+1}} \begin{bmatrix} \sqrt{\nu} \\ -1 \end{bmatrix}. \quad (3.41)$$

Seja  $Q = [\mathbf{x}_1 \ \mathbf{x}_2]$  e  $D$  uma matriz diagonal com elementos  $\epsilon_1$  e  $\epsilon_2$ . Então

$$T^n = Q D^n Q^{-1}. \quad (3.42)$$

Explicitamente

$$Q = \frac{1}{\sqrt{\nu+1}} \begin{bmatrix} \sqrt{\nu} & \sqrt{\nu} \\ 1 & -1 \end{bmatrix}, \quad D = \begin{bmatrix} \epsilon_1 & 0 \\ 0 & \epsilon_2 \end{bmatrix}, \quad Q^{-1} = \frac{\sqrt{\nu+1}}{2\sqrt{\nu}} \begin{bmatrix} 1 & \sqrt{\nu} \\ 1 & -\sqrt{\nu} \end{bmatrix}. \quad (3.43)$$

Então

$$\begin{aligned} T^n &= \frac{1}{2\sqrt{\nu}} \begin{bmatrix} \sqrt{\nu} & \sqrt{\nu} \\ 1 & -1 \end{bmatrix} \cdot \begin{bmatrix} \epsilon_1^n & 0 \\ 0 & \epsilon_2^n \end{bmatrix} \cdot \begin{bmatrix} 1 & \sqrt{\nu} \\ 1 & -\sqrt{\nu} \end{bmatrix} \\ T^n &= \frac{1}{2\sqrt{\nu}} \begin{bmatrix} \sqrt{\nu}(\epsilon_1^n + \epsilon_2^n) & \nu(\epsilon_1^n - \epsilon_2^n) \\ \epsilon_1^n - \epsilon_2^n & \sqrt{\nu}(\epsilon_1^n + \epsilon_2^n) \end{bmatrix}. \end{aligned} \quad (3.44)$$

Agora

$$\begin{aligned} \begin{bmatrix} f_n \\ g_n \end{bmatrix} &= T^{n-1} \cdot \begin{bmatrix} \rho \\ 1 \end{bmatrix} = \frac{1}{2\sqrt{\nu}} \begin{bmatrix} \rho\sqrt{\nu}(\epsilon_1^{n-1} + \epsilon_2^{n-1}) + \nu(\epsilon_1^{n-1} - \epsilon_2^{n-1}) \\ \rho(\epsilon_1^{n-1} - \epsilon_2^{n-1}) + \sqrt{\nu}(\epsilon_1^{n-1} + \epsilon_2^{n-1}) \end{bmatrix} \\ &= \frac{1}{2\sqrt{\nu}} \begin{bmatrix} \sqrt{\nu}\epsilon_1^{n-1}(\rho + \sqrt{\nu}) + \sqrt{\nu}\epsilon_2^{n-1}(\rho - \sqrt{\nu}) \\ \epsilon_1^{n-1}(\rho + \sqrt{\nu}) - \epsilon_2^{n-1}(\rho - \sqrt{\nu}) \end{bmatrix}. \end{aligned} \quad (3.45)$$

Finalmente

$$f_n = \frac{1}{2}(\epsilon_1^n + \epsilon_2^n), \quad g_n = \frac{1}{2\sqrt{\nu}}(\epsilon_1^n - \epsilon_2^n). \quad (3.46)$$

Voltando a (3.29) verificamos a convergência da serie

$$\sum_n \sum_v m^{(n)}(u, v) = \sum_n \lambda^n (f_n + u g_n). \quad (3.47)$$

O radio de convergência desta serie é ( pela fórmula de Cauchy-Hadamard)

$$R = \frac{1}{\overline{\lim}_{n \rightarrow \infty} (f_n + u g_n)^{1/n}}. \quad (3.48)$$

Vamos achar o R:

$$f_n + u g_n = \frac{1}{2} \left[ \left(1 + \frac{u}{\sqrt{\nu}}\right) \epsilon_1^n + \left(1 - \frac{u}{\sqrt{\nu}}\right) \epsilon_2^n \right] \quad (3.49)$$

$$= \epsilon_1^n \frac{1}{2} \left[ \left(1 + \frac{u}{\sqrt{\nu}}\right) + \left(1 - \frac{u}{\sqrt{\nu}}\right) \left(\frac{\epsilon_2}{\epsilon_1}\right)^n \right]. \quad (3.50)$$

Sabemos que  $\epsilon_1 = Eu + \sqrt{Eu^2}$  é positivo pois  $P(u \geq 0) = 1$  e se fosse  $Eu = 0$  então  $P(u = 0) = 1$  o que podemos excluir da consideração . Já  $\epsilon_2$  não é positivo, pois  $Eu \leq \sqrt{Eu^2}$ .

Além disso  $\epsilon_2 + \epsilon_1 = 2Eu > 0$  e então  $\frac{\epsilon_2}{\epsilon_1} \in [-1, 0]$ .

Logo,

$$\frac{1}{2} 2 \min\left(1, \frac{u}{\sqrt{\nu}}\right) \leq \frac{1}{2} \left[ \left(1 + \frac{u}{\sqrt{\nu}}\right) + \left(1 - \frac{u}{\sqrt{\nu}}\right) \left(\frac{\epsilon_2}{\epsilon_1}\right)^n \right] \leq \frac{1}{2} 2 \max\left(1, \frac{u}{\sqrt{\nu}}\right). \quad (3.51)$$

Disto e de (3.50) segue que

$$\epsilon_1 \left(\min\left(1, \frac{u}{\sqrt{\nu}}\right)\right)^{1/n} \leq (f_n + u g_n)^{1/n} \leq \epsilon_1 \left(\max\left(1, \frac{u}{\sqrt{\nu}}\right)\right)^{1/n} \quad (3.52)$$

onde ambos os lados limitando  $(f_n + u g_n)^{1/n}$  vão para  $\epsilon_1$ , quando  $n \rightarrow \infty$ . Então

$$\lim_{n \rightarrow \infty} (f_n + u g_n)^{1/n} = \epsilon_1 \quad (3.53)$$

de onde segue

$$R = \frac{1}{\epsilon_1} = \frac{1}{\rho + \sqrt{\nu}}. \quad (3.54)$$

Portanto, e por  $\lambda > 0$ , concluimos que

1. Se

$$\lambda < \frac{1}{\rho + \sqrt{\nu}} \quad (3.55)$$

então a série (3.47) converge absolutamente e consequentemente, o processo  $b_n$  é sub-crítico.

2. se  $\lambda > \frac{1}{\rho + \sqrt{\nu}}$  então a série (3.47) diverge e o processo  $b_n$  pode ser super-crítico.

Chamamos de *lambda crítico* para o processo  $b_n$ , a quantidade

$$\lambda_c^* = \frac{1}{\rho + \sqrt{\nu}}. \quad (3.56)$$

No caso geral suponha que o número de tipos não é contável. Seja agora  $V$  o conjunto de todos os tipos e, pela Seção 2.3.1, observamos a série

$$\sum_{n \geq 1} \int_V m^{(n)}(u, dv) \quad (3.57)$$

onde

$$m^{(n)}(u, dv) = \int_V m^{(n-1)}(u, dw) m(w, dv) \quad (3.58)$$

é dado indutivamente. Então

$$\int_V m^{(n)}(u, dv) = \int_V \int_V \dots \int_V m(u, dv_1) m(v_1, dv_2) \dots m(v_{n-1}, dv). \quad (3.59)$$

Supondo que  $\pi$  é uma densidade i.e. que a distribuição de comprimentos de chamadas é absolutamente contínua, teremos

$$m(u, dv) = \lambda(u + x)\pi(x)dx. \quad (3.60)$$

Então

$$\int_V m(v_{n-1}, dv) = \int_0^\infty \lambda(v_{n-1} + x)\pi(x)dx = \lambda(v_{n-1} + \rho) = \lambda(f_1 + v_{n-1}g_1) \quad (3.61)$$

e logo

$$\int_V m(v_{n-2}, dv_{n-1}) \cdot \lambda(f_1 + v_{n-1}g_1) \quad (3.62)$$

$$\begin{aligned} &= \int_0^\infty \lambda(v_{n-2} + x)\pi(x) \cdot \lambda(x + \rho)dx \\ &= \lambda^2 \int_0^\infty x^2\pi(x) + x(v_{n-2} + \rho)\pi(x) + v_{n-2}\rho\pi(x)dx \\ &= \lambda^2(\nu + \rho(v_{n-2} + \rho) + v_{n-2}\rho) = \lambda^2(\nu + \rho^2 + v_{n-2}2\rho) = \lambda^2(f_2 + v_{n-2}g_2) \end{aligned} \quad (3.63)$$

onde  $f_1, g_1, f_2, g_2, \dots$  são exatamente os mesmos (3.37). Então, o cálculo é análogo a do caso discreto e teremos

$$\int_V m^{(n)}(u, dv) = \lambda^n (f_n + u g_n) \quad (3.64)$$

e de novo a convergência da série (3.47) significa que o processo é sub-crítico.

*Observação :* Se  $\pi$  é a distribuição  $U(0, 1)$ ,  $\lambda_c^* \approx 0.9282$ .

Fernández et al. (2000) ofereceram uma condição suficiente para sub-criticalidade desse processo de ramificação tratando uma classe de processos dentre os quais o processo de redes com perdas. A condição deles é

$$\alpha = \sup_G \frac{1}{\mu(G)} \int_{\mathbb{R}} \lambda dx \int_{G_x} \pi_x(dH) \mu(H) I(H, G) < 1 \quad (3.65)$$

onde  $G_x$  é o conjunto de indivíduos, no nosso caso chamadas, possivelmente posicionados em  $x$ , e  $I$  definido em (3.5). Escolhendo  $\mu(v) = c$ , onde  $c$  é alguma constante  $\geq 1$  obtém-se

$$\alpha = \sup_G \lambda \int_{\mathbb{R}} dx \int_{G_x} I(G, H) \pi(dH). \quad (3.66)$$

Sem perda de generalização consideremos uma chamada  $G = (0, L)$ , posicionada no origem  $(0,0)$ . Seus ancestrais seriam os retângulos, com vidas suficientemente longas, cujas bases intersectam a chamada. Isto inclui chamadas começando em qualquer ponto ocupado pela chamada inicial e também as chamadas começando na parte negativo da reta mas longas suficiente para intersectar a chamada inicial. Então,

$$\alpha = \lambda \sup_L \left( \int_{-\infty}^0 P(|H| > -x) dx + \int_0^L dx \right) \quad (3.67)$$

$$= \lambda \sup_L (\rho + L) = \lambda (\rho + \sup L). \quad (3.68)$$

Para os casos nos quais o comprimento de chamada é limitado quase certamente e  $M = \inf\{y \geq 0 | P(|H| \leq y) = 1\}$ , a condição (3.65) será

$$\lambda (\rho + M) < 1 \Leftrightarrow \lambda < \frac{1}{\rho + M}. \quad (3.69)$$

Esta condição coincide com (3.55) somente quando o comprimento de chamada é fixo, para todos os outros casos é mais fraca do que (3.55) pois  $\sqrt{E|H|^2} \leq M$ . No caso particular,  $U(0, 1)$ , esta condição garante sub-criticalidade do processo de ramificação para  $\lambda < \frac{2}{3} \approx 0.6667$  enquanto o nosso resultado é  $\lambda < 0.9282$ .

### 3.4 Um caso particular: $\pi$ é uniforme discreta

Consideremos um processo de redes com perdas onde chamadas são distribuídas de acordo uma distribuição uniforme discreta. Chamadas podem ser de comprimento  $\frac{1}{m}, \frac{2}{m}, \dots, \frac{m}{m}$  com probabilidades iguais i.e.  $\pi_m(\frac{i}{m}) = P(u = \frac{i}{m}) = \frac{1}{m}$ , para todos  $i = 1, \dots, m$ .

