UNIVERSIDADE FEDERAL DE MINAS GERAIS
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

LEANDRO PFLEGER DE AGUIAR

DESCOBERTA DE PADRÕES DE ALARMES REDUNDANTES COM TÉCNICAS DE MINERAÇÃO DE DADOS E REDES COMPLEXAS

Belo Horizonte
2010

LEANDRO PFLEGER DE AGUIAR

DESCOBERTA DE PADRÕES DE ALARMES REDUNDANTES COM TÉCNICAS DE MINERAÇÃO DE DADOS E REDES COMPLEXAS

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Departamento de Ciência da Computação da Universidade Federal de Minas Gerais para obtenção do grau de Mestre em Ciência da Computação.

Linha de Pesquisa: Descoberta de Conhecimento (KDD) e Redes Complexas

Orientador: Virgílio A. F. Almeida

BELO HORIZONTE

2010

Ficha catalográfica:

Universidade Federal de Minas Gerais
Departamento de Ciência da Computação
Programa de Pós-Graduação em Ciência da Computação

Dissertação intitulada Descoberta de Padrões de Alarmes Industriais Redundantes com Técnicas de Mineração de Dados e Redes Complexas, de autoria de Leandro Pfleger de Aguiar, aprovada pela banca examinadora constituída pelos seguintes professores:

_____

Prof. PhD. VIRGÍLIO AUGUSTO FERNANDES ALMEIDA - ORIENTADOR
Departamento de Ciência da Computação - UFMG

_____

Prof. PhD. WAGNER MEIRA JR.
Departamento de Ciência da Computação - UFMG

_____

Prof. PhD. FABIO GONÇALVES JOTA
Departamento de Engenharia Eletrônica - UFMG

_____

Prof. PhD. GISELE L. PAPPA
Departamento de Ciência da Computação - UFMG

_____

Prof. José Marcos Silva Nogueira
Coordenador(a) do Programa de Pós-Graduação em Ciência da Computação - PPGCC
DCC/UFMG

Data de aprovação: Belo Horizonte, de ..... de 20....

Aos meus pais, que me ensinaram a humildade antes que eu vivenciasse o orgulho, e à minha amada esposa, Terumí, que me apoia em todos os momentos e em todas as decisões, a quem devo, pela paciência e perseverança, eternamente compartilhar os frutos deste trabalho.

AGRADECIMENTOS

Agradeço à Universidade Federal de Minas Gerais, com todos os seus professores e funcionários que me acolheram, e em especial, aos bons professores do DCC que reconhecem a sua importância e fazem do ensino da Computação um orgulho, uma inspiração.

Ao meu orientador, Virgílio Almeida, de capacidade e conhecimento ímpares, pelo suporte em todo o tempo e em todas as minhas decisões.

Aos membros da banca, que contribuíram com revisões minuciosas para a melhoria do trabalho.

Agradeço à empresa Chemtech, pela flexibilidade e pelo incentivo à pesquisa.

Agradeço especialmente aos engenheiros Daniel dos Santos Ferreira Soares e Magno Lucio de Araujo, pelas grandes contribuições na dissertação, e à engenheira Ludmila Villela Morais Almeida, que me acompanhou e me deu suporte durante boa parte deste projeto, compartilhando seu conhecimento de causa e justificando meus achados.

Finalmente, agradeço a todos os demais que contribuíram de alguma forma ou permitiram que este trabalho e os seus resultados viessem à tona.

“Pouco conhecimento faz com que as pessoas se sintam orgulhosas. Muito conhecimento, que se sintam humildes. É assim que as espigas sem grãos erguem desdenhosamente a cabeça para o Céu, enquanto que as cheias as baixam para a terra, sua mãe.”

Leonardo da Vinci

RESUMO

A área de gerenciamento de alarmes vem crescendo rapidamente e se destacando em tópicos de processos industriais. Sistemas de alarmes mal projetados ou com funcionamento inadequado já foram causas de diversos incidentes e grandes acidentes em plantas industriais ao redor do mundo, causando prejuízos financeiros e perdas de vidas. Uma das etapas mais importantes dentro das metodologias de gerenciamento é a racionalização dos alarmes, em que o volume de eventos gerados por dia é reduzido a um número adequado para que um ser humano possa compreendê-los e tratá-los. Uma das maiores dificuldades neste processo é identificar, dentre arquivos e bases de dados contendo dezenas de milhares de registros diários, padrões que possam indicar alarmes desnecessários, dado que os padrões temporais que correlacionam estes alarmes podem ser complexos por serem dependentes de variáveis de processo. Além da análise de correlação cruzada, técnica recomendada pelas normas da área para a solução deste problema, poucas tentativas foram feitas em termos de desenvolvimento de uma técnica adequada especificamente para a descoberta dos padrões de alarmes industriais redundantes. Nenhuma destas tentativas foi suficientemente capaz de realizar a atividade de maneira abrangente, com resultados previsíveis e confiáveis e complexidade gerenciável.

Este trabalho apresenta uma alternativa às soluções atualmente utilizadas como técnicas de mineração dentro do processo de racionalização de alarmes, capaz de tratar algumas das dificuldades práticas enfrentadas pelas soluções anteriores e facilitar o processo. A proposta consiste em uma nova abordagem que combina mineração de sequências, mineração de regras de associação com MNR (Regras de Associação Mínimas Não Redundantes), análise de correlação cruzada e modelagem de redes complexas na visualização dos resultados, criando uma alternativa mais abrangente em termos de padrões detectados. O desempenho em termos de exatidão da solução comprova melhorias em relação à melhor abordagem existente, resultando em uma alternativa mais confiável e previsível na identificação de padrões significativos.

Palavras-chave: Gerenciamento de alarmes. Descoberta de conhecimento. Mineração de sequências. Regras de associação. Redes complexas.

ABSTRACT

Alarm management is a research area that is growing rapidly on industrial automation topics. One of the major challenges in alarm rationalization, in which the volume of generated alarms is reduced to an appropriate number so that a human being can handle them, is to identify, between files and databases containing tens of thousands of daily records, patterns that might indicate unnecessary alarms. This work presents a new approach which combines sequence mining, association rules extraction with MNR (Minimum Non Redundant Association Rules), cross-correlation analysis, and complex network modeling for visualization, creating a more comprehensive alternative to the detection process. The solution´s performance in terms of accuracy shows improvements over the best existing approach, resulting in an more reliable and predictable alternative for the identification of meaningful patterns.

Key-words: Alarm management, Knowledge Discovery KDD, Sequence Mining, Association Rules, Complex Networks.

LISTA DE FIGURAS

Figura 1 - Exemplo de Sistema de Supervisão e Controle (SCADA) de Uma Usina Hidrelétrica. 1

Figura 2 - Relação de Quantidade de Alarmes Gerados por Indústrias Típicas no Mercado e Recomendação da Norma EEMUA 191 [1] 2

Figura 3 - Exemplo de Um Arquivo de *Log* de um Sistema SCADA[5] 7

Figura 4 - Exemplo da Não Trivialidade do Problema da Detecção de Alarmes em Cascata em Sistemas Industriais 7

Figura 5 - Cross Correlograma: Diferentes Valores de Lags Temporais (de -2 a +2 segundos) são Aplicados para Identificar o Maior Valor de Correlação (encontrado no lag -2s) 9

Figura 6 - Janelas W A-R e W R-A do Algoritmo Cross-effect Test de KORDIC et al. 2008 [13] 13

Figura 7 - Sugestão de Representação da Força da Conexão Entre Nós de Uma Rede Proposta por Leskovek et al. 18

Figura 8 – Possíveis Relações Entre Períodos de Tempo (ALLEN e FERGUSON, 1994) 21

Figura 9 - Diagrama P&ID (*Process and Instrumentation Diagram*) Ilustrativo Hipotético 23

Figura 10 - Função de Massa de Probabilidade para Duração dos Alarmes Considerando a Base de Dados de Alarmes com uma Única Classe (a) ou com Duas Classes: Usina e Britagem (b) 32

Figura 11 - Função de Massa de Probabilidade para Distribuições de Intervalos de Início (a) e Fim (b) para as Duas Classes de Componentes da Carga: Usina e Britagem 33

Figura 12 - Diagrama de Dispersão Quantidade de Alarmes/Dia Usina em Função da Quantidade de Alarmes/Dia Britagem 35

Figura 13 - Ajuste de Curva para Duração dos Alarmes: Normal, Lognormal, Poisson e Exponencial (melhor ajuste) 37

Figura 14 - Distribuições da Duração Real da Britagem (violeta), Duração na Curva Ajustada (preto) e Curva da Duração Gerada pelo Gerador Sintético (verde) 38

Figura 15 - Ajuste de Curva para Duração dos Alarmes: Normal, Logistic, Poisson e Exponencial (melhor ajuste) 39

Figura 16 - Distribuições dos Intervalos entre CFNs Real da Britagem (violeta), na Curva Ajustada (preto) e na Curva dos Intervalos Gerados pelo Gerador Sintético (verde) 40

Figura 17 - Diagrama de Classes do Gerador de Cargas 42

Figura 18 - Redes Ponderadas com Frequência e Peso para Valor de Corte de Arestas Superiores a 0,1 51

Figura 19 - Redes Ponderadas com Frequência e Peso para Valor de Corte de Arestas Superiores a 0,5 51
Figura 20 - Grafo Gerado com Corte de Arestas Inferiores a 2 com Ponderação por Peso e 60 Segundos de Janela 53
Figura 21 - Uso das janelas WA-R e WR-A na Formação das Transações 57
Figura 22 - Posição das Regras Fechadas e MNR diante de Todas as Regras de Associação. LASZLO, 2006 [29] 62
Figura 23 - Exemplo da visualização de um conjunto de padrões: a espessura das arestas indica a força da relação encontrada e o tamanho dos vértices (que representam os tipos diferentes de alarmes), representam a quantidade relativa deste alarme no conjunto de dados (suporte). 67
Figura 24 - *Exemplo de visualização de um padrão: existe uma relação de precedência com força 100% (1.0 na aresta de cor verde) que indica que após a ocorrência do alarme código 65, o alarme de código 66 também é iniciado (CFN). A aresta de cor vermelha representa o valor de correlação máximo encontrado, no deslocamento de 0 segundos de 0.68.* 67
Figura 25 - *Variação do parâmetro minw no algoritmo cross-effect-test para detectar o valor de menor erro* 69
Figura 26- *Erros de cross-effect-test vs nova abordagem em relação a cada padrão artificial inserido (sem cross-correlation. Eixo X corresponde aos erros encontrados e varia de 0 a 1 para cada um dos tipos e para a média e, de 0 a 6 para a Soma dos erros.* 70
Figura 27 - *Erros de cross-effect-test vs nova abordagem em relação a cada padrão artificial inserido (cross-correlation+PrefixSpan)* 70
Figura 28 - Média dos Erros para os Não-padrões 73
Figura 29 - Destaque dos Padrões Sintéticos Através da Separação dos Componentes Conexos no Grafo 76
Figura 30 - Padrões Sintéticos Encontrados com Corte de Relevância Menor que 0,5 77
Figura 31 - Sequência de Cortes de 1 a 3 e Destaque dos Componentes Conexos em 4 80
Figura 32 - Opções de Layout do Pajek para Visualização dos Padrões a) circular-random b) circular-tile-components c) energy-fruchterman-reingold-2d d) energy-kamada-kawai-separate-components (opções descritas em [32]) 81
Figura 33 – Visualização dos Padrões Minerados na Base de Dados Real da Britagem para Significância Superior a 0,2 (a), 0,3 (b), 0,9 (c) e 0,9 com os Componentes Conexos Identificados (d) 82
Figura 34 - Detalhes dos Padrões Encontrados na Lista de Componentes Conexos 83
Figura 35 – Visualização do Supervisório da Britagem Primária [35] 86
Figura 36 – Visualização do Supervisório da Britagem Secundária [35] 87
Figura 37 – Visualização do Supervisório do Peneiramento [35] 88

LISTA DE TABELAS

Tabela 1 - Variação da Métrica de Média dos Caminhos Entre Dois Vértices para o Grafo de Padrões 52

Tabela 2 - Exemplo sobre a desconsideração da ordem dos itens em uma transação na mineração de padrões frequentes [27] 59

Tabela 3- Cálculo do Intervalo de Confiança para a Média das Diferenças na Amostra Pareada 71

Tabela 4 - Cálculo dos Intervalos de Confiança para as Médias da Confiança dos Não-Padrões de Cross-effect-test e resultados da análise de RTN 74

Tabela 5 - Resultados do Teste-t na comparação das Médias de *cross-effect-test* e os resultados da análise com RTN 75

LISTA DE ABREVIATURAS

SCADA - Supervisory Control and Data Acquisition

DCS - Digital Control System

ISA - International Society of Automation

PHA - Process Hazard Analisys

SIL - Safety Interlock Level

LOPA - Layer Protection Analysis

CFN - Change from Normal

RTN - Return to Normal

PMF - Probability Mass Function

CDF - Cumulative Distribution Function

XEY - X equals Y / X igual a Y

XSY - X starts Y / X inicia Y

XFY - X finishes Y / X finaliza Y

YMX - Y met by X / início de Y coincidente com término de X

YOX - Y overlapped by X / Y sobreposto por X

YDX - Y during X / Y durante X

YOXTRIPLE - Y overlapped by X in Triple / X sobrepõe Y triplamente

SUMÁRIO

1 INTRODUÇÃO 1

1.1 MOTIVAÇÃO 1

1.2 CONTRIBUIÇÕES 3

1.3 ESTRUTURA DA DISSERTAÇÃO 4

2 CONCEITOS GERAIS E REVISÃO DA LITERATURA 5

2.1 RACIONALIZAÇÃO DE ALARMES 5

2.1.1 Supressão de Alarmes 6

2.2 IDENTIFICAÇÃO DE PADRÕES 8

2.2.1 Identificação de Padrões Através de Análise de Correlação 8

2.2.2 Identificação de Padrões Temporais Através de Mineração de Sequências e Mineração de Regras de Associação 9

2.2.3 Identificação de Padrões Através de Modelagem de Grafos 14

2.2.4 O Desafio da Visualização na Interpretação dos Resultados 16

2.3 FORMALIZAÇÃO DE CONCEITOS E DEFINIÇÃO DO PROBLEMA 19

2.3.1 Tipos de Padrões Temporais 21

2.3.2 Seleção da Medida de Interesse para a Força das Relações 24

2.3.3 Aspectos de Desempenho 25

3 METODOLOGIA 27

3.1 DADOS DE ENTRADA REAIS 27

3.2 DADOS DE ENTRADA SINTÉTICOS 29

3.2.1 Caracterização de Carga 29

3.2.2 Construção e Validação do Gerador 40

3.3 UMA ABORDAGEM SIMPLISTA 48

3.3.1 Algoritmo e Ferramentas 50

3.3.2 Resultados 52

3.3.3 Deficiências 53

3.4 DESCRIÇÃO DA SOLUÇÃO PARA A NOVA ABORDAGEM 54

3.4.1 Algoritmo de Geração de Transações com Janela Deslizante (*Sliding Window*) 55

3.4.2 Mineração de Sequências de Tamanho 2 com PrefixSpan 57

3.4.3 Mineração de Regras Complexas com MNR Mining 59
3.4.4 Visualização 64
4 RESULTADOS EXPERIMENTAIS 68
4.1 RESULTADOS NA MINERAÇÃO DA BASE DE DADOS SINTÉTICA 68
4.1.1 Análise dos Falsos Positivos 72
4.1.2 Análise dos Padrões Encontrados 76
4.1.3 Navegação nos Padrões 79
4.2 RESULTADOS NA MINERAÇÃO DA BASE DE DADOS REAL 81
5 CONCLUSÕES E TRABALHOS FUTUROS 89
5.1 CONCLUSÕES 89
5.2 TRABALHOS FUTUROS 92
REFERÊNCIAS 93

1 INTRODUÇÃO

1.1 Motivação

Em razão da grande importância que a área de Gerenciamento de Alarmes representa para o setor industrial, o tema virou assunto de diversos artigos, discussões e normas internacionais. Normas como a EEMUA 191 [1] e a ISA SP18 [2] fornecem orientações para a adoção de abordagens sistemáticas para a solução do problema, sugerindo metodologias e ferramentas. Uma das causas para a incapacidade dos operadores das plantas de entender e reagir ao surgimento de um alarme em um sistema de supervisão e controle (vide Figura 1 - SCADA) é o número excessivo de alarmes que é gerado na maior parte dos sistemas. É possível e comum encontrar sistemas em que o operador lida com centenas e até milhares de alarmes ocorrendo durante um único dia[1], o que torna a atividade de compreender o comportamento do processo e dos sistemas humanamente inviável.

Figura 1 - Exemplo de Sistema de Supervisão e Controle (SCADA) de Uma Usina Hidrelétrica.

Fonte: www.itaipu.gov.br

Com a adoção dos sistemas digitais, a adição de alarmes tornou-se uma atividade trivial, comparativamente ao esforço necessário no passado nos sistemas analógicos. Com isso, vários dos desenvolvedores de sistemas SCADA optaram pela

transferência da “responsabilidade” pela interpretação de um evento qualquer na planta ao operador, o que, normalmente, é uma ação menos custosa do que a realização de um projeto completo de engenharia [1] para tornar o sistema de alarmes adequado. Como consequência, muitas das ações como a abertura ou o fechamento de uma válvula, ou a interrupção de sensores, que normalmente deveriam ser tratados como simples eventos, são interpretados como alarmes, pressupondo indevidamente a necessidade de uma ação do operador.

Segundo a norma EEMUA 191, existe um número aceitável sobre a quantidade de alarmes que um operador de planta industrial é capaz de lidar por hora, cujo limite não pode ultrapassar a capacidade do ser humano de perceber, compreender e reagir ao alarme. De acordo com a Figura 2, é possível observar, no entanto, que o valor praticado pela maior parte das indústrias do mercado supera exageradamente os valores aceitáveis sugeridos pela norma [1].

	EEMUA	Óleo e Gás	Petroquímica	Energia	Outros
Média de Alarmes por Dia	144	1200	1500	2000	900
Média de Alarmes Críticos	9	50	100	65	35
Pico de Alarmes por 10 minutos	10	220	180	350	180
Média de Alarmes por 10 minutos de intervalo	1	6	9	8	5
Distribuição percentual da criticidade dos alarmes (Baixa/Média/Alta)	80/15/5	25/40/35	25/40/35	25/40/35	25/40/35

Figura 2 - Relação de Quantidade de Alarmes Gerados por Indústrias Típicas no Mercado e Recomendação da Norma EEMUA 191 [1]

Além da questão do tempo necessário para tratar uma quantidade elevada de alarmes, o excesso de informação pode também distrair o operador, tirando a sua atenção de problemas mais graves que podem ocorrer. Um exemplo disso [3] ocorreu em Julho de 1994, na cidade de Milfard Haven, Reino Unido, em uma refinaria da Texaco, onde a causa de uma grande explosão seguida de incêndio foi atribuída principalmente ao excessivo número de alarmes gerado em uma situação de emergência. Após uma perturbação inicial

que gerou uma grande quantidade de alarmes durante 5 horas, os operadores foram incapazes de perceber nos alarmes gerados uma situação mais grave que conduziu à catástrofe.

A otimização dos alarmes é feita, tipicamente, através de projetos de engenharia nomeados como processos de Racionalização de Alarmes [4], onde estudos são aplicados com o intuito de identificar, entender e eliminar alarmes desnecessários. Uma parte importante destes projetos é a análise para a descoberta dos alarmes duplicados dinâmicos, que são os alarmes que ocorrem de forma periódica, de acordo com algum padrão temporal que pode variar dependendo das características do processo e da forma de construção e configuração do sistema. O tratamento destes alarmes pode reduzir um número elevado de ocorrências, melhorando o sistema, mas o levantamento dos padrões que representam tais alarmes pode ser uma atividade complexa, exigindo o uso de ferramentas de suporte.

Além da análise de correlação cruzada, principal técnica recomendada para a solução deste problema [1], poucas tentativas foram feitas em termos de desenvolvimento de uma técnica adequada especificamente para a descoberta dos padrões de alarmes industriais redundantes. Nenhuma destas tentativas foi suficientemente capaz de realizar a atividade de maneira abrangente, com resultados previsíveis e confiáveis e complexidade gerenciável.

Este trabalho apresenta uma alternativa, através de uma combinação inovadora de algoritmos, às soluções atualmente utilizadas como técnicas de mineração dentro do processo de racionalização de alarmes, capaz de tratar algumas das dificuldades práticas enfrentadas pelas soluções existentes. A exatidão da solução, demonstrada pela primeira vez dentre os trabalhos neste tema, comprova melhorias em relação à melhor abordagem existente, resultando em uma alternativa mais confiável e previsível na identificação de padrões significativos.

1.2 Contribuições

Este trabalho apresenta como principais contribuições os seguintes pontos:

- Três técnicas são combinadas: análise de correlação cruzada, mineração de sequências mineração de regras de associação mínimas não redundantes, explorando as vantagens de cada abordagem e a sua complementaridade, ampliando as chances de interpretação correta pelo minerador;

- A capacidade de detecção de diferentes tipos de padrões em relação a todas as soluções anteriores é ampliada, uma vez que o algoritmo foi construído com base na exploração de todas as relações temporais possíveis e não apenas sobre um tipo de relação temporal, como nos trabalhos anteriores;

- A quantidade e influência dos parâmetros nos resultados do processo de mineração é reduzida através da combinação de soluções existentes;

- A visualização dos padrões é simplificada e o significado dos padrões minerados é ampliado através da apresentação dos resultados na forma de redes complexas, com vértices e arestas representando os alarmes e as relações temporais direcionais;

- O algoritmo foi executado utilizando uma base de dados real e uma base de dados sintética, gerada a partir de um processo de caracterização de carga dos dados reais. Um gerador de cargas sintéticas foi construído para este propósito.

1.3 Estrutura da Dissertação

Esta dissertação está organizada da seguinte forma: o capítulo 2 fornece uma visão geral dos conceitos envolvidos no problema e possíveis soluções e introduz as alternativas existentes. No capítulo 3 é descrita a metodologia, envolvendo o processo de caracterização e geração da base de dados de alarmes sintética e a descrição da solução propriamente dita. O capítulo 4 apresenta os resultados experimentais na execução na base de dados sintética e também na base de dados real. Por fim, o capítulo 5 destaca as principais conclusões e contribuições deste trabalho e aponta as direções para trabalhos futuros.

2 CONCEITOS GERAIS E REVISÃO DA LITERATURA

2.1 Racionalização de Alarmes

A racionalização dos alarmes, segundo HOLLIFIELD e HABIBI [4], é um dos sete passos necessários para a criação de um sistema de alarmes altamente efetivo, sendo os resultados desta efetividade expressos em termos de segurança, confiabilidade, disponibilidade, performance e, principalmente, lucratividade. O tratamento eficiente dos alarmes reduz custos operacionais, como os custos de manutenção de equipamentos, evitam paradas da planta que causam prejuízos financeiros diretos, melhoram a produtividade geral dos processos, colocando os equipamentos a operar em um ponto ótimo, e reduzem os valores gastos com seguros, já que a operação inadequada cria altos riscos de incidentes, penalidades ambientais e de danos aos equipamentos.

A racionalização envolve a utilização de uma metodologia onde os alarmes são determinados, priorizados e documentados [4]. Nesta atividade, cada um dos alarmes é re-examinado em busca de uma ou mais das seguintes ações:

1) Corrigir sistemas mal configurados que estão gerando alarmes em situações que não exigem intervenção do operador;
2) Garantir a consistência dos alarmes, em relação ao significado para o processo;
3) Configurar os limiares (*thresholds*) para o disparo dos alarmes para valores que gerem alarmes em situações onde realmente é necessária intervenção;
4) Configurar os limiares para atender às exigências estipuladas pelos processos de Análise de Risco PHA – *Process Hazard Analisys*, SIL – *Safety Interlock Level* e LOPA – *Layer Protection Analysis*;
5) Eliminar alarmes duplicados ou alarmes que ocorram sob condições idênticas;
6) Eliminar ou reduzir a ocorrência de alarmes em cascata (*alarm flood*);
7) Identificar lógicas de surgimento de alarmes e incluir estas lógicas para reduzir a quantidade de alarmes apresentados;
8) Identificar causa raiz para conjuntos de alarmes para agrupar tais conjuntos com significado único através da inserção de lógicas no sistema.

As ações 5, 6, 7 e 8 são ações de racionalização que apresentam um potencial de redução do número de alarmes muito elevado, já que se relacionam a problemas no

sistema de alarmes muito comuns, como a duplicação. O seu tratamento consiste essencialmente na aplicação da solução de agrupamento e de supressão, apresentada a seguir.

2.1.1 Supressão de Alarmes

A supressão de alarmes consiste em resumir os alarmes existentes, apresentando-os sem redundância ao operador somente quando a situação da planta tornar a apresentação relevante.

A supressão normalmente atua na remoção de dois tipos característicos de alarmes: os alarmes duplicados e os alarmes duplicados dinâmicos, explicados a seguir. Há casos em que uma determinada variável do processo é medida em dois pontos, ou ainda há casos em que uma medida é enviada de um sensor a um ponto de controle ou totalizador, o que pode resultar em eventos simultâneos no sistema, caracterizando um alarme duplicado. A eliminação neste caso consiste na identificação, através dos registros de eventos (*logs*), das repetições de maior frequência para posterior interpretação semântica do processo. Nem sempre a correção consiste na remoção do alarme duplicado. Os alarmes duplicados dinâmicos são mais difíceis de se racionalizar, uma vez que, nem sempre as relações entre os alarmes são facilmente destacáveis sem o auxílio de alguma ferramenta. Em alguns casos, os alarmes podem estar relacionados a múltiplos anúncios diferentes do mesmo evento ou ainda a múltiplas consequências para uma mesma condição de anormalidade, situações que podem ser difíceis de identificar dentre centenas de tipos diferentes de alarmes distintos.

Suprimir alarmes duplicados dinâmicos pode ser considerada, na grande maioria das vezes, uma atividade complexa, dada a natureza que se configura. Tipicamente os registros de alarmes são feitos em arquivos de *logs* com milhares de entradas diárias, gerados automaticamente e com volumes que variam diariamente dependendo da estabilidade da planta.

A Figura 3 apresenta parte de um arquivo de *log* típico de um sistema SCADA [5]. Os eventos são ordenados pelo tempo e, a cada evento está também associado um “TAG” (código do alarme), que identifica este evento junto ao controlador e a qualquer outro sistema de automação que venha a manipulá-lo, um valor de criticidade (alto, muito alto, baixo, muito baixo, etc.), um identificador de entrada/saída do estado e uma descrição.

O que torna o problema da descoberta de padrões e correlações em alarmes de sistemas industriais um problema interessante e não trivial é justamente o fato de que as relações temporais entre os eventos sequenciais correlacionados, por dependerem diretamente de variáveis de processo, são complexas, desafiando a construção de algoritmos que os destaquem dos demais. Considere três alarmes A, B e C sabidamente correlacionados no processo e que ocorrem de acordo com o desenho da Figura 4. A falta de perfeição no sincronismo temporal[1] nas sequências dificultaria a aplicação da detecção, já que (a) a simples ativação dos alarmes simultâneos é inadequada, pois nem sempre os alarmes relacionados ocorrem ao mesmo tempo (b) o simples uso do coeficiente de correlação de Pearson, descrito em detalhes na seção 2.2.1 (Eq. 2.1), não seria capaz de detectar com precisão, já que, além de haver uma defasagem temporal, este deslocamento não é estacionário, e c) estes alarmes ocorrem em intervalos em que outras dezenas de alarmes poderiam estar acionados, dificultando a descoberta.

DATA	HORA	TAG	CFN/RTN	RESPOSTA	DESCRIÇÃO
4/19/2007	19:38:11 .	[SCADA_CE] CO13500_ATAH6	OK	NORMAL	CO-13500 Temp. alta do óleo após cooler
4/19/2007	19:38:17 .	[SCADA_CE] ES6601_APAL	OK	NORMAL	ES6601 - Baixa pressão
4/19/2007	19:38:24 .	[SCADA_CE] SL3081_ALAH_A	OK	NORMAL	SL-3081A Nível alto
4/19/2007	19:38:24 .	[SCADA_CE] SL3081_ALAH	OK	NORMAL	Nível alto 3081 A e B
4/19/2007	19:38:24 .	[SCADA_CE] SL4082_ALAH	CFN	ATUADO	SL4082 - Nível alto no silo

Figura 3 - Exemplo de Um Arquivo de *Log* de um Sistema SCADA[5]

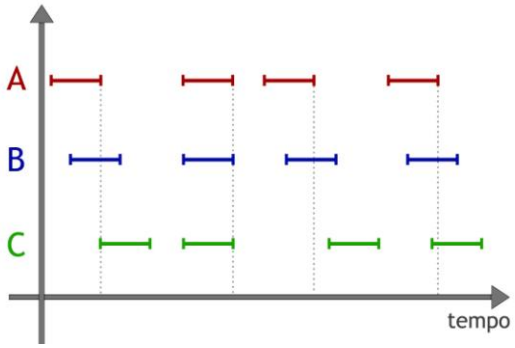

Figura 4 - Exemplo da Não Trivialidade do Problema da Detecção de Alarmes em Cascata em Sistemas Industriais

[1] Um exemplo de uma relação perfeita seria: início do alarme B = início do alarme A + 5s, para todas as ocorrências na base de dados. Esta precisão é muito difícil de se encontrar em casos práticos.

