Universidade Estadual de Londrina

MARIO HENRIQUE A. C. ADANIYA

DETECÇÃO DE ANOMALIAS UTILIZANDO FIREFLY HARMONIC CLUSTERING ALGORITHM E ASSINATURA DIGITAL DE SEGMENTO DE REDE

Londrina
2012

MARIO HENRIQUE A. C. ADANIYA

DETECÇÃO DE ANOMALIAS UTILIZANDO FIREFLY HARMONIC CLUSTERING ALGORITHM E ASSINATURA DIGITAL DE SEGMENTO DE REDE

Dissertação apresentada a Universidade Estadual de Londrina como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação.

Orientador: Prof. Dr. Mario Lemes Proença Júnior.

Londrina
2012

Catalogação elaborada pela Divisão de Processos Técnicos da Biblioteca Central da Universidade Estadual de Londrina

Dados Internacionais de Catalogação-na-Publicação (CIP)

A221d Adaniya, Mario Henrique A. C.

Detecção de Anomalias utilizando *Firefly Harmonic Clustering Algorithm* e Assinatura Digital de Segmento de Rede. / Mário Henrique Akihiko da Costa Adaniya. – Londrina, 2012.

82 f.: il.

Orientador: Mario Lemes Proença Junior.

Dissertação (Mestrado em Ciência da Computação) – Universidade Estadual de Londrina, Centro de Ciências Exatas, Programa de Pós–Graduação em Ciência da Computação, 2011.

Inclui bibliografia.

1. Algoritmos de computador – Teses. 2. Redes de computadores – Medidas de segurança – Teses. 3. Computação – Teses. I. Proença Junior, Mario Lemes. II. Universidade Estadual de Londrina. Centro de Ciências Exatas. Programa de Pós-Graduação em Ciência da Computação. III. Título.

CDU 519.68.021

MARIO HENRIQUE A. C. ADANIYA

DETECÇÃO DE ANOMALIAS UTILIZANDO FIREFLY HARMONIC CLUSTERING ALGORITHM E ASSINATURA DIGITAL DE SEGMENTO DE REDE

Dissertação apresentada a Universidade Estadual de Londrina como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação.

BANCA EXAMINADORA

_____

Orientador. Prof. Dr. Mario Lemes Proença Jr.
Universidade Estadual de Londrina - UEL

_____

Prof. Dr. José Valdeni de Lima
Universidade Federal do Rio Grande do Sul - UFRGS

_____

Prof. Dr. Jacques Duílio Brancher
Universidade Estadual de Londrina - UEL

_____

Prof Dr. Taufik Abrão
Universidade Estadual de Londrina - UEL

Londrina, 15 de fevereiro de 2012.

Agradecimentos

A Deus pela presença constante em minha vida.

Ao meu orientador Prof. Dr. Mario Lemes Proença Jr. por toda a paciência e dedicação durante a orientação.

Aos meus pais, por tudo que me ensinaram, ensinam e ensinarão. Serei grato eternamente.

A minha família, por ser um pilar importante na minha vida.

A minha namoradinha, pelo companheirismo durante a jornada.

Ao Prof. Dr. Taufik Abrão, Prof. Dr. Rodolfo Miranda de Barros e demais docentes que fizeram parte de um aprendizado importante na minha vida.

Aos amigos e colegas, de longa data ou que conheci durante esta jornada, em especial ao Lucas Dias H. Sampaio, Alexandro M. Zacaron, Rafael Herrera, Gabriel Ulian Briganó, Álvaro Souza, Fábio Engel e Rodrigo Luna. Aos que não foram nominados, a importância que tiveram

em algum momento não é menor ou maior, e peço sinceras desculpas.

Ao pessoal da JCI Londrina, por me fazer lutar, e acreditar que ainda há esperanças numa sociedade justa e equilibrada.

A todos os funcionários do Departamento de Computação e da Universidade que contribuíram para a realização deste trabalho.

A CAPES pela bolsa concedida.

Aos meus pais e familiares

"Stay hungry,
stay foolish."
Steve Jobs

ADANIYA, Mario H. A. C. **Detecção de Anomalias utilizando Firefly Harmonic Clustering Algorithm e Assinatura Digital de Segmento de Rede**. 2011. 81f. Dissertação (Mestrado em Ciência da Computação) - Universidade Estadual de Londrina, Londrina. 2011.

RESUMO

As redes deixaram de ser meras ferramentas e tornaram-se parte vital de qualquer empresa, provedor de serviço entre outros elementos. Isto torna a atividade de gerenciamento muito importante, uma vez que a saúde financeira e o próprio desenvolvimento das empresas estão conectados através das redes e qualquer interrupção ou falha de serviço acarretam em situações dramáticas. Neste cenário, as anomalias são grande geradoras de transtornos para os administratores de redes e oriundas de diversas causas, como falhas de configurações, erros por falha do usuário, erros por falhas de software e/ou hardware ou ataque de agentes maliciosos. Para mitigar os erros e falhas, é proposto nesta dissertação um modelo para detecção de anomalias baseado no algoritmo de clusterização otimizado aplicado aos dados coletados pela ferramenta Gerenciamento de Backbone Automático (GBA). Uma contribuição deste trabalho é o algoritmo de clusterização otimizado, chamado de Algoritmo de Clusterização Firefly Harmonic. Este algoritmo é a junção do algoritmo de clusterização, K-Harmonic Means (KHM), com a heurística, Firefly Algorithm. Também é apresentado uma descrição de anomalia que utiliza a Assinatura Digital de Segmento de Rede (DSNS) gerado pelo GBA, para criar uma área fixa que determina se o intervalo analisado é anômalo ou não, produzindo um gabarito. Para validação do modelo proposto, foram realizados testes com dados coletados da rede da Universidade Estadual de Londrina. Os resultados obtidos se mostraram promissores em diferentes cenários apresentados.

Keywords: Detecção de anomalia. DSNS. Firefly algorithm. K-Harmonic means. K-means.

ADANIYA, Mario H. A. C. **Detecção de Anomalias utilizando Firefly Harmonic Clustering Algorithm e Assinatura Digital de Segmento de Rede.** 2011. 81f. Dissertação (Mestrado em Ciência da Computação) - Universidade Estadual de Londrina, Londrina. 2011.

ABSTRACT

Nowadays the networks are not just tools and play an important role in enterprises, service providers and others elements. The network management gain an important status, once the financial health and the development itself are connected through the networks and any interruption or system failure leads to dramatic situations. In this scenario, the anomalies are great problems generators to the network managers and provide from misconfiguration, users failures, software and/or hardware failures or attacks. To decrease the number of failures and errors, in this dissertation it is proposed a model for anomaly detection based on optimized clustering algorithm applied to collected data from Automatic Backbone Management (GBA) tool. A contribution of this work is the optimized clustering algorithm, named Firefly Harmonic Clustering Algorithm (FHCA). This algorithm is the joint K-Harmonic Means and the heuristic Firefly Algorithm. It is proposed an anomaly description which is built from a Digital Signature of Network Segment (DSNS) generated by GBA tool. The anomaly description creates a fixed area that determines if an analysed interval will be consider an anomaly or not, originating a template. To evaluate the model, tests have been carried out using real data collected from the State University of Londrina. The obtained results have shown promising in different scenarios presented.

Keywords: Anomaly detection. DSNS. Firefly algorithm. K-Harmonic. Means, K-means.

Lista de Figuras

Lista de Tabelas

Lista de Algoritmos

Lista de siglas e abreviaturas

ACC	Taxa de Acurácia
ACO	Otimização por Colônia de Formigas
BLGBA	Baseline para gerenciamento de backbone automático
DSNS	Digital Signature of Network Segment
FA	Firefly Algorithm
FHCA	Algoritmo de Clusterização Firefly Harmonic
FPR	Taxa de Falso Positivo
GBA	Gerenciamento de Backbone Automático
IETF	Internet Engineering Task Force
IP	Internet Protocol
MIB	Management Information Base
KM	K-means
KHM	K-Harmonic means
PRE	Taxa de Precisão
PSO-Cls	Particle Swarm Optimization with Clustering
SDA	Sistema de detecção de anomalias
SNMP	Simple Management Protocol
TCP	Transmission Control Protocol
TPR	Taxa de Verdadeiro Positivo ou Taxa de Detecção
UDP	User Datagram Protocol

Convenções e Lista de Símbolos

Na notação das equações, as seguintes convenções foram utilizadas:

- letras minúsculas em negrito expressam vetores, exemplo: **x** e **p**;
- letras maiúsculas em negrito expressam matrizes, exemplo: **X**;

Os seguintes símbolos foram utilizados:

Símbolo	*Descrição*
α	Coeficiente de aleatoriedade do *Firefly Algorithm*.
γ	Coeficiente de absorção de luz do *Firefly Algorithm*.
β	Coeficiente de atratividade do *Firefly Algorithm*.
λ	Limiar para que uma amostra de tráfego seja classificada como anomalia.
λ_{MAX}	Limiar máximo.
λ_{MIN}	Limiar mínimo.
Λ	variação aceitável do tráfego em relação ao DSNS.
Δ	Intervalo de tempo.
$std(\cdot)$ ou σ	Desvio padrão
$var(\cdot)$ ou σ^2	Variância
$mean(\cdot)$ ou μ	Média
max[$\cdot$]	Operador que retorna o maior valor.
min[$\cdot$]	Operador que retorna o menor valor.
L	Intensidade de luz.
L_c	Limite de cada classe.
$VO(i)$	Relação de variação de volume do intervalo i entre o tráfego e o DSNS.
c_j	Centróide j.
$m(c_j\|x_i)$	Função de associação do algoritmo *K-Harmonic Means*.
$w(x_i)$	Função de peso do algoritmo *K-Harmonic Means*.

Símbolo	*Descrição*
D	Número de dimensões.
I_t	Número máximo de iterações.
I	Intervalo de coleta.
J	Número total de intervalos.
M	Tamanho da população de vagalumes.
K	Número de clusters.
S	Número de amostras de um conjunto.

Sumário

1 INTRODUÇÃO

O gerenciamento de redes é uma atividade essencial para empresas, provedores de serviços e outros elementos para quem as redes se tornaram uma necessidade. A necessidade de gerenciamento é originada pelo crescimento contínuo das redes que introduziram uma extensa lista de serviços com diferentes maneiras de uso por usuários de diferentes perfis. O principal objetivo do gerenciamento é assegurar o menor número de falhas e possíveis vulnerabilidades para não afetar a operação das redes. Existem diversos fatores que podem conduzir para uma anomalia, como erros de configuração, utilizacão de maneira indevida ou sem conhecimento adequado por parte dos usuários, erros de programação, ataques maliciosos internos e externos, entre muitas outras causas. Uma anomalia pode ser interpretada como uma não-conformidade com os dados normais e, dependendo do campo, é referido como anomalia, *outlier*, exceções, entre outros termos peculiares do campo em estudo. É de senso comum a utilização dos termos anomalia e *outlier* quando tratamos anomalias em redes de computadores [4].

Uma ferramenta para auxiliar a tarefa de gerenciamento de redes é o Sistema de Detecção de Anamolias (SDA). A detecção de anomalias é um tema pesquisado por diversos campos, como a estatística, aprendizagem de máquina, teoria da informação, mineração de dados, entre outros, que geram inúmeras técnicas adotadas para a detecção de anomalias. Em uma visão ampla, o objetivo é traçar um perfil do comportamento normal, chamado *baseline* [4], e confrontar para determinar se ocorrem diferenças ou semelhanças com as amostras coletadas. Consequentemente, uma atividade que desvia do *baseline* é tratada como uma possível intrusão ou anomalia [5].

De acordo com Hodge et. al [6], é possível classificar as técnicas de detecção de anomalias em três abordagens principais: técnicas supervisionadas, técnicas semi-supervisionadas e técnicas não supervisionadas. Técnicas supervisionadas necessitam tanto do dados normais quanto anormais, técnicas semi-supervisionadas necessitam apenas dos dados normais rotulados, e técnicas não supervisionadas não exigem conhecimento prévio dos dados.

Um SDA pode conter uma base de dados com assinaturas de ataques e anomalias, o que aumenta sua precisão e eficácia no suporte ao gerenciamento. Mas ao adotar uma base de dados, esta precisa de uma atualização constante pois os ataques alteram seu modo de operação constantemente, e consequentemente a sua assinatura também é alterada. Quando o SDA adota a não utilização de base de dados, existe a possibilidade de detecção de ataques previamente desconhecida [4]. Um SDA não busca substituir o administrador de redes, mas auxiliar no processo da tomada de decisão mediante o conhecimento que o administrador possui acerca do comportamento e das necessidades da rede.

É possível simplificar a detecção de anomalias como a tarefa de classificar os dados como anomâlos os dados que não pertencem a região adotada como normal. Através desta simplificação, são listadas algumas dificuldades apresentadas abaixo e as contribuições propostas às discussões do problema:

- Em muitas áreas o comportamento normal está em constante evolução, e a noção atual de comportamento normal pode não ser suficientemente representativo no futuro. Portanto, para modelar o comportamento normal, é adotado o modelo de Assinatura Digital de Segmento de Rede (DSNS) proposto por Proença et. al [7] e brevemente discutido no capítulo 3. A ferramenta de Gerenciamento de Backbone Automático (GBA) gera a predição do comportamento padrão de um determinado dia de um segmento de rede através de técnicas estatísticas, analisando algumas semanas antes, cada perfil gerado é chamado de DSNS.

- O obstáculo para definir a região considerada normal é muito desafiador, e o limiar entre o que define um comportamento anômalo de um comportamento normal é, muitas vezes, impreciso. A noção exata do que considerar anomalia também é diferente para cada aplicação. Para este fim, nesta dissertação é apresentada uma definição adotada para o contexto estudado, que pode ser estendida para outros cenários. Uma área é calculada e adotada como a variação aceita para o comportamento do tráfego, baseado no comportamento normal. A definição é discutida no capítulo 4.

- Para o problema de detecção de anomalia em si, é proposto um modelo utilizando um algoritmo de clusterização otimizada chamado Firefly Harmonic Clustering Algorithm (FHCA) aplicado aos dados coletados pela ferramenta GBA. O FHCA é a junção do algoritmo de clusterização K-Harmonic Means (KHM), proposto por Zhang [8], e uma heurística, Firefly Algorithm, proposta por Yang [9]. A junção se deve pela característica de algoritmos de clusterização baseados em centros não possuírem dispositivos para

escapar de ótimos locais; com isto, a heurística auxilia nesta função, resultando em uma resposta ótima ou muito próxima da ótima global. O modelo é discutido no capítulo 5.

Na detecção de anomalia, as técnicas de *clustering* são técnicas importantes, pois buscam encontrar padrões em dados não rotulados [4]. Segundo Hoghe et. al [6], podemos categorizar o modelo proposto nesta dissertação em técnicas semi-supervisionadas. Dito isto, adotamos o axioma de que o DSNS é o comportamento normal esperado da rede (dados rotulados) e queremos encontrar padrões no tráfego de rede (dados não rotulados) que sejam similares ou diferentes, para então inferir se os dados em determinado intervalo são anômalos ou não. Isto é alcançado com o modelo proposto que tem como base o algoritmo FHCA.

O restante deste trabalho está organizado da seguinte forma. O capítulo 2 apresenta os trabalhos relacionados ao desenvolvimento da pesquisa. No capítulo 3, é apresentado o modelo de caracterização de tráfego. No capítulo 4, a descrição de anomalia adotada pelo trabalho é discutida. O capítulo 5 apresenta o modelo de detecção de anomalia, descrevendo o conceito central do algoritmo Firefly Harmonic Clustering Algorithm (FHCA) e o sistema de geração de alarmes. Os resultados obtidos são apresentados no capítulo 6. No capítulo 7, temos as considerações finais acerca do trabalho e as futuras direções da pesquisa.

2 TRABALHOS RELACIONADOS

Neste capítulo, são apresentados os conceitos e trabalhos relacionados à pesquisa de anomalias em redes de computadores. São apresentados alguns tipos de anomalias e suas causas, as técnicas e diferentes métodos aplicados para detecção de anomalia, bem como um compilado de recentes propostas para detecção de anomalias em redes.

2.1 ANOMALIAS

Thottan et. al [10] assume duas categorias de anomalias. A primeira categoria está relacionada a falhas na rede e problemas de desempenho, onde inexiste o agente malicioso. Enquadra-se nesta categoria o *flash crowd*, onde um servidor recebe uma enorme quantidade de requisições de clientes não maliciosos em um mesmo período de tempo; por exemplo, o dia do anúncio do resultado do vestibular na própria universidade. Se a universidade não preparar uma infraestrutura para suportar o acesso dos milhares de vestibulandos, o servidor pode interromper suas operações, o que ocasiona um congestionamento no servidor *web*. O próprio congestionamento é outra forma de anomalia que se enquadra nesta categoria, devido ao aumento brusco do tráfego em determinado ponto da rede, acarretando em atrasos na entrega dos pacotes até a saturação do enlace, com descartes de pacotes. O congestionamento também pode ser gerado por erros de configuração; neste caso, o servidor não responde adequadamente às requisições enviadas pelos clientes pelas configurações feitas de maneira errada.

Na segunda categoria, são enquadradas as anomalias que surgem por problemas relacionados à segurança. Ataques de negação de serviço ou *Denial of Service* (DoS) estão enquadrados nesta categoria. DoS é definido quando um usuário não consegue obter determinado serviço pois algum agente malicioso utilizou de alguns métodos de ataque que ocuparam os recursos da máquina como CPU, memória RAM. Além do DoS, temos também o ataque distribuido de negação de serviço ou *Distributed Denial of Service* (DDoS) onde uma máquina mestre domina outras máquina zumbis para praticar um DoS [11]. DoS e

DDoS se diferenciam de *flash crowd* por causa do agente malicioso. Os vermes ou *worms* são definidos como processos que possuem a habilidade de criar uma cópia de si mesmo e executar em outras máquinas [12]. O escaneamento de portas ou *port scan* são técnicas utilizadas para executar tarefas de gêrencia, mas agentes maliciosos utilizam para descobrir vulnerabilidades nas redes [13]. Estas anomalias costumam gerar um volume de tráfego maior que o usual.

2.2 TÉCNICAS DE DETECÇÃO DE ANOMALIAS

As técnicas implementadas nos Sistema de Detecção de Anomalias (SDA) estão presentes em diversas áreas como: detecção de intrusão, detecção de fraude, detecção de anomalia médica, prevenção de danos industriais, processamento de imagem, redes de sensores, entre outros [5]. Por apresentar tantos domínios de aplicação diferente, muitas ferramentas foram desenvolvidas especificamente para alguma atividade e outras soluções são mais genéricas. Segundo Chandola et. al [5], as técnicas podem ser agrupadas em: baseado em classificação, clusterização, teoria da informação, estatistico e teoria espectral. Os trabalhos [5, 6] apresentam um *survey* mais generalizado, mas é possível encontrar na literatura *surveys* voltados para a área de rede de computadores, como em [4, 14]. A nomemclatura e algumas categorias podem ser diferentes, mas o conceito apresentado é consistente entre si.

Patcha et. al [4] divide as técnicas de detecção de anomalias em três categorias: baseados em assinatura, baseados na caracterização do comportamento normal e técnicas híbridas. As técnicas baseadas em assinatura fundamentam-se no conjunto de assinaturas de padrões de ataques conhecidos ou construídos. Um dos pontos fortes desta forma de detecção é a baixa taxa de falsos positivos. As técnicas baseadas em caracterização do comportamento normal do tráfego constroem o perfil do tráfego de rede, e qualquer evento que desvie do comportamento normal é considerado anomalia. As técnicas híbridas são uma junção das duas técnicas anteriores [4].

Muitos autores consideram o trabalho proposto por Denning [15] como um divisor de águas entre os métodos baseados em assinatura e os métodos que utilizam a caracterização do comportamento normal do tráfego, sendo que estes métodos consistem em duas fases: fase de treinamento e a fase de teste; na fase de treinamento é gerado o perfil da rede e na fase de teste é aplicado o perfil obtido para avaliação. O trabalho apresentado nesta dissertação enquadra-se na categoria de métodos baseados na caracterização do comportamento normal, porque adotamos a *Assinatura Digital de Segmento de Rede* (DSNS) gerado pela ferramenta de Gerenciamento de *Backbone* Automático (GBA) como

o comportamento normal do tráfego.

2.2.1 Detecção baseada nas assinaturas das anomalias

As técnicas de detecção baseadas nas assinaturas das anomalias necessitam da construção de uma base de dados com os eventos relacionados a determinadas anomalias, gerando assim as assinaturas. As assinaturas descrevem eventos específicos que formam um ataque ou anomalia específicos; desta forma, quando a ferramenta monitora o comportamento do tráfego, a comparação com as assinaturas é realizada, e se ocorrer uma correspondência de eventos como o descrito na assinatura, é gerado um alarme [4].

Utilizando-se de assinaturas, a metodologia apresenta baixas taxas de falso positivos, uma vez que a assinatura descreve de forma clara o que é necessário para ser considerado uma anomalia, em contrapartida, ataques com características desconhecidas e não formuladas nas assinaturas passam desapercebidas. Outro ponto negativo é a necessidade de constante atualização da base de dados das assinaturas [4].

2.2.2 Detecção baseada na caracterização do comportamento normal

Ao contrário da detecção através de assinaturas, o foco deste método é detectar anomalias baseadas na caracterização do comportamento normal. O primeiro e fundamental passo é gerar o comportamento normal do tráfego, chamado *baseline*, ou adotar algum modelo que descreve de maneira mais precisa o *baseline*. Consequentemente, qualquer atividade monitorada que desviar do *baseline* construído será considerada anomalia. A construção do *baseline* pode ser estática ou dinâmica. Estática, quando o *baseline* é construído e substituído apenas quando um novo *baseline* é construído; dinâmico, quando o *baseline* é atualizado conforme o comportamento da rede se altera.

Um ponto positivo é a possibilidade de detecção de novas anomalias, considerando que estas novas anomalias descreverão um comportamento diferente do normal. Outro ponto notório é a dificuldade criada para o agente malicioso conceber um ataque, porque ele desconhece o *baseline* e com isto existe a possibilidade dele não conseguir simular um ataque descrevendo o *baseline* e gerando um alarme [4]. Mas existem as desvantagens na construção do *baseline* como o período de treinamento necessário ou a quantidade de informações na base de dados histórico. A dificuldade na caracterização do próprio tráfego gera uma alta porcentagem na taxa de falsos positivos, pois o SDA pode apontar muitas

variações naturais da rede como um comportamento anômalo.

Existem diversas técnicas, por isso, abaixo é enumerado algumas técnicas relevantes que enriquecem a discussão com diversas propostas diferentes:

- **Baseada em Técnicas de Aprendizado:** As soluções baseadas em técnicas de aprendizado possuem habilidade de aprender e melhorar a execução ao longo do tempo, porque o sistema altera a estratégia de execução baseado nos resultados anteriores. Redes *Bayesianas*, cadeias de *Markov*, redes neurais são técnicas aplicadas para a geração e detecção. A maior vantagem desta abordagem é a capacidade de detecção de anomalias desconhecidas e adaptação às mudanças no comportamento do ambiente monitorado, entretanto, essa adaptação necessita de uma grande quantidade de dados para a geração de um novo perfil [4].

- **Baseada em Especificação:** Estas soluções são contruídas por um *expert*, visto que as especificações do comportamento normal do sistema são manualmente desenvolvidas. Se o sistema for bem representado, a taxa de falso negativo será minimizada, evitando qualquer comportamento não previsto, mas podem aumentar se algum comportamento for esquecido ou não bem descrito. A técnica mais utilizada para esta tarefa são as máquinas de estado finito. Um ponto inconveniente nesta abordagem é o tempo e a complexidade para o desenvolvimento das soluções [14].

- **Baseada em Processamento de Sinais:** As técnicas mais comumente utilizadas são as transformadas de *Fourier*, *wavelets* e algoritmos como ARIMA (*Autoregressive Integrated Moving Average*). Apresenta como vantagem a adaptação às evoluções do ambiente monitorado e detecção de anomalias desconhecidas e um baixo período de treinamento. A complexidade é apresentada como uma desvantagem desta abordagem [16].

