IDENTIFICAÇÃO E ANÁLISE DE COMPORTAMENTOS ANÔMALOS

Anderson Fernandes Pereira dos Santos

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DE FORMAÇÃO DE RECURSOS HUMANOS DO LABORATÓRIO NACIONAL DE COMPUTAÇÃO CIENTÍFICA COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM MODELAGEM COMPUTACIONAL.

Aprovada por:

_____

Prof. Augusto César Noronha Rodrigues Galeão, D.Sc.

_____

Prof. Ronaldo Moreira Salles, Ph.D.

_____

Prof. Antonio José da Silva Neto, Ph.D.

_____

Prof. Fernando Antonio Campos Gomide, Ph.D.

_____

Prof. Karla Tereza Figueiredo Leite, D.Sc.

PETRÓPOLIS, RJ – BRASIL
AGOSTO 2009

SANTOS, ANDERSON FERNANDES PEREIRA DOS

S237i Identificação e Análise de Comportamentos Anômalos

Petrópolis, 2009, xvi, 106 p., 29 cm. (MCT/LNCC, D.Sc., Modelagem Computacional, 2009)

Orientador: Renato Simões Silva

Tese – Laboratório Nacional de Computação Científica, LNCC

1. Sistemas de Segurança.2. Segurança da Informação, Negação de Serviço, Negação de Serviço Distribuído. I. MCT/LNCC II. Título (série)

CDD - 005.8

“Man is still the most extraordinary computer of all”

(John F Kennedy)

Aos meus dois filhos João Pedro e Antonio Pedro, que sirva de inspiração para todos os desafios de suas vidas, e que entendam que nada é impossível.

Agradecimentos

Nenhuma conquista é conseguida sozinha, e vou tentar aqui agradecer a todos àqueles que me ajudaram neste estudo. Em primeiro lugar, agradeço à minha mãe e ao meu pai (*in memoriam*) por terem me ensinado os valores éticos e morais que trago comigo.

Agradeço aos meus professores do Instituto Militar de Engenharia que me ensinaram não só a parte técnica, mas a "fazer as coisas acontecerem", e que hoje são meus amigos de trabalho, mas que serão sempre "*meus mestres*", em particular ao Prof^{o} Edmundo Lopes Cecílio. Agradeço, também, à Prof^{a} Ana Maria Moura de Carvalho, que muito me incentivou nesta pesquisa, junto com o Prof^{o} Carlos Teobaldo Gutiérrez Vidalon (*in memoriam*) da Universidade de São Paulo.

Agradeço aos meus professores do Laboratório Nacional de Computação Científica que permitiram que este estudo fosse possível, em particular ao meu orientador, Renato Simões Silva que acreditou em nossa missão, desde o primeiro dia, sem duvidar um momento sequer.

Agradeço ao Exército Brasileiro por ter me apoiado durante todo o período deste trabalho.

E finalmente agradeço, muito, à Deus, por permitir que eu conseguisse chegar aqui.

Resumo da Tese apresentada ao MCT/LNCC como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D. Sc.)

Identificação e Análise de Comportamentos Anômalos

Anderson Fernandes Pereira dos Santos

Ago/2009

Orientador: Renato Simões Silva

Modelagem Computacional

Um processo pode ser monitorado de várias maneiras diferentes. A mais usual é a medição de alguma característica do processo. O acompanhamento desta característica, agora representada através de uma variável de controle, permite a identificação de um comportamento não usual, também dito anômalo.

Existem diversas medidas que podem ser realizadas sobre um fenômeno. Estas medidas podem sofrer alterações durante todo o ciclo de vida do fenômeno. Destas, algumas podem representar o fenômeno em seus vários estados de transição, permitindo assim, que seja possível acompanhar o fenômeno pela monitoração desta medida. Desta forma, se o fenômeno for interpretado como um processo, esta medida pode ser interpretada como um produto deste processo.

Na área de controle estatístico da qualidade há inúmeras ferramentas que permitem o acompanhamento de um processo, em especial de um processo de fabricação. Dentre estas ferramentas há o gráfico de controle que, a partir da interpretação da normalidade de um processo, pode indicar quando um processo torna-se não controlável.

A partir de um gráfico de controle para processos não normais e de lógica fuzzy, foi desenvolvido o Algoritmo de Detecção de Anomalias com Janelas Adaptativas. Este algoritmo foi aplicado nas áreas de Segurança da Informação e Vazão em uma rede de distribuição.

No primeiro caso foi utilizado na identificação de ataques de negação de serviço. Ataques deste tipo são caracterizados pelo aumento do número de solicitações a um servidor. O intervalo de tempo entre os pacotes foi a variável selecionada para monitorar o processo.

No segundo caso foi utilizado na identificação de vazamentos em redes de distribuição de água. A variável de controle utilizada neste caso foi da vazão obtida em um ponto da rede.

Abstract of Thesis presented to MCT/LNCC as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

Identification and Analysis of Anomalous Behavior

Anderson Fernandes Pereira dos Santos

Ago/2009

Advisor: Renato Simões Silva

Computational Modeling

A process can be monitored in many different ways. The most usual is the measurement of some characteristic of the process. The attendance of this characteristic, now acted through a control variable, it allows the identification of an unusual or anomalous behavior.

There are many measures that can be made about a phenomenon. These measures may change during all phenomenons' life cycle. These can represent the phenomenon in the various states, allowing that, can be possible to follow the phenomenon by monitoring these measures. In that way, if the phenomenon can be interpreted as a process, these measures can be interpreted as a process' product.

In Statistical Quality Control there are various tools that permit the monitoring of the manufactured process. Among these tools, there is the control chart that, from the process' normality assumption, can indicate when a process becomes out of control.

From the control chart for a non normal process and fuzzy logic, it was developed the Algorithm of Anomalous Detecting with Adaptive Window. This algorithm was applied in Information Security and Yield.

In the first case, it was used in detection of denial of service attacks. This kind of attack is characterized by the increase of the number of the solicitations to a server. In that case, the inter-arrive time was the variable used to monitoring the process.

In the second case it was used in the identification the leak in hydraulic system. The control variable used was the flow that was measured in a certain point of the system.

Índice

Índice de Figuras

Índice de Tabelas

Capítulo 1

O início da Internet aconteceu na década de 60, porém foram nos últimos 15 anos que o seu uso alcançou proporções globais. Atualmente a Internet não apenas serve de canal de comunicação entre diversas entidades, como também abriga serviços variados que são fornecidos para todas as pessoas, empresas e governos do mundo. Uma grande parte dos serviços disponibilizados é gratuita e tem se tornando indispensável.

Por outro lado, essa abrangência e dependência global transformam a Internet como um alvo de ataques de qualquer magnitude. Nos últimos anos o número de ataques aumentou significativamente.

Dos ataques existentes, os mais populares são conhecidos como ataques de negação de serviço distribuídos (*DDoS*) (TAROUCO 2003) e são caracterizados por realizarem solicitações maciças de serviços. Estes ataques são variantes dos ataques de negação de serviço (*DoS*) (HUSSAIN 2003, MIRKOVIC 2004), que são caracterizados pela exploração das vulnerabilidades existentes nas implementações dos protocolos e serviços.

1.1. Motivação

Os primeiros ataques com esse padrão foram relatados há pouco menos de dez anos (HOUSEHOLDER 2001, PAUL 2001). De acordo com o site ShadowServer[1], de 2007 para 2008 o aumento de ataques *DDoS* aumentou aproximadamente cinco vezes mais (pouco mais de 190.000 ataques, Figura 1.1).

[1] http://www.shadowserver.org

Em 2009, a contagem de ataques já ultrapassou 120.000, Figura 1.2. Além disso, os ataques estão objetivando sites mais famosos, e teoricamente, mais bem protegidos. Nos últimos meses, mais de vinte sites foram vítimas de um ataque maciço. Dentre os sites atacados encontram-se sites como o da Casa Branca[2], da Presidência da República da Coréia do Sul[3], da NASDAQ[4], da NYSE[5]. Pouco mais de duas semanas depois, os sites de redes de relacionamento, como o Facebook[6] e Twitter[7], também foram atacados.

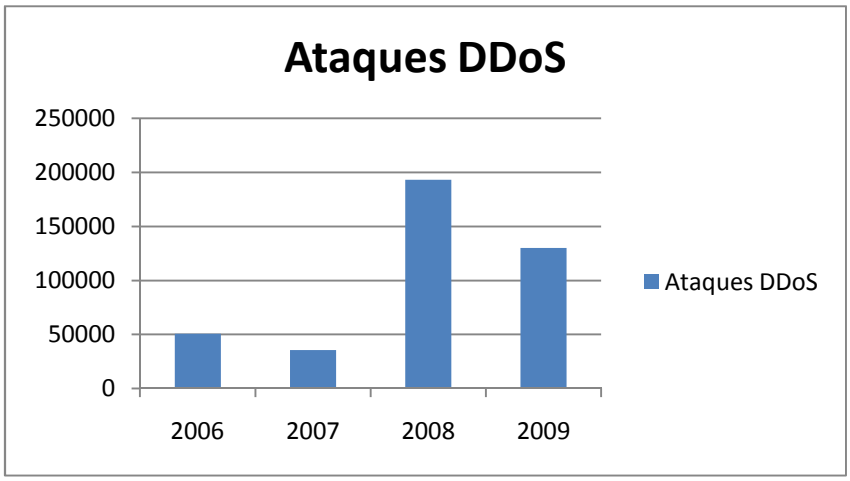

Figura 1.1: Ataques de *DDoS* identificados por ShadowServer[8] nos últimos quatro anos;

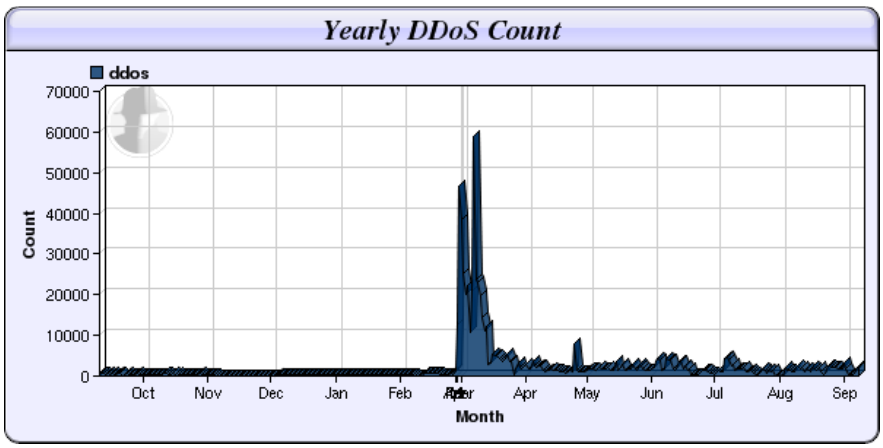

Figura 1.2: Ataques DDoS identificados por ShadowServer[8] em 2009;

[2] http://www.whitehouse.gov
[3] http://www.president.go.kr
[4] http://www.nasdaq.com
[5] http://www.nyse.com
[6] http://www.facebook.com
[7] http://www.twitter.com
[8] http://www.shadowserver.org

Esses ataques, em geral, possuem como características comuns o anonimato do atacante, a pesquisa de vulnerabilidades em *softwares*, o desenvolvimento de vírus/*worms*, o planejamento da infecção em massa e a precisa coordenação do ataque propriamente dita. Estes fatos refletem as diversas fases existentes em um ataque e o grau de planejamento do ataque, o que reflete certo grau de sofisticação dos atacantes.

Desde os primeiros relatos de ataques de negação de serviço, este assunto tem sido pesquisado e bons resultados foram obtidos[9]. A partir destas pesquisas, novos mecanismos foram desenvolvidos e implementados. Entretanto, pequenas variações dos ataques de negação de serviço permitem torná-lo imune aos mecanismos de defesa desenvolvidos. Desta forma, as construções de mecanismos de defesa tornam-se realmente eficientes quando estão focados na forma de operação de um ataque, mas não em um ataque específico.

1.2. Objetivo

Desta forma, o principal objetivo deste estudo é propor um algoritmo para identificar ataques do tipo DDoS através da mudança do comportamento de um rede de computadores, em condições reais. Três prerrogativas básicas foram definidas para esta pesquisa:

- Não conhecimento se a rede está ou não sob ataque;
- Estudo mais focado em uma categoria de ataques, conhecidos como ataques de negação de serviço distribuído (*DDoS*), e não em ataques específicos; e
- Uso de técnicas mais simples que permitam identificar os ataques em poucas interações.

Para desenvolver este algoritmo, Algoritmo de Detecção de Anomalias com Janelas Adaptativas, foram utilizadas técnicas baseadas em gráficos de controle não

[9] As principais pesquisas realizadas nos últimos anos serão descritas no capítulo 4 – Algoritmo de Detecção de Anomalias com Janelas Adaptativas.

normal (DUCLOS 1997), para monitorar o comportamento da rede, e lógica fuzzy (SANDRI 1999), para representar o conhecimento do especialista e o comportamento que o algoritmo deverá ter ao selecionar seu domínio de aplicação.

1.3. Organização da Tese

Esta tese está organizada da seguinte forma, o capítulo 2 apresenta uma introdução à teoria de gráficos de controle. O capítulo 3 realiza uma introdução à teoria Fuzzy. Com base nestas duas teorias, no capítulo 4 é apresentado o Algoritmo de Detecção de Anomalias com Janelas Adaptativas. O capítulo 5 exibe os resultados da aplicação deste Algoritmo nas áreas de segurança da informação e vazão de uma rede de distribuição. O capítulo 6 possui a conclusão da aplicabilidade do algoritmo nestas áreas. Os anexos A, B e C possuem, respectivamente, o código fonte do algoritmo desenvolvido em MatLab, a descrição dos ataques de negação de serviço e as figuras do capítulo 5.

Capítulo 2

Gráficos de Controle para Processos Não Normais

2.1. Introdução

O processo de fabricação de produtos sofreu um grande impacto com a revolução industrial. A conseqüente substituição do trabalho humano por máquinas iniciou uma preocupação, cada vez maior, com a qualidade dos produtos gerados e com o próprio processo fabril.

Garantir que os produtos oriundos de um mesmo processo, com os mesmos equipamentos tivessem a mesma qualidade não era uma tarefa fácil e necessitava de constante monitoramento, quer seja dos operadores, quer seja do maquinário envolvido.

No início do século passado, estas preocupações foram traduzidas em procedimentos e técnicas que são usadas até os dias de hoje, não só no processo de fabricação, mas em todo o mundo corporativo.

2.2. Histórico

A área de controle estatístico da qualidade trata do monitoramento das características de um processo através da manutenção de padrões pré-estabelecidos. Estes padrões são responsáveis por assegurar a qualidade de produtos e serviços que influenciam diretamente a escolha do consumidor.

Para este monitoramento existem sete ferramentas que podem auxiliar o controle (REIS 2001), são elas:

a) Histograma: é a representação gráfica da distribuição de freqüências dos valores de uma variável de processo (Figura 2.1).

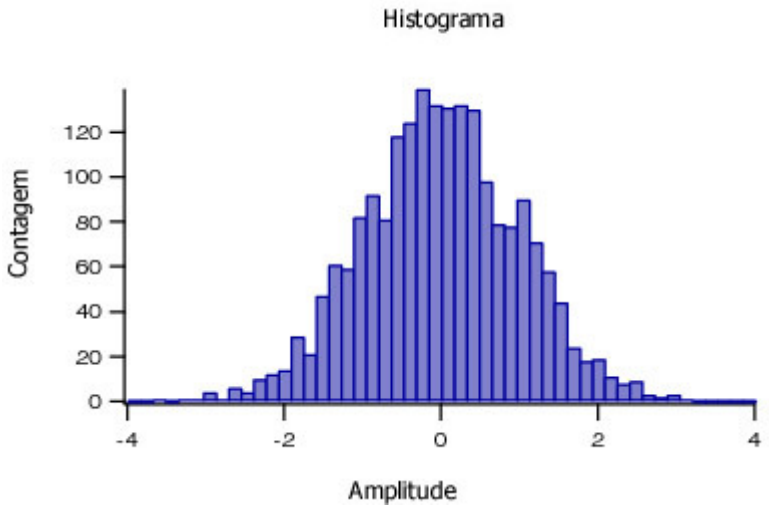

Figura 2. 1: Exemplo de histograma.

b) Folha de Verificação: são planilhas/tabelas usadas nas coletas de dados.

c) Gráfico de Pareto: é a representação gráfica ordenada de frequências das ocorrências das amostras, do maior para o menor valor (Figura 2.2).

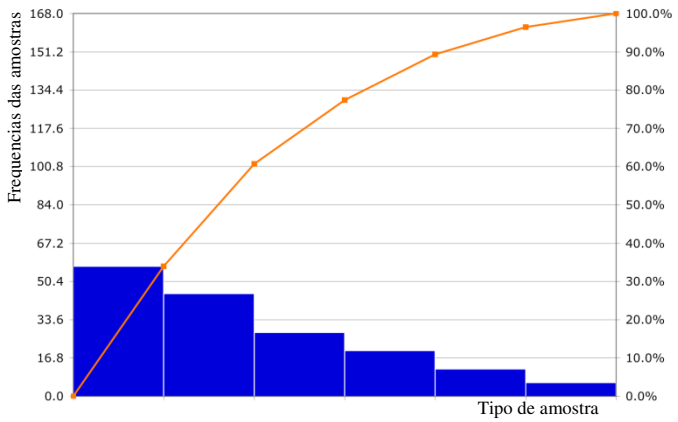

Figura 2. 2: Exemplo de gráfico de Pareto.

d) Diagrama Causa-e-Efeito: também conhecido como diagrama “Espinha de Peixe” ou Ishikawa mostra um sistema disposto hierarquicamente de acordo com as causas de um certo problema (Figura 2.3).

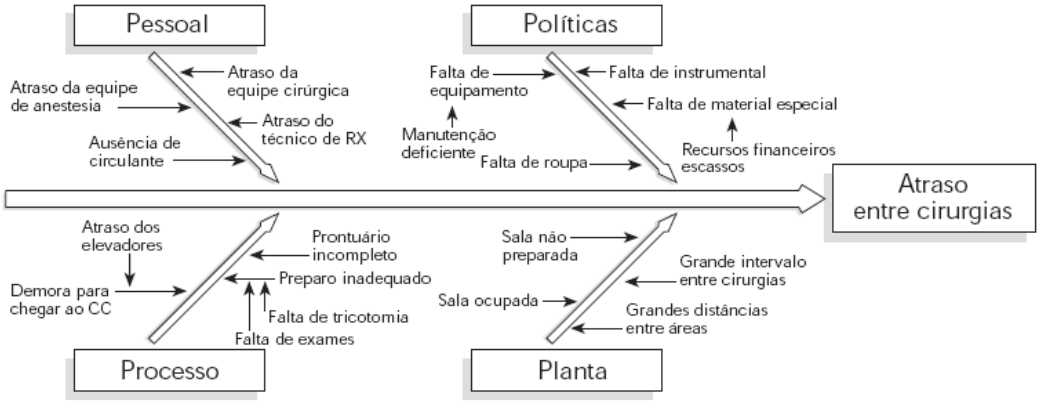

Figura 2. 3: Exemplo de diagrama de causa-e-efeito.

e) Diagrama de Concentração de Defeito: é a separação da amostra original, agrupando aqueles fortemente relacionados.

f) Diagrama de Dispersão: é a representação gráfica dos valores de duas variáveis quantitativas medidas em cada elemento do conjunto de dados. Este tipo de diagrama é usado, principalmente, para visualizar a relação/associação entre duas variáveis (Figura 2.4).

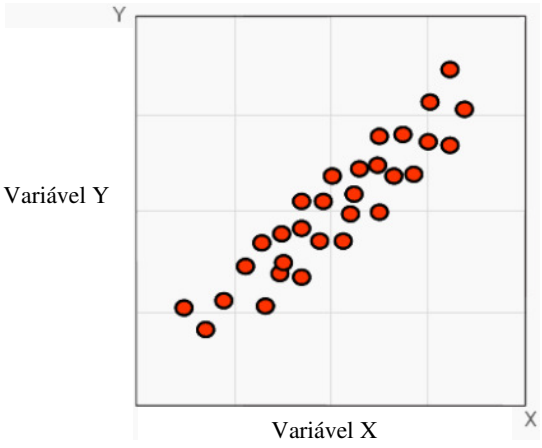

Figura 2. 4: Exemplo de gráfico de dispersão.

g) Gráficos de Controle: são gráficos que permitem verificar se as amostras seguem um comportamento de acordo com uma distribuição gaussiana (Figura 2.5).

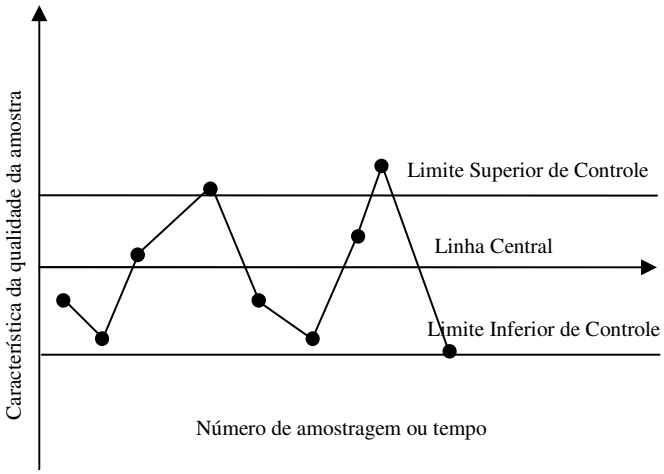

Figura 2. 5: Exemplo de gráfico de controle.

Um gráfico de controle é considerado um instrumento para o monitoramento da variabilidade e para a avaliação da estabilidade de um processo, ou ainda segundo DUNCAN (1986), uma ferramenta estatística que trata de processos repetitivos. Esta foi utilizada inicialmente por Walter A. Shewhart na década de 20, nos laboratórios da *Bell Telephone*, e posteriormente na publicação de diversos artigos e livros, como o *Economic Control of Quality of Manufactured Product,* de 1931, (MONTGOMERY 1999).

A adoção destes gráficos de controle pelo governo norte americano começou em 1939, no início da Segunda Guerra Mundial, quando o interesse na melhoria da produção industrial aumentou significativamente. Em decorrência, outras nações e empresas se interessaram por estas ferramentas, como foi o caso do Japão na década de 60, quando Deming a implantou na indústria manufatureira japonesa, obtendo resultados significativos.

Há dois tipos de gráficos de controle: os de controle para variáveis e os de controle para atributo. Quando são usados para medir dados que podem sofrer variação contínua, são ditos serem Gráficos de Controle para Variáveis. Quando a grandeza

utilizada no gráfico de controle pode apenas ser contada, ou classificada, o gráfico é dito ser Gráfico de Controle para Atributos.

2.3. Gráficos de Controle

Em geral todo processo possui duas variabilidades. A primeira é uma variação intrínseca e que não pode ser alterada. Esta variabilidade, dita **comum**, é tida como normal e aceitável e pode ser interpretada como um fator aleatório que está presente na maioria dos fenômenos naturais.

A segunda, chamada de **especial**, é resultado de alguma perturbação interna ou externa ao sistema e pode ser entendida e corrigida, através da eliminação desta interferência no sistema. Os gráficos de controle permitem a distinção entre estas duas causas e também determinar **se** e **quando** uma ação corretiva pode ser aplicada. Porém esta ferramenta não permite definir **como** esta ação corretiva pode ser realizada, uma vez que não possui informações suficientes do ambiente físico relacionado. Desta forma, os processos que estão sujeitos apenas às causas comuns são ditos estarem **controlados** e os sujeitos às causas especiais são ditos estarem **descontrolados**.

Processos controlados, portanto, possuem alguma variabilidade, presumidamente limitada, sobre um valor fixo de tendência central. Estes fatos podem ser interpretados geometricamente como sendo uma linha central e uma região delimitada por dois limites, uma superior, denominada Limite Superior de Controle – L.S.C. e uma inferior, o Limite Inferior de Controle – L.I.C., onde uma dada característica da amostra está contida, conforme exibido na Figura 2.5.

Desde o início do uso desta ferramenta na década de 20, por simplicidade, é feita uma aproximação da distribuição dos processos reais com a distribuição normal. Portanto, o comportamento de variabilidade sobre um valor de tendência central é modelado através de distribuições normais, e desta forma, modela-se (MONTGOMERY 1999) este gráfico considerando que a linha central é o estimador de localização (média (μ)) de valores de uma determinada característica da amostra considerada (X), com

$$\mu = \bar{X} = E(X) = \sum_x xf(x) \tag{2.1}$$

onde *f(x)* é a função densidade de probabilidade do processo, e os limites são múltiplos do estimador de dispersão (desvio-padrão (σ)) da mesma amostra considerada, onde,

$$\sigma^2 = V(X) = \sum_x (x-\mu)^2 f(x) \tag{2.2}$$

Os valores dos Limites Superior e Inferior de Controle são proporcionais ao desvio padrão da amostra (Equações 2.3 e 2.4), e normalmente, são usados três limites superiores e inferiores, conforme Figura 2.6. Busca-se, com isso, assegurar que a característica da amostra que está sendo monitorada possua um padrão de qualidade definido.

$$L.I.C = \mu - k.\sigma \tag{2.3}$$

$$L.S.C. = \mu + k.\sigma \tag{2.4}$$

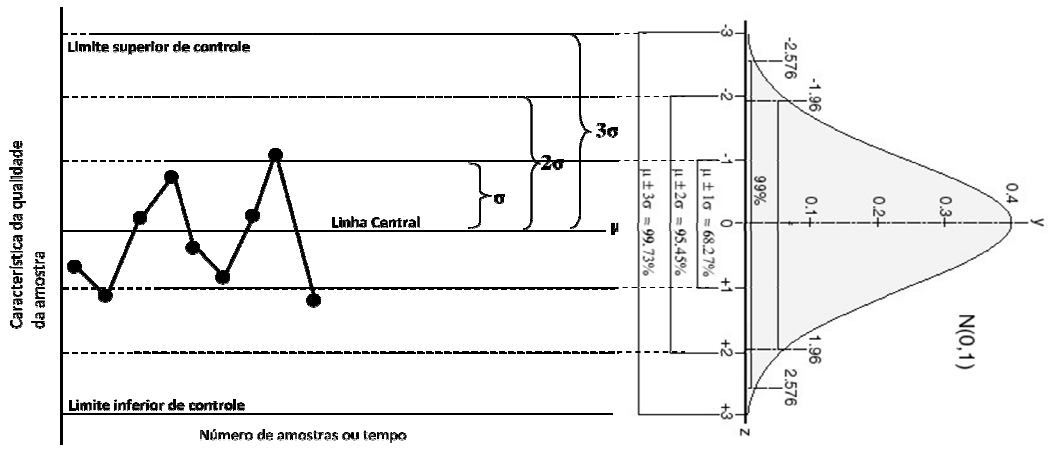

Figura 2. 6: Gráfico de Controle de Shewhart;

Alternativamente, os gráficos de controle podem ser modelados através de testes de hipóteses, onde é pressuposto que o processo esteja em um estado de controle estatístico, ou seja,

$$H_0: \mu = \mu_0$$
$$H_1: \mu \neq \mu_0$$

sendo conhecido o desvio padrão σ_0 da amostra.

Os valores dos limites superior e inferior de controle foram obtidos por Shewhart de modo experimental. Ele observou o processo de fabricação dos produtos na *Bell Telephones* e, além de estipular o uso dos limites em três desvios padrões, também estabeleceu critérios em que o processo de fabricação estaria fora de controle. Estes critérios podem ser resumidos através da Tabela 2.1.

Tabela 2. 1: Regras de detecção de processo fora de controle[10]

Item	Critério
1	Variação Extrema: um ou mais pontos estão fora dos limites do gráfico.
2	Desvio: dois de três pontos consecutivos estão entre os limites de 2σ e 3σ, de um mesmo lado da linha central.
3	Desvio: quatro de cinco pontos consecutivos estão entre os limites de 1σ e 2σ do mesmo lado da linha central.
4	Tendência: nove pontos consecutivos estão do mesmo lado da linha central.
5	Ciclos: são arranjos de pontos que se repetem, mostrando valores máximos e mínimos periódicos.
6	Falta de variabilidade: acontece quando os pontos permanecem próximos à linha central.
7	Variabilidade Excessiva: grande variação da distribuição dos pontos pelo gráfico.

Um gráfico de controle não possui informações suficientes para identificar a razão do descontrole de um processo, porém, as experiências acumuladas pela indústria

[10] *Statistical Engineering Handbook* – http://itl.nist.gov/div898/handbook/

permitiram que algumas causas pudessem ser identificadas. Desta forma, as variações extremas, identificadas nos gráficos de controle, são normalmente causadas quando há erros de cálculos ou medições, ou mesmo quando são realizados ajustes durante o procedimento de medição; os desvios são encontrados quando há mudança de algum material/insumo ou mesmo alteração do operador/inspetor; a tendência é normalmente causada por desgaste de ferramentas/aparelhos de medição; os ciclos são normalmente ocasionados por defeitos sazonais ou ainda pela troca de operadores; a falta de variabilidade normalmente ocorre quando há medições tendenciosas ou realizadas por equipamentos não apropriados, e a variabilidade excessiva que pode ser ocasionada por diferentes problemas (equipamento-material-humano) no decorrer do processo.