Analogamente à Seção 3.3 considera-se um processo de ramificação multitypo  $b_n^m$  o qual domina o clã de ancestrais. Neste caso o número de tipos é finito (igual á  $m$ ) o que possibilita uma aplicação da Teorema 2.3.2. As distribuições de prole são de Poisson com médias

$$m(i, j) = \lambda \frac{1}{m} \left( \frac{i}{m} + \frac{j}{m} \right) = \frac{\lambda}{m^2} (i + j), \quad i, j = 1, \dots, m. \quad (3.70)$$

Então  $\mathbf{M} = [m(i, j)]_{m \times m} = \frac{\lambda}{m^2} [i + j]_{m \times m} = \frac{\lambda}{m^2} A$ , onde  $A = [i + j]_{m \times m}$ . A fim de achar a condição de sub-criticalidade do processo precisamos do seguinte lema

**Lema 3.4.1** *Sejam  $c_i, i = 1, \dots, m$ , alguns constantes não negativas e seja  $A_{m \times m} = [a(i, j)]$ , onde  $a(i, j) = c_i + c_j$ . Então*

a) *O polinômio característico de  $A$  é dado por*

$$|A - \rho I| = (-1)^m \rho^{m-2} (\rho^2 - \text{tr} A \cdot \rho + \text{coef}(A)) \quad (3.71)$$

onde

$$\text{tr} A = 2 \sum_{i=1}^m c_i, \quad \text{coef}(A) = \left( \sum_{i=1}^m c_i \right)^2 - m \sum_{i=1}^m c_i^2. \quad (3.72)$$

Portanto,  $A$  possui somente dois autovalores diferentes de 0 e eles são dados como raízes da equação quadrática de (3.71)

$$\rho_{1,2} = \sum_{i=1}^m c_i \pm \sqrt{m \sum_{i=1}^m c_i^2}. \quad (3.73)$$

b) *A raiz de Peron-Frobenius é então*

$$\varrho = \sum_{i=1}^m c_i + \sqrt{m \sum_{i=1}^m c_i^2} \quad (3.74)$$

**Prova:** Apêndice A.

Observamos que no nosso caso consideramos os coeficientes  $c_i = i$ .

$$|\mathbf{M} - \rho I| = 0 \Leftrightarrow |A - \rho \frac{m^2}{\lambda} I| = 0 \quad (3.75)$$

E a raiz de Peron-Frobenius de A é, pelo Lema 3.4.1 igual a

$$\sum_{i=1}^m i + \sqrt{m \sum_{i=1}^m i^2} = \frac{m(m+1)}{2} + \sqrt{\frac{m^2(m+1)(2m+1)}{6}}. \quad (3.76)$$

De (3.75) obteremos o autovalor de Peron-Frobenius de  $\mathbf{M}$

$$\varrho = \frac{\lambda}{m} \left( \frac{(m+1)}{2} + \sqrt{\frac{(m+1)(2m+1)}{6}} \right). \quad (3.77)$$

Agora temos a condição desejada sob a qual o processo  $b_n^m$  é subcrítico:

$$\varrho < 1 \Leftrightarrow \lambda < \frac{m}{\frac{(m+1)}{2} + \sqrt{\frac{(m+1)(2m+1)}{6}}}. \quad (3.78)$$

O valor crítico para o processo com  $m$  tipos é dado por

$$\lambda_c^*(m) = \frac{m}{\frac{(m+1)}{2} + \sqrt{\frac{(m+1)(2m+1)}{6}}} \quad (3.79)$$

e é uma função crescente em  $m$ . Na verdade, seja

$$f(x) = \frac{x}{\frac{(x+1)}{2} + \sqrt{\frac{(x+1)(2x+1)}{6}}}, \quad \text{onde } x > 0 \quad (3.80)$$

então

$$\frac{df}{dx} = \frac{\frac{x+1}{2} + \sqrt{\frac{(x+1)(2x+1)}{6}} - \frac{x}{2} \left( 1 + \frac{4x+3}{6\sqrt{\frac{(x+1)(2x+1)}{6}}} \right)}{\left( \frac{(x+1)}{2} + \sqrt{\frac{(x+1)(2x+1)}{6}} \right)^2} \quad (3.81)$$

$$\frac{df}{dx} > 0 \Leftrightarrow \frac{1}{2} + \sqrt{\frac{2x^2+3x+1}{6}} - \frac{x(4x+3)}{12\sqrt{\frac{2x^2+3x+1}{6}}} > 0 \Leftrightarrow \frac{1}{2} + \frac{x+1}{4\sqrt{\frac{2x^2+3x+1}{6}}} > 0 \quad (3.82)$$

o que é satisfeito para todos  $x > 0$  de onde segue que  $f(x)$  é crescente para todos  $x > 0$ .

Então, quanto mais tipos temos tanto maior é o  $\lambda_c^*$ . Refinando a divisão do intervalo  $[0, 1]$  daremos o peso menor para tipos maiores e assim se diminui a probabilidade de processo sobreviver para sempre. Até intuitivamente é fácil ver que uma chamada de comprimento 1 tem muito mais possibilidades de ter filhos do que, por exemplo, uma chamada de comprimento  $1/100$ . No caso, por exemplo, de 4 tipos, uma chamada de comprimento 1 aparece com probabilidade  $\frac{1}{4}$  enquanto no caso, por exemplo, de 100 tipos esta probabilidade é muito menor, igual a  $\frac{1}{100}$ .

Vamos ver o que acontece quando  $m \rightarrow \infty$ , caso da divisão mais refinada. Dividindo (3.79) com  $\frac{m}{m}$  obteremos

$$\lim_{m \rightarrow \infty} \lambda_c^*(m) = \lim_{m \rightarrow \infty} \frac{1}{\frac{1}{2}(1 + \frac{1}{m}) + \sqrt{\frac{1}{3} + \frac{1}{2m} + \frac{1}{6m^2}}} = \frac{1}{\frac{1}{2} + \sqrt{\frac{1}{3}}} \approx 0.9282 \quad (3.83)$$

Sejam  $\rho_m$  e  $\nu_m$  respectivamente o primeiro e o segundo momento da distribuição  $\pi_m$ . Então

$$\rho_m = \sum_{i=1}^m \frac{i}{m} \frac{1}{m} = \frac{m+1}{2m} \quad (3.84)$$

$$\nu_m = \sum_{i=1}^m \frac{i^2}{m^2} \frac{1}{m} = \frac{(m+1)(2m+1)}{6m^2}. \quad (3.85)$$

Comparando equações acima com (3.79) vê-se

$$\lambda_c^*(m) = \frac{1}{\frac{(m+1)}{2m} + \sqrt{\frac{(m+1)(2m+1)}{6m^2}}} = \frac{1}{\rho_m + \sqrt{\nu_m}}. \quad (3.86)$$

Lembrando se que o primeiro e o segundo momento da distribuição  $U(0, 1)$  são, respectivamente  $\rho_u = \frac{1}{2}$  e  $\nu_u = \frac{1}{3}$  podemos ver também que

$$\lim_{m \rightarrow \infty} \lambda_c^*(m) = \frac{1}{\rho_u + \sqrt{\nu_u}}. \quad (3.87)$$

Fechamos esta seção concluindo que, neste caso particular, chegamos ao mesmo resultado que na seção anterior usando outra abordagem.

## Conclusão

O processo de rede com perdas existe em  $\mathbb{R}$  e converge à sua única medida invariante se

$$\lambda < \frac{1}{\rho + \sqrt{\nu}}$$

## Capítulo 4

# Algoritmo Backward-Forward (BFA) & Redes com perdas

*Simulações perfeitas* ou *amostragem exata* são rótulos para um conjunto de técnicas recentemente desenvolvidas para produzir um resultado cuja distribuição é garantida a seguir a lei de probabilidade desejada.

### 4.1 O Algoritmo “Backward-Forward” (BFA)

O BFA é uma das técnicas para simular perfeitamente e foi introduzido por Fernández, Ferrari e Garcia (2000). Este algoritmo é aplicável para qualquer medida que (1) seja absolutamente contínua com respeito a um processo pontual de Poisson e (2) possa ser obtida como medida invariante de um processo espacial de nascimento e morte. Este esquema tem características próprias:

- Não há acoplamento envolvido, conseqüentemente é insensitivo à presença de monotonicidade.
- O esquema amostra diretamente de uma janela finita da medida em equilíbrio em *volume infinito*
- Esta construção tem o valor adicional de ser comprovadamente uma ferramenta teórica para a análise de propriedades da medida a ser estudada. Por exemplo, em Fernández, Ferrari e Garcia (2001) esta construção foi utilizada para obter propriedades de “mixing”, correções a volume finito e distribuição assintótica (na temperatura) de “defeitos” das medidas extremas invariantes por translação para o modelo de Ising a baixa temperatura.

Neste esquema, a parte relevante é que o processo de nascimento e morte é construído em direção ao passado. Esta construção naturalmente induz uma percolação orientada ao passado, obtida quando olhamos os “ancestrais”, isto é, aqueles nascimentos ocorridos no passado que possam ter influência no nascimento sob consideração. Isto constitui o passo para trás na simulação. Uma vez que o clã de ancestrais é obtido, o algoritmo procede em movimento para a frente “limpando” as gerações sucessivas de acordo com as taxas de nascimento apropriadas a cada processo sob consideração. Portanto o BFA é apropriado para medidas invariantes de processos que podem ser obtidos como “emagrecimentos” ou “podas”, determinísticas ou estocásticas, de processos de nascimento ou morte de taxas constantes. O primeiro passo do algoritmo, construção do processo livre, é sempre possível mas o segundo passo, “limpeza”, é possível desde que número de operações envolvidas em “emagrecimento” seja finito. Então, o algoritmo é aplicável sempre que o processo de percolação orientada associado ao processo livre é sub-crítico.

## 4.2 Aplicação ao processo de redes com perdas

Para aplicar o BFA no processo de rede com perdas precisamos que a distribuição  $\pi$  seja uniformemente limitada.  $\pi$  pode ser, por exemplo, uma distribuição Uniforme ou Beta, e definimos  $M = \inf\{y \mid \pi((0, y)) = 1\}$ .

**Um algoritmo para construir o clã de ancestrais de uma região finita  $\Lambda = [a, b] \subset \mathbb{R}$**

- C1. Gerar o processo livre  $\alpha_0 = \{\xi_1^0, \dots, \xi_N^0\}$ ; processo de Poisson  $\lambda$ -homogêneo no intervalo  $[a - M, b]$  usando algoritmo PP (definido na Seção 2.1.1);  
 $s_L^0 = a$ ;  $s_R^0 = b$ .
- C2. Gerar  $U_1^0, \dots, U_N^0$  variáveis aleatórias iid com distribuição comum  $\pi$  e seja  $\eta = \emptyset$ .  
 Para cada  $i$  de 1 até  $N$

$$\text{se } (\xi_i^0, \xi_i^0 + U_i^0) \cap [a, b] \neq \emptyset \text{ então } \eta = \eta \cup (\xi_i^0, \xi_i^0 + U_i^0) \quad (4.1)$$

Simplemente, vamos gerar retângulos, cujas bases intersectam  $[a, b]$ . Agora temos  $n_0 = |\eta| \leq N$  bases.

- C3. Gerar  $S_1^0, \dots, S_{n_0}^0$  variáveis aleatórias iid com distribuição Exponencial(1) e construir os seguintes retângulos:

$$\mathbf{R}_0 = \{(\xi_i^0, \xi_i^0 + U_i^0) \times [-S_i^0, 0]; i = 1, \dots, n\}. \quad (4.2)$$

Agora considere o seguinte subconjunto de  $\mathbb{R} \times (-\infty, 0]$

$$\Lambda_0 = \bigcup_{i=1}^n (\xi_i^0 - M, \xi_i^0 + U_i^0) \times [-S_i^0, 0] \quad (4.3)$$

C4.  $k=1$ ;  $\Delta = \Lambda_0$ ;

$$\begin{aligned} \text{C5. } s_L^k &= \min(s_L^{k-1}, \min_{i \leq n_{k-1}} (\xi_i^{k-1} - M)) \\ s_R^k &= \max(s_R^{k-1}, \max_{i \leq n_{k-1}} (\xi_i^{k-1} + U_i^{k-1})) \end{aligned}$$

C6. Gerar um processo de Poisson  $\lambda$ -homogêneo  $\{(\xi_1^k, \tau_1^k), \dots, (\xi_{n_k}^k, \tau_{n_k}^k)\}$  em  $\Delta \cup [s_L^k, s_L^{k-1}) \cup (s_R^{k-1}, s_R^k]$  usando o Algoritmo PP.

C7. Gerar  $U_1^k, \dots, U_{n_k}^k$  v.as iid com distribuição  $\pi$ ;  $S_1^k, \dots, S_{n_k}^k$  v.as iid. com distribuição  $\text{Exp}(1)$  e construir os retângulos

$$\mathbf{R}_k = \{(\xi_i^k, \xi_i^k + U_i^k) \times [\tau_i^k - S_i^k, \tau_i^k]; i = 1, \dots, n_k\}, \quad (4.4)$$

e considere

$$\Lambda_k = \bigcup_{i=1}^{n_k} (\xi_i^k - M, \xi_i^k + U_i^k) \times [\tau_i^k - S_i^k, 0] \quad (4.5)$$

C8. – se  $n_k = 0$  então faça o clã de ancestrais de  $\eta$

$$A^\eta := \bigcup_{i=0}^{k-1} \mathbf{R}_i \quad (4.6)$$

e PARE.

- caso contrário, faça
- $\Delta = \Lambda_k \setminus \Lambda_{k-1}$ ;
- $k=k+1$ ;
- volta para C5;

Sob o ponto da vista prático, a fim de melhorar velocidade do programa executando este algoritmo, sugerimos no passo C6 excluir da consideração os retângulos satisfazendo  $\tau_i^k - S_i^k < \min_{j=1, \dots, n_{k-1}} (\tau_j^{k-1} - S_j^{k-1})$ .

Acabamos de construir o clã de ancestrais (do processo livre) indo para trás, ou seja do tempo 0 até um ponto no passado o que corresponde ao passo *BACKWARD* do esquema. O próximo passo é voltar ao passado no início do clã e a partir daí ir para o tempo

0, decidindo quais retângulos serão mantidos; o que corresponde ao passo *FORWARD* do esquema. Assim, ao conjunto  $A^n$  vamos aplicar um procedimento determinístico de “limpeza”, de acordo com a regra de exclusão da definição do processo. Em particular, caso a capacidade da rede seja igual a  $C$  (quando nenhum ponto espaço-temporal  $(x, t)$  possa pertencer a mais de  $C$  retângulos), o procedimento fará o seguinte: se  $(x, t)$  pertencer a mais de  $C$  retângulos, então manter os  $C$  retângulos mais velhos e apagar os restantes.

### O algoritmo de limpeza

Chamamos de  $\mathbf{T}$  o conjunto de retângulos para testar e de  $\mathbf{K}$  o conjunto de retângulos mantidos.