2.2 Identificação de Padrões

2.2.1 Identificação de Padrões Através de Análise de Correlação

Para que a supressão de alarmes possa ser praticada, antes é necessário identificar padrões dentre as bases de dados ou arquivos de *logs* com os registros dos eventos, que possam levar à eliminação de alarmes desnecessários. A norma EEMUA 191 [1] sugere a resolução deste problema através de técnicas da estatística, como análise de correlação ou correlação cruzada de variáveis, ou ainda através da interpretação semântica do conteúdo dos arquivos de *logs*. Na análise de Correlação, calculada através do coeficiente de correlação de Pearson[2] (Eq. 2.1), é utilizada a covariância para fornecer uma medida do grau no qual duas variáveis se movem em cada período no tempo. A correlação cruzada, definida por $f \phi g$ (Eq. 2.2), é uma medida de similaridade entre duas funções que também utiliza o coeficiente de correlação de Pearson, mas que aplica diferentes valores de *lags* temporais para detectar correlação deslocada no tempo. O sinal é discretizado através de uma janela de tempo deslizante, criando o "*sliding window cross-correlogram*" (vide esquema da Figura 5) sobre o qual a covariância é calculada para cada deslocamento (*shift*) positivo ou negativo da janela. O algoritmo funciona bem quando as relações entre os dois sinais são estacionários (*time-invariant*) em relação aos tempos de início, fim e duração dos período ativos, mas não gera bons resultados quando as correlações variam ao longo do tempo. O uso de correlação isoladamente é também incapaz de estabelecer relações mais complexas entre conjuntos maiores de alarmes, já que a correlação é feita sempre dois-a-dois, tornando difícil identificar relações entre grupos de alarmes.

$$\rho x,\ y = \frac{cov(x,\ y)}{\sigma x \sigma y} \qquad \textbf{(Eq. 2.1)}$$

$$f \phi g\left[t \right] \stackrel{def}{=} \sum_{s=-\infty}^{\infty} f^{*}\left[s \right] g\left[t+s \right] \qquad \textbf{(Eq. 2.2)}$$

[2] Detalhes sobre o conceito podem ser verificados em [25], no capítulo 12.1 (Conceitos Básicos de Probabilidade e Estatística).

A imperfeição da técnica de correlação cruzada para este propósito, em conjunto com a sua baixa performance, são razões que motivaram a própria norma EEMUA 191 a substituí-la [1] apenas pela interpretação semântica direta do conteúdo dos arquivos de *logs* por um profissional com experiência no processo produtivo. Esta alternativa, porém, dada a grande complexidade e dimensão possível da base de registros de *logs*, torna a sugestão da norma inviável, já que, para ser capaz de suprimir alarmes desnecessários eficientemente, o profissional teria que estabelecer relações mentais sobre possíveis correlações considerando as centenas de tipos de alarmes diferentes, cada qual ocorrendo em milhares ou milhões de registros em *log*.

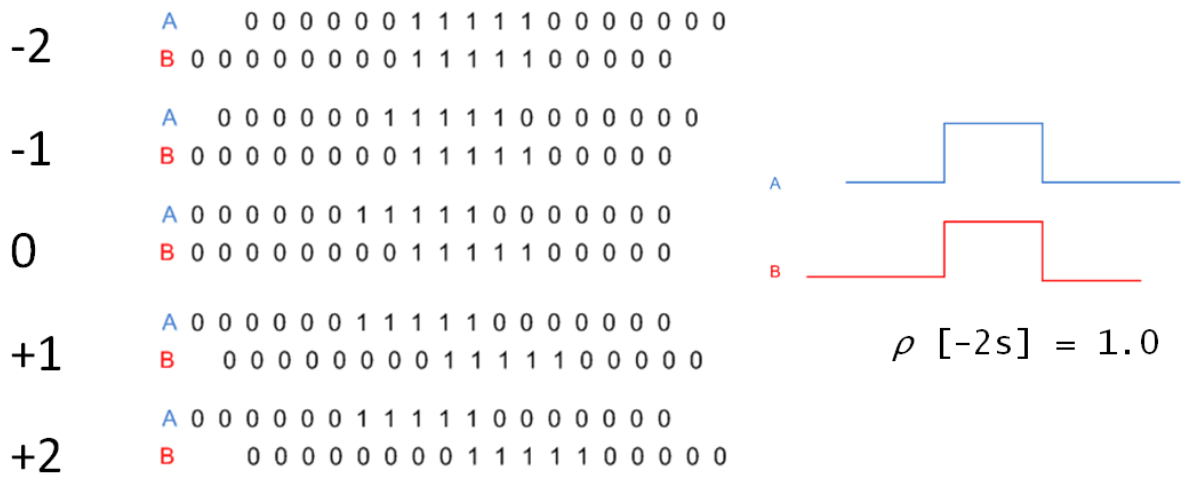

Figura 5 - Cross Correlograma: Diferentes Valores de Lags Temporais (de -2 a +2 segundos) são Aplicados para Identificar o Maior Valor de Correlação (encontrado no lag -2s)

2.2.2 Identificação de Padrões Temporais Através de Mineração de Sequências e Mineração de Regras de Associação

A detecção de cascatas ou correlações de alarmes já foi estudada anteriormente no campo de mineração de dados (KDD – *Knowledge Discovery in Databases*), embora os trabalhos (descritos a seguir nesta mesma seção) tenham se concentrado em alarmes como os que ocorrem em uma rede de computadores, onde a identificação da causa raiz é essencial para evitar cascatas em plataformas (softwares) de gerência de redes que monitoram pontos redundantes na rede [6]. Os alarmes de plantas industriais, entretanto, possuem características diferentes. Diferentemente dos originados em equipamentos como

"*switches*" e "roteadores", os alarmes de um sistema de controle industrial sempre são disparados por um evento e cessados por outro evento (também registrado em *log*), como uma intervenção do operador no sistema, por exemplo[3]. Fundamentalmente, as técnicas abordadas nas pesquisas já desenvolvidas dentro da disciplina de mineração, fazem uso da mineração de sequências utilizando janelas de tempo móveis (*sliding window*) para tratar a questão da ordenação temporal.

Desde a proposição da primeira versão da técnica de mineração de sequências (vide definição formal do conceito na seção 2.3) que incluiu o conceito de janela de tempo como restrição, com o algoritmo GSP proposto em [7] por Srikant e Agrawal, em 1996 (uma generalização da versão original do algoritmo de mineração de sequências proposto pelos próprios autores no ano anterior [8]), alguns autores já tentaram explorar a técnica em contextos similares. Em [9], Manilla et al. definiram o conceito de episódios, conjuntos frequentes que ocorrem em sequências de eventos em determinados intervalos de tempo e em determinada ordem. Os autores propuseram um *framework* baseado no trabalho de Srikant e Agrawal capaz de encontrar sequências seriais ou paralelas de eventos de uma rede de telecomunicações com dois algoritmos: WINEPI e MINEPI. O WINEPI reconhece os episódios através de uma "janela deslizante" que passa pela sequência de entrada contando a frequência com que cada episódio aparece dentre o total de janelas geradas. O MINEPI faz a contagem das ocorrências mínimas dos episódios (*mo – minimal occurrences*) com relação a limiares de tempo especificados pelo usuário. Em [6], Harle e Gardner propuseram uma extensão do trabalho de Manilla através de um *framework* para o estabelecimento de correlações de eventos de alarmes em uma rede de telecomunicações. Em [10], ZAKI et al. introduziram o algoritmo de mineração de sequências PLANMINE para predizer eventos de falha em redes de computadores através da análise de registros em *logs*. Fundamentalmente, a técnica utiliza mineração de sequências para identificar causas de falhas e realimentar ferramentas de planejamento com os padrões descobertos para melhorar e predizer situações futuras.

Além das diferenças já mencionadas nas características dos alarmes industriais em relação aos explorados nestes trabalhos, todos enfrentam a dificuldade de ter que especificar um tamanho adequado para a janela deslizante para que os padrões possam ser identificados. Se a janela for muito grande, muito ruído é adicionado dentro de cada janela, aumentando o tamanho médio dos episódios (já que em uma janela mais abrangente "cabem" mais ocorrências). Se a janela for muito pequena, então haverá perda de

[3] Essencialmente, o conceito formal de alarme é definido como um alerta que pressupõe uma ação ou intervenção humana [1].

informação, e padrões maiores podem não ser detectados. Além do tamanho da janela e limiares de tempo para o WINEPI, é necessário estipular também o valor de suporte mínimo a partir do qual os padrões são considerados frequentes. De fato, todos os autores reconhecem que a qualidade dos resultados está fortemente ligada à necessidade de especificação de bons parâmetros pelo usuário.

Em geral, a solução para este problema descrito acima se resume à sugestão de várias execuções variando os valores dos parâmetros aliadas a um forte conhecimento do domínio do problema. Tal sugestão, além de onerar o custo da atividade de mineração, elimina a possibilidade de automatização da tarefa, já que parâmetros bons são essenciais para bons resultados. Em [11], em um estudo sobre estado e direções futuras para a mineração de padrões frequentes, HAN et al. reforçam esta ideia, mencionando a necessidade de automatização das tarefas de mineração para reduzir a dependência dos algoritmos do conhecimento do usuário. Eles afirmam ainda que uma direção promissora de pesquisa consiste na redução do número de entradas (parâmetros), com foco em adequar as técnicas à realidade em termos de capacidade dos usuários que efetivamente as utilizam.

Fora da área de pesquisa acadêmica, alguns softwares comerciais já foram criados com o intuito de promover o gerenciamento de alarmes. Em geral, soluções deste tipo como *Matrikon Alarm Management Suíte*®, o *PAS Plant State*® ou o *UReason* OASYS-AM®, possuem foco no processo de gerenciamento como um todo, dando ênfase à racionalização da topologia de alarmes, que foca muito mais nos aspectos semânticos. Não há evidências do uso, por estes softwares, de técnicas para descoberta dos padrões além da análise de correlação e análise de causa raiz, que é feita baseando-se novamente em informações sobre o processo.

Em resumo, os algoritmos já sugeridos para resolver problemas semelhantes baseiam-se, fundamentalmente, em mineração de sequências, cuja adoção enfrenta as seguintes dificuldades:

- Os resultados são altamente dependentes dos parâmetros de suporte e tamanho da janela de tempo (*sliding window*);
- Melhorias dos resultados são, em geral, obtidas apenas através da variação dos parâmetros de entrada e recálculo da saída, o que pode ser custoso;
- A maior parte dos trabalhos com mineração de sequências não foi aplicada diretamente ao contexto de mineração de alarmes industriais e, portanto, as particularidades sobre a forma como os padrões ocorrem não foram consideradas;
- A saída de um algoritmo de mineração de sequências tradicional (vide maiores explicações sobre funcionamento do algoritmo na seção 2.3) não apresenta

indicação sobre “a força” das relações intra-sequências, ou seja, as sequências resultantes são apresentadas no formato ABCD com suporte *sup*, o que impede uma eventual discriminação pelo interpretador especialista das diferenças na intensidade A-B e B-C, por exemplo;

- A sobreposição de ocorrências de sequências alarmes, ainda que eventual, pode levar a falsos positivos [12].

2.2.2.1 Análise da Solução *Cross-effect Test*

Em 2008, KORDIC et.al.[13] propuseram uma abordagem nova, específica para o contexto de detecção de padrões de alarmes industriais, que atenua o problema do parâmetro de tamanho da janela deslizante. A solução utiliza uma estratégia de segmentação de eventos e filtragem de dados baseada no que os autores chamaram de *cross-effect test*, que utiliza a duração de cada alarme para determinar tamanhos de janelas diferentes. Nesta abordagem, a mineração é executada em duas grandes etapas. Primeiramente, as sequências de alarmes são transformadas em um conjunto de transações com um algoritmo que utiliza duas janelas deslizantes: A-R *window*, que é a janela com o intervalo para a ativação do alarme (CFN) e R-A *window*, que é a janela com o intervalo para o retorno do alarme. Alarmes que comecem dentro de uma mesma janela A-R e terminem dentro de uma mesma janela R-A são considerados como pertencentes a um único *itemset* (grupo de itens), ou, formalmente falando, o critério de geração de uma transação poderia ser dado por *Ativações* (A-R) ∩ *Retornos* (R-A ou R-w). À medida que as janelas vão deslizando, é criada uma lista de transações ordenadas no tempo de alarmes que ocorreram de forma repetida ou não nas janelas A-R e R-A. No exemplo abaixo (Figura 6), da sequência s={(1,0),(2,10),(3,20),(-1,30),(-3,40),(-2,50),(1,60)}, em que o sinal negativo indica o término do alarme (ou RTN), a transação t={1,2,3} seria extraída a respeito da TAG 1, cujo tempo de duração determina o tamanho das janelas $W_{A\text{-}R}$ e $W_{R\text{-}A}$. Assim, as transações são formadas a cada *shift* da janela $W_{A\text{-}R\ (i)} + W_{R\text{-}A/R\text{-}w\ (i)}$, detectando os eventos de ativação e retorno únicos no intervalo tA – tR.

Na segunda etapa, esta lista de transações é utilizada como entrada para um algoritmo de mineração de regras de associação modificado, o *FP-Growth*, que extrai os padrões frequentes das transações. A modificação apresenta apenas os *itemsets* maximais (aqueles para os quais não existe superconjunto frequente) associados a algumas TAGs de interesse, já que os dados de entrada são gerados a partir de simulação.

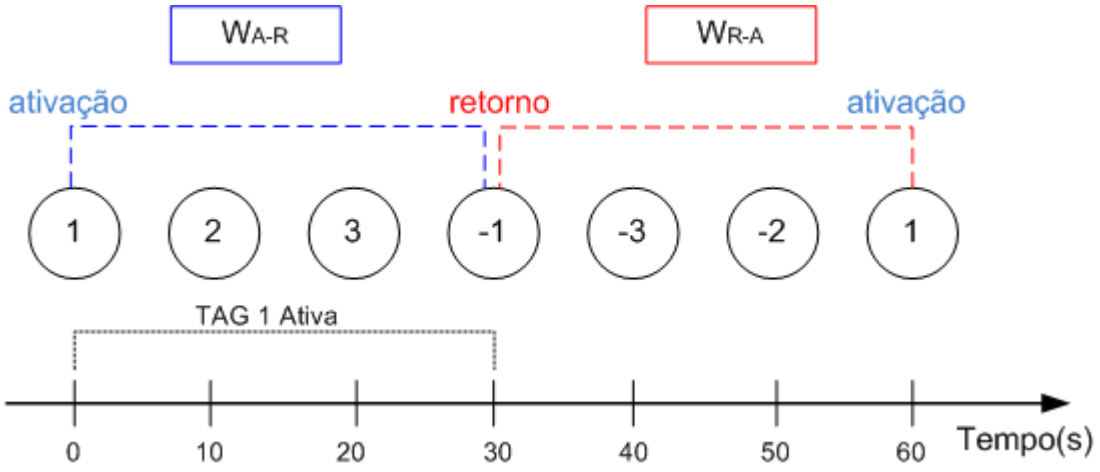

Figura 6 - Janelas W A-R e W R-A do Algoritmo Cross-effect Test de KORDIC et al. 2008 [13]

A abordagem de Kordic et. al. trouxe várias contribuições significativas para o processo, como a estratégia de geração das transações a partir da lista de alarmes, a determinação do tamanho da janela a partir da duração dos próprios alarmes e o uso de regras de associação, que acrescenta o valor de confiança para definir a força das relações. No entanto, as seguintes deficiências podem ser listadas:

- O algoritmo de geração da lista de transações considera apenas um tipo de padrão de falha, que é o padrão em que o alarme B inicia-se após o início de A e termina depois de A (chamado neste trabalho de “Y Overlaps X”). Há diversos outros tipos de padrões possíveis que não foram tratados;
- Apesar da eliminação da necessidade de estipulação do parâmetro “tamanho da janela”, um novo parâmetro foi inserido, *minw* que é um valor que limita o tamanho da janela R-A (descrito apenas como $W_{R\text{-}A}$ na figura). Os próprios autores sugerem que várias execuções sejam feitas com diferentes valores de *minw*, o que leva ao problema original;
- Em função dos tipos de medidas de interesse utilizadas, suporte e confiança, correlações perfeitas entre dois tipos de alarmes podem ser detectadas com um nível de significância inferior a 100%, apesar da possibilidade concreta de se apresentar o valor real;
- Os resultados de relações entre alarmes não são ordenados no tempo, ou seja, não é possível diferenciar em um padrão AB se A ou B ocorreu antes no tempo, em função do algoritmo de descoberta de regras de associação utilizado;
- As regras de associação geradas são calculadas manualmente e de maneira pouco abrangente, levantando apenas as regras de implicação com antecedentes unitários;

- Nenhuma importância é dada à visualização dos resultados em termos de regras mineradas.

Estes aspectos serão abordados em detalhes nas seções 2.3 e 3.4.3.

2.2.3 Identificação de Padrões Através de Modelagem de Grafos

Poucos esforços até hoje foram direcionados às abordagens do tema através de grafos ou redes complexas. Apenas recentemente, talvez em razão da evolução das pesquisas na área motivadas pela evolução do hardware e crescimento dos volumes de dados, alguns autores sugeriram a modelagem de problemas similares desta forma. Em [14] Faloutsos et al. utilizaram o conceito de "cascatas de informação", originalmente proposto por Bikhchantani et al. em [15], para modelar a detecção precoce de padrões de trocas de informações entre pessoas. Nestas redes os eventos são modelados como nós e as relações entre os eventos indicam o fluxo temporal de informação. No trabalho intitulado "*Cost-effective Outbreak Detection in Networks*", os autores focaram na redução do tempo de detecção do efeito cascata para reduzir a população afetada pelo problema através de um algoritmo aproximado chamado CELF, que reduz o tempo de detecção em relação às abordagens com algoritmos gulosos [16] através da exploração de técnicas utilizadas em redes sociais. O estudo, entretanto, parte do princípio de que o efeito cascata é uma constante e a tarefa do algoritmo é, portanto, dada uma rede de distribuição de *commodities* (como a rede de distribuição de água, no caso em questão), determinar estrategicamente, utilizando o menor número de sensores posicionados, como detectar a falha antes que a mesma se propague por toda a rede. Apesar das similaridades, esta abordagem é inviável no caso da detecção de cascatas de alarmes em plantas industriais, já que as situações de cascatas não são conhecidas a priori, assim como ocorre neste caso.

Em [17], no artigo "*Cascading Behavior in Large Blog Graphs*", os autores propõem uma forma de estudar a propagação de informação nas redes através da detecção das sub-estruturas que se repetem, quando o problema de troca de mensagens é modelado na forma de um grafo. Eles utilizam algoritmos que calculam o isomorfismo destas sub-estruturas para detectar formas frequentes no comportamento social e elaboram conclusões sobre os padrões topológicos detectados. A vantagem que torna a técnica viável, é que, novamente, é possível detectar os nós de origem através das características do problema em si (no caso em particular citado no artigo, sobre blogs que fazem referências entre si, o

blog que possui grau de saída igual a zero é o nó origem). Esta característica não é presente no caso dos *logs* de alarmes de sistemas de controle industriais, sendo a questão da detecção da origem um dos problemas a resolver. Além disso, é sabido que o problema de isomorfismo de grafos é conhecido na computação como pertencente à classe NP-completo[16], havendo apenas soluções aproximadas disponíveis como heurísticas ou abordagens probabilísticas.

Em [10] DUFFIN et al., introduziram o conceito de Proximidade Topográfica (TP), uma medida útil na correlação de eventos de alarmes que utiliza informações topográficas codificadas nos alarmes para validar as sequências candidatas em tempo de execução. Apesar de contribuir em relação à automatização da tarefa, a técnica exige o entendimento pleno dos atributos dos alarmes a priori por algum “expert” no assunto, o que pode ser tão custoso a ponto tornar a atividade inviável. A ideia é solicitar a este profissional experiente para que identifique os nós e os tipos de relações que podem haver entre estes nós para que se crie uma informação topográfica disponível durante a mineração. O processo de mineração em si utiliza a técnica de mineração de sequências com janela móvel tradicional mas, ao invés de utilizar o limiar de frequência para a seleção dos padrões, o algoritmo (MINEPI otimizado), utiliza a informação de conectividade entre os nós e similaridade com os padrões topográficos conhecidos. Novamente, esta é uma solução que ainda não oferece resposta aos problemas existentes, como o problema dos parâmetros fortemente conectados à qualidade dos resultados e o alto grau exigido de conhecimento do processo e das possíveis relações entre alarmes.

KLEINBERG et al., em [18], sugerem o estudo da dinâmica temporal de eventos de comunicação através do uso de técnicas de modelagem através de Redes Complexas. O trabalho faz uso de vetores de tempo (*vector clocks*), antes sugeridos por LAMPORT [19], para modelar “*lags*” temporais em uma rede, considerando a frequência e a atualização da comunicação entre os nós. No estudo os autores sugerem o conceito de “estruturas de *backbone*”, que compreendem nós fundamentais de uma rede complexa em relação a um dado período no tempo, e sugerem um algoritmo para o cálculo do “*backbone* agregado”, ou seja, um grafo único que resume a espinha dorsal desta rede durante todo o tempo, considerando a sua dinâmica de inter-relação entre os nós. A solução não se aplica diretamente ao problema da detecção de padrões de alarmes, mas, dentre as vantagens do trabalho de Kleinberg et. al., destacam-se a grande quantidade de possibilidades em termos de métricas que a teoria de redes complexas oferece e a evolução que o formato de representação visual traz à atividade de interpretação semântica.

Apesar da inaplicabilidade direta de nenhuma das abordagens sobre grafos ou redes complexas, as soluções neste campo trazem novas possibilidades para o tratamento do problema da descoberta de padrões de alarmes redundantes. Dentre os benefícios presentes, os seguintes podem ser destacados:

- A modelagem da ordenação temporal das sequências em conjunto com a força da relação para cada par, no trabalho de Kleinberg et. al., cria a possibilidade de geração de novas conclusões sobre os padrões mais relevantes, o que foi lá chamado de espinha dorsal;
- Há novas conclusões que podem ser extraídas com a modelagem dos resultados sobre grafos ou redes complexas, utilizando um novo conjunto de métricas e técnicas de análise antes inexplorada pelas técnicas tradicionais de mineração de dados;
- A representação visual dos resultados em termos de padrões significativos traz uma série de facilidades para os usuários que fazem a interpretação dos resultados.

2.2.4 O Desafio da Visualização na Interpretação dos Resultados

A importância dos mecanismos para entendimento e interpretação de padrões frequentes minerados tem sido reconhecida, recentemente, como uma área de pesquisa a ser mais explorada. Historicamente [10], o foco dos trabalhos de pesquisa da mineração de dados tem se concentrado na composição dos padrões e tratamento da frequência de ocorrência, embora se saiba que a semântica de um padrão frequente inclua extensas informações, dentre elas: o significado do padrão, os padrões sinônimos e os tipos mais comuns de transações nas quais o padrão reside. Uma subsequência válida, por exemplo, extraída como resultado em uma atividade de mineração de sequências da base de dados <(7,3,9,4,5,6,8),(7,9,6), (7,9)> poderia ser <(7),(9)>, com suporte mínimo (*minsup*) igual a 3 (a quantidade de ocorrências que suporta esta sequência). Considerando que esta subsequência, na forma como apresentada, poderia ser o resultado da mineração, as seguintes restrições poderiam ser observadas em relação à sua usabilidade:

- A forma de modelagem e apresentação dos resultados dificulta a visualização de ramificações das sequências. No caso da mineração de alarmes redundantes, isto é particularmente importante, já que as decisões de supressão são determinadas para um tipo em especial de alarme considerando todas as relações possíveis deste alarme;

- É incomum, para a maior parte dos algoritmos de mineração de sequências, definirmos múltiplos valores de suporte para sequências internas, ou seja, mesmo que haja subsequências de alarmes significativas dentro de uma sequência mãe, tal subsequência não seria detectada caso o valor de limiar de suporte definido para toda a sequência não fosse atendido também para a subsequência. Para a maior parte dos algoritmos conhecidos, os “cortes” de subsequências que não são frequentes são feitos em tempo de mineração e, como um número muito grande de combinações é tipicamente executado, para a melhoria do desempenho, os valores abaixo do limiar de suporte são imediatamente desprezados. Se houver necessidade de se compreender a influência de uma subsequência de suporte inferior ao limiar, o algoritmo deve ser executado novamente. Considerando o exemplo acima, uma subsequência válida e interessante é <(7),(9),(6)> com suporte igual a 2, mas como o valor mínimo de suporte definido é 3, esta informação não é apresentada;
- Não há marcação ou registro a respeito da “intensidade” da frequência dentro das sequências. Assim, subsequências com suporte muito superior ao valor de limiar são tratadas, na apresentação dos resultados finais, da mesma forma com que são tratadas todas as demais subsequências com valor maior ou igual ao limiar de suporte. No exemplo acima, se imaginássemos o valor de suporte igual a 3, os dois resultados <(7),(9)> sup=3 e <(7),(9),(6)> sup=3 deveriam ser lidos para construir mentalmente a informação de que há uma relação na sequência <(7),(9),(6)> que é mais intensa na subsequência <(7),(9)>. Entretanto, fazer esta construção mental para muito resultados seria inviável.

Em quaisquer das técnicas de racionalização, o resultado final sempre é dependente de uma validação semântica dos padrões encontrados para a eliminação ou tratamento da redundância. Assim, a forma utilizada para apresentar os padrões encontrados e o grau de profundidade no conhecimento que esta forma permite, tem relação direta com o nível de aproveitamento destas informações e, consequentemente, com a eficiência do processo de racionalização. Em todas as observações de usabilidade mencionadas acima, há grande influência da forma como, visualmente, os resultados são apresentados.

Uma contribuição adicional em relação a tudo o que já foi proposto na área seria encontrar uma forma atrativa, em termos de visualização, para modelar esta rede de padrões de alarmes minerados, tal que a tarefa de interpretação das sequências e tendências identificadas pudesse simplificar o processo e reduzir o tempo necessário para a

atividade de interpretação. Este problema de representação de redes muito grandes é tema de pesquisa dentro da área de estudo das redes complexas. Em [20], por exemplo, Leskovec et al. propõem, no trabalho intitulado "*Planetary-Scale Views on a Large Instant-Messaging Network*", o desenho de arestas mais ou menos espessas (Figura 7) como forma de representar a força das relações entre nós de uma rede. Esta poderia ser uma maneira interessante e rica para representar possíveis relações entre diferentes alarmes, diferenciando a intensidade de cada relação através da espessura da aresta.

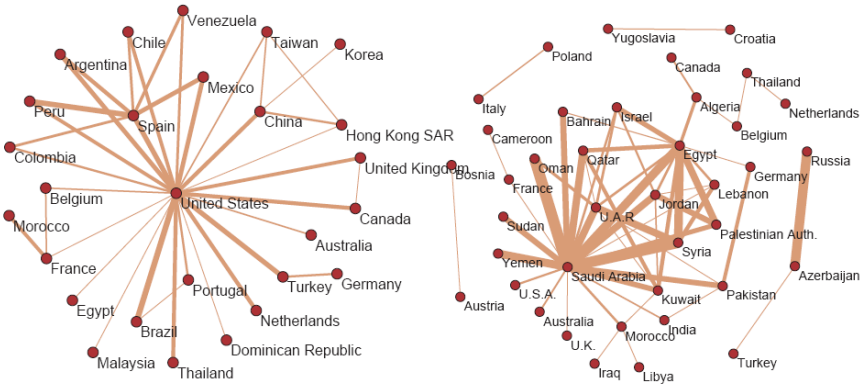

Figura 7 - Sugestão de Representação da Força da Conexão Entre Nós de Uma Rede Proposta por Leskovek et al.

2.3 Formalização de Conceitos e Definição do Problema

Esta seção apresenta os principais conceitos formais envolvidos no problema e solução de mineração de padrões de alarmes redundantes. Demais formalizações serão apresentadas quando se fizerem necessárias em seções posteriores.