A freqüência de coleta de dados esperada entre os desvios, onde podem ocorrer os erros acima descritos, é determinada através do fator *ARL* (*average run length*). Este parâmetro pode ser calculado através da Equação 2.5, como função da probabilidade (α) de existência de pontos inferiores ao LIC e superiores ao LSC.

$$ARL = \frac{1}{\alpha} \tag{2.5}$$

Para a distribuição normal, α=0,0027, ou ainda ARL= 370 amostras.

2.4. Outros Gráficos de Controle

Na literatura, (MONTGOMERY 1999), podem ser encontrados outros tipos de gráficos de controle, como é o caso do *EWMA* (média móvel moderada exponencialmente) e da soma acumulada (*CUSUM* – somas acumuladas) que são utilizados no monitoramento de processos cujas observações podem ser descritas por um modelo auto-regressivo de primeira ordem, também conhecido como processo de Markov (DUNCAN 1986). Estes modelos foram desenvolvidos visando detectar pequenos desvios que são imperceptíveis no gráfico de Shewhart.

2.4.1. Gráfico de Controle EWMA

O gráfico de controle da média móvel exponencialmente ponderada é caracterizada pelo acúmulo da influência das amostras passadas, através da constante λ.

$$z_i = \lambda . \bar{x}_i + (1 - \lambda) . z_{i-1} \tag{2.6}$$

onde $0 \leq \lambda \leq 1$, z_i representa o valor acumulado e z_{i-1} representa a influência das amostras passadas. O valor inicial z_0 é considerado o alvo do processo, ou seja, $z_0 = \bar{x}$.

O processo é considerado fora de controle quando os valores amostrados encontram-se fora dos limites (L.I.C. e L.S.C.). O processo sob controle apresenta amostras com variação em torno da média. Pequenos valores de λ permitem que sejam detectadas pequenas variações do processo. O gráfico de controle de Shewhart é um caso particular do gráfico de controle EWMA, quando $\lambda = 1$.

2.4.2. Gráfico de Controle das Somas Acumuladas (*CUSUM*)

Proposto inicialmente por E. S. Page em 1954, os Gráficos de Controle de Somas Acumuladas são caracterizados por armazenarem informações incorporadas às suas estatísticas através de somas acumuladas, (ALVES 2003b).

A partir de um valor nominal μ_0, calcula-se a soma acumulada de acordo com a Equação 2.7, ou em sua forma normalizada, Equação 2.8.

$$C_i = \sum_{j=1}^{i} \left(X_j - \mu_0 \right) = (X_i - \mu_0) + C_{i-1}, i \geq 1 \tag{2.7}$$

$$C_i = \sum_{j=1}^{i} \frac{(X_j - \mu_0)}{\sigma} = \frac{(X_i - \mu_0)\sqrt{n}}{\sigma} + C_{i-1}, i \geq 1 \quad (2.8)$$

onde σ é o desvio padrão da amostra considerada e X_j representa os valores das amostras. Estatisticamente as Equações 2.7 e 2.8 representam o mesmo fenômeno, com a vantagem da Equação 2.8 ter média zero, e com isto, qualquer variação positiva ou negativa indica a existência de tendência.

2.5. Processos Não Normais

A principal razão de existirem processos não normais está diretamente relacionada com a natureza do processo. Há diversos fenômenos, (MONTGOMERY 1999), cujas distribuições são conhecidamente não normais[11].

Como exemplos de processos não normais, pode-se citar o caso da quantidade de ligações que chegam numa central telefônica (Poisson), a probabilidade de um bit, que está trafegando em um canal de comunicação, apresentar um erro (binomial), o tempo entre requisições de um servidor TELNET (exponencial), injeção de plástico (DUCLOS 2005), a vazão de rios devido a dependência das estações das chuvas, dentre outros.

A aplicação de técnicas estatísticas, inicialmente desenvolvidas de acordo com a distribuição normal, em processos que não possuem aderência a esta distribuição podem gerar erros na determinação da variabilidade do processo, (MORAES 2006). Técnicas de normalização de dados, nestes casos, devem ser aplicados, como, as transformações de Box-Cox e Yeo-Johnson.

[11] Há processos (injeção de plástico) que, mesmo formado por subprocessos normais, devido a características de cada um destes, resultam em um processo não normal. A adoção da normalidade, em diversos aspectos, possui a intenção de simplificar a modelagem e a aplicação de ferramentas estatísticas (DUCLOS 2005).

2.6. Generalização dos Gráficos de Controle para Processos Não Normais

Quando os processos são normalmente distribuídos, os valores, por exemplo, de média, mediana e moda coincidem. Desta forma, a média é naturalmente escolhida como estimador de localização do processo. Caso o processo não seja normalmente distribuído, esta coincidência de valores não ocorre, tornando-se necessário selecionar qual será a variável que melhor representará a distribuição.

O gráfico de controle desenvolvido por DUCLOS (1997) é utilizado, principalmente, para processos não normais onde esta coincidência não existe. Desta forma, torna-se necessária a escolha das variáveis que substituirão a média e o desvio padrão no caso clássico. Estas medidas são denominadas de estimadores de localização e dispersão, respectivamente.

2.6.1. Determinação dos Estimadores de Localização e Dispersão

Uma das formas disponíveis para a determinação do estimador de localização é através do critério de TAGUCHI (TOLEDO 1999). Este critério realiza o cálculo através da minimização da perda de qualidade do estimador de localização $\overline{L}$ (variância mínima).

Este método, originado da área de engenharia de produção, objetiva aumentar a qualidade dos produtos pela diminuição de fatores, conhecidos como "ruídos", originados do ambiente da linha de produção. A primeira etapa deste processo (Equação 2.9) consiste na determinação destes fatores e posterior formulação matemática. Com essa formulação, minimizam-se estes fatores, resultando em um aumento da qualidade do produto (Equação 2.10).

$$\overline{L} = K.E[(X-\mu)^2] \tag{2.9}$$

$$\frac{\partial \overline{L}}{\partial \mu} = K(2.E(X) - 2\mu) = 0 \Leftrightarrow \mu = E(X) \quad (2.10)$$

onde K é um fator multiplicativo positivo constante.

Na determinação do estimador de dispersão, outro critério foi utilizado. Neste caso, não ocorre a coincidência de valores encontrados, como no caso da média. O estimador de dispersão selecionado foi o desvio padrão, por ser a variável normalmente utilizada.

2.6.2. Cálculo dos Estimadores de Localização e Dispersão

Para a obtenção destes estimadores, faz-se necessário o cálculo de uma matriz de covariância (*matriz Ω*) e de momentos dos vetores unitários (*vetor α*). Este cálculo não pode ser realizado para processos não estáveis. Para assegurar que os coeficientes da matriz de covariância sejam representativos da distribuição da população, torna-se necessário que a estimação dos momentos seja realizada em processos estacionários.

No caso de processos estáveis, a estacionariedade do processo pode ser assumida em pequenas amostras, sobre o qual a estatística de interesse possa ser calculada. Por outro lado, uma vez que períodos de estabilidade são curtos, torna-se difícil coletar dados suficientes para o cálculo dos estimadores. A idéia, portanto, consiste em utilizar métodos de *bootstrap* para garantir a convergência dos cálculos da matriz de covariância e do vetor de momentos para o universo amostral.

Métodos de *Bootstrap* (MOORE 2005) são procedimentos estatísticos que modelam uma população pela amostragem com reposição de um conjunto amostral. Propostos por Efron no final da década 70, a abordagem realizada por estes métodos é a utilização da amostragem através de Monte Carlo para gerar uma estimação empírica da distribuição amostral de um conjunto. Portanto, se um parâmetro pode ser expresso como um funcional de uma distribuição desconhecida, então o estimador através deste método é o mesmo funcional da função de distribuição empírica.

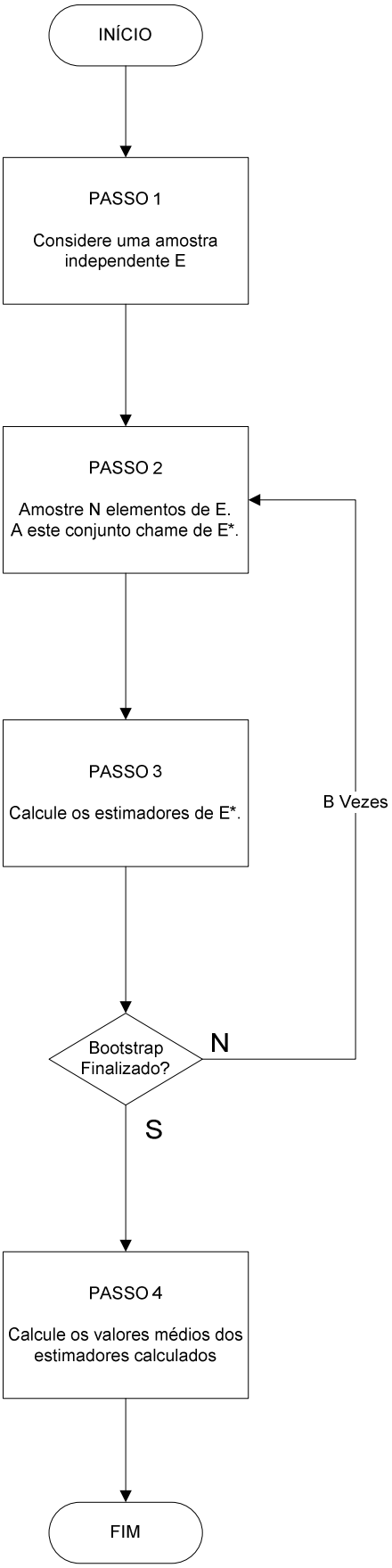

Figura 2. 7: Diagama de blocos do algoritmo de Duclos;

A aleatoriedade na amostragem está diretamente relacionada com a qualidade dos cálculos. Para o cálculo dos estimadores por mínimos quadrados, a repetição excessiva de alguns pontos gera instabilidade no procedimento manifestada pela singularidade da matriz de covariância. Desta forma, para assegurar que haja qualidade suficiente na amostragem, é realizada uma amostragem estratificada. Esta amostragem é realizada pela divisão do conjunto amostral em subconjuntos, onde é realizada uma amostragem aleatória. O algoritmo desenvolvido por DUCLOS (1997) é descrito na Figura 2.7.

O cálculo dos estimadores de localização e dispersão, realizado no PASSO 3 da Figura 2.8, é realizado por mínimos quadrados. Para isto, sejam:

- μ e σ os estimadores de localização e dispersão da função de distribuição; e

- ($x_{(1)}$, $x_{(2)}$,...,$x_{(n)}$) um conjunto formado pelas observações ordenadas do vetor $X_k \in \mathbb{R}^n$.

A partir deste vetor, podem-se definir vetores normalizados $\boldsymbol{U_{(r)}}$, tais que:

$$U_{(r)} = \frac{(x_{(r)} - \mu)}{\sigma} \tag{2.11}$$

E, portanto, os respectivos momentos de primeira e segunda ordem

$$E[U_{(r)}] = \alpha_r \tag{2.12}$$

$$Var = [U_{(r)}] = \varpi_{rr} \tag{2.13}$$

$$Cov[U_{(r)}, U_{(s)}] = \varpi_{rs} \tag{2.14}$$

Desta forma, pode-se escrever a Equação 2.15, como

$$Cov(X_k) = \sigma^2.\Omega \tag{2.15}$$

onde $\boldsymbol{\Omega}$ é uma matriz $\boldsymbol{n}$ x $\boldsymbol{n}$ formada pelos elementos $\boldsymbol{\omega_{rs}}$.

Sejam θ e A tais que

$$\theta = \begin{pmatrix} \mu \\ \sigma \end{pmatrix} \tag{2.16}$$

$$A = \begin{bmatrix} 1 & \alpha_1 \\ \vdots & \vdots \\ 1 & \alpha_n \end{bmatrix} \tag{2.17}$$

Desta forma,

$$\theta = (A^t.\Omega^{-1}.A)^{-1}.A^t.\Omega^{-1}.X \tag{2.18}$$

A Equação 2.18 fornece um fator da aproximação para os estimadores de localização e dispersão, através do cálculo de uma interação do método de *bootstrap* do algoritmo. Para completar o cálculo da aproximação, é necessário que esta interação seja realizada um grande número de vezes (***B*** vezes). O valor final dos estimadores é realizado através da média destes valores obtidos em cada interação.

A variância destes estimadores é dada por:

$$Var[\hat{\mu}] = \sigma^2.\frac{\alpha^t.\Omega.\alpha}{\det(A^t.\Omega^{-1}.\mathrm{A})} \tag{2.19}$$

$$Var[\hat{\sigma}] = \sigma^2.\frac{e^t.\Omega.e}{\det(A^t.\Omega^{-1}.\mathrm{A})} \tag{2.20}$$

2.6.3. Limites de Controle

Os gráficos de controle de Shewhart, conforme item 2.3, utilizam três desvios padrões como limites superior e inferior de controle, e a partir de experimentos realizados na indústria, foram definidas regras para definir quando um determinado processo encontra-se fora de controle (Tabela 2.1). Para os gráficos de controle definidos para processos não normais através da metodologia de DUCLOS (2005), foram definidos os mesmos limites de controle, validados por DUCLOS (1997).

Desta forma, são definidos:

$$L.S.C. = \mu + 3\sigma \tag{2.21}$$

$$L.I.C. = \mu - 3\sigma \tag{2.22}$$

onde $\boldsymbol{\mu}$ e $\boldsymbol{\sigma}$ são os estimadores de localização e dispersão obtidos pelo algoritmo desenvolvido por DUCLOS (1997).

2.6.4. *ARL* para o Gráfico de Controle Não Normal

O cálculo do fator *ARL* para o contexto de processos não normais é complexo e calculado de forma particular em cada situação. Uma das possibilidades é o uso da aproximação do processo pela composição de processos normais. DUCLOS (1997), (2005) realizou experimentos que demonstram que, se o processo em estudo tem uma distribuição próxima da normal, então os *ARL*s de Shewhart e o calculado para processos não normais são bem próximos.

Capítulo 3

Sistemas Fuzzy

3.1. Introdução

A lógica *fuzzy*[12] é caracterizada pela transformação do conhecimento impreciso do ser humano, muitas vezes fornecida por especialistas, em modelos matemáticos. Essa transformação permite que termos normalmente utilizados, como "está muito quente", "está um pouco frio", possam ser traduzidos em valores nominais.

O conhecimento humano, desenvolvido através dos milhares de anos, é normalmente caracterizado pela incerteza e imprecisões. Buscando representar esta imprecisão, ZADEH (1978) desenvolveu a Teoria da Possibilidade, baseado na extensão da teoria clássica de conjuntos que já havia desenvolvido anteriormente (1965) e denominado teoria dos conjuntos fuzzy.

Das teorias mais conhecidas, a teoria das possibilidades é a "mais adequada", (TRON 2004), para o tratamento de informações fornecidas por seres humanos, por ser menos restritiva: "é mais fácil dizer que um evento é possível do que provável", (SANDRI 1999).

3.2. Teoria de Conjuntos Fuzzy

Um conjunto fuzzy, (ZADEH 1965), é uma extensão de um conjunto clássico, onde a pertinência de um elemento ao conjunto é interpretada como sendo uma situação intermediária entre a total pertinência e a não-pertinência (JANTZEN 1998).

[12] Há autores que traduzem o termo *fuzzy* como nebuloso e difuso.

Por exemplo, o conjunto das pessoas altas na teoria clássica pode ser representado por:

$$A = \{x_i \mid altura(x_i) \geq 1{,}75m\} \tag{3.1}$$

Desta forma, os elementos pertencerão a este conjunto se, e somente se, atenderem a esta condição. Na teoria desenvolvida por Zadeh, uma dada pessoa terá um grau de compatibilidade com o conceito de alto, descrito por uma função de pertinência, Figura 3.1.

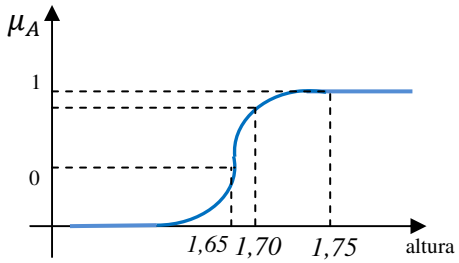

Figura 3.1: Exemplo da função de pertinência.

Um conjunto fuzzy em um universo U é caracterizado por uma função de pertinência $\mu_F: U \rightarrow [0,1]$ que associa a cada $x \in U$ um número real $\mu_F(x)$ no intervalo [0,1], representando o grau de pertinência de x em F. Os valores extremos deste intervalo representam, respectivamente, a compatibilidade ($\mu_F(x) = 1$) e a incompatibilidade ($\mu_F(x) = 0$). Os demais pontos do intervalo, [0,1], representam graus de compatibilidade intermediários.

A cardinalidade, $|F|$, de um conjunto fuzzy F é definida como sendo $\sum_{x \epsilon \mathrm{U}} \mu_F(x)$ para conjuntos discretos e $\int_{\mathrm{U}} \mu_F(x)$ para conjuntos contínuos.

Como exemplo da representação dos reais ($U = \mathbb{R}$), sejam os conjuntos fuzzy "pequeno", "médio" e "grande", definidos como:

$$\mu_P(x) = \begin{cases} 1, se\ x < -1 \\ -x, se\ -1 \leq x \leq 0 \\ 0, se\ x > 0 \end{cases} \quad (3.2)$$

$$\mu_M(x) = \begin{cases} 0, se\ x < -1 \\ x + 1, se\ -1 \leq x \leq 0 \\ -x + 1, se\ 0 \leq x \leq 1 \\ 0, se\ x \geq 1 \end{cases} \quad (3.3)$$

$$\mu_G(x) = \begin{cases} 0, se\ x < 0 \\ x, se\ 0 \leq x \leq 1 \\ 1, se\ x > 1 \end{cases} \quad (3.4)$$

A Figura 3.2 mostra os conjuntos fuzzy P, M e G no universo $U = \mathbb{R}$, e a representação gráfica normalmente utilizada, REZENDE (2003).

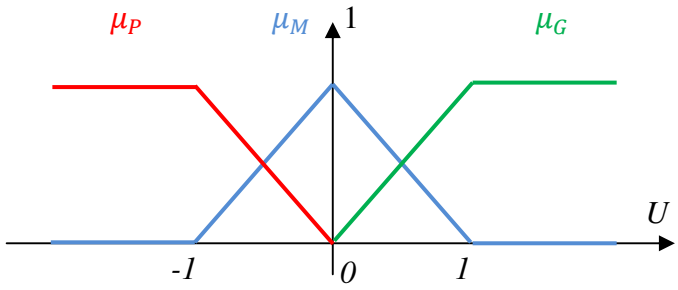

Figura 3.2: Conjuntos fuzzy P, M e G;

3.3. Operações Básicas

As operações básicas sobre estes conjuntos são o complemento (¬A), união (A+B ou A∪B) e interseção (A.B ou A∩B), representados, respectivamente pelas correspondentes funções de pertinência 3.5, 3.6 e 3.7, considerando que *A* e *B* sejam conjuntos fuzzy definidos em *U*.

$$\mu_{\neg A}(x) = 1 - \mu_A(x), \forall x \in U \tag{3.5}$$

$$\mu_{A \cup B}(x) = \max[\mu_A(x), \mu_B(x)], \forall x \in U \tag{3.6}$$

$$\mu_{A \cap B}(x) = \min[\mu_A(x), \mu_B(x)], \forall x \in U \tag{3.7}$$

Além destas, são citados alguns operadores:

- Função $\nabla: [0,1]^2 \rightarrow [0,1]$, satisfazendo as propriedades de comutatividade ($\nabla(\mathrm{a}, \mathrm{b}) = \nabla(\mathrm{b}, \mathrm{a})$), associatividade ($\nabla\big(\mathrm{a}, \nabla(\mathrm{b}, \mathrm{c})\big) = \nabla(\nabla(\mathrm{a}, \mathrm{b}), \mathrm{c})$) e monotonicidade ($\nabla(\mathrm{a}, \mathrm{b}) \leq \nabla(\mathrm{c}, \mathrm{d})$ se $\mathrm{a} \leq \mathrm{c}$ e $\mathrm{b} \leq \mathrm{d}$);

- Operador $\mathrm{T}: [0,1]^2 \rightarrow [0,1]$, definido como sendo o operador *t-norma*, sendo comutativo, associativo e monotônico, além de atender a $\mathrm{T}(\mathrm{a}, 1) = \mathrm{a}$, desde que 1 é o elemento neutro, $T(a, 0) = 0, \forall \mathrm{a} \in [0,1]$;

- Operador $\perp: [0,1]^2 \rightarrow [0,1]$, definido como sendo o operador *t-conorma*, sendo comutativo, associativo e monotônico, além de atender a $\perp (\mathrm{a}, 0) = \mathrm{a}$, 0 o elemento neutro, para e $\perp (\mathrm{a}, 1) = 1, \forall \mathrm{a} \in [0,1]$.

- Leis de Morgan:

  - $\neg\big(\mathrm{T}(\mathrm{a}, \mathrm{b})\big) = \perp (\neg \mathrm{a}, \neg \mathrm{b})$; (3.8)
  - $\neg\big(\perp (\mathrm{a}, \mathrm{b})\big) = \mathrm{T}(\neg \mathrm{a}, \neg \mathrm{b})$; (3.9)

O operador negação, denotado por ¬, é definido por $\neg a = 1 - a$ e os operadores *t-norma* e *t-conorma* são duais com relação a este operador, se eles satisfizerem as Leis de Morgan.

Ainda há outros operadores de implicação, cujos principais são descritos na Tabela 3.1, (SANDRI 1999).

Tabela 3.1: Principais operadores de implicação

Implicação	**Nome**
$\max(1-a, b)$	Kleene-Diemes
$\min(1-a+b, 1)$	Lukasiewicz
$\begin{cases} 1, & se\ a \leq b \\ 0, & caso\ contrário \end{cases}$	Rescher-Gaines "Sharp"
$\begin{cases} 1, & se\ a \leq b \\ b, & caso\ contrário \end{cases}$	Brower-Gödel
$\begin{cases} \min(b/a), & se\ a \neq b \\ 0, & caso\ contrário \end{cases}$	Goguen
$1 - a + a.b$	Reichenbach "Estocástica"
$\max(1-a, \min(a, b))$	Zadeh-Wilmott

3.4. Representação do Conhecimento

Algumas vezes a representação do conhecimento não pode ser realizada através de variáveis quantitativas, principalmente quando se está buscando uma forma de representar o conhecimento humano, sendo neste caso mais utilizada a representação fuzzy, por sua característica qualitativa.

Essa representação pode ser realizada através de variáveis lingüísticas que são definidas como sendo entidades utilizadas para representar o conhecimento de modo lingüístico, admitindo como valores termos como "muito" e "um pouco". Desta forma,

valores imprecisos serão classificados em um termo lingüístico. São as funções de pertinência que representam o quanto estes elementos satisfazem o conhecimento representado pelos conjuntos fuzzy.

Formalmente, uma variável lingüística é definida como sendo formada pela quádrupla (X, U, T(X), M), onde X é o nome da variável, U é o universo de discurso de X, T(X) é um conjunto de nomes de valores de X, e M é uma função que associa uma função de pertinência a cada elemento de T(X).

A variável lingüística, utilizada de forma sintática, é definida a forma com que as informações lingüísticas são armazenadas. Esta coleção de informações armazenadas é conhecida como base de dados e possui as sentenças atômicas.

A variável lingüística, utilizada de forma semântica, é definida a partir da forma com que o conhecimento é representado através de declarações condicionais, conhecidas como regras fuzzy. Estas regras são representadas por sentenças do tipo **se-então.** Se o condicionante de uma regra for satisfeita, então o conseqüente será válido e será processado, contribuindo com a saída do sistema. O sistema construído desta maneira é conhecido como sistema (ou modelo) baseado em regras fuzzy.

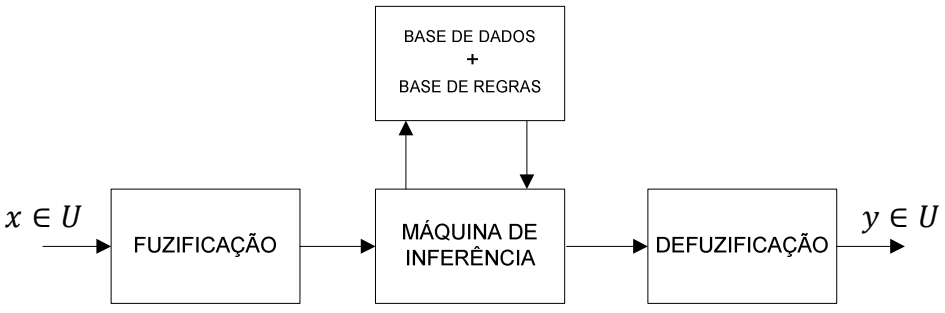

Figura 3.3: Diagrama de um sistema baseado em regras fuzzy.

3.5. Sistema Baseado em Regras Fuzzy

A forma com que a informação é armazenada irá influenciar diretamente na inferência, na forma com que os antecedentes serão processados, quais serão os níveis de ativação das regras e quais serão os operadores que serão ativados nos conjuntos fuzzy.

Sendo esta máquina somente capaz de tratar termos lingüísticos, torna-se necessário, para o caso onde as entradas e saídas serem valores numéricos, uma etapa prévia de conversão destes valores para estes termos lingüísticos, chamada de fuzificação e outra, após a máquina, chamada de desfuzificação, representadas na Figura 3.3, SANDRI (1999).

3.5.1. Fuzificação

A primeira conversão, conhecida como fuzificação (do inglês *fuzzification*), é responsável pela conversão dos valores numéricos da entrada, para termos linguísticos, que alimentarão a máquina de inferência.

Nesta etapa, são formuladas as funções de pertinência para cada variável, de forma a englobar totalmente o domínio dos valores escalares encontrados na entrada do sistema. Estas funções são responsáveis por representar o conhecimento humano, conforme exemplificado na introdução deste capítulo. As funções mais comuns encontradas são a triangular, trapezoidal e *singleton*.

Funções de pertinência triangulares, Figura 3.4, são elaboradas de forma a determinar estados conhecidos (“quente” e “frio”, por exemplo).

As funções trapezoidais podem ser interpretadas como sendo uma extensão do caso triangular. Nestas funções, os valores dos estados não estão mais caracterizados em instantes pré-determinados, como nas triangulares, mas sim em patamares, conforme representado na Figura 3.5.

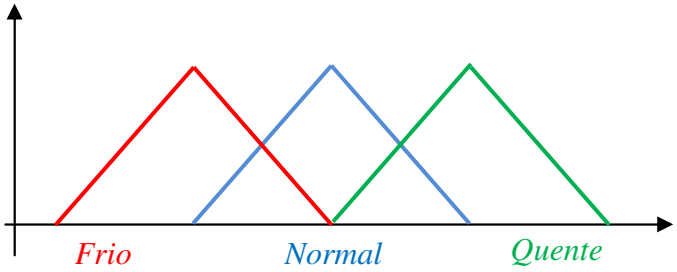

Figura 3.4: Função de pertinência triangulares.

A função de pertinência do tipo *singleton*, Figura 3.6, é caracterizada por converter um valor real, e é representado graficamente por uma linha vertical.

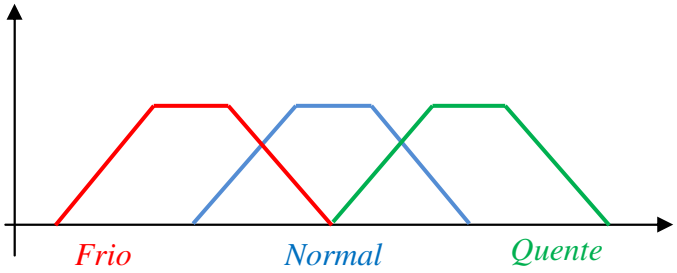

Figura 3.5: Função de pertinência trapezoidais.

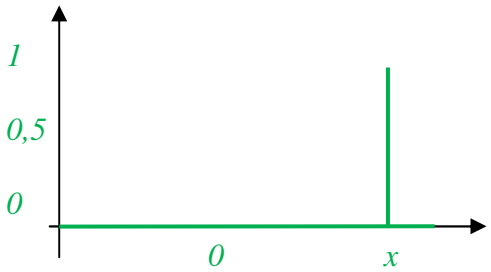

Figura 3.6: Conjunto unitário (*singleton*).

Portanto, com as funções de pertinência caracterizadas, o primeiro passo é realizar a conversão, propriamente dita, entre os valores adquiridos do processo para termos linguísticos obtidos do conhecimento humano de um especialista, por exemplo.

Estas funções de pertinência possuem em suas abscissas valores coerentes com aqueles existentes no processo.

3.5.2. Máquina de Inferência – Base de Conhecimento

Com os valores de entrada, a máquina de inferência é acionada e a base de conhecimento é consultada. É formada por uma base de dados, onde ficam armazenadas as definições de discretização e de normalização, e pela base de regras, que é formada por regras que seguem a forma genérica descrita na Equação 3.10.

$$SE\ x_1 = A_i\ E\ x_2 = A_j\ E\ \ldots E\ x_p = A_i\ ENTÃO\ y_1 = B_i\ E\ y_2 = B_m \quad (3.10)$$

A partir deste ponto, as regras são acionadas e avaliadas pela máquina de inferência. O procedimento de inferência, descrito na Figura 3.7, irá, baseado na teoria de conjuntos formulada por ZADEH (1965), calcular a resposta do conjunto de regras de produção acionadas.