L1.  $\mathbf{K}=\emptyset$ ;  $\mathbf{T}= A^n$ ;

L2. Se  $\mathbf{T} = \emptyset$  vá para L4.

se não, ordene  $\mathbf{T}$  pelo tempo de nascimento. Seja  $R_1$  o primeiro retângulo em tal ordem.

$\mathbf{K} = \mathbf{K} \cup R_1$ ;  $\mathbf{T} = \mathbf{T} \setminus R_1$

L3. Dependendo de capacidade da rede,  $C$

1. Se  $C = 1$ ; Para todo  $R \in \mathbf{T}$  tal que  $R \approx R_1$ ,  $\mathbf{T} = \mathbf{T} \setminus R$ .  
volta para L2.

2. Se  $C > 1$ ;

para  $i=1$  até  $|\mathbf{T}| - C$

$R_i \in \mathbf{T}$ , se  $R_i \approx R_1$  então chamemos de  $Area=R_i \cap R_1$ ,  $C(Area)=2$  e  $\mathbf{K} = \mathbf{K} \cup R_i$ ;  $\mathbf{T} = \mathbf{T} \setminus R_i$ ;

para  $j=1$  até  $|\mathbf{T}|$

se  $R_j \cap Area \neq \emptyset$  então  $C(Area) = C(Area) + 1$  se  $C(Area) > C$  então  $\mathbf{T} = \mathbf{T} \setminus R_j$

volta para L2.

L4. Faça  $\mathbf{K}^n = \mathbf{K}$  e PARE.

Obtendo  $\mathbf{K}^n$ , define se

$$\eta^*(\gamma) = \begin{cases} 1 & \eta(\gamma) = 1 \text{ e } \exists R \in \mathbf{K}^n \text{ tal que Basis}(R) = \gamma \\ 0 & \text{caso contrário} \end{cases} \quad (4.7)$$

A Teorema 3.18 em Fernández et al. (2000) garante que a configuração  $\eta^*$  tem distribuição da medida invariante do processo de rede com perdas.

### 4.2.1 Simulações

Apresentamos os resultados obtidos usando o algoritmo BFA para diferentes valores de  $\lambda$ ,  $C$  ( a capacidade do cabo) e janela  $\Lambda$ . A distribuição  $\pi$  é  $\mathbf{U}(0, 1)$ . Neste caso, segundo (3.55),  $\lambda < 0.9282$  garante a possibilidade de construção .

Os programas para as simulações foram feitos em MATLAB 5.0. A fim de facilitar o entendimento da lógica do algoritmo, apresentamos separadamente o resultado do passo para trás, geração do clã de ancestrais, e o resultado da “limpeza”. As bases de retângulos sobreviventes, vivos em tempo  $t = 0$  são amostras da medida invariante do processo.

Figura 4.1: O clã de ancestrais para  $U(0, 1)$ ,  $\lambda = 0.5$ ,  $\Lambda = [0, 10]$ .

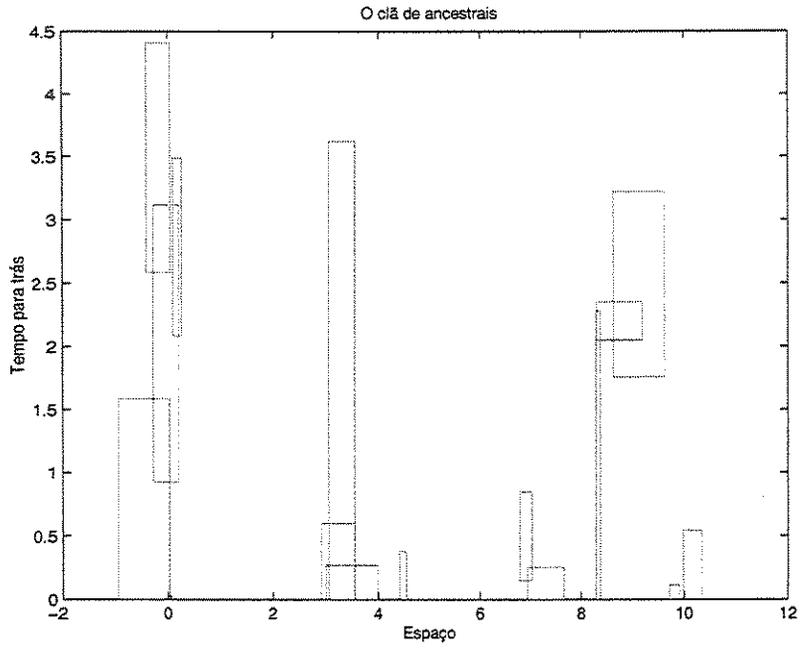


Figura 4.2: Resultado de limpeza para  $C = 1$  do clã na Figura 4.1.

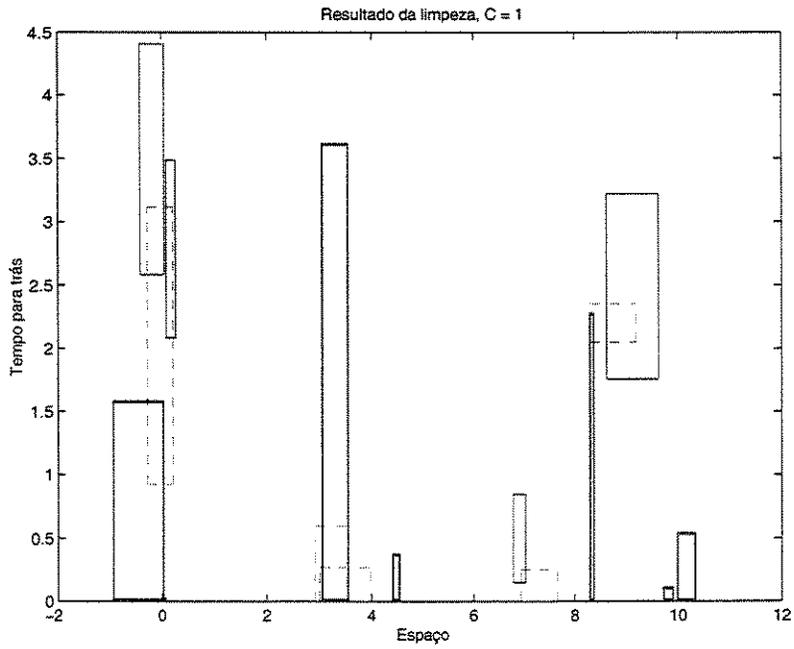


Figura 4.3: O clã de ancestrais para  $U(0, 1)$ ,  $\lambda = 0.9$ ,  $\Lambda = [0, 10]$ .

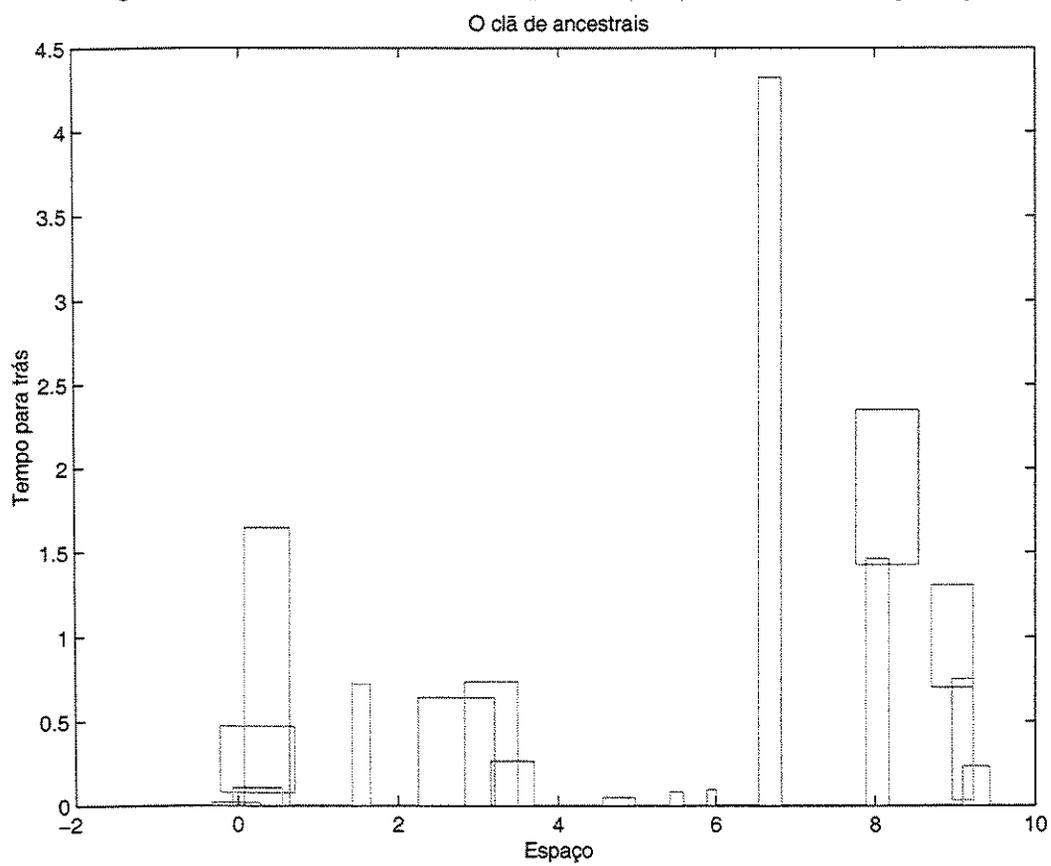


Figura 4.4: Resultado de limpeza do clã na Figura 4.3 para a)  $C = 1$  b)  $C = 2$ .

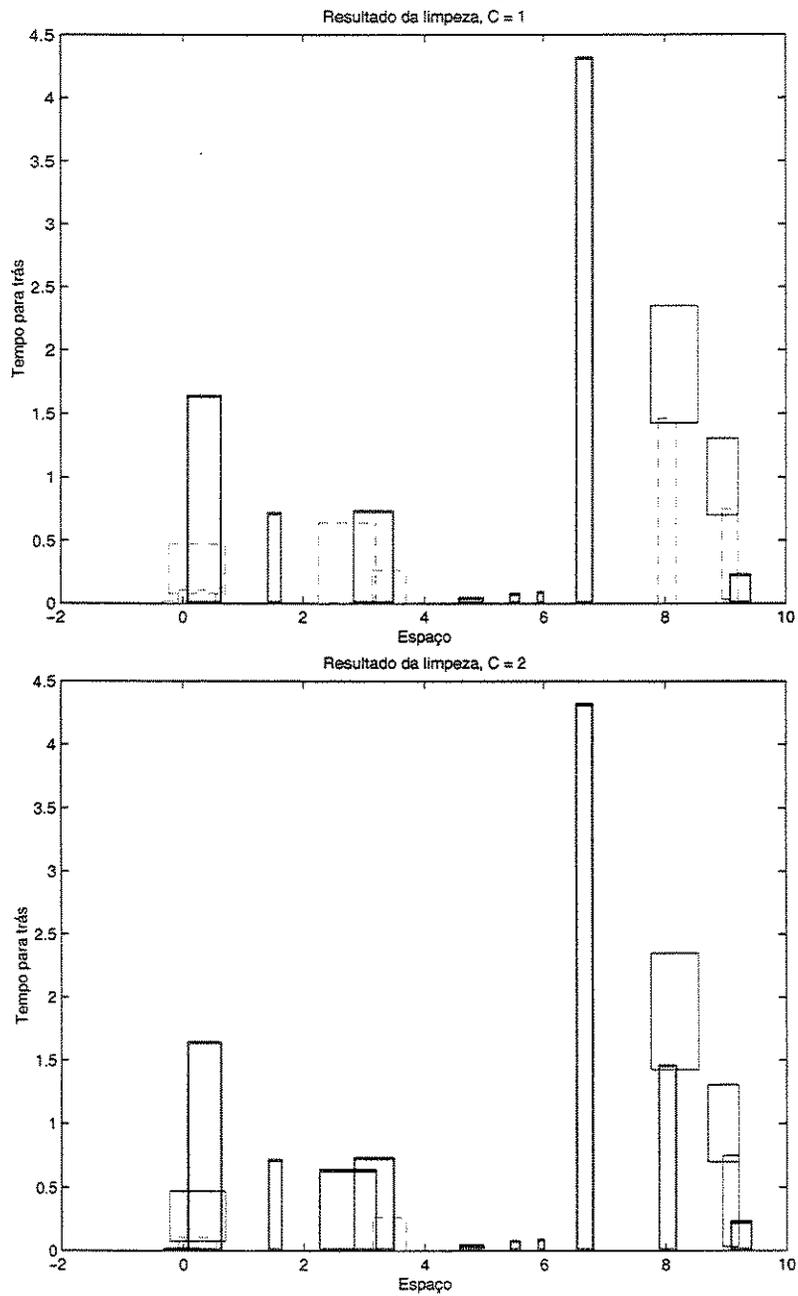


Figura 4.5: O clã de ancestrais para  $U(0, 1)$ ,  $\lambda = 1$ ,  $\Lambda = [0, 10]$ .