- **Baseada em Mineração de Dados:** As técnicas de Mineração de Dados costumam lidar com uma enorme quantidade de dados, buscando padrões para formar conjuntos de dados normais. Análise de Componentes Principais (PCA), algoritmos de clusterização são comumente empregados nestas soluções [4]. O método proposto encaixa-se nesta categoria, pois é utilizado o algoritmo de clusterização proposto, Algoritmo de Clusterização Firefly Harmonic (FHCA), para encontrar padrões nos dados do DSNS e amostras do tráfego coletados pela ferramenta GBA.

2.3 PROPOSTAS RECENTES

Como descrito na seção anterior, o trabalho apresentado nesta dissertação é baseado em técnicas de Mineração de Dados [4]. Muitos trabalhos têm proposto soluções que utilizam algoritmos de clusterização para detectar anomalias. Clusterização é a organização de padrões em grupos baseado na similaridade [17].

Em [18], os autores propõem a utilização do algoritmo *K-means* (KM) para a detecção de anomalias. KM é um algortimo tradicional muito utilizado. O conceito é basicamente dividir um conjunto de dados com n amostras, em um conjunto de k clusters, de modo que a detecção de anomalias ocorre através da avaliação da distância entre as novas amostras e os centroides. Os experimentos realizados demonstraram que a utilização do KM é eficiente no particionamento de um volume grande de dados, e no contexto da detecção de intrusão demonstrou resultados promissores.

Utilizando o KM e um algoritmo proposto chamado de KD, Zhang et. al [19] apresentam uma nova proposta para detecção de intrusões em redes, uma abordagem híbrida baseada em assinaturas e caracterização do comportamento normal. É utilizado um conjunto de dados para o treinamento, onde um *threshold* é utilizado como parâmetro para marcar como normal os clusters. Para classificar um novo registro d, são calculadas as distâncias com cada centroide gerado, agrupando d ao cluster mais próximo. Se esse cluster for classificado como normal, d também é classificado como normal, caso contrário é classificado como anomalia. Nos experimentos, KM superou KD nas intrusões individuais, porém, KD obteve um ganho significativo para intrusões mistas.

Em [20], os autores tratam do problema de *outliers* baseados na noção de distância. A abordagem proposta é baseada na distância Euclideana do k-ésimo vizinho mais perto de um ponto O. É criada uma lista de "grau de proximidade" do ponto O denotando a distância entre os k-ésimos vizinhos. Ordenando a lista e assumindo que os pontos do topo são *outliers*, eles desenvolveram um algoritmo altamente eficiente de partição. Particionando os dados de entrada em subgrupos, e eliminando partições inteiras assim que eles não resultam em partições com *outliers*, este algoritmo guloso apresenta um redução de tempo computacional substancial.

Em [21], os autores propõem *SFK-means* para detecção de anomalias de rede, uma união dos algoritmos KM, *Fuzzy K-Means* e Otimização por Enxame de Partículas (PSO). Adicionando a teoria de *Fuzzy*, os autores acrescentam valores que mensuram o grau de determinado ponto pertencer a um dado cluster e não a outro. O PSO é uma heurística

inspirada no comportamento de um bando de pássaros ou cardume de peixes atrás de alimentos. Pode-se definir uma heurística como sendo um algoritmo de busca baseado em uma função custo ou objetivo que examina um subespaço, geralmente muito menor que o espaço total, e que apresenta uma elevada probabilidade de encontrar a solução, ótimo global, do problema a partir de mecanismos de escape de ótimos locais.

Soluções utilizando heurísticas geralmente são empregadas para resolver problemas de otimização, apresentado vantagens e desvantagens. Um benefício ocorre quando a solução analítica do problema apresenta maior custo computacional, tendo a heurística um desempenho mais vantajoso. Estudos em algoritmos inspirados por comportamentos da natureza não são recentes e, de um modo interessante, as complexas interações sociais têm inspirado benéficas soluções, que resultam em respostas ótimas ou muito perto de ótimas - se comparadas com soluções existentes, i.e., *Simulated Annealing* (SA), Otimização por Enxame de Partículas (PSO), Otimização por Colônia de Formigas (ACO), entre outros algoritmos. Yang [22] prôpos um algoritmo recente, chamado *Firefly Algorithm* (FA), que é baseado no comportamento dos vagalumes.

O estudo do comportamento dos vagalumes não é só de interesse do campo biológico e inspira áreas como economia [23] e a detecção autônoma de grupos de rôbos [24], para citar alguns exemplos. Basicamente, o comportamento das luzes emitidas pelos vagalumes é mapeados de alguma forma. No campo da economia, os autores lidam com o problema de estimar o valor dinamicamente em mercados *online*. Um jeito diferente é proposto utilizando o valor de venda dos outros vendedores e o comportamento do comprador para estimar o valor atual de venda, ao invés de utilizar apenas valores históricos. É um problema complexo porque os outros vendedores atualizam as taxas de forma assíncrona e de acordo com suas próprias estratégias, tornando desafiadora a tarefa de estimar o preço [23].

Na área da robótica, Christensen et. al [24] propõem um algoritmo descentralizador para a detecção de falhas. Os robôs emitem flashes de luzes de forma síncrona, logo, o robô que emitir os flashes de luzes dessincronizados possivelmente apresentará falhas. O algoritmo foi aplicado em simulações e em um sistema real, dependendo do número de robôs, os tempos de sincronização são alterados de forma automática. Estas pesquisas são soluções inspiradas no comportamento natural dos vagalumes.

A utilização da heurística para otimização do cálculo da função custo dos algoritmos de clusterização é comumente encontrado na literatura. Em [25], onde os autores utilizam o *Bee Algorithm* para auxiliar no problema de ótimos locais do KM. Em [26, 27], os autores propõem a otimização do KM através do PSO: no primeiro trabalho o domínio

aplicado é a detecção de intrusão, já no segundo trabalho, é o enriquecimento na discussão de técnicas de Mineração de Dados. Um problema apontado pelos autores em [28, 8], em relação ao KM, é o problema de inicialização, pois, se os pontos de inicialização forem muito próximos, o algoritmo tende a convergir para um ótimo local.

Para solucionar o problema de inicialização, Zhang et. al [8] prôpos o *K-Harmonic Means* (KHM). Através da média harmônica, função de associação e uma função de peso, KHM demonstra ser insensível ao problema de inicialização. Mas como uma solução de clusterização baseado em centros, o KHM continua a apresentar o problema de não escapar de ótimos locais. Para tanto, em [28] é apresentada a junção de uma heurística, *Simulated Annealing* (SA), com o KHM resultando no algoritmo *Simulated Annealing K-Harmonic Clustering* (SAKHMC).

3 CARACTERIZAÇÃO DE TRÁFEGO

Neste capítulo, é descrito os dados coletados e o modelo utilizado para a caracterização do tráfego empregado para construção da definição de anomalia no contexto estudado, descrito no capítulo 4. O modelo de detecção de anomalia proposto é fundamentado nos devios do tráfego acerca do comportamento normal da rede.

3.1 DADOS DA REDE

As fontes de dados da rede que serão analisadas influenciam diretamente na performance do Sistema de Detecção de Anomalias (SDA), e os tipos de anomalias detectadas são diretamente ligados às fontes de dados. Em Thottan et. al [10], é apresentado uma classificação de acordo com a fonte dos dados: Estatísticas Coletadas por Sondas ou *Network Probes*, Estatísticas por Filtragem de Pacotes Baseados em Fluxos ou *Packet Filtering for Flow-Based Statistics*, Dados dos Protocolos de Roteamento ou *Data From Routing Protocols* e Dados dos Protocolos de Gerenciamento de Redes ou *Data From Network Management Protocols*.

Sondas são ferramentas especializadas que medem parâmetros específicos como perda de pacotes, podendo oferecer dados pouco expressivos para detecção de anomalias. Na abordagem utilizando fluxos, os dados são coletados por técnicas sofisticadas de amostra, coletando uma gama de parâmetros, como endereços de origem e destino, portas de origem e destino, provendo muito mais detalhes. Na abordagem utilizando-se protocolos de roteamento, é construído a topologia de rede obtendo os estados dos *links*, que refletem em pouca e limitada informação sobre a rede. E, por fim, a abordagem que utiliza os protocolos de gerenciamento de rede, pois fornecem informações sobre as estatísticas de tráfego de rede. Estes protocolos correspondem às contagens de tráfego no nível do dispositivo e são monitorados passivamente. A informação obtida pode não fornecer diretamente uma métrica de desempenho de tráfego, mas poderia ser usada para caracterizar o comportamento da rede e, portanto, utilizada para a detecção de anomalias de rede [10].

3.1.1 Gerenciamento com SNMP

De acordo com as categorias apresentadas por Thottan et. al [10], o modelo apresentado pode ser enquadrado na abordagem que utiliza Dados dos Protocolos de Gerenciamento de Redes. O protocolo de gerencimento adotado é o *Simple Network Management Protocol* (SNMP) [29, 30] definido pelo IETF como protocolo padrão no gerenciamento de redes TCP/IP. O SNMP funciona em um paradigma cliente-servidor, formado basicamente pelos componentes: o dispositivo gerenciado, o agente e o gerente.

Um nó da rede será um dispositivo gerenciado se contiver um agente SNMP. Por sua vez, o agente SNMP é o elemento responsável por reunir as informações do dispositivo e disponibilizar ao gerente. O gerente é responsável pela comunicação com os agentes na busca das informações dos nós. Um único gerente SNMP pode monitorar vários agentes SNMP que estão localizados em dispositivos de rede. O SNMP é definido como um protocolo da camada de aplicação e utiliza o protocolo de transporte UDP para efetuar a troca de dados entre agentes e gerentes .

O servidor SNMP mantém uma base de dados das variáveis chamado *Management Information Base* (MIB) [31]. Os grupos que as variáveis MIB estão organizadas são: *system, interfaces (if), address translation (at), internet protocol (ip), internet control message protocol (icmp), transmission control protocol (tcp), user datagram protocol (udp), exterior gateway protocol (egp)* e *simple network management protocol (snmp)*. Cada váriavel MIB descreve a funcionalidade de um protocolo específico do dispositivo de rede que estiver sendo monitorado; por exemplo, se a variável MIB escolhida for do grupo *if*, as informações serão relativas ao número de bytes trafegado, se a váriavel escolhida for do grupo *tcp*, as informações resgatadas serão do número de pacotes que foram enviadas ou recebidas [32].

As informações na MIB possuem um identificador do objeto que é escrito por números separados por pontos (por exemplo, 1.3.6.1.2.1.4.3). Estes identificadores são estruturados de forma hierárquica, em que cada número representa um nível; logo o identificador representa um caminho a ser percorrido pela árvore observada na figura 3.1.

O grupo *if* é responsável pelas informações da camada de enlace; o grupo *ip* na camada de rede; os grupos *tcp* e *udp* na camada de transporte. Dentro de cada um destes grupos, foram utilizados os seguintes objetos:

- *ifInOctets* : quantidade total de bytes recebidos pela interface;
- *ipInReceives* : quantidade total de datagramas IP recebidos pelo dispositivo de rede,

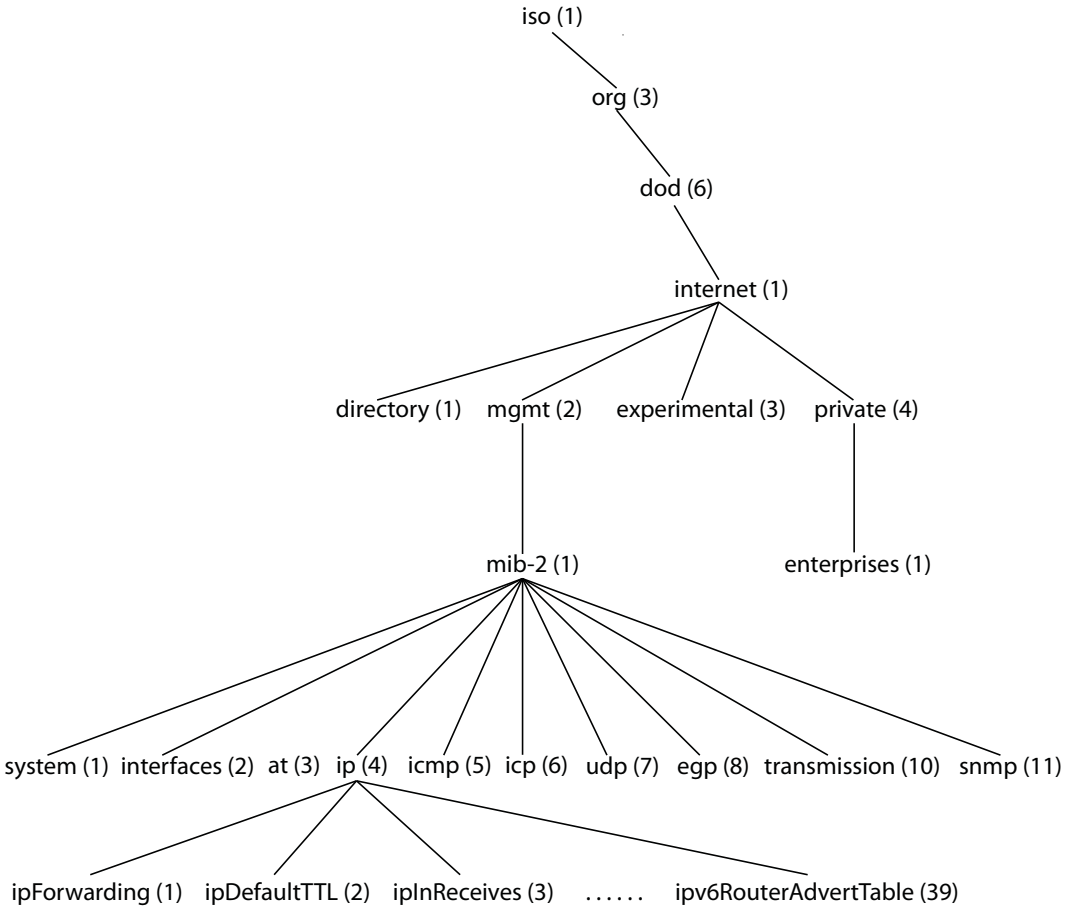

Figura 3.1: Informações da MIB dispostas de forma hierárquica em árvore.

incluindo os datagramas com erro;

- *ipInDelivers* : quantidade de datagramas IP entregues com sucesso aos protocolos clientes do protocolo IP;
- *tcpInSegs* : quantidade de segmentos TCP recebidos, incluindo aqueles que apresentam erros;
- *udpInDatagrams* : quantidade total de datagramas UDP recebidos e entregues à camada superior com sucesso.

3.2 BLGBA e DSNS

A caracterização ou predição do tráfego da rede é uma atividade desafiadora, devido à natureza comportamental inconstante influenciada por alguns fatores, como: a alocação dos recursos de rede é feita de forma dinâmica refletindo nas características, pequenos períodos de tempos com alta transferência de dados e a forte influência das horas de trabalho das pessoas [33]. Desta maneira, é adotado como comportamento padrão da rede o modelo gerado pela ferramenta de Gerenciamento de *Backbone* Automático (GBA), o *Baseline* para Gerenciamento de *Backbone* Automático (BLGBA) proposto por Proença et. al [34].

O BLGBA consiste na aplicação de técnicas de estátisticas para a criação de perfis normais do tráfego denominados Assinatura Digital de Segmento de Rede ou *Digital Signature of Network Segment* (DSNS), proposto por Proença Jr. [34]. Ao longo do dia, as informações obtidas como volume de tráfego, número de erros, tipos de protocolos e serviços que são transportados fornecem embasamento para a construção do DSNS.

O comportamento do tráfego é formado por ciclos distintos para cada dia da semana, e, para cada hora, o comportamento da rede é diretamente influenciado; diferenças são claramente visíveis quando comparados os tráfegos de dias úteis com os sábados, domingos e feriados [35]. Por isto a ferramenta GBA analisa cada segundo do dia individualmente para construir um DSNS mais próximo da realidade. Para construir um DSNS, é necessária a utilização de 4 a 12 semanas de dados históricos; porém, para se obter um DSNS mais preciso o tempo ideal é de 12 semanas, segundo resultados de testes de regressão linear, Bland e Altman e análise de resíduos [34].

O modelo BLGBA determina um valor esperado para cada segundo do dia analisando os valores para o mesmo segundo semanas anteriores, mas para a geração de um DSNS, é necessária antes a construção da matriz **M**, onde m_{ij} representa o valor da amostra de um dado dia i de determinado instante j. A matriz **M** é representada por (3.1):

$$\mathbf{M} = \begin{bmatrix} m_{11} & \dots & m_{1j} \\ \vdots & \ddots & \vdots \\ m_{i1} & \dots & m_{ij} \end{bmatrix} \tag{3.1}$$

O I é definido pelo intervalo de coleta, se o dispositivo de rede analisado coletar de 10 segundos em 10 segundos, teremos um total de $I = 86400/10 = 8640$ amostras em um dia. Se o histórico de dados para criar um *baseline* para um dado dia da semana adotado for de 8 semanas anteriores, uma matriz 8640 x 8 é obtida, considerando-se por exemplo,

todas as segundas-feiras anteriores. Construída a matriz **M**, é adotada uma classificação com 5 classes ($c = 1, \ldots, 5$) baseada na diferença entre o maior ($\max_i$) e menor ($\min_i$) elemento de cada linha, utilizado para calcular a amplitude, denotada pela equação (3.2).

$$A(i) = \frac{\max[m_{i1}, m_{i2}, \ldots, m_{ij}] + \min[m_{i1}, m_{i2}, \ldots, m_{ij}]}{5} \tag{3.2}$$

Através do calculo da amplitude, $A(i)$, obtemos o limite de cada classe, $L_c(c)$ que é obtido através da equação (3.3):

$$L_c(c) = \min[m_{i1}, m_{i2}, \ldots, m_{ij}] + (A(i) * c) \tag{3.3}$$

onde c representa a classe a ser avaliada. O valor com maior quantidade de elementos inserida na classe com freqüência acumulada igual ou superior a 80% é incluído no DSNS. Com isto, os valores mais representativos são utilizados para calcular o comportamento esperado, e os valores com freqüência menor são descartados por se tratarem de dados menos representativos para o resultado final.

A figura 3.2 apresenta gráficos dos dias úteis, do dia 01/03/2011 até 05/03/2011, ilustrando o tráfego e seu respectivo DSNS, gerados pela ferramenta GBA. Os dados foram coletados do objeto *tcpInSegs* que pertence ao grupo *tcp* da MIB-II [31], do servidor *proxy* da Universidade Estadual de Londrina (UEL). O movimento é representado em verde e o respectivo DSNS pela linha azul. A linha vermelha representa que o movimento excedeu o DSNS, mas não implica necessariamente que ocorreu uma anomalia naquele instante.

Na figura 3.2, a linha em azul, representando o DSNS, não distância-se das áreas em verde e vermelho. Pode-se observar um grande ajuste do comportamento do tráfego monitorado em relação ao DSNS. Na figura 3.3, é apresentado os gráficos do período de 08/11/2010 até 12/11/2010, para o objeto *ifInOctets* pertencente ao grupo *if* do servidor *web* da UEL.

As figuras 3.2 e 3.3, exemplificam a diferença comportamental dos diferentes objetos dos dados coletados para diferentes dias da mesma semana. Os dias úteis da semana são fortemente influenciados pelo horário de funcionamento da instituição, com um crescimento em torno das 8h da manhã e um declínio por volta das 12h, período do almoço. O tráfego retoma o crescimento às 13h e mantém um nível até decrescer novamente em torno das 18h. Como a universidade oferece cursos noturnos, o período de 18h às 22h apresenta um tráfego

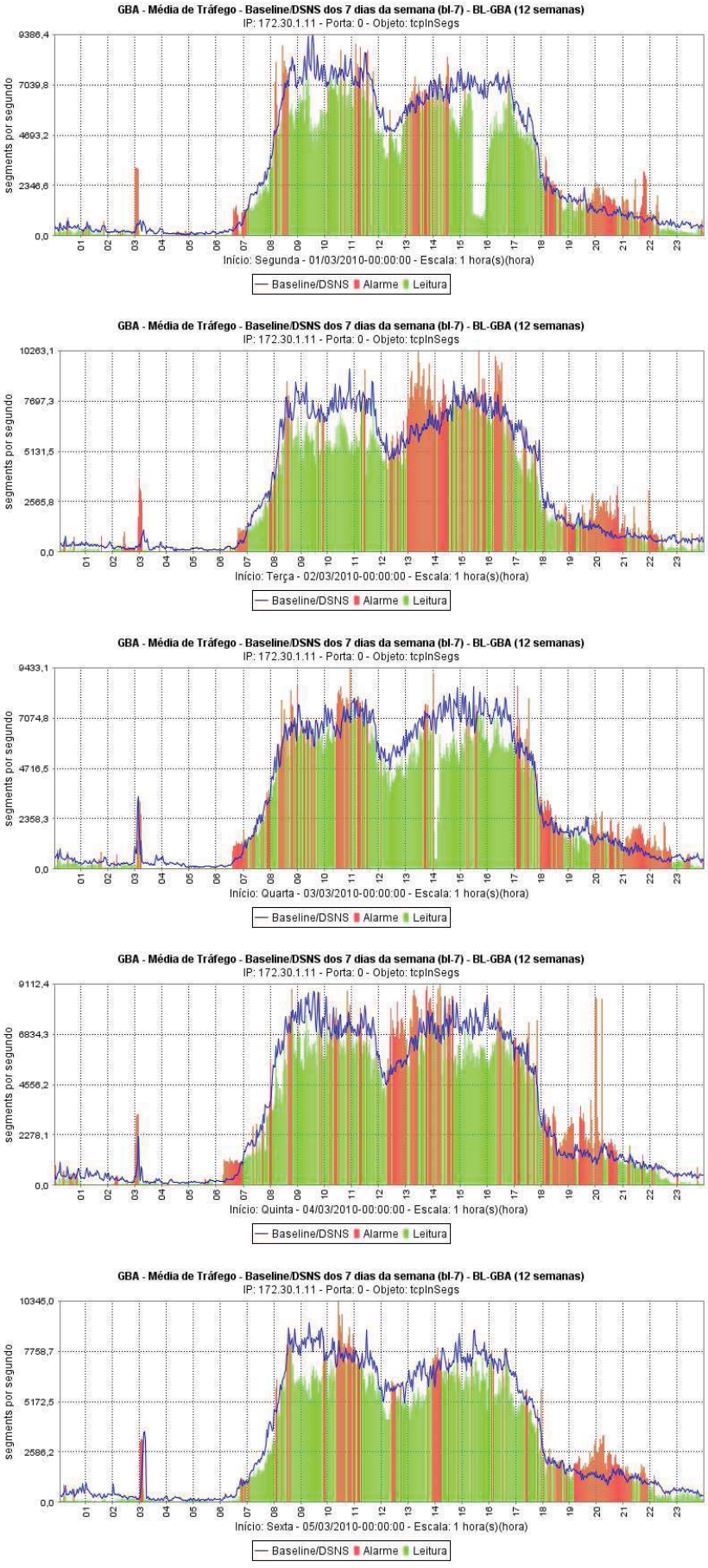

Figura 3.2: Tráfego e DSNS gerados pela ferramenta GBA para o período de 01/03/2011 até 05/03/2011 do objeto *tcpInSegs*.