Alarme/Registro de Alarme: dada uma classe de diferentes tipos de TAGs de alarmes *T*, uma ocorrência ou registro de um alarme pode ser dito como sendo a trinca (a, c, r), onde $a \in T$, c é um inteiro que representa o instante no tempo em que o alarme foi ativado (CFN - change from normal) e r é um inteiro que representa o instante no tempo em que o alarme foi desativado (RTN – return to normal), sendo $c > r$ (já que o instante de retorno necessariamente deve ser superior ao instante de início);

Base de Dados de Alarmes / Lista de Registros de Alarmes: uma base de dados de alarmes *D* é uma coleção de registros de alarmes ordenada por c ou r, tal que *D* = $\{(a_1, c_1, r_1), (a_2, c_2, r_2), \ldots (a_n, c_n, r_n)\}$ para todo i=1, 2, 3, ..., n, $a_i \in T$ e $c_{i+1} \geq c_i$, para *D* ordenado por c e $r_{i+1} \geq r_i$, para *D* ordenado por r;

Transação: na mineração de sequências, uma transação consiste em uma trinca (*Transaction-ID; Transaction-Time; Transaction-Items*). Um exemplo de trinca poderia ser dado por (100; 2010-12-01 12:35; 1,2,3) No caso dos alarmes, uma transação é um conjunto de TAGs de alarmes que ocorreram dentro de uma mesma janela de tempo;

Itemset: na mineração de sequências [8], um itemset é um conjunto não vazio de items, onde cada item I é uma variável binária que significa se o item está presente ou não. Cada variável é representada por um número inteiro, mapeado a partir dos diferentes tipos de itens. Desta forma, um itemset i é definido como $(i_1, i_2, i_3, \ldots, i_m)$, onde i_j é um item. No caso dos registros de alarme, cada itemset contém apenas um item, que é o código do alarme em questão;

Sequência: segundo [8] uma sequência é uma lista ordenada de itemsets. Uma sequência $(a_1, a_2 \ldots a_n)$ está contida em outra sequência $(b_1, b_2 \ldots b_n)$ se existem inteiros $i_1 < i_2 < i_3$ tal que $a_1 \subseteq bi_1$; $a_2 \subseteq bi_2$; ... $a_n \subseteq bi_n$. Os conceitos de sequência e suporte se aplicam igualmente ao contexto de alarmes;

Suporte: o suporte *sup* de uma sequência é definido como sendo a fração total de transações (registros de alarmes) em *D* que suporta (que contém) esta sequência. O

suporte pode ser dado de maneira absoluta (número de transações) ou relativa (percentual de transações). O suporte mínimo *minsup* é o valor de suporte *sup* a partir do qual as sequências são consideradas frequentes;

Mineração de Sequências: o problema de mineração de sequências consiste em encontrar, em uma base de dados *D,* as sequências maximais que possuem determinado suporte determinado pelo usuário [8]. Note que uma consequência importante da mineração de sequências é a preservação da ordem em que os itens aparecem nos itemsets e transações;

Regra de Associação: segundo [21] uma regra de associação é uma implicação na forma $X \Rightarrow Ij$, onde X é um conjunto com um ou mais items de *T* e Ij é um único elemento que não está presente em X. A regra $X \Rightarrow Ij$ é satisfeita em *D* com um fator de confiança $0 \leq conf \leq 1$ se, e somente se, pelo menos *conf*% das transações em *D* que satisfazem X também satisfaçam Ij. X é chamado antecedente e Ij é chamado de consequente na regra resultante. Uma observação importante é que, na definição original, a ordem (sequência) dos itens no itemset X não é relevante. Para os algoritmos de mineração de regras de associação, portanto, <a b $\Rightarrow$ c> pode ter o mesmo significado de <b a $\Rightarrow$ c>;

Confiança: segundo [10] uma regra de associação $X \Rightarrow Y$ é definida como a probabilidade condicional de que um objeto inclui Y, dado que inclui X com uma confiança $conf(r) = sup(X \cup Y)/sup(X)$. A confiança da regra, portanto é dada pelo percentual de transações que possuem X e Y dividido pelo percentual de transações que possuem apenas X. Diferentemente do suporte, que indica a significância estatística, a confiança é uma medida de interesse que denota a força da regra [21];

Definição do problema: seja c é um inteiro que representa o instante no tempo em que o alarme foi ativado (CFN - change from normal) e r é um inteiro que representa o instante no tempo em que o alarme foi desativado (RTN – return to normal). Dois alarmes a_1 e $a_2 \in T$, são considerados alarmes redundantes se existir uma relação temporal de associação entre a_1 e a_2 do tipo $a_1 \rightarrow a_2$ ou $a_2 \rightarrow a_1$ relativa exclusivamente a c, exclusivamente a r, ou relativa a c e r simultaneamente que indique que, em termos de processo industrial, os sinais que provocaram a_1 e a_2 possam ser submetidos a uma expressão lógica que resumiria a_1 e a_2 em um único alarme $a_{1,2}$. Deve-se observar, entretanto, que a semântica da relação $a_1 \rightarrow a_2$ foi estendida e significa que a_1 é antecedente,

a_2 é consequente (a1⇒a2) e a_1 precede a_2 no tempo em relação a c e/ou a r. Em resumo, a intenção é encontrar as regras de associação entre os diferentes tipos de alarmes, preservando nos resultados a ordem de precedência para que esta informação seja apresentada ao usuário. Deve-se notar que, diferentemente dos dados de entrada da mineração de regras de associação, os registros de alarmes não estão ordenados na forma de transações com itens, de forma que uma combinação dos algoritmos de mineração de sequências e de mineração de regras de associação é esperada.

2.3.1 Tipos de Padrões Temporais

A pressuposição de KORDIC et al. em [13] de que dois alarmes somente podem estar relacionados com uma relação do tipo *overlapping* é simplista, dado que o universo de possibilidades é desconhecido a priori. Em [22], Allen e Ferguson, ao estudarem a complexidade das relações e interações entre ações e eventos, apresentaram axiomas avaliando a lógica dos intervalos temporais e a estrutura do tempo e levantaram sete (7) possíveis relações temporais (seis tipos mais a relação *Equals*) entre dois possíveis períodos de tempo (Figura 8).

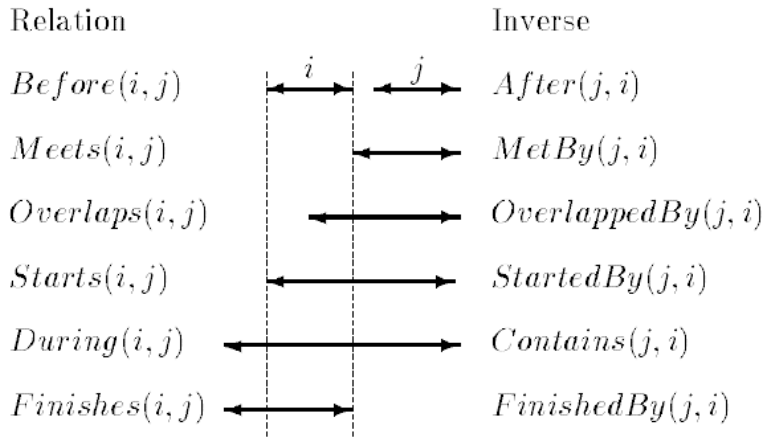

Figura 8 – Possíveis Relações Entre Períodos de Tempo (ALLEN e FERGUSON, 1994)

As relações temporais de Allen, como são referenciadas, se aplicam a diversos contextos e foram utilizadas não somente no algoritmo de geração de transações aqui proposto, mas na construção de um gerador sintético de carga capaz de simular as relações para testar a sua capacidade de detecção em termos de exatidão. Com exceção da relação

do tipo “*Before*”, que normalmente representa situações típicas de ocorrências de alarmes sucessivos no processo não redundantes, todas as demais relações foram geradas. Para exemplificar que tais situações são possíveis na prática, algumas configurações de processos industriais de diferentes naturezas são citadas abaixo envolvendo tais tipos de relação (estas relações foram levantadas por um engenheiro de processo experiente). A relação “*Before*” foi substituída pela relação “Equals”, que não é apresentada no desenho, mas que é citada dentre as relações no trabalho de Allen e Ferguson.

XEqualsY: este caso é comum nos sistemas de diagnóstico, que são alarmados em paralelo aos sistemas que eles diagnosticam, gerando alarmes. As salas elétricas[4], por exemplo, têm sistema de detecção de incêndio e estes mesmos sistemas tem auto-diagnósticos projetados para avisar sobre falhas nas detecções feitas por eles. Assim, pode ocorrer um alarme falso X de aviso de incêndio em uma sala e no mesmo instante Xi o sistema de diagnóstico registrar que o sistema está com falha de funcionamento no instante Yi, tal que Yi=Xi. Depois que o sistema de detecção normaliza seu funcionamento, ele mesmo retorna o alarme à condição de normalidade e o sistema de diagnóstico também retorna avisando do retorno correto do sistema de detecção. Em resumo, ocorreu um falso alarme de incêndio e um alarme de defeito no sistema de incêndio. A sequência de ocorrências abaixo exemplifica este caso:

Exemplo:

1/8/2008	11:56:43.1 [CE] SE6971_03_04_AINC	CFN	ATUADO	SE6971.03 04 Incêndio na sala
1/8/2008	11:56:43.1 [CE] SE6971_03_04_ADINC	CFN	ATUADO	SE6971.03 04 Defeito no sist. incêndio
1/8/2008	11:56:43.1 [CE] USINA_AINC	CFN	ATUADO	Incêndio na usina - chamar brigada

XmeetsY: em um processo de extração e beneficiamento de minério de ferro há diversas correias transportadoras. Quando uma correia (TC) carregada, rodando normalmente, sofre parada por defeito de motor (ALARME X), no instante em que o motor volta a funcionar, pode ocorrer sobrecorrente, já que a correia está cheia de minério, o que gera um ALARME Y, exatamente no instante de retorno do alarme X.

XOverlapsY /YOverlappedByX: considere diagrama de exemplo da Figura 9, criado hipoteticamente para representar parte de um processo industrial composto por

[4] Salas elétricas são ambientes fechados onde os painéis elétricos e controladores lógicos são instalados. Em função da grande quantidade de equipamentos elétricos, estas salas costumam ter requisitos rigorosos em relação a sistemas de proteção e combate a incêndio.

uma válvula de abertura (FCV2), um tanque (E-1), um medidor de nível do tanque (LI), uma válvula de controle de saída (FCV1) e uma válvula de bypass (FV1). Quando a válvula de controle FCV1 fica entupida, ou o posicionador da válvula não responde ao comando (por mal funcionamento, por exemplo), o medidor de vazão na saída da válvula indica uma vazão baixa ou nula e esta indicação gera um alarme que vai para o operador (alarme Y no instante Yi). Com a continuação da abertura da válvula FCV2, o tanque se enche e o medidor de nível LI gera um segundo alarme de risco de transbordo do tanque (gerando o alarme X no instante Xi). O operador age fechando FCV2 (se possível) ou abrindo a válvula de bypass (FV1) para esvaziar o tanque. Se a ação motivada pelos alarmes for a de abrir a válvula de bypass, o alarme X retorna ao normal (RTN) no instante Yf, mas o nível ainda estará alto, fazendo com que o alarme retornando num instante Xf > Yi, gerando portanto a sequência Yi→Xi→Yf→Xf, que caracteriza a relação temporal *XOverlapsY*.

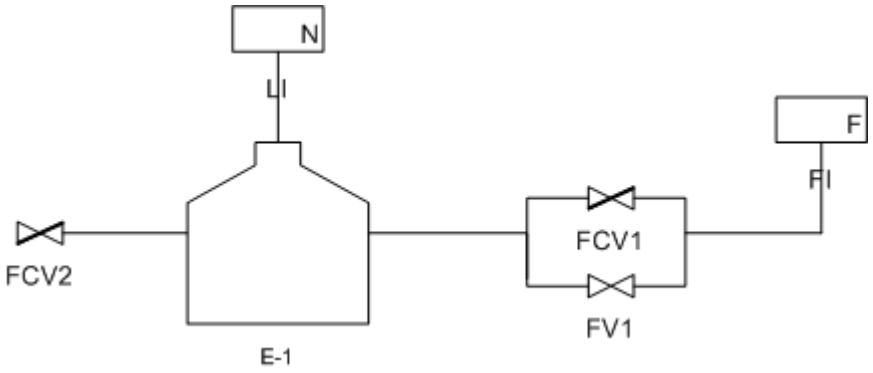

Figura 9 - Diagrama P&ID (*Process and Instrumentation Diagram*) Ilustrativo Hipotético

XstartsY: quando uma linha de processo (tubulação) é bloqueada por razões diversas como entupimento, fechamento de válvula de bloqueio ou aumento de densidade do fluido, as variáveis de monitoramento da pressão e vazão geram alarmes instantaneamente. São gerados alarmes de pressão muito alta (MA / HH) e vazão baixa na linha. Entretanto, quando a causa da parada do fluxo é resolvida por ordem da sala de controle, os alarmes de pressão alta na linha retornam a normalidade, mas os de vazão baixa demoram mais tempo para normalizar em razão da resposta dinâmica mais rápida da variável de pressão em relação à de vazão.

XduringY: nos sistemas de alarme e combate a incêndio, é comum a instalação de instrumentação para detecção de fumaça e detecção de fogo. Ambos podem gerar

alarmes. O que ocorre é que, em caso de incêndio, o detector de fumaça é ativado gerando um alarme Y no instante Yi. Em seguida a chama será detectada pelo sensor de temperatura (fogo) gerando um alarme X no instante Xi posterior a Yi. Quando o operador atua no incêndio por acionamento da brigada contra incêndio ou corpo de bombeiros ou através do sistema automático de combate a incêndio, o fogo é extinto primeiro causando a redução brusca da detecção de fogo e retorno do alarme X para RTN num instante Xf. Mas o detector de fumaça continua alarmado até a fumaça se dissipar num instante Yf levando o alarme ao RTN, o que dependendo da extensão pode durar vários minutos.

XfinishesY: em um processo de mineração, no espessador de minério, quando a densidade do minério fica muito alta, isto gera um alarme (ALARME X). Ao permanecer alta, o REIKI (Agitador) vai parar devido à alta carga de torque e isso levará a sobrecorrente do motor, gerando um alarme (ALARME Y). Caso o problema seja corrigido sob ordem da sala de controle, os dois alarmes devem desaparecer simultaneamente, já que, quando a densidade for reduzida, o motor voltará a operar normalmente.

2.3.2 Seleção da Medida de Interesse para a Força das Relações

Suporte e confiança são medidas de interesse tipicamente utilizadas na mineração de sequências e de regras de associação. Entretanto, em todas as soluções já propostas que fazem uso destas medidas, o uso da confiança isoladamente, não é suficiente para medir a força das relações quando há correlação perfeita ou deslocada no tempo com a mesma eficiência com que o faria o algoritmo de correlação cruzada, por exemplo. Em [23], Ning et.al. fazem um estudo sobre a seleção da medida de interesse adequada em processos de descoberta de padrões e geram duas conclusões importantes a) não existe medida consistentemente melhor do que todas as outras em todos os casos e b) a seleção da melhor medida deve ser feita confrontando as características das medidas possíveis com as expectativas e conhecimento dos experts na área de interesse.

A interpretação destas conclusões discutidas acima e a resposta em relação ao problema de mineração de alarmes redundantes é que ambas as medidas de confiança/suporte e coeficiente de correlação devem ser apresentadas, confiando ao usuário com conhecimento semântico do processo a decisão sobre quando utilizar cada medida. Isso facilita também o processo de interpretação dos resultados, já que informações

complementares podem surgir. Pegue-se o exemplo em que dois alarmes A e B possuem um coeficiente de correlação de 0.5 e uma relação A→B com confiança de 0.99 em relação a CFN. Avaliando estes dois resultados em conjunto, pode-se concluir, com uma probabilidade elevada, que A e B começam juntos (em função da relação A→B com confiança de 0.99) e um dos alarmes possui uma duração que corresponde à metade da duração do outro alarme (em função do coeficiente de correlação de 0.5). Caso a relação identificada fosse A→B com confiança de 0.99 em relação a RTN, a conclusão seria que A e B terminam juntos (XFinishesY).

2.3.3 Aspectos de Desempenho

Uma das sugestões para o processo de mineração de alarmes que será apresentada no próximo capítulo é, ao invés de executar consecutivamente o algoritmo com vários valores de suporte, reduzir o valor de *minSup* (vide conceito na introdução da seção 2.3) a um valor mínimo, deixando a atividade de corte dos padrões irrelevantes para uma etapa posterior, na qual visualmente isto seria feito com maior facilidade após a representação através de uma rede complexa. Esta decisão, combinada com a necessidade de uso de pelo menos dois algoritmos diferentes, leva a crer que problemas de desempenho e escalabilidade do algoritmo poderiam surgir.

Os aspectos de performance e escalabilidade, de fato, têm sido insistentemente tratados na área de pesquisa desde a proposição dos primeiros algoritmos de mineração de dados. No entanto, há razões para crer que a qualidade dos resultados e a capacidade de síntese dos algoritmos desempenham um papel mais significativo, quando se trata de conduzir o usuário aos padrões mais interessantes. Em [10], HAN et al. mencionam, de fato, que o gargalo atual das metodologias de mineração não é a capacidade de gerar um conjunto **completo** de padrões frequentes, mas sim um conjunto **compacto de alta qualidade e útil nas aplicações**.

Considerando a evolução do hardware (especialmente a evolução da capacidade de memória RAM e disco) e os casos práticos de uso das técnicas de mineração em projetos reais, podemos afirmar que há uma preocupação exagerada em muitos casos quanto a evitar o aumento do tempo de execução em detrimento de espaço em disco. Em um dos primeiros algoritmos criados para a mineração de regras de associação, por exemplo, AGRAWAL e SRIKANT [24] apresentam resultados em que

nenhuma das execuções para uma base de dados de 100 mil registros, o tempo de execução é superior a 700 segundos[5], valor absolutamente aceitável em um caso prático para a detecção de padrões de alarmes.

Uma das características do processo de racionalização de alarmes é que, independentemente da técnica utilizada, sempre é necessária a presença de uma pessoa com conhecimento profundo sobre os processos industrias relativos ao sistema de supervisão e controle (SCADA) para que os padrões possam ser interpretados e racionalizados com segurança. Frequentemente, esta interpretação semântica depende não de um, mas de diversos profissionais simultaneamente, o que torna esta atividade mais cara e complicada. Neste contexto, mesmo o custo elevado de tempo da análise de correlação cruzada não é suficiente para inviabilizar a atividade, já que a racionalização não é uma atividade crítica que exija características de resposta em curto prazo ou em tempo real.

Neste trabalho, com uma base de dados de 250 mil registros e 243 tipos diferentes de alarmes, cerca de vinte horas de processamento foram necessárias em um Intel Pentium IV Core Duo 1.8GHz 2GB de RAM para gerar os padrões com todos os algoritmos combinados, mas apenas alguns segundos diante do expert foram suficientes[6] para eliminar os padrões menos relevantes na apresentação durante os testes com a base de dados real.

[5] Neste caso os autores utilizaram um equipamento com CPU de 33MHz e 64MB de RAM em uma comparação de desempenho de algoritmos que discursa sobre o impacto que a quantidade de memória gera de limitação na execução dos algoritmos.

[6] Considerando apenas o tempo necessário pelo algoritmo para remover os padrões abaixo de determinado valor de corte estipulado pelo usuário.

3 METODOLOGIA

Este capítulo apresenta os métodos utilizados que conduziram aos resultados em termos de melhoria na exatidão da descoberta de padrões de alarmes industriais redundantes. O algoritmo construído foi aplicado sobre dados de entrada reais e dados sintéticos, gerados por um gerador de carga construído a partir de um processo de caracterização[7], que será descrito a seguir nos tópicos sobre Dados de Entrada. A solução em si é descrita na seção 3.4.

3.1 Dados de Entrada Reais

Os arquivos de *logs* analisados foram coletados de uma planta industrial de mineração real e correspondem a aproximadamente três meses (90 dias) de operação, agrupados em arquivos de texto (um arquivo por dia). Em cada arquivo os seguintes atributos estão disponíveis:

- **TAG** (Código do alarme): trata-se de um *String* que identifica cada alarme dentro do sistema de supervisão e controle da planta. A cada tipo de alarme está associado também um nível que indica a criticidade do alarme, que pode ser classificado, por exemplo, em Muito Alto (“HH”, de *High High*), Alto (“H”, de *High*), Muito Baixo (“LL” – *Low Low*) e assim por diante. São exemplos de códigos: CA6490_ALAH, CO6805_APAL e CM6981_AFCC. A informação de criticidade não foi explorada neste trabalho em função da limitação de escopo, sendo o seu uso sugerido apenas para trabalhos futuros;
- **Descrição**: é a descrição correspondente ao código do alarme. Exemplo: CCM-6981 Falta de corrente continua. A descrição foi utilizada apenas durante a atividade de interpretação dos resultados (análise semântica);
- **CFN Date**: é a data/hora que indica que houve uma alteração no valor da variável (TAG) para fora de sua faixa normal de operação (CFN = *change from normal*). Para

[7] Os conceitos de Gerador de Carga e Caracterização de carga são descritos em detalhes por JAIM em [25].

permitir a comparação, as datas foram convertidas para o formato *timestamp* (*Unix time*), que permite uma precisão no nível de segundos;

- **RTN Date**: é a data/hora em que houve o retorno para a situação normal de operação, também convertida para o formato *timestamp*;
- **Hora de Reconhecimento**: a data e hora (incluindo minutos e segundos) em que o alarme foi reconhecido por um operador da planta. Este dado não foi explorado neste trabalho em função da limitação de escopo, sendo o seu uso sugerido apenas para trabalhos futuros;
- **Operador**: o nome ou código do usuário operador que foi responsável pelo reconhecimento do alarme. Este dado não foi explorado neste trabalho em função da limitação de escopo, sendo o seu uso sugerido apenas para trabalhos futuros.

Antes da execução do algoritmo propriamente dito, esta base de dados de registros de alarmes passou por uma etapa de limpeza prévia em que as seguintes atividades foram executadas:

- Os registros inconsistentes foram removidos[8]: registros com algum dos campos inválidos;
- Os registros decorrentes de falhas de comunicação foram removidos: quando há uma falha de comunicação, todos os tipos de alarmes possíveis são acionados em sequência de forma automática, como parte do processo de inicialização do sistema. Os registros deste tipo foram identificados e removidos, uma vez que não representam alarmes a serem considerados no processo de racionalização (não representam o acionamento do alarme de fato);
- Os registros decorrentes de início ou parada do sistema foram removidos: da mesma forma quando ocorre uma falha de comunicação, quando há reinício do sistema após uma parada, todos os tipos de alarmes são acionados. Os registros deste tipo foram identificados e removidos;
- Os campos de data e hora foram transformados em campos no formato "*Timestamp*" ou "*Unix Time*", em que a data e hora são representados na forma de um número inteiro que representa a quantidade de segundos desde 00:00:01 de 01 de Janeiro

[8] A remoção dos registros inconsistentes não pode ser considerada uma distorção da base de dados já que sua origem, em geral, está associada a erros de execução do software ou erros primários na programação cuja identificação é trivial.

de 1970, para facilitar a atuação do algoritmo na atividade de comparação e deslize da janela de tempo;

- Os demais campos não utilizados foram removidos;
- Os TAGs foram convertidos em números inteiros para facilitar a execução dos algoritmos de mineração. Uma tabela de equivalência foi criada para permitir a interpretação posterior.

Os arquivos foram agrupados em um arquivo único para facilitar a execução do algoritmo, sendo que o arquivo analisado continha 1.107.650 linhas de registros de alarmes.

3.2 Dados de Entrada Sintéticos

Para que os resultados da proposta de solução aqui apresentada possam ser validados e comparados em termos de exatidão com uma solução anterior, a base de dados real não pode ser utilizada diretamente, já que os padrões lá existentes não são conhecidos a priori. Por esta razão, um gerador de cargas sintéticas foi construído especificamente para geração de uma base de dados com características próximas da base de dados real, contendo padrões conhecidos e especialmente criados para testar toda a capacidade de detecção. Para criar esta similaridade de características, foi executada uma atividade de caracterização de carga, descrita a seguir, antes da apresentação do gerador propriamente dito. A construção do gerador e a inserção dos padrões artificiais também são descritas a seguir.

3.2.1 Caracterização de Carga

Segundo Jaim [25], o processo de caracterização de carga consiste em caracterizar um usuário típico ou um componente da carga (*workload component* ou *workload unit*). No contexto deste trabalho de mineração de alarmes, um componente da carga pode ser visto como a ocorrência de um alarme e os parâmetros da carga (*workload parameters*) podem ser vistos como as características principais dos *logs* de alarmes que causam impacto nos algoritmos de mineração.

3.2.1.1 Parâmetros da Carga

O objeto de caracterização consiste em um conjunto de ocorrências de alarmes de uma base de dados de alarmes de uma planta real de mineração, organizados na forma de uma série de arquivos de logs emitidos mensalmente por um sistema de Supervisão e Controle (SCADA) entre os meses de Novembro de 2008 e Fevereiro de 2009. Os seguintes atributos dos arquivos foram considerados inicialmente: TAG, CFN Date e RTN Date.

Pode-se afirmar que o conjunto ideal de parâmetros de carga a ser caracterizados consiste no conjunto mínimo tal que as principais características dos algoritmos de análise sejam afetadas. Podemos afirmar, por exemplo, que o intervalo entre ocorrências de alarmes é um parâmetro interessante, já que o principal parâmetro do algoritmo de mineração de sequências, o tamanho da janela deslizante, é fortemente afetado pela forma como os intervalos entre ocorrências se dão. Após a realização desta análise, os seguintes elementos foram definidos como parâmetros para a caracterização:

- Duração: consiste no tempo decorrido entre o início (CFN) e o término do alarme (RTN);
- Intervalos entre Inícios: consiste no tempo decorrido entre a hora de ocorrência de um alarme (CFN) e a hora de ocorrência do próximo alarme (CFN);
- Quantidade: consiste na quantidade de ocorrência para cada tipo de alarme no intervalo de tempo avaliado.

O sistema de alarmes gerador dos logs pode ser considerado um sistema único, mas que atende a planta industrial nos seus vários subprocessos. É possível, portanto, que haja subsistemas cujos alarmes não possam estar relacionados e que, portanto, devem ser separados em classes de alarmes distintas. Uma das formas de identificar classes diferentes de alarmes é desenhar a função de massa de probabilidade (PMF) de um ou mais atributos e observar a tendência de concentração dos valores em torno de mais de um padrão. Na Figura 10(a), que descreve a PMF do parâmetro Duração dos Alarmes, é possível observar uma tendência de separação em duas classes. De fato, é possível classificar os logs de alarmes com tais durações como pertencentes a dois subsistemas, dentro da planta: Usina e Britagem. A Britagem é a parte do processo em que o minério bruto tem o seu tamanho reduzido através de mecanismos que operam por impacto, compressão e cisalhamento (normalmente um britador é um elemento giratório com mandíbulas). Depois da moagem, o minério segue para a classificação e concentração,

etapas executadas na Usina. É lá que ocorrem as demais etapas que geram os produtos finais. Tratam-se de processos industriais diferentes que compartilham um sistema de alarmes único, mas que se comportam de forma isolada. A Figura 10 (b) apresenta a mesma distribuição, porém com a distinção entre as duas classes identificadas. Como conclusão da exploração inicial e entendimento dos metadados, a base foi dividida em duas porções: alarmes provenientes do sistema de alarmes da Usina e alarmes provenientes da Britagem.