As determinações dos valores de conclusão e dos valores obtidos pelo procedimento de inferência podem mudar de acordo com o modelo que está sendo adotado. Estes modelos são, basicamente, classificados através de dois tipos de controladores fuzzy: lingüísticos e funcionais.

Os controladores lingüísticos são caracterizados pelos resultados obtidos em cada regra ser um termo fuzzy pertencente a um conjunto fixo de termos. Exemplos de modelos lingüísticos são os de MAMDANI (1976) e o de Larsen.

O modelo de Mamdani foi desenvolvido na década de 1970 e é um dos primeiros a se tornarem referência. É caracterizado por possuir relações fuzzy tanto em seus antecedentes como em seus consequentes. Neste método o operador $\mathbf{min}\ (\boldsymbol{a}, \boldsymbol{b})$ é utilizado como operador t-norma e o operador $\mathbf{max}\ (\boldsymbol{a}, \boldsymbol{b})$ é usado como operador de

agregação. O modelo de Larsen, por sua vez, usa como operador t-norma e $\boldsymbol{a} * \boldsymbol{b}$, enquanto que usa $\mathbf{max}\ (\boldsymbol{a}, \boldsymbol{b})$ como operador de agregação.

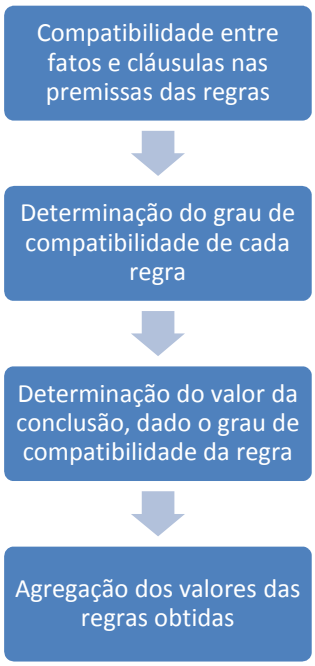

Figura 3.7: Procedimento de inferência.

Para os modelos fuzzy funcionais, para cada regra é obtido um valor para uma variável de controle. O valor final é resultado de uma média ponderada dos valores obtidos levando-se em consideração o grau de compatibilidade entre a premissa da regra.

São exemplos desta categoria os controladores de Tsukamato e o de Takagi-Sugeno. No primeiro a conclusão é representada através de uma função estritamente monotônica, geralmente não linear, tendo como domínio os possíveis graus de compatibilidade (α_j) entre cada premissa e as entradas. Os valores obtidos como conclusão são representados pela média ponderada calculada através da compatibilidade já citada, conforme Equação 3.11.

$$y' = \frac{\Sigma_{j=1}^{n}(\alpha_j . y'_j)}{\Sigma_{j=1}^{n}\alpha_j} \tag{3.11}$$

Neste modelo, em particular, não há processo de desfuzificação, dada a forma com que o valor final é obtido.

O modelo Takagi-Sugeno, proposto por Takagi e Sugeno na década de 80, (REZENDE 2003) , é baseado em regras condicionais de inferência, com o conseqüente formado por equações relacionando entradas e saídas do processo, conforme Equação 3.12.

$$SE\ x_1 = A_i\ E\ x_2 = A_j\ E\ \ldots E\ x_p = A_i\ ENTÃO\ y = \phi(x_1, x_2, .. x_p) \quad (3.12)$$

3.5.3. Desfuzificação

A segunda conversão, desfuzificação (do inglês *defuzzification*), ocorre após o processo de inferência e é responsável por quantizar o resultado obtido pelo processo de inferência. Presente somente nos controladores fuzzy clássicos. Esta etapa é responsável por obter a ação única a ser realizada pelo sistema, através da tradução de termos lingüísticos para valores escalares.

Nesta quantização, existem dois métodos que são os mais usados: centro de massa e média dos máximos. Dos demais modelos encontrados na literatura, (JANTZEN 1998), destacam-se, ainda, a média ponderada dos máximos que representa a média dos valores centrais ativados com os graus de pertinência representando os respectivos pesos e o método critério de máximo (ou mínimo) que produz um valor numérico igual ao máximo (mínimo) valor ativado, que é mais adequado quando a forma de distribuição de possibilidades apresenta picos.

a) Centro de Massa: representa o centro de gravidade da distribuição de possibilidade de saída do sistema fuzzy. O cálculo é realizado através da média ponderada entre as áreas dos gráficos e os valores médios das abscissas $\bar{y}_i$.

$$y = \frac{\sum \bar{y}_i S_i}{\sum \bar{y}_i} \quad (3.13)$$

b) Média dos Máximos: representa o valor numérico médio de todos os valores ativados. É calculado pela média aritmética dos valores obtidos.

$$y_i = \frac{\sum m_i}{n} \tag{3.14}$$

Como resultado da desfuzificação, são obtidos valores numéricos que são fornecidos ao processo. Assim, esta técnica possibilita a criação de uma ferramenta de controle para processos, baseada no conhecimento humano adquirido pela experiência prática ou de um especialista.

Capítulo 4

Algoritmo de Detecção de Anomalias com Janelas Adaptativas

4.1. Negação de Serviço Distribuído

"Segurança da Informação está relacionada com métodos de proteção aplicados sobre um conjunto de dados no sentido de preservar o valor que possui para um indivíduo ou uma organização." (ISO 2005)

Dentro desta área, as características básicas são a confidencialidade, integridade e a disponibilidade. Os ataques de negação de serviço objetivam indispor informações e serviços ofertados.

Conforme definição dada pelo CERT[13], ataques de negação de serviço (*Denial of Service - DoS*) são caracterizados por uma tentativa explícita de um atacante impedir que um usuário legítimo utilize um determinado serviço[14]. Este tipo de ataque pode ocorrer, basicamente de três formas: consumo de recursos escassos, limitados e não renováveis; destruição ou modificação da informação e destruição física ou modificação dos componentes da rede.

[13] Centro de Coordenação de Incidentes na Internet, fundado pelo DARPA (*Defense Advanced Research Projects Agency*) a partir do Instituto de Engenharia de Software (SEI) da Universidade Canergie Mellon em Pittsburgh, Pensilvânia, EUA.
[14] Denial of Service Attacks, Coordinator Center http://www.cert.org/tech_tips/denial_of_service.html

Estes tipos de ataques obtêm sucesso porque a Internet foi inicialmente desenvolvida tendo com o objetivo de prover algumas funcionalidades desejáveis, porém muitas vezes sem considerar os requisitos mínimos necessários de segurança.

É, por exemplo, o caso do aplicativo *ping*. Este aplicativo tem por objetivo verificar a interconexão entre dois computadores, e possui tamanho (do pacote ICMP) pré-determinado. Porém, a simples alteração deste tamanho provocava[15] um ataque de negação de serviço, conhecido como *ping* da morte (pod – *ping of death*).

O primeiro ataque de negação de serviço notificado aconteceu em 1996, quando um provedor de serviços de Internet da cidade de Nova Iorque tornou-se indisponível durante algumas semanas sem razão identificada. Este ataque ocorreu com o PANIX (*Public Access Networks Corporation*) e, teoricamente já havia sido previsto para ocorrer anos antes, por causa da vulnerabilidade do protocolo TCP (HOUSEHOLDER 2001). Mais tarde, em fevereiro de 2000 (PAUL 2001), uma versão distribuída deste ataque aconteceu com servidores de várias empresas, simultaneamente. Estes servidores, dentre eles os do Yahoo e do E-bay, tornaram-se indisponíveis por dois dias.

Um ataque de negação de serviço distribuído (*Distributed Denial of Service - DDoS*) é uma variação do ataque de negação de serviço em que um conjunto muito grande de máquinas contaminadas são usadas para realizar o ataque, (RAZMOV 2000).

Nesta modalidade, (TAROUCO 2003), é criada uma hierarquia de máquinas composta de atacante, que é o equipamento que dispara inicialmente todo o processo; estações mestres, que são as máquinas infectadas diretamente pelo atacante e que irão mobilizar as máquinas que efetivamente realizarão o ataque, chamadas de zumbi, Figura 4.1.

[15] Os sistemas operacionais mais modernos possuem proteções que evitam esta alteração de tamanho do pacote ICMP – Internet Control Message Protocol.

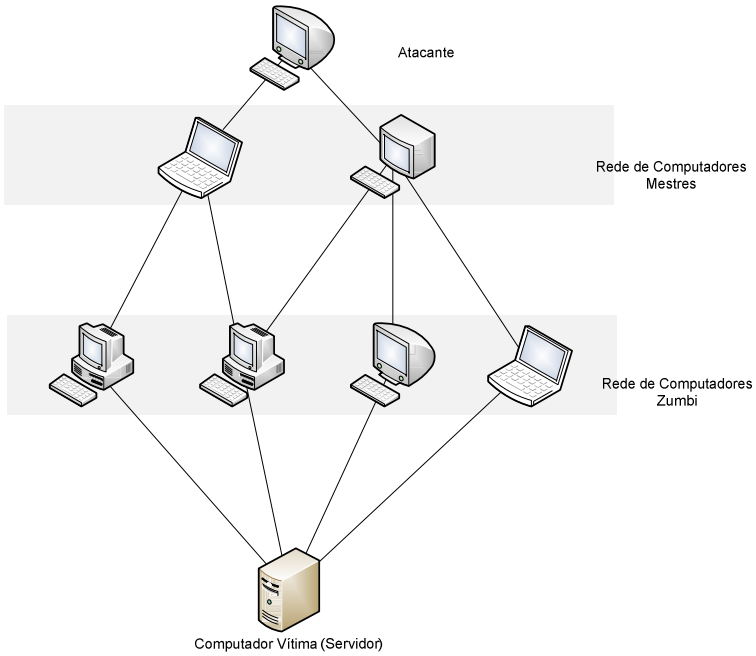

Figura 4. 1: Exemplo de uma rede de ataques de negação de serviços distribuídos.

4.2. Classificação dos Ataques de Negação de Serviço Distribuído

Em (MIRKOVIC 2004) é estabelecido uma taxonomia de ataques de negação de serviços e de mecanismos de defesa para ataques de negação de serviço distribuído. Esta taxonomia será descrita nos próximos itens, visando realizar o enquadramento do Algoritmo de Detecção de Anomalias com Janelas Adaptativas nesta taxonomia.

4.2.1. Com Relação ao Grau de Automação

Eles podem ser classificados em manuais, semi-automáticos e automáticos. Os ataques semi-automáticos podem ainda ser classificados, com relação à comunicação em diretos e indiretos. Nos diretos, o equipamento atacado precisa ter contato direto com a máquina atacante; no indireto, a comunicação entre estes equipamentos acontece por outros meios, como por exemplo por canais de IRC[16].

[16] Internet Relay Chat – protocolo de comunicação utilizado, basicamente, para bate-papo e troca de arquivos.

Os ataques automáticos e semiautomáticos podem ainda ser classificados pela estratégia de escaneamento da rede:

a) aleatória

b) listagem: através de uma listagem externa, que é preenchida por uma terceira máquina que testa continuamente as vulnerabilidades dos equipamentos.

c) topológica: através de informações da topologia da rede que uma máquina já infectada está conectada.

d) permutação: ataques a equipamentos cujos endereços são obtidos através da permutação de um endereço base.

e) subrede: escanea todos os equipamentos que estão na mesma subrede de um equipamento já atacado.

E com relação ao mecanismo de propagação, pode ser classificada em central, encadeada e de forma autônoma. No primeiro caso um equipamento central é infectado e a partir dele, todas as demais da mesma rede. No segundo caso, as máquinas são infetadas uma a uma e no terceiro caso, sem ordem pré-estipulada.

4.2.2. Com relação à exploração de vulnerabilidades

Eles podem ser classificados em ataques por protocolo, onde fraquezas do protocolo são exploradas, e por força bruta. Esta última ainda pode ser classificada se podem (filtráveis) ou não (não filtráveis) serem bloqueadas por um *firewall*.

4.2.3. Com Relação à Taxa de Infecção Dinâmica

O ataque é considerado contínuo quando, após ter sido iniciado, o mesmo acontece com força total. É considerada variável quando o ataque não acontece com força total, exatamente para evitar que seja identificada. Neste caso, pode ser classificado como flutuante ou crescente.

4.2.4. Com Relação ao Impacto

Pode ser considerado destrutivo ou degradante. No primeiro caso, o ataque altera as configurações do servidor alvo. Já no segundo caso, ao invés de destruir, o ataque consome parte dos recursos do equipamento.

O quadro apresentado na Figuras 4.2 e 4.3 resumem a classificação exposta.

4.3. Classificação dos Mecanismos de Defesa dos Ataques de Negação de Serviço Distribuído

4.3.1. Quanto ao Nível de Atividade

4.3.1.1. Preventivos

São aqueles que possuem mecanismos para eliminar a possibilidade de ataques DDoS ou para proteção de vítimas em potencial, sem negar os seus serviços à clientes legítimos. Podem ser classificados com relação aos mecanismos de prevenção de ataques e e de prevenção de negação de serviço.

a) Prevenção de ataques: modifica as configurações do sistema para eliminar a possibilidade de ataques. Baseados no foco de segurança, ainda podem ser divididos em segurança do sistema e segurança no protocolo.

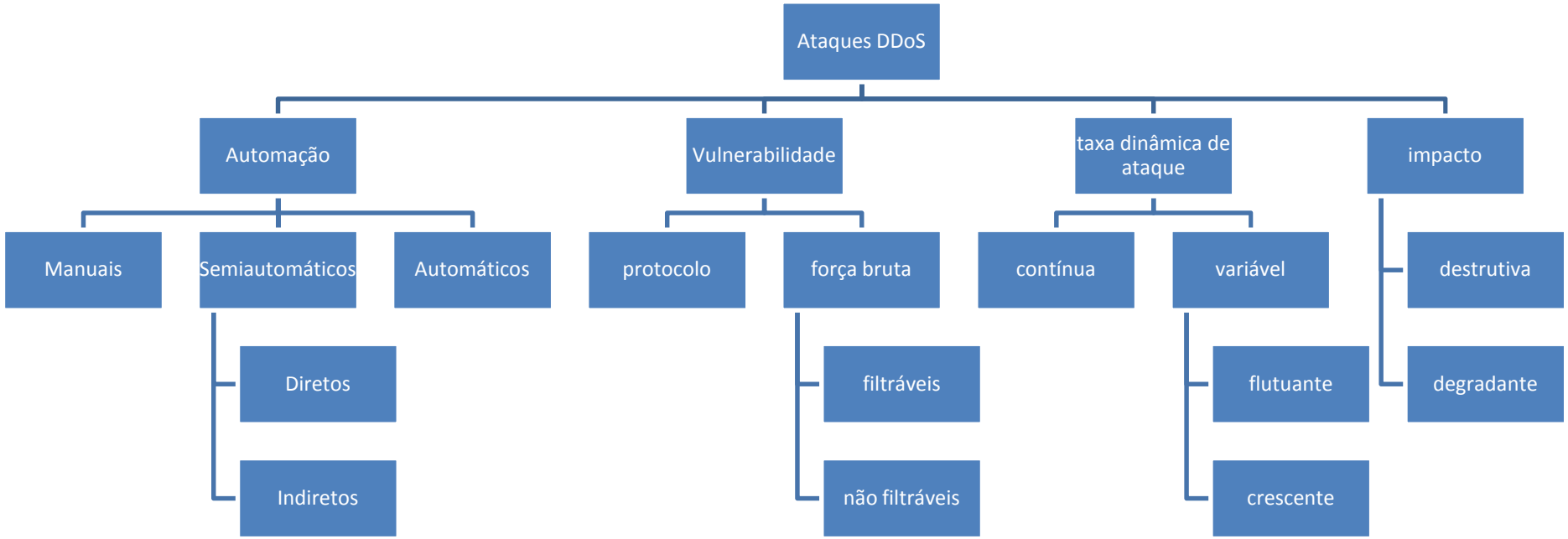

Figura 4. 2: Classificação dos ataques DDoS.

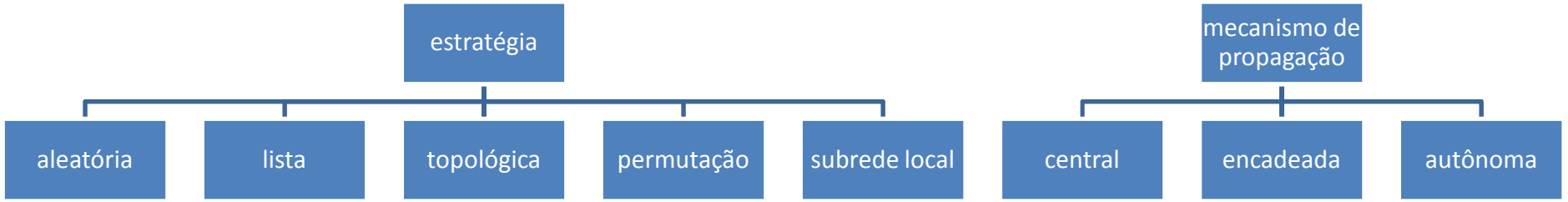

Figura 4. 3: Classificação dos ataques DDoS, pela estratégia e pelo mecanismo de propagação

i) Segurança do Sistema: altera as configurações do sistema, alterando as configurações, removendo *bugs* e realizando atualizações.

ii) Segurança do Protocolo: mecanismos que objetivam prover segurança relacionados aos problemas de vulnerabilidade dos protocolos.

b) Prevenção de *DoS* : mecanismos que provêem mais segurança no sistema, evitando a negação de serviços a clientes legítimos.

i) Acesso à recursos: mecanismos que asseguram que somente usuários autorizados possam acessar determinados recursos.
ii) Multiplicação dos recursos: permite o balanceamento de carga dos servidores.

4.3.1.2. Reativos: Possuem a Estratégia de "Detectar e Responder"

a) Quanto à estratégia de deteção

i) Reconhecimento de padrões: realiza a deteção através da comparação de padrões existentes com os padrões de ataques conhecidos. Possui baixos falsos positivos e negativos. Alto grau de eficiência.

ii) Anomalia: realiza a comparação com uma situação considerada normal. Quaisquer diferenças podem ser consideradas como ataque. Torna-se particularmente útil na identificação de novos ataques.

iii) Híbrido: possui características comuns aos mecanismos de reconhecimento de padrões e anomalia.

iv) Outros: utilizam mecanismos adicionais às infraestruturas de rede, como camadas adicionais aos protocolos.

b) Quanto à estratégia de resposta

i) identificação dos agentes: provê informações sobre o equipamento que proferiu o ataque.

ii) limitação de taxa: realiza uma limitação de recursos que estão sendo utilizados pelo atacante.

iii) filtragem: realiza a deteção e filtra os dados enviados do computador atacante.

iv) reconfiguração: realiza a deteção e reconfigura a topologia da rede de forma a isolar o equipamento atacado do computador atacante.

c) Quanto ao grau de cooperação

i) autônomo: possui mecanismos separados para a deteção e resposta ao ataque.

ii) cooperativo: possui mecanismos separados para a deteção e ataque, igual ao item anterior, porém realiza interações com outras entidades nestes procedimentos.

iii) interdependente: os mecanismos de ataque e resposta são interligados.

d) Quanto à localização

i) rede da vítima: provê mecanismos de defesa na rede em que o equipamento da vítima se encontra.

ii) por intermédio da rede: provê mecanismos que possibilitam defender o equipamento vítima por intermédio da infraestrutura de rede.

iii) rede de origem: provê mecanismos que evitam que um ataque *DDoS* possa ser originado na rede local.

Na Figura 4.4 é apresentado um resumo desta classificação.

4.4. Estado da Arte

Os estudos relacionados à pesquisa de técnicas de identificação e defesa contra ataques cibernéticos iniciaram, praticamente, com a era da Internet. Embora já existissem sistemas de informação anteriormente, estes não eram alvos de ataques cibernéticos, mas sim de ataques de outra natureza. Além disso, uma das características marcantes dos ataques é a busca da confidencialidade dos seus autores. São exemplos desta situação os casos de *ip spoofing* (PARK 2001 e JIN 2003), quando o ataque é realizado com endereço IP diferente do real e no caso de *DDoS*, onde o ataque é realizado por máquinas, ditas zumbis, e não pelo atacante propriamente dito.

Outro fator importante é que estes ataques são caracterizados, essencialmente, pelo envio, em massa ou não, de pacotes de rede artificiais, ou seja, que não correspondem a nenhuma operação verdadeira na rede. Além disso, os ataques são disparados por ferramentas disponíveis e que sejam fáceis de usar, como TFN, TFN2K e Trinoo, dentre outras, ou que possuam uma forma de operação padronizada, como é o caso dos ataques distribuídos.

Desta forma, alguns ataques poderão ter:

- uma assinatura predeterminada (SCHUBA 1997, NOURELDIEN 2002, SOMMER 2003 e LIMWIWATKUL 2004);
- um padrão predeterminado (SOMMER 2003 e MOORE 2006); e
- alterar, significativamente, o tráfego de rede.

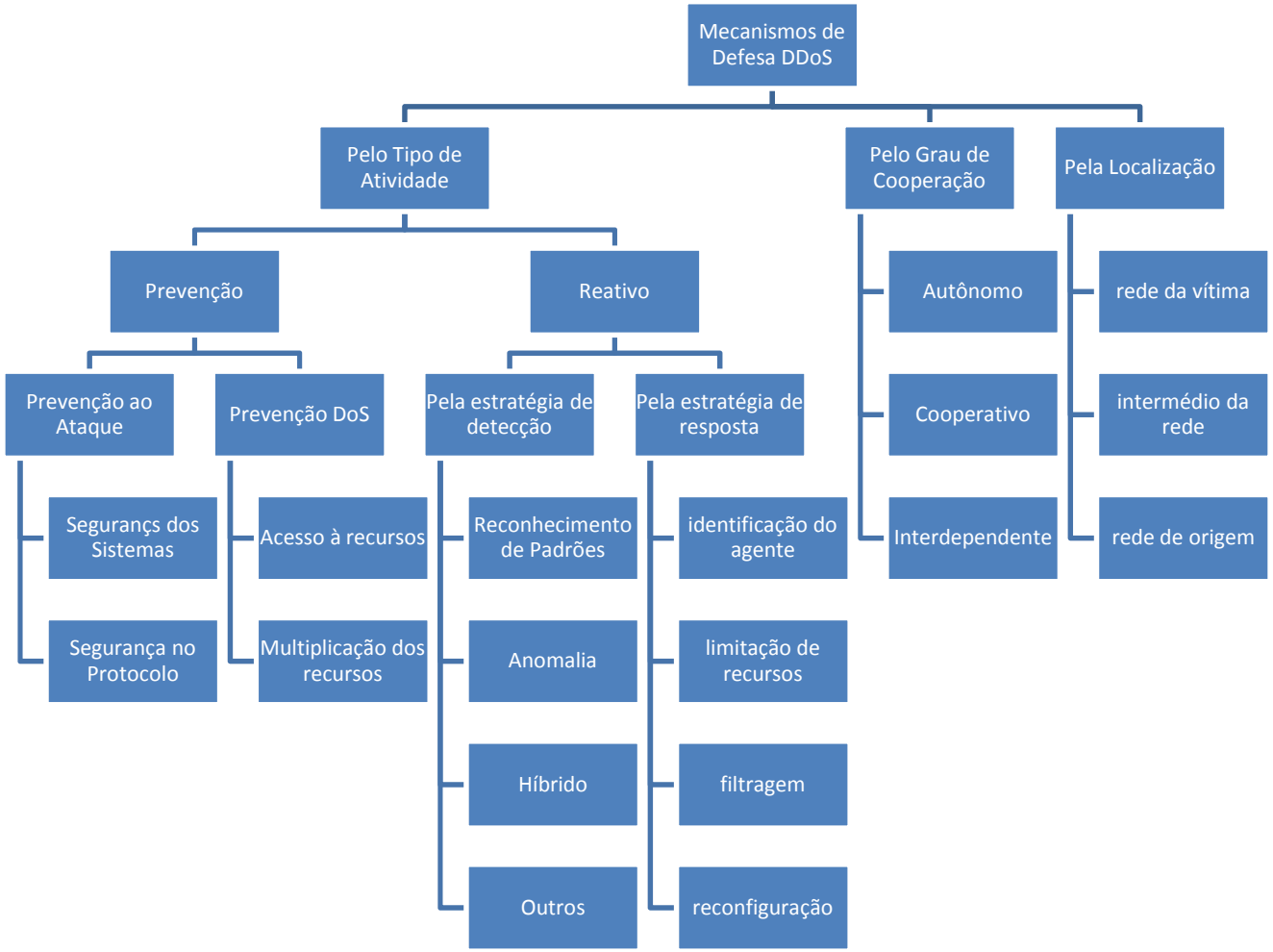

Figura 4. 4: Mecanismos de defesa DDoS;

Caso o ataque altere significativamente o tráfego da rede, algumas estratégias poderão ser realizadas visando sua identificação. Dentre estas, podemos destacar:

- Pesquisa do aumento do tráfego (SAVAGE 1999, SINCLAIR 1999, PARK 2001, BARFORD 2002, ESTAN 2003, YEGNESWARAN 2003, IOANNIDIS 2002, LAKHINA 2004a, LAKHINA 2004b, LAKHINA 2004c, QIN 2004a, DÜBENDORF 2005, CANSIAN 2007, OLIVEIRA 2006, ROJKOVA 2007 e ROJKOVA 2007b);

- Mudança do comportamento do tráfego da rede (PTACEK 1998, GIL 2001, HANDLEY 2001, BARFORD 2002, MIRKOVIC 2002, MOHINDDIN 2002, PENG 2003, YEGNESWARAN 2003, LAKHINA 2004a, MOORE 2004, PANG 2004, LIMWIWATKUL 2004 e LI 2005).

Como os ataques nada mais são do que a geração de tráfegos não verdadeiros, porém muitas vezes tecnicamente válidos, uma outra abordagem está relacionada na busca de anomalias geradas. Neste caso, são realizadas comparações do tráfego da rede em momentos considerados normais e em outras situações, suspeitas de ataques. Nesta categoria, destacam-se os trabalhos: TENG 1990, LANE 1997a, 1997b, 1999, SAVAGE 1999, SINCLAIR 1999, DICKERSON 2000, YE 2000, LEE 2001, BARFORD 2002, MAHADIK 2002, MIRKOVIC 2002, FEINSTEIN 2003, LAKHINA 2004a, 2004b, 2004c, LIMWIWATKUL 2004, QIN 2004b, SIATERLIS 2004, CANSIAN 2007 e HU 2007.

Todavia, é possível aplicar outras teorias como ferramentas auxiliares na identificação e detecção dos ataques. Dentre as abordagens mais utilizadas, destacam-se:

- Medidas estatísticas: através do estudo de estatísticas do tráfego da rede (MAHADIK 2002, FEINSTEIN 2003, KUZMANOVIC 2003, LI 2005 e ROJKOVA 2007);

- Gráficos de controle (MAHADIK 2002, SANTOS 2007a e 2007b);

- Qualidade de serviço (MAHADIK 2002);
- Algoritmos genéticos (SINCLAIR 1999);
- Conceitos de inteligência artificial (TENG 1990, LANE 1997a, 1997b, 1999, SINCLAIR 1999, LEE 2000, SIATERLIS 2005, SILVA 2005 e CANSIAN 2007);
- Técnicas de clusterização (LANE 1999, MARCHETTE 1999 e JIN 2003);
- Análise multivariada de séries temporais (LAKHINA 2004a e JIN 2004);
- Wavelets (BARFORD 2002);
- Entropia (LEE 2001 e FEINSTEIN 2003);
- Cadeias de Markov (YE 2000);
- Mineração de dados (DICKERSON 2000, LEE 2000, QIN 2004a e 2004b);
- Autômatos finitos (BRANCH 2002);
- *IP Traceback* (SAVAGE 2000, PARK 2001, SONG 2001, JOZIC 2002, YAAR 2003, WONG 2006 e CASTELUCIO 2009);
- P2P (CHEN 2004);
- Agentes móveis (MELL 2000);
- *Pushback* (MAHAJAN 2002 e IOANNIDIS 2004);
- Análise espectral (CHENG 2002 e HUSSAIN 2003);

- PCA (LAKHINA 2004c); e
- Fuzzy (DICKERSON 2000).

A partir de técnicas de detecção e identificação, conforme apresentado, a defesa adequada poderia ser implementada. Outra forma proposta é a simples realocação de recursos, direcionando o ataque para outros equipamentos, como (DEFRAWY 2006), diferente dos focos iniciais do ataque.

4.5. O Problema

Os ataques de negação de serviço tem se tornado um problema freqüente, muito embora diversas pesquisas já tenham sido realizadas. Exemplos de ataques ocorridos nos últimos meses, citados na introdução deste trabalho e os trabalhos citados no item 4.4 comprovam este fato.

A anatomia deste tipo de ataque permite que diversas abordagens possam ser exploradas pelos atacantes, destacando-se como principal o anonimato do atacante. Portanto, o estudo mais generalizado deste tipo de ataque poderá permitir avanços na sua detecção.

Esta anatomia pode ser caracterizada em um passo inicial onde o atacante pesquisa pequenos erros (conhecidos como *bugs*) na implementação dos *softwares*. Com este conhecimento obtido, o atacante infecta o maior número possível de computadores que irão realizar o ataque. Estes computadores infectados com programas (por exemplo, cavalos de tróia[17]) que, de maneira coordenada e simultânea, realizarão o ataque.