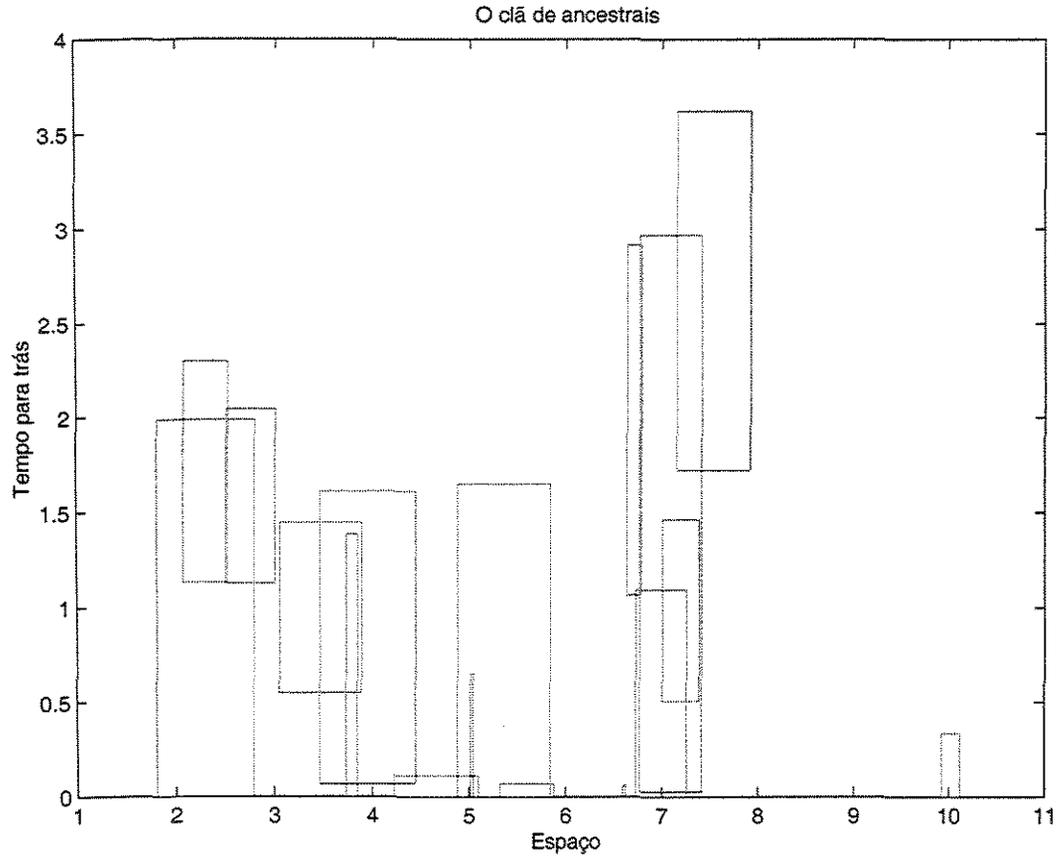


Figura 4.6: Resultado de limpeza do clã na Figura 4.5 para a)  $C = 1$  b)  $C = 2$ .

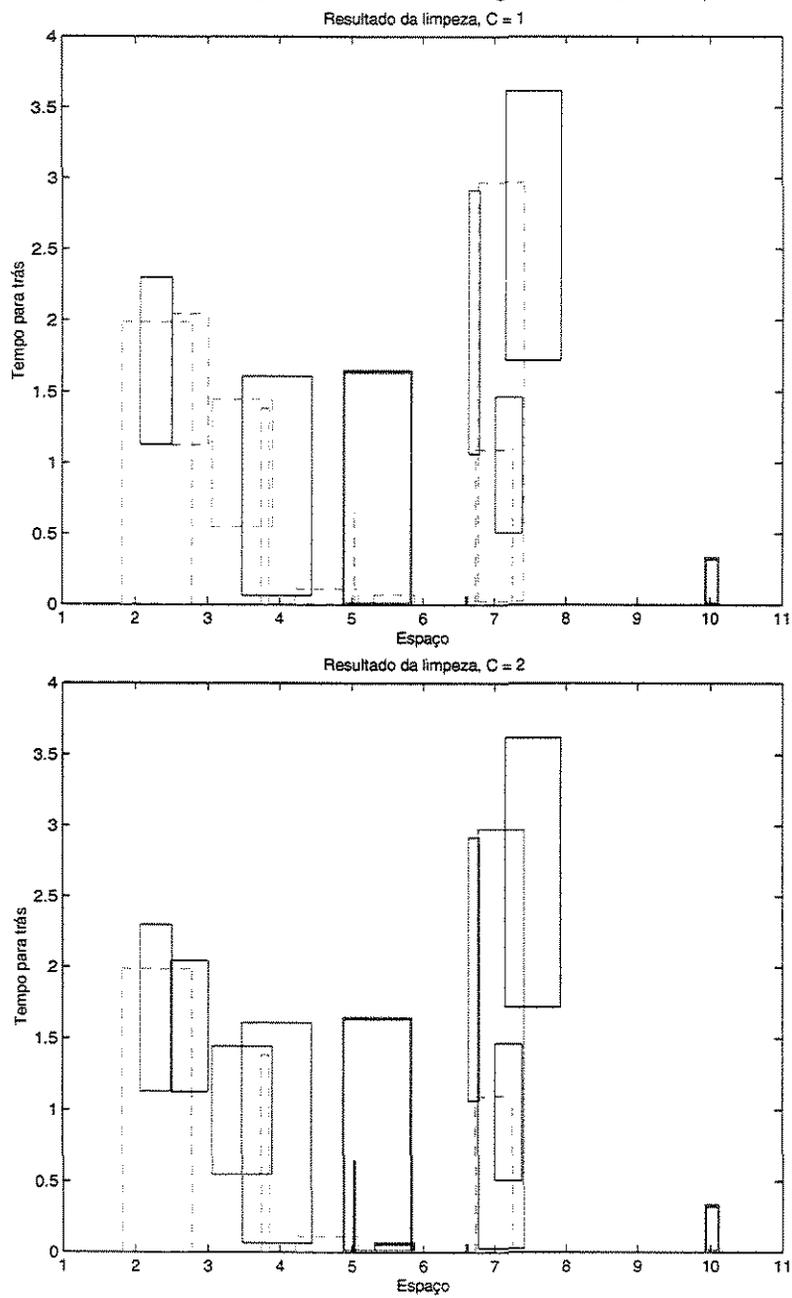


Figura 4.7: O clã de ancestrais para  $U(0, 1)$ ,  $\lambda = 1.2$ ,  $\Lambda = [0, 8]$ .

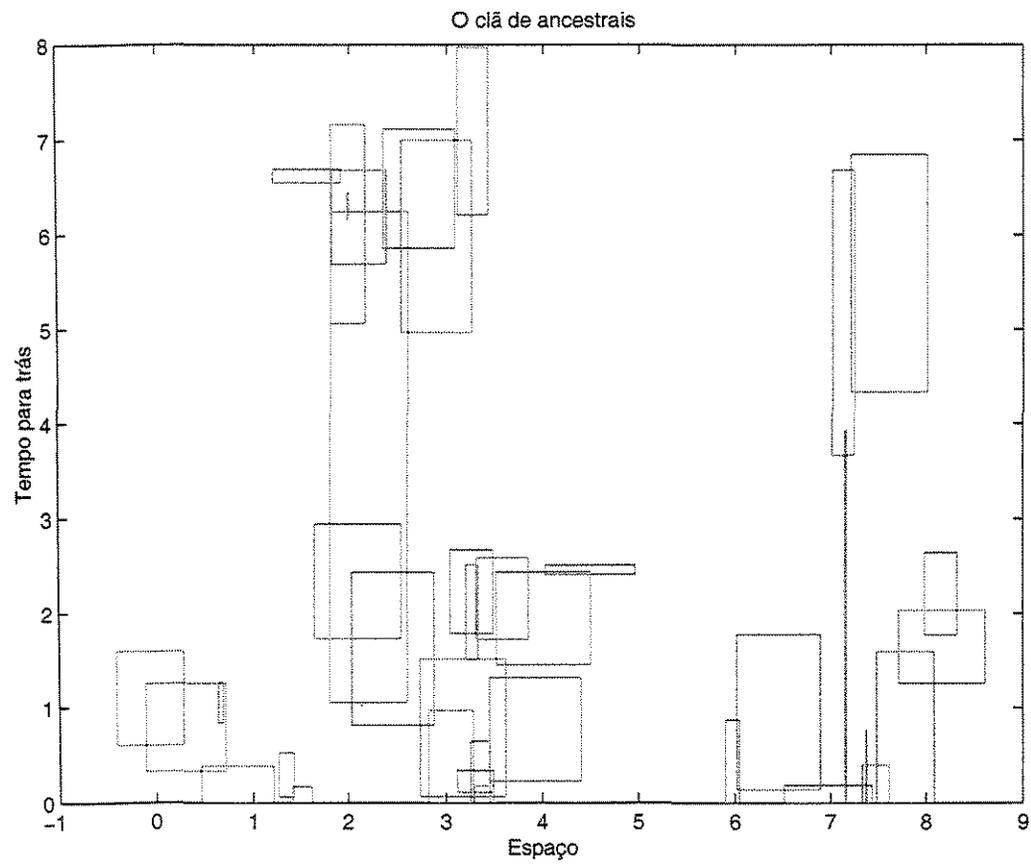


Figura 4.8: Resultado de limpeza do clã na Figura 4.7 para a)  $C = 1$  b)  $C = 3$ .

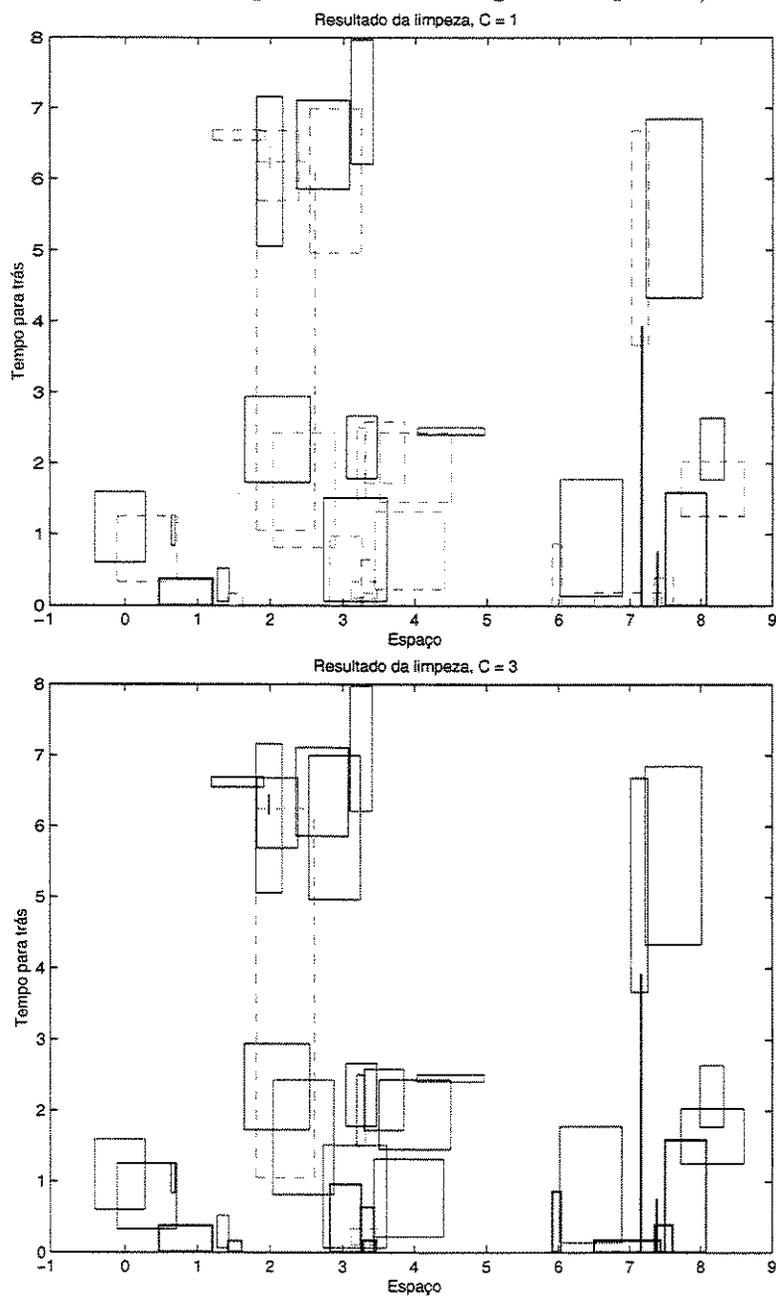


Figura 4.9: O número total de retângulos ( $E(N)$ ) no clã de ramificação e de rede com perdas para a)  $U(0, 1)$  b)  $d = 0.5$