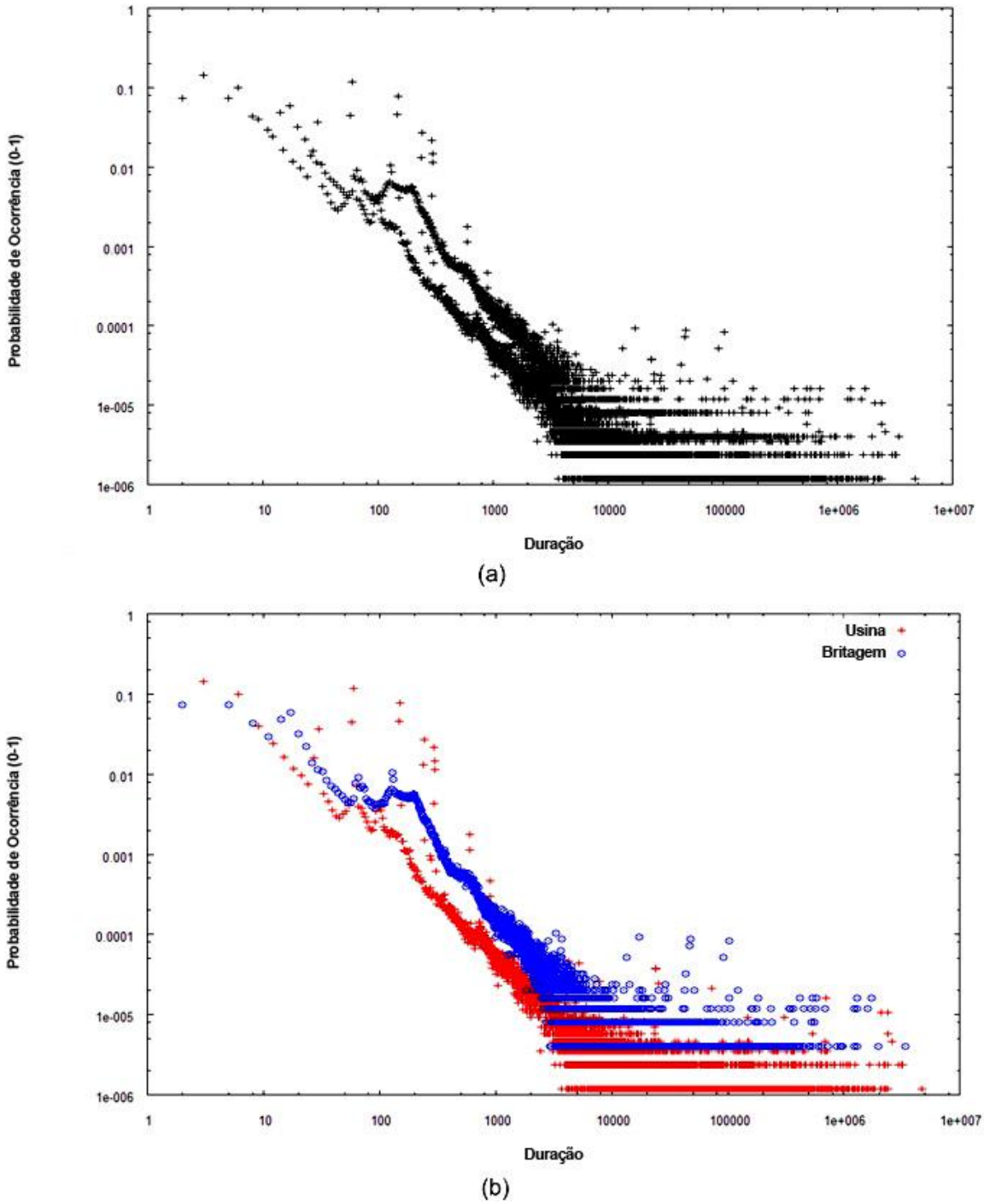

Figura 10 - Função de Massa de Probabilidade para Duração dos Alarmes Considerando a Base de Dados de Alarmes com uma Única Classe (a) ou com Duas Classes: Usina e Britagem (b)

Após a identificação e separação das classes de alarmes é importante buscar entender se há diferenças no comportamento das duas classes que justifiquem a criação de um gerador de cargas capaz de gerar as duas classes. Ao avaliar a Figura 10 é possível

concluir que, em termos de duração dos alarmes, ambas as distribuições podem ser aproximadas por funções similares com pequenas diferenças nos parâmetros. Para concluir, entretanto, sobre se podemos tratar apenas uma das classes na caracterização (e, portanto, no gerador), é necessário avaliar as demais métricas definidas.

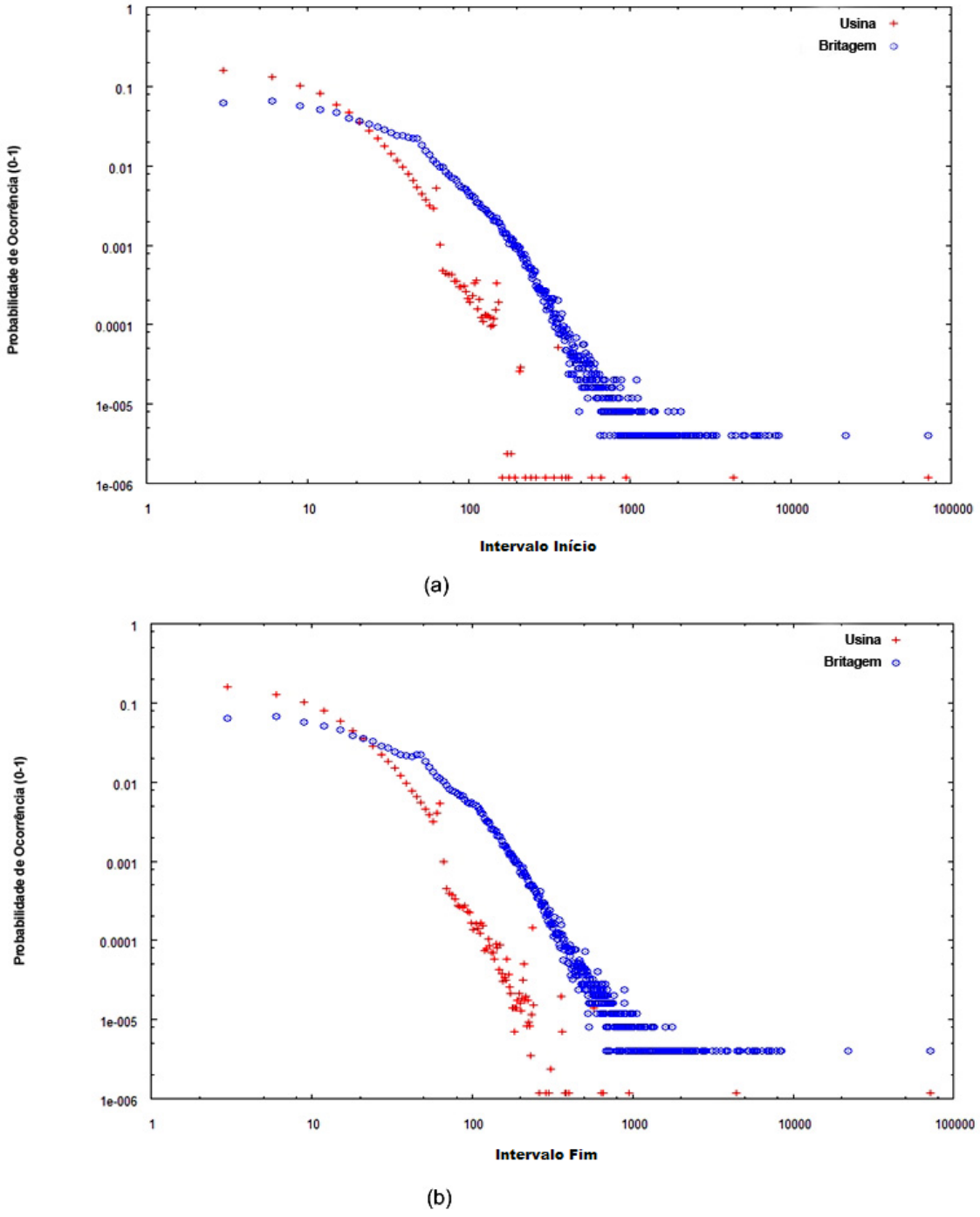

Figura 11 - Função de Massa de Probabilidade para Distribuições de Intervalos de Início (a) e Fim (b) para as Duas Classes de Componentes da Carga: Usina e Britagem

Na Figura 11 são apresentados os resultados da construção das funções de massa de probabilidade (PMF) para os intervalos entre início (CFN) e fim (RTN) para as duas classes de componentes de carga identificados. Novamente observa-se que há um comportamento similar no que tange à distribuição exponencial com cauda pesada, o que justificaria considerar apenas uma classe no gerador. O último parâmetro a ser avaliado para concluir sobre se isto é aceitável ou não é avaliar o parâmetro “quantidade”. Em termos práticos, é esperado que a quantidade implique em pouco impacto no comportamento dos algoritmos, com exceção do tempo de execução do algoritmo. Como as distribuições de probabilidade se aproximam, mesmo com a diferença de 250.875 ocorrências na Britagem para 856.775 na Usina, as quantidades da Britagem foram utilizadas.

Um último aspecto a avaliar é sobre a existência de correlação entre ocorrências de alarmes da Usina e da Britagem. Uma correlação significativa poderia exigir um tratamento diferente, com uma investigação mais detalhada sobre relações de causalidade antes de dispensar um dos componentes da carga. Desta forma, a análise de correlação foi executada.

3.2.1.2 Análise de Correlação Usina - Britagem

Como já foi mencionado, Britagem e Usina são considerados processos distintos cujos alarmes não se espera que possuam correlação. Poderíamos imaginar, entretanto, que, devido à característica do processo em que os insumos de uma área são providos por outra, situações atípicas pudessem afetar o número de alarmes, criando uma possível correlação no parâmetro “quantidade”. Esta possibilidade deve ser investigada, portanto.

O que caracteriza uma situação típica em uma planta é a estabilidade do sistema de supervisão e controle. Em uma situação de normalidade a planta opera com uma incidência menor de alarmes críticos e poucas ou nenhuma falha. Nas situações atípicas há perturbação ou anormalidade na operação que leva os sistemas de controle a gerar dezenas ou centenas de alarmes além do normal. Durante o período analisado a planta passou por períodos de situações típicas e atípicas, o que gera insumos para que uma análise de correlação confiável possa ser realizada (já que alguns dos possíveis alarmes só são gerados em situações atípicas, de perturbação).

O índice de correlação entre duas variáveis aleatórias pode ser facilmente calculado através do cálculo do Coeficiente de Correlação de Pearson r, já apresentado na seção 2.2.1, que fornece uma equação para o cálculo da correlação a partir de uma amostra normalmente distribuída. Os resultados são expressos através de um número entre -1 e +1 indicando o nível de correlação negativa ou positiva. Valores próximos a zero indicam um nível de correlação fraco ou inexistência de correlação linear. O valor de r obtido na análise foi igual a 0.02325679, indicando uma correlação praticamente desprezível. O diagrama de dispersão apresentado na Figura 12 ilustra e reforça esta conclusão.

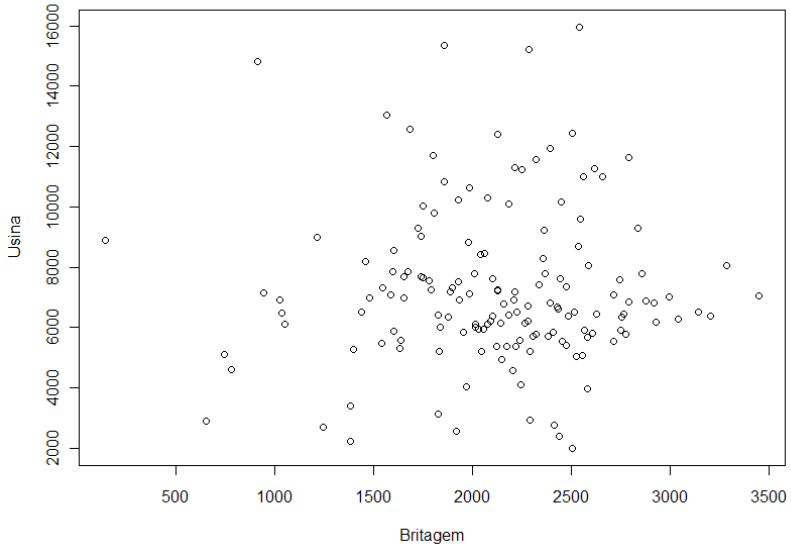

Figura 12 - Diagrama de Dispersão Quantidade de Alarmes/Dia Usina em Função da Quantidade de Alarmes/Dia Britagem

3.2.1.3 Escolha da Classe Representativa

A conclusão, após os resultados das análises apresentadas, é que a consideração de duas classes no gerador é dispensável, dada a similaridade nas distribuições de durações e intervalos de início/fim e independência das ocorrências de alarmes geradas na Usina e Britagem. Diferenças significativas entre Usina e Britagem para quaisquer destes atributos justificariam considerar duas classes de dados, para que o gerador pudesse se aproximar da realidade.

Considerando a possibilidade de uso dos dados da Usina ou Britagem, em função da quantidade menor, como já foi mencionado, as características das distribuições dos alarmes da Britagem foram utilizadas na construção do gerador.

3.2.1.4 Caracterização da Duração dos Alarmes

A forma como as durações de cada ocorrência de alarme se dão, causa impacto direto nos algoritmos de mineração. Tanto na mineração de sequências, quanto na análise de correlação cruzada, a escolha do tamanho de janela é diretamente influenciada pela conclusão sobre como estes valores são distribuídos. A determinação da função que melhor representa a distribuição de probabilidade das durações dos alarmes foi feita através método MLE – *Maximum Likelihood Estimation*, aplicado para a análise de regressão não-linear sobre a distribuição em uma escala LOG-LOG das durações dos alarmes na Britagem.

O objetivo do método MLE é determinar os parâmetros que maximizem a probabilidade dos dados amostrais em relação a um modelo paramétrico, que é uma função de distribuição escolhida. Várias possíveis distribuições foram avaliadas na tentativa de identificar a função que pudesse fornecer o melhor ajuste de curva de forma a representar adequadamente o comportamento real da duração dos alarmes. Na comparação das funções (Figura 13), entre as funções Normal, Lognormal, Poisson e Exponencial, a função Exponencial é a que forneceu o melhor ajuste de curva, com o maior valor para o Coeficiente de Determinação $R^2 = 0,74$.

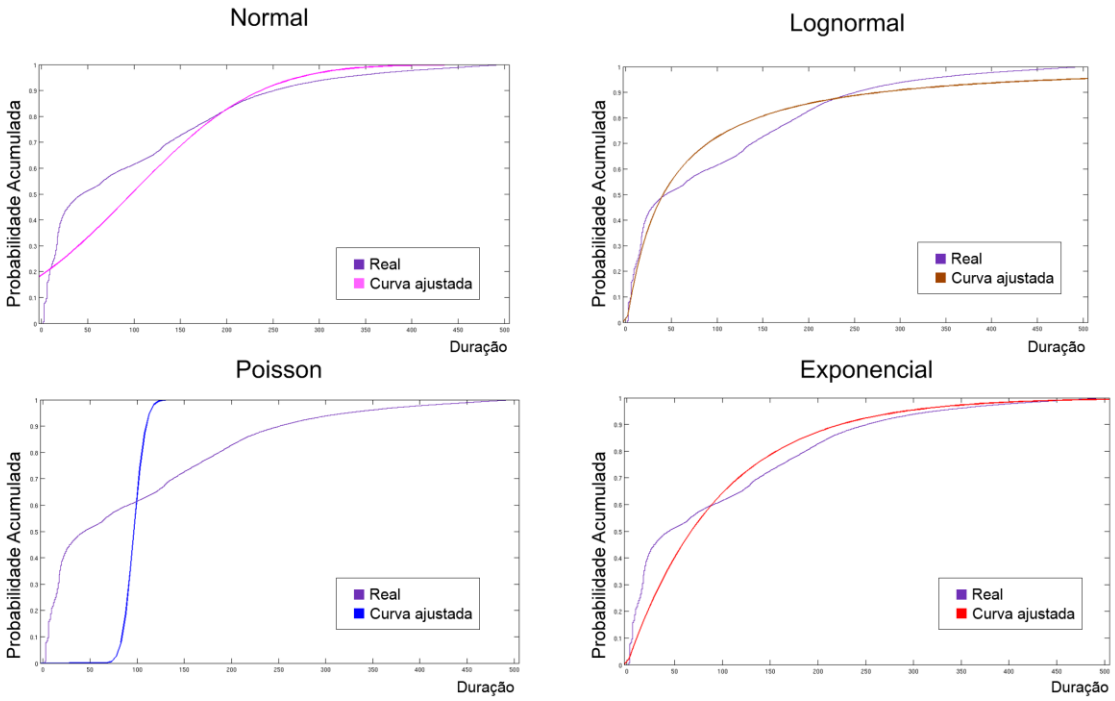

Figura 13 - Ajuste de Curva para Duração dos Alarmes: Normal, Lognormal, Poisson e Exponencial (melhor ajuste)

Os parâmetros da função Exponencial **(Eq 3.1** para PDF – *Probability Density Function* – Função de Densidade de Probabilidade**)** foram estimados com base no valor de β **(Eq 3.2**), uma alternativa de parametrização recíproca ao parâmetro λ, sendo, portanto, $\beta=\lambda^{-1}$.

$$f(x;\lambda) = \begin{cases} \lambda e^{-\lambda x}, & x \geq 0, \\ 0, & x < 0. \end{cases} \qquad \textbf{(Eq 3.1)}$$

$$f(x;\beta) = \begin{cases} \frac{1}{\beta} e^{-x/\beta}, & x \geq 0, \\ 0, & x < 0. \end{cases} \qquad \textbf{(Eq 3.2)}$$

A Figura 14 apresenta os resultados do melhor ajuste de curva (*fitting*) da função exponencial na distribuição das durações de alarmes, onde o valor de $\beta=97,2361 \approx \lambda=0,01$. Para o ajuste, sem perda de generalidade (considerando o escopo e propósito do trabalho), os valores da cauda pesada de duração acima de 500 segundos foram desconsiderados. O pequeno desvio da curva na região abaixo da região de 50 segundos pode ser atribuído a uma concentração de alarmes correlacionados nesta região, comprovada mais tarde na análise sobre os dados reais.

A leitura da distribuição das durações da Britagem na Figura 10 (cor azul) mostra que há uma probabilidade elevada de ocorrências de alarmes com curta duração – alarmes espúrios [1] – e uma baixa probabilidade de ocorrência de alarmes com tempos de duração muito grandes. Neste trecho há uma tendência de cauda pesada (concentração de valores elevados de alta duração no fim da curva).

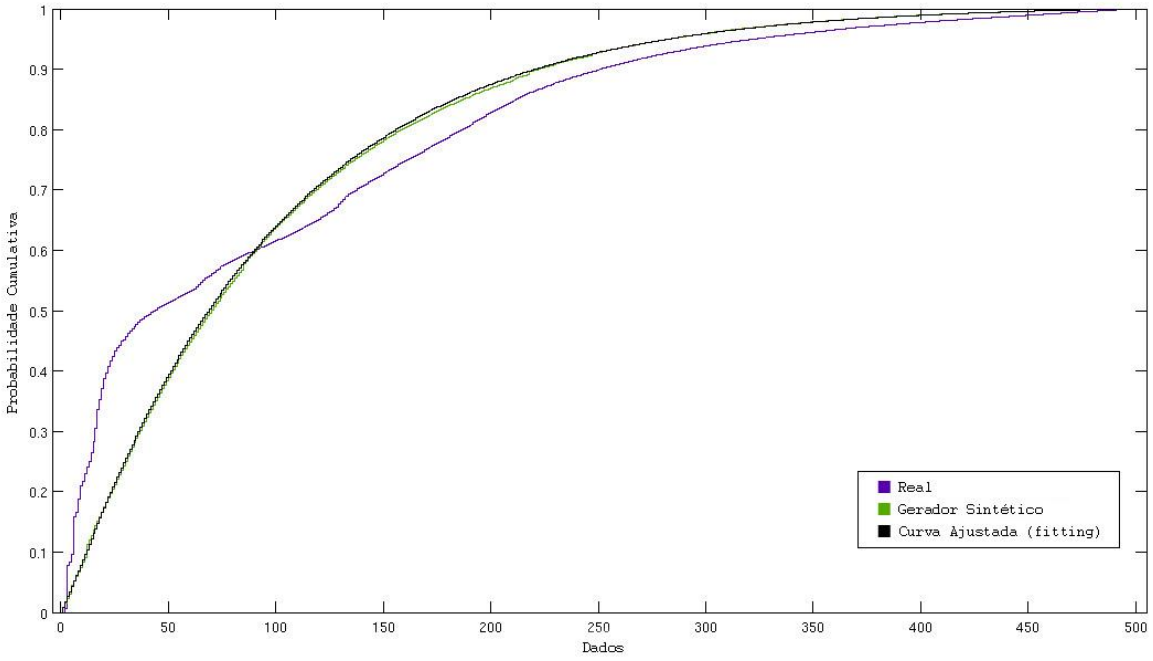

Figura 14 - Distribuições da Duração Real da Britagem (violeta), Duração na Curva Ajustada (preto) e Curva da Duração Gerada pelo Gerador Sintético (verde)

3.2.1.5 Caracterização dos Intervalos Entre Inícios (CFN) dos Alarmes

A identificação da função que melhor pudesse representar o comportamento dos intervalos entre inícios de alarmes (CFN) também foi feita utilizando o método MLE em ferramenta apropriada para o *fitting*. Na comparação das funções (Figura 16), entre as funções Normal, Logistic, Poisson e Exponencial, a função Exponencial também foi a que forneceu o melhor ajuste de curva, com o maior valor para o Coeficiente de Determinação R^2 = 0,87. Neste ajuste, sem prejuízo à generalidade, também foi aplicando o corte dos intervalos superiores a 500 segundos.

A Figura 16 apresenta os resultados do melhor ajuste de curva (*fitting*) da função exponencial na distribuição dos intervalos, onde o valor de $\beta=39,3517 \approx \lambda=0,02$.

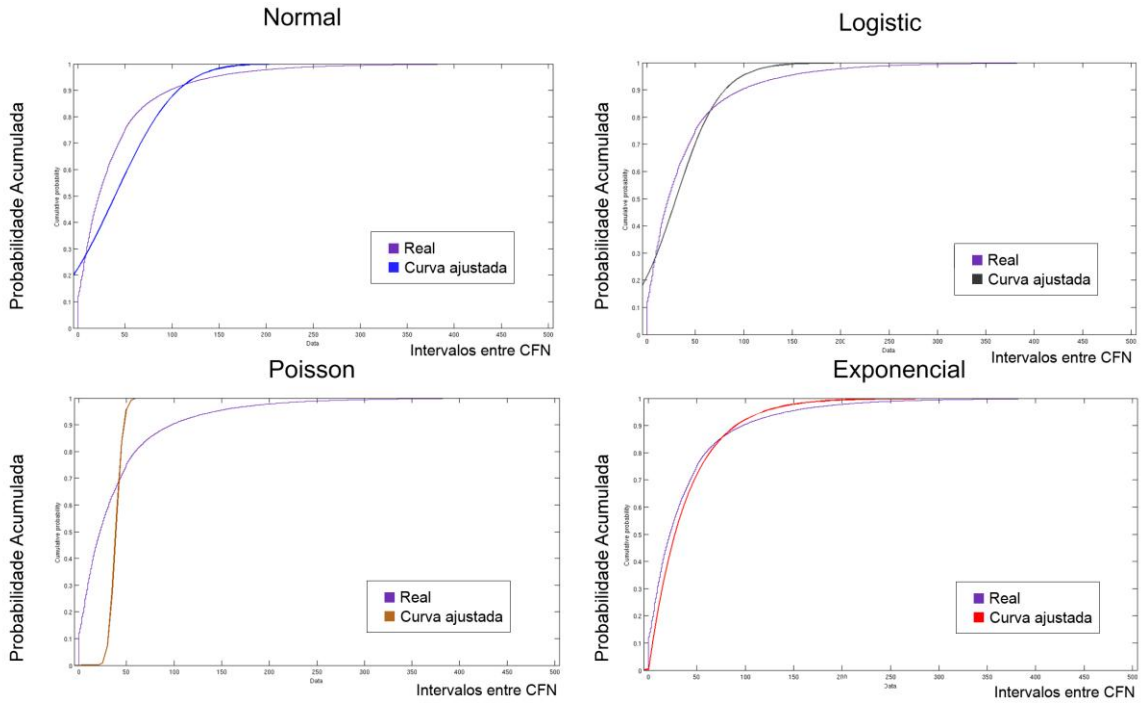

Figura 15 - Ajuste de Curva para Duração dos Alarmes: Normal, Logistic, Poisson e Exponencial (melhor ajuste)

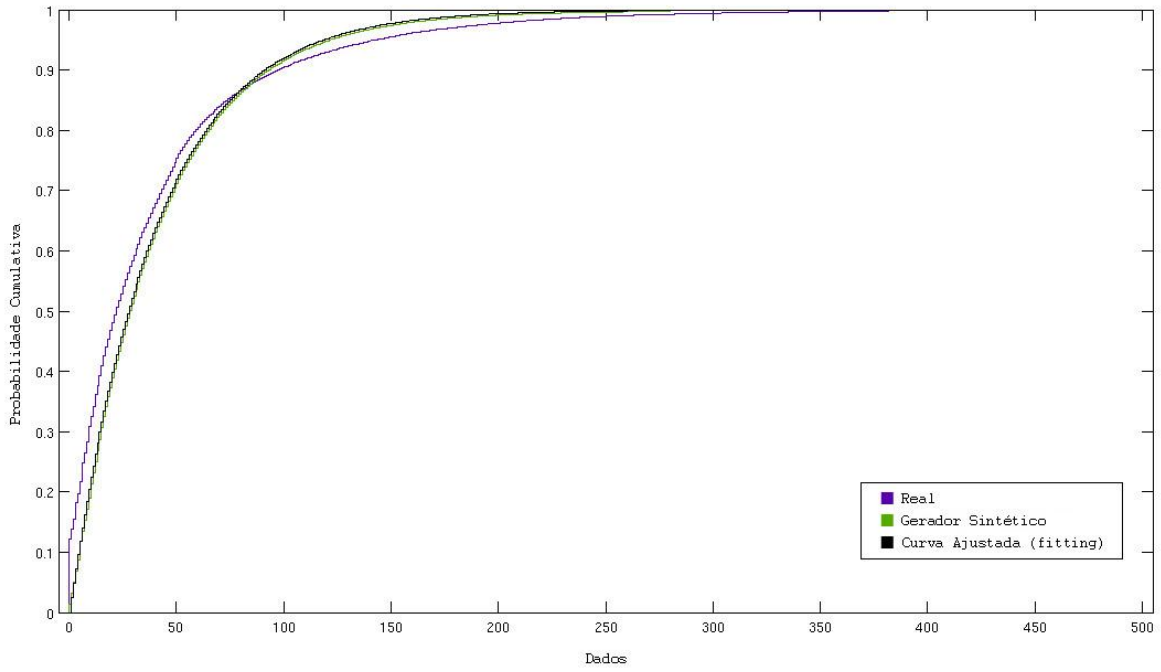

Figura 16 - Distribuições dos Intervalos entre CFNs Real da Britagem (violeta), na Curva Ajustada (preto) e na Curva dos Intervalos Gerados pelo Gerador Sintético (verde)

3.2.2 Construção e Validação do Gerador

O gerador de carga sintética foi construído a partir dos parâmetros das funções descobertos na etapa de caracterização e todo o processo foi concluído em três passos fundamentais: a) criação do gerador de carga sem padrões, b) planejamento e inserção dos padrões temporais artificiais e c) validação. Estas etapas serão apresentadas em mais detalhes a seguir.

3.2.2.1 Criação do Gerador de Carga Sem Padrões

O gerador de cargas foi construído com o objetivo simples de gerar uma base de dados de alarmes cujas distribuições das quantidades, durações e intervalos entre CFNs se aproximassem das características da base de dados da Britagem, para a qual os parâmetros das curvas foram determinadas na etapa de caracterização.

A Figura 17 apresenta um diagrama de classes do gerador de cargas implementado na linguagem Java, destacando os principais componentes necessários ao seu funcionamento:

- **WorkloadGenerator**: principal componente do gerador, que possui o método generate(),que gera um arquivo de texto onde cada linha possui uma ocorrência de alarme (AlarmOccurrence). Os valores de β para as curvas ajustadas das distribuições de duração e intervalo entre CFNs são parâmetros para a geração da base sintética;
- **AlarmOccurrence**: objeto que irá representar cada linha na base de dados de alarmes no formato <Código> <CFN> <RTN>;
- **Quantities**: em função da quantidade gerenciável de tipos de alarmes na Britagem, a distribuição das quantidades sintéticas foi feita diretamente a partir de cada quantidade de cada tipo presente na base de dados original, contada e modelada em uma estrutura estática do tipo Enumeração (*Enumeration*);
- **DurationGenerator**: o gerador de durações é uma instância da classe ExponentialGenerator, um gerador de distribuições exponenciais. O seu funcionamento é muito simples. Durante a construção do objeto, uma lista temporária com as quantidades para cada valor de duração é criada a partir da PDF construída a partir do parâmetro β pela classe Exponential. Para cada par duração-quantidade desta lista temporária, são inseridos na lista PDFPoints [quantidade] elementos de cada [duração], criando uma lista contínua com todos os pontos da PDF. A partir de então, dois métodos ficam disponíveis: getRandomic(), que seleciona de maneira aleatória uniformemente distribuída um ponto desta lista, e getSpecific(int key), que seleciona um ponto da PDF cujo valor de duração seja [key]. Este segundo método é utilizado mais tarde para a geração dos padrões artificiais. . A cada execução de getRandomic() ou getSpecific(), um ponto é removido da lista;
- **CFNGenerator**: o gerador de intervalos entre CFNs funciona de maneira análoga ao gerador de durações, com a diferença que os pontos da lista PDFPoints representam os pontos de intervalos entre inícios dos alarmes, de acordo com a distribuição caracterizada;

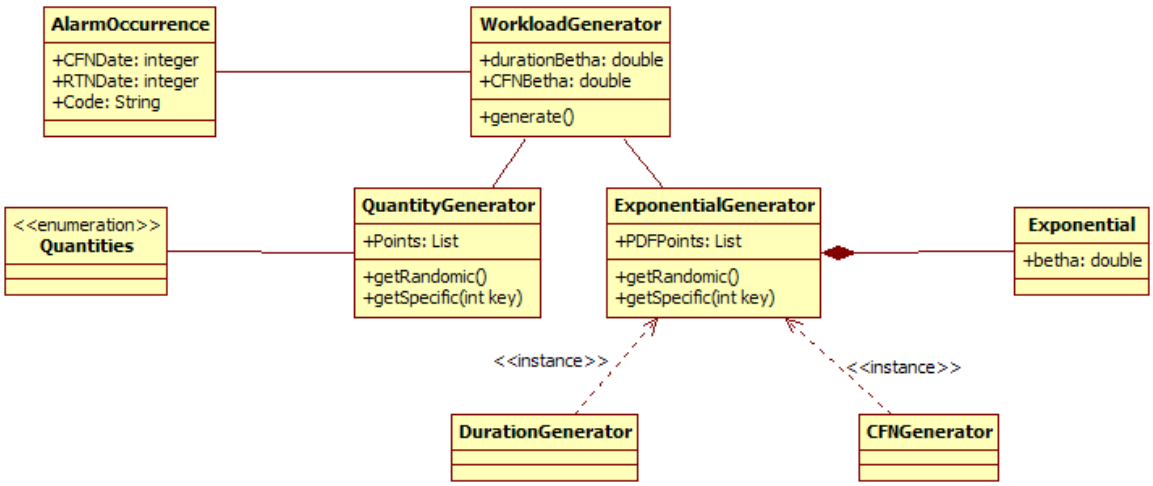

Figura 17 - Diagrama de Classes do Gerador de Cargas

O funcionamento do algoritmo de geração é bem simples e pode ser explicado através do pseudo-algoritmo abaixo (Algoritmo 3.1). A cada iteração um objeto do tipo "ocorrência de alarme" é criado a partir dos geradores de duração e CFN, que escolhem aleatoriamente, um dos pontos da PDF, gerados a partir dos parâmetros estimados das curvas. O código é selecionado pelo gerador de quantidades, que decrementa uma unidade para cada código a cada execução. O valor de RTN é calculado a partir dos valores de CFN e duração. O algoritmo é executado até que a quantidade de alarmes calculada pelo gerador de quantidades seja igual à zero.