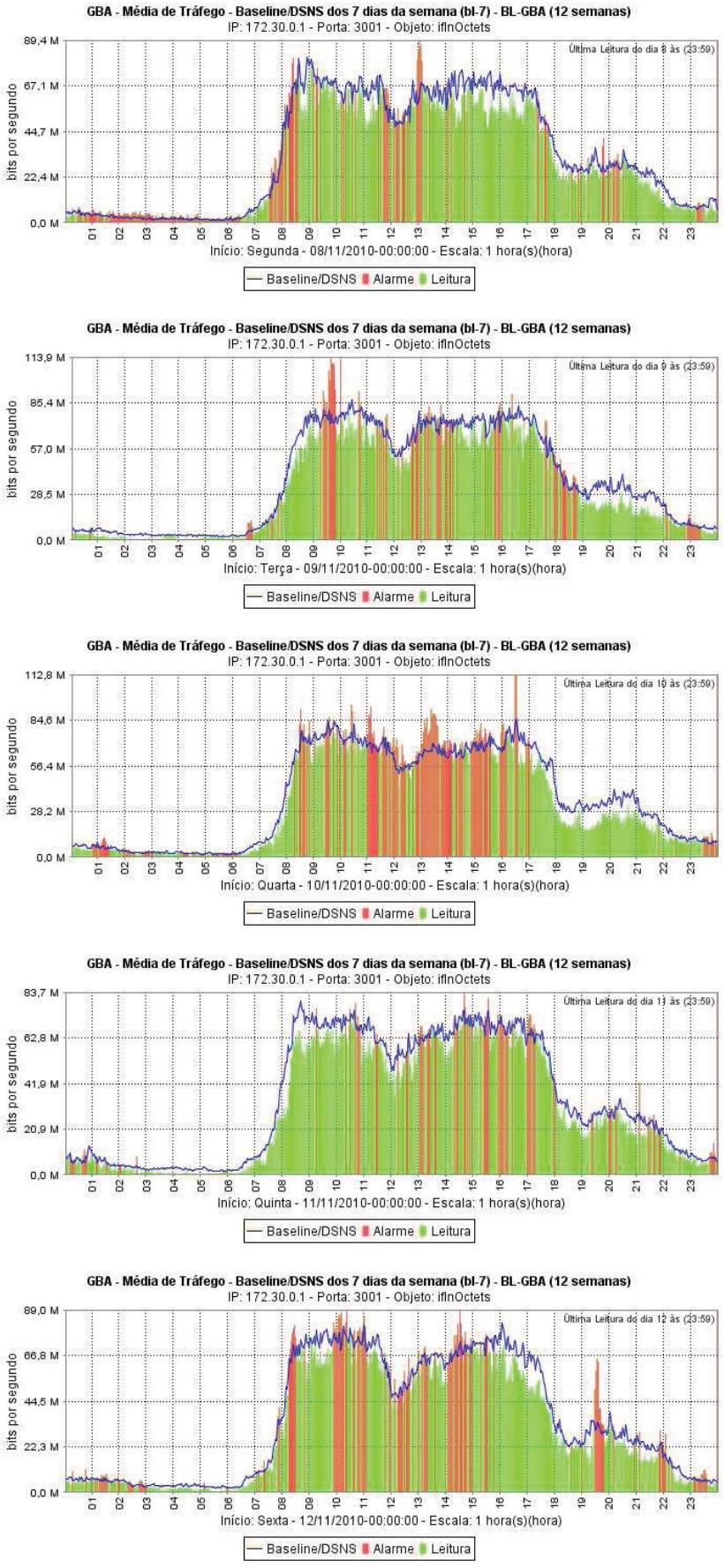

Figura 3.3: Tráfego e DSNS gerados pela ferramenta GBA para o período de 08/11/2010 até 12/11/2010 do objeto *ifInOctets*.

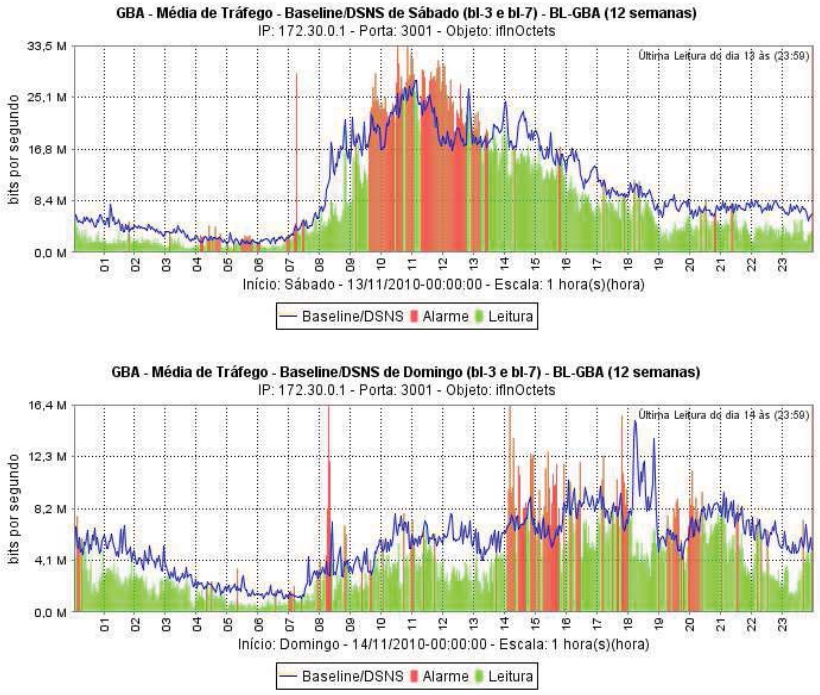

Figura 3.4: Tráfego e DSNS gerados pela ferramenta GBA para o final de semana do objeto *ifInOctets*.

relativamente baixo se comparado ao horário de funcionamento e alto se comparado ao período de 0h às 7h.

O sábado e o domingo apresentam um comportamento do tráfego diferente dos dias úteis da semana. A figura 3.4 apresenta os dias 13/11/2010 e 14/11/2010, sábado e domingo, respectivamente. O objeto coletado é o *ifInOctets* do grupo da camada de enlace *if*, do servidor de *proxy* da universidade.

É possivel apontar diferenças comportamentais claras entre o tráfego do final de semana e dias úteis, e entre os dias do próprio final de semana. No sábado o tráfego inicia um crescimento lento em torno das 7h da manhã, alcançando o ápice perto das 11h. O pico do volume alcançando no sábado é aproximadamente 25Mbits/s, e durante a semana, o pico alcança um valor perto de 70Mbits/s. A partir das 15h, tem inicio uma queda lenta do tráfego até uma estabilização as 19h. O comportamento do sábado é reflexo dos cursos, eventos, entre outras atividades que ocorrem. No domingo, temos um comportamento do volume abaixo de 10Mbits/s durante quase todo o período do dia.

A ferramenta GBA está organizada como apresentado na figura 3.5. O módulo **Coletor GBA** é responsável por coletar os dados dos roteadores, *switch*, servidores de *e-mail*, *web*, *proxy* e outros elementos da rede e armazenar na base de dados das amostras. Por sua vez, as amostras armazenadas são utilizadas pelo **GBA Gerador de DSNS**, que

através do algoritmo BLGBA calcula e cria o DSNS, que é armazenado na base de dados do DSNS. Por fim, temos o módulo de **Sistema de Alarme** onde encontramos o algoritmo proposto, discutido no capítulo 5, que utiliza os dados armazenados nas duas base de dados para a geração de alarmes para alertar o administrador de redes.

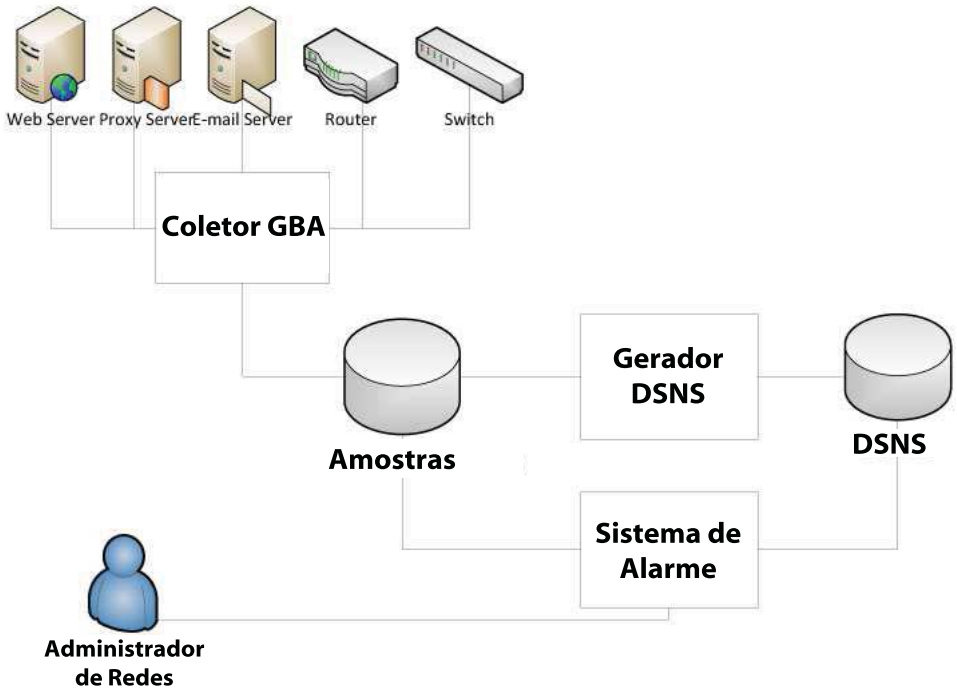

Figura 3.5: Estrutura funcional da ferramenta GBA.

4 CARACTERIZAÇÃO DE ANOMALIA

Neste capítulo, é apresentada a definição de anomalia adotada para criar um gabarito, tendo em vista a comparação com os intervalos classificados pelo Algoritmo de Clusterização *Firefly Harmonic* (FHCA). Este gabarito é gerado com os dados da Assinatura Digital de Segmento de Rede (DSNS), explicado no capítulo 3, e das amostras de tráfego do dia desejado.

Os dados do DSNS podem ser representados pelo vetor coluna $\mathbf{d}$, com dimensão $I \times 1$, onde I é o intervalo de coleta. O índice da posição referenciado pelo *timestamp* da coleta é i, e $\mathbf{d}(i)$ representa o valor do DSNS calculado para o tempo i. Para saber se determinado intervalo J será considerado anomalia ou não, definimos o intervalo como sendo Δ, onde Δ representa um montante de tempo a ser avaliado em segundos, uma vez que a ferramenta de Gerenciamento de *Backbone* Automático (GBA) lida com tempos na fração de segundos. Por exemplo, se quisermos analisar o intervalo de 5 minutos, $\Delta = 300$. Com a definição de Δ, trabalhamos com a divisão de $\mathbf{d}$ em J intervalos, representados por (4.1).

$$[\mathbf{d}_1, \ldots, \mathbf{d}_j, \ldots, \mathbf{d}_J]^\mathsf{T} = \mathbf{d}, \tag{4.1}$$

onde J é o número total de intervalos e é descrito da seguinte maneira: $J = I/\Delta$. Por exemplo, se o tempo de coleta adotado for de 1 segundo, em um dia temos 86400 segundos, assumindo intervalos de 5 minutos, $J = 86400/300 = 288$. $\mathbf{d}_j$ representa uma parte de $\mathbf{d}$ no intervalo de J.

Após a reestruturação dos dados, é importante apresentar o parâmetro λ adotado na definição para calcular um limiar que é utilizado para determinar se os dados contidos no j-ésimo intervalo são anômalos ou não. Este parâmetro representa um certo volume de variação considerado normal do tráfego pela própria natureza comportamental dos dados trabalhados. Na figura 4.1 é possível visualizar o DSNS, e uma variação λ do DSNS dentro de um intervalo Δ.

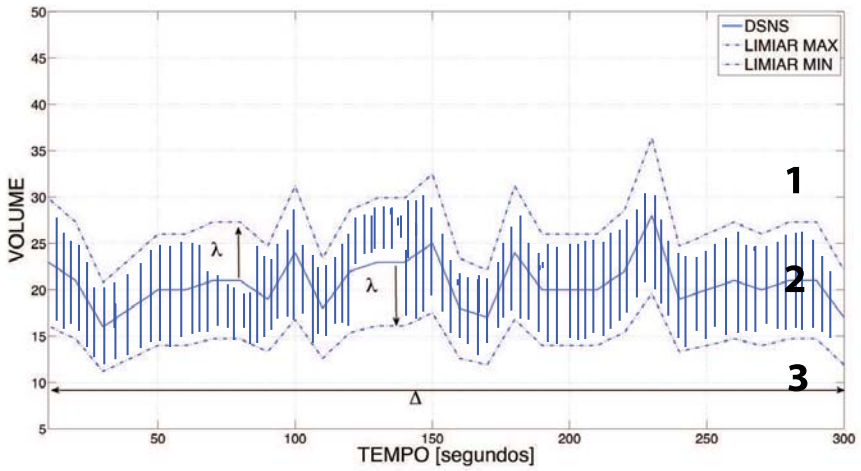

Figura 4.1: Variação do parâmetro λ em relação ao DSNS dentro de um intervalo Δ.

Com este limiar traçado, criamos uma área onde os dados do tráfego se localizam: 1 - acima do limiar máximo; 2 - entre o limiar mínimo e o limiar máximo, descrito pela área hachurada, 3 - abaixo do limiar mínimo. O ideia é mensurar o número de amostras que estão entre os dois limiares para inferir se determinado intervalo do tráfego em estudo vai ser considerado anômalo ou não. Podemos visualizar essa operação na figura 4.2.

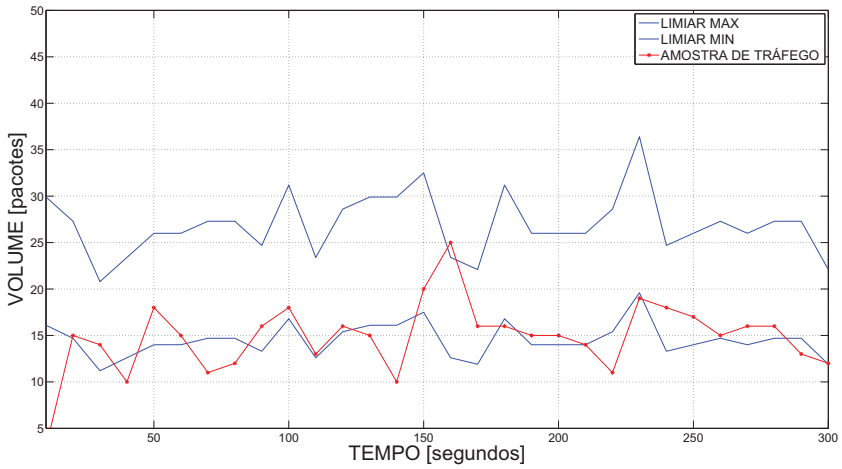

Figura 4.2: Amostra do tráfego de rede e os limiares mínimo e máximo.

O parâmetro λ pode assumir um valor constante baseado no conhecimento prévio da rede ou utilizando-se de uma medida estatística. Inicialmente, os testes adotados para determinar o valor de λ foram os cálculos da média e o desvio padrão, que descrevem as curvas da figura 4.3.

Na figura 4.3, é possivel visualizar que a área traçada pelos limiares calculado quando λ assume a média (linha azul escuro tracejada) é maior do que a área traçada quando

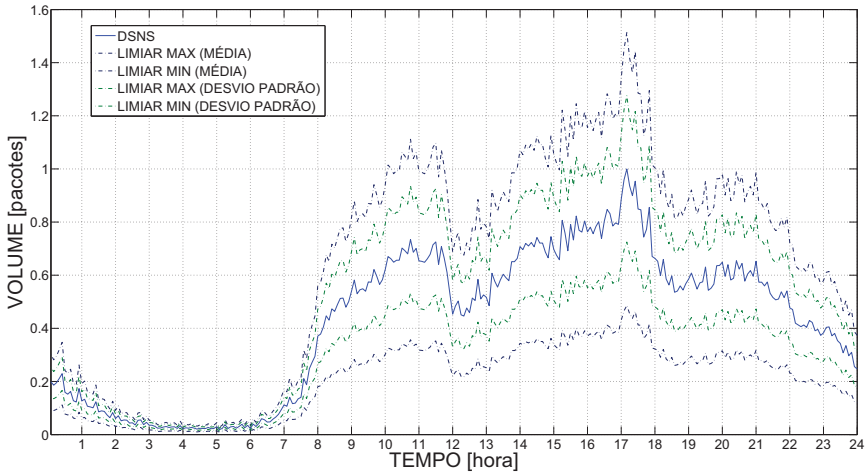

Figura 4.3: Limiares mínimo e máximo com λ assumindo o valor de média e desvio padrão.

λ assume o valor do desvio padrão (linha verde tracejada). No gerenciamento de redes, uma variabilidade nos dados é natural que ocorram, mas se assumirmos valores muito altos, podemos deixar passar muitos comportamentos que certamente podem ser uma ameaça ou falha. Entretanto, se assumirmos valores muito baixos, aceitaremos muitos comportamentos normais dentro de um intervalo como anomalia, gerando um número excessivo de alarmes. Para evitar um zelo excessivo ou uma falta de cuidado, em nosso trabalho, o valor de λ proposto é descrito por:

$$\lambda = \frac{\frac{1}{J}\sum_{j=1}^{J}\sigma(\mathbf{d}_j)}{\max[\sigma(\mathbf{d}_1),\ldots,\sigma(\mathbf{d}_j),\ldots,\sigma(\mathbf{d}_J)]}; \tag{4.2}$$

onde $\sigma(\cdot)$ é o desvio padrão do conjunto de dados no intervalo j de $\mathbf{d}_j$ e $\max[\cdot]$ é o operador que retorna o maior valor de σ de todos os intervalos do DSNS.

Não é esperado que todos os pontos do tráfego de rede tenham o mesmo valor do DSNS, mas o tráfego real deve seguir o DSNS com um desvio tolerável em escalas diferentes, uma vez que o DSNS descreve uma predição do tráfego. Na figura 4.4, as linhas pontilhadas representam o intervalo aceitável do DSNS. É possível observar que o tráfego (linha vermelha) segue, na maior parte do tempo, o DSNS (linha azul) e está dentro do intervalo na maior parte. Dependendo do segmento de rede e o objeto MIB coletado, os parâmetros podem variar por causa do volume. Por exemplo, em um servidor HTTP o tráfego é medido pelo endereço IP de destino e origem, e é diferente de um *Firewall*, onde todos os tráfegos passam antes de entrar e/ou sair de um segmento de rede. Como o volume passando pelo *Firewall* é maior do que em um servidor HTTP, o λ calculado também é diferente para cada um.

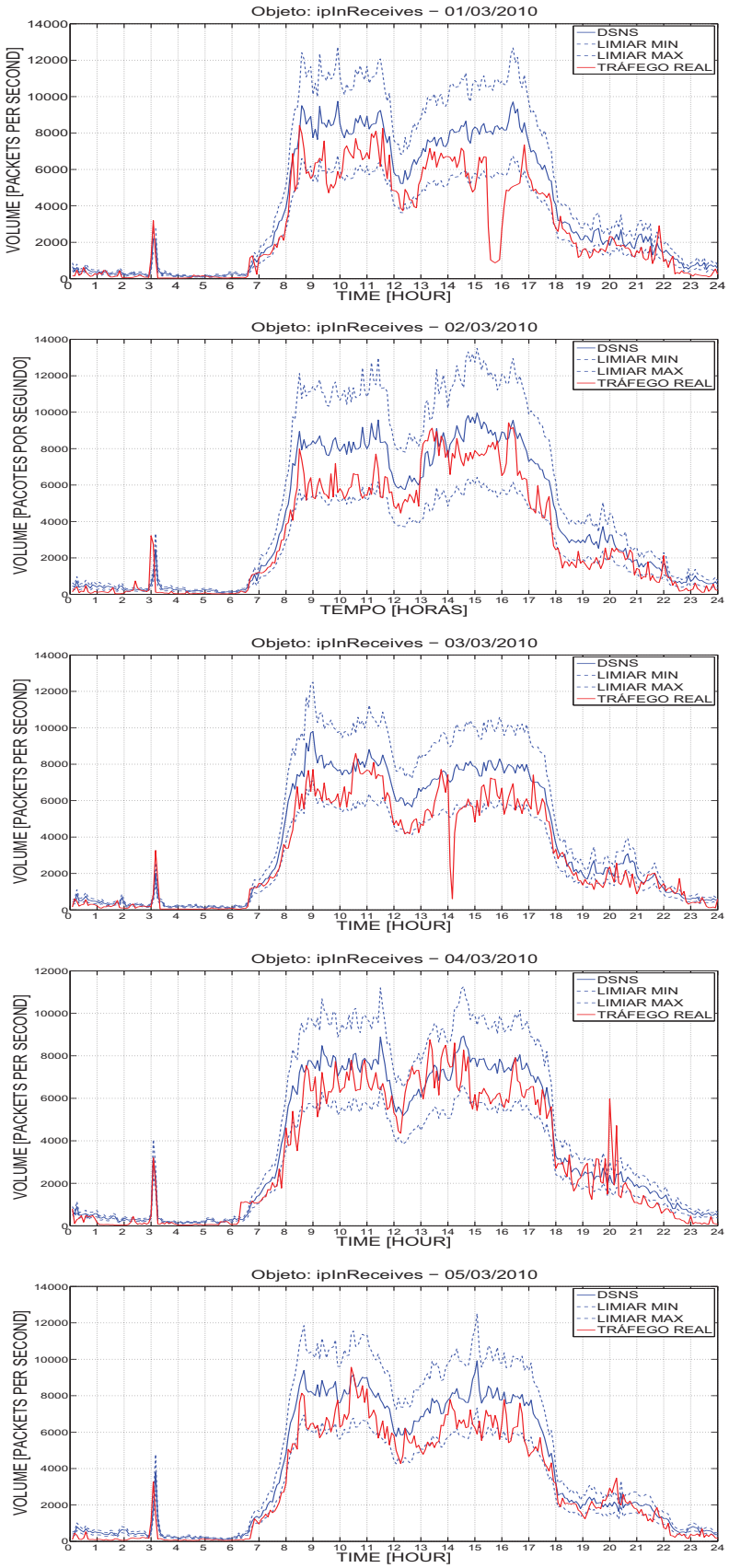

Figura 4.4: DSNS, Tráfego e limiares mínimo e máximo.

Para determinar se um $\mathbf{d}_j$ é anomalia ou não, a equação (4.3) descreve:

$$\mathbf{a}(j) = \begin{cases} 0, \lambda_{MIN} < \mathbf{d}_j < \lambda_{MAX} \\ 1, c.c. \end{cases} \tag{4.3}$$

$\mathbf{a}(j)$ é um vetor com valores booleanos para o $j-$ésimo intervalo. Ressaltando que este gabarito é utilizado para comparação entre os resultados obtidos dos algoritmos testados apresentados no capítulo 6. Os valores de λ_{MIN} e λ_{MAX} são obtidos através da equação (4.4).

$$\begin{cases} \lambda_{MIN} = \mathbf{d}_j * (1 - \lambda) \\ \lambda_{MAX} = \mathbf{d}_j * (1 + \lambda) \end{cases} \tag{4.4}$$

As figuras 4.5, 4.6, 4.7, 4.8 mostram em detalhes os intervalos do dia 02/03/2010, com gráficos apresentados de 1 em 1 hora, divididos em intervalos de 5 minutos.