Este ataque é caracterizado pelo envio de muitos pacotes, oriundos dos diversos computadores infectados, para um determinado computador alvo. Estes pacotes acarretarão a sobrecarga da rede e a conseqüente negação de serviço do computador

[17] Do inglês *trojan horse*, fazendo uma alusão a lenda do artifício usado por Odisséi na conquista de Tróia.

alvo. Esta sobrecarga pode ser caracterizada, dentre outras formas, pelo intervalo de tempo entre os pacotes, que durante o ataque terão uma queda significativa do seu valor, de forma brusca e rápida.

Esta particularidade permitiu que fosse explorada a teoria de gráficos de controle, ao invés de buscar um modelo para a transmissão de dados pela Internet, já realizado anteriormente diversas vezes, como ALVES (2003a).

Uma das formas disponíveis de realizar esta identificação é monitorar a variável intervalo de tempo entre os pacotes através da teoria generalizada de gráficos de controle para processos não normais (DUCLOS 1997), visando identificar mudanças bruscas de comportamento. A seleção do domínio de cálculo dos estimadores de localização e dispersão foi realizada através de um sistema *fuzzy*, tornando o algoritmo mais adaptável às condições particulares de cada rede.

4.6. Base de Dados

O Laboratório Lincoln do Instituto de Tecnologia de Massachusetts (MIT), junto com a Agência de Projetos de Pesquisa Avançada em Defesa dos Estados Unidos (DARPA[18]), nos anos de 1998 até 2000, realizou experimentos relacionados à segurança da informação gerando uma simulação baseada em uma rede de computadores de uma base área americana (Eyrie AFB).

Esta pesquisa foi desenvolvida com quatro objetivos principais (HAINS 1999):

- Prover suporte aos desenvolvedores de sistema detectores de invasão;
- Avaliar abordagens de detectores de invasão;
- Auxiliar a pesquisa de diferentes abordagens; e
- Auxiliar o DARPA em futuras pesquisas.

A base de dados de 1998 foi a primeira a ser desenvolvida e tinha objetivos limitados para avaliar somente as tecnologias de deteção de intrusão desenvolvidos pelo

[18] http://www.ll.mit.edu/mission/communications/ist/corpora/ideval/

DARPA, e não as demais, incluindo as comerciais. Durante o período de nove semanas, 38 (trinta e oito) tipos de ataques foram realizados, classificados em negação de serviço (*dos – denial of service*), acesso não autorizado a recursos locais oriundos de equipamentos remotos (*r2l – remote to local*), acesso não autorizado a conta de administrador da rede oriundo de contas de usuários (*u2r – user to root*), levantamento não autorizado das vulnerabilidades da rede e de seus equipamentos (*probe*) e acesso não autorizado a dados em servidores locais e remotos (*data*). Estes dados foram coletados com auxílio da ferramenta *tcpdump*[19] e disponibilizados para a pesquisa no site do Laboratório Lincoln. A lista de ataques desta base de dados está apresentada na Tabela 4.1.

Com a base de dados de 1999, foram acrescentados ataques internos e equipamentos com o sistema operacional Windows NT. Ainda foram disponibilizadas informações importantes do sistema operacional, como os registros de auditoria do Windows NT, e dados do fluxo interno de pacotes da rede (*sniffed data*), Figura 4.5. Também foram acrescentados cinco novos ataques de negação de serviço, cinco novos ataques *u2r*, cinco novos ataques *r2l* e quatro novos ataques do tipo *probe*.

Tabela 4.1: Tipos de ataques da base de dados de 1998[20].

Tipo de ataque	Ataques	Quantidade de tipos	Quantidade de eventos
Dos	back, neptune, ping of death, smurf, syslogd, land, apache2, mailbomb, process table, udp storm, teardrop.	11	43
r2l	dictionary, ftp-write, guest, phf, httptunnel, xlock, xsnoop, imap, named, sendmail, rootkit, warez, warezmaster, warezclient	14	16
u2r	eject, ffbconfig, fdformat, ps, loadmodule, perl, xterm	7	38
probe	ip sweep, nmap, port sweep, satan, mscan, saint	6	17

[19] http://www.tcpdump.org
[20] No apêndice B está a descrição dos ataques de DoS e DDoS das bases de dados do DARPA de 1998 e 1999.

A base de dados gerada em 2000 foi idealizada de acordo com um cenário específico de um ataque considerando as fases deste ataque.

Como o objetivo deste estudo foi a aplicabilidade do algoritmo em uma situação que se aproximasse de um caso real, foi selecionada a base de dados de 1999. Não foi feita distinção entre as semanas de treinamento e de ataque, ou seja, ao invés de ser fornecido inicialmente três semanas para "treinamento" do modelo e depois as duas últimas para verificação do modelo, preferiu-se fornecer as cinco semanas para o algoritmo. Isto se deve, por que em situações reais não é possível assegurar que uma rede esteja livre de ataques, em um determinado momento.

A base de dados de 1999 possui cinco períodos semanais de dados, com os períodos descritos na Tabela 4.2 (HAINS 1999).

A rede simulada, Figura 5.5, é composta de duas partes. Ao centro da figura está representado o roteador CISCO que separa a parte interna da rede da parte externa. Na parte interna constam cinco equipamentos: Solaris (Locke – 172.16.112.10), Linux (Marx – 172.16.114.50), SunOS (Zeno – 172.16.113.50), Solaris (Pascal – 172.16.112.50) e o gerador de tráfego interno (Hobbes – 172.16.112.20). Na parte externa há três equipamentos: sniffer (SunOS – Solomon – 192.168.1.90), o gerador de dados externos (Calvin – 192.168.1.10) e um servidor web (Aesop – 192.168.1.20).

Tabela 4. 2: Dados produzidos no experimento de 1999.

Semana	Descrição	Início	Final
1	Semana de treinamento, sem ataques	1 Mar 99 – 8:00h	6 Mar 99 – 6:00h
2	Semana de treinamento, com ataque	8 Mar 99 – 8:00h	13 Mar 99 – 6:00h
3	Semana de treinamento, sem ataque	15 Mar 99 – 8:00h	20 Mar 99 – 6:00h
4	Semana de teste	29 Mar 99 – 8:00h	3 Abr 99 – 6:00h
5	Semana de teste	5 Abr 99 – 8:00h	10 Abr 99 – 6:00h

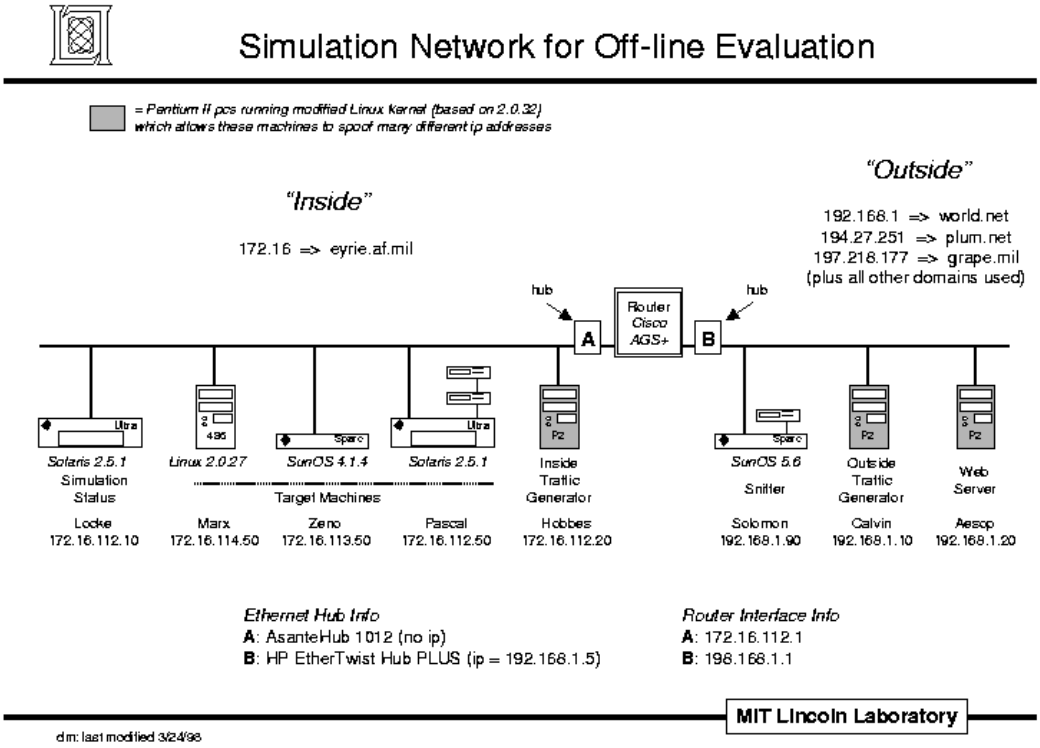

Figura 4. 5: Rede Simulada, (HAINS 1999).

4.7. Cálculo da Variável de Controle

A teoria que DUCLOS (1997) desenvolveu em sua tese permite que sejam criados gráficos de controle para processos não normais e possui alta aplicabilidade em diversas áreas de conhecimento, da mesma forma que a teoria clássica, desenvolvida por Shewhart, (MONTGOMERY 1999), utilizada a partir da década de 20 até os dias de hoje.

Os gráficos de controle são utilizados, basicamente, para assegurar que um produto, oriundo de uma linha de fabricação, por exemplo, possa ter algumas características bem definidas e com variações aceitáveis, ou seja, limitadas a determinadas faixas de valores. Da mesma forma, definimos que a alteração do comportamento de um dado fenômeno pode ser interpretado como sendo o momento em que uma determinada variável (estimador de localização) fica fora de determinados limites (torna-se “fora de controle”).

O fenômeno físico é interpretado como sendo o processo de fabricação de variáveis que serão controladas e que devem ser monitoradas durante todo o tempo. Ou seja, para um determinado fenômeno, as medidas realizadas durante o processo fornecem valores de certas variáveis. Estas variáveis são interpretadas como resultado de um terceiro processo: o de fabricação destas. Assim, pretende-se assegurar que estas variáveis possuam valores bem definidos, ou com pouca variabilidade, buscando, assim, assegurar a qualidade de todo o processo.

O primeiro passo do algoritmo proposto, portanto, consiste no cálculo dos estimadores de localização e dispersão para pequenas janelas de intervalos de tempo entre os pacotes de rede, obtidos durante o fenômeno, conforme descrito na Figura 4.6. A primeira etapa consiste na seleção de intervalos de valores, denominados de janelas, onde o algoritmo é aplicado.

No início desta pesquisa, (SANTOS 2007a, 2007b), o tamanho da janela era fixo e representava uma das variáveis do algoritmo. Foi visto, no capítulo 2, que a teoria de gráficos não normais não possui um cálculo de ARL bem definido, ficando para ser estipulado em cada caso particular, desta forma, não pode ser usado este valor para a determinação do tamanho da janela. Além disso, a existência de desvios e tendências, por exemplo, obriga que seja adotada uma técnica mais robusta.

Para exemplificar este problema, considere a Figura 4.7, onde são grafados os valores obtidos a partir de um determinado fenômeno físico. Por inspeção visual, percebe-se que a partir da abscissa de valor 300 há uma variação de valores na ordenada.

Realizando o cálculo do estimador de localização para as janelas [1-300] e [301-600], são obtidos os valores descritos na Tabela 4.3 e representados na Figura 4.8, pelas duas linhas horizontais. Assim, estas duas linhas centrais representam os valores médios do primeiro ([1-300]) e segundo ([301-600]) intervalos.

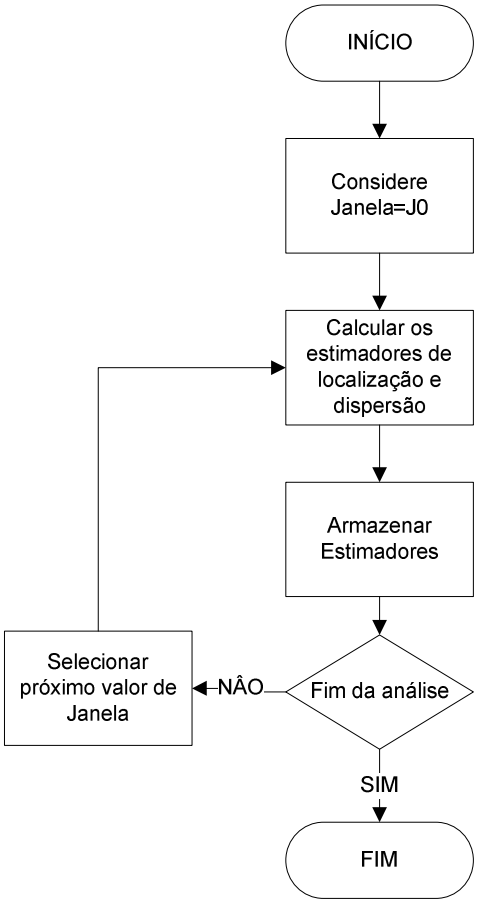

Figura 4. 6: Diagrama de blocos da primeira etapa do algoritmo;

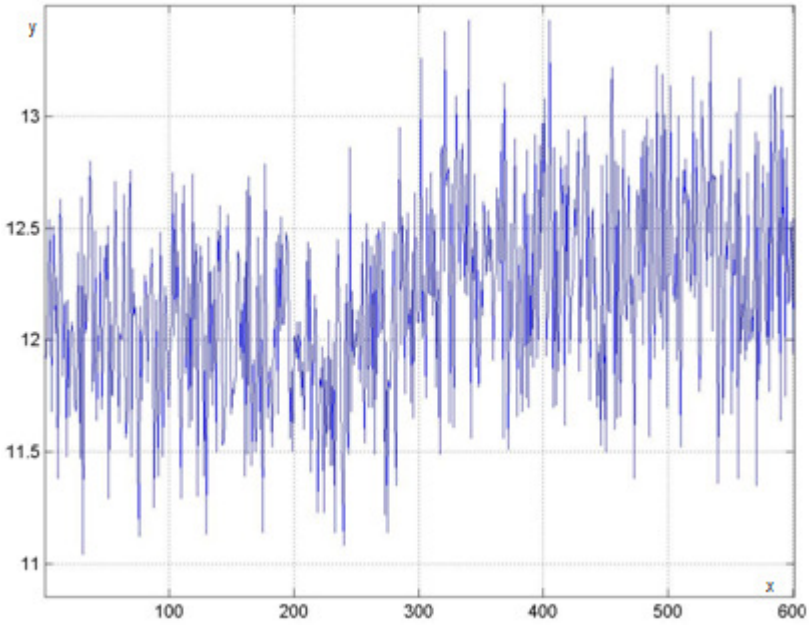

Figura 4. 7: Gráfico das medidas y de um fenômeno y=f(x);

Tabela 4.3: Valores dos estimadores de localização;

Intervalo	Estimador de Localização
[1-300]	12,0003
[301-600]	12,3602

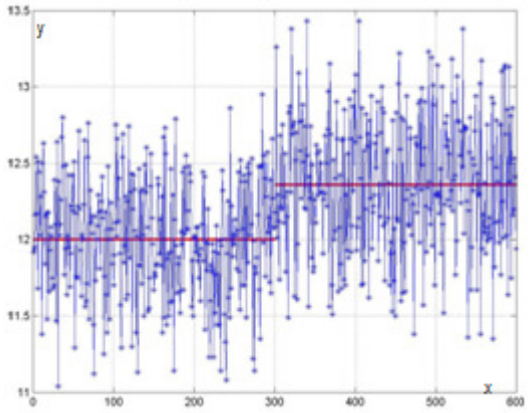

Figura 4. 8: Gráfico das medidas y de um fenômeno y=f(x) (representados pelas linhas verticais) e dos estimadores de localização (representados pelos patamares horizontais);

Porém, alterando os tamanhos das janelas para 200, seriam encontrados novos patamares, Figura 4.9 e Tabela 4.4:

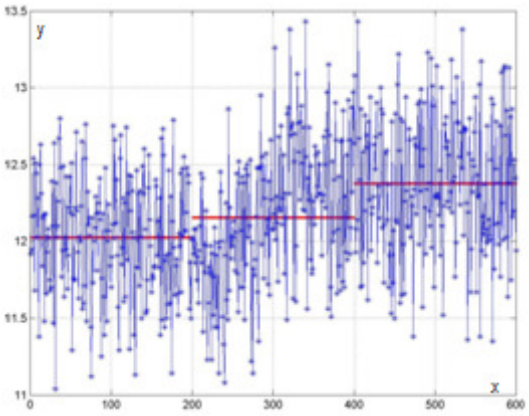

Figura 4. 9: Gráfico das medidas y de um fenômeno y=f(x) e dos estimadores de localização;

Tabela 4. 4: Valores dos estimadores de localização;

Intervalo	Estimador de Localização
[1-200]	12,0243
[201-400]	12,1537
[401-600]	12,3755

A comparação entre os valores dos estimadores de localização obtidos demonstra que pode haver ocultamento da mudança de comportamento no processo em função do tamanho da janela.

Na primeira etapa de desenvolvimento do algoritmo, foi estipulada uma janela fixa com o tamanho de 100 intervalos de tempo. Este fator foi escolhido sem nenhuma razão pré-definida.

4.8. Algoritmo de Detecção de Anomalias com Janelas Fixas

Na primeira parte desta pesquisa, com os valores de janela fixas (e convenientemente escolhidas) foram identificados ataques nas três semanas iniciais[21], de acordo com a Tabela 4.5 (SANTOS 2007b). Nesta tabela estão os resultados obtidos com uma janela fixa de 100 intervalos e com o fator multiplicativo do estimador de dispersão, que define os limites superior e inferior, *k*, variável de 1 até 10. O valor da constante *k* historicamente tem seu valor igual a três. Este mesmo valor foi utilizado por DUCLOS (2005), sem prévias definições. O valor de *k* igual a três reflete uma boa escolha, como pode ser visto na Tabela 4.5, onde nenhum dos ataques deixou de ser identificado.

O mesmo resultado é exibido na Figura 4.10 (SANTOS 2007b), onde é realizada uma comparação entre o gráfico com os ataques originais, presentes na base de dados do DARPA de 1999 e os demais identificados pelo algoritmo. Em cada gráfico deste, as linhas verticais exibem uma ocorrência de um tipo de ataque e as três semanas são identificadas pelo primeiro dia da semana:

[21] A base de dados disponível pelo DARPA é composta de cinco semanas, porém foram utilizadas apenas as três semanas iniciais, onde na primeira e na terceira não há ataques e na segunda há ataques conhecidos. (http://www.ll.mit.edu/mission/communications/ist/corpora/ideval).

- Primeira semana, sem ataque: 28/02/1999 – 06/03/1999;
- Segunda semana, com ataques identificados: 07/03/1999 – 13/03/1999; e
- Terceira semana, sem ataques: 14/03/1999 – 20/03/1999

Tabela 4. 5: Resultado Quantitativo (SANTOS 2007b);

k	Ataques Identificados	Falsos Negativos	Falsos Positivos	Total dos Pontos
1	29	0	166	195
2	29	0	165	194
3	29	0	159	188
4	27	2	116	143
5	16	13	54	70
6	12	17	37	49
7	12	17	32	44
8	11	18	29	40
9	6	23	25	31
10	4	25	24	28

As identificações realizadas pelo algoritmo, dependendo do valor de *k* selecionado é representado por uma linha vertical. As linhas verticais presentes nos gráficos identificados com os valores de *k=2, 4, 6, 8* e *10* na primeira e segunda semanas são falsos positivos. Conforme descrito anteriormente, o valor da janela escolhido tem importância fundamental para a eficiência do algoritmo. Devido a importância do tamanho da janela, da dinâmica do fenômeno e da falta de conhecimento prévio, adotou-se o cálculo do valor da janela através de sistemas *fuzzy*, visando assegurar que não haja ocultamento do comportamento do processo.

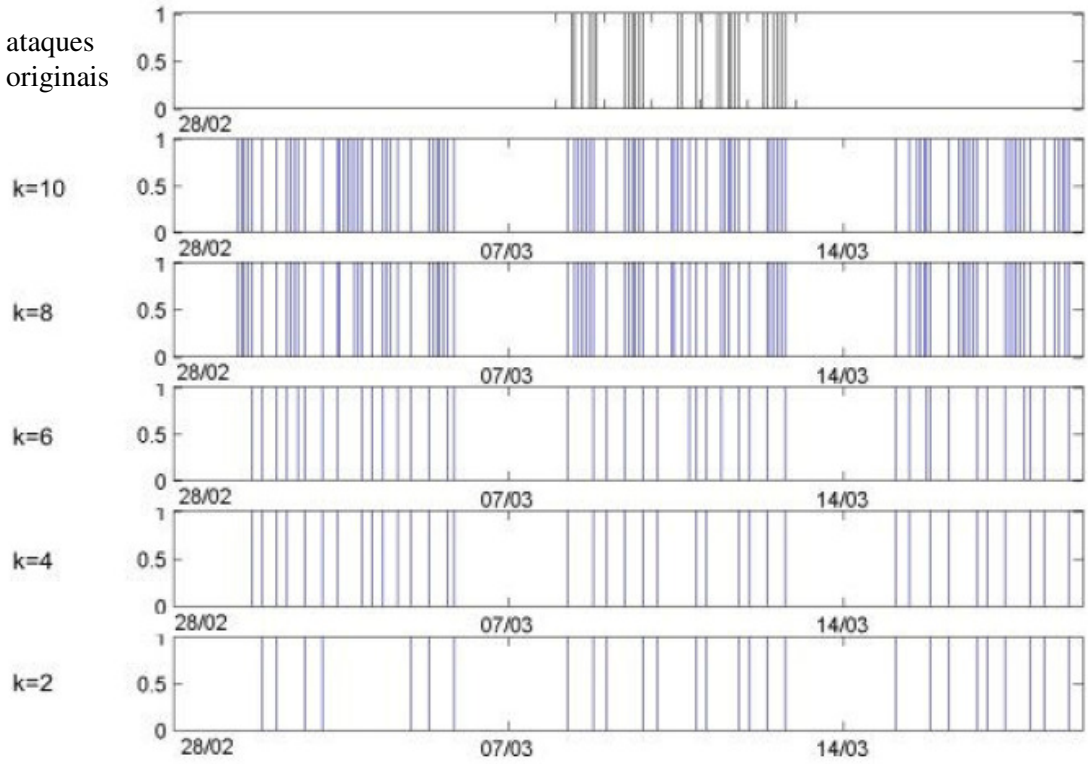

Figura 4.10: Resultado obtido com o tamanho de janela fixo (SANTOS 2007b);

4.9. Alteração Dinâmica do Tamanho da Janela.

O valor da janela a ser utilizada como domínio no algoritmo de Duclos, é fundamental e está diretamente relacionado com a qualidade do resultado. Desta forma, tornou-se necessária a aplicação de alguma técnica que possa alterar seu tamanho dinamicamente pelos valores apresentados.

A teoria de conjuntos *fuzzy* mostrou-se apropriada neste contexto e foi desenvolvido um modelo para monitorar e alterar o tamanho da janela. Este cálculo é realizado no bloco "Selecione o próximo valor de janela", que pode ser substituído pelo diagrama contido na Figura 3.3 do capítulo 3 de Sistemas *Fuzzy*.

Foram selecionadas como variáveis de entrada os valores dos estimadores de localização e dispersão e valores de saída o acréscimo/decréscimo a ser aplicado no tamanho da janela. As variáveis lingüísticas criadas foram ZERO (zero ou muito

próximo de zero), PEQUENO e GRANDE. Por não se saber, previamente, o que seria um estimador de localização zero, pequeno e grande, optou-se por criar um histórico com estes valores. Assim, a cada iteração do algoritmo, uma sequência de valores históricos é gerada. Desta forma, foi adicionado ao algoritmo a característica de memória, não presente na teoria de gráficos de controles não normais, da mesma forma que os de controle de Shewhart, porém importante na de CUSUM e EWMA.

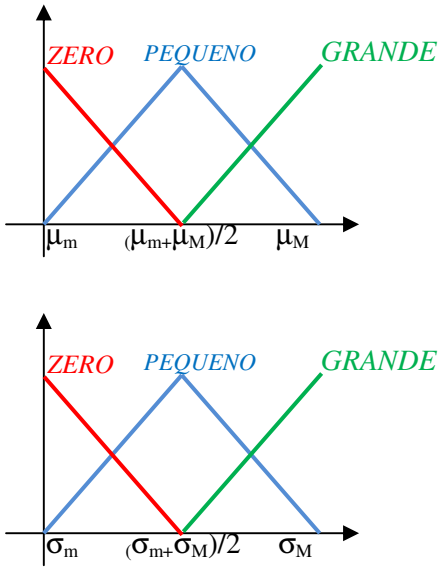

Figura 4.11: Funções de pertinência para os estimadores de localização e dispersão, respectivamente.

Para os estimadores de localização e dispersão foram elaboradas as funções de pertinência, Figura 4.11. Com valores mínimos e máximos[22] que definem o domínio das funções de pertinência são calculados através de dados históricos, conforme explicado anteriormente. Desta forma, a influência histórica do processo é representada através dos limites inferior e superior das funções de pertinência.

E para os valores de ampliação e redução das janelas, a serem usadas na desfuzificação, a função de pertinência apresentada na Figura 4.12.

[22] Representados na figura 4.7 através dos índices *m* e *M*.

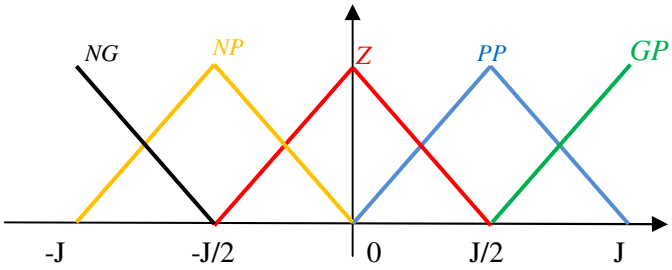

Figura 4. 12: Função de pertinência para o valor da janela.

Para finalizar o projeto do modelo fuzzy, foi elaborada a seguinte base de regras, baseado na experiência prévia do assunto:

Regra 1) Se os estimadores de localização e dispersão são próximos de zero, então considerar um grande aumento no valor da janela, uma vez que entende-se que neste caso não há problema acontecendo e que poder-se-ia aumentar o respectivo tamanho;

Regra 2) Se o estimador de localização é próximo de zero e o de dispersão é pequeno ou o estimador de localização é pequeno e o de dispersão é próximo de zero, então considerar um pequeno aumento no valor da janela. Neste caso, entende-se que há alguma variabilidade no processo, porém esta variabilidade não representa nenhum descontrole no processo;

Regra 3) Se os estimadores de localização e dispersão são muito grandes, então considerar uma significativa diminuição no valor da janela. Neste caso, percebe-se que os valores estão muito grandes e poderão "esconder" o fenômeno, portanto sendo necessário reduzir o respectivo tamanho;

Regra 4) Se o estimador de localização é muito grande e o de dispersão é pequeno ou próximo de zero, ou se o de localização é pequeno ou próximo de zero e o dispersão é muito grande, então considerar uma pequena diminuição no valor da janela. Neste caso são dois fenômenos distintos, porém resultando em um mesmo fenômeno: o de

um grande intervalo para a análise. Neste caso, é necessário diminuir o tamanho para melhorar a análise;

Regra 5) Se os estimadores de localização e dispersão forem pequenos, então considerar um pequeno aumento no valor da janela. Neste caso, pequenos tamanhos de janela tornam o algoritmo ineficiente.

A Tabela 4.6 apresenta o resumo destas regras.

Tabela 4. 6: Valores dos estimadores de localização;

	Estimadores	Dispersão		
		PEQUENO	MÉDIO	GRANDE
Localização	PEQUENO	Aumentar o tamanho da janela	Aumentar o tamanho da janela	Reduzir o tamanho da janela
	MÉDIO	Aumentar o tamanho da janela	Reduzir o tamanho da janela	Reduzir o tamanho da janela
	GRANDE	Reduzir o tamanho da janela	Reduzir o tamanho da janela	Reduzir o tamanho da janela

Assim, um modelo fuzzy foi acrescentado ao processo para determinar o tamanho de janelas mais adequado, considerando as características dos estimadores.

Desta forma, com os valores dos estimadores calculados em sucessivas janelas, podem-se construir dois vetores, um de estimadores de localização e outro de estimadores de dispersão que representam os dados do fenômeno físico.

4.10. Heurística de Identificação da Mudança de Comportamento.

Os valores dos estimadores calculados em cada janela são usados para verificar se há ou não problemas de mudança de comportamento. Para isso, é realizada uma comparação entre um intervalo *i* e o intervalo seguinte, *i+1*. No primeiro intervalo são calculados os limites de operação, dado por um intervalo de centro no valor do estimador de localização do intervalo *i* e vizinhança igual ao triplo do desvio padrão do mesmo intervalo, ou seja,

$$i = [\mu_i - 3.\sigma_i, \mu_i + 3.\sigma_i] \quad (4.1)$$

O fator triplo foi escolhido devido ao trabalho de pesquisa realizado por DUCLOS (1997) e comprovada a eficiência prática em (SANTOS 2007b). Deve-se salientar a coincidência entre este valor e o clássico obtido por Shewhart na teoria clássica (MONTGOMERY 1999).