Algoritmo 3.1 – Geração da Base de Dados de Alarme Sintética Sem Padrões

```
//initialization
quantAlarms = calculateQuantities();
createDurationGenerator(β);
createCFNGenerator(β);
code = "";
CFN = 0;
RTN = 0;
duration = 0;
List alarmOccurrences; //create empty list for occurrences

//alarm list generation
WHILE quantAlarms > 0 DO
  code = QuantityGenerator.getRandomic();
  duration = DurationGenerator.getRandomic();
  CFN = CFN + CFNGenerator.getRandomic();
  RTN = CFN + duration;
  alarmOccurrences.put (code, CFN, RTN);
END
writeFile(alarmOccurrences);
```

3.2.2.2 Planejamento e Inserção dos Padrões Temporais Artificiais

Considerando o objetivo de construção de um algoritmo capaz de detectar todos os tipos de padrões temporais dos axiomas de Allen (vide Figura 8), mencionados na seção 2.3.1, uma relação artificial para cada padrão foi criada. Os pares foram escolhidos a partir da lista de quantidades para cada tipo de alarme de maneira arbitrária, porém, envolvendo diversos valores de quantidades (padrões com pequenas e grandes quantidades convivem, portanto, na mesma base de dados). Os padrões são listados e descritos a seguir:

- XEY (X *equals* Y / X igual a Y): neste padrão, dois alarmes tem hora de início e término idênticos e, portanto, duração idêntica. Para implementar este

padrão, o valor de duração é fixado de acordo com o ponto na distribuição de durações para a quantidade correspondente aos códigos selecionados;

- XSY (X *starts* Y / X inicia Y): neste padrão as horas de início de X e Y são idênticas, mas as horas de término e, portanto, duração, são diferentes. Para implementar este padrão, o valor de CFN de X é determinado aleatoriamente (sempre a partir da lista de pontos da PDF, neste e nos demais padrões) e o valor de CFN de Y é copiado de X. As durações de X e Y são determinadas aleatoriamente e o valor de RTN de X e Y calculados a partir desta duração;

- XFY (X *finishes* Y / X finaliza Y): neste padrão, as horas de término (RTN) de X e Y são idênticas, mas as horas de início e, portanto, duração, são diferentes. Para implementar este padrão, os valores de CFN de X e Y são determinados aleatoriamente, o valor de duração de X é fixado (sempre de acordo com o ponto na distribuição de durações para a quantidade correspondente aos códigos selecionados, através dos métodos *getSpecific*), e o valor de duração de Y é calculado. RTN de X e Y é igual a CFN de X + duração de X e a duração de Y é igual a RTN de X – CFN de Y;

- YMX (Y *met by* X / início de Y coincidente com término de X): neste padrão, CFN de Y é igual ao RTN de X, ou seja, Y começa no mesmo instante em que X termina. Para implementar este padrão, basta fixar o valor de CFN de Y igual ao valor de CFN de X + duração de X. Os valores de CFN de X, duração de X e duração de Y são determinados aleatoriamente;

- YOX (Y *overlapped by* X / Y sobreposto por X): neste padrão, X e Y são sobrepostos, como em uma cascata sequêncial de alarmes. Para implementar este padrão, o valor de duração de X e Y é fixado, o valor de CFN de X é selecionado aleatoriamente nos pontos da PDF e o valor de CFN de Y é igual ao CFN de X + K, uma constante que determina o tempo de sobreposição. Neste caso, o valor de K foi determinado com o valor de 10s;

- YDX (Y *during* X / Y durante X): neste padrão, os valores de duração de X e Y são fixos, mas a duração de Y é menor, de forma que Y comece e termine durante o tempo de ocorrência de X. Para implementar este padrão, além de fixar as durações de X e Y, o intervalo entre os inícios de X e Y também é

fixado. Os valores de CFN de X e Y são determinados aleatoriamente nos pontos da PDF e os valores de RTN são calculados;

- YOXTRIPLE (Y *overlapped by* X in Triple / X sobrepõe Y triplamente): este padrão é a extensão do padrão XOY, com a exceção de que três, ao invés de dois tipos de alarmes diferentes são envolvidos nesta relação de cascata de alarmes. Este padrão não foi considerado nos gráficos da análise de resultados, já que o padrão é equivalente ao padrão XOY.

Estas relações foram chamadas de relações simples, já que são padrões de antecedentes e consequentes unitários (note que, mesmo a relação YOXTRIPLE continua sendo uma relação deste tipo, já que uma cascata pode ser descrita no formato A→B→C) e que, portanto, podem ser identificadas na mineração das sequências de tamanho dois (2), uma etapa do processo que será apresentada adiante (seção 3.4). Adicionalmente, uma relação do tipo complexa envolvendo uma sequência de tamanho três (3) foi adicionada para testar a capacidade do algoritmo neste ponto:

- YOXAR (Y *overlapped by* X *in triple association rule* / regra de associação X sobrepõe Y): esta regra se comporta de maneira semelhante à regra YOX, com a exceção de que é composta por três tipos de alarmes X, Y e Z. X e Y possuem uma relação XEY em parte dos dados. Sempre que XEY ocorre, ZO(XEY) ocorre, ou seja, o terceiro tipo Z é sobre posto pelo par XEY. O resultado é uma regra de associação XY=>Z. Para implementar esta relação os tipos X e Y são alarmes com mesmos valores de quantidade e Z possui um valor inferior. Sempre que Z ocorre, X e Y são forçados a ocorrerem nesta relação.

A modificação do algoritmo (Algoritmo 3.1) para a inserção dos valores pode ser vista de maneira grifada no Algoritmo 3.2. Para simplificar a apresentação, apenas a geração de um dos tipos de padrões (XSY) é mostrada.

A introdução dos padrões artificiais é feita com pequenas modificações no algoritmo original. Como os padrões são conhecidos a priori, após a seleção aleatória do código dentre a distribuição das quantidades para cada tipo, um teste é feito para verificar se o código corresponde a algum código que faz parte de um padrão. Caso negativo, o algoritmo se comporta como o anterior. Caso positivo, dois alarmes correspondendo ao par do padrão artificial são inseridos simultaneamente, ao invés de apenas um. Caso valores

específicos para duração ou intervalo entre CFNs sejam necessários, estes pontos são removidos da lista de pontos da PDF através do método getSpecific. O mesmo ocorre para remover os códigos do padrão da lista de quantidades. Isto garante que a distribuição não é afetada, mesmo com a inserção dos padrões com valores fixados manualmente. Embora esta estratégia não possa ser avaliada em relação ao caso real, já que todos os padrões de uma base de dados real não podem ser conhecidos a priori, a solução tenta inserir padrões de maneira abrangente sem interferir nas distribuições de probabilidade para quantidade, duração e intervalos entre inícios dos alarmes.

Algoritmo 3.2 – Geração da Base de Dados de Alarme Sintética COM Padrões

```
//initialization
quantAlarms = calculateQuantities();
createDurationGenerator(β);
createCFNGenerator(β);
code = "";
CFN, CFNX, CFNY = 0;
RTN, RTNX, RTNY = 0;
duration, durationX, durationY = 0;
List alarmOccurrences; //create empty list for occurrences

//alarm list generation
WHILE quantAlarms > 0 DO
  code = QuantityGenerator.getRandomic();
  IF (isCodeFromPatternXSY(code)) //if code is X or Y then insert
     //generate X
     CFNX = CFNGenerator.getRandomic();
     durationX = DurationGenerator.getRandomic();
     CFNX = CFNX + CFNGenerator.getRandomic();
     RTNX = CFNX + durationX;
     alarmOccurrences.put (X, CFNX, RTNX);
     // generate Y
     CFNY = CFNGenerator.getSpecific(0);//zero because its same time
     durationY = DurationGenerator.getRandomic();
     CFNX = CFNX + CFNGenerator.getRandomic();
     RTNY = CFNY + durationY;
     alarmOccurrences.put (Y, CFNY, RTNY);
  ELSE //non pattern
    duration = DurationGenerator.getRandomic();
    CFN = CFN + CFNGenerator.getRandomic();
    RTN = CFN + duration;
    alarmOccurrences.put (code, CFN, RTN);
END
writeFile(alarmOccurrences);
```

3.2.2.3 Validação

Esta etapa foi executada com o objetivo de atestar que os parâmetros originalmente caracterizados continuavam seguindo as distribuições identificadas, mesmo com os padrões artificiais inseridos. Para que isto fosse possível, novas curvas de distribuição de duração e intervalos entre CFNs foram desenhadas. Nas figuras Figura 14 e Figura 16 é possível perceber uma similaridade das curvas de cores verde, que representa a curva ajustada, e preta, que representa as distribuições com os dados da carga sintética com padrões, o que confirma a ausência de interferência significativa.

3.3 Uma Abordagem Simplista

Para avaliar as alternativas de solução em termos do parâmetro janela temporal deslizante e o impacto da modelagem visual dos resultados, uma abordagem mais simples foi implementada no primeiro momento, sem considerar ainda as demais alternativas de mineração de dados possíveis.

Esta primeira abordagem envolveu a utilização dos conceitos propostos em [18] para identificar padrões temporais em conjuntos de dados de alarmes reais de uma planta industrial de mineração através das definições de “estruturas de backbone” e “backbone agregado”. No trabalho de KLEINBERG et al. [18], os autores propuseram determinar a espinha dorsal de uma rede onde os vértices representavam os indivíduos de uma rede social e as arestas representavam as trocas de mensagens entre eles. As arestas eram ponderadas pelo “*delay*” existente entre as trocas de mensagens e a espinha dorsal era definida como sendo o menor caminho entre quaisquer pares de vértices considerando todo o período de estudo. O *gap* entre a troca de mensagens entre dois vértices quaisquer *u* e *v* era calculado através da análise da quantidade de mensagens trocadas por intervalo de tempo, considerando uma divisão do tempo total T em períodos iguais. Assim, o *delay* entre *u* e *v* era dado pela divisão de T pela quantidade de mensagens trocadas entre *u* e *v* no tempo total T. A espinha dorsal, em resumo é dada pelo subgrafo contendo todas as “arestas essenciais”, ou seja, as arestas com a menor soma total de *delays* entre dois nós.

O conceito de espinha dorsal proposto no trabalho de Kleinberg et. al. foi explorado utilizando as seguintes modificações na proposta original:

- Os vértices da rede foram modelados como sendo os tipos diferentes de alarmes presentes nos arquivos de *logs* no período analisado;
- As arestas foram modeladas como sendo os alarmes que ocorreram dentro de uma mesma janela temporal. Fazendo uma analogia ao trabalho de Kleinberg et. al., em que as arestas representavam as trocas de mensagens entre pessoas, aqui as arestas representam trocas de mensagens (alarmes) que ocorreram dentro das janelas de tempo definidas;
- O peso das arestas foi modelado de duas formas diferentes, para uma comparação posterior:
    a) Utilizando a frequência das ocorrências de alarmes dentro das janelas de tempo (ex. se, ao deslizar a janela de tempo, os alarmes A e B ocorreram juntos duas vezes dentro da janela, então o valor da aresta é igual a 2). Apesar de utilizar a frequência, esta abordagem difere da mineração de sequências, uma vez que este valor é utilizado para ponderar as arestas que compõem um grafo e as análises são feitas par-a-par apenas;
    b) Utilizando as distâncias temporais entre início e término de dois alarmes diferentes dentro da mesma janela (ex. se ao deslizar uma janela de tempo de 60 segundos, os alarmes A e B foram iniciados ao mesmo tempo, então o valor da aresta é igual a 60; caso tenham sido iniciados com uma diferença de 59 segundos, então o valor da aresta é igual a 1). Neste caso, à medida em que a janela desliza, comparando alarmes par-a-par, o valor das arestas vai sendo incrementado à medida em que pares de alarmes ocorrem de forma repetida dentro da janela de tempo;

Foram considerados na contabilização do peso das arestas, os alarmes ocorridos com hora de início ou hora de término ocorrendo dentro da janela de tempo. Assim, o algoritmo de geração do peso foi executado duas vezes, gerando um conjunto único de padrões resultantes: deslizando a janela sobre os horários de início (CFN) e deslizando a janela sobre os horários de término (RTN). Os valores de arestas foram também normalizados para um ajuste visual na representação e comparação.

3.3.1 Algoritmo e Ferramentas

O algoritmo foi implementado na uma linguagem Java foi e dividido em duas partes:

1. Algoritmo de *"parse"* com janela deslizante sobre horário de início do alarme: este algoritmo utilizou como entrada o arquivo contendo os registros de alarmes ordenados pela hora de início do alarme;
2. Algoritmo de "*parse"* com janela deslizante sobre CFN e RTN: este algoritmo utilizou como entrada o arquivo com os registros de alarmes ordenados pela hora de término do alarme e também um arquivo de saída do algoritmo 1 contendo os vértices e arestas coletados sobre os horários de início.

Em resumo, o algoritmo analisa ocorrências de alarmes dentro da janela de tempo com um valor pré-determinado e monta duas listas (dois vetores): uma contendo a frequência em que os alarmes iniciaram ou terminaram juntos dentro da janela e outra lista contendo a soma das diferenças dos tempos de início ou término dentro da janela. O custo computacional do algoritmo é limitado e inferior a $O(n^2)$, considerando que a travessia pela base de dados é feita apenas uma vez, com um loop interno executado tantas vezes quantos forem os alarmes que se encontram dentro da mesma janela de tempo. O algoritmo foi executado duas vezes, variando o valor de TamanhoDaJanela para os valores de 30 e 60 de segundos. Os arquivos contendo as listas de arestas e vértices foram utilizados como entrada do software Pajek[32], para o cálculo das métricas de redes complexas e comparação dos diferentes resultados gerados por cada abordagem. Da mesma forma como ocorre na mineração de sequências, é necessário estipular valores de corte para a geração do grafo contendo a espinha dorsal com os alarmes mais frequentes. A escolha dos valores de corte para a geração foi feita no primeiro momento, com base em uma análise preliminar das distribuições de probabilidade dos valores de arestas considerando a análise dos vetores de frequência e peso, e, em seguida, visualmente, destacando os alarmes com relações mais intensas dos grupos de relações menos relevantes.

Para descobrir as relações entre alarmes que compõem a espinha dorsal (backbone agregado) as estruturas foram analisadas visualmente, inicialmente, através da geração dos das redes utilizando o software Pajek. As imagens foram geradas para os valores de corte de arestas (normalizadas entre 0 e 10) com valores superiores a 0,1, ou

seja, com relevância superior a 1% (Figura 18), superiores a 0,5, 5% (Figura 19) e superiores a 20% de relevância (Figura 20). Os resultados são discutidos na seção 3.3.2.

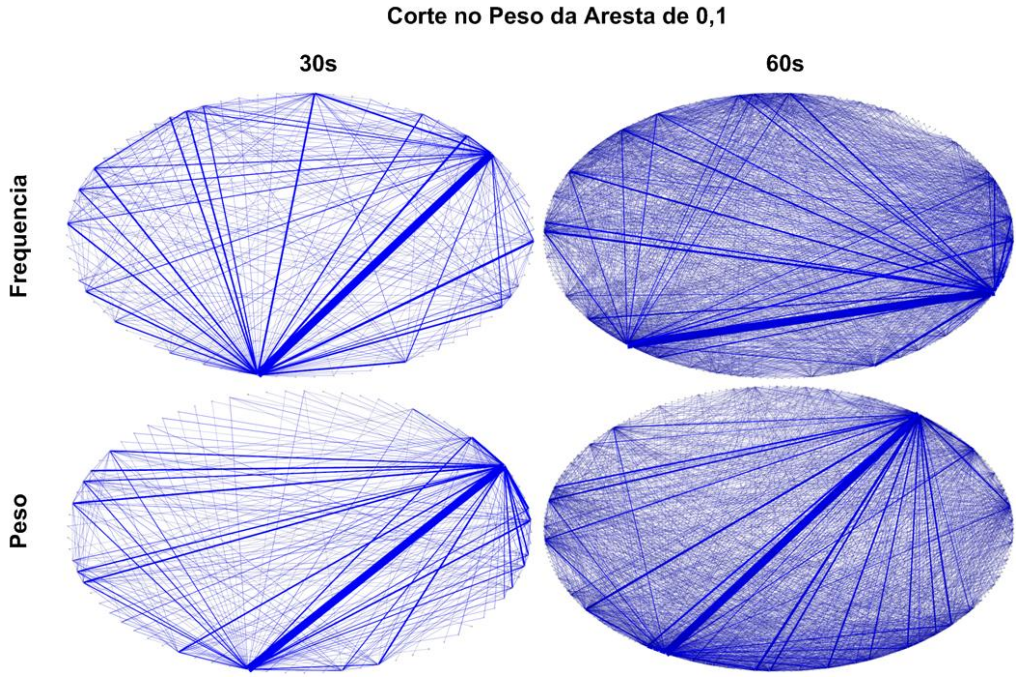

Figura 18 - Redes Ponderadas com Frequência e Peso para Valor de Corte de Arestas Superiores a 0,1

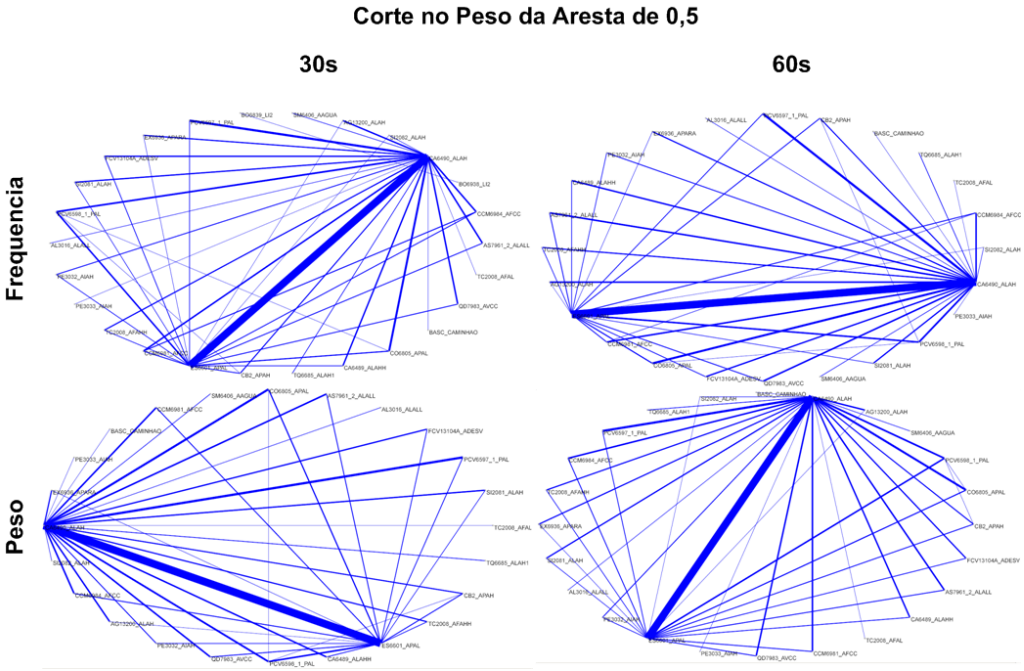

Figura 19 - Redes Ponderadas com Frequência e Peso para Valor de Corte de Arestas Superiores a 0,5

3.3.2 Resultados

Na mineração de sequências, um problema crítico é a determinação do tamanho da janela deslizante, como já foi mencionado. Para avaliar a influência da variação do tamanho da janela nos padrões gerados, foi utilizada a métrica de redes complexas de média de distância entre dois vértices. Esta métrica considera o valor das arestas e fornece para todos os pares alcançáveis, o valor médio de distância entre eles. A mudança neste valor pode servir como um indicativo da intensidade da variação dos resultados quando este parâmetro é alterado. Os valores são indicados na Tabela 1 e mostram que, no grafo mais denso, a variação da média chegou ao máximo de cerca de 50% do maior para o menor valor com a variação deste parâmetro, razão pela qual pode-se concluir que a métrica de média de distância entre dois vértices sofre grande variação em função das alterações nas características do algoritmo (por exemplo, alterando o tamanho da janela deslizante).

Tabela 1 - Variação da Métrica de Média dos Caminhos Entre Dois Vértices para o Grafo de Padrões

	Janela 30s				**Janela 60s**			
	0,1		**0,5**		**0,1**		**0,5**	
	Freq	**Peso**	**Freq**	**Peso**	**Freq**	**Peso**	**Freq**	**Peso**
Média dos Caminhos	2,879127	2,952413	5,347356	5,353795	1,524005	1,498187	5,383616	5,312436

Em relação à representação através do grafo, a abordagem pode ser considerada uma melhoria para o processo de análise semântica pelos seguintes motivos:

- Através do suporte visual com arestas e vértices é possível identificar relações e grupos de relações entre alarmes de forma mais intuitiva;
- É possível, com facilidade, alterar os valores de corte para gerar grafos mais ou menos complexos. As métricas de redes complexas podem ser utilizadas para comparar diferentes grafos antes de partir para a análise semântica;

- A avaliação considerando CFNs e RTNs possibilita a inclusão das relações temporais que são unilaterais (apenas CFN ou apenas RTN);
- É possível identificar diferenças na intensidade das relações dentro de grupos que representam cascatas sequenciais.

As relações descobertas no grafo gerado foram enumeradas validadas através de análise de correlação. Para esta avaliação foi gerado um grafo contendo arestas com peso superior a 2, ou os valores de relevância superiores a 20% (Figura 20), utilizando-se o vetor de pesos com janela de 60 segundos.Os padrões encontrados foram identificados, posteriormente, também nos resultados gerados pela execução do algoritmo definitivo, demonstrando uma capacidade existente de identificação de padrões.

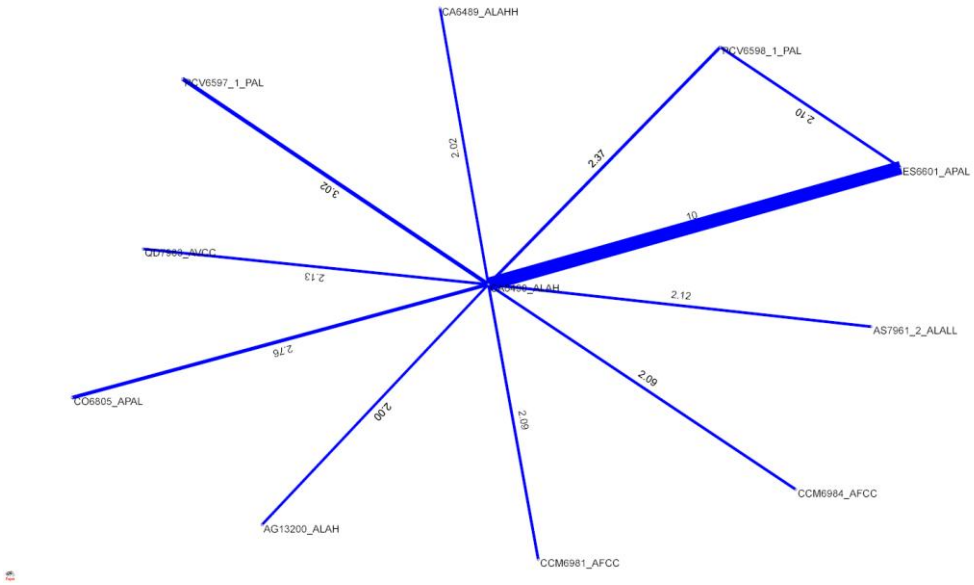

Figura 20 - Grafo Gerado com Corte de Arestas Inferiores a 2 com Ponderação por Peso e 60 Segundos de Janela

3.3.3 Deficiências

O algoritmo proposto pode ser visto como uma generalização do algoritmo GSP[5], com contagem de suporte por parâmetro de janela deslizante. Apesar dos resultados em termos de detecção de padrões válidos com baixo custo computacional, as seguintes deficiências podem ser enumeradas:

- Assim como nos demais algoritmos de mineração de sequências, o parâmetro de tamanho da janela exerce grande influência sobre os resultados;
- A quantidade com que os alarmes ocorrem exerce uma relação de proporcionalidade direta com a força das relações, não existindo nenhuma outra métrica (como grau de confiança, por exemplo) para possibilitar a detecção das relações menos frequentes, mas relevantes;
- A junção dos resultados da análise para CFN e RTN conduz o algoritmo à detecção de padrões altamente correlacionados em termos de início e término dos alarmes e, portanto, correlacionados em termos de duração, o que faz com que os alarmes de alto coeficiente de correlação é que sejam detectados com maior destaque.

3.4 Descrição da Solução para a Nova Abordagem

A proposta de solução para o problema não consiste apenas em um algoritmo ou na combinação de algoritmos, mas de um **processo** (um conjunto de atividades) que conduz o minerador aos alarmes redundantes a partir da base de dados de registros de alarmes. Este processo é apresentado resumidamente a seguir:

1) A base de dados de alarmes é tratada em um processo de limpeza, descrita na seção 3.1;
2) Duas cópias são geradas da base de dados original tratada: uma ordenada pelo CFN e outra ordenada pelo RTN;
3) Uma variação do algoritmo *cross-effect-test* (apresentado n seção 2.2.2.1) é executado sobre cada base de dados para gerar transações a partir dos registros de alarmes. Duas novas bases de dados contendo transações de alarmes que ocorreram dentro das janelas deslizantes são geradas nesta etapa;
4) As duas bases de dados são utilizadas para alimentar o algoritmo de mineração de sequências PrefixSpan [26], modificado para gerar apenas as sequências de tamanho 2 que são tratadas como regras de associação, calculando-se o valor da confiança da relação. Dois conjuntos de padrões, para CFN e para RTN são gerados como resultado;

5) Para determinar as regras complexas com tamanho maior que 2, as duas bases de dados são utilizadas para alimentar o algoritmo de mineração de regras de associação mínimas não redundantes - MNR[31], que gera outros dois conjuntos de padrões;
6) O algoritmo análise de correlação-cruzada é executado sobre a base de dados original;
7) Para a visualização, duas redes complexas (dois grafos) são gerados: o primeiro combinando os resultados da correlação cruzada e os padrões de tamanho 2, e o segundo contendo apenas os resultados da mineração de regras complexas com MNR;
8) Uma ferramenta de exploração e análise de redes complexas (Pajek) é utilizada para navegar nos padrões, realizar os cortes das arestas menos relevantes e detectar os componentes conexos que representam relações interessantes.

A seguir, os principais componentes serão investigados com maior profundidade.