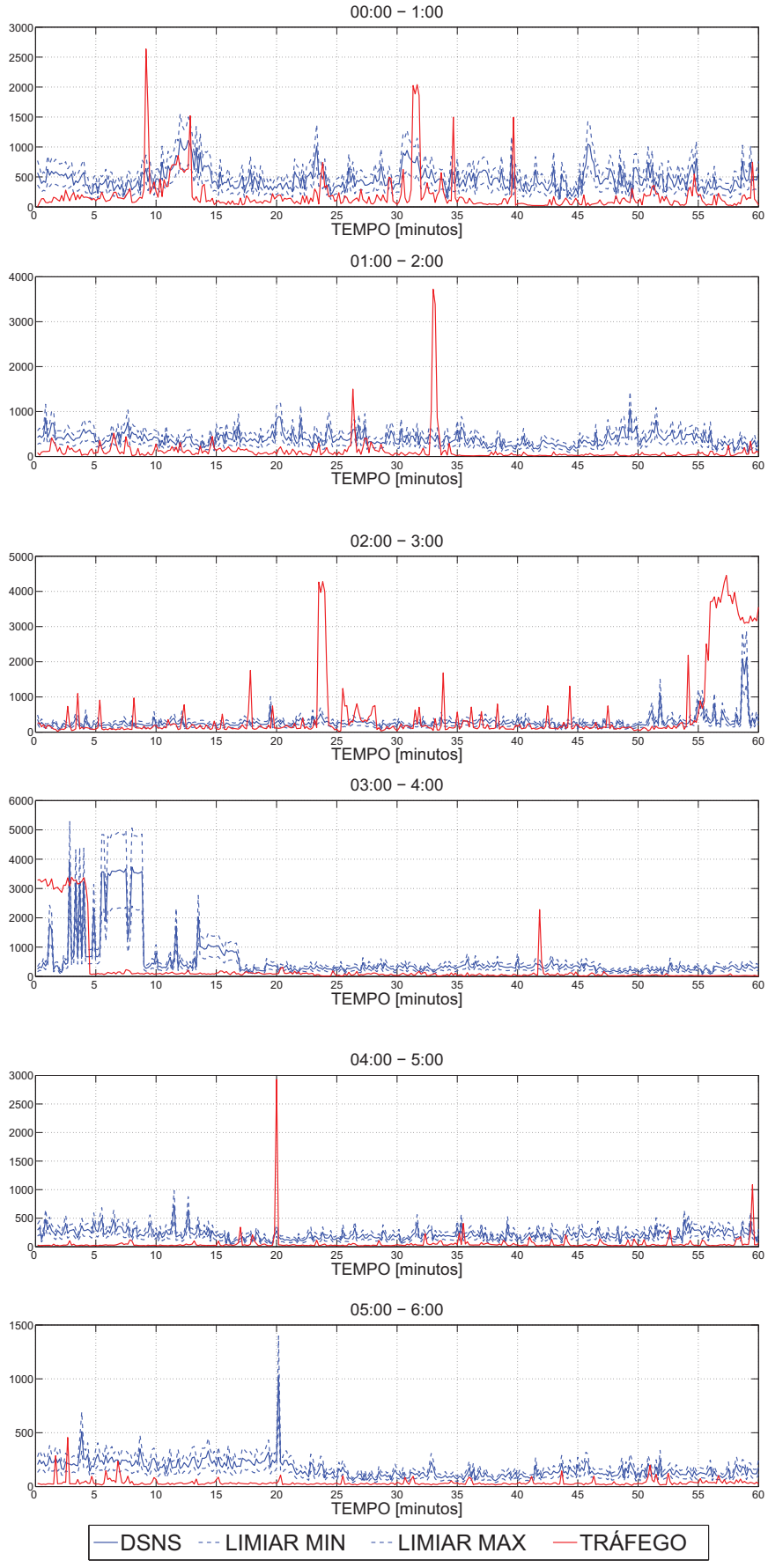

Figura 4.5: Detalhes do intervalo 03/02/2010 do objeto *ipInReceives entre 00:00 – 06:00.*

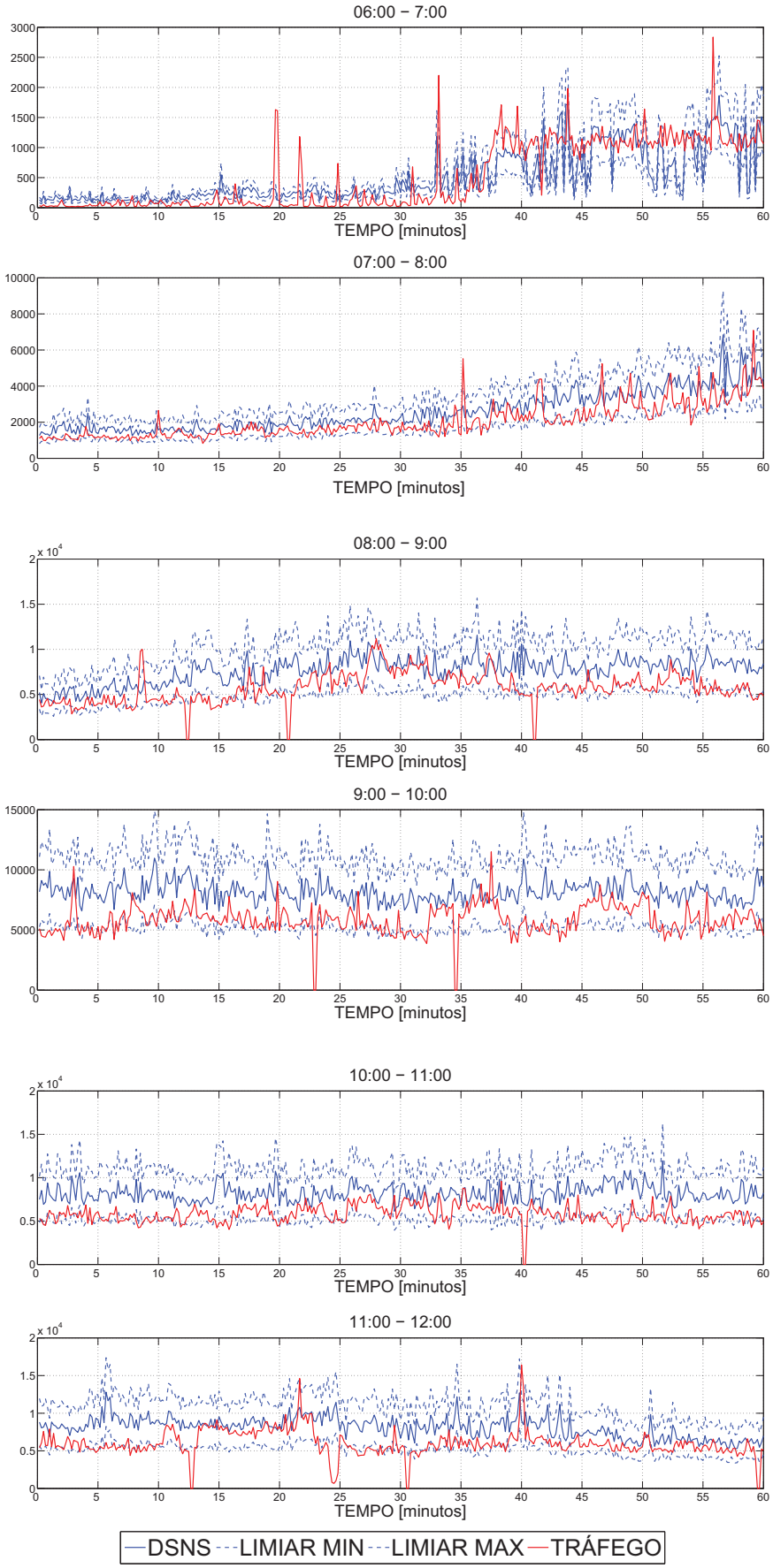

Figura 4.6: Detalhes do intervalo 03/02/2010 do objeto *ipInReceives entre 06:00 – 12:00.*

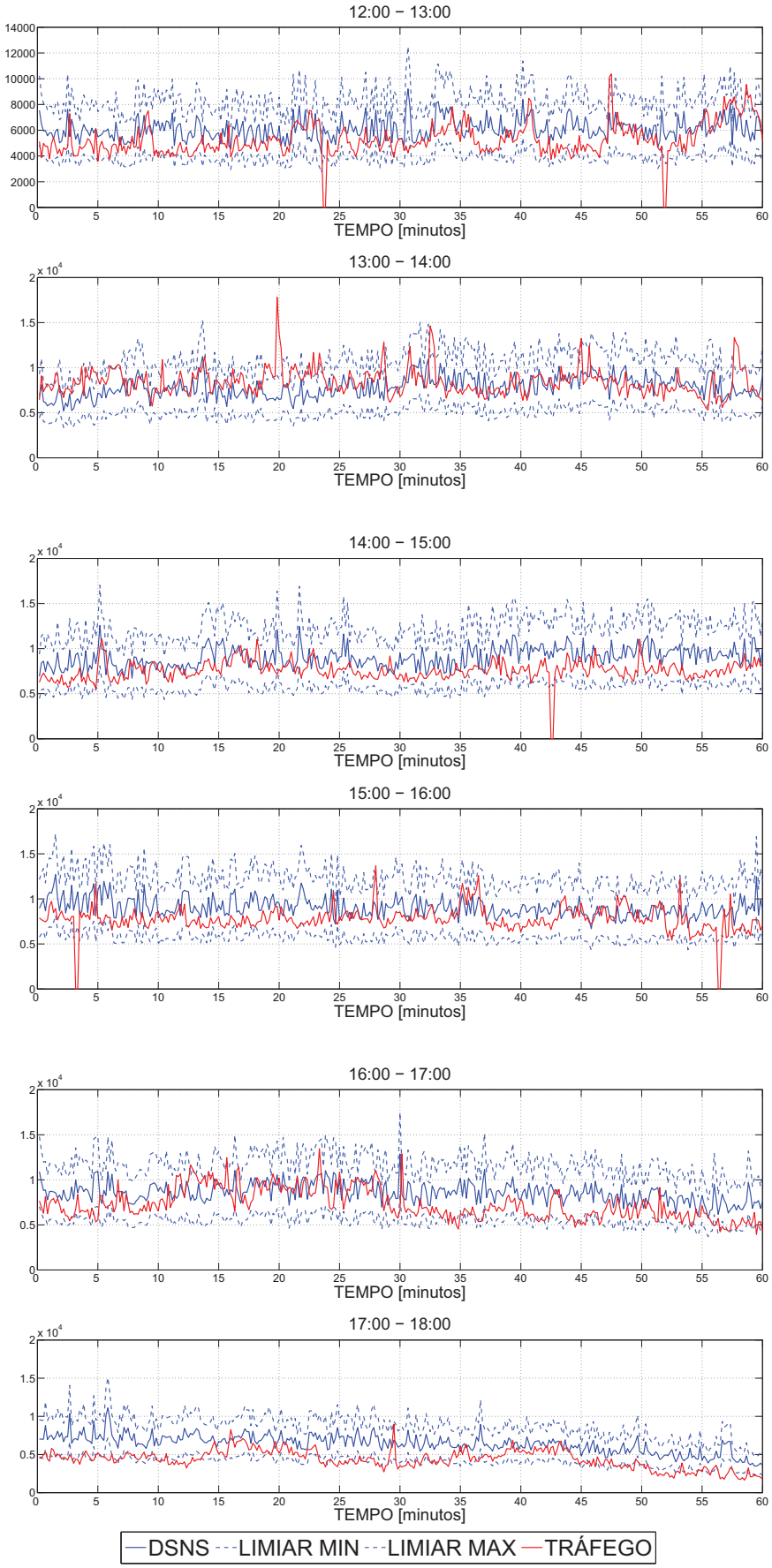

Figura 4.7: Detalhes do intervalo 03/02/2010 do objeto *ipInReceives entre 12:00 – 18:00.*

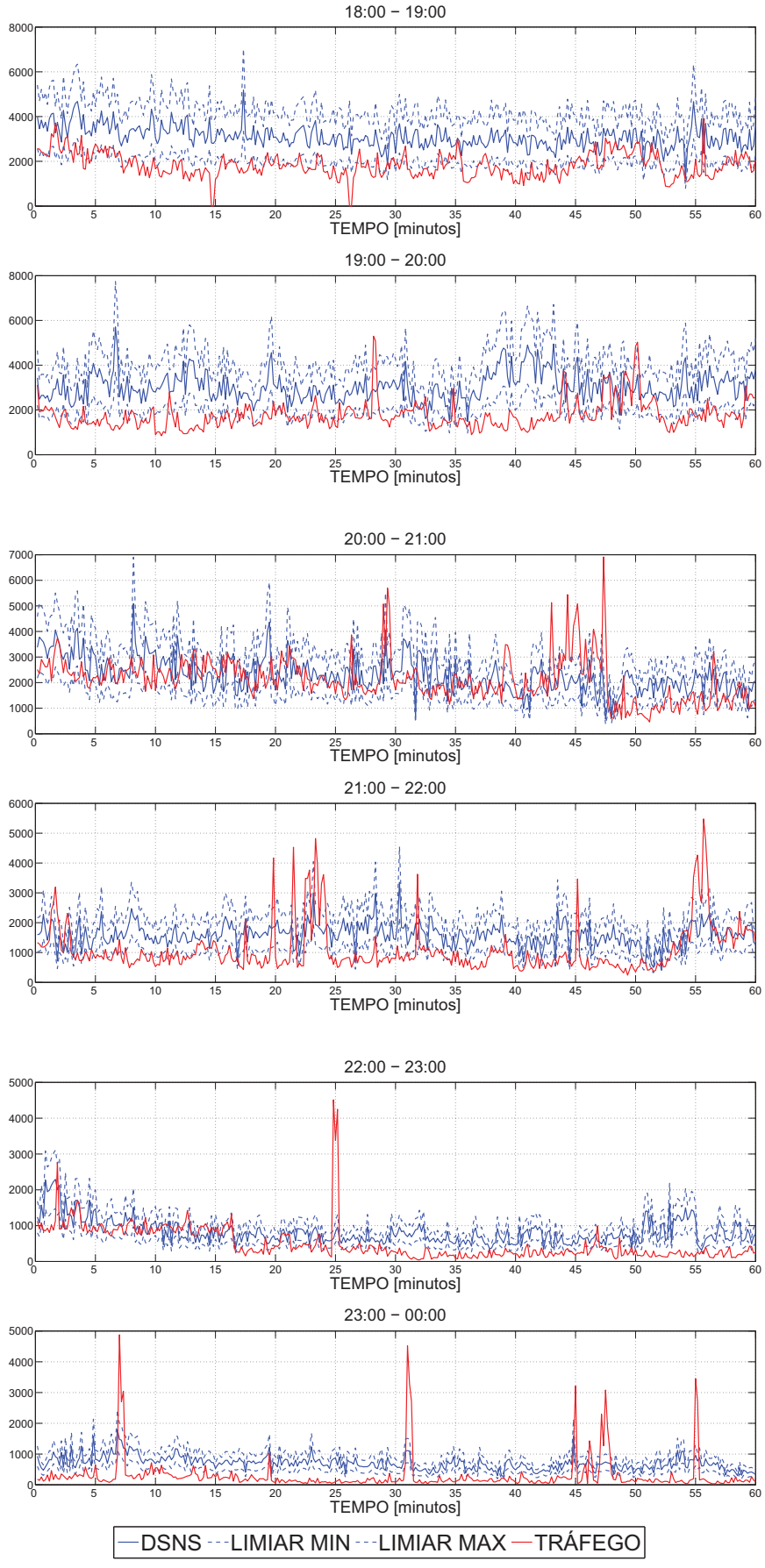

Figura 4.8: Detalhes do intervalo 03/02/2010 do objeto *ipInReceives entre 18:00 – 00:00.*

5 SISTEMA DE DETECÇÃO DE ANOMALIAS

Este capítulo traz detalhes acerca da solução do Sistema de Detecção de Anomalias (SDA) proposto, o qual utiliza Assinatura Digital de Segmento de Rede (DSNS) e o algoritmo Firefly Harmonic Clustering Algorithm (FHCA). Em um primeiro momento, são discutidos os conceitos básicos do FHCA, que é uma proposta de um algoritmo de clusterização formado pela junção dos algorimos K-Harmonic Means (KHM) e Firefly Algorithm (FA). É apresentado o problema de inicialização presente em algoritmos de clusterização baseados em centros, e como o KHM contorna este problema, e outro inconveniente é a falta de mecanismos para escapar de ótimos locais, sendo assim aplicado o FA para contornar este problema. Num segundo momento é discutido o algoritmo FHCA aplicado ao contexto de detecção de anomalias em tráfego de redes.

5.1 ALGORTIMO DE CLUSTERIZAÇÃO *FIREFLY HARMONIC*

O Algoritmo de Clusterização *Firefly Harmonic* (FHCA) é uma proposta para clusterização de dados com o objetivo de definir centros que representem os dados em grupos com similaridades. A ideia é incorporar ao algoritmo de clusterização *K-Harmonic Means* [8] a habilidade de escapar de ótimos locais da heurística *Firefly Algorithm* [22].

5.1.1 K-Harmonic Means

Algoritmos de clusterização baseados em centros buscam agrupar n dados em torno de k centros, onde o n-ésimo dado é associado ao k-ésimo centro com que apresenta a menor distância entre ambos. Desta maneira, em torno do k-ésimo centro estão agrupados os dados similares entre si, e diferentes entre os k centros. O método proposto por [36], K-Means (KM), é um algoritmo clássico e largamente difundido por apresentar características

como simples implementação e alta velocidade para agrupar um grande conjunto de dados. O conceito é categorizar um conjunto de dados N em K clusters, onde o número de centros é definido pelo usuário. Definido os K centros de cada cluster, cada ponto do conjunto de dados é associado ao cluster mais próximo, através do cálculo da distância entre o dado n e o centro k. Então, os centros são recalculados e a associação também é recalculada, e o método é refeito até que o cálculo dos novos centros não seja alterado e nenhum dado n altere sua associação. A questão de clusterização tratado como um problema de otimização busca minimizar a função descrita pela equação (5.1):

$$\mathrm{KM}(\mathbf{x}, \mathbf{c}) = \sum_{i=1}^{n} \sum_{j=1}^{k} \|x_i - c_j\|^2 \tag{5.1}$$

onde **x** é o vetor de dados e **c** é o vetor de centros. O parâmetro n é o número de elementos em **x** e k é o número de centros em **c**.

Apesar de simples e eficiente, o KM apresenta o problema de inicialização, pois se os pontos forem inicializados próximos um do outro, o algoritmo tende a estacionar, visto que não possui mecanismos para escapar de ótimos locais que a maior parte dos algoritmos de clusterização baseados em centros possuem [28, 8]. Para ilustrar tal problema de inicialização, as figuras 5.1, 5.2 e 5.3 mostram um mesmo conjunto de dados com três posições diferentes de inicialização dos centros. Podemos observar que o conjunto de dados apresenta três grupos para serem classificados e, como tal, queremos que o algoritmo responda com três centros localizados nos pontos centrais de cada grupo.

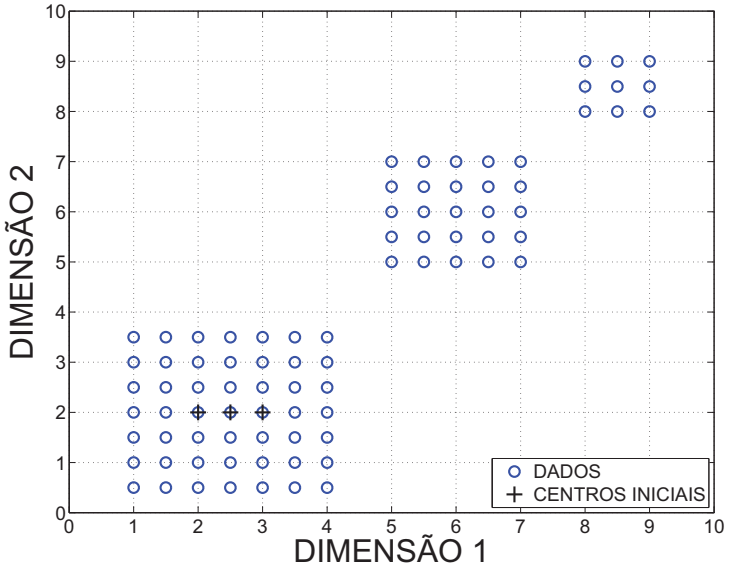

Figura 5.1: Experimento 1 - inicialização dos centros no grupo mais populoso.

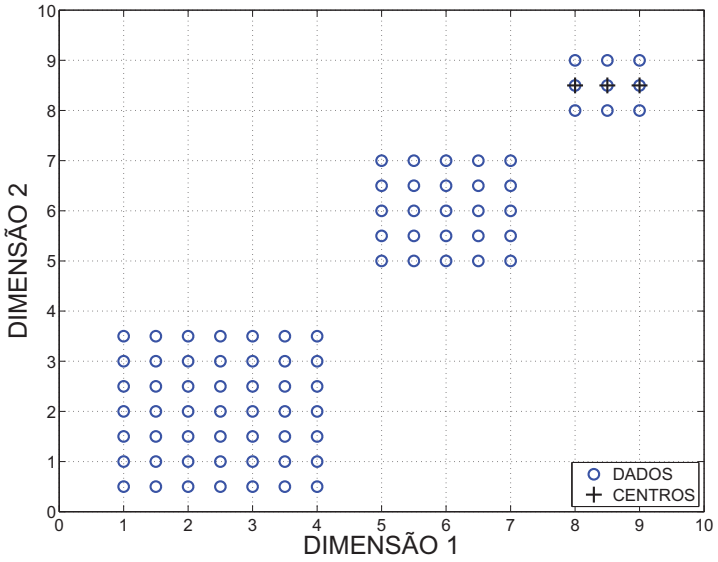

Figura 5.2: Experimento 2 - inicialização dos centros no grupo menos populoso.

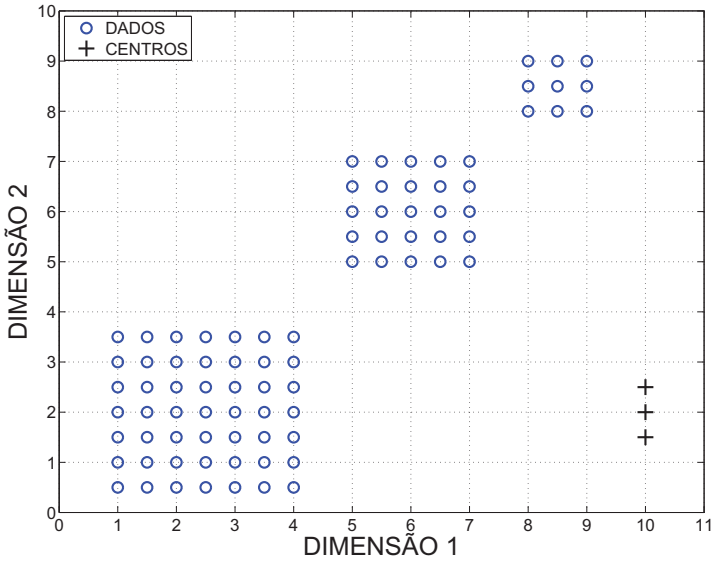

Figura 5.3: Experimento 3 - inicialização dos centros afastado dos grupos.

A figura 5.1 ilustra o experimento 1, onde os centros são inicializados no grupo mais populoso. A figura 5.4 é a resposta do algoritmo KM. Como podemos observar, o algoritmo não conseguiu escapar de um grupo densamente aglomerado, resultando em 2 centros localizados no grupo mais populoso, e o terceiro centro abrangendo os outros dois grupos.

Na figura 5.2, o experimento 2 ilustra a localização inicial dos centros no grupo menos populoso, e, como resultado da clusterização utilizando o KM, obtemos a figura

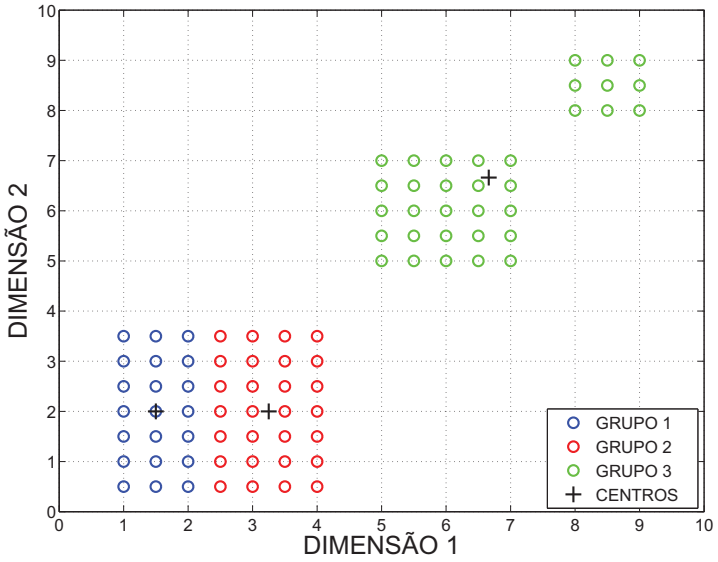

Figura 5.4: Resultado do experimento 1 utilizando o KM.

5.5 como reposta. Desta vez, o algoritmo responde de uma maneira ótima, onde cada centro se encontra no meio de cada grupo.

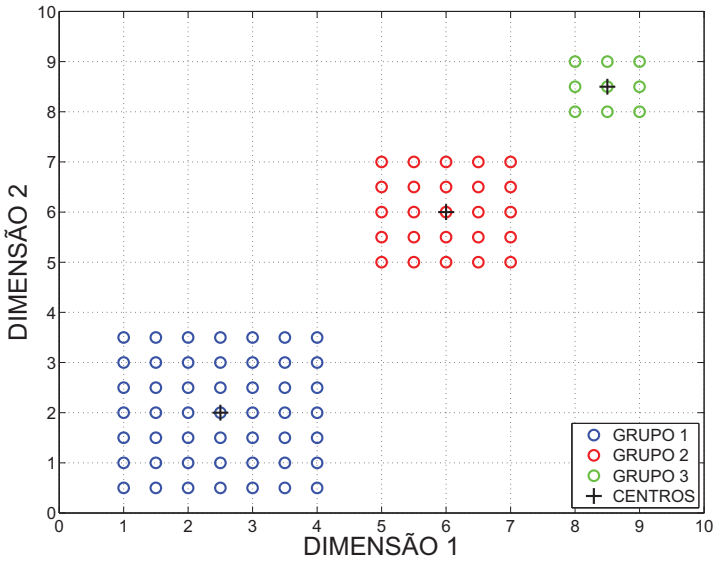

Figura 5.5: Resultado do experimento 2 utilizando o KM.