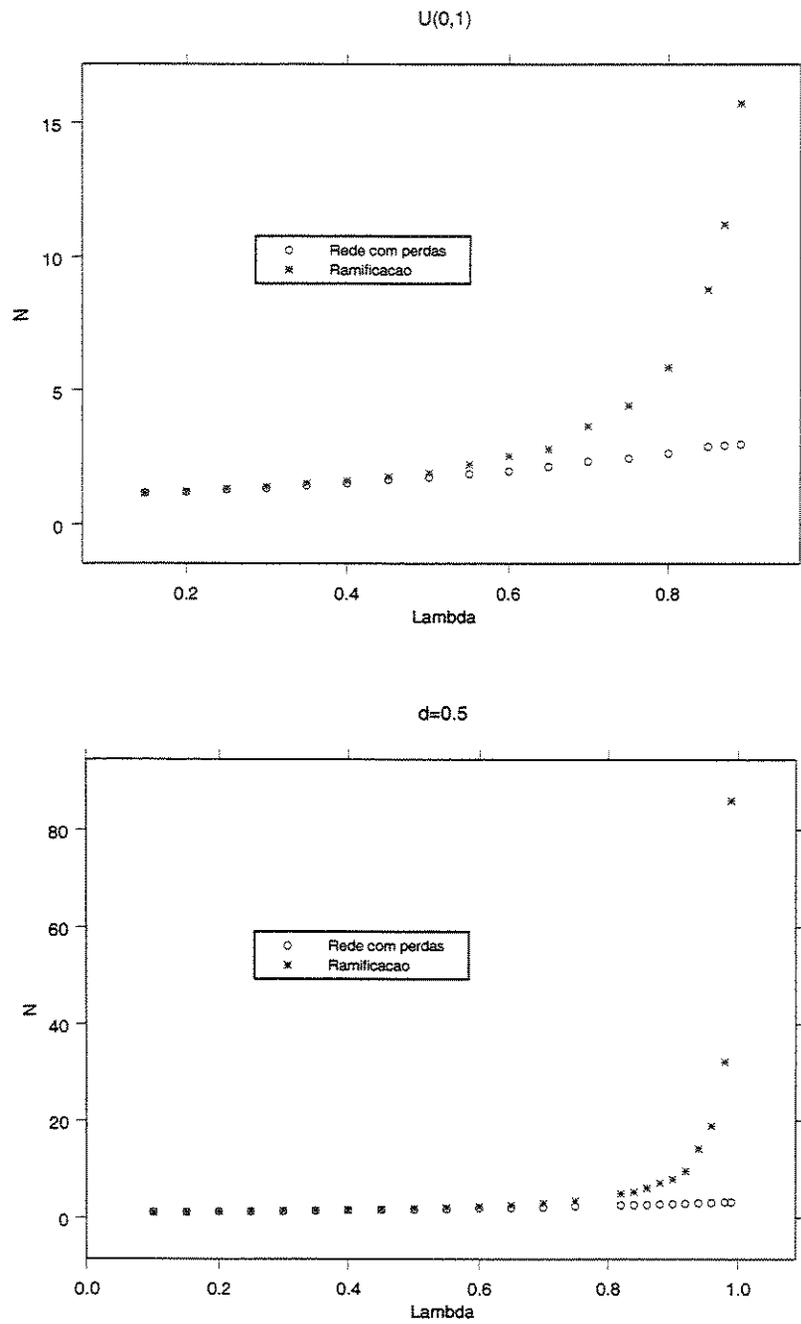


Figura 4.10: Comprimento Temporal ( $E(TL)$ ) do clã de ramificação e de rede com perdas para a)  $U(0,1)$  b)  $d = 0.5$

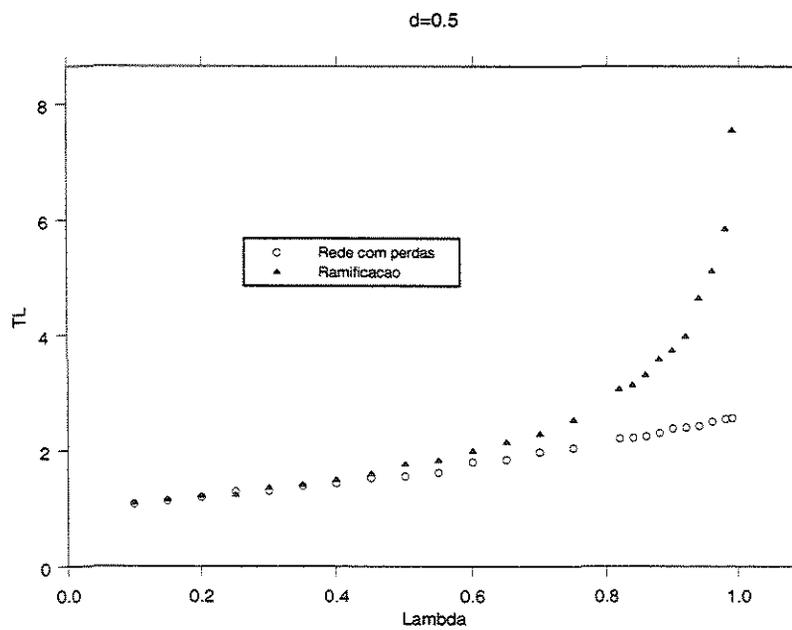
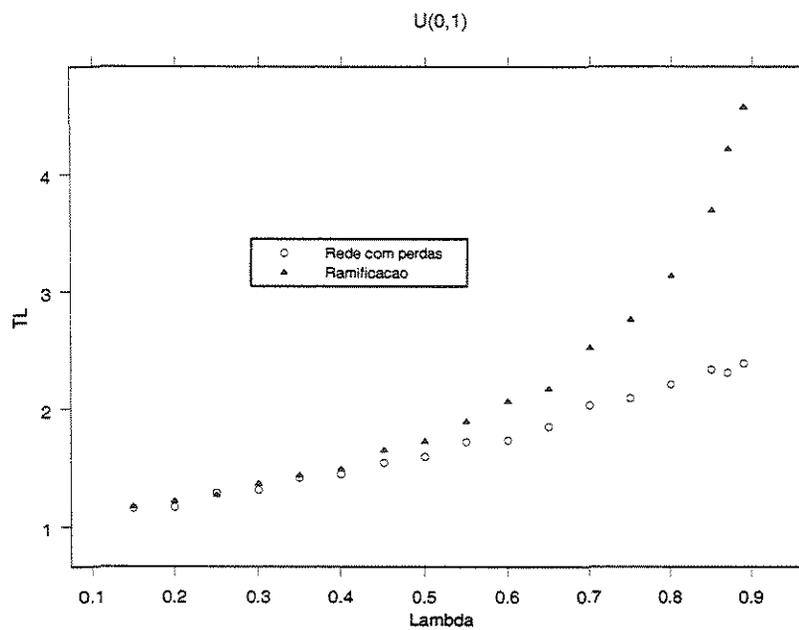
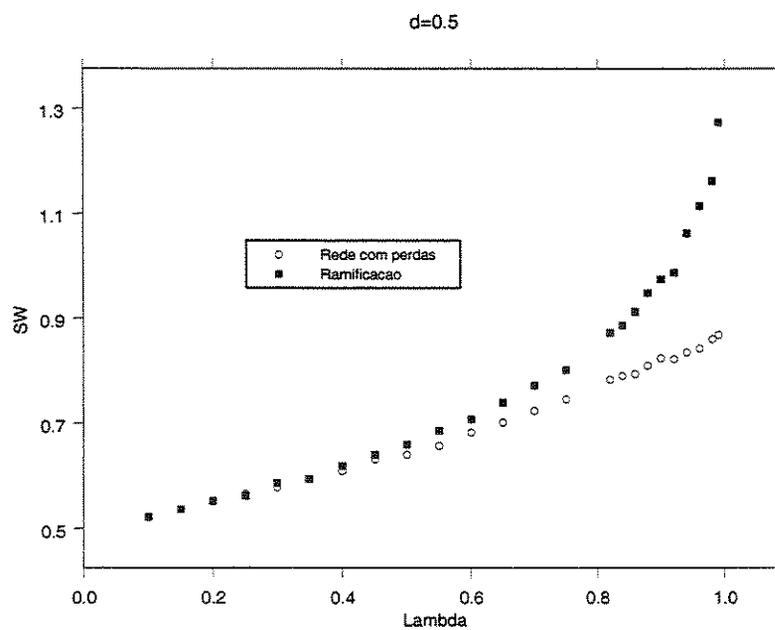
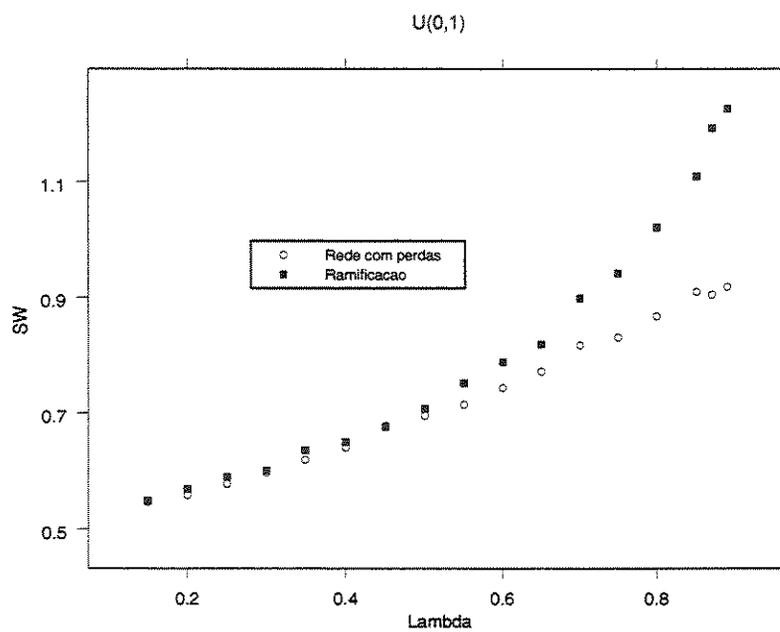


Figura 4.11: Largura espacial ( $E(SW)$ ) do clã de ramificação e de rede com perdas para a)  $U(0,1)$  b)  $d = 0.5$



O tamanho espaço-temporal do clã de ancestrais de rede com perdas é descrito pelas v.a.'s  $SW$  e  $TL$ . As distribuições destas v.a.'s são ainda uma questão aberta. Usando nossas simulações é possível obter umas informações adicionais sobre elas. Apresentamos ilustrativamente, alguns histogramas (Figuras 4.12, 4.13 e 4.14).

Figura 4.12: Histograma de  $TL$  para  $U(0, 1)$  e  $\lambda = 1$

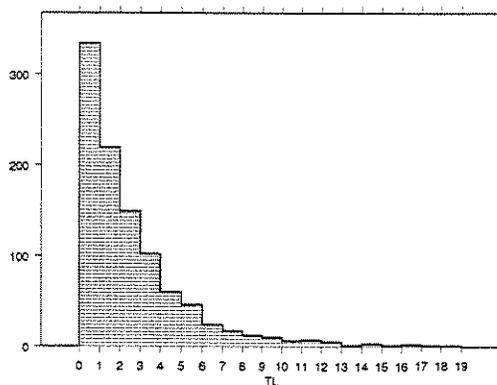


Figura 4.13: Histograma de  $SW$  para  $U(0, 1)$  e  $\lambda = 1$

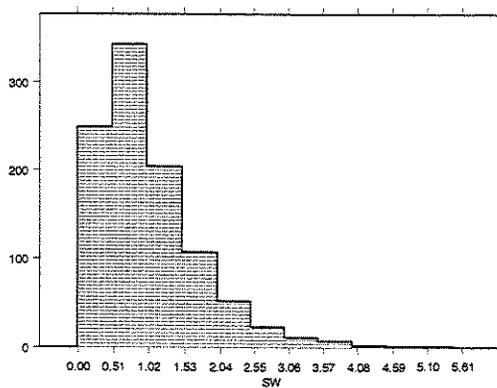
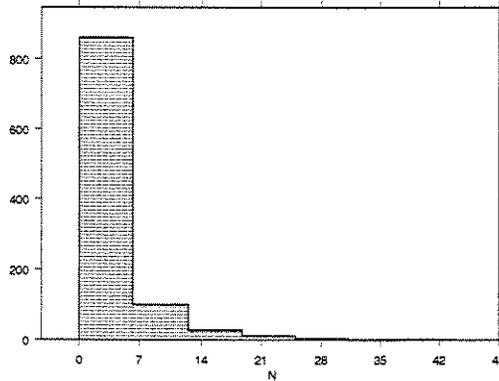


Figura 4.14: Histograma de  $N$  para  $U(0, 1)$  e  $\lambda = 1$



### 4.3.1 Lambda crítico via simulação

Vamos ver o que acontece com o processo de rede com perdas quando  $\lambda > \lambda_c^*$ . A ideia é gerar amostras aumentando  $\lambda$  e acompanhar o que acontece com  $N$ . Perto de  $\lambda_c$  o número total de retângulos deveria crescer muito, em média i.e.  $E(N) \rightarrow \infty, \lambda \rightarrow \lambda_c$ .

Para o processo de ramificação, onde conhecemos o valor crítico  $\lambda_c^*$ ,  $E(N)$  estimado se comporta exatamente assim. Até visualmente, Figura 4.9, diríamos que  $E(N)$  tem uma assíntota vertical em, ou perto de  $\lambda_c^*$ .

Então nossa suposição foi que o clã de ancestrais para o processo de rede com perdas exibe o comportamento análogo. Se  $E(N)$  estimado exibir o crescimento significativo quando  $\lambda$  se aproxima a um valor, acreditamos que este valor é exatamente  $\lambda_c$ . Nossa pretensão é mais obter informação sobre a expressão algébrica geral do  $\lambda_c$  do que calculá-lo numericamente mais precisamente possível.

Daqui para frente, para todas as distribuições analisadas, foram gerados 1000 amostras de  $N$  para cada valor de  $\lambda$ . A Figura 4.15 contem os resultados obtidos para comprimento de chamada fixo,  $d = 0.5$ .

A primeira vista vê-se que esta função,  $E(N)$ , tem uma assíntota perto de 2.8. A fim de localizá-la, observamos a função  $1/\log(E(N))$  e tentamos achar o zero desta função (veja Figura 4.16). Nós usamos uma aproximação polinomial da função e localizamos uma raiz apropriada deste polinômio.

Neste caso, usando um polinômio de grau 19, obtemos o zero em  $\lambda = 2.8231$ .