3.4.1 Algoritmo de Geração de Transações com Janela Deslizante (*Sliding Window*)

Na abordagem de KORDIC et al. [13], as transações são geradas a partir de duas janelas: a janela de ativação WA-R e a janela de retorno WR-A/R-w, que, na verdade é limitada por um valor R-w atribuído pelo usuário. O tamanho da janela WA-R é determinado pelo tempo de duração do alarme corrente no deslizamento da janela (no exemplo da Figura 6, o tamanho da janela atual é determinado pelo primeiro alarme da lista, de TAG 1, com duração de 30 segundos, que termina com a marcação -1). A janela de retorno WR-A é uma janela com tamanho idêntico à janela de ativação, com início a partir do término da mesma (limitada ao valor R-w). A transação é composta pela intersecção entre os conjuntos de alarmes que foram ativados dentro da janela WA-R e finalizados dentro da janela WR-A/R-w.

A justificativa para a abordagem *cross-effect-test* baseia-se no fato de que todas as possíveis relações são relações que iniciam-se na janela WA-R e finalizam dentro da janela WR-A/R-w. Este tipo de relação se enquadraria no tipo de padrão temporal definido por Allen como o padrão *Overlaps(i,j)*, já que, nesta configuração, o início de *j* ocorre depois do início de *i* e o término de *j* ocorre depois do término de *i*. Esta solução de geração do conjunto de transações é, portanto, insuficiente para detectar os demais padrões temporais possíveis.

O novo algoritmo proposto para a geração das transações, faz uso da mesma estratégia para a determinação do tamanho da janela, mas apenas as janelas WA-R e WR-A são utilizadas de maneira independente (razão pela qual o parâmetro *minw* não é necessário), sendo a janela WA-R utilizada na base ordenada pelo CFN e a janela WR-A é utilizada na base ordenada pelo RTN. Assim, na análise da entrada ordenada pelo CFN, dois alarmes A e B que sejam iniciados dentro de uma mesma janela WA-R de maneira consecutiva são considerados como parte de uma mesma transação, indicando, portanto, que A precede B nesta transação. Todos os demais alarmes que sejam iniciados dentro desta janela serão também considerados parte da transação corrente. Da mesma forma, na análise da entrada ordenada por RTN, a ocorrência consecutiva indicaria que o término de A é sucedido pelo término de B nesta transação. Considerando o exemplo da Figura 21 e dois conjuntos de transações T_{CFN}={∅} e T_{RTN}={∅}, inicialmente vazios, na etapa de execução para CFN, ao posicionar o cursor na TAG 1, uma janela $W_{A\text{-}R}$ (cor azul) de 30 segundos seria formada (que é igual ao tempo de duração do alarme). Todas as ativações que ocorrerem nesta janela farão parte da primeira transação (a transação correspondente ao posicionamento do cursor na posição TAG 1). Após esta execução, o conjunto de transações, T_{CFN} = { (1,2,3) }, ou seja, uma nova transação foi inserida contendo os elementos (1,2,3), que correspondem à sequência temporal de ocorrência de ativação (CFN) destes três alarmes (TAGs). Na etapa de execução para RTN (considerando que a base de dados de alarmes da Figura 21 está agora ordenada por RTN), a janela $W_{R\text{-}A}$ (cor vermelha) de 30 segundos (duração da TAG 1) seria formada no intervalo 30 a 60 segundos. Todos os retornos que ocorrerem nesta janela farão parte da primeira transação da lista T_{RTN}, correspondente ao posicionamento do cursor na posição do TAG 1, que retorna no instante 30s. Após esta execução, o conjunto de transações, T_{RTN} = { (1,3,2,4) }, ou seja, uma nova transação foi inserida contendo os elementos (1,3,2,4), que correspondem à sequência temporal de ocorrência de retorno à normalidade (RTN) destes alarmes (TAGs). Deve-se notar que a sequência de ocorrência é preservada, razão pela qual, por exemplo, um possível conjunto (1,2,3) é diferente de outro conjunto (1,3,2).

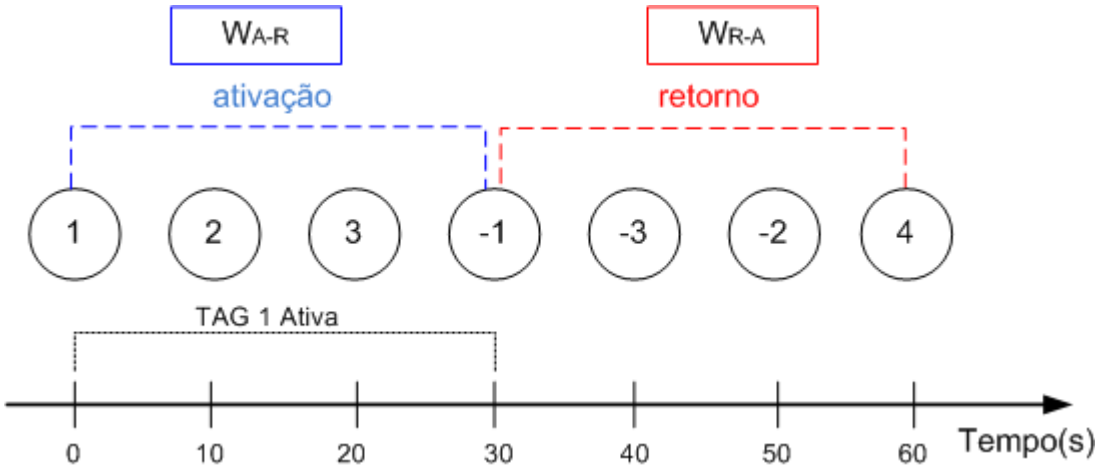

Figura 21 - Uso das janelas WA-R e WR-A na Formação das Transações

3.4.2 Mineração de Sequências de Tamanho 2 com PrefixSpan

A etapa anterior gera duas listas de transações com sequências ordenadas pelo tempo que indicam sequências de inícios ou de retornos de alarmes. A partir deste ponto, qualquer algoritmo de descoberta de padrões frequentes poderia ser utilizado. Entretanto, para manter a ordenação temporal e sintetizar os resultados, atendendo aos objetivos definidos, foi utilizada uma abordagem que combina dois algoritmos.

A mineração a partir das transações é dividida em duas etapas: os padrões que representam as relações de redundância entre alarmes consideradas simples (que são as relações mais frequentes neste contexto) são mineradas com um algoritmo de mineração de sequências de custo inferior, PrefixSpan. Fazem parte desta categoria as regras com antecedente e consequente unitários. Relações do tipo A→B ou A→B→C são exemplos de relações consideradas simples e que podem ser descobertas nesta etapa, independentemente do tipo de relação temporal de Allen existente. Na segunda etapa, as relações complexas, como AB→C, por exemplo, são tratadas com outro algoritmo (descrito na próxima seção).

PrefixSpan [26] é um algoritmo de mineração de sequências proposto por PEI et al., que reduz o esforço na geração das subsequências candidatas, obtendo uma melhoria de performance em relação às abordagens baseadas no algoritmo APRIORI [8]. Estes algoritmos utilizam uma abordagem de geração de candidatos de múltiplas passagens pela base de dados em que, a cada passagem, são gerados os itens frequentes que servem

como sementes para o próximo passo, chamadas de “sequências candidatas”. Na abordagem de mineração de sequências de Prefixos Projetados (*Prefix-projected Sequential Pattern Mining – PrefixSpan*), a ideia é examinar as subsequências que são prefixos e projetar somente as subsequências que são sufixos nas bases projetadas. Além de ser capaz de minerar o conjunto completo de padrões, o algoritmo é mais eficiente e mais rápido [26] que as abordagens anteriores GSP e FreeSpan [11].

Apesar de preservar a ordem temporal de ocorrência, o algoritmo PrefixSpan gera como resultado sequências cujo valor de suporte é superior ao parâmetro *minsup*, sem informar, no entanto, o valor de confiança para as relações. Este valor é necessário para indicar a força da regra que é o padrão de redundância entre dois alarmes. De fato, conceitualmente, a confiança só pode ser calculada para regras de associação em que há o cálculo da probabilidade condicional para as regras de implicação do tipo “antecedente=>consequente”. Neste caso, duas alternativas são possíveis: a) modificar um algoritmo de mineração de regras de associação para que a ordem temporal seja preservada, ou b) modificar o algoritmo de mineração de sequências para que calcule a confiança das regras de tamanho dois. A segunda opção foi escolhida em função em razão de três aspectos principais:

- Em termos de performance, executar o algoritmo de mineração de sequências é menos custoso, já que a enumeração dos geradores mínimos (que será apresentada em seguida) para o cálculo das regras de associação mínimas não redundantes possui alto custo computacional. Existe, portanto, a alternativa por parte do usuário de uso do algoritmo apenas para a enumeração dos padrões simples com boa performance. Além disso, nos resultados da mineração das regras complexas, a sequência temporal ainda não é preservada;
- Em termos de facilidade de implementação, a opção de modificar o algoritmo de mineração de sequências é a alternativa menos custosa, dado que os valores de suporte para as sequências de tamanho dois e subsequências de tamanho um já são calculados;
- Empiricamente percebe-se que há uma probabilidade maior de que a maior parte dos padrões, assim como os mais relevantes, faça parte da categoria definida como “padrões simples”.

Além da implementação do cálculo da confiança, o algoritmo foi modificado para gerar apenas as sequências de tamanho dois.

3.4.3 Mineração de Regras Complexas com MNR Mining

As relações de implicação com antecedente e consequente unitários são suficientes para a detecção da maior parte dos padrões e, mesmo cascatas com sequências maiores que 2 são destacadas com a estratégia de visualização utilizada, como já foi mencionado. Entretanto, há relações que não podem ser reveladas apenas com o PrefixSpan.

No trabalho de KORDIC et al. [13], os autores escolheram o algoritmo de mineração de regras de associação *FP-growth* [27]. *FP-growth*é um algoritmo de mineração de padrões frequentes, cuja definição formal é a seguinte:

Padrão frequente: seja um conjunto de itens $I = \{a_1, a_2, \ldots, a_m\}$ e uma base de dados de transações (*transaction database*) $DB = T_1, T_2, \ldots, T_n$, onde T_i $(i \in [1 \ldots n])$ é uma transação que contém um conjunto dos itens em I . O suporte *sup* de um padrão *A*, onde *A* é um conjunto de itens, é o número de transações contendo *A* em *DB*. Um padrão *A* é frequente se o suporte de *A* é maior que o definido em um limiar *minsup*.

Pela definição de padrão frequente pode-se concluir que a ordem com a qual os itens aparecem em uma transação (a sequência temporal) não é considerada no algoritmo. No exemplo da Tabela 2, extraída do artigo que introduz o algoritmo *FP-growth* [27], os autores afirmam que há dois conjuntos de itens idênticos nas transações 100 e 500, embora a sequência em que eles ocorram nas transações seja diferente (*"f,a"* e *"a,f"*, por exemplo, são tratados na ordem lexicográfica pelo algoritmo).

Tabela 2 - Exemplo sobre a desconsideração da ordem dos itens em uma transação na mineração de padrões frequentes [27]

ID da Transação	Itens comprados	Itens frequentes (ordenados)
100	f, a, c, d, g, i, m, p	f, c, a, m, p
200	a, b, c, f, l, m, o	f, c, a, b, m
300	b, f, h, j, o	f, b
400	b, c, k, s, p	c, b, p

500	a, f, c, e, l, p, m, n	f, c, a, m, p

Uma outra dificuldade ao utilizar a versão original do algoritmo *FP-growth* [27] é que, naturalmente, o valor de confiança não é calculado, já que este também não é um algoritmo que gera regras de associação. Para contornar esta dificuldade, KORDIC et al. [13] optaram também pelo cálculo manual da confiança. Segundo os autores, "calcular a confiança não é difícil, considerando que a sequência de alarmes é segmentada em transações associadas a cada TAG específica". No entanto, no artigo de KORDIC et al. [13], o cálculo é feito de maneira simplista e pouco abrangente, calculando apenas as regras de associação com antecedentes unitários.

Em resumo, a abordagem de KORDIC et al. [13] não é considerada adequada pelas seguintes razões:

- O algoritmo *FP-growth* não é capaz de calcular regras de associação e, portanto, fornece apenas o suporte para os padrões descobertos, que não serve com indicador para a força das relações entre os alarmes;
- A ordem temporal de ocorrência dos alarmes não é preservada em nenhum momento, o que pode ser considerado um desperdício de uma informação já disponível;
- O cálculo manual da confiança considera regras de associação com antecedentes unitários, enquanto os algoritmos de mineração de regras de associação geram diversas combinações de antecedentes-consequentes. Como resultado, diversos tipos de regras não são calculadas.

Com estes argumentos, fica evidente que a adoção de um algoritmo de mineração de regras de associação é apropriada. Para prosseguir, a definição formal do conceito é apresentada a seguir:

Regra de Associação: uma definição mais completa de regra de associação é dada pelo autor original AGRAWAL et al. em [21]. Seja $I = I_1; I_2; ...; I_m$ um conjunto de atributos binários, chamados de itens. Seja *T* uma base de dados de transações. Cada transação *t* é representada por um vetor binário com *t[k] = 1* se o item *Ik* está presente e *t[k] = 0* se está ausente. Existe uma tupla na base de dados para cada transação. Seja *X* um conjunto de alguns itens de *I*. É dito que a transação *t* satisfaz *X* se para todos os items *Ik* em *X*, *t[k] =1*. Uma regra de associação é uma implicação na forma $X \Rightarrow Ij$, onde *X* é um conjunto de alguns itens de *I*, e *Ij* é um único item em *I* que não está presente em *X*. A regra $X \Rightarrow Ij$ é satisfeita no

conjunto de transações *T* com um fator de confiança *0≤conf≤1* se pelo menos *conf*% das transações em *T* que satisfazem *X* também satisfizerem *Ij*.

Mineração de Regras de Associação: em [21], o problema de mineração de regras de associação é dividido em dois subproblemas:

1) Gerar todas as combinações de itens que possuem valor de suporte, em termos de quantidade de transações, acima do limiar de corte *minsup*;
2) Para um dado conjunto de itens (itemset) $Y = I1I2 \ldots Ik$, $k \geq 2$, gerar todas as regras (até máximo de *k* regras) que usem os itens do conjunto *I1; I2; ...; Ik*. O antecedente de cada uma destas regras será um subconjunto *X* de *Y* tal que *X* tem $k - 1$ itens, e o consequente será o item $Y - X$. Para gerar a regra $X \Rightarrow Ij$ com confiança *conf*, onde $X = I1I2 \ldots I_{j-1}I_{j+1} \ldots Ik$, pega-se o suporte de *Y* de divide-se pelo suporte de *X*. Se o valor for superior a *minConf* então a regra é satisfeita com confiança *conf*.

Um dos desafios na mineração de dados mencionados por HAN et. al. [10] é gerar padrões mais significativos, reduzindo a quantidade de padrões sinônimos. A redundância de resultados já foi apresentada por vários autores, incluindo o trabalho de HAN et al. em [10] que dá um panorama geral das soluções disponíveis. Uma forma comum para reduzir a redundância presente nos padrões frequentes minerados está nos conceitos de padrão fechado e padrão maximal, descritos a seguir a partir do próprio trabalho de HAN et. al..

Padrão fechado (*closed pattern*): um padrão α é um padrão frequente fechado em um conjunto de dados D se α é frequente em D e não existe nenhum super padrão β tal que β tenha o mesmo suporte que α em D.

Padrão maximal: um padrão α é um padrão frequente maximal (ou *max-pattern*) em um conjunto D se α é frequente e não existe super padrão β tal que $\alpha \subset \beta$ e β é frequente em D.

Para o mesmo limiar *minsup*, o conjunto de padrões frequentes fechados contém todas as informações em relação aos padrões frequentes correspondentes, ou seja, trata-se de uma informação sem redundância. Já um conjunto de padrões maximais, embora mais compacto, usualmente não contém todas as informações representadas por seus padrões frequentes correspondentes, ou seja, trata-se de uma informação resumida.

O conceito de Regras de Associação Mínimas Não Redundantes (MNR) foi proposto por Kryszkiewicz et al. em [28] e é tido como uma das formas mais sucintas de representação das regras de associação sem perda de informação. A figura Figura 22 de

LASZLO, 2006 [29], apresenta um comparativo do volume de informação gerada por cada uma das abordagens.

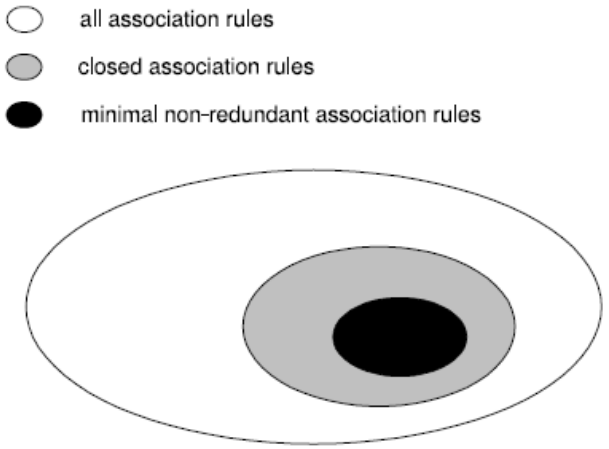

Figura 22 - Posição das Regras Fechadas e MNR diante de Todas as Regras de Associação. LASZLO, 2006 [29]

Para melhor compreender o significado das regras mínimas não redundantes, as definições apresentadas em [29] serão apresentadas a seguir:

Regra de Associação Fechada: uma regra de associação X⇒Y é fechada se X∪Y é um padrão fechado.

Regra de Associação Exata: uma regra é dita exata se a sua confiança for igual a 1, ou seja, *sup*(A∪B)=*sup*(A). Caso contrário, a regra é dita **aproximada**.

Geradores (*generators*): considere uma regra X1⇒Y, onde X1⊂ Y. Se X1⇒Y é uma regra válida, então Y precisa ser um *itemset* frequente e X1, que é um subitem de Y, precisa ser frequente também. Então, qualquer regra no formato X2⇒Y, onde X1 ⊆ X2 ⊂ Y é válida também. Por exemplo, sabendo que a regra {ab}⇒{cd} é válida, pode ser deduzido que {abc}⇒{d} e {abd}⇒{c}. Isto nos mostra que, quanto menor o tamanho da condição (antecedente) de uma regra de associação na forma X1⇒Y, mais pode-se deduzir regras válidas da forma X2⇒Y, com X1 ⊆ X2 ⊂ Y. Neste caso, o mínimo lado esquerdo (antecedente) de uma regra de associação é chamado **gerador** e o máximo lado direito (consequente) da regra corresponde ao **padrão fechado** (*closed itemset*).

Regra de Associação Mínima Não Redundante: LASZLO em [29] define uma família de 5 diferentes tipos de regras mínimas não redundantes. Três delas serão apresentadas a seguir para o entendimento da definição de MNR:

1) **GB (*generic basis*) – base genérica para regras de associação exatas**: seja FC o conjunto de padrões frequentes fechados (frequent closed itemsets). Para cada padrão fechado frequente f, seja FGf o geradores frequentes de f. GB pode ser definido através da equação abaixo **(Eq 3.3)**:

$$GB = \{r : g \Rightarrow f \mid f \in FC \wedge g \in FGf \wedge g \neq f\} \quad \textbf{(Eq 3.3)}$$

2) **IB (*informative basis*) – base informativa para regras de associação aproximadas**: seja FC o conjunto de padrões frequentes fechados (*frequent closed itemsets*) e FG cojunto de geradores frequentes. A notação $\Upsilon(g)$ denota a propriedade de g de ser um padrão fechado. IC pode ser definido através da equação abaixo **(Eq 3.4)**:

$$IC = \{r : g \Rightarrow f \mid f \in FC \wedge g \in FG \wedge Y(g) \subset f\} \quad \textbf{(Eq 3.4)}$$

3) **MNR – regras de associação mínimas não redundantes**: as regras de associação mínimas não redundantes podem ser definidas da seguintte forma **(Eq 3.5**):

$$MNR = GB \cup IC \quad \textbf{(Eq 3.5)}$$

A definição de MNR [28] [29] estabelece apenas que as regras possuem a forma $X \Rightarrow Y$, onde $X \subset Y$ e X é um gerador e Y é um padrão frequente fechado. Em outras palavras, uma definição mais simples para uma regra **MNR é a regra que possui o mínimo antecedente e o máximo consequente**.

Para gerar as regras de associação mínimas não redundantes basta ter os padrões (*itemsets*) frequentes fechados e os correspondentes geradores mínimos. Ainda segundo [29], do ponto de vista da aplicação, a representação mais útil dos padrões é dada pelos padrões frequentes fechados e os geradores frequentes, constituindo uma representação concisa e sem perda de informação de todas as regras ou episódios de interesse.

Para descobrir as regras de associação complexas, de tamanho superior a dois, utilizando o conceito de regras de associação mínimas não redundantes, foi utilizado o algoritmo ZART por LAZLO et. al. [30]. ZART é um algoritmo multifuncional capaz de identificar padrões frequentes fechados (*frequent closed itemsets*), levantar os geradores

(*generators*) e associar os geradores aos padrões fechados correspondentes, o que é suficiente para encontrar as regras de associação mínimas não redundantes. O algoritmo enumera os *itemsets* em ordem ascendente de acordo com o tamanho, seguindo uma estratégia de busca em largura.

3.4.4 Visualização

Assim como em qualquer atividade de mineração, uma das etapas mais importantes do processo de descoberta de padrões de alarmes redundantes é a interpretação dos padrões que permite que as ações de supressão possam ser planejadas.

A estratégia de visualização aqui apresentada pode ser considerada um diferencial em relação a todas as abordagens anteriores das referências relacionadas ao contexto de racionalização de alarmes. Após a descoberta dos padrões pelos algoritmos de correlação cruzada, PrefixSpan e ZART [30], os resultados de cada algoritmo são utilizados na modelagem de duas redes complexas, que são dois grafos dirigidos, onde as seguintes estratégias são adotadas:

- **Vértices no Grafo**: no grafo de relações simples, um padrão frequente é dado por uma relação com antecedente e consequente unitários. Neste caso, cada vértice do grafo representa um código (TAG) de alarme diferente (ex. “A” ou “B”). No grafo das relações complexas, em que cada padrão pode possuir antecedentes e consequentes maiores que um, cada vértice pode representar um antecedente ou um consequente seguindo várias combinações de TAGs. No padrão AB⇒CDE, por exemplo, dois vértices são criados: (AB) e (CDE);
- **Modelagem do Suporte**: o suporte representa a significância estatística e, dentro do contexto de alarmes, isso possui um significado particularmente interessante, já que o objetivo é reduzir a quantidade de alarmes gerados e, portanto, quanto maior o suporte de cada alarme possivelmente suprimível, melhor. Por esta razão, o suporte de cada grupo antecedente/consequente foi modelado como o tamanho do vértice no grafo;
- **Modelagem da Confiança**: a confiança das regras representa a força da relação e, portanto, esta intensidade foi representada através da espessura de arestas nos grafos. Assim, uma relação com confiança igual a 1, será

representada com a maior espessura possível, e a relação com confiança próxima de, será representada com uma espessura menos espessa. Regras com confiança 0 não são representadas;

- **Orientação das Arestas pela Precedência da Regra de Associação**: a orientação das arestas nos grafos é feita através da definição de antecedente e consequente das regras de associação descobertas. Assim, uma relação do tipo A⇒B será representada através de uma aresta ligando o vértice A ao vértice B, saindo de A e entrando em B. No grafo das relações simples, como a orientação temporal é preservada, o significado desta orientação pode ser estendido à relação de precedência temporal de A e B;
- **Representação do Resultado da Correlação**: no grafo das relações simples, as regras de associação são representadas por arestas na cor verde. As regras resultantes da análise de correlação cruzada também são representadas no grafo (na cor vermelha), permitindo que o usuário utilize as duas informações complementares para gerar suas conclusões sobre o significado do padrão no processo. Com a representação mútua, mesmo que uma ou outra alternativa seja menos eficiente na detecção do padrão, a informação sobre a relação verdadeira é sempre apresentada;
- **Anotações para os Detalhes da Regra**: cada aresta é também anotada com os detalhes correspondentes da regra, seguindo o seguinte padrão:
    - Para correlação (aresta vermelha): {X [ρ] [*s*] [*t*]}, onde ρ é o coeficiente de correlação de Pearson correspondente ao maior valor encontrado para a correlação entre os dois alarmes, *s* é o sinal do deslocamento (shift) da janela para o maior valor de correlação encontrado e *t* é a defasagem temporal correspondente ao deslocamento.
    - Para regra de associação (aresta verde): {CFN [α] RTN [β]}, onde α é o valor de confiança na relação para a mineração sobre a base de alarmes ordenada por CFN e β é o valor de confiança na relação para a mineração sobre a base de alarmes ordenada por RTN;
- **Representação da Regra Mais Forte entre CFN/RTN:** embora possam existir valores diferentes para a confiança na mineração de CFN e RTN, para simplificar a visualização, apenas uma aresta é desenhada, sendo a sua espessura determinada para o maior valor encontrado. As anotações

possuem sempre os dois valores, permitindo uma investigação detalhada, se necessária, através do próprio grafo;

A Figura 23 apresenta um exemplo de visualização do resultado dos padrões simples. No layout circular, selecionado dentre várias opções disponíveis no software Pajek [32], os vértices são modelados na periferia e anotados com o número inteiro correspondente ao TAG. Podemos ver facilmente, por exemplo, que os alarmes mais frequentes não são os participantes da maior parte das relações mais significativas (de maior confiança). Podemos ver também o claro destaque dos padrões mais significativos na diferença de intensidade das arestas. As anotações são ativadas ou desativadas sob demanda, sendo possível ao usuário mostrá-las apenas durante uma análise detalhada (após o corte dos padrões menos significativos).

A Figura 24 apresenta um exemplo de visualização dos detalhes de um padrão entre os TAGs 65 e 66. Daí pode-se concluir que:

a) **Precedência**: 65 precede 66 no tempo, ou seja, 65→66;
b) **Tipo de relação temporal**: esta precedência é mais intensa no início do alarme, ou seja, sempre que 65 é iniciado, 66 é iniciado também, mas nem sempre que 65 retorna à normalidade, 66 retorna também. Assim, pode-se concluir que esta é uma relação do tipo XSY. Como não há arestas no sentido contrário, pode-se concluir que 65 sempre precede 66;
c) **Correlação**: pode-se concluir que o maior valor de correlação foi obtido sem deslocamento temporal (sem *shift* de janela ou com *shift* zero) com coeficiente de correlação igual a 0,68, indicando que, possivelmente, em 68% do tempo 65 e 66 estão ativos simultaneamente.

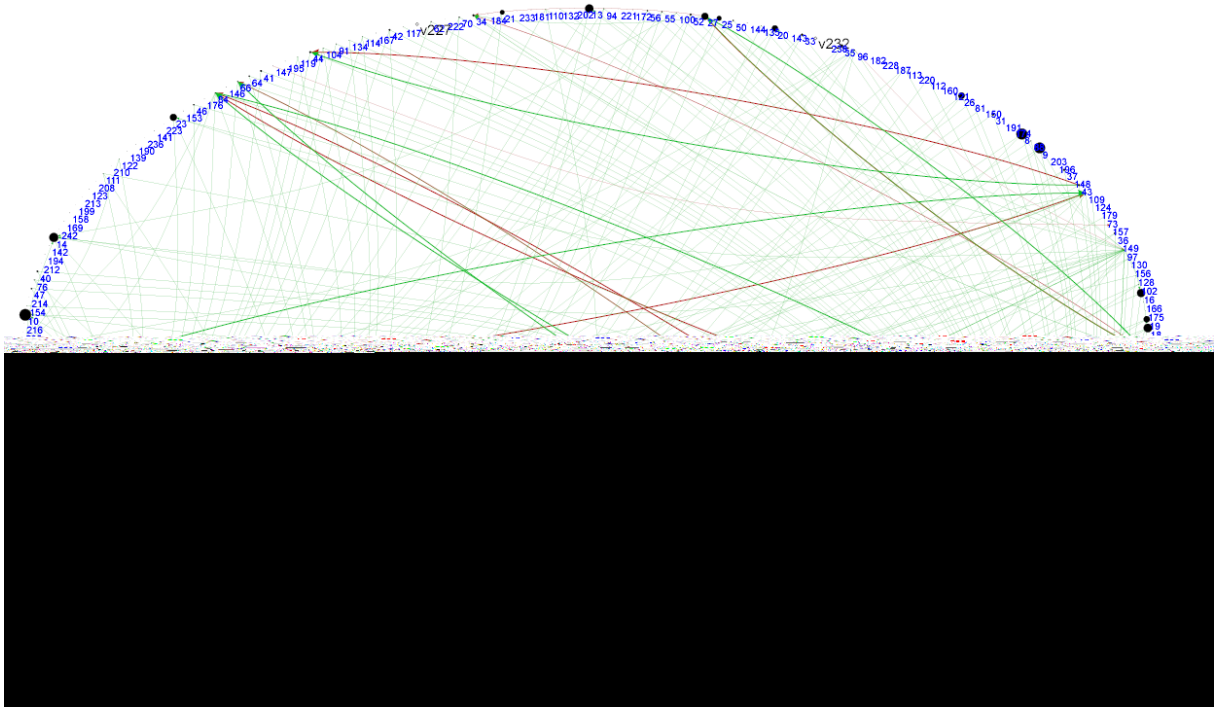

Figura 23 - Exemplo da visualização de um conjunto de padrões: a espessura das arestas indica a força da relação encontrada e o tamanho dos vértices (que representam os tipos diferentes de alarmes), representam a quantidade relativa deste alarme no conjunto de dados (suporte).