Por fim, o experimento 3 é ilustrado na figura 5.3, onde a localização inicial dos centros é afastada dos grupos. Uma vez mais, o algoritmo tende a concentrar os centros em um grupo onde a densidade é maior, como é observado na figura 5.6.

Claramente, os centros iniciais foram dispostos de forma intencional em uma posição afastada de todos os grupos para mostrar o efeito comportamental do KM em relação

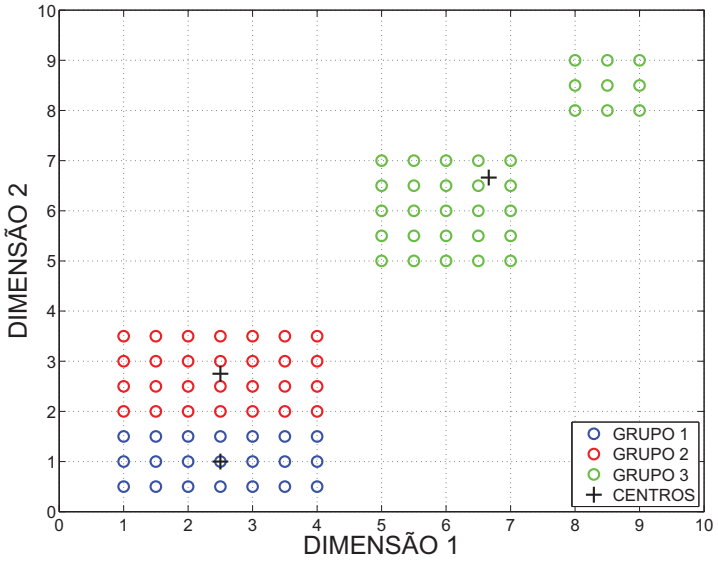

Figura 5.6: Resultado do experimento 3 utilizando o KM.

aos centros iniciais; e é possível observar o efeito que as posições de inicialização do algoritmo KM reflete diretamente na resposta final do algoritmo.

Como uma proposta para o problema de inicialização, Zhang et. al [8] prôpos o algoritmo K-Harmonic Mean (KHM), insensível à inicialização dos centros através da proposta do cálculo da função de associação ou *membership*, e uma função peso, os quais auxiliam no cálculo dos novos centros. Além destas funções, Zhang et. al prôpos a alteração da função objetivo alterando para a média Harmônica, resultando na equação (5.2):

$$\text{KHM}(\mathbf{x}, \mathbf{c}) = \sum_{i=1}^{n} \frac{k}{\sum_{j=1}^{k} \frac{1}{\|x_i - c_j\|^p}} \tag{5.2}$$

onde p é um parâmetro de KHM e assume $p \geq 2$, $\mathbf{x}$ é o vetor de dados e $\mathbf{c}$ é o vetor de centros. O parâmetro n é o número de elementos em $\mathbf{x}$ e k é o número de centros em $\mathbf{c}$. A cada iteração do KM, a função objetivo atribui pesos iguais para todos os pontos. Entretanto, o KHM associa um peso dinâmico para cada ponto baseado na média harmônica. A média harmônica atribuirá um peso maior para os pontos que não estão próximos de qualquer centro e pesos menores para os pontos que estiverem perto de um ou mais centros. Isto é importante para evitar a criação de áreas densas com múltiplos centros. Aumentando o peso dos pontos que estão distantes de qualquer centro, o algoritmo pode atrair centros fora das áreas densas sem aumentar o peso dos pontos que estão contidos nas áreas densas, desta maneira, contornando o problema de inicialização.

Para o mesmo conjunto de experimentos realizados com o KM demonstrados pelas figuras 5.1, 5.2 e 5.3, o KHM respondeu a todos os experimentos com os centros localizados nos três grupos, como ilustrado na figura 5.7.

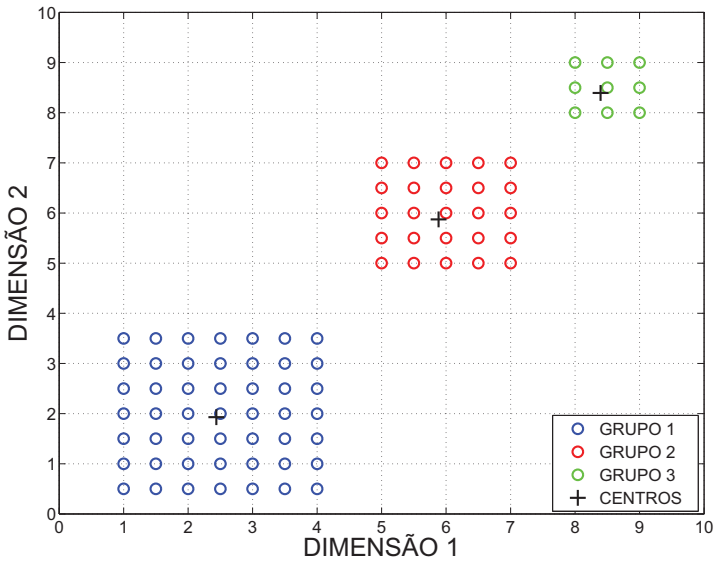

Figura 5.7: Resultado do algoritmo KHM para os experimentos 1, 2 e 3.

A função de associação é definida pela proporção do ponto x_i pertencer ao centro c_j, descrevendo a afinidade de x_i em relação a c_j, descrito pela equação 5.3:

$$m(c_j|x_i) = \frac{\|x_i - c_j\|^{-p-2}}{\sum_{j=1}^{k} \|x_i - c_j\|^{-p-2}} \tag{5.3}$$

no numerador da fração temos apenas a distância $\|x_i - c_j\|$, no denominador temos a soma das distâncias de x_i em relação aos k centros existentes. Logo, a função de associação descreve uma relação do quão perto x_i está de c_j. Na função de associação, o parâmetro p atribui um peso maior para as menores distâncias; assim, se x_i estiver próxima de c_1 e distante de c_2, então $m(c_1|x_i) > m(c_2|x_i)$.

A função de peso define o quanto o ponto x_i tem influência no recálculo dos centros na próxima iteração, descrito pela equação 5.4:

$$w(x_i) = \frac{\sum_{j=1}^{k} \|x_i - c_j\|^{-p-2}}{(\sum_{j=1}^{k} \|x_i - c_j\|^{-p})^2} \tag{5.4}$$

no numerador, encontramos a soma das distâncias de x_i e os c_j centros existentes, e no denominador a soma das distâncias ao quadrado. A função peso atribui maiores valores aos

pontos que estiverem próximos de algum centro, sendo responsável pela atração do centro próximo aos pontos onde os pesos forem maiores. Novamente, o parâmetro p na função de peso atribui maior importância as menores distâncias, resultando em um maior peso final.

A equação (5.5) descreve como recalcular as posições dos centros c_j:

$$c_j = \frac{\sum_{i=1}^{n} m(c_j|x_i)w(x_i)x_i}{\sum_{i=1}^{n} m(c_j|x_i)w(x_i)} \tag{5.5}$$

o numerador apresenta a soma dos valores de associação $m(c_j|x_i)$ e de peso $w(x_i)$ de x_i multiplicado pelo próprio x_i, e no denominador temos a normalização pela soma dos valores de associação e de peso de x_i. As funções $m(c_j|x_i)$ e $w(x_i)$ que apresentarem os maiores valores terão um peso maior na nova posição de c_j. Sendo o contrário verdade, onde $m(c_j|x_i)$ e $w(x_i)$ forem menores, apresentarão uma menor representavidade no valor final de c_j.

O algoritmo básico do KHM é apresentado em Algoritmo 5.1, como descrito em [28]:

Algoritmo 5.1 K-Harmonic means

1. Inicializa o algoritmo com centros iniciais aleatoriamente escolhidos;
2. Calcula o valor da função objetivo descrita pela Equação (5.2);
3. Para cada ponto x_i, calcular o valor da função de associação descrita pela Equação (5.3);
4. Para cada ponto x_i, calcular a função peso descrita pela Equação (5.4);
5. Para cada centro c_j, recalcular as localizações do centro baseado na Equação (5.5);
6. Repetir os passos 2-5 até o valor de KHM(X,C) não alterar ou número predefinido de iterações;
7. Associar o ponto x_i ao cluster c_j com o maior valor de $m(c_j|x_i)$.

Análise de Complexidade A complexidade apresentada pelo K-Harmonic Means é da ordem inicial de $O(X)$, onde X é o conjunto de dados a ser agrupado. Para cada ponto x, calculamos a função de associação (5.3) e função de peso (5.4). Como o KHM é um algoritmo baseado em centros, a atualização do centro tem um custo determinado pelo número de centros, tornando a complexidade $O(XC)$, C é o conjunto de centros. Esta complexidade é apresentada por algoritmos que trabalham com classificação baseado em centros. Outro parâmetro em questão é a dimensão D dos dados trabalhados, determinando uma complexidade de $O(XCD)$. Para estabelecer a complexidade final, se definirmos um limite I_t de iterações para o algoritmo, uma complexidade de $O(XCDI_t)$ é estimada.

A figura 5.8 mostra uma estimativa para a complexidade em número de operações. Para a estimativa, foram utilizados os dados dos próprios experimentos; assim, para um intervalo, definimos $X = 30, I_t = 100, C = 2$ e $D \in [1; 10]$, onde D em nosso contexto é a adição de mais objetos SNMP para análise.

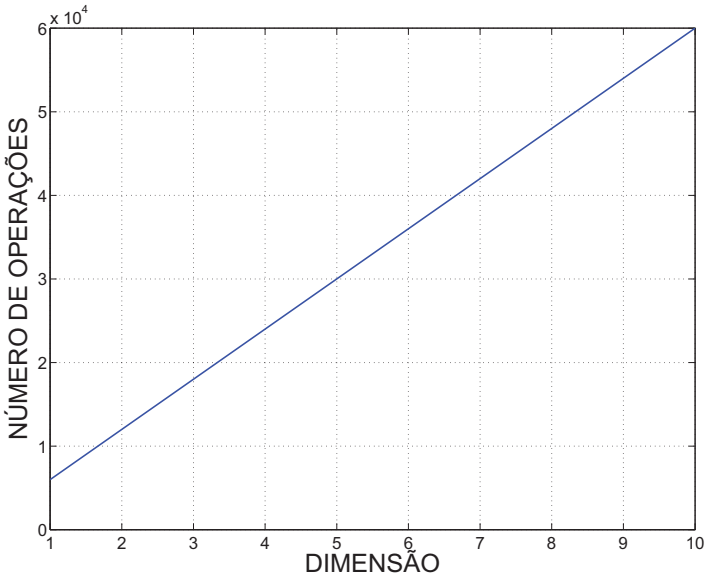

Figura 5.8: Estimativa da complexidade do KHM em número de operações efetuadas.

5.1.2 Firefly Algorithm

Infelizmente, o K-Harmonic Mean (KHM) apresenta o mesmo problema do K-Means (KM) em não possuir mecanismos para escapar de ótimos locais. Na literatura, é comum a associação de algoritmo heurístico para solucionar o problema de escapar de ótimos locais como proposto em [37]. Nosso trabalho é baseado na heurística proposta por Yang [22], o Firefly Algorithm (FA). FA é baseado no comportamento dos vagalumes e as características de emissão de luzes. A verdadeira função da emissão dos *flashes* é desconhecida e provoca um debate interessante no meio acadêmico, mas comprovadamente os *flashes* possuem a função de atrair parceiros para o acasalamento, para atrair presas e como mecanismo de segurança. O padrão é único e particular para cada espécie, mas encontramos espécies como a *photuris*, onde as fêmeas imitam o padrão dos *flashes* emitidos por outras espécies, atraindo e enganando machos de outras espécies para se alimentar [22].

Para o desenvolvimento do algoritmo inspirado no comportamento dos vagalumes, Yang idealizou três regras:

1. Todos os vagalumes são unissex, desta maneira um vagalume será atraído por outro, independente do sexo;

2. Atratividade é proporcional ao brilho, sendo assim, o vagalume menos brilhante será atraído em direção ao vagalume mais brilhante;

3. O brilho de um vagalume é determinado pela função objetivo a ser otimizada.

Baseado nestas três regras, Yang desenvolveu o pseudo código do FA, descrito no Algoritmo 5.2.

Algoritmo 5.2 Firefly Algorithm

Função objetivo $f(\mathbf{x}), \mathbf{x} = (x_m, ..., x_d)^T$
Inicializa a população de vagalumes $\mathbf{x}_m (m = 1, 2, ..., n)$
Definição do coeficiente de absorção de luz γ
ENQUANTO (t < MaxGeração)
  PARA $m = 1$ to M
    PARA $n = 1$ to M
      Intensidade de luz L_m em $\boldsymbol{x_m}$ é determinado por f($\boldsymbol{x_m}$)
      SE $(L_n > L_m)$
        Move vagalume m em direção de n em todas dimensões d
      fim SE
      Variação da atratividade com a distância
      Avaliação das novas soluções e atualização da intensidade de luz
    fim PARA n
  fim PARA m
  Ordena os vagalumes e procura pelo melhor
fim ENQUANTO
Processa e apresenta os resultados

Sabe-se que a intensidade de luz L observada de uma distância r obedece $L \propto 1/r^2$, se a distância do observador aumentar, a intensidade de luz é menor, se a distância diminuir, a intensidade de luz sentida é maior. O ar também auxilia no processo, absorvendo parte da luz, diminuindo a intensidade de acordo com a distância, tornando-a mais fraca. Estes dois fatores combinados fazem que os vagalumes sejam visíveis a distâncias limitadas. Yang [22] baseia-se na distância Euclideana entre dois vagalumes m e n escritos pela equação 5.6:

$$r_{mn} = \|\mathbf{x}_m - \mathbf{x}_n\| = \sqrt{\sum_{k=1}^{d} (x_{m,k} - x_{n,k})^2}, \tag{5.6}$$

O movimento de atração executado pelo vagalume m em relação ao vagalume mais brilhante n é descrito pela equação 5.7 [22]:

$$\mathbf{x}_m = \mathbf{x}_m + \beta_0 e^{-\gamma r_{mn}^2} (\mathbf{x}_n - \mathbf{x}_m) + \alpha(\mathbf{rand} - \frac{1}{2}), \tag{5.7}$$

onde o primeiro termo é o próprio $\mathbf{x}_m$, o segundo termo é responsável pela atração e convergência do algoritmo, e o terceiro termo representa a aleatoridade, $\mathbf{rand}$, adicionada ao processo proporcional à α. **rand** é um número aleatório gerado por uma distribuição uniforme, $[0, 1]$, e visa auxiliar no processo de escape de ótimos locais. γ é o coeficiente

de absorção de luz. Yang adota os valores de $\beta_0 = 1$ e $\alpha \in [0, 1]$. As partículas no FA possuem um ajuste de visiblidade e uma variação de atratividade versátil, resultando em uma alta mobilidade pelo espaço de busca, explorando melhor e com uma capacidade para escapar de ótimos locais [22].

Sabemos que a intensidade L de um vagalume em um determinado ponto é dado por $L(x) \propto f(x)$, sendo $f(x)$ a função objetivo definido. Entretanto, a atratividade β é relativa ao vagalume que observa outro vagalume e não de si próprio. Com isto, ela continua variando em relação a distância r_{mn} entre o vagalume m e o vagalume n. Adicionamos também o coeficiente de absorção da luz, que permite que a intensidade do brilho seja percebida em diferentes graus em diferentes distâncias. Deste modo, a intensidade de luz vista por um vagalume é dada pela equação 5.8.

$$\beta(r) = \beta_0 \exp^{-\gamma r^2}, \tag{5.8}$$

onde β_0 é a atratividade quando $r = 0$, γ é o coeficiente de absorção de luz e r é a distância, descrita pela equação 5.6. Quando a distância toma valores altos, menor é a contribuição para o resultado final, o contrário sendo verdade, quanto menor a distância, mais forte será a atração. O parâmetro γ auxilia no controle da convergência do algoritmo, se o valor de γ é baixo o algoritmo tende a convergir mais rápido, se γ assume valores altos, a atração tende a diminuir. Seguindo a analogia, se γ assumir um valor baixo, é como se o campo onde os vagalumes estão dispostos encontra-se em condições climáticas limpas, com um campo de visão aberto e claro. Na medida que γ aumenta, é como se o campo fosse coberto por uma neblina, que dificulta a visibilidade dos vagamules, fazendo com que o campo de visão diminua.

Análise de Complexidade A complexidade apresentada pelo algoritmo inicia-se com o número de iterações I_t, determinando $O(I_t)$. A cada iteração é efetuado uma comparação para todos os vagalumes M entre o próprio conjunto de vagalumes M, estabelecendo uma complexidade final de $O(M^2 I_t)$. Neste algoritmo temos uma complexidade polinomial, de ordem quadrática. Esta abordagem possui a vantagem de convergir para a solução mais rápida utilizando-se dos M vagalumes distribuídos pelo espaço de busca. Em contrapartida, dependendo da implementação e do cenário aplicado, pode-se utilizar as I_t iterações completas e o algoritmo convergir em poucas iterações, tendo um alto custo pela dependência do número de vagalumes.

5.1.3 Firefly Harmonic Clustering Algorithm

Como discutido nas seções anteriores, neste trabalho é proposto o algoritmo *Firefly Harmonic Clustering Algorithm* (FHCA) que busca unir os pontos fortes dos dois algoritmos estudados e construir um algoritmo de clusterização otimizado. A idéia é utilizar o ponto forte do *Firefly Algorithm* (FA) em escapar de ótimos locais e a busca dos melhores centros através do K-Harmonic Means (KHM) para clusterizar os dados. O pseudocódigo é apresentado em Algoritmo 5.3.

Algoritmo 5.3 Firefly Harmonic Clustering Algorithm

Entrada: amostra do tráfego real, DSNS;
Saída: tráfego e DSNS clusterizado, centros dos clusters;
1. Inicializa a população de vagalumes em posições aleatórias;
2. Define o coeficiente de absorção de luz γ;
3. **ENQUANTO** (t $<$ NumIteração)||(erro $<$ ErroAceito(KHM(x,C)))
   Calcular o valor da função objetivo descrito na Equação (5.2);
   PARA $i = 1$ to M
      PARA $j = 1$ to M
         Intensidade de luz L_i em $\boldsymbol{x_i}$ determinado por f($\boldsymbol{x_i}$)
         SE $(L_j > L_i)$
            Move vagalume i em direção de j
         fim SE
         Atratividade varia com a distância
         Avalia novas soluções e atualiza a intensidade de luz
      fim PARA j
      Calcula função de associação descrita na Equação (5.3)
      Calcula função de peso descrito na Equação (5.4)
      Para cada centro c_j, recalcular as localização baseado na Equação (5.5)
   fim PARA i
   Ordenar os vagalumes e procurar o melhor
fim ENQUANTO
4. Processar e visualizar os resultados

Como dados de entrada, temos a Assinatura Digital de Segmento de Rede (DSNS) e as amostras do tráfego de rede, o algoritmo responde com os respectivos conjuntos de centros $\mathbf{k}_j^d$ e $\mathbf{k}_j^t$, para o DSNS e tráfego de rede. Estes centro serão utilizados na classificação dos intervalos em anômalos ou não pelo algoritmo apresentado na seção 5.2.

Análise de Complexidade A complexidade apresentada pelo algoritmo é dada primeiramente pelo particionamento inicial de um conjunto de dados X por C centros em D dimensões, resultando em $O(XCD)$. Adicionamos a população de vagalumes para auxiliar na busca dos melhores centros para o agrupamento dos dados, e, como todos os vagalumes comparam-se entre si na busca da solução final, uma complexidade quadrática é adicionada,

corroborando para O($XCDM^2$). Se levarmos em consideração o número de iterações I_t como critério de parada do algoritmo, temos uma complexidade final de O($NKDM^2I_t$).

A figura 5.9 mostra uma estimativa para a complexidade em número de operações. Para a estimativa, foram utilizados os dados dos próprios experimentos; logo, para um intervalo, definimos $X = 30, I_t = 100, C = 2$ e $D \in [1, 10]$, onde D em nosso contexto é a adição de mais objetos SNMP para análise.

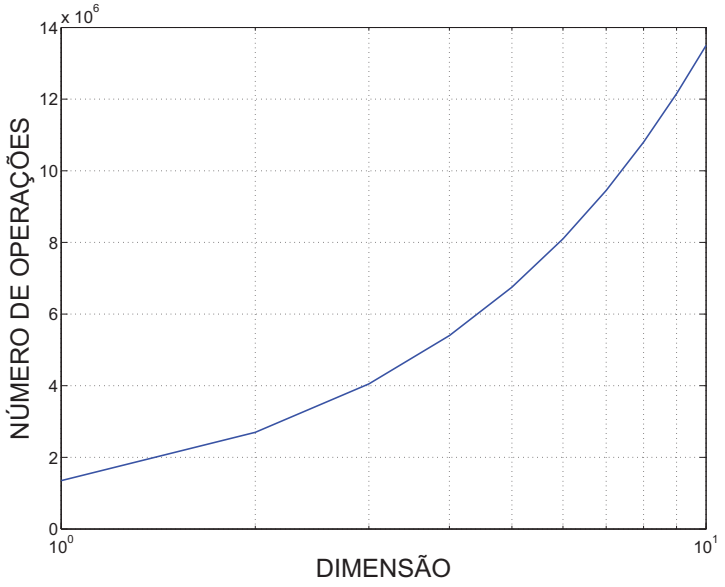

Figura 5.9: Estimativa da complexidade do FHCA em número de operações efetuadas.

5.2 SISTEMA DE ALARMES

Nesta seção é descrito o algoritmo utilizado para a geração de alarmes. Através dos dados da Assinatura Digital de Segmento de Rede (DSNS) e das amostras de tráfego, o algoritmo *Firefly Harmonic Clustering Algorithm* (FHCA) gera um conjunto de centros, sendo $\mathbf{k}_j^d$ conjunto de centros do intervalo j do DSNS e $\mathbf{k}_j^t$ é o conjunto de centros do intervalo j das amostras de tráfego de rede, cujos elementos são calculados a cada intervalo de tempo j de um total de J intervalos.

Desta maneira, para a geração da classificação dos j intervalos do dia analisado, é aplicado o algoritmo 5.4.

O algoritmo de classificação é baseado na equação (5.9).

Algoritmo 5.4 Alarme FHCA

Input: Conjunto de centros das amostras do tráfego e do DSNS;
Output: Intervalos classificados;
$\mathbf{k}_j^t$ = centros das amostras de tráfego; $\mathbf{k}_j^d$ = centros do DSNS;
PARA $j = 1$ to J
  $VO(j) = \mu(\mathbf{k}_j^t)/\mu(\mathbf{k}_j^d)$;
  SE $VO(j) < \Lambda$
    Intervalo j é NORMAL;
  SENÃO
    Intervalo j é uma ANOMALIA, gera um alarme;
  fim SE
fim PARA
- -
Λ = variação no volume

$$VO(j) = \frac{\mu(\mathbf{k}_j^t)}{\mu(\mathbf{k}_j^d)} \tag{5.9}$$

onde, $\mathbf{k}_j^t$ é o conjunto dos centros das amostras do tráfego e $\mathbf{k}_j^d$ o conjunto de centros do DSNS, μ é a média. A equação 5.9 descreve uma relação entre os centros do tráfego e os centros do DSNS gerados pelo algoritmo. Portanto, para determinar se um intervalo j será considerado anomalia, adotamos um parâmetro Λ para comparação com $VO(i)$. Se o valor obtido em $VO(i)$ for menor que Λ, o intervalo é classificado como normal, caso contrário, é considerado anomalia. Como o DSNS é adotado como o comportamento padrão da rede, é desejavel que os centros resultantes das amostras do tráfego sejam próximos e o valor de $VO(i)$ seja mais próximo de 1. O Λ descreve uma variação no volume aceito, e $\Lambda \in \{0; 1\}$.