Uma análise análoga foi feita para vários valores de  $d$  e para vários graus polinomiais (para detalhes veja Apêndice B). Resultados obtidos são sintetizados na Tabela 4.1 e

Figura 4.15: Número total de retângulos ( $E(N)$ ) para  $d = 0.5$

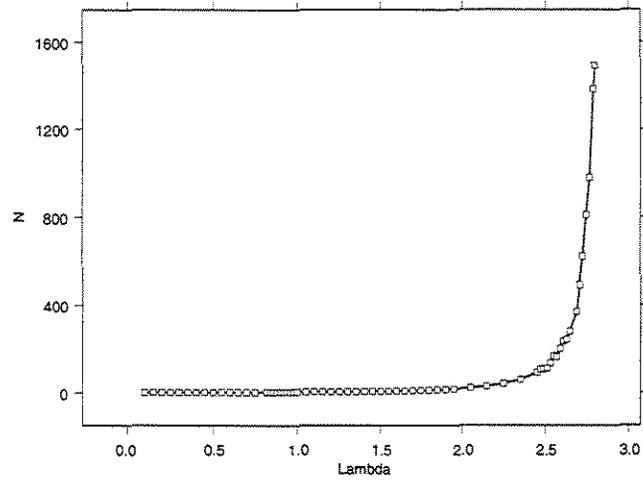
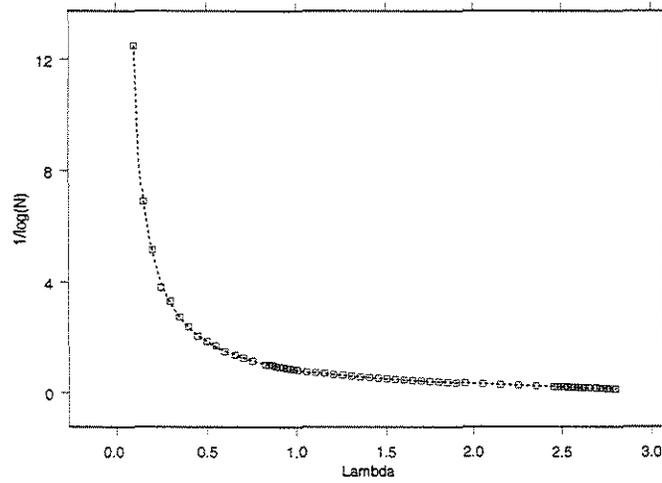
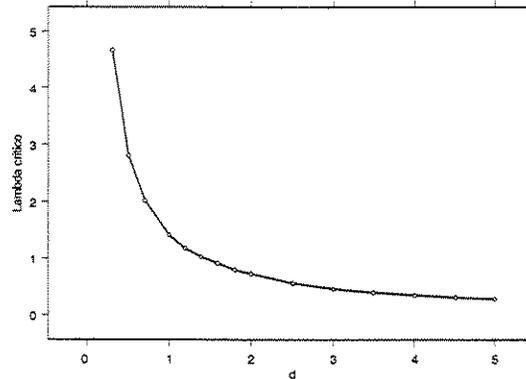


Figura 4.16:  $1/\log(E(N))$  para  $d = 0.5$



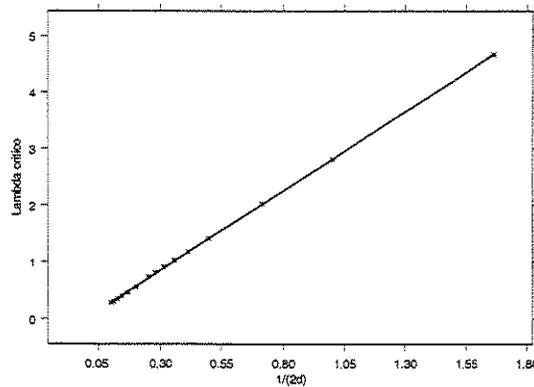
visualizados na Figura 4.17.

Figura 4.17: Lambda crítico obtido via simulações para comprimento de chamada fixo



O próximo passo é comparar, para mesmos valores de  $d$ ,  $\lambda_c^* = \frac{1}{2d}$  e  $\lambda_c$ .

Figura 4.18:  $\lambda_c^*$  vs.  $\lambda_c$  para comprimento de chamada fixo



Havendo uma óbvia tendência linear foi ajustado, usando mínimos quadrados ponderados, o seguinte modelo:

$$\lambda_c = 2.8088 \cdot \frac{1}{2d} + 0.013. \quad (4.11)$$

Isto é já um resultado interessante. Embora não saibamos nada, teoricamente, sobre  $\lambda_c$  é interessante somente a idéia que todas características de  $\pi$  são refletidas no  $\lambda_c^*$  e que

Tabela 4.1: Lambda crítico obtido via simulações para comprimento de chamada fixo

d	$\lambda_c$
0.3	4.6688
0.5	2.8231
0.7	2.0641
1.0	1.4193
1.2	1.1905
1.4	1.0254
1.6	0.9120
1.8	0.8084
2.0	0.7312
2.5	0.5682
3.0	0.4537
3.5	0.3931
4.0	0.3530
4.5	0.3103
5.0	0.2833

a diferença entre  $\lambda_c$  e  $\lambda_c^*$  depende somente de como o processo de ramificação domina o clã de ancestrais.

Observamos que o intercepto do modelo (4.11) já é pequeno mas intuitivamente nem esperamos de um intercepto. Se o comprimento da chamada tende a infinito então  $\lambda_c^* \rightarrow 0$ , e também  $\lambda_c$  tem que tender ao 0. Portanto, ajustamos um modelo linear sem intercepto:

$$\lambda_c = 2.8246 \cdot \frac{1}{2d}. \quad (4.12)$$

O resultado (4.12) induz uma pergunta natural: Será que uma relação análoga vale para outras distribuições uniformemente limitadas?

Usando o mesmo método, achamos  $\lambda_c$  para distribuição  $U(0,1)$  e também para alguns combinações de parâmetros da distribuição *Beta*. Na Tabela 4.2 estabelecemos os resultados obtidos, juntamente com razão  $\lambda_c/\lambda_c^*$ .

Tabela 4.2: Lambda crítico obtido via simulações para distribuições *Uniforme* e *Beta*

Distribuição	$\lambda_c$	$\lambda_c/\lambda_c^*$
U(0,1)	2.6135	2.8157
Beta(2,1)	2.0888	2.8695
Beta(2,2)	2.6746	2.8022
Beta(3,1)	1.8597	2.8353
Beta(3,2)	2.3079	2.8444
Beta(1,2)	3.7981	2.8166

Da Tabela 4.2 vêemos que os  $\lambda_c$ 's obtidos são bem próximos ao  $2.82\lambda_c^*$ . Precisamente, a reta ajustada neste caso ( de novo usando mínimos quadrados, sem intercepto) é

$$\lambda_c = 2.8243 \cdot \lambda_c^*. \quad (4.13)$$

Resíduos de todos os casos estão no Apêndice B. Lembramos que as distribuições *Beta*(2,1), *Beta*(3,1), ... são assimétricas, significamente diferente do que, por exemplo, *U*(0,1). É lógico supor, se a diferença entre  $\lambda_c$  e  $\lambda_c^*$  contivesse alguma característica da distribuição  $\pi$ , e.g. primeiro momento, esta diferença deveria variar muito mais entre distribuições diferentes. A única característica em comum para estas distribuições é que são definidas i.e. têm densidades positivas no intervalo (0,1). O comprimento máximo, nestes casos igual à 1, também podemos excluir da consideração pois já foram analisados vários comprimentos fixos diferentes de 1.

## Conjectura

Levando em consideração todos os resultados obtidos oferecemos a conjectura:

$$\lambda_c = c\lambda_c^* \tag{4.14}$$

onde  $c$  é uma constante, mais precisamente,  $\lambda_c$  para comprimento de chamada fixo,  $d = 0.5$  e

$$c \approx 2.82. \tag{4.15}$$

# Capítulo 5

## Conclusão

Nós verificamos a viabilidade do algoritmo “Baskward-Forward” para o processo de redes com perdas. Sendo um algoritmo introduzido recentemente, isso é uma das primeiras implementações já feitas. As simulações baseadas neste algoritmo mostraram múltipla utilidade. Além de servir a obter amostras exatas de medida invariante, elas são uma ferramenta poderosa para explorar várias características do processo. Nós as usamos principalmente para obter as condições de sub-criticalidade do processo de percolação orientada (contínua) associado. Neste caso, quando quase nada é conhecido sobre o processo, as simulações podem ser de alta importância. Obter um resultado via simulações facilita dirigir a pesquisa teórica.

Gerando um clã de ancestrais, obtemos amostras de várias variáveis aleatórias de interesse: comprimento temporal ( $TL$ ), largura espacial ( $SW$ ), número total de rectângulos ( $N$ ), tamanho de uma geração, número de gerações. Tendo uma vez uma amostra grande de qualquer uma destas v.a.'s podem se aplicar vários métodos estatísticos a fim de explorar suas distribuições. Achar qualquer uma destas distribuições levaria a obter valor crítico,  $\lambda_c$ . Também é possível usar outros acessos, como nós fizemos, localizando uma assíntota vertical de  $E(N)$  estimado e acabamos dando uma conjectura sobre a expressão geral de  $\lambda_c$ .

O tamanho do clã, descrito por  $SW$  e  $TL$ , embora ser importante é mal conhecido, e é aqui também onde se pode usufrir das simulações.

# Apêndice A

## Provas

### A.1 Prova do Lema 3.4.1

Seja  $\rho$  um autovalor de matriz  $A$  e  $\mathbf{x}$  um autovetor a direita correspondente a  $\rho$ . Da relação  $A\mathbf{x} = \rho\mathbf{x}$  segue

$$\rho x_i = \sum_{j=1}^m a(i, j)x_j = c_i \sum_{j=1}^m x_j + \sum_{j=1}^m c_j x_j = c_i S + D \quad (\text{A.1})$$

onde

$$S = \sum_{j=1}^m x_j, \quad D = \sum_{j=1}^m c_j x_j. \quad (\text{A.2})$$

Então

$$\begin{aligned} \rho(x_2 - x_1) &= (c_2 - c_1)S \Rightarrow x_2 = x_1 + \frac{S}{\rho}(c_2 - c_1) \\ \rho(x_3 - x_1) &= (c_3 - c_1)S \Rightarrow x_3 = x_1 + \frac{S}{\rho}(c_3 - c_1) \\ &\vdots \\ \rho(x_m - x_1) &= (c_m - c_1)S \Rightarrow x_m = x_1 + \frac{S}{\rho}(c_m - c_1). \end{aligned} \quad (\text{A.3})$$

Agora

$$S = \sum_{j=1}^m x_j = mx_1 + \frac{S}{\rho} \left( \sum_j c_j - mc_1 \right)$$

$$\begin{aligned}
&\Rightarrow S\left(1 - \frac{1}{\rho}\left(\sum_j c_j - mc_1\right)\right) = mx_1 \\
&\Rightarrow S = \frac{\rho mx_1}{\rho - \left(\sum_j c_j - mc_1\right)}. \tag{A.4}
\end{aligned}$$

De (A.3) e (A.4) temos explicitamente

$$\begin{aligned}
x_i &= x_1 + \frac{mx_1(c_i - c_1)}{\rho - \left(\sum_j c_j - mc_1\right)}, \quad i = 1, \dots, m \\
x_i &= x_1(1 + d_i), \quad \text{onde } d_i = \frac{m(c_i - c_1)}{\rho - \left(\sum_j c_j - mc_1\right)}. \tag{A.5}
\end{aligned}$$

Vamos calcular  $D$ , usando (A.2) e (A.5)

$$\begin{aligned}
D &= \sum_{i=1}^m c_i x_i = \sum_i c_i x_1 (1 + d_i) \\
&= x_1 \sum_i c_i \left(1 + \frac{m(c_i - c_1)}{\rho - \left(\sum_j c_j - mc_1\right)}\right) \\
&= \frac{x_1}{\rho - \left(\sum_j c_j - mc_1\right)} \sum_i c_i \left(\rho - \sum_j c_j + mc_1 + mc_i - mc_1\right) \\
&= \frac{x_1}{\rho - \left(\sum_j c_j - mc_1\right)} \left(\rho \sum_i c_i - \left(\sum_i c_i\right)^2 + m \sum_i (c_i)^2\right). \tag{A.6}
\end{aligned}$$

De (A.1), (A.4) e (A.6) temos

$$\rho x_1 = c_1 S + D = c_1 \frac{\rho mx_1}{\rho - \left(\sum_j c_j - mc_1\right)} + \frac{x_1 \left(\rho \sum_i c_i - \left(\sum_i c_i\right)^2 + m \sum_i (c_i)^2\right)}{\rho - \left(\sum_j c_j - mc_1\right)}. \tag{A.7}$$

Se  $\rho \neq 0$  então, pelo (A.5),  $x_1$  tem que ser diferente de 0 ( caso contrário daria  $\mathbf{x} = \mathbf{0}$ ).

Portanto, supondo que  $\rho \neq 0$  tem-se

$$\begin{aligned}
\rho &= \frac{\rho c_1 m + \rho \sum_i c_i - \left(\sum_i c_i\right)^2 + m \sum_i c_i^2}{\rho - \left(\sum_j c_j - mc_1\right)} \\
&\Rightarrow \rho \left(\rho - \sum_j c_j + mc_1\right) = \rho c_1 m + \rho \sum_i c_i - \left(\sum_i c_i\right)^2 + m \sum_i c_i^2 \\
&\Rightarrow \rho^2 - 2\rho \sum_j c_j + \left(\sum_i c_i\right)^2 - m \sum_i c_i^2 = 0. \tag{A.8}
\end{aligned}$$

E portanto únicos autovalores diferentes de 0 são raízes de (A.8) e assim o parte **a** do Lema 3.4.1 é provado.