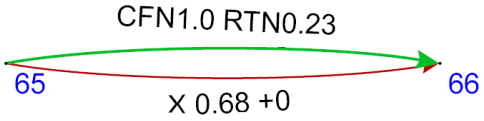

Figura 24 - ***Exemplo de visualização de um padrão: existe uma relação de precedência com força 100% (1.0 na aresta de cor verde) que indica que após a ocorrência do alarme código 65, o alarme de código 66 também é iniciado (CFN). A aresta de cor vermelha representa o valor de correlação máximo encontrado, no deslocamento de 0 segundos de 0.68.***

4 RESULTADOS EXPERIMENTAIS

Os resultados foram avaliados em duas etapas: resultados da mineração da base de dados sintética e da mineração da base de dados real da Britagem, utilizada na etapa de caracterização de carga. Cada uma das etapas é descrita a seguir.

4.1 Resultados na Mineração da Base de Dados Sintética

Este experimento consistiu na avaliação da capacidade que cada algoritmo possui de detectar cada um dos padrões artificiais na base de dados sintética. Seis tipos diferentes de padrões temporais estavam presentes nesta base de dados de 250 mil registros, gerada artificialmente a partir da caracterização da base de dados real da Britagem com todos os tipos de padrões temporais possíveis segundo Allen.

A eficiência dos algoritmos foi medida em relação à capacidade de detectar que um padrão inserido artificialmente possui força igual a 1.0. O desvio do valor informado pelos resultados para um determinado padrão, em relação a este valor é considerado como erro. Por exemplo, se uma relação temporal XEY perfeita foi inserida, mas esta relação foi detectada com maior confiança igual a 0,9, então o erro calculado é de 1,0 -0,9 = 0,1. Considerando que a abordagem *cross-effect-test* de KORDIC et al. [13] é a mais recente e adequada (até então), do ponto de vista de aplicabilidade prática, a comparação do novo algoritmo foi feita considerando esta abordagem como referência.

Duas análises foram feitas: a primeira comparando a capacidade de detecção da mineração dos padrões com PrefixSpan isoladamente (sem a junção dos resultados da análise de correlação cruzada), e a segunda análise utilizando os resultados combinados PrefixSpan+*Cross-correlation*. Para possibilitar a comparação, valores mínimos de suporte e confiança foram estipulados (para não haver cortes significativos), respectivamente iguais a 0,0001 e 0,001. Para a análise de correlação cruzada, o parâmetro *window size* foi fixado em 10 segundos.

Considerando que o algoritmo *cross-effect-test* possui um parâmetro que pode influenciar nos resultados, seguindo a recomendação dos autores [13], ao invés de uma execução apenas, várias execuções foram feitas, variando o valor do parâmetro *minw* entre 10 e 500 segundos, que é o valor da maior duração existente na base de dados. Para cada

execução, a média de erros era computada, seguindo o raciocínio de cálculo do erro mencionado acima. A Figura 25 apresenta o valor médio dos erros. O valor de parâmetro *minw* do melhor resultado (com menor erro), de 200s, foi utilizado na comparação dos métodos.

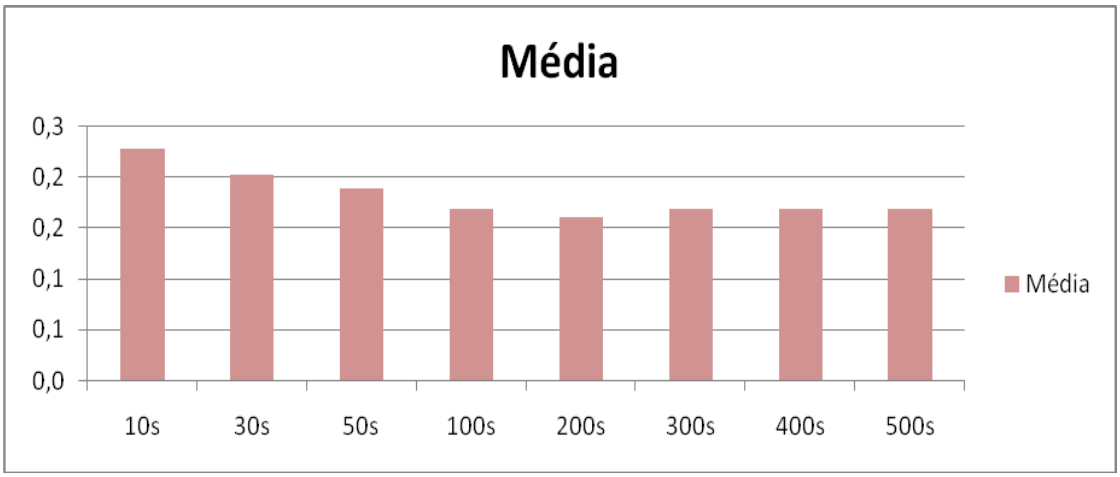

Figura 25 - *Variação do parâmetro minw no algoritmo cross-effect-test para detectar o valor de menor erro*

Uma constatação possível, a partir da Figura 25 é a grande variação dos resultados da abordagem *cross-effect-test* em relação à variação do parâmetro *minw*, o que comprova o argumento de que os resultados podem sofrer grandes variações com erros grandes em caso de falta de conhecimento do usuário minerador.

A Figura 26 apresenta o gráfico dos erros encontrados na comparação do algoritmo *cross-effect-test* com a nova abordagem. Os valores do eixo Y podem variar entre 0 e 1 e indicam o erro em relação ao grau de confiança máximo. É possível observar que, com exceção do padrão YOX (Y *overlapped by* X), em todos os demais tipos de padrões temporais, a nova abordagem resultou em erros menores. O padrão YOX é exatamente o tipo de padrão que o algoritmo *cross-effect-test* busca, razão pela qual os erros foram ligeiramente inferiores. Entretanto, apesar de haver detectado os demais padrões, ainda que com erro superior, tal comportamento poderia ser considerado um efeito colateral (*side effect*), já que o algoritmo não possui este propósito.

Outra conclusão que pode ser obtida é a capacidade limitada das duas abordagens de detectar o tipo de relação YMX (Y *met by* X), na qual o início de Y ocorre simultaneamente com o término de X. Uma possível justificativa para este problema

encontra-se no fato de que o menor tempo de intersecção entre X e Y, e consequente aumento do tempo de ocorrência isolada de um dos dois alarmes, faz com que aumente a probabilidade de que X ou Y se relacionem a outros alarmes de maneira independente, o que reduz a confiança.

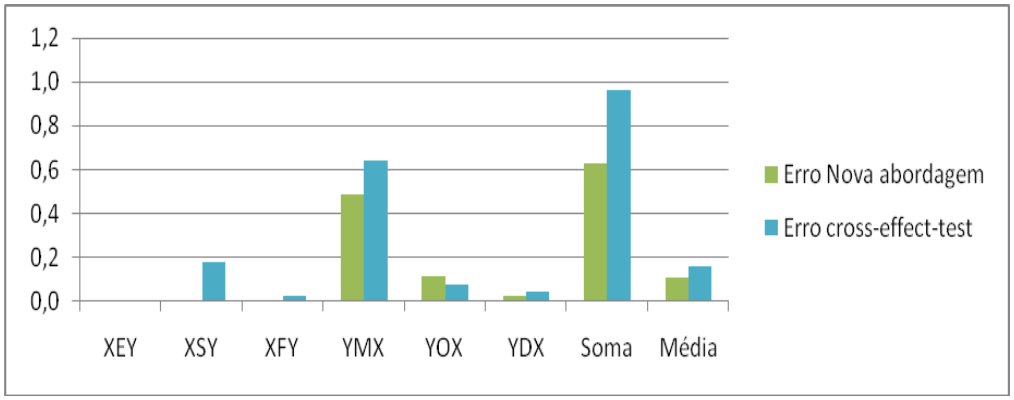

Figura 26- ***Erros de cross-effect-test vs nova abordagem em relação a cada padrão artificial inserido (sem cross-correlation. Eixo X corresponde aos erros encontrados e varia de 0 a 1 para cada um dos tipos e para a média e, de 0 a 6 para a Soma dos erros.***

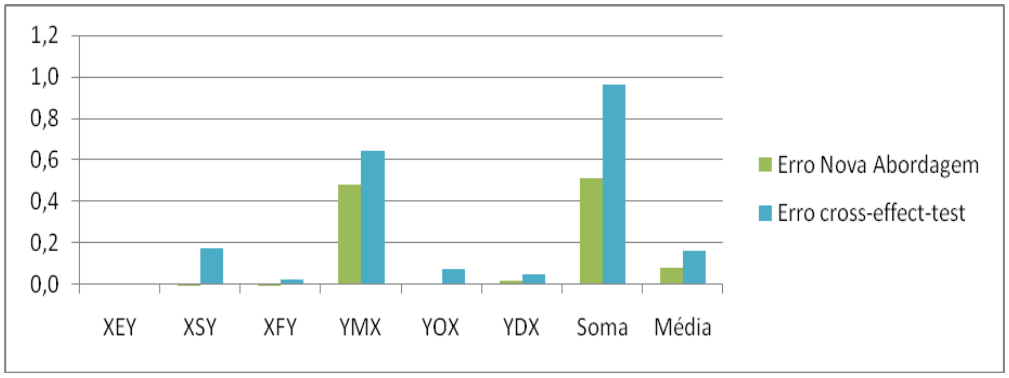

Figura 27 - ***Erros de cross-effect-test vs nova abordagem em relação a cada padrão artificial inserido (cross-correlation+PrefixSpan)***

A Figura 27 apresenta o gráfico de comparação dos erros para cada padrão quando a abordagem combinada dos algoritmos PrefixSpan e *Cross-correlation* é utilizada. A comparação é feita utilizando sempre o maior valor de confiança atribuído pelos dois algoritmos, já que, visualmente, o usuário é capaz de ver simultaneamente os dois resultados. Neste caso, podemos ver que o erro na detecção dos padrões foi menor em todos os tipos de padrões temporais, incluindo o YOX. Todos os padrões que possuem

correlação perfeita, ainda que deslocada no tempo, são detectados sem erro quando as abordagens são combinadas, diferentemente da abordagem *cross-effect-test*.

Para generalizar os resultados a uma população distribuída supostamente seguindo uma distribuição normal, um teste estatístico de amostras pareadas, descrito em [25], foi executado. Este teste pode ser descrito de acordo com o seguinte procedimento:

1. Para cada par de amostras (ex. erro XSY da abordagem antiga e erro XSY da abordagem nova) é computada a diferença A-B;
2. O valor da média $\overline{x}$ é calculado para todas estas diferenças;
3. O desvio padrão S é calculado para estas diferenças;
4. Utilizando o desvio padrão e o número de amostras, o intervalo de confiança para esta média é calculado;
5. O resultado do intervalo de confiança é avaliado:
    a. Caso o intervalo **contenha o valor zero**, **não se pode afirmar**, com o nível de confiança utilizado no cálculo, que as populações são significativamente diferentes;
    b. Caso o intervalo **não contenha o valor zero**, **então pode-se afirmar**, com o nível de confiança utilizado no cálculo, que as populações são significativamente diferentes;

Para este cálculo, foi utilizada a distribuição t de *student*, com intervalos de confiança definidos segundo a equação abaixo (Eq 4.1):

$$\overline{x} \pm t_{\left[1-\alpha/2;n-1\right]}\left(\frac{S}{\sqrt{n}}\right) \qquad \textbf{(Eq 4.1)}$$

Os resultados do cálculo são apresentados abaixo na Tabela 3:

Tabela 3- Cálculo do Intervalo de Confiança para a Média das Diferenças na Amostra Pareada

$\overline{x}$	0,0756
S	0,0731
n	6

(número de amostras)	
Lim.superior IC	0,1357
Lim.inferior IC	0,0154

Como o intervalo de confiança não inclui o valor 0 (zero), utilizando a distribuição t, podemos concluir que, com 90% de confiança, a nova abordagem **é superior em termos de exatidão** na detecção dos padrões artificiais com menos erros.

4.1.1 Análise dos Falsos Positivos

A análise da seção anterior nos mostrou a exatidão da nova abordagem para a descoberta dos padrões de alarmes considerando a capacidade de detectar um padrão existente, gerando valores de confiança próximos de 1. Contudo, uma outra análise que deve ser feita para entender a exatidão do algoritmo é em relação à sua capacidade de afirmar que um padrão que não existe possui valor de confiança próximo de 0. Nesta análise, padrões que não foram inseridos, mas que surgiram com valores de confiança superior a 0, são considerados falsos positivos (considerando os limiares de suporte e confiança definidos na seção anterior). Para fazer esta análise, as seguintes etapas foram executadas:

- As relações sintéticas verdadeiras foram removidas dos resultados;
- A média dos valores restantes (chamados aqui de não-padrões) foi calculada para a) *cross-effect-test*, b) análise com CFN, c) análise com RTN e d) análise com correlação cruzada;
- Os valores de média foram comparados e estatisticamente verificados.

A Figura 28 mostra a comparação de médias dos não-padrões para os resultados do algoritmo *cross-effect-test*, da análise sobre RTN, para análise sobre CFN e para a análise da correlação cruzada (representadas nas arestas pela anotação “Xcorr”). Duas conclusões que podem ser obtidas apenas com a interpretação direta destes resultados é que, a) considerando erros de 0 a 1, a média dos erros pode ser considerada baixa e b) os erros para a análise de correlação cruzada são menores.

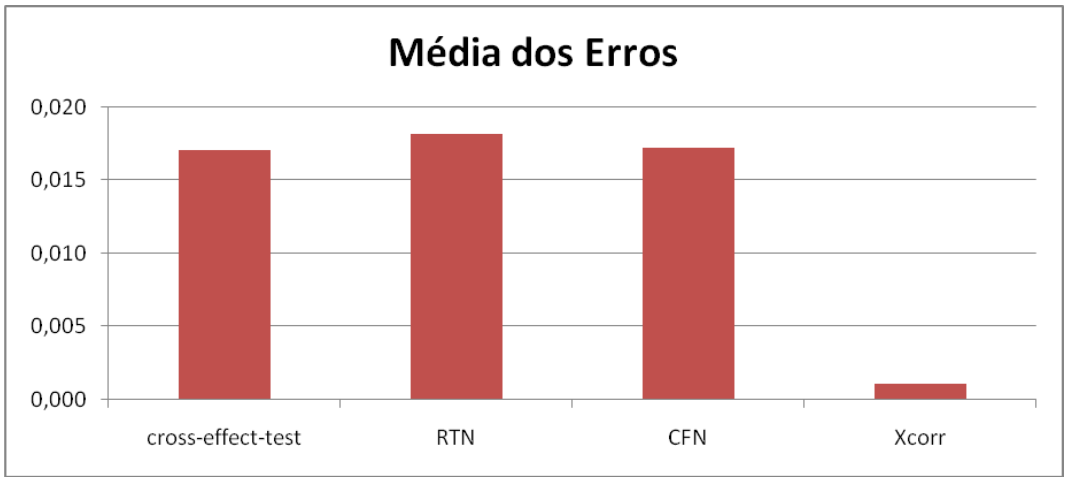

Figura 28 - Média dos Erros para os Não-padrões

Considerando que na apresentação visual, sempre os três resultados são apresentados ao usuário (RTN, CFN e XCorr), o maior erro possível a ser considerado na comparação é, simplesmente, o maior erro encontrado dentre as três abordagens. Para comparar se a média dos falsos positivos da abordagem *cross-effect-test* é significativamente diferente (e menor) da média da nova abordagem, portanto, temos que comparar se a média de *cross-effect-test* é significativamente menor da média da análise de RTN, que é o maior encontrado entre RTN, CFN e XCorr.

Para realizar a comparação, o teste para amostras não pareadas descrito em [25] foi utilizado. Este teste é resumidamente descrito abaixo:

1. O valor da média $\overline{x}$ é calculado para os erros dos não padrões para a abordagem antiga e abordagem nova;
2. O desvio padrão S é calculado para cada conjunto de dados (para abordagem antiga e nova abordagem);
3. Utilizando o desvio padrão e o número de amostras, o intervalo de confiança para cada média é calculado (gerando um limite superior e limite inferior para o intervalo de confiança);
4. Os intervalos de confiança das abordagens antiga e nova são confrontados:
    a. Se não houver sobreposição nos intervalos, pode-se afirmar que as médias são significativamente diferentes e, portanto, as populações

são diferentes. Neste caso, a maior média indica a população com valor maior (neste caso valor maior de erros);

b. Se houver sobreposição e cada IC contiver a outra média, então os algoritmos não são diferentes;

5. Se houver sobreposição e cada IC não contiver a outra média, então é necessário realizar os cálculos do teste-t de *student*;

a. Se o intervalo de confiança do teste-t contiver zero então não há diferença significativa;

b. Se o intervalo de confiança do teste-t não contiver zero então há diferença significativa e, novamente, a maior média indica a população com valor maior para erro médio.

Os intervalos de confiança das médias foram calculados utilizando a distribuição normal unitária Z (Eq 3.1), que gera como resultado um limite inferior e um limite superior do intervalo. Os resultados para o nível de significância de 90% são apresentados na Tabela 4.

$$\bar{x} \pm z_{1-\alpha/2}\left(\frac{s}{\sqrt{n}}\right) \qquad \textbf{(Eq 4.2)}$$

Tabela 4 - Cálculo dos Intervalos de Confiança para as Médias da Confiança dos Não-Padrões de Cross-effect-test e resultados da análise de RTN

cross-effect-test	$\bar{x}$	0,01709
	s	0.03423
	n	3968
	Lim.superior IC	0,01799
	Lim.inferior IC	0,01620
RTN	$\bar{x}$	0,01821
	s	0,03261

	n	4804
	Lim.superior IC	0,01898
	Lim.inferior IC	0,01743

Podemos perceber que o resultado do cálculo dos intervalos de confiança se enquadra na situação 4, sendo, portanto, necessário realizar os cálculos do teste-t de *student*. As equações utilizadas são apresentadas abaixo (Eq 4.3), (Eq 4.4), (Eq 4.5) e os resultados do teste-t na Tabela 5.

$$\left(\overline{x}_a - \overline{x}_b\right) \mp t_{[1-\alpha/2;\nu]} s \qquad \textbf{(Eq 4.3)}$$

$$\nu = \frac{\left(s_a^2 / n_a + s_b^2 / n_b\right)^2}{\frac{1}{n_a - 1}\left(\frac{s_a^2}{n_a}\right)^2 + \frac{1}{n_b - 1}\left(\frac{s_b^2}{n_b}\right)^2} - 2 \qquad \textbf{(Eq 4.4)}$$

$$s = \sqrt{\frac{s_a^2}{n_a} + \frac{s_b^2}{n_b}} \qquad \textbf{(Eq 4.5)}$$

Tabela 5 - Resultados do Teste-t na comparação das Médias de *cross-effect-test* e os resultados da análise com RTN

$\overline{x}_a - \overline{x}_b$	-0,00111
s	0,0007188
Lim.superior IC	0,00006614
Lim.inferior IC	-0,0022988

Podemos concluir, com a análise dos intervalos de confiança do teste-t se enquadra na situação 4a e que, portanto, **não há diferença significativa** na média dos falsos positivos dos resultados da análise com *cross-effect-test* em relação aos resultados da análise com RTN.

4.1.2 Análise dos Padrões Encontrados

Como já foi mencionado, seis tipos de padrões temporais artificiais diferentes foram inseridos na base de dados de alarmes sintética. Adicionalmente, para avaliar a capacidade de detecção de cascatas, um padrão triplo do tipo YOX também foi inserido. A Figura 29 apresenta a rede com os padrões com corte de arestas com força inferior a 0,5, onde os sete padrões mencionados podem ser encontrados visualmente.

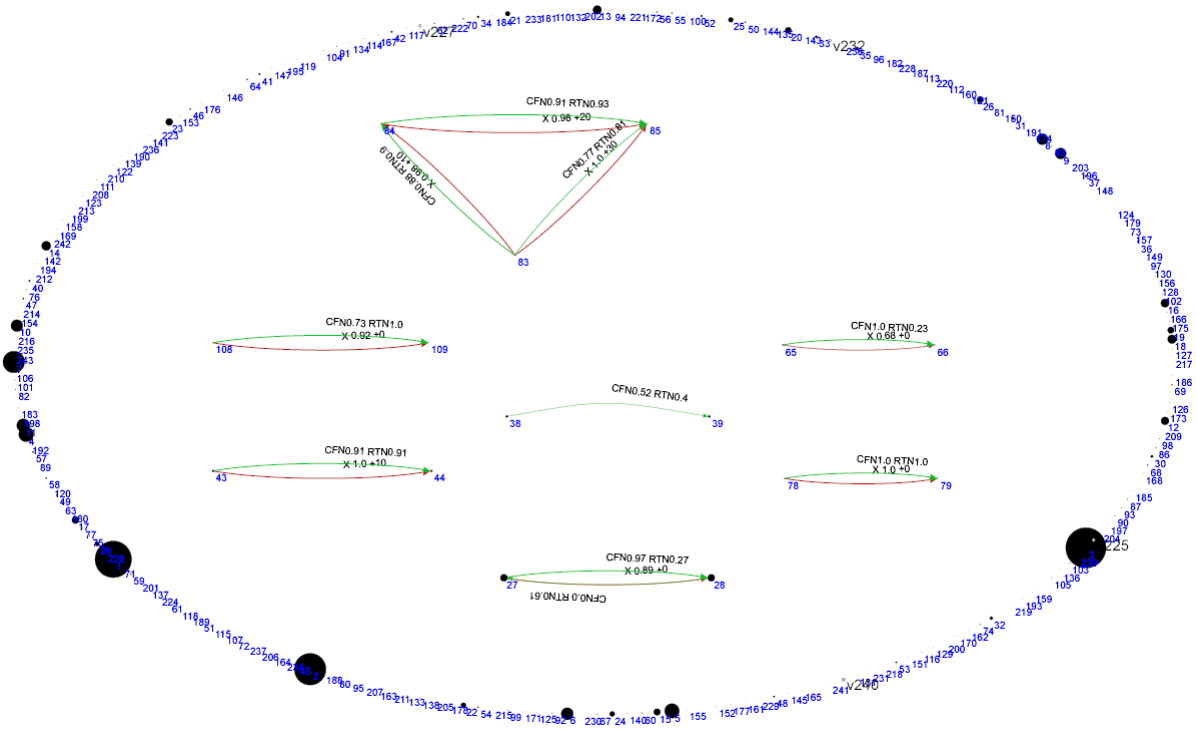

Figura 29 - Destaque dos Padrões Sintéticos Através da Separação dos Componentes Conexos no Grafo

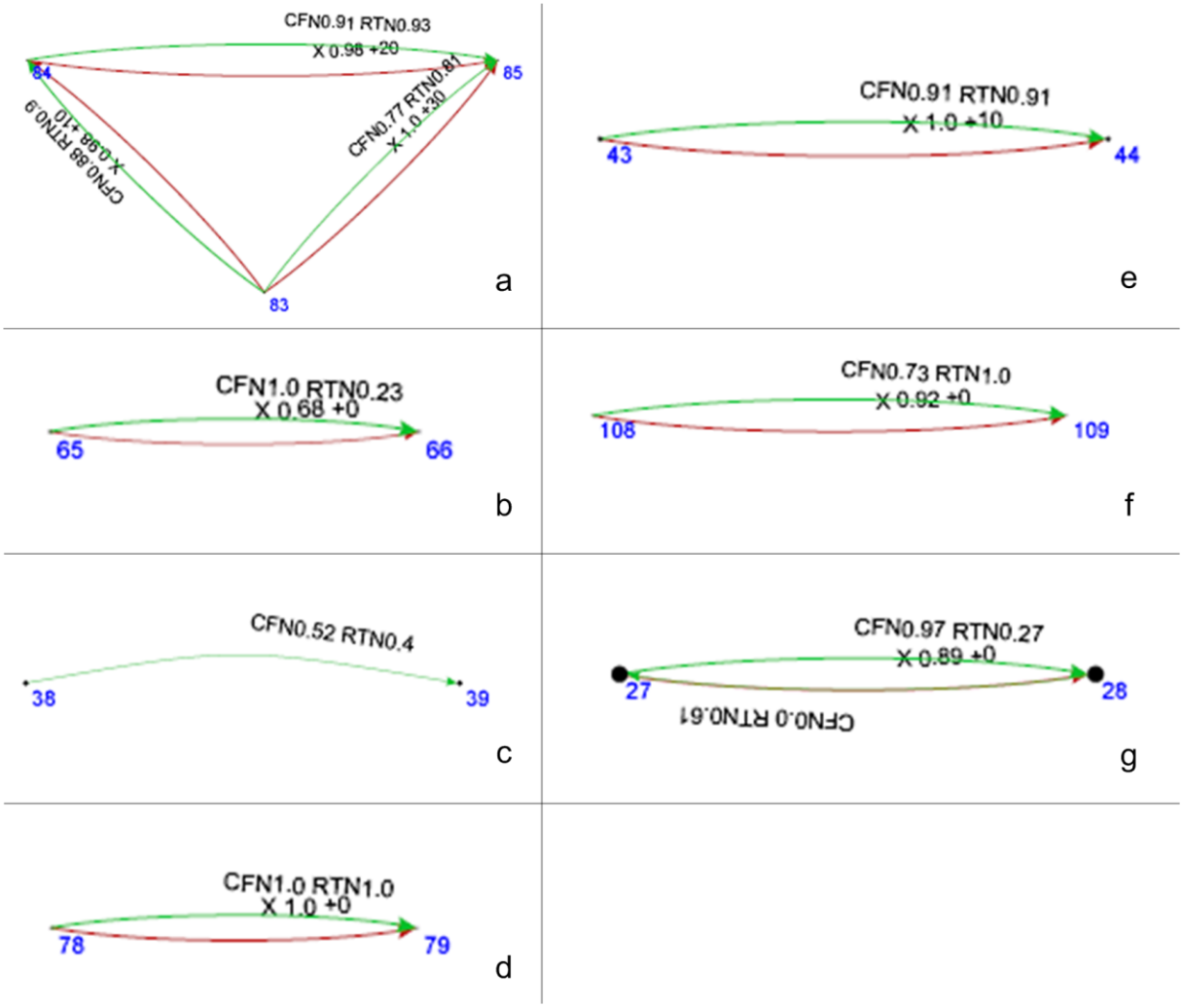

Figura 30 - Padrões Sintéticos Encontrados com Corte de Relevância Menor que 0,5

A Figura 30 apresenta o detalhamento de cada um dos padrões, com a representação visual resultante para cada um deles. A seguir, estes padrões serão descritos e relacionados aos padrões artificiais inseridos:

a. Este padrão corresponde à cascata do tipo YOX e foi implementado na sequência 83→84→85. Os três alarmes possuem durações idênticas, mas são deslocados no tempo com uma defasagem de 10 e 20 segundos, o que pode ser constatado na anotação das arestas da correlação (“+10” e “+20”). A orientação das arestas confirma a ordem temporal da relação, tanto pelas regras de associação, quanto pela correlação. A cascata também foi detectada em relação a CFN e RTN através das regras de associação (arestas verdes) com valores similares e elevados, confirmando que os

alarmes em questão ocorrem em sequência. O efeito colateral da correlação 83→85 também foi detectado corretamente com deslocamento de +30 segundos;

b. O padrão artificial criado para os TAGs 65 e 66 é do tipo XSY, em que 65→66. Os valores de duração para X e Y (65 e 66) foram escolhidos aleatoriamente, razão pela qual o valor da correlação é menor. A anotação “CFN 1.0” indica que todas as ocorrências do TAG 65 foram sucedidas pelo TAG 66, ou seja, o valor de confiança é máximo, ou ainda, trata-se de uma regra de associação “exata”;

c. A relação para os TAGs 38 e 39 é do tipo YMX. Em razão da sobreposição temporal quase nula, o algoritmo de correlação não detectou correlação significativa, sendo a regra mais forte detectada pela regra de associação. A direção da aresta representa corretamente o padrão artificial inserido;

d. O padrão criado para 78 e 79 é do tipo XEY, ou seja, uma correlação perfeita entre os alarmes. Em função de CFNs e RTNs, a regra de associação exata também expressa corretamente que o início de X é sempre sucedido pelo início de Y, o que também ocorre para o retorno;

e. O padrão para os TAGs 43 e 44 é do tipo YOX. Além do elevado valor de confiança para CFN e RTN na associação 43→44, induzindo o usuário a crer corretamente que o início de Y é precedido pelo início de X e o término de Y é precedido pelo término de X, a aresta da correlação confirma que se trata de uma correlação perfeita deslocada em 10 segundos;

f. O padrão criado para os TAGs 108 e 109 é do tipo XFY. Assim como ocorre para XSY, há um valor significativo para a correlação, determinado pela escolha aleatória das durações, corretamente representado pela aresta vermelha. A anotação e a direção das arestas confirma que a relação mais significativa é a que afirma que o retorno de X é sucedido pelo retorno de Y (“RTN 1.0”);

g. O padrão escolhido para os TAGs 27 e 28 é do tipo YDX, em que Y ocorre durante o período de ocorrência de X. Neste caso, o TAG 28 ocorre durante o TAG 27. Esta relação pode ser claramente identificada através da

probabilidade elevada para sucessão em relação a CFN (0,97) na aresta 27→28, mas também através da aresta com direção 28→27, indicando confiança elevada no retorno consecutivo nesta ordem. A correlação que ocorre como consequência da ativação simultânea também é expressa corretamente.