Se o estimador de localização do intervalo for maior que o limite superior de controle, Equação 4.2, considerar-se-á que há mudança de comportamento. De forma similar, se o valor for menor, que o limite inferior de controle, Equação 4.3. A interpretação da mudança de comportamento está relacionada com o tipo de fenômeno que esteja ocorrendo. Por exemplo, no caso de ataque *DDoS*, a Equação 4.3 pode identificar o início do ataque e Equação 4.2 pode identificar o final do ataque.

$$\mu_i > \mu_{i-1} + 3.\sigma_{i-1} \quad (4.2)$$

$$\mu_i < \mu_{i-1} - 3.\sigma_{i-1}, com\ \mu_i > 0 \quad (4.3)$$

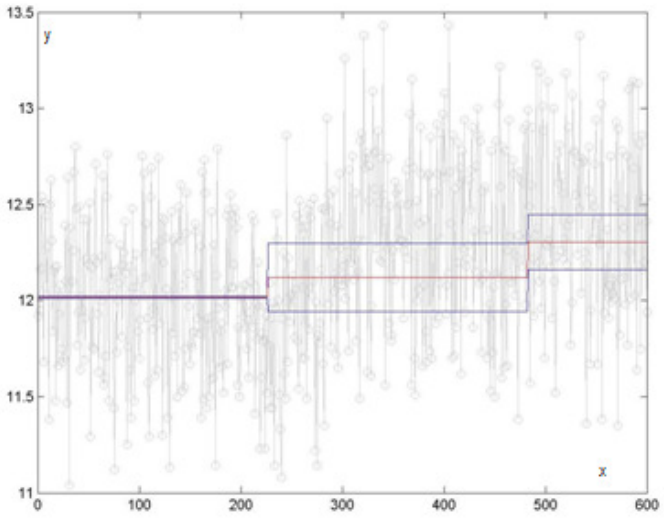

Figura 4. 13: Exemplo de detecção de mudança brusca de comportamento detectado, do fenômeno *y=f(x)*;

No exemplo da Figura 4.13 o fenômeno físico é representado pelas linhas em cinza. Através do componente fuzzy do algoritmo, três intervalos são representados: [1-222], [223-483] e [484-600]. A primeira informação que deve ser visualizada é o tamanho desigual dos intervalos, um resultado do funcionamento da ação do modelo *fuzzy*.

Nessa figura, os valores do estimador de localização estão em vermelho e os limites superior e inferior em azul. No primeiro intervalo, o estimador de dispersão tem seu valor bem pequeno e com isto, o estimador de localização do intervalo seguinte [223-483] tem seu valor maior que o limite superior do intervalo anterior, [1-222]. Desta forma, o algoritmo identifica uma primeira mudança de comportamento. O mesmo não ocorre com o intervalo [484-600], uma vez que neste caso, o valor do estimador de localização deste terceiro intervalo é menor que o limite superior do intervalo anterior, [223-483], ou seja,

$$\mu_{(484-600)} < \mu_{(226-483)} + 3.\sigma_{(226-483)} \tag{4.4}$$

4.11. Algoritmo de Detecção de Anomalias com Janelas Adaptativas.

Do exposto nos itens anteriores, foi elaborado o algoritmo descrito na Figura 4.14. O algoritmo tem início com um valor inicial de tamanho da janela (Figura 4.14 item 1).

Com esta janela, são calculados valores iniciais dos estimadores de localização e dispersão (Figura 4.14 item 2). Estes estimadores são armazenados (Figura 4.14 item 3) e a partir destes, através do controlador fuzzy, é calculado a variação do tamanho da janela (Figura 4.14 itens 8, 9, 10 e 11). Na primeira iteração do algoritmo, os testes de mudança de comportamento e as comparações com intervalos anteriores não são executadas.

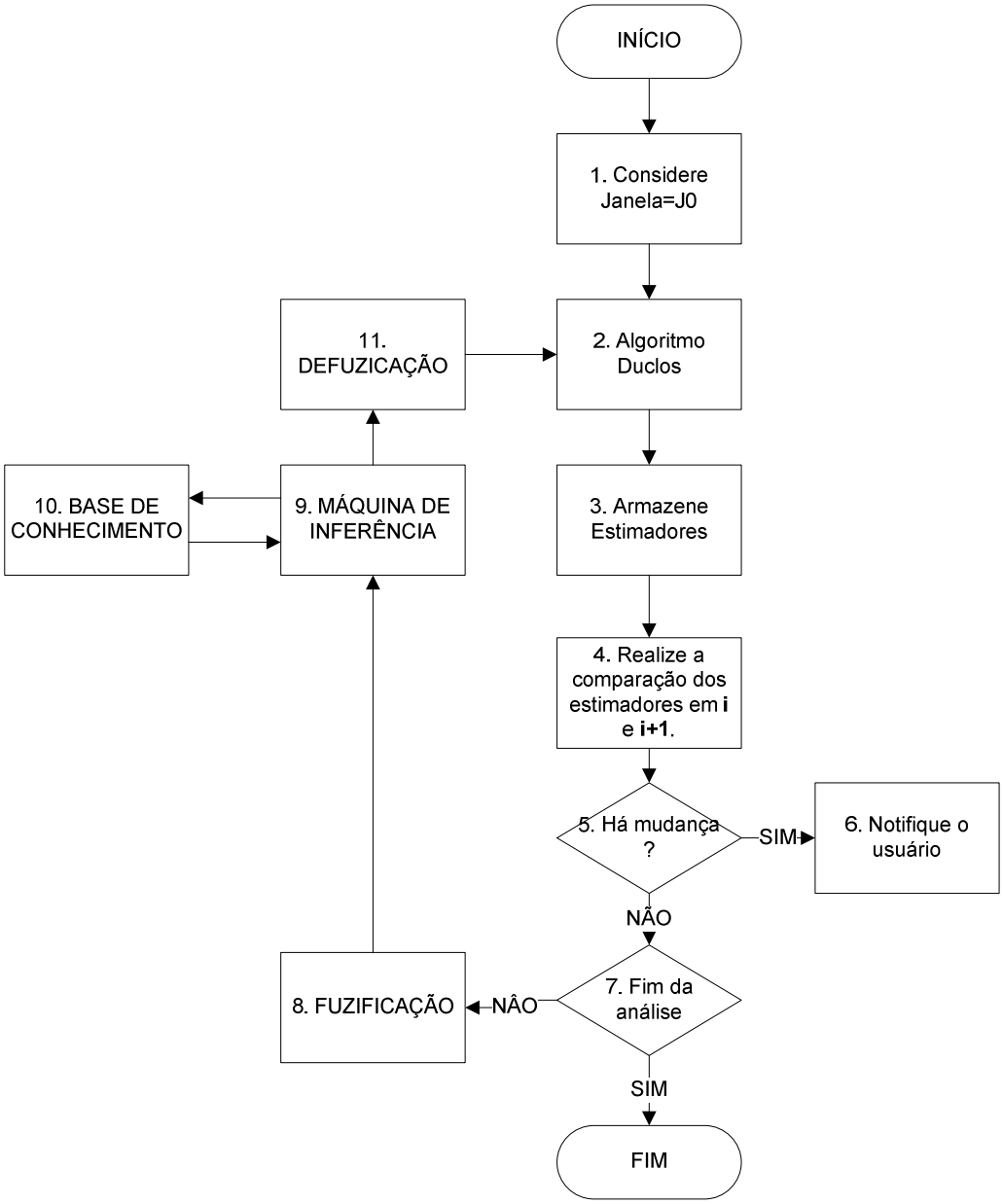

Figura 4.14: Diagrama de blocos do Algoritmo de Detecção de Anomalias com Janelas Adaptativa.

Uma nova janela é capturada pelo sistema, e desta, novos valores de estimadores são calculados (Figura 4.14 item 2) e armazenados com os anteriores (Figura 4.14 item 3). Caso haja discrepância entre estes valores (Equações 4.3 e 4.4) o usuário será notificado e o algoritmo acompanhará esta situação (Figura 4.14 itens 5, 6), esperando que a situação anterior retorne. Caso contrário, é calculada a variação do valor da janela (Figura 4.14 itens 8, 9, 10 e 11), e assim sucessivamente, até o processamento de todos os dados de entrada.

Capítulo 5

Testes e Resultados

5.1. Experimentos / DARPA

Na primeira parte desta pesquisa foi desenvolvido um algoritmo (Algoritmo de Detecção de Anomalias com Janelas Fixas) onde resultados significativos foram obtidos (SANTOS 2007a, 2007b), porém, ainda, com alguns problemas já discutidos na seção 4.7. A partir destes problemas, foi desenvolvido o Algoritmo de Detecção de Anomalias com Janelas Adaptativas, cujos resultados são expostos na próxima seção. Neste algoritmo foram submetidos dois períodos, um livre de ataques (seção 5.1.1) e outro com ataques conhecidos (seção 5.1.2), para verificar sua eficácia.

5.1.1. Experimento 1: Segunda-Feira da Primeira Semana

Neste dia não há ataques existentes, portanto quaisquer identificações ocorridas são consideradas falsos positivos. Das 22h de dados disponíveis, foram identificados 228 intervalos, correspondendo a um total de 1h 21min 13s, ou seja, 6,15% de falsos positivos. A Figura 5.1 exibe os períodos de identificação dos falsos positivos. Neste gráfico, os períodos (ou janelas) de identificação positiva são representados com a linha vertical no instante em que ocorre a identificação.

Os demais períodos deste dia correspondem a identificações negativas, ou seja, períodos que foram identificados como não havendo ataques.

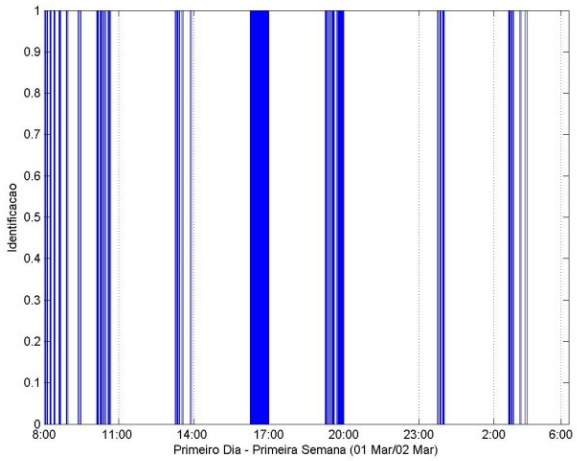

Figura 5. 1: Ataques identificados (Falsos Positivos)
no Primeiro Dia da Primeira Semana

5.1.2. Experimento 2: Segunda, Quarta e Quinta Semanas de Ataques

Foi submetido ao algoritmo a segunda, quarta e quinta semana de ataques. Nestes períodos, somente os ataques com a característica de sobrecarga da rede foram objetivados para serem detectados. Nesta categoria se enquandram os seguintes ataques: apache2, mailbomb, smurf, warezmaster e udpstorm[23].

Outros ataques puderam ser detectados, devido a resultados indiretos ocasionados destes. É o caso dos demais ataques DoS existentes na base, exceto teardrop e dosnuke, que usam brechas na implementação do TCP/IP para proferirem seus ataques. Na Tabela 5.1 estão descritos os ataques que foram foco da identificação e na Tabela 5.2 os que foram identificados, através do resultado do ataque.

Através da análise destas duas tabelas, pode-se perceber que os ataques que foram objetivados na pesquisa obtiveram uma margem significativa de identificações positivas, citando-se em particular que o ataque warezmaster foi identificado em 100% das ocasições (4 instâncias). O mesmo não ocorreu com o ataque udpstorm, cuja única instância não pode ser identificada. O mesmo não ocorreu com outros ataques, que embora não fossem foco desta pesquisa, puderam ser identificados, na grande maioria em 50% das ocasiões.

[23] No Apêndice B consta a descrição de cada ataque.

Tabela 5.1: Ataques objetivados neste estudo.

Ataque	Identificação		Falso Negativo	
apache2	2	66%	1	33%
Mailbomb	2	33%	4	66%
Smurf	3	60%	2	40%
Udpstorm	0	0%	1	100%
Warezmaster	4	100%	0	0%

Tabela 5.2. Ataques que não foram foco do estudo.

Ataque	Identificação		Falso Negativo	
Back	3	50%	3	50%
pod	3	50%	3	50%
land	2	50%	2	50%
crashiis	1	9%	10	91%
synflood	1	17%	5	83%
processtable	1	25%	3	75%
arppoison	2	40%	3	60%
dosnuke	0	0%	4	80%
syslogd	3	75%	1	25%
selfping	2	100%	0	0%
tcpreset	2	66%	1	33%
Teardrop	0	0%	2	100%

5.1.3. Comparação dos Resultados

De acordo com LIPPMANN (2000), sete grupos de pesquisa participaram da avaliação realizada pelo DARPA nesta base de dados: (NEUMANN 1999), (SCHWARTZBARD 1999), (SEKAR 1999), (VIGNA 1999), (JAJODIA 2000), (TYSON 2000) e (VIGNA 2000). Nestes, a taxa de falsos alarmes foi baixa (menos que dez por dia). Considerando apenas os ataques de negação de serviço, os melhores resultados foram obtidos para ataques conhecidos e os piores resultados foram obtidos com os ataques desconhecidos.

Destes ataques, dois não foram percebidos por nenhuma técnica, foi o caso dos ataques DoS *selfping*, *warezclient.* Dois obtiveram identificações abaixo da média, porém com alguma identificação, foi o caso do arppoisson e tcpreset. O ataque dosnuke foi identificado e duas das quatro instâncias que existiam na base, ficando com um resultado de 50%. Todos os demais ataques (back, pod, land, mailbomb, crashiis, synflood, smurf, processtable, apache2, syslogd, teardrop e udpstorm) foram identificados em mais da metade das vezes, porém nenhum dos ataques acima foi identificado todas as vezes, ou seja, 100%.

Para que fosse possível realizar uma comparação com outras técnicas utilizadas com a mesma base de dados, considerou-se que as sete pesquisas acima descritas formassem um resultado único, sendo então este resultado comparado ao resultado desta pesquisa. A partir desta comparação, foi elaborada a Tabela 5.3. Para tanto, foi realizado a seguinte interpretação: se esta pesquisa obtivesse mais identificações de um determinado ataque, que o melhor resultado dos sete anteriores, a linha da tabela seria marcada com a cor verde. No caso da quantidade de identificações for a mesma, a cor amarela e no caso de menos identificações, vermelha.

Tabela 5.3: Comparativo de resultados

Ataques	Instâncias	Melhor resultado[24]		Este Trabalho	
		quantidade	eficiência	quantidade	eficiência
Back	6	3 ou +	> 50%	3	50%
Pod	6	3 ou +	> 50%	3	50%
Land	4	2 ou +	> 50%	2	50%
Mailbomb	6	3 ou +	> 50%	2	33%
Crashiis	11	5 ou +	> 50%	1	9%
Synflood	6	3 ou +	> 50%	1	17%
Smurf	5	2 ou +	> 50%	3	60%
Processtable	4	2 ou +	> 50%	1	25%
Warezmaster	4	0	0%	4	100%
Arppoisson	5	1	20%	2	40%
Dosnuke	4	2	50%	0	0%
Apache2	3	1 ou +	> 50%	2	66%
Syslogd	4	2 ou +	> 50%	3	75%
Selfping	2	0	0%	2	100%
Tcpreset	3	1	33%	2	67%
Teardrop	2	1 ou +	> 50%	0	0%
Udpstorm	1	1 ou +	> 50%	0	0%

Assim, dos dezessete ataques, esta pesquisa identificou quatro ataques de maneira mais eficiente que as sete pesquisas juntas, cinco ataques com a mesma eficiência e oito com menos eficiência. Considerando que estas sete pesquisas correspondem a diferentes técnicas, observa-se que a eficiência desta pesquisa, na média, torna-se superior às sete pesquisas juntas. Este fato é resultado, principalmente, da natureza do algoritmo desenvolvido quando comparado aos demais. O principal foco dos demais algoritmos está relacionado com o conhecimento prévio do ataque, ou seja, com a forma de operação/assinatura do ataque, diferentemente desta pesquisa que objetivou a identificação da mudança de comportamento. Assim, percebe-se a aplicabilidade desta pesquisa em redes reais, conforme objetivos anteriormente previstos.

Além disso, a pouca informação fornecida ao cálculo evidenciou esta possibilidade de identificar outros tipos de ataques não conhecidos, uma vez que,

[24] Dentre todos os oito trabalhos já enunciados.

mesmo que um novo protocolo seja implementado e sobre este novo protocolo um ataque for criado, a técnica aqui desenvolvida permite que a identificação seja realizada, tornando-se necessário, se for o caso, alterar a base de conhecimento implementada nas regras. Por outro lado, como as informações utilizadas para o desenvolvimento do algoritmo foram estas regras e o intervalo de tempo entre os pacotes da rede, este algoritmo permite que seja empregado em outros tipos de problemas. Visando comprovar isto, foi realizado o teste deste algoritmo em um problema de vazão de redes de distribuição de água, exposto na seção 5.2.

5.2. Experimentos / Vazão de redes de distribuição

As redes de distribuição de água fazem parte das principais infra-estruturas para as cidades atuais. Problemas que resultem em perda de qualidade ou mesmo interrupção do fornecimento causam problemas sérios à população.

Com o crescimento da população e com a crescente e necessária busca da preservação ambiental, regulamentações mais rígidas tornaram-se necessárias para que houvesse um aumento da eficiência e maximização destas redes, (THORNTON 2002).

Assim, diversas iniciativas foram realizadas, dentre elas, a pesquisa para aumentar a eficiência do fornecimento através de técnicas simples, como o monitoramento em diversas partes da rede e técnicas mais complexas, como a busca por modelos matemáticos mais fiéis às situações encontradas na prática.

Neste estudo, com o uso do algoritmo desenvolvido no capítulo 4, foi realizado o monitoramento da vazão em um determinado ponto da rede. Esta vazão, quando modelada pelos métodos tradicionais, resulta em uma modelagem complexa e não exata.

Uma queda do valor da vazão pode ser exemplificada, em casos reais, como sendo o furto de água nos sistemas de distribuição, tão comum nas mídias, como noticiado no Correio Braziliense (VELEDA 2008), e/ou a interligação de redes clandestinas, como noticiado em no canal web G1 (2007), fato este já alertado por WALLACE (1987), como um dos mais rotineiros encontrados nas principais

metrópoles mundiais. A Figura 5.2 exibe o balanço hídrico em um sistema de distribuição; as perdas que ocorrem neste tipo de sistema estão marcadas com retângulos vermelhos.

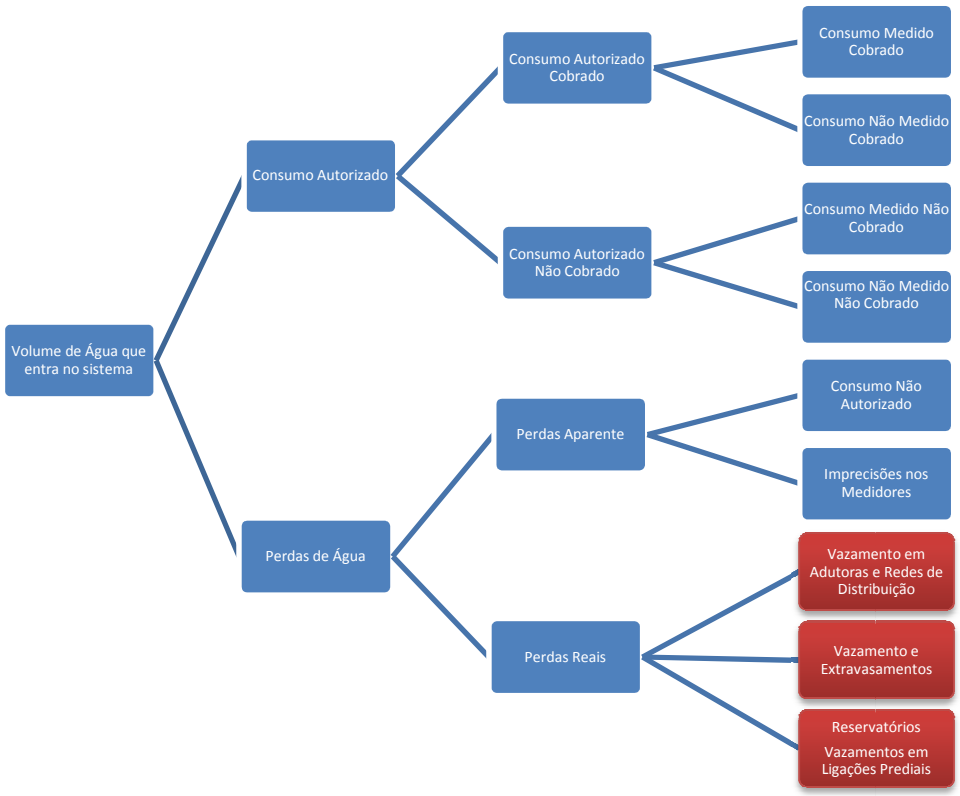

Figura 5.2. Balanço Hídrico (GALVÃO 2007)

Algumas pesquisas sobre este assunto desenvolveram simuladores das redes de distribuição como SILVA (1999) que consideram as perdas do sistema hídrico e possibilitam o estudo da distribuição da demanda espacial e temporal. Em (SOARES 2003) é desenvolvido um modelo que leva em consideração as perdas por vazamento e a dependência das demandas com a pressão. Outros objetivam não modelar as redes, mas aumentar a eficiência das metodologias de calibração dos modelos das redes, (SILVA 2002a), (SILVA 2002b).

Outro problema comum de ser encontrado é a contaminação das redes de distribuição. Além de aumentar os custos operacionais necessários para a descontaminação, dependendo do volume contaminado, ainda põe em risco a tubulação

existente. Este aumento também pode ser causado pela deposição do próprio material que compõe as tubulações das redes, como os materiais de revestimento e outras substâncias químicas usadas na limpeza da água, muito embora que nesta situação, o aumento da vazão possa ser discreto.

Uma possível forma de identificar a presença de contaminantes no sistema de distribuição é monitorar pontos do sistema e verificar se há aumento da vazão. Para assegurar a qualidade da água, diversas abordagens têm sido realizadas. CLARK (1993) verificou que em um sistema de distribuição contaminado com cloro, há variação na concentração destes resíduos tanto espacialmente quanto temporalmente. Por outro lado, também foi verificado que a armazenagem da mistura em tanques por um grande período de tempo diminui, consideravelmente, ou elimina totalmente a concentração do poluente.

COSTA (2002) apresenta uma metodologia de coleta e amostragem com o objetivo de assegurar a conformidade da qualidade da água com os padrões determinados pela legislação vigente, com base de critérios estatísticos, visando maximizar a capacidade de detecção de contaminadores. FORMIGA (2007) realiza um estudo comparativo entre os métodos de resolução do sistema de equações lineares quando associado ao modelo de redes de distribuição.

Neste estudo de caso, o algoritmo foi utilizado para monitorar os valores de vazão. Para isto, realiza-se a modelagem da rede de forma detalhada, onde são considerados valores pontuais dentro do sistema de distribuição. Esta forma de modelar as redes de distribuição contrapõe-se à modelagem global, conhecida como modelagem estratégica, (FUERTES 1999), onde os valores pontuais não são considerados.

Este tipo de modelagem é normalmente utilizada para verificar o comportamento da rede, e desta forma, torna-se necessário realizar a comparação com valores padronizados do sistema, (MARTÍNEZ-SOLANO 2008). Estes valores são função diretamente do número de nós do sistema e do consumo médio dos usuários, que é interpretada como uma das variáveis do sistema. Há resultados, (LINGIREDDY 1998),

(BURN 2002) e (JANKOVIĆ-NIŠIĆ 2004) que indicam que este consumo está diretamente relacionado com a razão entre o pico da vazão e o seu valor médio. Outras abordagens realizadas, (BUTLER 1995), (BUCHBERGER 1996), (GARCÍA 2004) e (BUCHBERGER 1995), obtém este coeficiente através da análise do comportamento estatístico dos usuários.

5.2.1. Base de Dados

A base de dados utilizada foi obtida a partir de um modelo elaborado de acordo com a Figura 5.3.

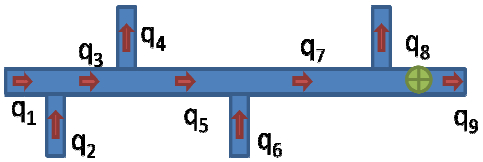

Figura 5.3. Sistema hidráulico simulado;

Neste sistema, são pressupostos um reservatório principal, cuja vazão é q_1, dois reservatórios secundários, cujas vazões são, respectivamente, q_2 e q_6, e dois sumidouros q_4 e q_8. A vazão q_9 é, portanto, interpretada como variável de controle de todo o sistema, e será a partir desta variável que se pretende monitorar se há problemas na rede de distribuição, como diminuição substanciais dos valores das vazões q_1, q_2, q_4, q_6 e q_8.

Estas vazões formam um sistema de equações que é composta pela equação da continuidade em cada nó da rede de distribuição e de conservação de massa para cada trecho, sendo suposto que não há perda de massa. Desta forma, podem-se construir as Equações 5.1, 5.2, 5.3 e 5.4.

$$q_1 + q_2 = q_3 \quad (5.1)$$

$$q_3 - q_4 = q_5 \quad (5.2)$$

$$q_5 + q_6 = q_7 \quad (5.3)$$

$$q_7 - q_8 = q_9 \quad (5.4)$$

Para a realização desta simulação, os valores foram gerados a partir da hipótese de que estas simulações sofrem variações de acordo com a distribuição normal, equação 6.5, onde σ=*1,25* e os valores das vazões médias são dadas de acordo com a Tabela 5.4.

$$q_i = \overline{q_i} + \sigma U(0,1), \qquad i = 1,2,4,6,8. \tag{5.5}$$

Tabela 5.4: Valores de vazões médias usadas na simulação;

i (Q_i)	1	2	4	6	8
$\bar{Q}$	10	5	3	2	1

5.2.2. Experimento 1: Perturbação Passageira em t=300, Δt=25 e Definitiva em t=400.

Nesta primeira base de dados, foram gerados, seguindo a distribuição normal (Tabela 5.4) e de acordo com o sistema de equações formada pelas Equações 5.1, 5.2, 5.3 e 5.4, 700 valores de vazão na simulação do modelo exibido na Figura 5.2. Considere que estes valores de vazão estão associados a instantes de tempo iniciados em t_1=1 e finalizados em t_{700}=700. Nesta base de dados, foram gerados dois aumentos significativos da vazão, interpretados como perturbações na rede de distribuição: uma temporária, iniciada no instante 300 e finalizada no instante 325; e outra iniciada no instante 400 e que perdurou até o instante final (700). O gráfico do valor da vazão pelo instante de tempo é exibido na Figura 5.4.

Para esta situação, o algoritmo (Figura 4.14) foi utilizado com a combinação de valores de entrada descritos na Tabela 5.5. Isto foi realizado para comparar o grau de dependência do algoritmo às suas variáveis de entrada. A quantidade de valores de vazão amostrados em cada janela foi considerada como sendo sempre o dobro da quantidade de valores existentes dentro da janela. Isto foi realizado sem nenhuma

suposição inicial, visando apenas simplificar a escolha de valores para serem usados no algoritmo.

Tabela 5.5: Valores de entrada do algoritmo para a situação 1;

Medidas	Valores das medidas usadas
Nº Repetições *Bootstrap* (B)	20, 100, 200, 400
Tamanho da Janela (J)	10, 20

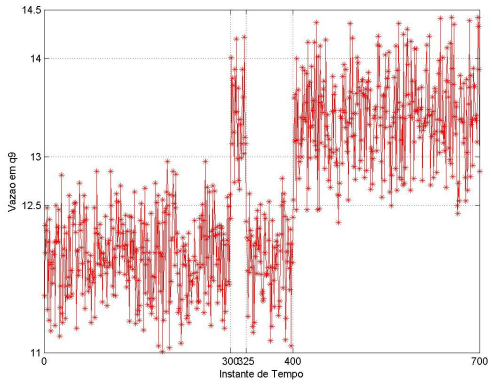

Figura 5.4: Gráfico vazão por instante de tempo, para a situação 1.

Desta forma, foram elaborados oito cenários diferentes nesta situação. Os instantes positivamente identificados, os falsos positivos e negativos estão descritos na Tabela 5.6.

Tabela 5.6: Resultado da Situação 1;

Cenário		Ident.	Falso Positivo	Falso Negativo	OBS
J	B				
10	20	2	1	0(1)	(1) Entre os dos períodos de variação há um período de normalidade que não foi identificado, muito embora esteja contido dentro da janela de análise. (2) O Nº de repetições de bootstrap é diretamente proporcional à precisão do algoritmo, que por sua vez altera, sensivelmente, o valor da janela.
10	100	2	3	0(2)	
10	200	2	1	0(1)	
10	400	2	4	0	
20	20	2	1	0	
20	100	2	1	0(1)	
20	200	2	1	0(1)	
20	400	2	1	0(1)	

5.2.3. Experimento 2: Sem Nenhuma Perturbação.

Nesta situação, não há nenhuma perturbação no sistema, e está representado na Figura 5.5 e na Tabela 5.7.