6 RESULTADOS

Neste capítulo, são apresentadas as métricas, descritas na seção 6.1, utilizadas para avaliar os resultados do algoritmo através da sua aplicação em diferentes cenários propostos. O objetivo neste capítulo é discutir o desempenho do Sistema de Detecção de Anomalias (SDA) aplicado em diferentes cenários descritos. Os cenários estão descritos na tabela 6.1.

Cenário	**Experimento**	**Objeto**	**Período**
1	Avaliação dos parâmetros do algoritmo FHCA	*ifInOctets*	$01 - 07/08/2010$
2	Avaliação do SDA desenvolvido	*ifInOctets* *tcpInSegs*	$01 - 28/08/2010$
3	Avaliação do algoritmo FHCA desenvolvido comparado com o K-Means	*udpInDatagrams*	$03 - 07/01/2009$
4	Avaliação do algoritmo FHCA desenvolvido comparado com o PSO-CIs	*ifInOctets* *ipInReceives*	$01 - 31/03/2010$

Tabela 6.1: Cenários de teste

Os dados utilizados foram coletados da rede da Universidade Estadual de Londrina (UEL) através da ferramenta Gerenciamento de Backbone Automático (GBA), provendo um ambiente controlado e conhecimento prévio dos intervalos adotados.

6.1 MÉTRICAS ADOTADAS

Antes de discutirmos os resultados, é necessário apresentar as métricas adotadas para mensurar os resultados e enriquecer a discussão sobre a abordagem proposta. As métricas são compostas das seguintes váriaveis [38]:

- **Verdadeiro Positivo (True Positive)**: Se a instância for anomalia e é classificada como anomalia;
- **Falso Negativo (False Negative)**: Se a instância for anomalia e é classificada como normal;

- **Falso Positivo (False Positive)**: Se a instância for normal e é classificada como anomalia;
- **Verdadeiro Negativo (True Negative)**: Se a instância for normal e é classificada como normal;

Através da declaração destas variáveis é possível calcular:

$$\textbf{Taxa de Falso Positivo (FPR)} = \frac{\textbf{Falso Positivo}}{\textbf{Número de Dados Normal}} \tag{6.1}$$

$$\textbf{Taxa de Verdadeiro Positivo ou Taxa de Detecção (TPR)} = \frac{\textbf{Verdadeiro Positivo}}{\textbf{Número de Dados Anômalos}} \tag{6.2}$$

$$\textbf{Acurácia (ACC)} = \frac{\textbf{Verdadeiro Positivo} + \textbf{Verdadeiro Negativo}}{\textbf{Número de Dados Normal} + \textbf{Número de Dados Anômalos}} \tag{6.3}$$

$$\textbf{Precisão (PRE)} = \frac{\textbf{Verdadeiro Positivo}}{\textbf{Verdadeiro Positivo} + \textbf{Falso Positivo}} \tag{6.4}$$

A Equação (6.1) descreve quanto do intervalo apontado pelo algoritmo FHCA foi erroneamente classificado e a Equação (6.2) descreve o sucesso do algoritmo FHCA em classificar os dados. Através da combinação das equações (6.1) e (6.2), é possível construir gráficos *Receiver Operating Characteristics* (ROC), uma técnica para mensurar e visualizar os resultados de maneira eficiente com base em algum parâmetro. A Equação (6.3) mede o grau de aproximação das medidas do algoritmo em relação aos valores reais. A Equação (6.4) é a porcentagem de dados corretamente classificados entre todos os dados classificados. Para plotar os gráficos, foi alterado o parâmetro Λ, discutido na seção 5.2, no intervalo $[0, 1]$.

Encontrar o número ideal de centros é uma tarefa desafiadora em clusterização, e foram propostos alguns índices para medir a qualidade das soluções: índice de *Dunn's* [1], índice de *Davies-Bouldin* [2] e *Silhouette* [3].

O índice de *Dunn's* é calculado através da razão entre a distância mínima intracluster e a distância máxima intercluster. A ideia principal é identificar o conjunto de clusters que estão compactos e melhor separados. A Equação (6.5) descreve:

$$D = \frac{d_{min}}{d_{max}} \tag{6.5}$$

onde d_{min} é a menor distância entre dois objetos de diferentes clusters, e d_{max} é a maior distância entre dois objetos do mesmo cluster. D é limitado pelo intervalo $[0, \infty]$ e valores altos são almejados.

O índice de Davies-Bouldin [2] é descrito pela equação (6.6):

$$DB = \frac{1}{n} \sum_{i=1, i \neq j}^{n} \max[\frac{\delta_i + \delta_j}{d(c_i, c_j)}] \tag{6.6}$$

onde n é número de clusters, δ_i é a distância média de todos os objetos no cluster i em relação ao centro c_i, δ_j é a distância média de todos os objetos no cluster j em relação ao centro c_j e $d(c_i, c_j)$ é a distância entre os centros c_i e c_j. Se DB apresentar valores baixos, representam em clusters compactos e os centros estão distantes entre si.

A validação através da técnica de *Silhouette* [3] calcula a função de *silhouette*, $S(i)$, descrita pela Equação (6.7).

$$S(i) = \frac{b(i) - a(i)}{\max[a(i), b(i)]} \tag{6.7}$$

onde, $a(i)$ é a dissimilaridade média de i para todos os objetos do mesmo cluster, $b(i)$ é a dissimilaridade média miníma de i em relação ao objeto i com todos os objetos dos outros clusters. É possível observar que $S(i)$ está contido no intervalo $[-1, 1]$. Desta maneira, concluímos que, se $S(i)$ apresentar valores próximo de 1, as amostras estão bem associadas aos centros. Se $S(i)$ estiver próximo de 0, as amostras podem ser associadas a qualquer outro cluster próximo. Se $S(i)$ apresentar valores negativos, a amostra não foi bem classificada e está associada ao cluster errado.

Para melhor visualização dos índices apresentados, na figura 6.1 apresentamos dois conjunto de dados. No primeiro conjunto de dados, experimento 1, foram gerados quatro grupos com uma distribuição normal com $\sigma = 0, 1$, existindo uma delimitação definida entre os dados. No segundo conjunto, experimento 2, os grupos foram gerados através de uma distribuição normal com $\sigma = 0, 5$, e estão mais dispersos no espaço. A figura 6.2 apresenta o resultado da clusterização para ambos os conjuntos.

Na figura 6.2, para o experimento 1, o algoritmo encontrou e dividiu os dados em quatro grupos, como esperado, e os valores dos índices se comportaram como esperado. Para o experimento 2, foram encontrados quatro grupos, e por causa da dispersão dos dados, os valores dos índices também se comportaram como esperado. Os valores dos índices obtidos

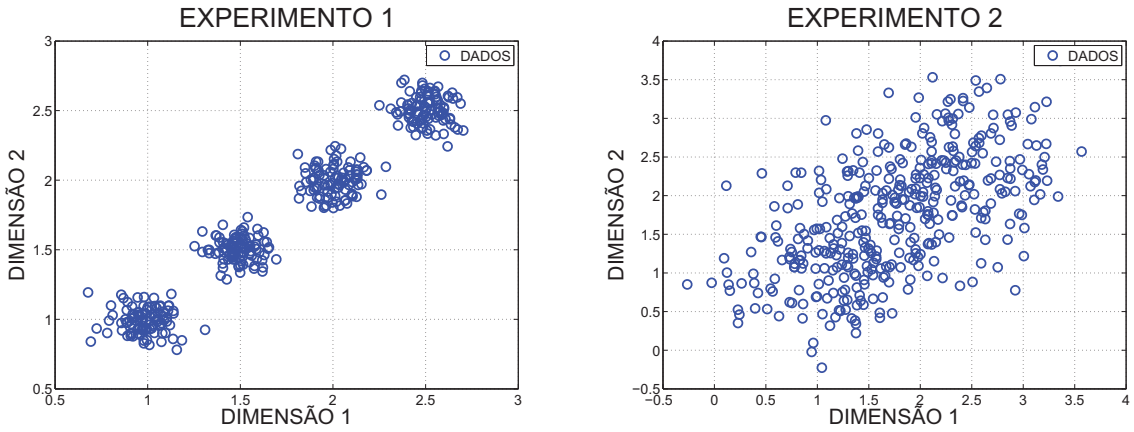

Figura 6.1: Conjunto de dados para demonstrar a validade dos índices.

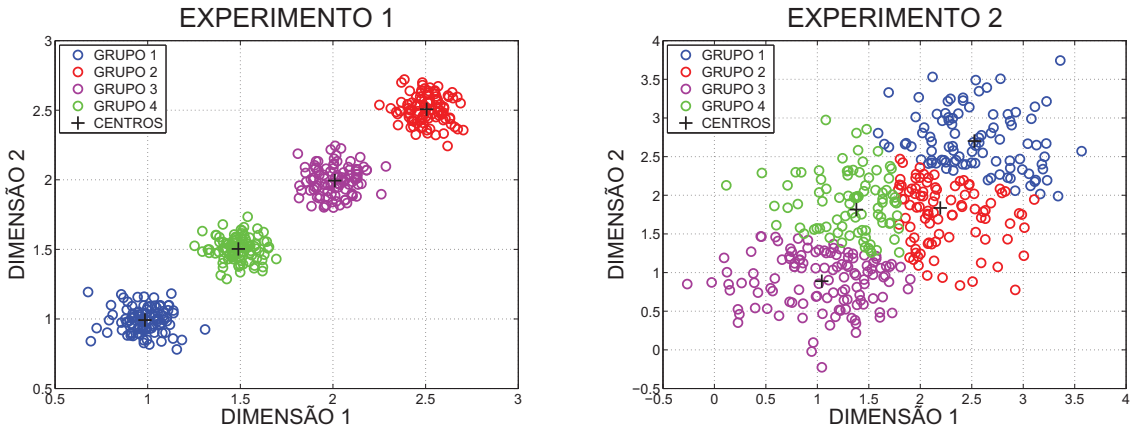

Figura 6.2: Resultado da clusterização do conjunto de dados da figura 6.1.

estão na tabela 6.2. Na figura 6.3, é apresentado o gráfico de silhueta.

Índice	**Experimento 1**	**Experimento 2**
Dunn's	5.28	1.74
Davies-Bouldin	0.57	0.99
Silhouette	0.90	0.48

Tabela 6.2: Resultado dos índices para os experimentos 1 e 2.

Na figura 6.3, podemos observar que, para o experimento 1, os quatro centros apresentam valores próximos de 1, resultando em centros bem localizados com o maior número de dados corretamente classificados. No experimento 2, temos valores variando entre 0,8 e -0,1. Por apresentarem valores negativos, alguns dados foram classificados de maneira inadequada. O objetivo deste experimento é demonstrar a validade dos índices de clusterização, pois os mesmos são utilizados para determinar o número de centros para os cenários estudados.

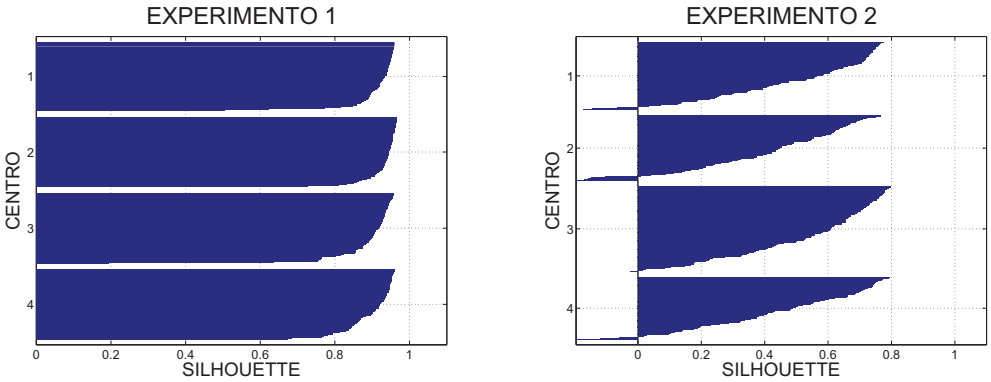

Figura 6.3: Gráfico silhueta da clusterização do conjunto de dados da figura 6.1.

6.2 CENÁRIO 1: Parâmetros do algoritmo FHCA

Este cenário utiliza tráfego real da rede da Universidade Estadual de Londrina, coletado durante uma semana no período de $01 - 7/08/2010$, do objeto *ifInOctets* da MIB, no servidor *web*. O objeto *ifInOctets* determina o número de octetos recebidos pela interface de rede monitorado.

Foi realizado uma série de experimentos para encontrar os parâmetros que melhor obtivessem uma taxa de detecção alta. O primeiro parâmetro testado foi o número de centros K a ser trabalho pelo algoritmo de clusterização, utilizando-se dos índices apresentados na seção 6.1. O gráfico disposto em 6.4, apresenta o resultado para os índices discutidos.

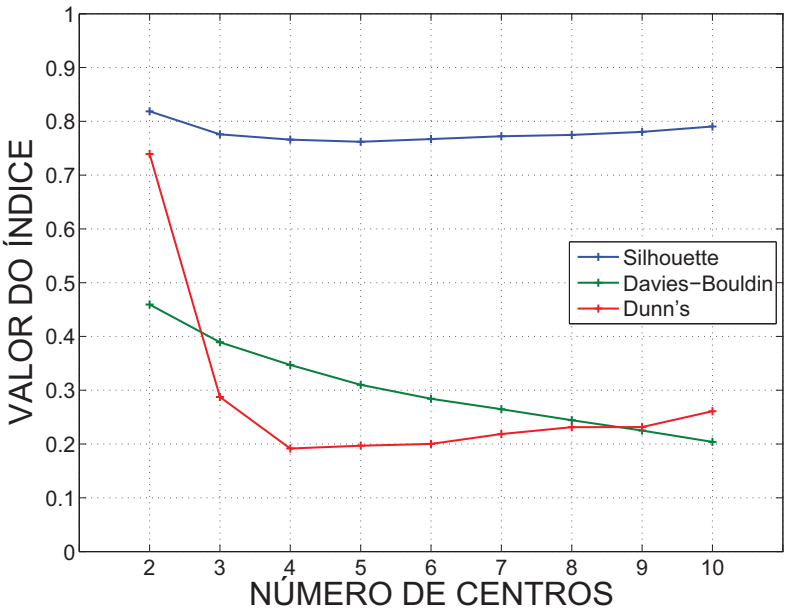

Figura 6.4: Validação do número de K para índice de *Dunn's* [1], índice de *Davies-Bouldin* [2] e *Silhouette* [3].

Adotamos $K = 2$ para o FHCA, porque descreve os melhores resultados. O índice de D apresenta o maior valor quando $K = 2$, entretanto o melhor índice DB é quando $K = 10$. Mas isto pode ser explicado porque a média dos cluster deve ser baixo, apresentado uma número de objetos associados menor e os centros são localizados distantes entre si. Os valores obtidos pelo teste de *silhouette* não apresentam muita variação entre os diferentes K testados porque neste gráfico é apresentado como a média de todos os intervalos.

Para validação usando *silhouette*, no gráfico 6.5 é apresentado o teste com diferentes valores de K, para um único intervalo. Para poder interpretar o gráfico, no eixo y é encontrado o número de centros utilizados para a clusterização, e no eixo x, o valor da função $S(i)$. Através disto, é desenhado uma área em torno dos centros, e o resultado esperado é de as áreas volumosas em torno do centro k, e com valores de $S(i)$ próximos de 1.

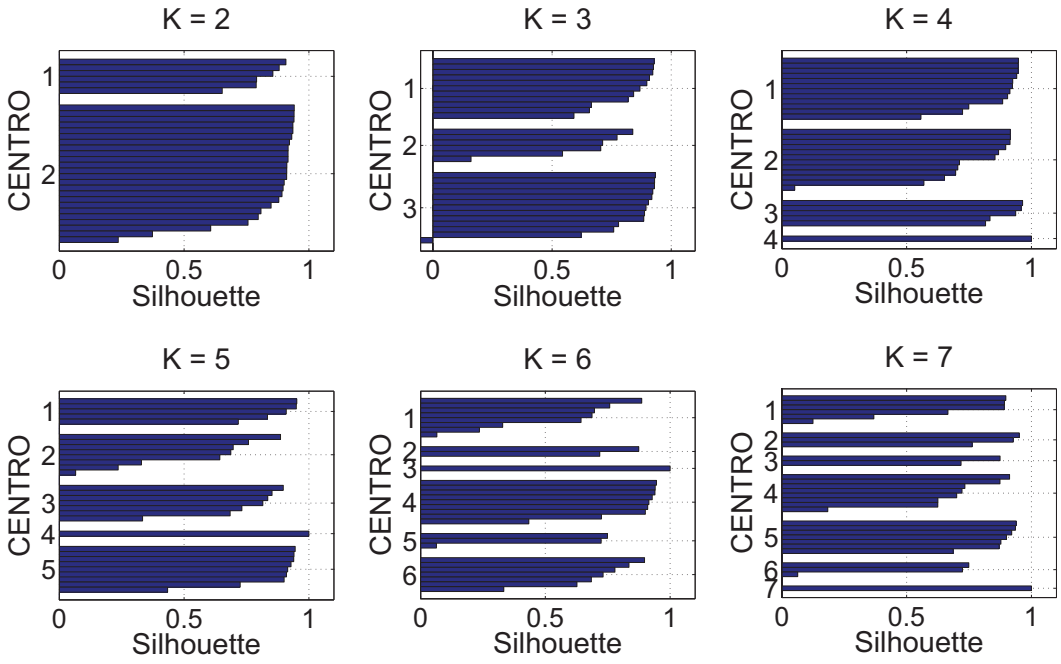

Figura 6.5: Silhouette para diferentes valores de K.

É possível apontar que K $= 2$ apresenta o melhor resultado novamente, porque apresenta um número maior de objetos associados entre os centros com índices de $S(i)$ maiores. Isto é representado pelas 2 áreas voluptosas formadas. Quando K $= 3$, o terceiro cluster apresenta dados classificados de maneira errada, gerando valores negativos de $S(i)$. Quando K $= 4$, no quarto cluster existe uma área com um alto valor de $S(i)$, porém a área descrita é muito pequena, associando poucos dados à este centro.

Para os outros valores de K, temos uma associação com $S(i)$ perto de 1, mas o volume de dados associado a cada cluster também diminui, de maneira que não torna

interessante a classificação, criando centros com poucos dados agrupados no seu entorno. Com isto, para o conjunto de dados trabalhado, K $= 2$ é o valor que retorna a melhor classificação e é adotado para os resultados posteriores.

Prosseguindo com o estudo dos parâmetros do algoritmo, o valor do parâmetro p, provindo do algoritmo de clusterização foi posto à prova. O gráfico 6.6 mostra o resultado em forma da curva ROC, medindo a taxa de falso positivo e taxa de verdadeiro positivo, variando o valor de Λ.

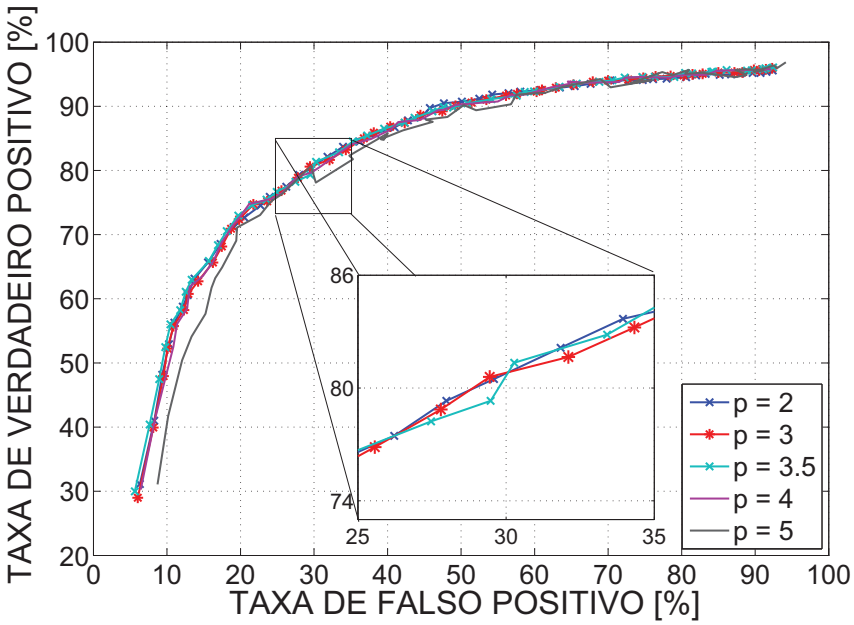

Figura 6.6: Gráfico ROC para diferentes valores de p.

As curvas descritas na figura 6.6 são muito próximas umas das outras. Como critério de escolha de p, foi adotado o valor que apresentasse um melhor trade-off na questão de detecção, ou seja, a curva que obtivesse uma TPR alta. Para os dados trabalhados, quando $p = 2,3$ e $3,5$, os valores mostraram diferenças nas casas decimais, exceto quando $p > 4$, onde as taxas de TPR demonstraram uma redução. Portanto, para os posteriores experimentos, $p = 2$.

O próximo passo foi testar os diferentes valores que α e γ poderiam assumir no FHCA onde o *trade-off* entre o TPR e FPR fosse maior que 80% e menor que 30% respectivamente, apresentando uma taxa de ACC elevada, apresentada na tabela 6.3. O número de vagalumes adotado foi $M = \Delta * 0,25$.

Através da tabela 6.3, é possível notar que os valores de α e γ não influenciaram diretamente nas taxas TPR e FPR no cenário estudado. A taxa de acurácia

α	0,1									
γ	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
TPR [%]	80,790	80,514	80,766	80,623	80,514	80,653	80,445	80,798	80,798	80,798
FPR [%]	29,602	30,023	29,704	29,802	29,789	30,033	29,606	29,798	30,015	30,018
Acurácia [%]	75,843	75,496	75,794	75,645	75,595	75,595	75,645	75,744	75,694	75,694
α	0,2									
γ	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
TPR [%]	80,790	80,514	80,766	80,628	80,514	80,653	80,445	80,523	80,798	80,798
FPR [%]	29,602	30,023	29,704	30,146	29,789	30,033	29,606	30,108	30,015	30,018
Acurácia [%]	75,843	75,496	75,794	75,496	75,595	75,595	75,645	75,446	75,694	75,694
α	0,3									
γ	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
TPR [%]	80,766	80,340	80,798	80,735	80,798	80,766	80,558	80,514	80,584	80,584
FPR [%]	30,112	30,155	29,700	29,962	29,920	29,703	29,790	29,929	29,594	29,874
Acurácia [%]	75,595	75,347	75,794	75,694	75,694	75,794	75,645	75,546	75,744	75,595
α	0,4									
γ	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
TPR [%]	80,415	80,491	80,515	80,592	80,515	80,491	80,592	80,415	80,491	80,592
FPR [%]	29,449	29,624	29,449	29,624	29,624	29,624	29,624	29,449	29,764	29,624
Acurácia [%]	75,794	75,794	75,843	75,843	75,794	75,794	75,843	75,794	75,744	75,843
α	0,5									
γ	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
TPR [%]	80,859	80,340	80,798	80,735	80,798	80,766	80,558	80,514	80,836	80,584
FPR [%]	29,616	30,155	29,700	29,962	29,920	29,703	29,790	29,890	29,616	29,640
Acurácia [%]	75,893	75,347	75,794	75,694	75,694	75,794	75,645	75,546	75,893	75,694
α	0,6									
γ	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
TPR [%]	80,415	80,491	80,515	80,592	80,515	80,491	80,592	80,415	80,491	80,592
FPR [%]	29,449	29,624	29,449	29,624	29,624	29,624	29,624	29,449	29,624	29,624
Acurácia [%]	75,794	75,794	75,843	75,843	75,794	75,794	75,843	75,794	75,794	75,843
α	0,7									
γ	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
TPR [%]	80,491	80,592	80,523	80,592	80,592	80,491	80,592	80,592	80,592	80,491
FPR [%]	29,624	29,449	29,449	29,624	29,624	29,624	29,624	29,764	29,449	29,624
Acurácia [%]	75,794	75,893	75,843	75,843	75,843	75,794	75,843	75,794	75,893	75,794
α	0,8									
γ	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
TPR [%]	80,415	80,491	80,515	80,592	80,515	80,491	80,592	80,415	80,491	80,592
FPR [%]	29,449	29,624	29,449	29,624	29,624	29,624	29,624	29,449	29,764	29,624
Acurácia [%]	75,794	75,794	75,843	75,843	75,794	75,794	75,843	75,794	75,744	75,843
α	0,9									
γ	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
TPR [%]	80,415	80,491	80,515	80,592	80,515	80,491	80,592	80,415	80,491	80,592
FPR [%]	29,449	29,624	29,449	29,624	29,624	29,624	29,624	29,449	29,764	29,624
Acurácia [%]	75,794	75,794	75,843	75,843	75,794	75,794	75,843	75,794	75,744	75,843
α	1									
γ	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
TPR [%]	80,491	80,592	80,523	80,592	80,592	80,491	80,592	80,592	80,592	80,491
FPR [%]	29,764	29,449	29,449	29,624	29,624	29,624	29,624	29,624	29,449	29,624
Acurácia [%]	75,744	75,893	75,843	75,843	75,843	75,794	75,843	75,843	75,893	75,794

Tabela 6.3: Resultados dos parâmetros α e γ para o Firefly Harmonic Clustering Algorithm.

também não sofre muitas alterações, demonstrando também que os parâmetros não influencia no contexto estudado. Os valores das taxas, na casa decimal não é levado em consideração para o modelo proposto, portanto, para os posteriores experimentos, é adotado $\alpha = 0,1$ e $\gamma = 1$.