Como  $c_i$ 's são não negativos então  $\sum_i c_i \geq 0$  e portanto

$$\sum_{i=1}^m c_i + \sqrt{m \sum_{i=1}^m c_i^2} \geq \left| \sum_{i=1}^m c_i - \sqrt{m \sum_{i=1}^m c_i^2} \right| \quad (\text{A.9})$$

da onde seguiu o parte **b** do Lema 3.4.1 ■

# Apêndice B

## Tabelas

Tabela B.1:  $E(N)$  estimado para comprimento de chamada fixo  $d = 0.3, 0.5, 0.7, 1, 1.2$ .  
 Parte 1.

$d = 0.3$		$d = 0.5$		$d = 0.7$		$d = 1$		$d = 1.2$	
$\lambda$	$E(N)$	$\lambda$	$E(N)$	$\lambda$	$E(N)$	$\lambda$	$E(N)$	$\lambda$	$E(N)$
0.20	1.135	0.10	1.083	0.15	1.216	0.15	1.339	0.15	1.414
0.30	1.234	0.15	1.156	0.20	1.318	0.20	1.464	0.20	1.653
0.40	1.276	0.20	1.214	0.25	1.398	0.25	1.624	0.25	1.785
0.50	1.363	0.25	1.300	0.30	1.554	0.30	1.907	0.30	2.171
0.60	1.520	0.30	1.352	0.35	1.697	0.35	2.175	0.35	2.580
0.70	1.554	0.35	1.439	0.40	1.786	0.40	2.435	0.40	3.101
0.80	1.672	0.40	1.517	0.45	2.062	0.45	2.958	0.45	3.598
0.90	1.795	0.45	1.626	0.50	2.142	0.50	3.235	0.50	4.450
1.00	2.001	0.50	1.709	0.55	2.367	0.55	3.954	0.55	5.281
1.10	2.141	0.55	1.795	0.60	2.774	0.60	4.467	0.60	6.268
1.20	2.257	0.60	1.956	0.65	2.898	0.65	5.120	0.65	7.823
1.30	2.512	0.65	2.071	0.70	3.283	0.70	5.942	0.70	9.143
1.40	2.705	0.70	2.201	0.75	3.583	0.75	7.716	0.75	12.187
1.50	2.987	0.75	2.368	0.80	3.902	0.80	8.733	0.80	16.846
1.60	3.168	0.82	2.649	0.85	4.507	0.85	10.517	0.85	21.545
1.70	3.421	0.84	2.683	0.90	4.726	0.90	12.657	0.90	29.857
1.80	3.609	0.86	2.703	0.95	5.224	0.95	15.793	0.95	48.652
1.90	3.969	0.88	2.827	1.00	6.336	1.00	20.032	1.00	75.858
2.00	4.673	0.90	2.911	1.05	7.354	1.05	24.565	1.05	120.559
2.10	4.915	0.92	2.953	1.10	7.778	1.10	36.300	1.10	258.096
2.20	5.322	0.94	3.079	1.15	7.975	1.15	50.261	1.15	801.849
2.30	6.037	0.96	3.118	1.20	10.394	1.20	72.546	1.17	1711.200
2.40	6.512	0.98	3.256	1.25	12.021	1.21	80.372		
2.50	6.924	1.00	3.323	1.30	13.006	1.22	82.688		
2.60	7.310	1.05	3.535	1.35	16.143	1.23	91.056		
2.70	9.095	1.10	3.908	1.40	16.295	1.24	119.905		
2.75	9.942	1.15	4.107	1.45	22.511	1.25	124.091		
2.80	9.540	1.20	4.506	1.50	26.446	1.26	125.939		
2.85	10.289	1.25	4.704	1.55	34.033	1.27	145.051		
2.90	10.356	1.30	5.201	1.60	38.654	1.28	159.504		
2.95	12.432	1.35	5.682	1.65	44.313	1.29	200.018		
3.00	12.320	1.40	6.036	1.70	66.802	1.30	217.680		
3.05	14.037	1.45	6.593	1.75	89.968	1.31	235.920		

Tabela B.2:  $E(N)$  estimado para comprimento de chamada fixo  $d = 0.3, 0.5, 0.7, 1, 1.2$ .  
 Parte 2.

$d = 0.3$		$d = 0.5$		$d = 0.7$		$d = 1$		$d = 1.2$	
$\lambda$	$E(N)$	$\lambda$	$E(N)$	$\lambda$	$E(N)$	$\lambda$	$E(N)$	$\lambda$	$E(N)$
3.10	14.297	1.50	7.089	1.80	130.711	1.32	251.962		
3.15	14.796	1.55	7.695	1.85	196.165	1.33	340.534		
3.20	15.522	1.60	8.446	1.90	353.698	1.34	421.940		
3.25	17.786	1.65	9.202	1.95	607.008	1.35	476.737		
3.30	19.234	1.70	10.142	2.00	1391.300	1.37	644.185		
3.35	20.794	1.75	11.415			1.39	1108.700		
3.40	23.581	1.80	12.523						
3.45	23.829	1.85	14.065						
3.50	25.244	1.90	15.635						
3.55	28.238	1.95	18.114						
3.60	28.911	2.05	22.692						
3.65	31.412	2.15	29.772						
3.70	39.130	2.25	42.433						
3.75	44.063	2.35	60.644						
3.80	43.910	2.45	92.705						
3.85	54.305	2.47	107.317						
3.90	56.145	2.49	109.557						
3.95	61.377	2.51	112.119						
4.00	69.201	2.53	136.218						
4.05	75.108	2.55	169.394						
4.10	88.583	2.59	201.911						
4.15	116.642	2.61	236.056						
4.20	118.483	2.63	243.618						
4.25	151.000	2.65	281.506						
4.30	179.920	2.69	370.031						
4.35	225.495	2.71	491.837						
4.40	259.434	2.73	621.044						
4.45	342.334	2.75	811.504						
4.50	495.552	2.77	979.468						
4.55	642.388	2.79	1384.300						
4.60	964.958	2.80	1492.100						
4.62	1007.700								
4.64	1345.900								

Tabela B.3:  $E(N)$  estimado para comprimento de chamada fixo  $d = 1.4, 1.6, 1.8, 2, 2.5$ .

$d = 1.4$		$d = 1.6$		$d = 1.8$		$d = 2$		$d = 2.5$	
$\lambda$	$E(N)$	$\lambda$	$E(N)$	$\lambda$	$E(N)$	$\lambda$	$E(N)$	$\lambda$	$E(N)$
0.15	1.507	0.15	1.616	0.15	1.689	0.15	1.806	0.10	1.631
0.20	1.849	0.20	1.980	0.20	2.253	0.20	2.395	0.13	1.990
0.25	2.072	0.25	2.425	0.25	2.802	0.25	3.127	0.15	2.062
0.30	2.529	0.30	2.865	0.30	3.628	0.30	3.966	0.16	2.320
0.35	3.151	0.35	3.738	0.35	4.516	0.35	5.419	0.19	2.762
0.40	3.760	0.40	4.688	0.40	6.060	0.40	7.906	0.20	2.965
0.45	4.855	0.45	6.241	0.45	8.846	0.45	11.548	0.23	3.755
0.50	5.758	0.50	8.410	0.50	13.527	0.50	18.737	0.25	4.424
0.55	7.543	0.55	12.056	0.55	18.722	0.55	34.059	0.26	4.616
0.60	9.523	0.60	16.054	0.60	30.883	0.60	69.956	0.29	6.298
0.65	11.876	0.65	24.872	0.65	53.828	0.65	184.236	0.30	6.656
0.70	18.970	0.70	39.322	0.70	113.437	0.70	1700.600	0.32	8.019
0.75	26.828	0.75	70.507	0.75	441.483			0.35	10.871
0.80	44.275	0.77	100.727	0.77	1004.600			0.38	14.620
0.85	65.309	0.79	124.732					0.40	18.812
0.90	132.300	0.81	189.760					0.41	19.323
0.92	178.006	0.83	270.538					0.44	32.585
0.94	242.262	0.85	547.142					0.45	43.745
0.96	421.129	0.86	736.900					0.47	57.354
0.98	680.055	0.87	1111.700					0.50	105.565
1.00	1431.400							0.52	189.916
								0.54	467.666
								0.55	795.215

Tabela B.4:  $E(N)$  estimado para comprimento de chamada fixo  $d = 3, 3.5, 4, 4.5, 5$ .

$d = 3$		$d = 3.5$		$d = 4$		$d = 4.5$		$d = 5$	
$\lambda$	$E(N)$	$\lambda$	$E(N)$	$\lambda$	$E(N)$	$\lambda$	$E(N)$	$\lambda$	$E(N)$
0.05	1.302	0.09	1.852	0.03	1.220	0.04	1.355	0.04	1.450
0.08	1.580	0.13	2.672	0.05	1.404	0.06	1.661	0.06	1.732
0.11	2.006	0.15	3.052	0.06	1.559	0.07	1.833	0.08	2.278
0.14	2.300	0.17	3.954	0.08	1.821	0.09	2.271	0.10	2.954
0.15	2.484	0.20	5.446	0.09	1.961	0.10	2.599	0.12	3.716
0.17	3.143	0.21	5.862	0.11	2.513	0.12	3.262	0.14	5.344
0.20	3.898	0.24	9.125	0.14	3.559	0.13	3.771	0.16	7.220
0.23	5.256	0.25	9.686	0.15	3.827	0.15	4.906	0.18	11.751
0.25	6.602	0.28	18.784	0.17	4.683	0.16	5.869	0.20	17.323
0.26	6.991	0.30	23.252	0.18	5.516	0.18	8.146	0.22	30.306
0.29	9.744	0.32	36.809	0.20	7.224	0.19	9.664	0.24	65.785
0.30	12.179	0.35	91.125	0.22	10.483	0.21	14.109	0.26	215.972
0.32	14.568	0.36	127.575	0.23	12.251	0.22	16.260	0.28	1417.000
0.33	17.542	0.38	343.926	0.25	19.581	0.24	29.773		
0.35	25.480			0.26	23.513	0.25	37.847		
0.36	32.118			0.28	36.625	0.27	74.826		
0.39	57.137			0.29	53.655	0.28	122.408		
0.40	63.605			0.30	63.433	0.29	209.976		
0.42	125.890			0.32	162.766	0.30	489.938		
0.45	467.730			0.33	277.220				
				0.35	1478.200				

Tabela B.5:  $E(N)$  estimado para distribuições *Beta*

<i>Beta</i> (2, 1)		<i>Beta</i> (2, 2)		<i>Beta</i> (3, 1)		<i>Beta</i> (3, 2)		<i>Beta</i> (1, 2)	
$\lambda$	$E(N)$	$\lambda$	$E(N)$	$\lambda$	$E(N)$	$\lambda$	$E(N)$	$\lambda$	$E(N)$
0.20	1.293	0.25	1.313	0.20	1.328	0.25	1.371	0.20	1.141
0.30	1.515	0.35	1.435	0.30	1.588	0.35	1.538	0.30	1.235
0.40	1.797	0.45	1.581	0.40	1.954	0.45	1.856	0.40	1.350
0.50	2.188	0.55	1.892	0.50	2.322	0.55	2.152	0.50	1.444
0.60	2.774	0.65	2.190	0.60	2.827	0.65	2.464	0.60	1.541
0.70	3.126	0.75	2.438	0.70	3.718	0.75	2.905	0.70	1.696
0.80	3.639	0.85	2.842	0.80	4.472	0.85	3.676	0.80	1.854
0.90	4.463	0.95	3.283	0.90	5.738	0.95	4.088	0.90	1.948
1.00	5.941	1.05	3.826	1.00	7.367	1.05	5.286	1.00	2.357
1.10	7.141	1.15	4.440	1.10	9.767	1.15	6.121	1.10	2.626
1.20	9.674	1.25	4.990	1.20	13.043	1.25	7.876	1.20	2.859
1.30	12.484	1.35	5.797	1.30	19.343	1.35	9.693	1.30	3.111
1.40	15.845	1.45	7.427	1.40	30.495	1.45	12.836	1.40	3.391
1.50	23.607	1.55	9.043	1.50	49.063	1.55	16.289	1.50	3.711
1.60	34.994	1.65	10.915	1.60	86.966	1.65	22.642	1.60	4.086
1.65	41.274	1.75	14.115	1.70	206.595	1.75	31.378	1.70	4.828
1.70	62.231	1.85	17.335	1.75	358.480	1.85	50.317	1.80	5.118
1.75	79.598	1.95	24.387	1.80	715.101	1.95	84.660	1.90	5.776
1.80	107.176	2.05	30.570			2.05	145.682	2.00	6.676
1.85	155.993	2.15	39.033			2.15	348.336	2.10	7.840
1.87	199.287	2.25	71.349			2.20	559.500	2.20	8.934
1.89	231.121	2.35	98.745			2.25	1302.800	2.30	9.963
1.91	273.033	2.45	197.144					2.40	11.648
1.93	356.756	2.55	494.952					2.50	14.630
1.95	410.822	2.60	679.359					2.60	18.235
1.97	605.203							2.70	20.523
1.99	761.088							2.80	24.979
2.00	862.600							2.90	32.915
2.02	1205.100							3.00	41.148
								3.10	53.075
								3.20	78.899
								3.30	96.997
								3.40	153.862
								3.50	242.795
								3.55	342.361
					60			3.60	502.651
								3.65	733.398