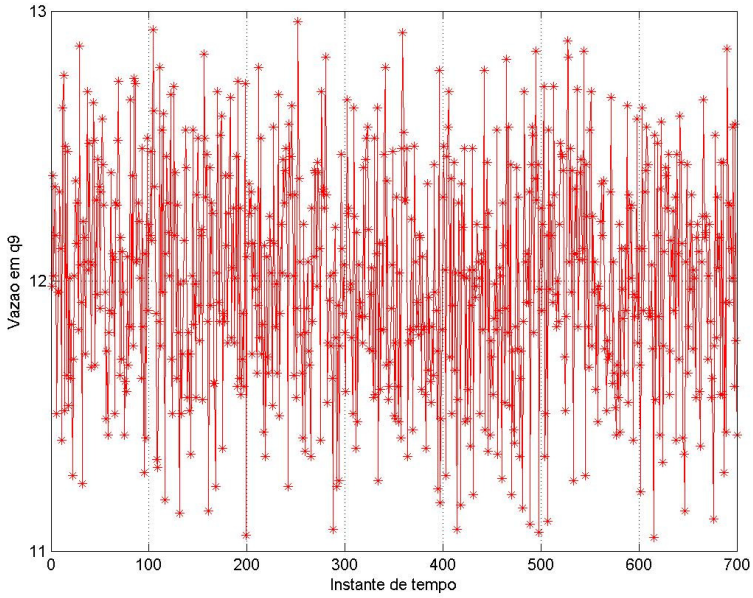

Figura 5.5: Gráfico vazão por instante de tempo, para a situação 2.

Tabela 5.7: Resultado da Situação 2;

Cenário		Ident.	Falso Positivo	Falso Negativo
J	B			
10	20	0	1	0
10	100	0	1	0
10	200	0	1	0
10	400	0	2	0
20	20	0	1	0
20	100	0	2	0
20	200	0	1	0
20	400	0	1	0

5.2.4. Análise de Resultados

Embora o presente estudo de caso tenha sido realizado sobre uma base de dados que foi obtida através de simulação, os resultados encontrados indicam que o algoritmo desenvolvido pode ser utilizado, para indicar uma alteração do sistema de rede de distribuição. Ainda há uma quantidade considerável de falsos positivos e negativos, o que indica que, provavelmente, este algoritmo, neste caso em particular, deva ser aplicado com outra variável de controle.

Capítulo 6

Conclusão

Historicamente as distribuições de processos reais são aproximadas para distribuições normais. Embora esta prática seja realizada desde a década de 20, matematicamente esta aproximação não é válida. Uma das principais razões para que esta aproximação possa ser feita, é o aproveitamento de ferramentas utilizadas em distribuições normais em processos reais, como é o caso dos gráficos de controle. Um exemplo é o trabalho desenvolvido por Shewhart, que utilizou gráficos de controle para monitorar a qualidade de produtos desenvolvidos de uma linha de produção nos Laboratórios Bell.

Através dos gráficos de controle é possível identificar mudanças de comportamento de um processo que siga a distribuição normal, e em algumas vezes, processos não normais. Porém, não é possível identificar o elemento causador da perda de controle.

DUCLOS (1997) desenvolveu um gráfico de controle direcionado para processos não normais. Neste gráfico, através de um processo de re-amostragem com repetição do conjunto amostral, métodos de *bootstrap*, é possível calcular estimadores de localização e dispersão, que são interpretados como média e desvio padrão, no sentido de Taguchi.

O primeiro objetivo desta tese era monitorar uma rede de computadores com relação à ataques, como os de negação de serviço (DoS). DoS são ataques que indisponibilizam um servidor através da exploração de vulnerabilidades existentes e/ou através do aumento excessivo da demanda ofertada pelo servidor. As defesas cibernéticas existentes para ataques deste tipo estão embasadas em três metodologias

básicas: identificação de assinaturas de ataques, mudança de comportamento ou híbrido, que engloba os dos tipos anteriores. Defesas do primeiro tipo possuem alto grau de eficiência (baixa taxa de falsos positivos), porém torna-se necessário conhecer previamente o ataque. As do segundo tipo, por outro lado, embora tenham alta taxa de falsos positivos, podem detectar ataques ainda não conhecidos. Nesta tese foi desenvolvido o Algoritmo de Detecção de Anomalias com Janelas Adaptativas, que pode ser enquadrado nesta segunda categoria.

Cada parque computacional, presente em quaisquer organizações, possui características particulares que o diferenciam, quiça o identificam, perante os demais. Por esta razão, criar um modelo para um parque computacional torna-se uma tarefa não trivial e não aplicável em outros lugares. Além disso, o próprio comportamento da rede de computadores é função do tempo, uma vez que o desenvolvimento de aplicativos que utilizem a Internet está crescendo todos os dias.

A metodologia proposta nesta tese procura verificar a alteração do comportamento, através da comparação de dois intervalos consecutivos. É pressuposto que o comportamento do tráfego da rede não se altere drasticamente, a menos que um ataque de negação de serviço esteja acontecendo. Desta forma, é realizado o cálculo dos estimadores de localização e dispersão em intervalos consecutivos e é realizada a verificação se o estimador de localização de um intervalo *i* encontra-se nos limites do intervalo anterior *i-1*. Assim, torna-se desnecessário armazenar períodos de referência, em tese, sem ataque para comparações futuras. Além disso, a aplicabilidade do algoritmo torna-se mais próximo às situações reais, uma vez que períodos livres de ataques não são assegurados em situações reais.

Esta pesquisa foi realizada, basicamente, em duas etapas. Na primeira etapa, os estimadores de localização e dispersão foram calculados em intervalos consecutivos e era realizada a verificação se o estimador de localização subseqüente estava dentro dos limites previamente estipulados. Caso o valor do estimador de localização estivesse acima ou abaixo dos limites, uma sinalização de ataque era emitida. Desta forma, as variáveis relacionadas a este algoritmo eram o tamanho da janela e o fator multiplicativo dos estimadores de dispersão (*k*) que formavam os limites inferior e superior de controle. Com valores convenientes de janela e *k*, foram obtidos resultados

importantes, (SANTOS 2007a) e (SANTOS 2007b), que comprovaram que o algoritmo é eficiente quando usado na base de dados selecionada. Assim, verifica-se que todos os 29 (vinte e nove) ataques existentes no período de referência eram detectados, muito embora possuísse uma alta taxa de falsos positivos, característico deste tipo de defesa cibernética.

Todavia, os tamanhos das janelas deveriam ter seus valores escolhidos de forma bem criteriosa, e uma seleção incorreta de valores poderia levar a interpretações erradas, conforme descrito no capítulo 4, seção 4.7, uma vez que poderia esconder mudanças de comportamento. Assim, sistema fuzzy foi acrescentado ao algoritmo para, a partir dos valores dos estimadores de localização e dispersão, que fosse determinada a variação adequada ao tamanho dos intervalos. Este variação baseia-se em todos os valores dos estimadores existentes, para calcular o percentual de aumento/diminuição da janela, caracterizando assim, um aprendizado do processo. Assim, o conhecimento do especialista é representado no algoritmo, através da base de conhecimento.

Com o uso deste aprendizado para o cálculo da janela, o algoritmo tornou-se menos dependente do usuário, o que permite uma independência da "pessoa que está aplicando". O conhecimento do especialista é aplicado diretamente base de conhecimento e não na escolha adequada do tamanho da janela. Com esta alteração o algoritmo apresentou resultados bem importantes. Na seção 5.1.3, capítulo 5, pode-se constatar que, os melhores resultados obtidos por pelas pesquisas que exploraram esta situação, detectam 7 (sete) dos dezessete (17) ataques, enquanto que esta pesquisa detecta 10 (dez) destes, refletindo assim um resultado melhor. Além disso, pelo item 5.1.1, percebe-se que do total de 22h de informações disponíveis, este algoritmo obteve uma eficiência de mais de 93%, ou seja, apenas pouco mais de 6% de falsos positivos foram emitidos.

No estudo de caso da vazão, nas duas situações, itens 5.2.1 e 5.2.2, os resultados obtidos demonstram que o algoritmo é eficaz na identificação dos problemas. As duas perturbações foram identificadas em todos os casos, com nenhum falso negativo e com uma quantidade significativa de falsos positivos, sendo este o ponto negativo da aplicabilidade do algoritmo neste caso. No segundo caso, houve problemas de identificação de falsos positivos, porém nenhum falso negativo.

Portanto, de forma geral, a partir da análise dos estudos de casos anteriores, pode-se concluir que, embora o algoritmo não seja suficiente para detectar e identificar os problemas com precisão, o mesmo possui algumas características que podem ser agregadas em outros métodos semelhantes, visando aumentar a eficiência. Assim, este algoritmo pode ser aplicado como um primeiro filtro, para uma análise mais aprofundada nos casos sinalizados por este.

Através das análises de resultados, percebe-se que ainda há oportunidades de melhoria no algoritmo. A primeira é a incorporação de critérios bayesianos de informação (BIC) na seleção de períodos de anomalia. Esta técnica (CHEN 93) no reconhecimento de interlocutores e apresenta a mesma situação dos problemas aqui encontrados. A característica básica deste método é a verificação da mudança de parâmetros em períodos (janelas) de uma base de dados.

No que se refere à metodologia geral empregada nesta pesquisa, o conceito de comitês, conforme empregado em MUNIZ (08), é interessante e poderá agregar mais funcionalidades, visto que este algoritmo ainda não é suficiente para a identificação e deteção do problema, mas sim a sinalização do que está ocorrendo.

Referências Bibliográficas

ALVES, J. R., NILTON, ASSIS, J. T., 2003, "Modelagem de tráfego Internet e aplicações", *VI Encontro de Modelagem Computacional*, Nova Friburgo, RJ, Dez.

ALVES, C. C., 2003 *Gráficos de controle CUSUM: Um enfoque dinâmico para análise estatística de processos, D*issertação de mestrado, UFSC, Santa Catarina, PR, Brasil.

BARFORD, P., KLINE, J., PLONKA, D., RON, A., 2002. "A Signal Analysis of Network Traffic Anomalies", *ACM SIGCOMM Internet Measurement Workshop*, Marseille, France, 6-8 Nov.

BRANCH, J. W., BIVENS, A., CHAN, C. Y., LEE, T. K., SZYMANSKI, B. K., 2002, "*Denial of Service Intrusion Detection Using Time Dependent Deterministic Finite Automata*" – Walter Lincoln Hawkins Graduate Research Conference – Troy, NY, EUA, 17 Oct.

BUCHBERGER, S. G., WU, L., 1995, "Model for instantaneous residential water demands". *Journal of Hydraulics Engineering*, Vol. 121-3, pp 232–246.

BUCHBERGER, S. G., WELLS, G. J., 1996, "Intensity, duration, and frequency of residential water demand", *Journal of Water Resource Planning Management*, Vol. 122, No. 1, pp. 11–19.

BURN, L. S., DE SILVA, D., Shipton, R. J., 2002, "Effect of demand management and system operation on potable water infrastructure cost", *Urban Water*, Vol. 4, No. 3, pp. 229–236.

BUTLER, D., GRAHAM, J. D., 1995, "Modeling dry weather wastewater flow in sewer network", *Journal of Environment Engineering*, Vol 121, No. 2, pp. 161–173.

CANSIAN, A. M., CORRÊA, J. L., 2007, "Detecção de ataques de negativa de serviço por meio de fluxos de dados e sistemas inteligentes" – *VII Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais* – Rio de Janeiro – RJ, 27-31 Ago.

CASTELUCIO, A., ZIVIANI, A., SALLES, R. M., 2009, "An AS-Level Overlay Network for IP Traceback" - *IEEE Network*, Vol. 23, pp. 36-41.

CHEN, S., CHOW, R., 2004, "A New Perspective in Defending against DDoS", *10th IEEE International Workshop on Future Trends of Distributed Computing Systems*, Suzhou, China, 26-28 May.

CHEN, S. S., GOPALAKRISHNAN, P. S., 1993, "Speaker, Environment and Channel Change Detection and Clustering via Bayesian Information Criterion". *Proceedings DARPA Broadcast News Transcription and Understanding Workshop*, Lansdowne, VA.

CHENG, C. M., KUNG, H. T., TAN, K. S., 2002 "Use of Spectral Analysis in Defense Against DoS Attacks" – *IEEE GlobeCom*, Taipei, Taiwan, 17-21 Nov.

CLARK, R. M., GRAYMAN, W. M., MALES, R. M., HESS, A, F., 1993. "Modeling contaminant propagation in drinking water distribution systems". *Journal of Environment Engineering*. ASCE, Vol. 119, No. 2, pp. 349-364.

COSTA, R. H. R., 2002 – "Controle Estatístico de Processo da Rede Pública de Abastecimento de Água" – *XXVIII Congresso Interamericano de Ingeniería Sanitaria y Ambiental*, Cancún, México. 27 Oct – 01 Nov.

DEFRAWY, K. E., MARKOPOULOU, A., ARGYRAKI, K., 2006, *Optimal Filtering for DDoS Attacks*, Cornell University Library (http://arxiv.org/abs/cs/0612066)

DICKERSON, J., DICKERSON, J., 2000, "Fuzzy Network Profilling for Intrusion Detection", *19th International Conference of the North American Fuzzy Information Processing Society* - Atlanta, Georgia, EUA, 13-15 July.

DÜBENDORFER, T., BOSSARDT, M., PLATTNER, B., 2005, "Adaptive Distributed Traffic Control Service for DDoS Attack Mitigation" - *19th IEEE International Parallel and Distributed Processing Symposium* - Denver, Colorado – EUA, 3-8 April.

DUCLOS E., 1997. *Contribution à la Maîtrise Statistique des Procédés - Cas des Procédés Non Normaux* , Thèse de Doctorat, Université de Savoie, France.

DUCLOS, E., PILLET, M., AVRILLON, L. 2005, "The L-Chart for Non-Normal Processes", *Quality Technology & Quantitative Management*. Vol. 2, No. 1, pp. 77-90.

DUNCAN, A. J., 1986. *Quality Control and Industrial Statistics*. 2nd. Homewood Eichard D. Irwin.

ESTAN, C., VARGHESE, G., 2003, "New Directions in Traffic Measurement and Accounting: Focusing on the Elephants, Ignoring the Mice" - *ACM Transactions on Computer Systems*, Vol. 21, , No. 3, pp: 270 – 313.

FEINSTEIN, L., SCHNACKENBERG, BALUPARI, R., KINDRED, D., 2003, "Statistical Approaches do DDoS Attack Detection and Response" - *DARPA Information Survivability Conference and Exposition*, Vol 1, pp. 303- 314.

FORMIGA, K. T. M., CHAUDHRY, F.H., 2007 – "Modelos de Análise Hidráulica de redes de distribuição de água considerando demanda dirigida pela pressão e vazamentos" – *Revista Engenharia Sanitária e Ambiental*, Vol. 13, – N^{o} 2 – abr/jun 2008, fls. 153-162.

FUERTES, V., GARCÍA-SERRA, J., PÉREZ, R. 1999. *The Modelling of Water Distribution Systems* - Proc., Universidad Internacional Menéndez Pelayo Int. Course on Drought Management Planning in Water Supply Systems, Kluwer Academic, Dordrecht, Holanda, 52–88.

G1, 2007, *Cedae descobre furto de água em condomínio do Grajaú* - http://g1.globo.com/Noticias/Rio/0,,MUL106573-5606,00.html acessado em 9 de Dezembro de 2008.

GALVÃO, J. R. B., 2007, *Avaliação da Relação Pressão x Consumo, em Áreas Controladas por Válvulas Redutoras de Pressão (VRPs)* – Estudo de Caso: Rede de Distribuição de Água da Região Metropolitana de São Paulo, Dissertação de Mestrado, USP, São Paulo/SP.

GARCÍA, V. J., GARCÍA-BARTUAL, R., CABRERA, E., ARREGUI, F., and GARCÍA-SERRA, J., 2004. "Stochastic model to evaluate residential water demands.", *Journal of Water Resource Planning and Management*, Vol. 130-5, pp. 386–394.

GIL, T. M. and POLETTO, M., 2001, "MULTOPS: a data-structure for bandwidth attack detection", *10th Usenix Security Symposium*, Washington, DC, USA, 13-17 Aug.

HAINS, J. W., LIPPMANN, R. P., FRIED, D. J., ZISSMAN, M. A., TRAN, E., BOSWELL, S. B., 1999, "1999 DARPA Intrusion Detection Evaluation: Design and Procedures". Technical Report 1062, Lincoln Laboratory – MIT.

HANDLEY, M., PAXSON, V., KREIBICH, C., 2001, "Network Intrusion Detection: Evasion, Traffic Normalization, and End-to-End Protocol Semantics", 10th conference on USENIX Security Symposium, Washington, DC, USA, 13-17 Aug.

HE, H., LUO, X, LIU, B., 2005, Detecting Anomalous Network Traffic with Combined Fuzzy-Based Approaches – Lectures Notes in Computer Science, Vol. 3645/2005.

HOUSEHOLDER, A., MANION, A., PESANTE, L. e WEAVER, G., 2001, Managing the threat of denial-of-service attacks, Technical report, CMU Software Engineering Institute CERT Coordination Center, Oct.

HU, W., LIAO, Y., VEMURI, V., 2007, "Robust Anomaly Detection Using Support Vector Machines" - The International Journal on Very Large Data Bases, Vol. 16, No. 4, pp 507 – 521.

HUSSAIN, A., HEIDEMANN, J., PAPADOPOULOS, C., 2003, "Identification of Repeated Attacks Using Network Traffic Forensics" - USC/Information Sciences Institute, Tech. Rep. ISI-TR-2003-569b.

IOANNIDIS, J., BELLOVIN, S. M., 2002, "Implementing Pushback: Router-Based Defense Against DDoS Attack" - NDSS.

ISO, 2005 – "NBR ISO/IEC 27002 – Tecnologia da Informação – Técnicas de segurança – Código de Prática para a gestão de segurança da informação", ABNT.

JAJODIA, S., BARBARA, D., SPEEGLE, B., WU, N., 2000, *Audit Data Analysis and Mining (ADAM)*, PhD Tesis, George Mason University.

JANKOVIĆ-NIŠIĆ, B., MAKSIMOVIĆ, Č., BUTLER, D., and GRAHAM, N. J. D., 2004, "Use of flow meters for managing water supply networks" – Journal of Water Resource Planning and Management, Vol 130, No. 2, pp. 171–179.

JANTZEN, J. 1998. *Design of Fuzzy Controllers*. Technical University of Denmark, Department of Automation, Bldg 326, DK-2800 Lyngby, DENMARK. Tech. report no 98-E 864 (design), 19 Aug.

JIN, C., WANG, H., SHIN, K. G., 2003, "Hop-Count Filtering: An Effective Defense Against Spoofed DDoS Traffic" - *10th ACM Conference on Computer and Communications Security*, Washington, DC, USA, 27-31 Oct.

JIN, S., YEUNG, D. S., 2004, "A Covariance Analysis Model for DDoS Attack Detection" - *IEEE International Conference on Communications*, Vol 4, pp 1882-1886.

JOZIC, K. 2002 – *Tracing Back DDoS attacks,* Master Thesis, University of Stockholm – Royal Institute of Technology, Stockholm, Sweden.

KUZMANOVIC, A., KNIGHTY, E., 2003, "Low-Rate TCP-Targeted Denial of Service Attacks" - *Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, Karlsruhe, Alemanha, 19-23 Aug.

LAKHINA, A., CROVELLA, M., DIOT, C., 2004, *Characterization of Network-Wide Anomalies in Traffic Flows* - Technical Report BUCS-2004-020, Boston University, USA.

LAKHINA, A., CROVELLA, M., DIOT, C., 2004, "Diagnosing Network-Wide Traffic Anomalies" - *Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications* - Portland, Oregon, EUA, 30 Aug – 3 Sep.

LAKHINA, A., PAPAGIANNAKI, K., CROVELLA, M., DIOT, C., KOLACZYK, E., TAFT, N., 2004, "Structural Analysis of Network Traffics Flows" - *ACM SIGMETRICS Performance Evaluation Review,* Vol. 32, No. 1, pp: 61-72.

LANE, T., BRODLEY, C. E., 1997, *Detecting the Abnormal: Machine Learning in Computer Security* – Tech report – TR ECE 97-1 West Lafayette – Purdue University, USA.

LANE, T., BRODLEY, C. E., 1997, "An application of Machine Learning to Anomaly Detection" - *National Information System Security Conference*, Vol 1 – pp. 366-380.

LANE, T., BRODLEY, C. E., 1999, "Temporal Sequence Learning and Data Reduction for Anomaly Detection" - *ACM Transactions on Information and System Security*, Vol 2, No. 3, pp 295-331.

LEE, W., STOLFO, S., 2000, "A Framework for Construction Features and Models for Intrusion Detection Systems" - *ACM Transactions on Information and System Security,* Vol. 3, No. 4, pp:227-261.

LEE, W., XIANG, D., 2001, "Information-Theoretic Measures for Anomaly Detection" - IEEE Symposium on Security and Privacy, Oakland, California, USA, 13-16 May.

LI, Q., CHANG, E., CHAN, M. C., 2005, "On the Effectiveness of DDoS Attacks on Statistical Filtering" – *Annual of 24th Annual Joint Conference of the IEEE Computer and Communications Societies,* Vol 2, pp 1373-1383.

LIMWIWATKUL, L., RUNGSAWANG, A., 2004, "Distributed Denial of Service Detection using TCP/IP Header and Traffic Measurement Analysis". *International Symposium on Communications and Information Technologies*, Sapporo, Japan, 26-29 Oct.

LINGIREDDY, S., WOOD, D. J., and NELSON, A., 1998, "Modified pipe network model for incorporating peak demand requirements", Journal of Water Resource Planning and Management, Vol. 124, No. 5, pp. 296–299.

LIPPMANN, R., HAINES, J. W., FRIED, D. J., KORBA, J., DAS, K., 2000, "*Analysis and Results of the 1999 DARPA Off-Line Intrusion Detection Evaluation"*, RAID 2000, LNCS 1907, pp. 162-182.

MAHADIK, V. A., Wu, X., REEVES, D. S., 2002, "Detection Denial-of-QoS Attacks Based on χ^2Statistic And EWMA Control Charts*" – 11th Usenix Security Symposium* – San Francisco – USA, 5-9 Aug.

MAHAJAN, R., BELLOVIN, S. M., FLOYD, S., IOANNIDIS, J., PAXSON, V., SHENKER, S., 2002, "Controlling High Bandwidth Aggregates in the Network" - *ACM SIGCOMM Computer Communication Review*, Vol. 32, No. 3, pp: 62-73.

MAMDANI, E. H., 1976. "Advances in the linguistic synthesis of fuzzy controllers", *International Journal of Man-Machine Studies*, Vol. 8, pp. 669-678.

MARCHETTE, D., 1999, "A Statistical Method for Profiling Network Traffic", *Workshop on Intrusion Detection and Network Monitoring, Santa Clara*, California, USA, 9-2 Apr.

MARTÍNEZ-SOLANO, J., IGLESIAS-REY, P. L., PÉREZ-GARCÍA, R., LÓPEZ-JIMÉNEZ, P. A., 2008, "Hydraulic Analysis of Peak Demandin Looped Water Distribution Networks", *Journal of Water Resources Planning and Management*, Vol. 134, No. 6, pp. 504-510.

MELL, P., MARKS, D., MCLARNON, M., 2000, "A denial-of-service resistant intrusion detection architecture" – *Computer Networks,* Vol. 34, pp. 641-658.

MIRKOVIC, J., PRIER, G., REIHER, P., 2002, "Attacking DDoS at the Source", *10th IEEE International Conference on Network Protocols*, Paris, France, 12-15 Nov.

MIRKOVIC, J., MARTIN, J., REIHER, P., 2004, "A Taxonomy of DDoS Attacks and DDoS Defense Mechanisms", *ACM Sigcomm Computer Communication Review*, Vol. 34, No. 2, pp. 39–53.

MOHINDDIN, S., HERSHKOP, S., BHAN, R., STOLFO, S., 2002, "Defending Against a large scale Denial-of-Service Attack" - *IEEE Workshop Information Assurance*, US Military Academy, West Point, EUA, Jun.

MONTGOMERY, D. C., RUNGER, G. C., 1999, *Applied Statistics and Probability for Engineers*. John Wiley & Sons.

MOORE, D., SHANNON, C., VOELKER, G. M., SAVAGE, S. 2004, "*Network Telescopes: Technical Report*". - Cooperative Association for Internet Data Analysis (CAIDA).

MOORE, D. S., McCABE, G. P., 2005, *Introduction to the Practice of Statistics*, fifth edition, W. H. Freeman.

MOORE, D., VOELKER, G., SAVAGE, S., 2006, "Inferring Internet Denial-of-Service Activity" - *ACM Transactions on Computer Systems*, Vol. 24, No. 2, pp: 115 – 139.

MORAES, C. F., FERREIRA, J. R., BALESTRASSI, P. P., 2006, "Análise crítica da aplicação de métodos estatísticos em processos definidos por dados que não apresentam distribuição normal", 2006 – *GEPROS*, No 1, pp. 7-18.

MUNIZ, C., FIGUEIREDO, K., VELLASCO, M., PACHECO, M., CHAVEZ, G., 2008 "Indicação de Suspeitos de Irregularidades em Instalações Elétricas de Baixa Tensão" – II Workshop on Computational Intelligence – Salvador, Bahia, Out 26-28

NEUMANN, P. G., PORRAS, P. A., 1999, "Experience with EMERALD to DATE", 1st Workshop on Intrusion Detection and Network Monitoring, Santa Clara, California, 11-12 Apr.

NOURELDIEN, N. A., OSMAN, I. M., 2002, "A Method for Defeating DoS/DDoS TCP SYN Flooding Attack –The SYNDEF" - *14th Annual FIRST Computer Security Incident Handling*, Hawaii, EUA, 24-28 Jun.

OLIVEIRA, L., Aschoff, R., Lins, B., Feitosa, E., Sadok, D., 2006, "Avaliação de Proteção contra Ataques de Negação de Serviço Distribuídos (DDoS) utilizando Lista de IPs Confiáveis" – *VII Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais* – Rio de Janeiro – RJ, 27-31 Ago.

PANG R., YEGNESWARAN, V., BARFORD, P., PAXSON, V., PETERSON, L., 2004, "Characteristics of Internet Background Radiation" - *4th ACM SIGCOMM Conference on Internet measurement*- Taormina, Sicily, Italia, 25-27 Oct.

PARK, K., LEE, H., 2001, "On the Effectiveness of Route-Based Packet Filtering for Distributed DoS Attack Prevention in Power-Law Internets" - *ACM SIGCOMM Computer Communication*, Vol. 31, No. 4, pp: 15-26.

PAUL, B, 2001, "*DDoS: Internet Weapons of Mass Destruction*" – Network Computing – http://networkcomputing.com/ 1201/1201f1c1.html.

PENG, T., LECKIE, C., RAMAMOHANARAO, K., 2003, "Protection from Distributed Denial of Service Attack Using History-based IP Filtering", *IEEE International Conference on Communications*, Anchorage, Alaska, USA, 11-15 May.

PTACEK, T. H., NEWSHAM, T. N., 1998 "*Insertion, Evasion and Denial of Service: Eluding Network Intrusion Detection*" – Secure Networks, Inc.

QIN, M., HWANG, K., 2004, "*Anomaly Intrusion Detection by Internet Datamining of Traffic Episodes*", Technical Report No. 2004-6, USC Internet and Grid Computing Lab.

QIN, M., HWANG, K., 2004, "*Internet Episodes Analysis for Detecting Anomalous Attacks on Information Infrastructures*" -Technical Report, USC Internet and Grid Computing Lab (TR 2004-11).

RAZMOV, V., 2000, *Denial of Service Attacks and How to Defend Against Them*, Survey Project Paper, Dept. of Computer Science and Engineering – University of Washington – EUA.

REIS, M. M., 2001, *"Um modelo para o Ensino de Controle Estatístico da Qualidade"* – Tese de Doutorado, Universidade Federal de Santa Catarina.

REZENDE, S. O., 2003. *Sistemas Inteligentes, Fundamentos e Aplicações*. 2ª Edição São Paulo, Ed Manole.

ROJKOVA, V., KANTARDZIC, M., 2007, "*Analysis of Inter-Domain Traffic Correlations: Random Matrix Theory Approach*" – Biblioteca Digital – Cornell University Library (http://arxiv.org/abs/0706.2520).

ROJKOVA, V., KANTARDZIC, M., 2007, "*Delayed Correlations in Inter-Domain Network Traffic*" – Biblioteca Digital – Cornell University Library (http://arxiv.org/abs/0707.1083).

SANDRI, S., CORREA, C., 1999. "Lógica Nebulosa" - *V Escola de Redes Neurais*, ITA, São José dos Campos, SP, Brasil, 19 Jul.

SANTOS, A. F. P., SILVA, R. S., 2007, "Detectando Ataques de Negação de Serviço". *XXX Congresso Nacional de Matemática Aplicada e Computacional*, Florianópolis, Brasil, 3-6 Set.

SANTOS, A.F.P., SILVA, R.S., 2007, "Detecting Bandwidth DDoS Attacks with Control Charts", *15th IEEE International Conference on Networks*, Adelaide, Austrália, 19-21 Nov.

SAVAGE, S., CARDWELL, N., WETHERALL, D., ANDERSON, T., 1999, "TCP Congestion Control with a Misbehaving Receiver" - *ACM SIGCOMM Computer Communications Review*, Vol. 29, No. 5, pp 71 - 78.

SAVAGE, S., WETHERALL, D., KARLIN, A., ANDERSON, T., 2000, "Pratical Network Support For IP Traceback" – *ACM Special Interest Group on Data Communication (SIGCOMM)*, Stockholm, Suécia, 28 Aug - 1º Sep.

SCHUBA, C. L., KRSUL, I. V., KUHN, M. G., SPAFFORD, E. H., SUNDARAM, A., ZAMBONI, D., 1997, "Analysis of a Denial of Service Attack on TCP" - *IEEE Symposium on Security and Privacy*, Oakland, California, 4-7 May.