6.3 CENÁRIO 2: Aplicação do algoritmo

Com os parâmetros definidos e discutidos na seção 6.2, testamos a abordagem proposta neste cenário que utiliza tráfego real da rede da Universidade Estadual de Londrina, coletado durante o período de 01-28/08/2010 para os objetos SNMP: *ifInOctets* e *tcpInSegs*. O objeto *ifInOctets* determina o número de octetos recebidos pela interface de rede, o objeto *tcpInSegs* determina taxa de segmentos TCP recebidos.

A figura 6.7 apresenta as curvas de precisão e acurácia obtidas pelo modelo proposto.

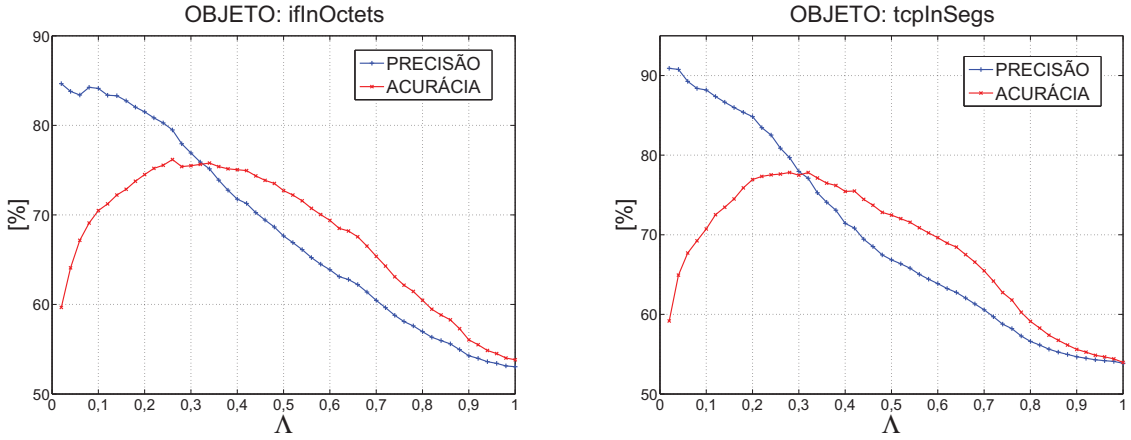

Figura 6.7: Precisão e acurácia média alcançada pelo algoritmo.

Na figura 6.7, a faixa de valores importante para ser analisado é $0,1 < \Lambda < 0,3$, como Λ representa a quantidade de variação aceita entre os dados do DSNS e do tráfego, a faixa de valores indica os valores aceitos no contexto estudado. Dentro da faixa de valores, a precisão alcançada apresenta uma curva decrescente. Para o objeto *ifInOctets*, uma taxa variando entre 85% e 75%, e para o objeto *tcpInSegs*, uma variação entre 88% e 78%. Porém, a taxa de acurácia apresenta uma curva crescente, apresentando para o *ifInOctets* uma variação de 70% e 75%, e para o *tcpInSegs*, 70% e 77%. De acordo com o Λ adotado, o algoritmo pode ser ajustado para aumentados a qualidade da detecção, em contrapartida, a quantidade de anomalias detectadas diminui.

A figura 6.8 apresenta os gráficos ROC para ambos os objetos testados, para

as quatro semanas. O objeto *ifInOctets* recebe um volume maior de dados, pois representa número de bytes recebidos e não faz distinção entre os pacotes. O objeto *tcpInSegs* representa apenas os segmentos TCP recebidos, logo, o volume reportado é menor que do objeto *ifInOctets*.

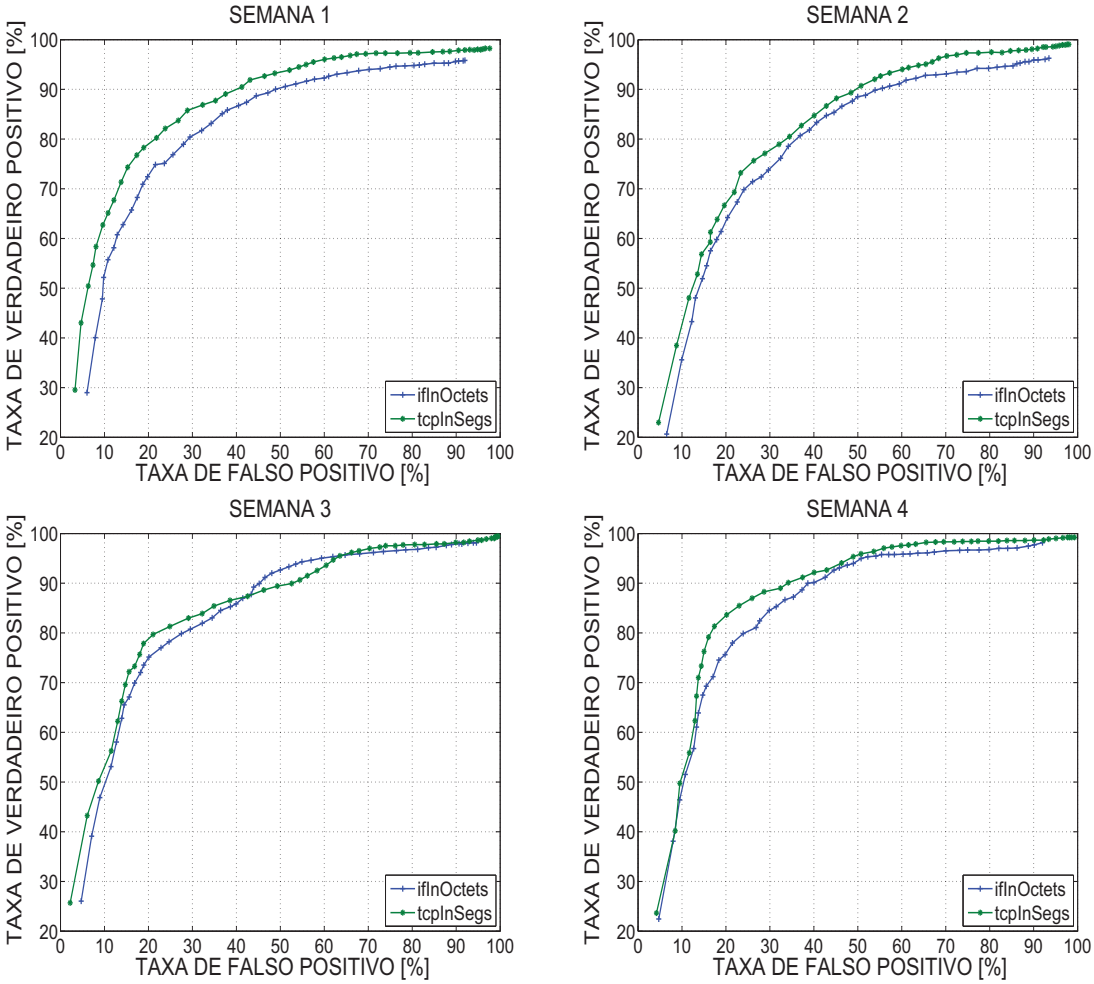

Figura 6.8: Gráfico ROC de cada semana analisada, para os objetos SNMP *ifInOctets* e *tcpInSegs*.

O comportamento do algoritmo apresentado é sólido e não apresenta muitas variações entre as 4 semanas estudadas, para os dois objetos propostos. Por causa da variação de volumes entre os objetos, as taxas apresentadas para o *ifInOctets* são menores por tratar um volume maior, por isso, existe uma variação maior nos dados trabalhados. Mas o algoritmo responde com taxas satisfatórias para ambos os objetos. Considerando a taxa de falso positivo, o algoritmo alcança taxas de verdadeiro positivo superiores a 70%.

6.4 CENÁRIO 3: Comparação com K-Means

Este cenário foi proposto para avaliar os efeitos da clusterização dos dados no modelo de detecção. Foi aplicado a clusterização dos dados pelos algoritmos, FHCA e K-Means (KM), e a validação com o algoritmo de alarmes descrito na seção 5.4. O cenário utiliza tráfego real da rede da Universidade Estadual de Londrina, coletado durante uma semana no período de 01-07/01/2009 para o objeto SNMP: *udpInDatagrams*, o objeto determina o número de datagramas recebidos pela interface de rede.

O gráfico 6.9 apresenta a curva de acurácia de ambos os algoritmos. Para poder interpretar o gráfico, é importante ressaltar que para efeito comparativo, adota-se intervalo $0,1 < \Lambda < 0,3$ para efeito de estudo, porque são os valores que melhor representam uma aceitação de variação do tráfego.

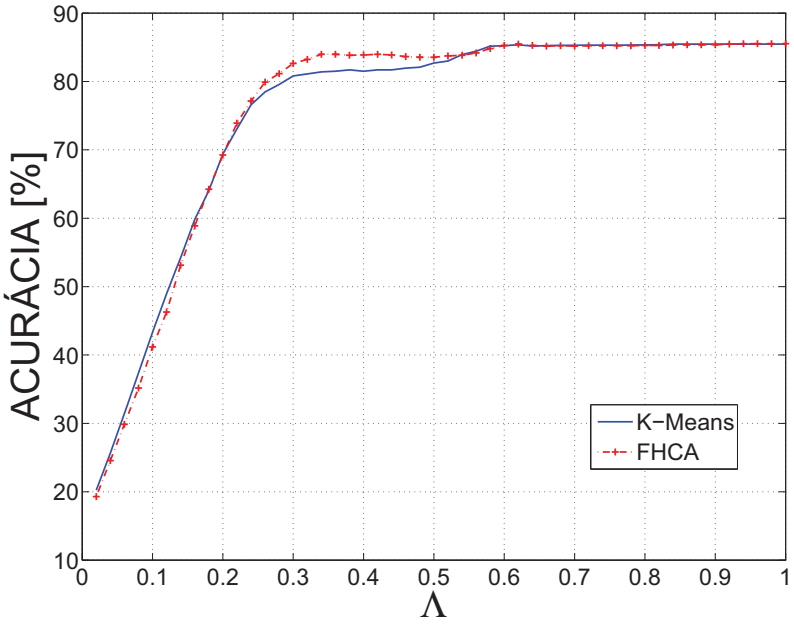

Figura 6.9: Gráfico de acurácia do K-means e FHCA.

Existe uma pequena vantagem do FHCA em relação aos resultados do KM. Este gráfico demonstra que os intervalos anômalos classificados pelo algoritmo FHCA estão, na sua maioria, melhores classificados comparado aos intervalos classificados pelo KM. Para $\Lambda = 0,2$, as taxas de acurácia para ambos atinge 70%, demonstrando que os algoritmos respondem com grupos clusterizados de modo similar.

O gráfico 6.10 apresenta a curva ROC com a média das taxas falso positivo e verdadeiro positivo das dados estudados. Podemos observar uma pequena diferença entre o

KM e o FHCA.

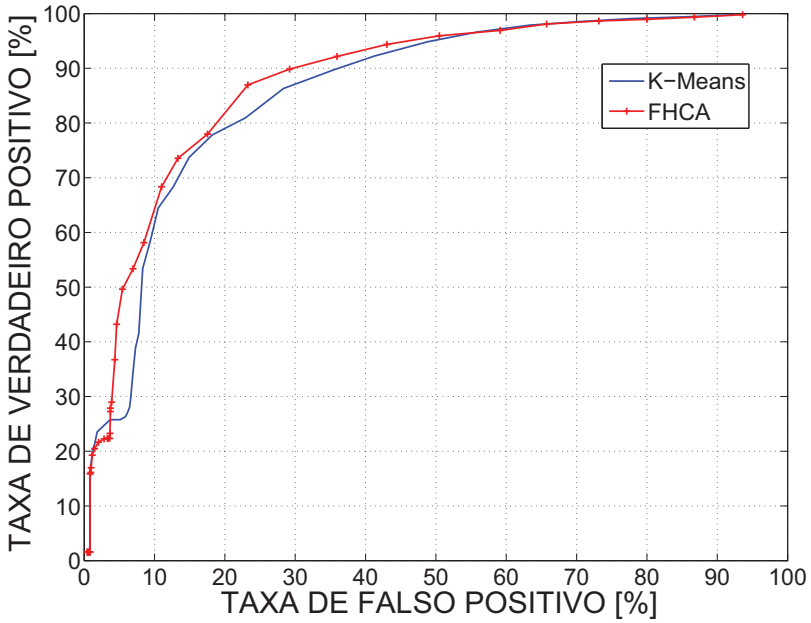

Figura 6.10: Gráfico ROC do K-means e FHCA.

Na figura 6.10, é observado uma vantagem do FHCA em relação ao KM. A taxa de verdadeiro positivo do algoritmo FHCA se mantem acima da taxa apresentada pelo KM. Tomando o intervalo onde a taxa de falso positivo encontra-se entre 10% e 30%, o FHCA mantém as taxas de verdadeiro positivo acima da curva descrita pelo KM. Podemos concluir que os grupos clusterizados pelo FHCA e utilizados para a detecção de anomalias através do modelo proposto, obtem melhores resultados que os grupos clusterizados pelo KM.

A complexidade dos algoritmos é apresentado na figura 6.11. A complexidade do KM é menor que a complexidade apresentada pelo FHCA, porém, como os experimentos apresentados nesta dissertão são unidimensional, desta maneira, a diferença de complexidade não influencia nos resultados de maneira impactante na escolha dos algoritmos de clusterização. Nenhuma vantangem ou desvantagem acerca da complexidade para o aumento do número de dimensões pode ser concluido, para isto, um estudo mais aprofundado é necessário e não foi escopo deste trabalho, abrindo uma possibilidade para trabalhos futuros.

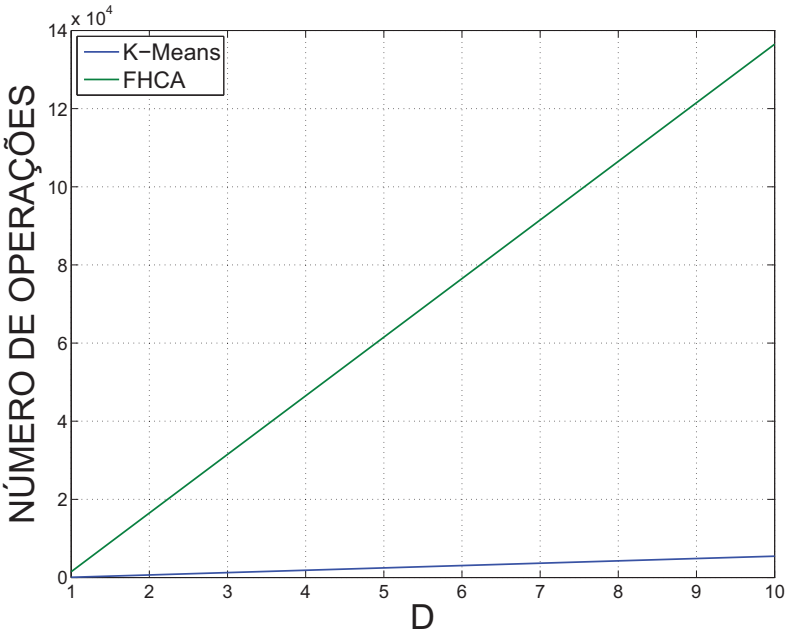

Figura 6.11: Gráfico da complexidade dos algoritmos K-means e FHCA.

6.5 CENÁRIO 4: Comparação com PSO-Cls

O segundo teste comparativo foi efetuado com a abordagem proposta por Lima et. al [39], que é uma proposta também utilizando um método de clusterização otimizado, Particle Swarm Optimization with Clustering (PSO-Cls) aplicado aos dados do (DSNS) e do tráfego. O conjunto de dados aplicado sobre eles foram os dados coletados da rede da Universidade Estadual de Londrina, coletado durante mês no período de 01-28/08/2010 para os objetos SNMP: *ifInOctets* e *ipInReceives*. O objeto *ifInOctets* determina o número de octetos recebidos pela interface de rede, o objeto *ipInReceives* determina o número de pacotes IP recebidos pelo segmento de rede analisado.

Na figura 6.12 é apresentado o gráfico ROC comparativo entre os dois algoritmos dividido em 4 semanas para o objeto *ifInOctets*.

É notável uma diferença entre as curvas descritas pelos algoritmos na figura 6.12. Na figura 6.13, é apresentado o gráfico ROC comparativo para o objeto *ipInReceives*, e o resultado é similar ao apresentado na figura 6.12.

Para os testes propostos, é notável a superioridade do FHCA em relação ao PSO-Cls. As figuras 6.12 e 6.13, representam a média das taxas de detecção e falso alarme para cada semana. Significa que foram gerados os gabaritos para cada dia da semana através da caracterização adotada, discutida na seção 4, e comparados aos intervalos classificados

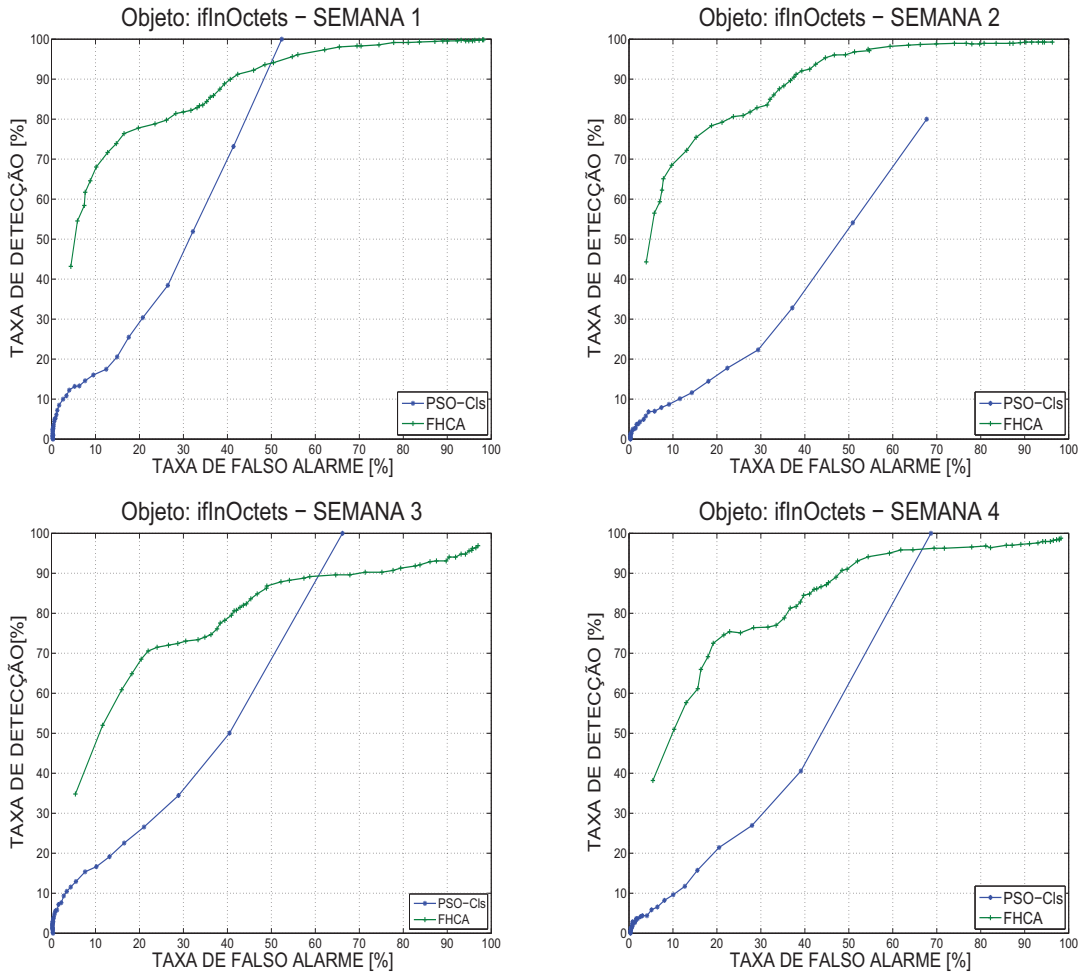

Figura 6.12: ROC graph for *ifInOctets* object.

por ambos os modelos. Portanto, mesmo dois modelos utilizando abordagens similares, os resultados são muito dispares. Importante ressaltar que na abordagem proposta nesta dissertação; introduzimos a descrição de anomalia de volume, gerando um gabarito utilizado para comparação com os intervalos apontados pelo algoritmo. No modelo PSO-CIs, não existe este gabarito, os intervalos apontados eram pontuais e de conhecimento do administrador de redes.

Foram efetuados testes em três períodos diferentes dentro destas semanas analisadas, de maneira a compreender e aferir melhor os algoritmos. O experimento 01, é formado por intervalos sem ocorrência de anomalias. A figura 6.14 ilustra o dia 02/03/2010 entre as 14h e 16h, para o objeto *ifInOctets*, e são intervalos considerados normais.

É possível observar três intervalos (14:40, 15:00 e 15:55) que contém picos, mas de acordo com a descrição, são intervalos considerados normais, nesse caso, os picos não devem ser representativos para a caracterização.

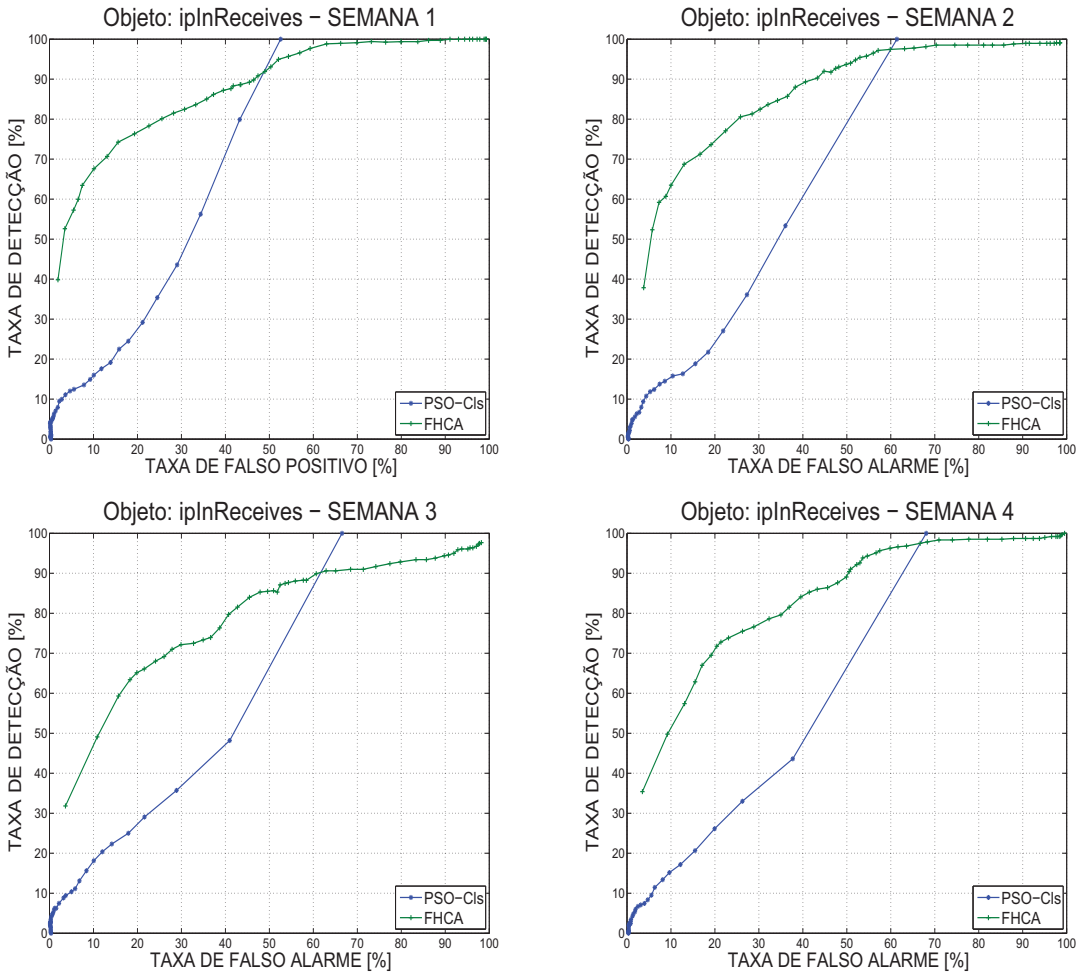

Figura 6.13: ROC graph for *ipInReceives* object.