Tabela B.6:  $E(N)$  estimado para distribuição  $U(0, 1)$

$U(0, 1)$					
$\lambda$	$E(N)$	$\lambda$	$E(N)$	$\lambda$	$E(N)$
0.15	1.152	1.15	4.644	2.15	54.601
0.20	1.214	1.20	4.900	2.20	73.848
0.25	1.310	1.25	5.260	2.25	82.705
0.30	1.380	1.30	5.686	2.27	105.530
0.35	1.423	1.35	6.898	2.29	118.562
0.40	1.521	1.40	7.249	2.31	131.436
0.45	1.611	1.45	8.095	2.33	139.988
0.50	1.696	1.50	8.342	2.35	157.897
0.55	1.875	1.55	9.793	2.37	183.215
0.60	1.996	1.60	10.607	2.39	212.336
0.65	2.345	1.65	12.550	2.41	241.399
0.70	2.257	1.70	14.482	2.43	286.491
0.75	2.429	1.75	14.704	2.45	361.078
0.80	2.637	1.80	19.090	2.47	392.659
0.85	2.751	1.85	20.513	2.49	534.019
0.90	3.090	1.90	24.230	2.51	615.289
0.95	3.310	1.95	28.176	2.53	814.978
1.00	3.580	2.00	32.199	2.55	949.358
1.05	3.901	2.05	34.584	2.57	1484.400
1.10	4.507	2.10	43.072		

Tabela B.7: Comprimento fixo  $d$ ; Lambda crítico obtido; Grau de polinômio usado para aproximar  $1/\log(E(N))$ ;  $SQ$ -soma de quadrados de resíduos obtidos usando aproximação polinomial

$d$	$\lambda_c$	Grau polinomial	$SQ$
0.3	4.6688	17	0.0736
0.5	2.8231	19	0.0585
0.7	2.0641	11	0.0415
1.0	1.4193	13	0.0049
1.2	1.1905	11	0.0065
1.4	1.0254	11	0.0007
1.6	0.9120	10	0.0006
1.8	0.8084	11	0.0000
2.0	0.7312	10	0.0000
2.5	0.5682	11	0.0066
3.0	0.4537	12	0.0012
3.5	0.3931	11	0.0003
4.0	0.3530	12	0.0009
4.5	0.3103	12	0.0002
5.0	0.2833	11	0.0003

Tabela B.8: Distribuições *Uniforme* e *Beta*; Lambda crítico obtido; Grau de polinômio usado para aproximar  $1/\log(E(N))$ ;  $SQ$ -soma de quadrados de resíduos obtidos usando aproximação polinomial

Distribuição	$\lambda_c^*$	$\lambda_c$	Grau polinomial	$SQ$
$U(0, 1)$	0.9282	2.6135	11	0.0924
$Beta(2, 1)$	0.7279	2.0888	11	0.0014
$Beta(2, 2)$	0.9545	2.6746	11	0.0110
$Beta(3, 1)$	0.6559	1.8597	13	0.0003
$Beta(3, 2)$	0.8114	2.3079	12	0.0013
$Beta(1, 2)$	1.3485	3.7981	11	0.0199

Tabela B.9: Resíduos obtidos ajustando modelos lineares entre  $\lambda_c$  e  $\lambda_c^*$ , com intercepto e sem intercepto; comprimento fixo  $d$

$d$	$\lambda_c = 2.8088 \cdot \lambda_c^* + 0.013$	$\lambda_c = 2.8246 \cdot \lambda_c^*$
0.3	-0.0255	-0.0389
0.5	0.0013	-0.0015
0.7	0.0448	0.0465
1.0	0.0019	0.0070
1.2	0.0072	0.0136
1.4	0.0093	0.0166
1.6	0.0213	0.0293
1.8	0.0152	0.0238
2.0	0.0160	0.0250
2.5	-0.0065	0.0033
3.0	-0.0274	-0.0171
3.5	-0.0211	-0.0104
4.0	-0.0111	-0.0001
4.5	-0.0148	-0.0035
5.0	-0.0106	0.0008

Tabela B.10: Resíduos obtidos ajustando modelos lineares entre  $\lambda_c$  e  $\lambda_c^*$ , com intercepto e sem intercepto; distribuições *Uniforme* e *Beta*

Distribuição	$\lambda_c = 2.7745 \cdot \lambda_c^* + 0.0479$	$\lambda_c = 2.8243 \cdot \lambda_c^*$
$U(0, 1)$	-0.0096	-0.0080
$Beta(2, 1)$	0.0214	0.0330
$Beta(2, 2)$	-0.0215	-0.0212
$Beta(3, 1)$	-0.0079	0.0072
$Beta(3, 2)$	0.0088	0.0163
$Beta(1, 2)$	0.0089	-0.0105

# Apêndice C

## Programas

```
function [x]= procreta(lambda,a,b);
t = poissrnd(lambda*(b-a),1,1);
eps = 0.0005;
if t ~= 0
x = unifrnd(a-eps,b+eps,1,t);
else
x=[];
end;
*****

function [p11]=cabl(x);
p11 = zeros(size(x));
y = rand(size(x));
p11 = x+y;
*****

function [p11,p22]=chamadas(p1,p2);
p11 = cabl(p1);
expo = exprnd(1,size(p1));
p22 = p2 + expo;
*****

function [p,x,y]= poisproc(lambda,a);
if isempty(a)~= 1
coef = lambda*(a(3)-a(1))*(a(4)-a(2));
if coef ~= 0
p = poissrnd(coef,1,1);
if p>0
```

```

x = unifrnd(a(1),a(3),1,p);
y = unifrnd(a(2),a(4),1,p);
else
x=[]; y= [];
end;
else
p=0;x=[];y=[];
end;
else
p=0;x=[];y=[];
end;
*****
function [res] = mantesec(A,B);
if (isempty(A)~= 1) & (isempty(B)~= 1)
res=zeros(size(A(1,:)));
for i=1:length(A(1,:)),
pos=A(:,i);
upind=find((B(4,:) >= pos(2))& (B(4,:) <= pos(4)));
if isempty(upind)~= 1
leftind=find((B(1,upind) <= pos(3))& (B(3,upind) >= pos(1)) );
if isempty(leftind)~= 1
res(i)=1;
end;
end;
end;
else
res=[];
end;
*****
function desrect(calls,sign);
if isempty(calls)~= 1
n = length(calls(1,:));
x=calls(1,:);
y=calls(2,:);
x11=calls(3,:);
y11=calls(4,:);
for i = 1:n,
plot([x(i) x11(i)], [y(i) y11(i)],sign,[x(i) x11(i)],[y(i) y(i)],sign,[x(i) x11(i)], [y11(i) y11(i)],sign,[x11(

```

```

x11(i)], [y(i) y11(i)], sign);
hold on
end;
end;
*****
function [calls]= zerocall(lambda,a,b);
x=procreta(lambda,a,b);
d=1;
if isempty(x)~= 1
y=zeros(size(x));
[x11,y11]=chamadas(x,y);
calls=[x
y
x11
y11
zeros(size(x))];
else
calls=[];
end;
*****CONSTRUÇÃO DO CLÃ*****
function [calls]=construction(lambda,s1,s2);
d=1;
todos=zerocall(lambda,s1-d,s2);
helpind=find(todos(3,*)<s1 );
if isempty(helpind)~= 1
help= todos(:,helpind);
else
help=[];
end;
callind=find(todos(3,*)>=s1);
calls=todos(:,callind);

if isempty(calls)~= 1
x=calls(1,:);y=calls(2,:);a=calls(3,:);b=calls(4,:);
firstNumber=length(calls(1,:));
k=1;
oblast=[];
reserva = help;

```

```

numberNew=1;

while (numberNew ~= 0)
numberNew=0;
xpot=x-1;
left = min(xpot);
right = max(a);
down = min(y);
up = max(b);
oblpot=[left down right up]';
if isempty(oblast)==1
obl=oblpot;
oldobl=obl;
else
oldobl=oblast(:,k-1);
obl=zeros(4,1);
obl(1)= min(left,oldobl(1));
obl(2)= min(down, oldobl(2));
obl(3)= max(right, oldobl(3));
obl(4)= max(up, oldobl(4));
end;
oblast = [oblast obl];
if k ~= 1
part1 = [obl(1) obl(2) oldobl(1) obl(4)];
part2 = [oldobl(1) oldobl(4) oldobl(3) obl(4)];
part3 = [oldobl(3) obl(2) obl(3) obl(4)];
[num1,novax1,novay1]=poisproc(lambda,part1);
[num2,novax2,novay2]=poisproc(lambda,part2);
[num3,novax3,novay3]=poisproc(lambda,part3);
pointnum= num1+num2+num3;
novax=[novax1 novax2 novax3];
novay=[novay1 novay2 novay3];
else
[pointnum, novax, novay]=poisproc(lambda,obl);
end;
[novaa,novab] =chamadas(novax, novay);
nova=[novax
novay

```

```

novaa
novab
k.*ones(size(novab));
if isempty(novaa)~= 1
end;
numGen=calls(5,:);
I = find(numGen==k-1);
oldreserva = reserva;
novareserva=[];
numberNew=0;
j = I(1);
goodind=find(novab(:)>= min(calls(4,I)));
if isempty(goodind)~= 1
forcheck=novaa(:,goodind);
res1=mantesec(forcheck,calls(:,I));
if isempty(res1)~= 1
calind=find(res1==1);
nresind=find(res1==0);
calls=[calls forcheck(:,calind)];
numberNew=numberNew+length(calind);
novareserva=forcheck(:,nresind);
end;
end;
if isempty(oldreserva)~= 1
bomind=find(oldreserva(4,:)>= min(calls(4,I)));
if isempty(bomind)~= 1
paracheck=oldreserva(:,bomind);
paracheck(5,:)=k.*ones(size(paracheck(1,:)));
res2=mantesec(paracheck,calls(:,I));
chamind=find(res2==1);
nadaind=find(res2==0);
calls=[calls paracheck(:,chamind)];
numberNew=numberNew+length(chamind);
oldreserva=paracheck(:,nadaind);
else
oldreserva=[];
end;
end;

```

```

reserva= [oldreserva novareserva];
gen2 = calls(:,(I(length(I))+1):length(calls(3,:)));
x = gen2(1,:);
y = gen2(2,:);
a = gen2(3,:);
b = gen2(4,:);
k = k+1;
xmin=min(x);
amax=max(a);
if xmin-d<s1
nlefthelp=zerocall(lambda,xmin-d,s1);
s1=xmin-d;
else
nlefthelp=[];
end;
if amax > s2
nrighthelp=zerocall(lambda,s2,amax);
s2=amax;
else
nrighthelp=[];
end;
reserva=[reserva nlefthelp nrighthelp];
end;
desrect(calls,'b-');
end;hold off;
*****LIMPEZA*****
function [amostra]=thinning(calls,capacity);
if (isempty(calls)~=1)&(capacity > 0)
n = length(calls(4,:));
[vektor, I] =sort(-1*calls(4,:));
ordered=calls(:,I);
ordered(5,:)=zeros(size(ordered(5,:)));
c=capacity;
for i=1:(n-1),
if ordered(5,i)~=(-1)
for j=(i+1):n,
if ordered(5,j)~=(-1)
if antesec(ordered(:,i),ordered(:,j))== 1

```

```

if c==1
ordered(5,j)=-1;
else
area=intersec(ordered(:,i),ordered(:,j));
cap=2;
for k=(j+1):n
if ordered(5,k)~=(-1)
if antesec(area,ordered(:,k))==1
cap=cap+1;
if cap >c
ordered(5,k)=-1;
end;
end; end; end; end; end; end; end; end; end;
O=find(ordered(5,:)==-1);
deleted=ordered(:,O);
K=find(ordered(5,:)~=-1);
kept=ordered(:,K);
desrect(deleted,'g-');
desrect(kept,'r-');
A = find(kept(2,:)==0);
amostra= kept(:,A);
else
amostra=[];
end;
hold off;
*****

```

# Bibliografia

- [1] Daley, D. J., Vere-Jones, D. (1988). *An introduction to the theory of point processes*, Springer-Verlag, New York.
- [2] Fernández, R., Ferrari, P. A., Garcia, N. L. (2001). Loss network representation of Pierls contours, *Annals of Probability* **29**(2): 902-937.
- [3] Fernández, R., Ferrari, P. A., Garcia, N. L. (2000). Perfect simulation for interacting point processes, loss networks and Ising models, *Preprint. Los Alamos archive: Mathematics. abstract math.PR/9911162t*.
- [4] Ferrari P. A., Garcia, N. L. (1998). One-dimensional loss networks and conditioned  $M/G/\infty$  queues. *J. Appl. Probab.* **35**,4: 963-975.
- [5] Garcia, N. L. (2000). Perfect simulation of spatial processes, *Resenhas IME-USP* **4**(3): 283-325.
- [6] Harris, T.E. (1989). *The theory of branching processes*. Dover publications, INC, New York.
- [7] Hall, P. (1988). *Introduction to the theory of coverage processes*. John Wiley & Sons Inc., New York
- [8] Kelly, F. P. (1991). Loss networks, *Ann. Appl. Probab.* **1**(3): 319-378.
- [9] Mode, C. J. (1971). *Multitype branching processes*, American Elsevier, New York.
- [10] Stoyan, D., Kendall, D.G., Mecke, J. (1987). *Stochastic Geometry and Its Applications*. Akademie-Verlag Berlin, GDR/John Wiley & Sons Ltd.