SCHWARTZBARD, A., GHOSH, A. K., 1999, "*A Study in the Feasibility of Performing Host-Based Anomaly Detection on Windows NT*", 2nd Recent Advances in Intrusion Detection (RAID 1999) Workshop, West Lafayette, Indiana, USA, 7-9 Sep.

SEKAR, R., UPPULURI, P., 1999, "Synthesizing Fast Intrusion Prevention/Detection Systems from High-Level Specification", 8th USENIX Security Symposium, Washington, DC, 23-26 Aug.

SHADOWSERVER, 2009 – *ShadowServer Foundation* – http://www.shadowserver.org

SIATERLIS, C., MAGLARIS, B., 2004, "Detecting DDoS attacks with passive measurement based heuristics" - *9th IEEE Symposium On Computers and Communications*, Alexandria, Egito, 28 Jun – 1º Jul.

SIATERLIS, C., MAGLARIS, V., 2005, "Detecting incoming and outgoing DDoS attacks at the edge using a single set of network characteristics", *10th IEEE Symposium on Computers and Communications* - La Manga del Mar Menor, Cartagena, Espanha, 27-30 Jun.

SILVA, F. G. B., GRATÃO, U., SANTOS, A., REIS, L. F. R., PORTO, R. M., CHAUDHRY, F. H., 1999, "Controle operacional em sistemas de distribuição de água para abastecimento visando o controle de perdas: experiência em sub-setores da cidade de São Carlos – SP" - *V Simpósio do Curso de Pós-Graduação em Ciências da Engenharia Ambiental da EESC-USP*, São Carlos, SP, Brasil.

SILVA, F. G. B., REIS, L. F. R., SOARES, A. K., 2002, *Calibração de Rede de Distribuição de Água Considerando as Perdas por Vazamento Explicitamente no Modelo com uso de Algoritmos Genéticos - Aplicação para Rede da Cidade de São Carlos, SP,* Planejamento, Projeto e Operação de Redes de Abastecimento de Água, O Estado da Arte e Questões Avançadas, João Pessoa, PB, Brasil.

SILVA, F. G. B., REIS, L. F. R., 2002, "Calibração de redes de distribuição de água com algoritmos genéticos aplicada a uma rede hipotética" – *VI Simpósio Ítalo Brasileiro de Engenharia Sanitária e Ambiental*, Vitória, ES, Brasil, 1-5 Set.

SILVA, R. M., 2005*, Redes neurais artificiais aplicadas à detecção de intrusão em redes TCP/IP, Dissertação de Mestrado, PUC-RJ, RJ, Brasil.*

SINCLAIR, C., PIERCE, L., MATZNER, S., 1999, "An Application of Machine Learning to Network Intrusion Detection", *15th Annual Computer Security Applications Conference*, Scottsdalle, Arizona, USA, 6-10 Dec.

SOARES, A. K., REIS, L. F. R., FORMIGA, K. T. M., SILVA, F. G. B., CARRIJO, I. B., 2003, "Aplicação de Modelo Dirigido pela Pressão na Estimativa de Parâmetros de Redes de Distribuição de Água" - *XV Simpósio Brasileiro de Recursos Hídricos*, Curitiba, Paraná, Brasil 23-27 Nov.

SOMMER, R., PAXSON, V., 2003, "Enhancing Byte-Level Network Intrusion Detection Signatures with Context" - *10th ACM conference on Computer and Communications Security*, Washington DC, EUA, 27-30 Oct.

SONG, D. X., PERRIG, A., 2001, "Advanced and Authenticated Marking Schemes for IP Traceback" - *20th Annual Joint Conference of the IEEE Computer and Communications Societies*, Anchorage, Alaska, 24-26 Apr.

TAROUCO, L. M. R., ANDREOLI, A. V., BERTHOLDO, L. M., 2003, "Compreendendo Ataques Denial of Services", *I Escola Regional de Redes de Computadores*, Porto Alegre, Brasil, 23-24 Set.

TENG, H. S., CHEN, K., LU, S.C., 1990, "Adaptative Real-time Anomaly Detection Using Inductively Generated Sequential Patterns", *IEEE Computer Society Symposium on Research in Security and Privacy*, Oakland, California, USA, 7-9 May.

THORNTON, J., 2002, *Water loss control manual*. 1st ed. McGraw-Hill Professional.

TOLEDO, J. C., AZEKA, F., AMARAL, D. C., 1999, Projeto Robusto/Método de Taguchi, http://www.numa.org.br/conhecimentos/conhecimentos_port/pag_conhec/Projeto_robustov5.html

TRON, E., MARGALIOT, M., 2004, "Mathematical Modeling of Observed Natural Behavior: A Fuzzy Approach", *Fuzzy and Set Systems*, Vol. 146, No. 3, pp. 437-450.

TYSON, M., BERRY, P., WILLIAMS, N., MORAN, D., BLEI, D., 2000, "DERBI: Diagnosis, Explanation and Recovery from Computer Break-Ins", Research in Artificial Intelligence Center, SRI International, Apr.

VELEDA, R., 2008, "Polícia desmonta esquema de furto de água da Caesb", *Correio Braziliense* – Polícia, 12 de Maio de 2008.

VIGNA, G., ECKMANN, S. T., KEMMERER, R. A., 1999, "NetSTAT: A Network-based Intrusion Detection System", *Journal of Computer Security* archive. Volume 7 , Issue 1, Sep.

VIGNA, G., ECKMANN, S. T., KEMMERER, R. A., 2000, "The STAT Tool Suite", *2000 DARPA Information Survivability Conference and Expositium*, Hilton Head, South Carolina, USA, 25-27 Jan.

WALLACE, L. P., 1987, *Water and Revenue Losses: Unaccounted-for Water*, American Water Works Association Research Foundation/American Water Works Association, Denver, EUA.

WONG, T. Y., LAW, K. T., LUI, J. C. S., WONG, M. H., 2006, "An Efficient Distributed Algorithm to Identify and Traceback DDoS Traffic", *The Computer Journal*, Vol. 49, No. 4, pp. 418-442.

YAAR, A., PERRIG, A., SONG, D., 2003, "Pi: A Path Identification Mechanism to Defend against DDoS Attacks" - *IEEE Computer Society Symposium on Research in Security and Privacy,* Oakland, California, USA, 11-14 May.

YE, N., 2000, "A Markov Chain Model of Temporal Behavior for Anomaly Detection", *Workshop on Information Assurance and Security,* United States Military Academy , West Point, NY, EUA, 6-7 Jun.

YEGNESWARAN, V., BARFORD, P., ULLRICH, J. , 2003, "Internet Intrusions: Global Characteristics and Prevalence", *Joint International Conference on Measurement and Modeling of Computer Systems - ACM SIGMETRICS international conference on Measurement and modeling of computer systems,* San Diego, CA, EUA, 9-14 Jun.

ZADEH, L. A., 1965, "Fuzzy sets", *Fuzzy Sets, Information and Control*, Vol. 8, pp. 338 – 353.

ZADEH, L. A., 1978, "Fuzzy sets as a basis for a theory of possibility", *Fuzzy Sets and Systems*, Vol. 1, pp. 3–28.

Anexo A – Código Fonte

A.1. Algoritmo de Detecção de Anomalias com Janelas Adaptativas

```
function [media, desvio, bandeiraVermelha]=AlgoritmoD(Vetor,
NumeroPontosAmostrados,NumeroRepeticoesBootstrap,Janela)

%Descricao basica
%1.Selecionar uma janela
%2.Nesta janela fazer uma amostragem RSS
%3.Para esta amostragem, calcular, por duclos, a media e o desvio
%4.Repetir 2 e 3 B vezes (bootstrap)
%5.Fazer a media dos valores de media e desvio
%6.Com esses valores de media e desvio totais, comparar com os valores do proximo
intervalo.
%Algoritmo do Duclos com a protecao do numero de condicionamento

fprintf('Setando variaveis iniciais...\n');
TamanhoVetor=size(Vetor);
NumeroElementos=TamanhoVetor(:,1);
n=NumeroPontosAmostrados;
JanelaOriginal = Janela;
Passo = Janela;
dPasso = 0;
i=1;
ataque=0;
while(i<NumeroElementos)
    b = 1;
    fprintf('Pos:%d/%d\n', i, Passo);
    while (b<NumeroRepeticoesBootstrap && i<NumeroElementos) %bootstrap
        fprintf('     Bootstrap:%d  ', b);
        fprintf('Amostragem:');
        for j=1:5:NumeroPontosAmostrados  %RSS
            l=1;
            for k=1:5:NumeroPontosAmostrados %amostragem
                indice=-1;
                if((Passo + i + 0) <= NumeroElementos)
                    while(~(indice>=0 && indice<=floor(Passo/5) && (indice + i+0) <=
                       NumeroElementos) )
                        IndiceAleatorio=rand(1)*Passo/5;
                        indice=floor(IndiceAleatorio);
                   end %while
                    indice = indice + i;
                    amostra(l+0)=Vetor(indice);
                end %if
                if((Passo + i + 1) <= NumeroElementos)
                    while(~(indice>=floor(Passo/5) && indice<=floor(2*Passo/5) &&
                      (indice + i + 1) <= NumeroElementos) )
                        IndiceAleatorio=rand(1)*Passo/5 + Passo/5;
                        indice=floor(IndiceAleatorio);
                   end % while
                    indice = indice + i;
                    amostra(l+1)=Vetor(indice);
                end %if
                if((Passo + i + 2) <= NumeroElementos)
```

```
            while(~(indice>=floor(2*Passo/5) && indice <= floor(3*Passo/5) &&
              (indice + i + 2) <= NumeroElementos) )
                IndiceAleatorio=rand(1)*Passo/5 + 2*Passo/5;
                indice=floor(IndiceAleatorio);
           end % while
            indice = indice + i;
            amostra(l + 2)=Vetor(indice);
        end %if
        if((Passo + i + 3) <= NumeroElementos)
            while(~(indice>=floor(3*Passo/5) && indice <= floor(4*Passo/5) &&
              (indice + i + 3) <= NumeroElementos) )
                IndiceAleatorio=rand(1)*Passo/5 + 3*Passo/5;
                indice=floor(IndiceAleatorio);
           end % while
            indice = indice + i;
            amostra(l + 3)=Vetor(indice);
        end %if
        if((Passo + i + 4) <= NumeroElementos)
            while(~(indice>=floor(4*Passo/5) && indice <= floor(5*Passo/5) &&
              (indice + i + 4) <= NumeroElementos) )
                IndiceAleatorio=rand(1)*Passo/5 + 4*Passo/5;
                indice=floor(IndiceAleatorio);
           end % while
            indice = indice + i;
            amostra(l + 4)=Vetor(indice);
        end %if
        if(indice==-1)
            indice
            %Passo = NumeroElementos - i - 3;
            %indice=floor(IndiceAleatorio);
            %k=NumeroPontosAmostrados+1;
            if(i+Passo>NumeroElementos)
                indice=-2;
                fprintf('Problema com a quantidade de pontos. Pos Atual
                   %d\nPasso %d\n', i, Passo);
            end %if
            i=NumeroElementos;
        end %if
        l=l+5;
    end %for
    %Passo 2: Ordenacao
    Xtemp=sort(amostra);
    %X(j)=Xtemp(j);
    X = Xtemp;
end %for
%Passo 3: Calculo de mi* e sigma*
fprintf('Duclos:');
%Passo 3.1: Calculo dos Vetores Padronizados
Mii=mean(X);
Vari=std(X);
if(Vari>0)
    IndiceVarianciaZero(i,b)=0;
    U=(X-Mii)./Vari;

    %Passo 3.2: Calculo dos alfas i - Como cada elemento u(r) eh um
    %numero escalar, a esperanca destes valores sao os proprios
    %valores.
    Alfa=U';
```

```
        A=zeros(n,2);
        %Passo 3.3: Calculo da matriz A
        A(:,2)=Alfa(:,1);
        A(:,1)=ones(n,1);
        %Passo 3.4: Calculo da matriz Omega
        Omega(:,:)=zeros(n,n);
        for m=1:1:n
            Omega(m,m)=Alfa(m)*Alfa(m);
        end %for
        fprintf('Condicionamento\n');
        if(cond(Omega) <= 1000000)
            X=X';
            %Passo 3.5: Calculo de mi* e sigma* (theta)
            Theta=(inv(A'*(inv(Omega))*A))*(A'*(inv(Omega))*X);
            Mi(b)=Theta(1);
            Sigma(b)=Theta(2);
            Media(b)=mean(Mi);
            Desvio(b)=mean(sqrt(Sigma));
            X=X';
            Alfa=Alfa';
            IndiceMalCondicionado(i,b)=0;
            b = b + 1;
        else
            fprintf('    Problemas de Condicionamento:%d\n', cond(Omega));
        end %if
    end %if
end %while
media(i)  = mean(Media);
desvio(i) = std(Media);
fprintf('     Identificacao\n');
if(ataque==0)
    if(i>1 && (media(i) > media(i-1) + 3*desvio(i-1)))
        bandeiraVermelha(i)=1;
        ataque=1;
        mediaataque = media(i-1) + 3*desvio(i-1);
    else
        bandeiraVermelha(i)=0;
    end
else
    if(i>1 && (media(i) > media(i-1) + 3*desvio(i-1)))
        bandeiraVermelha(i)=2;
        ataque = 1;
        mediaataque = media(i-1) + 3*desvio(i-1);
    else
        if(i>1 && media(i) < mediaataque)
            bandeiraVermelha(i)=0;
            ataque=0;
        else
            bandeiraVermelha(i)=0;
        end
    end %if
end

fprintf('     Fuzzy\n');
if(i+Passo > NumeroElementos)
    ifinal = NumeroElementos;
else
    ifinal = i + Passo;
```

```
    end

    media(i+1 : ifinal) = media(i);
    desvio(i+1 : ifinal) = desvio(i);
    bandeiraVermelha(i+1 : ifinal) = bandeiraVermelha(i);
    i=i+Passo;
    if(i < NumeroElementos)
        PassoFz = Passo;
        dPasso    =    Fz(media(i),    min(media),    max(media),    desvio(i)/media(i),
min(desvio)/media(i), max(desvio)/media(i), PassoFz);
        PassoNovo = floor(Passo + dPasso);

        if(i+PassoNovo > NumeroElementos)
            PassoNovo = NumeroElementos - i;
        else
            if((PassoNovo>1000*Janela))
                PassoNovo=floor(Passo/2);
            end
           if(PassoNovo<floor(Janela/2))
               PassoNovo=Passo;
           end
        end
        Passo = PassoNovo;
    end
end
```

Anexo B – Ataques de DoS e DDoS em DARPA 99

B.1. Ataques existentes na base de dados DARPA 99

Na simulação realizada em 1999, foram utilizados os ataques de negação de serviço contidos na tabela B.1. Na tabela B.2 estão presentes os momentos em que os ataques ocorrem.

Tabela B.1. Ataques presentes em 1999.

Ataque	Descrição
apache2	Ataque proferido contra o servidor web apache[25], que ocorre quando o servidor não consegue resolver uma quantidade elevada de solicitações de serviço.
arppoison	É um ataque que foi desenvolvido especificamente para a simulação e se baseia em alterar o endereço MAC de equipamentos com endereços IPs conhecidos.
back	Ataque realizado contra servidores web apache, onde o atacante envia para o servidor uma solicitação de endereço com várias caracteres barras.
crashiis	Ataque realizado contra servidores NT IIS, onde o atacante envia requisições GET mal formadas.
dosnuke	Ataque realizado contra servidores Windows NT, onde o atacante envia pacotes incorretos para a porta 139.
land	Ataque que explora a vulnerabilidade do protocolo TCP/IP.
mailbomb	Ataque em que o servidor de e-mail recebe mais mensagens do que pode tratar.
syn flood (neptune)	Ataque em que o servidor não consegue tratar solicitações de conexão, por estar sobrecarregado.
ping of death (pod)	Ataque em que pacotes ICMP são enviados com tamanhos incorretos.

[25] HTTP://www.apache.org

Ataque	Descrição
process table	Ataque criado especificamente para a avaliação, onde para cada conexão estabelecida com o servidor, um novo processo é criado.
selfping	Ataque realizado contra servidores, através do uso incorreto de parâmetros os pacotes ICMP, causando reinicialização do equipamento.
smurf	Ataque em que são enviados pacotes ICMP echo request para o endereço de *broadcast* da rede, usando o endereço da vítima.
sshprocesstable	Similar ao processtable, exceto que utiliza o serviço *sshd*.
syslogd	Ataque em que o serviço *syslogd* é desligado, causando quebra do sistema.
tcpreset	Ataque em que ocorre quebra nas conexões TCP da máquina alvo, através do envio do pacote TCP RESET, usando o endereço falso do cliente legítimo.
teardrop	Ataque que explora a vulnerabilidade do protocolo.
udpstorm	Ataque que explora a vulnerabilidade do protocolo UDP.

Tabela B.2. Eventos de ataques de negação de serviço contidos na base de dados de 1999.

Semana	Dia	Hora	Ataque
2	2ª	08:50:15	pod
2	2ª	09:36:16	back
2	2ª	15:57:15	land
2	3ª	10:06:43	back
2	3ª	14:25:16	mailbomb
2	4ª	13:44:18	mailbomb
2	4ª	23:56:14	crachiis
2	5ª	08:04:17	crahiis
2	5ª	11:04:16	neptune
2	5ª	15:47:15	land
2	6ª	09:18:15	pod
2	6ª	11:20:15	neptune
2	6ª	12:40:12	crashiis
4	2ª	16:13:08	crashiis
4	2ª	21:34:46	smurf
4	3ª	14:54:41	land

Semana	Dia	Hora	Ataque
4	3ª	15:51:48	mailbomb
4	3ª	17:49:44	processtable
4	3ª	21:04:10	crashiis
4	4ª	10:13:13	processtable
4	4ª	11:11:11	warezmaster
4	4ª	11:30:32	arppoison
4	4ª	14:45:47	smurf
4	4ª	16:54:22	mailbomb
4	5ª	11:00:00	dosnuke
4	5ª	16:49:44	processtable
4	5ª	18:32:34	mailbomb
4	6ª	08:45:47	smurf
4	6ª	09:00:00	arppoison
4	6ª	12:32:34	mailbomb
5	2ª	08:38:00	pod
5	2ª	08:50:00	pod
5	2ª	08:59:47	warezclient
5	2ª	09:31:23	smurf
5	2ª	10:27:00	apache2
5	2ª	11:13:33	arppoison
5	2ª	11:45:00	dosnuke
5	2ª	13:18:03	smurf
5	2ª	13:30:19	arppoison
5	2ª	14:01:00	apache2
5	2ª	14:21:00	pod
5	2ª	17:19:17	Syslogd
5	2ª	18:04:45	neptune
5	2ª	18:36:23	crashiis
5	2ª	19:47:15	dosnuke
5	2ª	20:17:15	selfping
5	3ª	08:11:09	tcpreset
5	3ª	08:32:36	teardrop
5	3ª	09:45:14	selfping
5	3ª	10:34:09	back
5	3ª	11:38:55	neptune
5	3ª	13:06:55	pod
5	3ª	13:50:03	crashiis
5	3ª	14:12:00	syslogd
5	3ª	18:16:37	neptune
5	3ª	20:56:05	dosnuke
5	3ª	05:08:13	udpstorm
5	4ª	04:54:54	selfping
5	4ª	09:20:39	tcpreset
5	4ª	10:26:17	back
5	4ª	15:01:10	processtable
5	4ª	15:26:48	back
5	4ª	17:13:50	apache2

Semana	Dia	Hora	Ataque
5	5ª	11:57:01	crashiis
5	5ª	15:53:38	teardrop
5	5ª	19:41:08	warezclient
5	5ª	22:51:31	arppoison
5	6ª	08:14:18	Crashiis
5	6ª	08:44:52	Back
5	6ª	10:20:00	Tcpreset
5	6ª	12:51:12	crashiis
5	6ª	12:58:30	crashiis
5	6ª	14:06:43	syslogd
5	6ª	16:34:47	land
5	6ª	17:19:17	syslogd
5	6ª	18:30:12	neptune
5	6ª	18:52:33	warezclient

Anexo C – Gráficos do Capítulo 5

C.1. Gráficos relacionados com a Situação 1[i].

Os gráficos C.1, C.2, C.3 e C.4 são, respectivamente, aos gráficos obtidos com a janela de tamanho inicial da janela de 10, enquanto que os gráficos C.5, C.6, C.7 e C.8 são os obtidos com o tamanho inicial da janela com 20. Em vermelho, encontra-se o gráfico original, igual à Figura 5.4, enquanto que em azul encontram-se os períodos de identificação do algoritmo. Um patamar superior significa que nesta região, está acontecendo uma contaminação, ou seja, que este intervalo há um aumento significativo da vazão. O patamar inferior indica que o período referenciado é de normalidade.

Deve-se lembrar que os tamanhos das janelas são dinâmicos, calculados através de controladores *fuzzy*, descritos no capítulo 4, e não há qualquer obrigatoriedade de iniciar ou terminar junto com o período de início ou término de contaminação. Desta forma, interpretamos como *falso positivo* quando há interseção entre um período de não contaminação da base de dados com um período identificado como contaminado. Da mesma forma, interpretamos como *falso negativo* quando um período de contaminação não tem nenhuma intersecção com um período de contaminação.

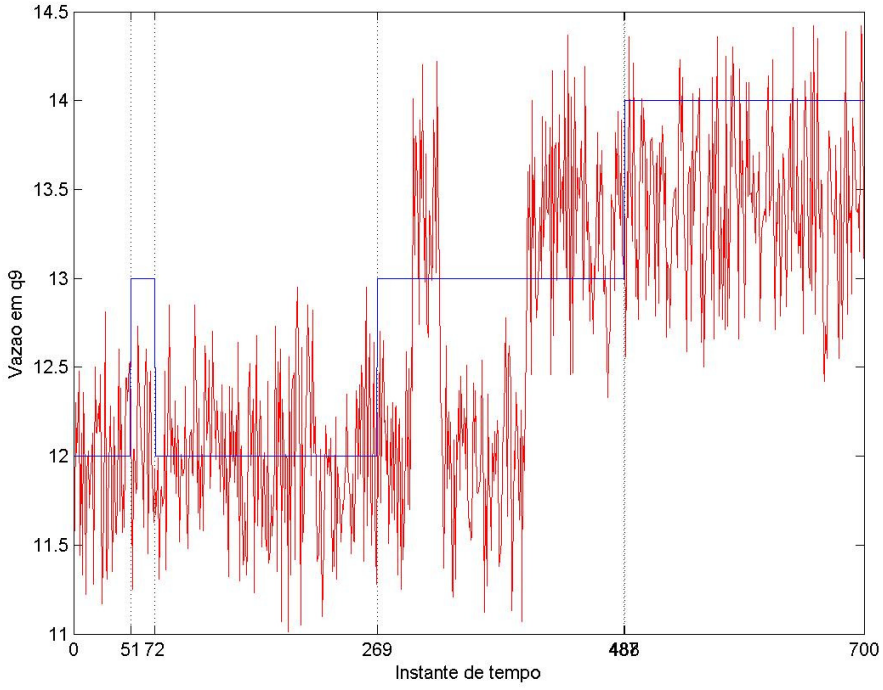

Figura C.1: Gráfico vazão por instante de tempo, para a situação 1 (B=20, J=10);

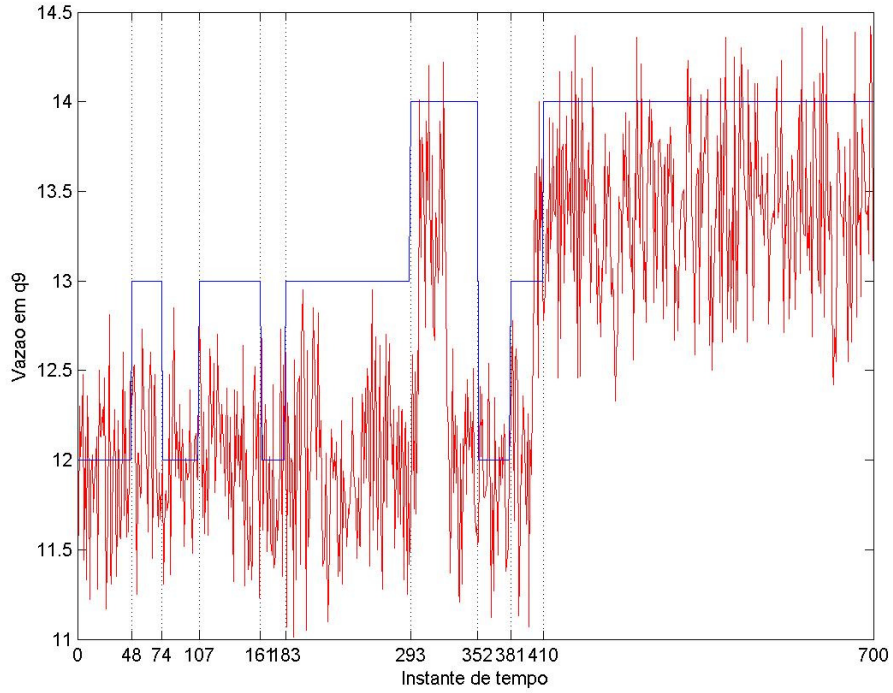

Figura C.2: Gráfico vazão por instante de tempo, para a situação 1 (B=100, J=10);

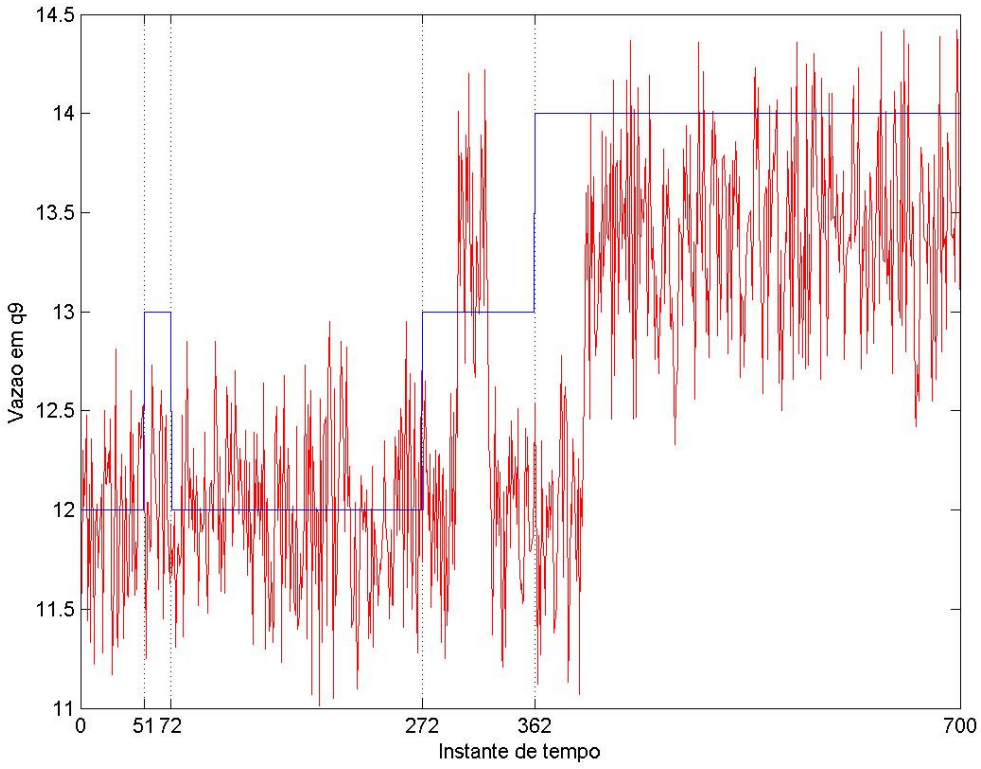

Figura C.3: Gráfico vazão por instante de tempo, para a situação 1 (B=200, J=10);

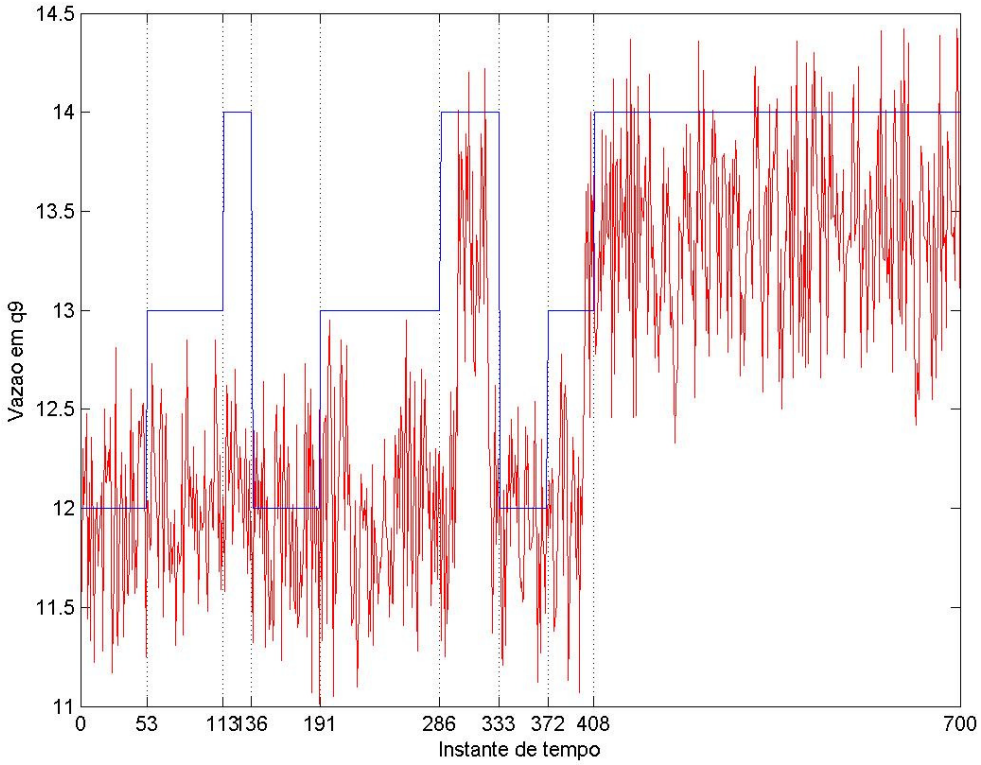

Figura C.4: Gráfico vazão por instante de tempo, para a situação 1 (B=400, J=10);