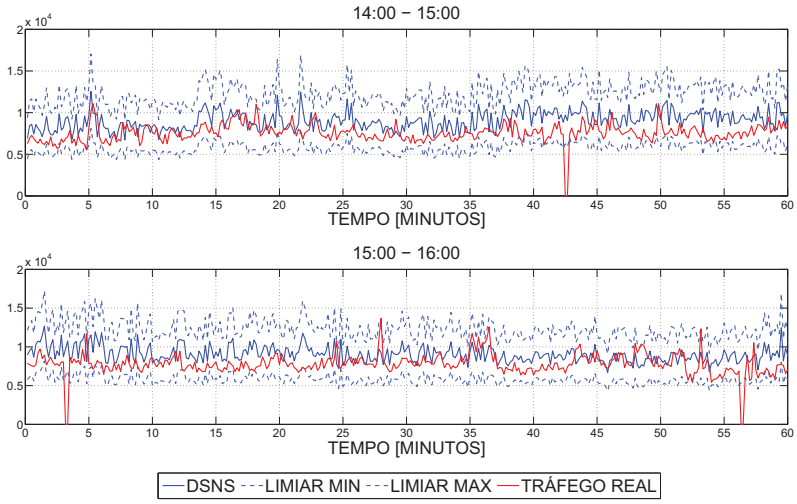

Figura 6.14: Experimento 1, intervalos considerados normais.

O experimento 02, é adotado intervalos contendo anomalias com uma pequena varição no volume e intervalos normais, visto na figura 6.15. Os intervalos estão

entre as 16h e 18h do dia 05/03/2010 para o objeto *ifInOctets*.

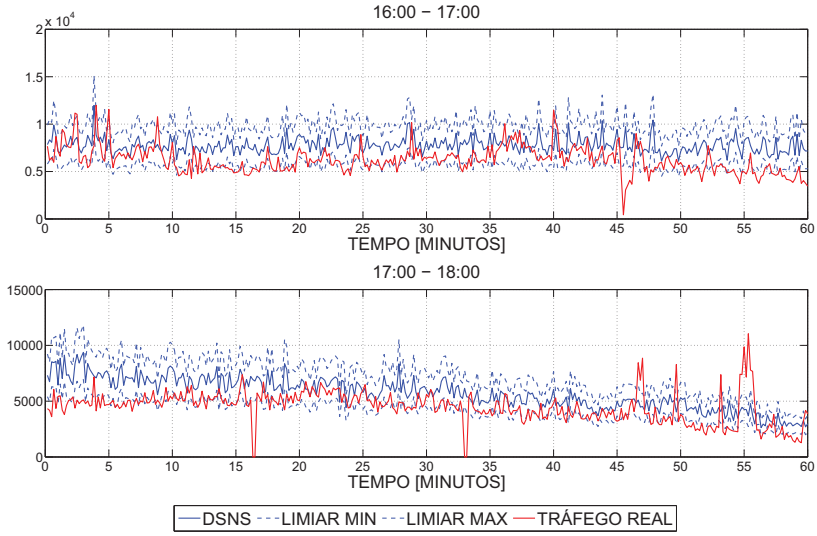

Figura 6.15: Experimento 1, intervalos considerados normais.

Os intervalos apresentados, são constituidos de intervalos anômalos com pouca variação em relação aos limiares traçados. As 16:45 até as 17:10, 17:45 até 18:00 são exemplares de intervalos anômalos.

O experimento 03, figura 6.16, é constituido por alguns intervalos normais e pouco intervalos contendo anomalias de volume explícito. Os intervalos estão entre as 14h e 16h, do dia 01/03/2010 para o objeto *ifInOctets*.

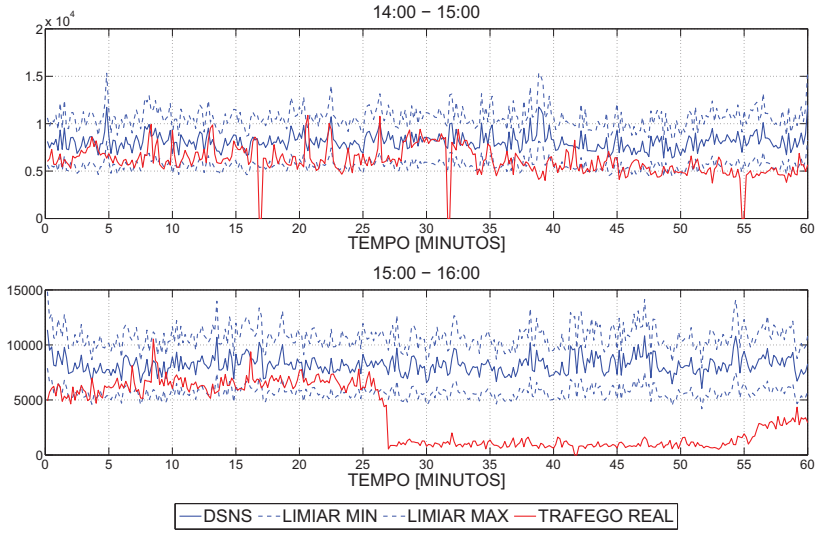

Figura 6.16: Experimento 1, intervalos considerados normais.

Apartir das 15:25 até as 16:00 é notado uma grande queda brusca do tráfego, caracterizando uma anomalia por falha do dispositivo da rede.

	FHCA		PSO-CIs	
	TPR [%]	FPR [%]	TPR [%]	FPR [%]
Experimento 01	0	4	0	12
Experimento 02	80	16	50	25
Experimento 03	88	20	77	33

Tabela 6.4: Resultado para os experimentos 1, 2 e 3.

A tabela 6.4 foi construída utilizando-se as taxas de detecção (TPR) e falso alarme (FPR) do FHCA e PSO-CIs nos experimentos 1, 2 e 3.

Através dos dados descritos na tabela 6.4, é possivel concluir que o algoritmo FHCA é melhor na questão da clusterização dos dados e o sistema de alarme proposto detecta muito mais intervalos corretamente, com uma taxa de falso positiva relativamente menor. Para os mesmos intervalos estudados, o PSO-CIs respondeu com uma taxa de falso positivo maior, se comparado ao FHCA, o que não é almejado.

A complexidade dos algoritmos é apresentado na figura 6.17. A complexidade do FHCA é menor que a complexidade apresentada pelo PSO-CIs, para uma avaliação de até 2 objetos. Para dimensões superiores, os papéis se invertem, PSO-CIs apresenta uma complexidade menor quando comparada ao FHCA. Nenhuma vantagem ou desvantagem acerca da complexidade para o aumento do número de dimensões pode ser concluido, para isto, um estudo mais aprofundado é necessário e não foi escopo deste trabalho.

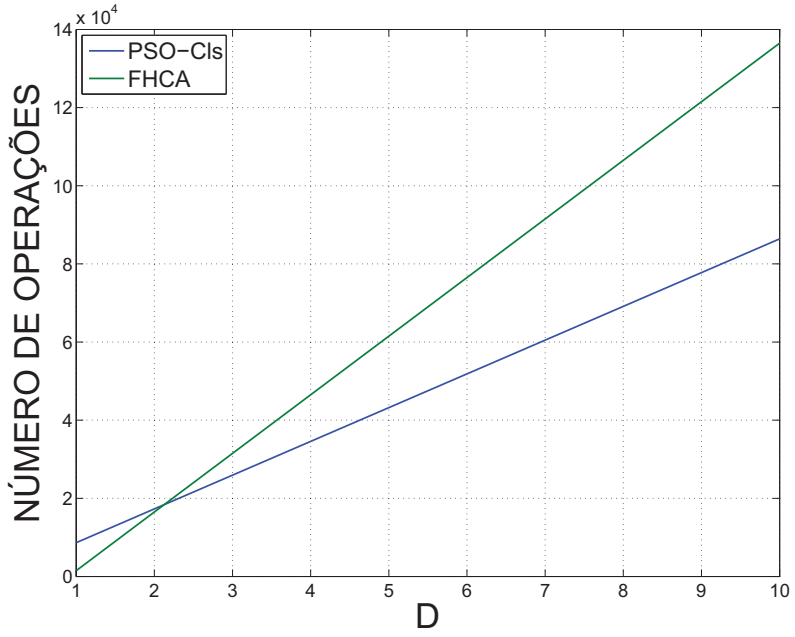

Figura 6.17: Gráfico da complexidade dos algoritmos FHCA e PSO-CIs.

7 CONCLUSÕES

Neste trabalho foi abordado o problema de detecção de anomalias em tráfego de redes de comunicação, através do monitoramento de objetos da *Management Information Base* (MIB). O domínio de aplicação utilizado foi a rede da Universidade Estadual de Londrina (UEL), que gera uma grande quantidade de dados que precisam ser monitorados em tempo real com o objetivo de manter a disponibilidade e a qualidade dos sistemas de ensino e pesquisa da universidade.

Visando otimizar o monitoramento e a qualidade de redes de comunicação, este trabalho inspirou-se nos resultados promissores obtidos pelas pesquisas de Proença e Lima [34, 7, 39], que consideram a utilização dos dados presentes nos objetos da MIB para a análise do tráfego em busca de anomalias, tendo como base a utilização de Assinatura Digital de Segmento de Rede (DSNS) [7], que conta com uma vasta base de dados histórica para a geração de perfis da rede da Universidade Estadual de Londrina. Esses trabalhos são baseados na aplicação de modelos estatísticos e determinísticos, que consistem de técnicas clássicas na literatura as quais vêm sendo amplamente utilizadas durante os últimos anos na área detecção de anomalias.

Com base na revisão bibliográfica realizada, nos trabalhos recentes encontrados na literatura, tem sido observado uma grande quantidade de pesquisas envolvendo a aplicação de técnicas heurísticas com algoritmos de clusterização. Com base nesse estudo, foi desenvolvido um sistema de detecção de anomalias (SDA) que utiliza uma abordagem de clusterização combinada com uma heurística aplicada ao DSNS e tráfego da rede.

7.1 Contribuições

Utilizando o DSNS como caracterização normal é proposto uma definição do que será tratado como anomalia. Grande parte das soluções de monitoramento existente atualmente apóia-se em operadores de rede que são responsáveis pela configuração manual

de *thresholds* ou limiares, o que consiste de uma atitude pouco eficiente e que não explora a natureza intrínseca da correlação entre os dados. Portanto, foi proposto um modelo para caracterizar uma anomalia de volume, onde, calcula-se uma área delimitada por limiares gerados através dos dados do DSNS e da própria amostra de tráfego. Este modelo, gera um gabarito; que indica se determinado dia é ou não anomalia. Este gabarito é utilizado para mensurar se o modelo de detecção de anomalias proposto atinge seus objetivos.

Pesquisas envolvendo a aplicação de técnicas heurísticas em conjunto com algoritmos de clusterização têm sido empregadas para a detecção de anomalias. A complexidade e a natureza instável das redes de comunicação são fatores para a escolha na utilização de modelos heurísticos, e nessa linha de pesquisa, foi desenvolvido o *Firefly Harmonic Clustering Algorithm* (FHCA). É um algoritmo de clusterização otimizado unindo o algoritmo de clusterização *K-Harmonic Means* (KHM) e a heurística *Firefly Algorithm* (FA).

Foram realizados diversos experimentos, sob a perspectiva de diferentes cenários, com objetivo de avaliar o desempenho e a precisão do SDA desenvolvido. O sistema foi testado através da aplicação de diferentes objetos SNMP, tais como o *ipInReceives*, *ipInDelivers*, *ifInOctets* e *tcpInSegs*. Foram avaliados os resultados obtidos através da aplicação do sistema para objetos únicos, bem como para objetos monitorados simultaneamente. Os dados utilizados nos experimentos foram coletados no ambiente de rede da Universidade Estadual de Londrina, em diferentes períodos de tempo.

Os resultados obtidos demonstraram que o SDA proposto é uma solução promissora, apresentando como resultados taxa de detecção 80% e falso positivo 25%. Foram criados alguns cenários para melhor avaliação do desempenho. Num primeiro momento, houve uma preocupação maior em avaliar os parâmetros do algoritmo, que foram testados e definidos para melhorar o desempenho dos testes efetuados posteriormente. Num segundo cenário, foi testado o algoritmo FHCA e comparado aos intervalos determinados pela definição, apresentando um ótimo desempenho. Em seguida, o algoritmo foi confrontado com o K-Means e o PSO-Cls. No primeiro confronto, foi análisado a questão de clusterização dos dados, e no segundo, o modelo de detecção como um todo foi análisado. FHCA demonstrou ser uma solução mais interessante, pelas taxas de detecção apresentadas serem superiores.

Os trabalhos futuros incluem o aperfeiçoamento da análise simultânea dos objetos da MIB, a fim de reduzir o ruído apresentado nas distâncias Euclidianas, e com isso obter uma redução na taxa de alarmes falsos. Também pretende-se, incluir um módulo para análise e classificação das anomalias, e também o desenvolvimento de um modelo para análise do tráfego baseado em pacotes IP, o que possibilitaria uma investigação mais profunda a

respeito das anomalias de rede. A utilização de técnicas para redução de dimensionalidade dos dados, também pode vir a ser combinada com o SDA desenvolvido, quando o número de objetos analisados simultâneamente crescer demasiadamente.

Referências

1 DUNN, J. Well separated clusters and optimal fuzzy partitions. *Journal of Cybernetics*, v. 4, p. 95–104, 1974.

2 DAVIES, D. L.; BOULDIN, D. W. A Cluster Separation Measure. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, n. 2, p. 224–227, 1979.

3 ROUSSEEUW, P. J. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, Elsevier Science Publishers B. V., v. 20, n. 1, p. 53–65, 1987.

4 PATCHA, A.; PARK, J. An overview of anomaly detection techniques: Existing solutions and latest technological trends. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, Elsevier North-Holland, Inc., New York, NY, USA, v. 51, p. 3448–3470, August 2007. ISSN 1389-1286.

5 CHANDOLA, V.; BANERJEE, A.; KUMAR, V. Anomaly detection: A survey. *ACM Computing Surveys.*, v. 41, n. 3, 2009.

6 HODGE, V. J.; AUSTIN, J. A survey of outlier detection methodologies. *Artif. Intell. Rev.*, v. 22, n. 2, p. 85–126, 2004.

7 PROENÇA JR., M. L. et al. Security and reliability in information systems and networks: Baseline to help with network management. In: ASCENSO, J. et al. (Ed.). *e-Business and Telecommunication Networks*. Springer Netherlands, 2006. p. 158–166. ISBN 978-1-4020-4761-9. Disponível em: <http://dx.doi.org/10.1007/1-4020-4761-4_12>.

8 ZHANG, B.; HSU, M.; DAYAL, U. *K-harmonic means - a data clustering algorithm*. Palo Alto, Outubro 1999.

9 YANG, X. Firefly algorithms for multimodal optimization. In: WATANABE, O.; ZEUGMANN, T. (Ed.). *SAGA*. Springer, 2009. (Lecture Notes in Computer Science, v. 5792), p. 169–178. ISBN 978-3-642-04943-9. Disponível em: <http://dblp.uni-trier.de/db/conf/saga/saga2009.html>.

10 THOTTAN, M.; JI, C. Anomaly detection in IP networks. *IEEE Transactions on Signal Processing*, v. 51, n. 8, p. 2191–2204, August 2003.

11 LIU, W. Research on DoS attack and detection programming. In: *Proceedings of the 3rd international conference on Intelligent information technology application*. Piscataway, NJ, USA: IEEE Press, 2009. (IITA'09, v. 1), p. 207–210. ISBN 978-1-4244-5212-5.

12 ELLIS, D. Worm anatomy and model. In: STANIFORD, S.; SAVAGE, S. (Ed.). *WORM*. ACM Press, 2003. p. 42–50. ISBN 1-58113-785-0. Disponível em: <http://dblp.uni-trier.de/db/conf/worm/worm2003.html>.

13 GADGE, J.; PATIL, A. Port scan detection. In: *16th IEEE International Conference on Networks*. USA: IEEE Press, 2008. (ICON), p. 1–6. ISSN 1556-6463.

14 ESTÉVEZ-TAPIADOR, J. M.; GARCIA-TEODORO, P.; DíAZ-VERDEJO, J. E. Anomaly detection methods in wired networks: a survey and taxonomy. *Computer Communications*, v. 27, n. 16, p. 1569–1584, 2004.

15 DENNING, D. An intrusion-detection model. *Software Engineering, IEEE Transactions on*, SE-13, n. 2, p. 222–232, February 1987. ISSN 0098-5589.

16 ZARPELÃO, B. B. *Detecção de Anomalias em Redes de Computadores*. Tese (Doutorado) — Universidade Estadual de Campinas (UNICAMP). Faculdade de Engenharia Eletrica e de Computação (FEEC)., 2010.

17 JAIN, A.; MURTY, M.; FLYNN, P. Data clustering: A review. *ACM Computing Survey*, v. 31, n. 3, p. 264–323, 1999.

18 JIANLIANG, M.; HAIKUN, S.; LING, B. The application on intrusion detection based on k-means cluster algorithm. In: *Proceedings of the 2009 International Forum on Information Technology and Applications - Volume 01*. Washington, DC, USA: IEEE Computer Society, 2009. p. 150–152. ISBN 978-0-7695-3600-2. Disponível em: <http://portal.acm.org/citation.cfm?id=1606748.1606787>.

19 ZHANG, C.; ZHANG, G.; SUN, S. A mixed unsupervised clustering-based intrusion detection model. In: . Los Alamitos, CA, USA: IEEE Computer Society, 2009. v. 0, p. 426–428. ISBN 978-0-7695-3899-0.

20 RAMASWAMY, S.; RASTOGI, R.; SHIM, K. Efficient algorithms for mining outliers from large data sets. In: *SIGMOD '00: Proceedings of the 2000 ACM SIGMOD international conference on Management of data*. New York, NY, USA: ACM, 2000. p. 427–438. ISBN 1-58113-217-4.

21 ENSAFI, R. et al. Optimizing fuzzy k-means for network anomaly detection using pso. In: *AICCSA 2008. IEEE/ACS International Conference on Computer Systems and Applications*. USA: IEEE Press, 2008. p. 686–693.

22 YANG, X. *Nature-Inspired Metaheuristic Algorithms*. UK: Luniver Press, 2008. ISBN 1-905986-10-6, 978-1-905986-10-1.

23 JUMADINOVA, J.; DASGUPTA, P. Firefly-inspired synchronization for improved dynamic pricing in online markets. In: BRUECKNER, S. A.; ROBERTSON, P.; BELLUR, U. (Ed.). *Second IEEE International Conference on Self-Adaptive and Self-Organizing Systems*. USA: IEEE Computer Society, 2008. (SASO '08), p. 403–412. ISBN 978-0-7695-3404-6.

24 CHRISTENSEN, A. L.; O'GRADY, R.; DORIGO, M. From fireflies to fault-tolerant swarms of robots. *IEEE Trans. Evolutionary Computation*, v. 13, n. 4, p. 754–766, 2009.

25 PHAM, D. T. et al. Data clustering using the bees algorithm. In: *Proc. 40th CIRP Int. Manufacturing Systems Seminar*. Liverpool: [s.n.], 2007.

26 XIAO, L.; SHAO, Z.; LIU, G. K-means algorithm based on particle swarm optimization algorithm for anomaly intrusion detection. In: *The Sixth World Congress on Intelligent Control and Automation*. USA: IEEE Press, 2006. (WCICA 2006, v. 2), p. 5854–5858.

27 BREABAN, M. E.; LUCHIAN, H. PSO aided k-means clustering: introducing connectivity in k-means. In: *Proceedings of the 13th annual conference on Genetic and evolutionary computation*. New York, NY, USA: ACM Request Permissions, 2011. (GECCO '11), p. 1227–1234.

28 GÜNGÖR, Z.; ÜNLER, A. K-harmonic means data clustering with simulated annealing heuristic. *Applied Mathematics and Computation*, v. 184, n. 2, p. 199–209, 2007.

29 CASE, J. D. et al. *Simple Network Management Protocol (SNMP)*. United States: RFC Editor, 1990.

30 CASE, J. D. et al. *Introduction to Community-based SNMPv2*. United States: RFC Editor, 1996.

31 MCCLOGHRIE, K.; ROSE, M. *Management information base for network management of tcp/ip-based internets: Mib 2*. United States: RFC Editor, 1991.

32 LIM, S.; JONES, A. Network Anomaly Detection System: The State of Art of Network Behaviour Analysis. In: *International Conference on Convergence and Hybrid Information Technology*. USA: IEEE, 2008. (ICHIT '08), p. 459–465. ISBN 978-0-7695-3328-5.

33 ABUSINA, Z. U. M. et al. An engineering approach to dynamic prediction of network performance from application logs. *Int. Journal of Network Management*, v. 15, n. 3, p. 151–162, 2005.

34 PROENÇA JR., M. L. *Baseline aplicado a gerencia de redes*. Tese (Doutorado) — Universidade Estadual de Campinas (UNICAMP). Faculdade de Engenharia Eletrica e de Computação (FEEC)., 2005.

35 BARFORD, P. et al. A signal analysis of network traffic anomalies. In: *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment*. New York, NY, USA: ACM, 2002. (IMW '02), p. 71–82. ISBN 1-58113-603-X.

36 MACQUEEN, J. B. Some methods for classification and analysis of multivariate observations. In: *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*. USA: University of California Press, 1967. v. 1, p. 281–297.

37 YANG, F.; SUN, T.; ZHANG, C. An efficient hybrid data clustering method based on k-harmonic means and particle swarm optimization. *Expert Syst. Appl.*, v. 36, n. 6, p. 9847–9852, 2009.

38 FAWCETT, T. An introduction to ROC analysis. *Pattern Recognition Letters*, v. 27, p. 861–874, 2005.

39 LIMA, M. F. et al. Anomaly detection using baseline and k-means clustering. In: *International Conference on Software, Telecommunications and Computer Networks*. USA: IEEE, 2010. (SoftCOM), p. 305–309. ISBN 978-1-4244-8663-2.

Trabalhos publicados pelo autor

1. Anomaly Detection Using Firefly Harmonic Clustering Algorithm. Mario H. A. C. Adaniya, Moisés F. Lima, Lucas D. H. Sampaio, Taufik Abrão, Mario Lemes Proença Jr. Proceedings of the International Conference on Data Communication Networking and International Conference on Optical Communication System. ICETE. Seville, Espanha, 2011.

2. Anomaly Detection Using DSNS and Firefly Harmonic Clustering Algorithm. Mario H. A. C. Adaniya, Moisés F. Lima, Joel J. P. C. Rodrigues, Taufik Abrão, Mario Lemes Proença Jr. Proceedings of the International Conference in Communications. ICC. Ottawa, Canadá, 2012.