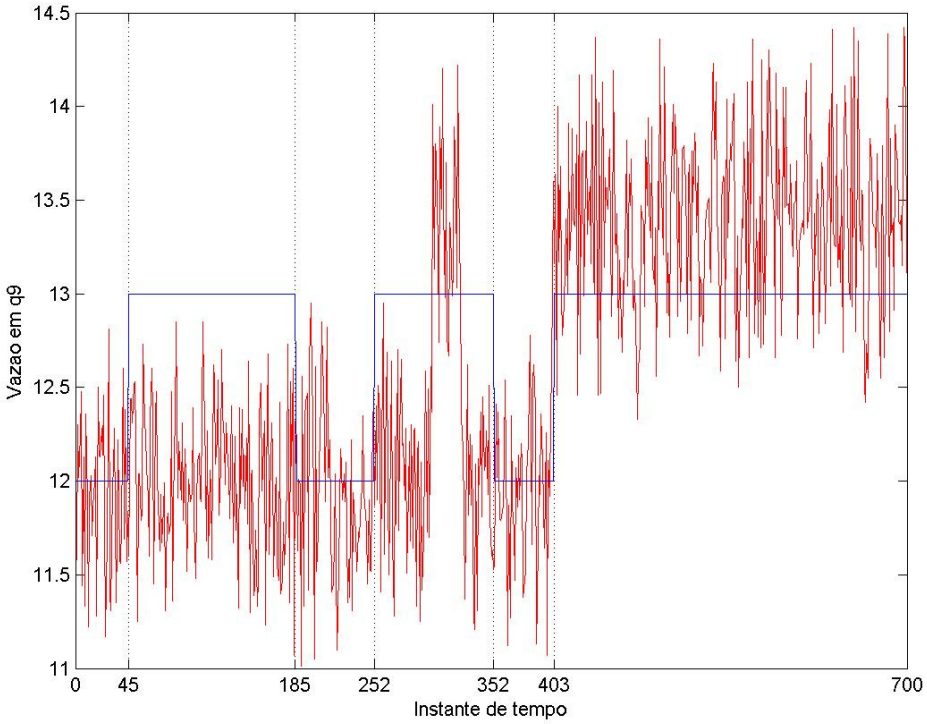

Figura C.5: Gráfico vazão por instante de tempo, para a situação 1 (B=20, J=20);

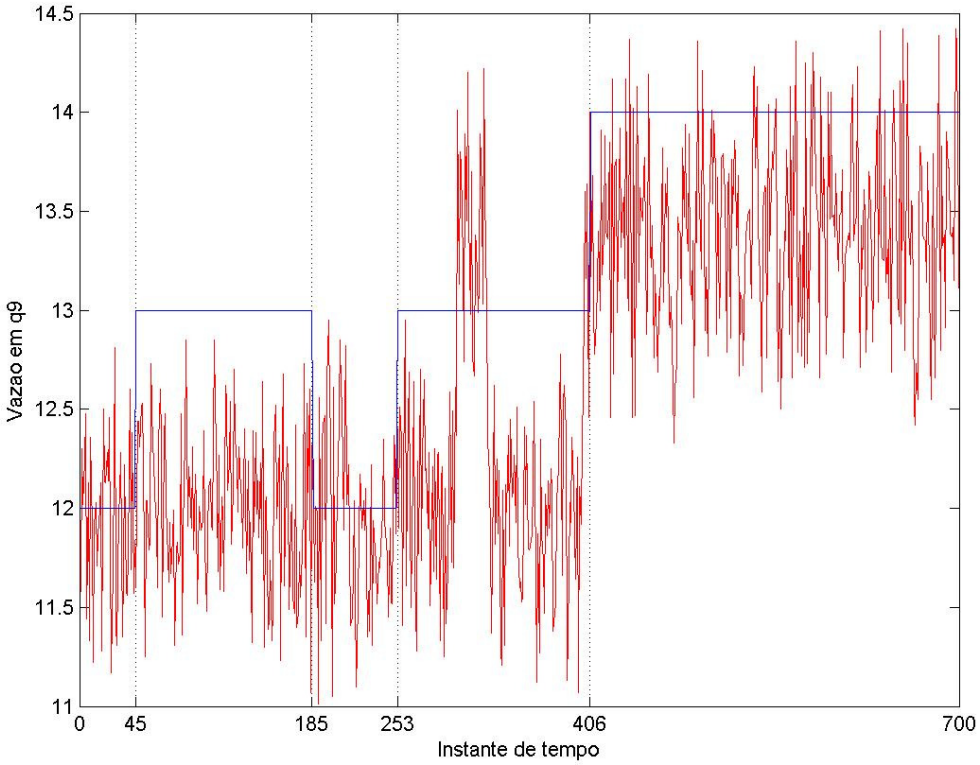

Figura C.6: Gráfico vazão por instante de tempo, para a situação 1 (B=100, J=20);

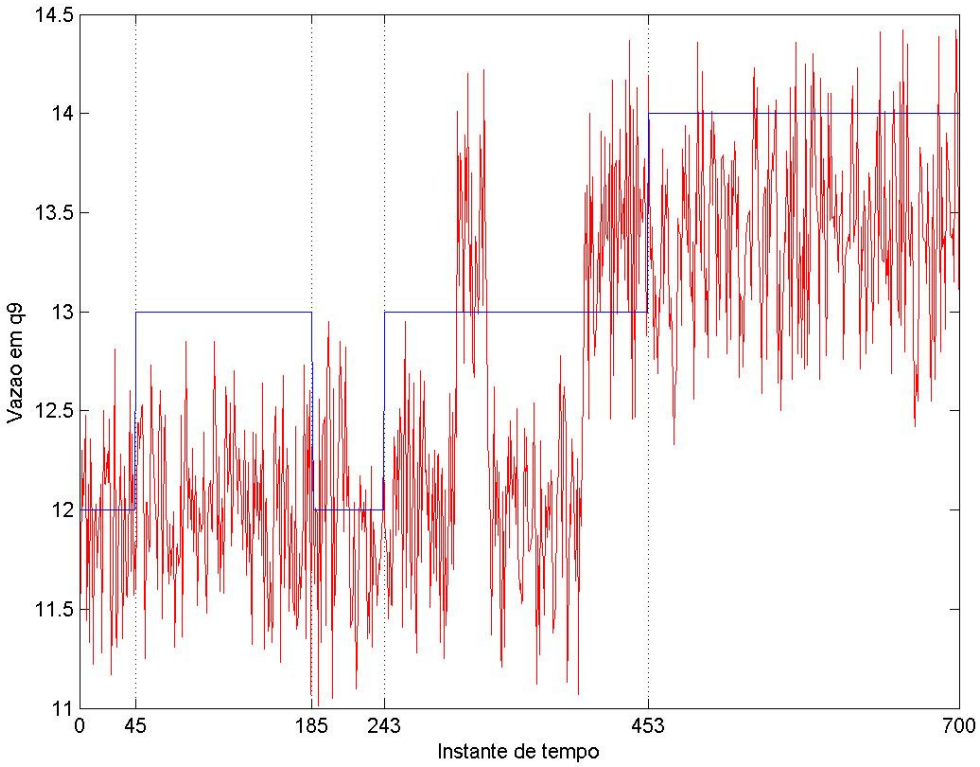

Figura C.7: Gráfico vazão por instante de tempo, para a situação 1 (B=200, J=20);

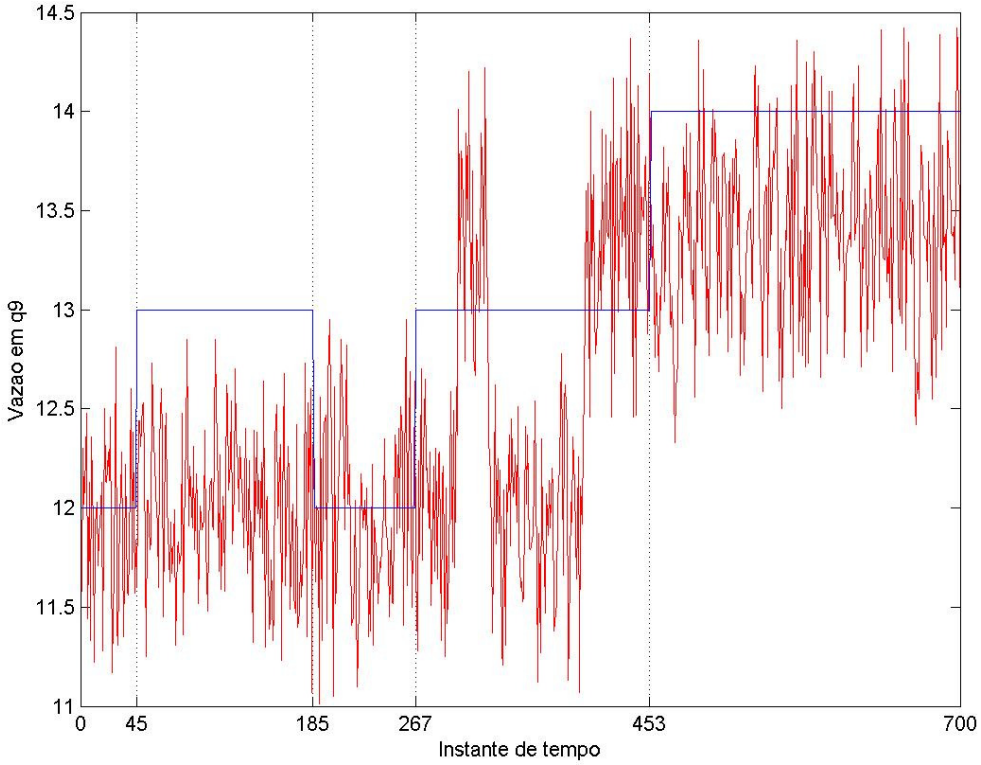

Figura C.8: Gráfico vazão por instante de tempo, para a situação 1 (B=400, J=20);

C.2. Gráficos relacionados com a Situação 2.

Para os gráficos deste subitem, as mesmas condicionantes foram impostas, diferindo, com relação ao subitem anterior, da base de dados, que neste caso não há perturbação existente. Portanto, é de se esperar que não haja identificação de aumento significativo da vazão. Por esta razão, as identificações encontradas, indicadas pelos gráficos de cor azul, são consideradas como falso positivo.

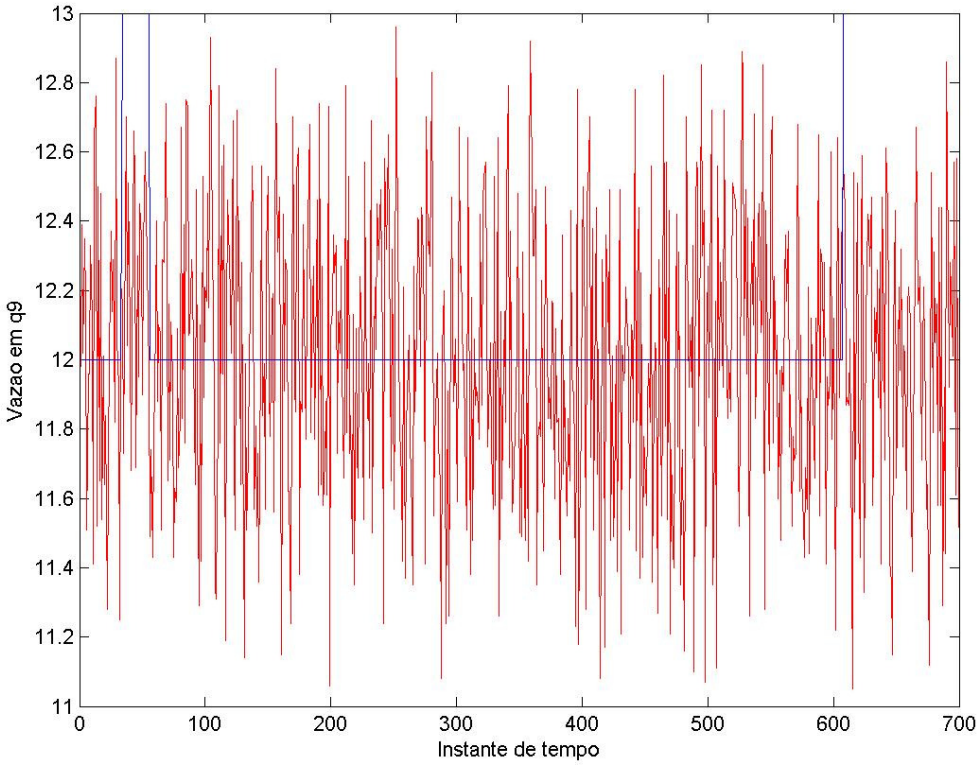

Figura C.9: Gráfico vazão por instante de tempo, para a situação 2 (B=20, J=10);

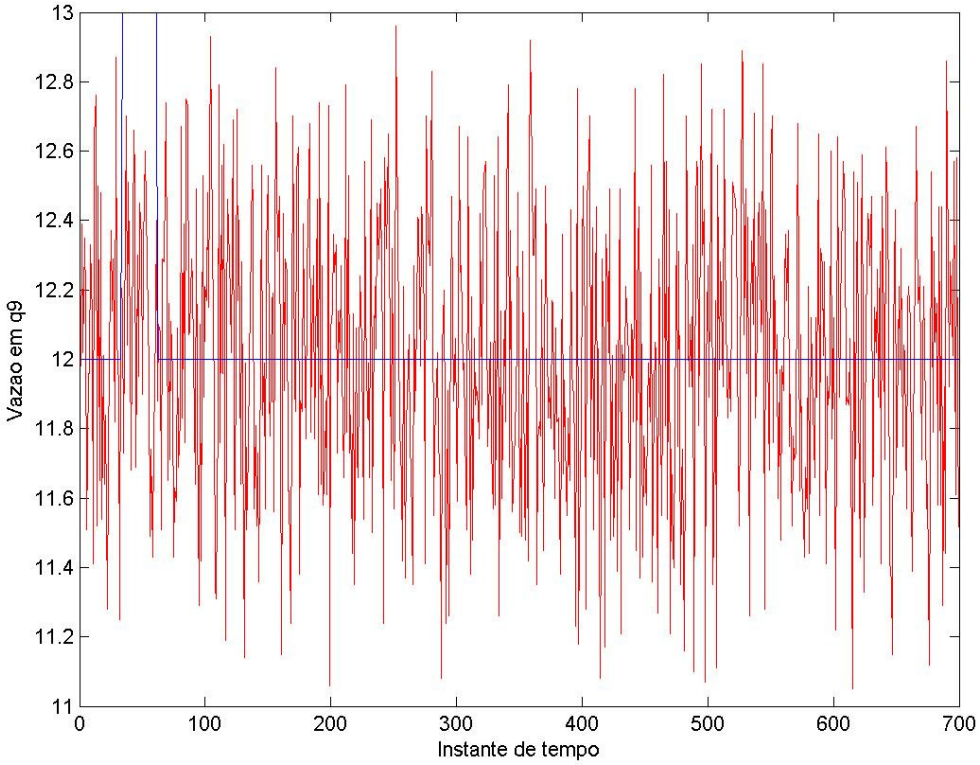

Figura C.10: Gráfico vazão por instante de tempo, para a situação 2 (B=100, J=10);

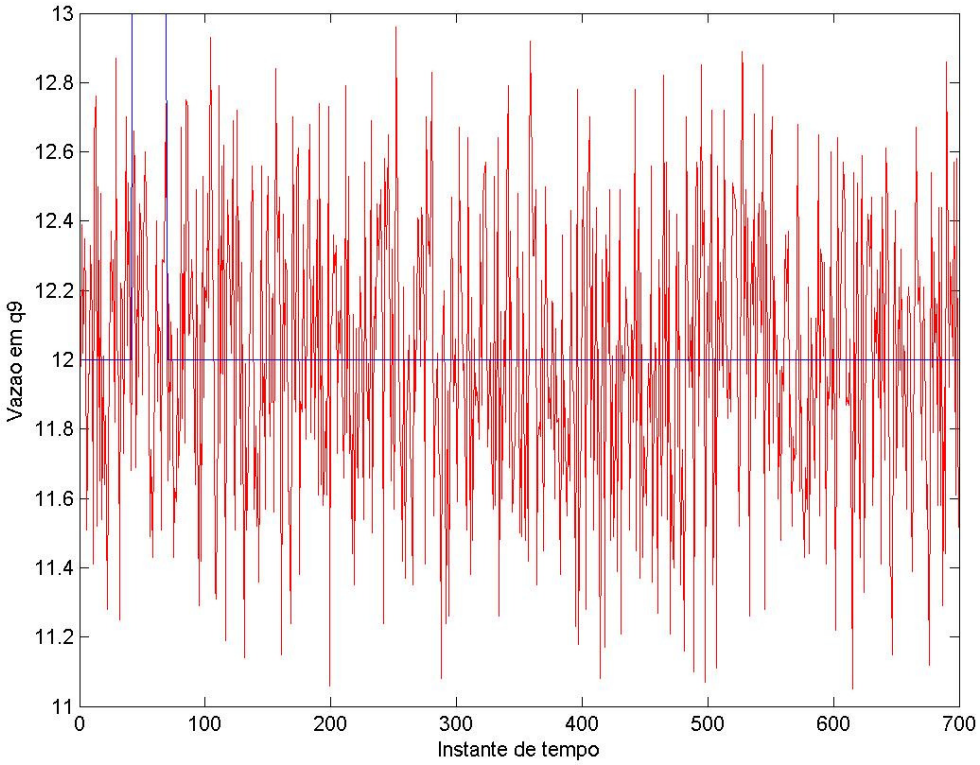

Figura C.11: Gráfico vazão por instante de tempo, para a situação 2 (B=200, J=10);

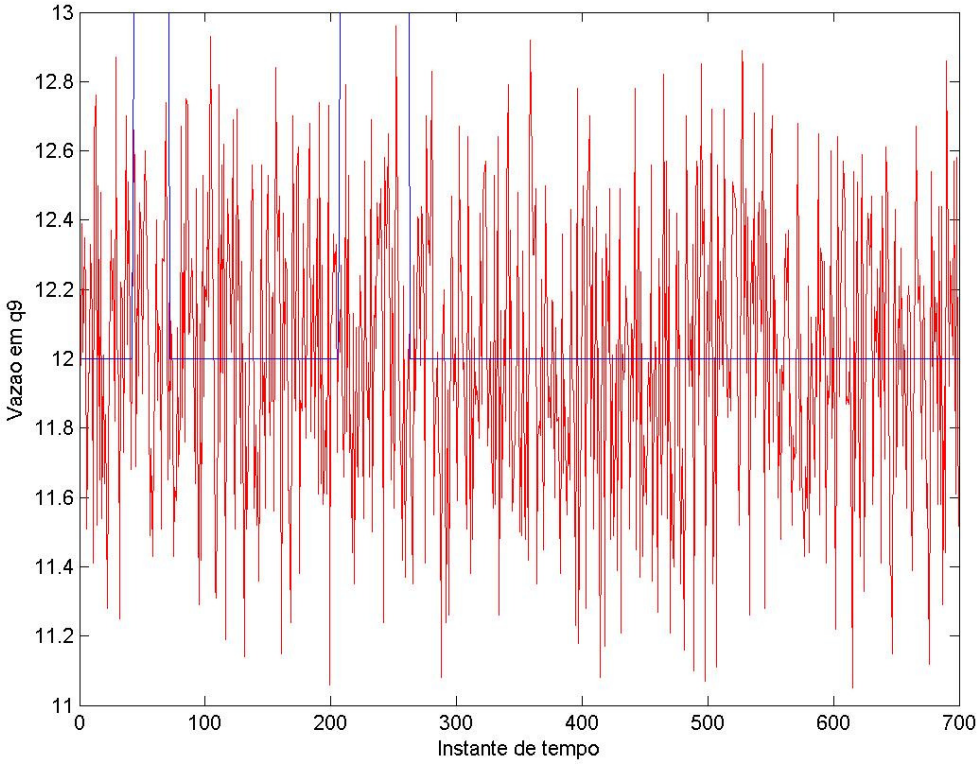

Figura C.12: Gráfico vazão por instante de tempo, para a situação 2 (B=400, J=10);

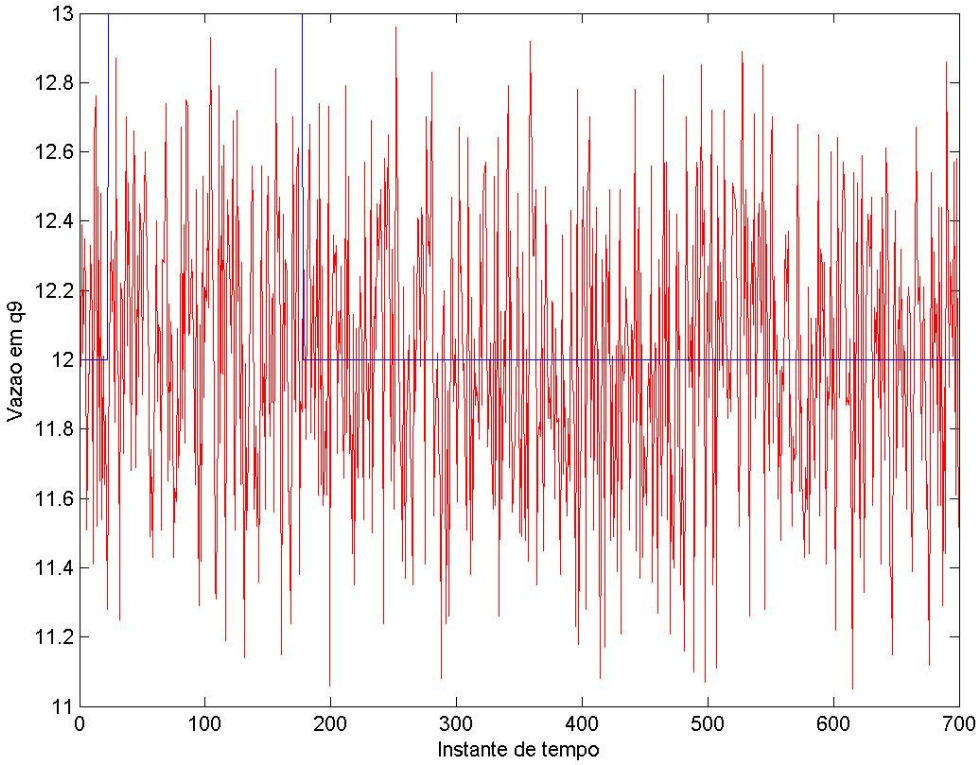

Figura C.13: Gráfico vazão por instante de tempo, para a situação 2 (B=20, J=20);

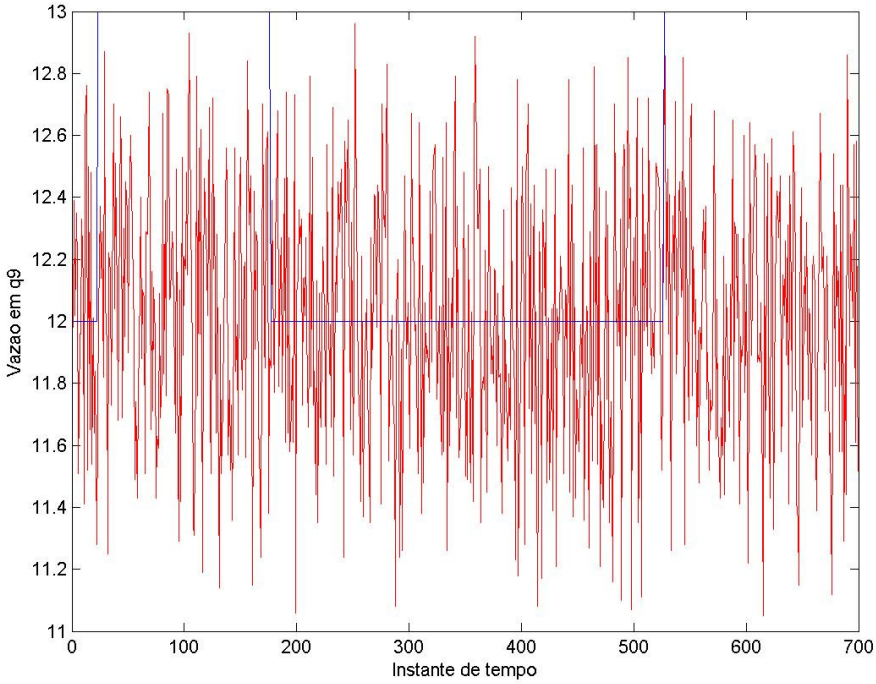

Figura C.14: Gráfico vazão por instante de tempo, para a situação 2 (B=100, J=20);

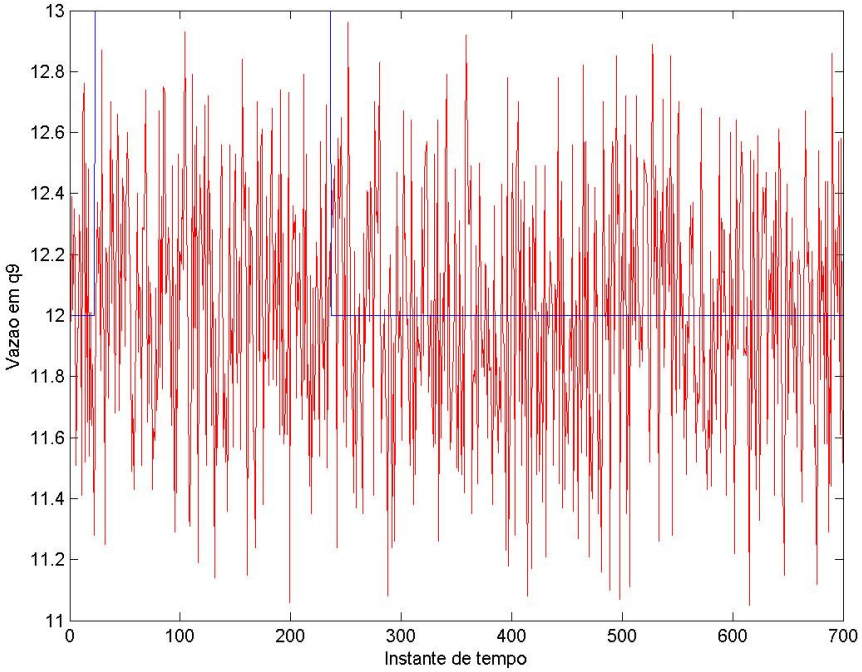

Figura C.15: Gráfico vazão por instante de tempo, para a situação 2 (B=200, J=20);

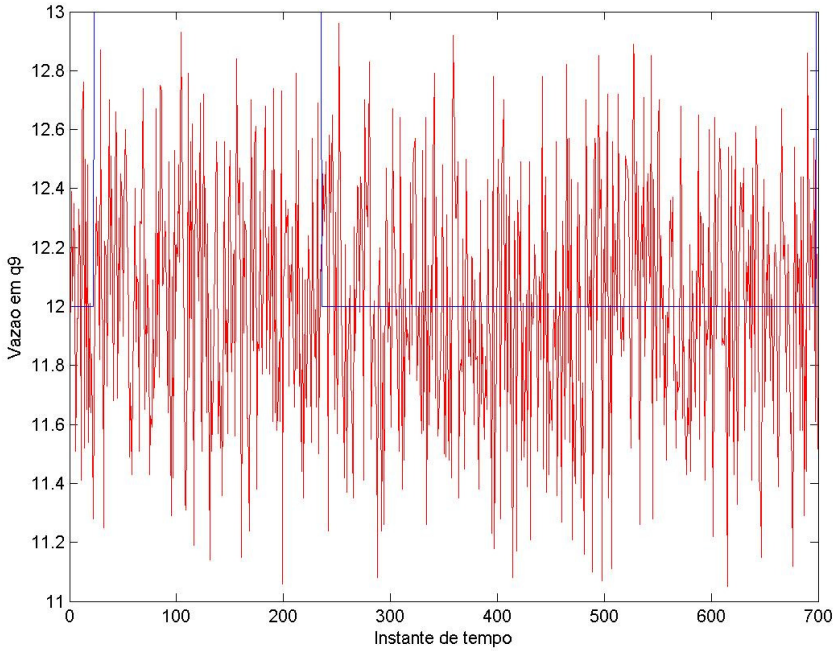

Figura C.16: Gráfico vazão por instante de tempo, para a situação 2 (B=400, J=20);

[i] Para os gráficos deste anexo, considerar-se-á B como sendo o número de repetições do *Bootstrap* e J como sendo o tamanho da janela.