4.1.3 Navegação nos Padrões

Informar os valores de suporte (*minsup*) e confiança (*minconf*) com base nos padrões já minerados é muito mais intuitivo para o usuário, quando a significância destes padrões já está representada graficamente. É possível, por exemplo, investigar graficamente os alarmes mais frequentes e seus relacionamentos com a intensidade das relações temporais relacionadas a estes alarmes antes de determinar o valor de corte. O exemplo da Figura 31 mostra o que poderia ser uma atividade de navegação nos padrões em que cortes sucessivos (de 1 a 3) são feitos até tornar os padrões mais importantes visíveis e, em seguida, as anotações (em 4) são inseridas para uma análise detalhada. Da mesma forma, ao invés de efetuar os cortes através de limiares para espessura de arestas (confiança), seria possível efetuar cortes com base no tamanho dos vértices (suporte).

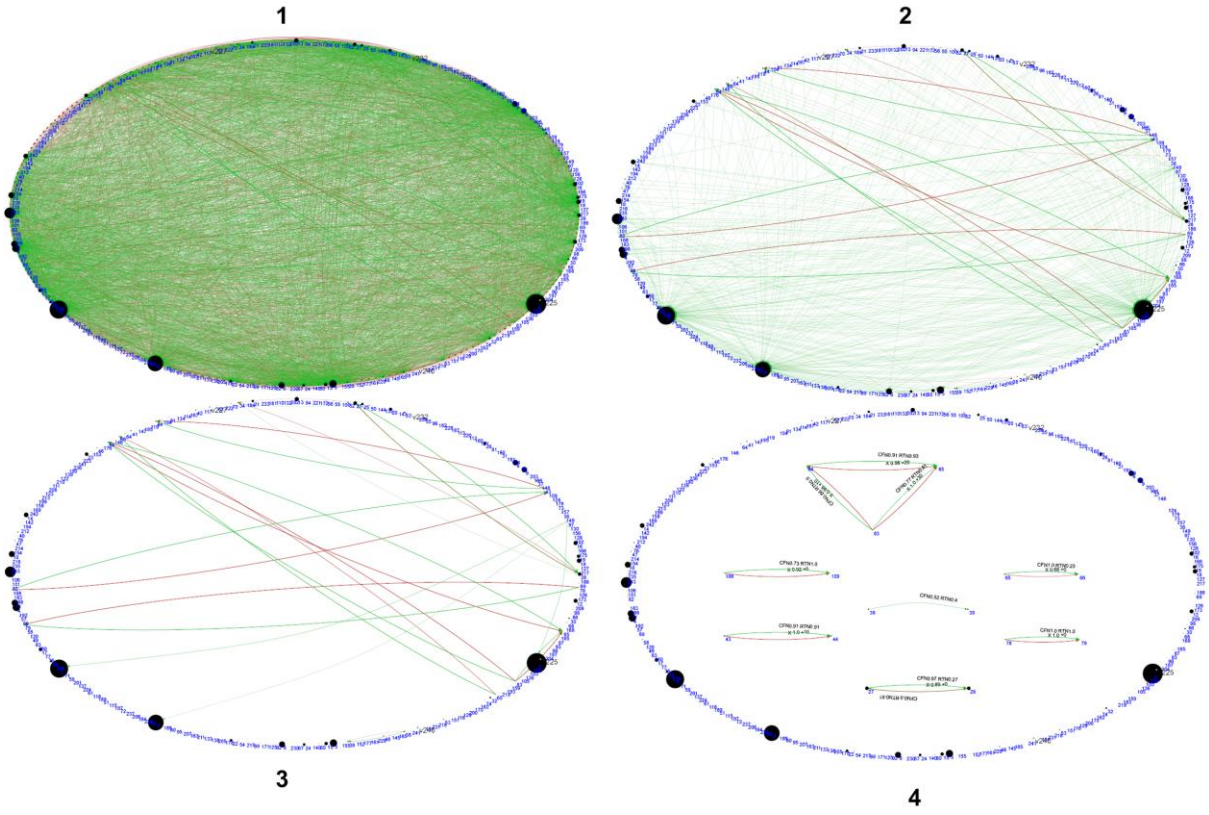

Figura 31 - Sequência de Cortes de 1 a 3 e Destaque dos Componentes Conexos em 4

A impressão dos vértices e arestas na rede complexa da Figura 31 foi feita seguindo um layout padrão do software Pajek, mas há diversas opções de formação de layout que facilitam a navegação, assim como exemplifica a Figura 32. Se somarmos este recurso com a capacidade de identificação dos componentes conexos e com a própria capacidade de interação visual com os padrões (arrastando e soltando em qualquer posição de um plano 2D ou 3D), podemos concluir que se tem à disposição um poderoso e ágil kit de mineração para a etapa de interpretação das regras mineradas.

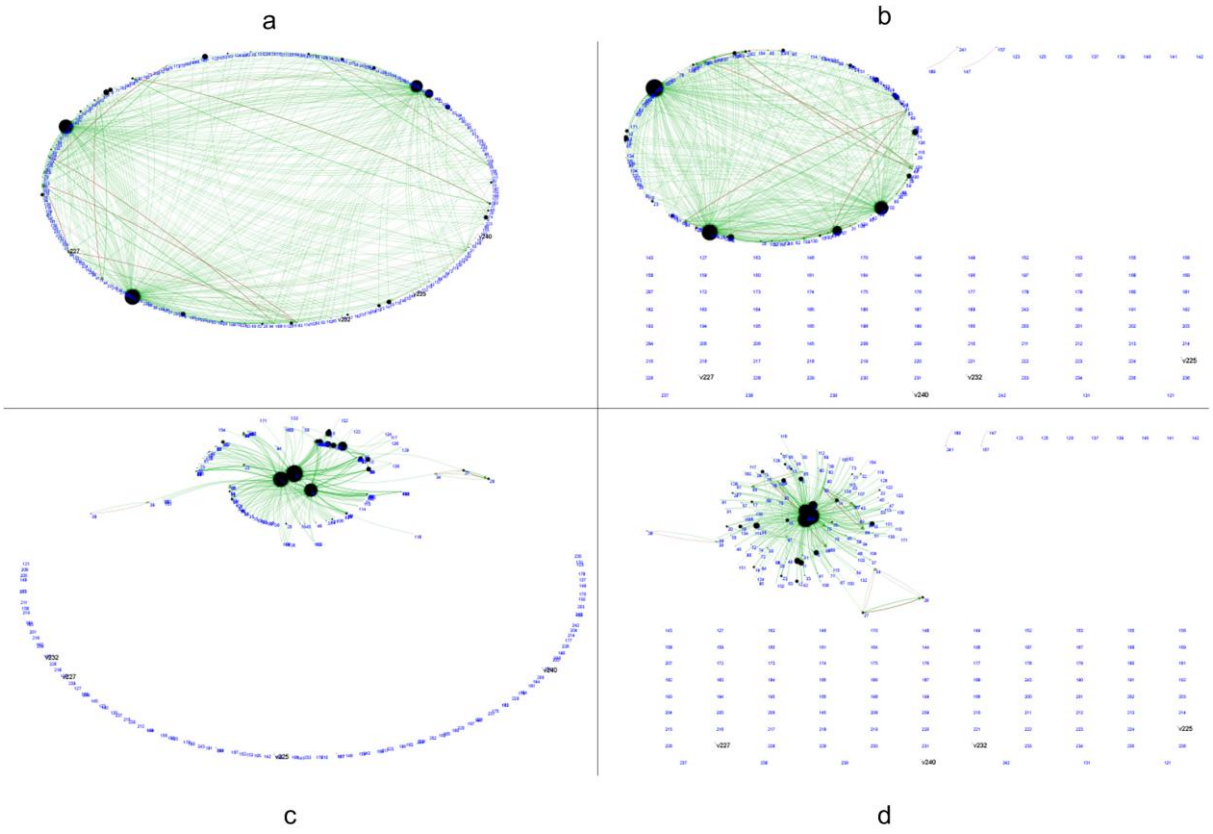

Figura 32 - Opções de Layout do Pajek para Visualização dos Padrões a) circular-random b) circular-tile-components c) energy-fruchterman-reingold-2d d) energy-kamada-kawai-separate-components (opções descritas em [32])

4.2 Resultados na Mineração da Base de Dados Real

O mesmo algoritmo e modelagem visual aplicados à base de dados sintética foram aplicados à base de dados real da Britagem[9]. A conclusão sobre a eficiência quando o algoritmo é aplicado diretamente sobre os dados reais, entretanto, não pode ser feita com precisão, já que os padrões não são conhecidos a priori. Por esta razão, a avaliação foi feita através da conferência da validade de parte dos resultados encontrados .

Como resultado, 731 relações com nível de significância superior a 0,5 foram identificadas dentre os 243 tipos diferentes de alarmes existentes na Britagem. Para nível

[9] A execução foi feita na base de dados da Britagem em função do tamanho menor da base de dados em relação à Usina. Além disso, a análise dos padrões sintéticos também foi feita sobre caracterização de carga feita sobre a base de dados da Britagem.

de significância superior a 0,9, 216 relações foram encontradas. As redes resultantes para estes valores de corte são apresentados nas figuras abaixo (Figura 33 e Figura 34).

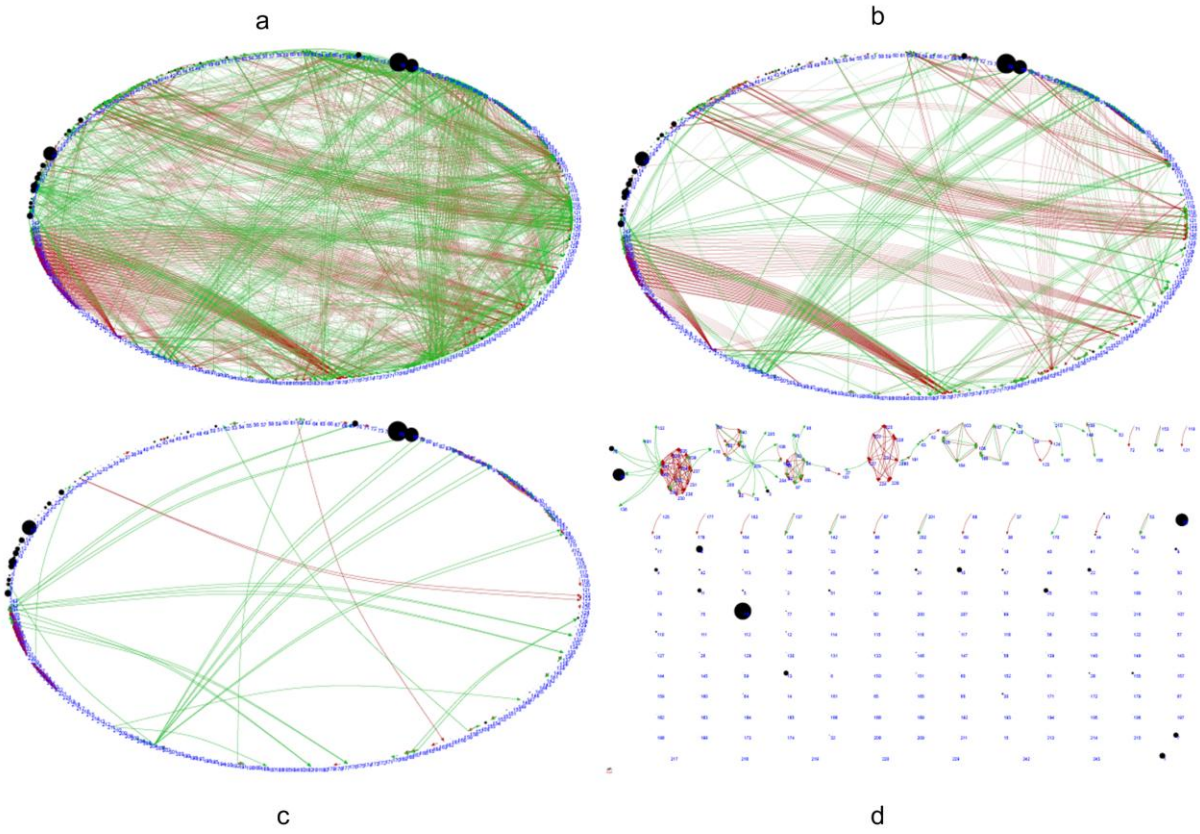

Figura 33 – Visualização dos Padrões Minerados na Base de Dados Real da Britagem para Significância Superior a 0,2 (a), 0,3 (b), 0,9 (c) e 0,9 com os Componentes Conexos Identificados (d)

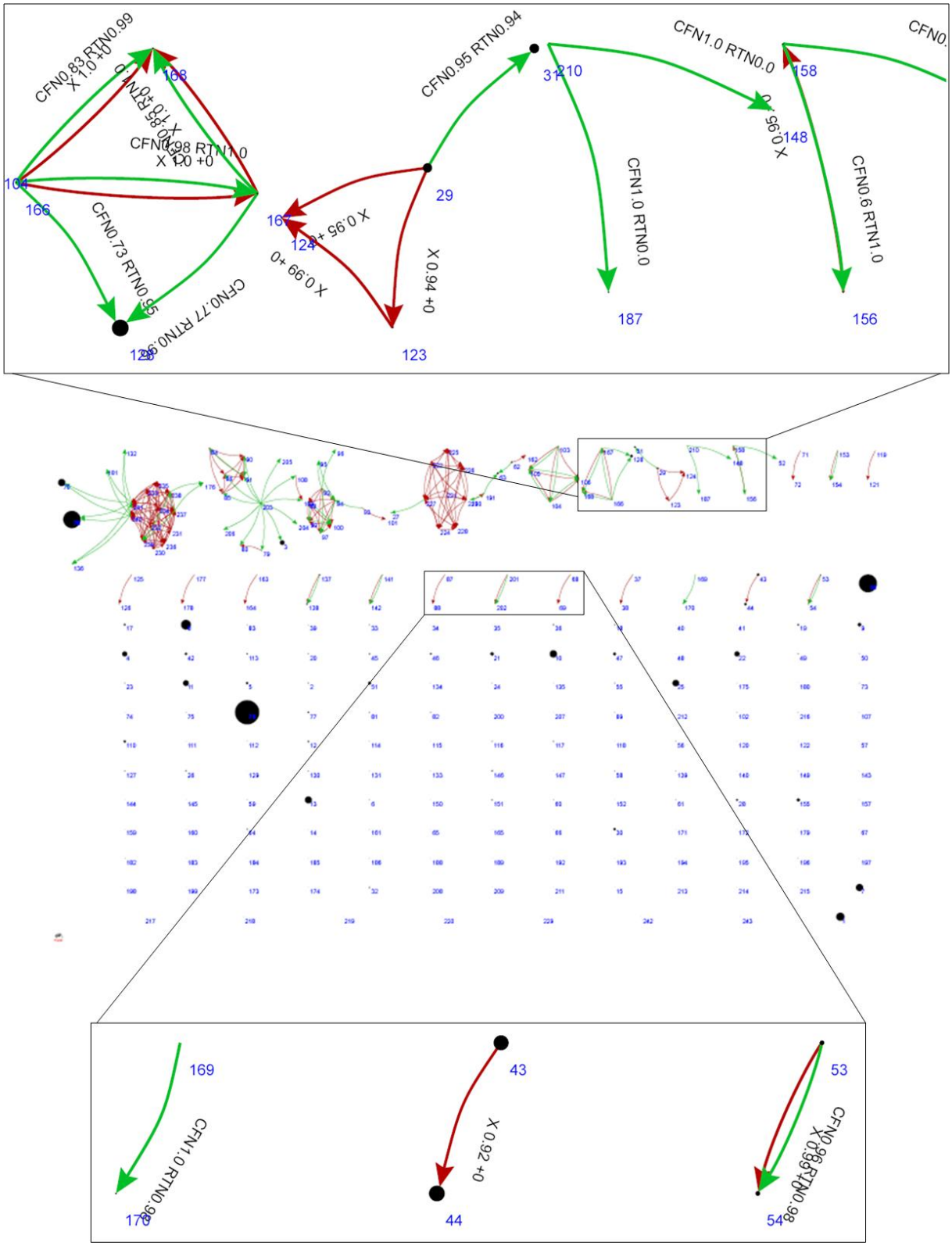

Figura 34 - Detalhes dos Padrões Encontrados na Lista de Componentes Conexos

Para avaliar a consistência dos resultados, o grafo das relações com significância acima de 0,9 com os componentes conexos identificados foi apresentado a um engenheiro de processo com sólidos conhecimentos acerca do processo e sistemas da planta da qual a base de dados de alarmes foi coletada. Após uma análise destes resultados, pode-se afirmar que **todos os padrões encontrados com relevância elevada e investigados, representavam, de fato, relações de redundância existentes no processo**. Para ilustrar e comprovar tal afirmação, alguns dos padrões considerados interessantes para o processo de racionalização serão apresentados a seguir. As imagens das figuras Figura 35, Figura 36 e Figura 37, apresentadas originalmente em [35], mostram as etapas do processo envolvido e serão utilizadas para descrever as relações de causa e efeito identificadas pelo algoritmo.

- **10→11 (BR2002_AFHEM→TC2007_AFAL):** a relação descoberta entre os TAGs 10 e 11 foi detectada através das arestas "X 0.84 +30" e "CFN0.6 RTN0.67", indicando que há correlação elevada e uma relação de precedência em relação a CFN e RTN no sentido 10→11, mas não o contrário. O TAG 10 corresponde ao TAG real BR2002_AFHEM, cujo significado é “Falta de Minério” no britador (canto direito superior da Figura 35). O TAG 11, que corresponde ao TAG real TC2007_AFAL significa “vazão baixa (falta de produto)” na correia transportadora, indicado na parte inferior da Figura 35. Pela imagem, é fácil perceber que há uma relação de causa-efeito entre os TAGs no processo, indicando um ponto possível de supressão[10] de um dos alarmes em situações em que ambos ocorram juntos;
- **155→22 (SL3082_ALAH_B→SL3082_ALAH):** esta relação foi identificada pela aresta "CFN0.79 RTN0.63" no sentido 155→22, indicando que a ocorrência do TAG SL3082_ALAH_B é sucedido pela ocorrência do TAG SL3082_ALAH em relação a CFN e RTN, com confiança maior para CFN. Os TAGs SL3082_ALAH e SL3082_ALAH_B indicam nível alto no silo, sendo A e B (parte central da Figura 36) alarmes redundantes, já que se trata do mesmo silo. Claramente estes alarmes terão uma relação de dependência física no processo[11];

[10] A supressão em si pode ocorrer através da criação de uma lógica no sistema de alarmes que apresente ao operador apenas um dos alarmes, quando esta sequência ocorrer.

[11] Apesar da redundância, é necessária uma investigação mais detalhada (ainda não realizada) sobre a razão que justificou dois alarmes na construção do sistema.

- **53→54 (SL3084_ALAHH_C→SL3084_ALAHH_D na Figura 34):** os alarmes SL3084_ALAHH_C SL3084_ALAHH_D também indicam nível alto no silo, mas para o silo 3084 (Figura 37), configurando a mesma situação descrita acima para a relação 155→22. Esta relação foi evidenciada pelas arestas "X 0.99 +0" e "CFN0.96 RTN0.98" na direção 53→54;
- **93→94 (SE01_AVCC→SE02_AVCC):** esta relação foi evidenciada pelas arestas "X 0.67 +0" e "CFN0.84 RTN0.91", indicando com maior confiança que o retorno do TAG SE01_AVCC é sucedido pelo retorno do alarme SE02_AVCC. SE01_AVCC e SE02_AVCC representam, respectivamente, falta de tensão contínua (DC) nas salas elétricas 01 e 02, que abrigam os controladores (CLPs). É fácil imaginar, considerando que os circuitos de alimentação de corrente alternada (AC) são relacionados, que em uma falta de tensão na sala 01, a sala 02 também seja afetada;
- **158→52 (TC4010_ALAH→PE4007_AFAHH):** esta relação foi evidenciada através da aresta "CFN0.6 RTN1.0", que indica uma relação de retorno simultâneo para os TAGs TC4010_ALAH e PE4007_AFAHH. Estes TAGs representam, respectivamente, "nível alto na pilha" de minério de saída do transportador de correia TC4010 e "vazão muito alta" na peneira PE4007. Considerando que a peneira e a pilha são posicionados em sequência no processo, já que a peneira alimenta a pilha, a normalização de um dos alarmes será sucedida pela normalização do outro[12].

[12] A supressão neste caso, novamente, não consiste na eliminação de um dos alarmes, mas na criação de uma lógica no sistema para que a ocorrência sequencial 158->52 seja apresentada ao operador através de um alarme único.

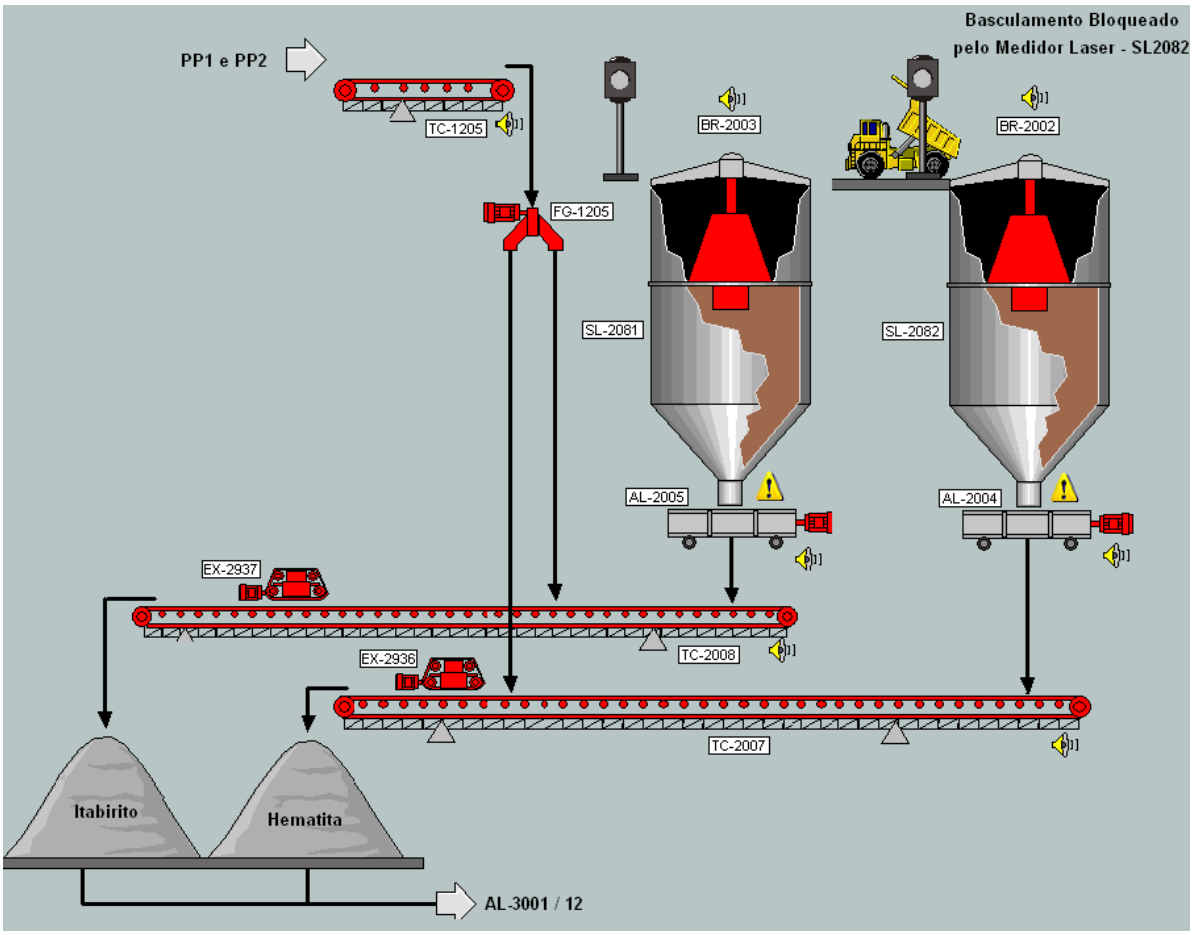

Figura 35 – Visualização do Supervisório da Britagem Primária [35]

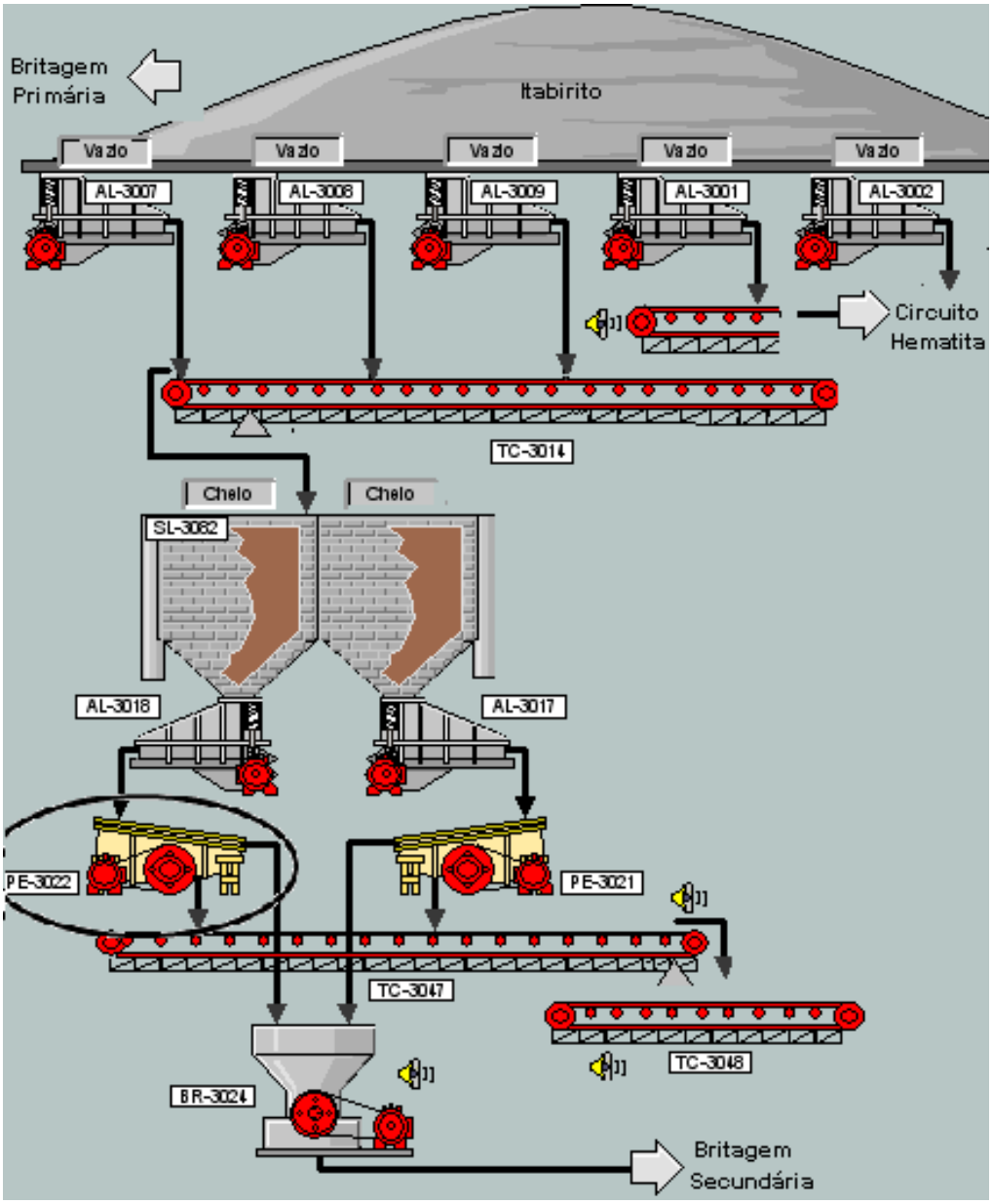

Figura 36 – Visualização do Supervisório da Britagem Secundária [35]

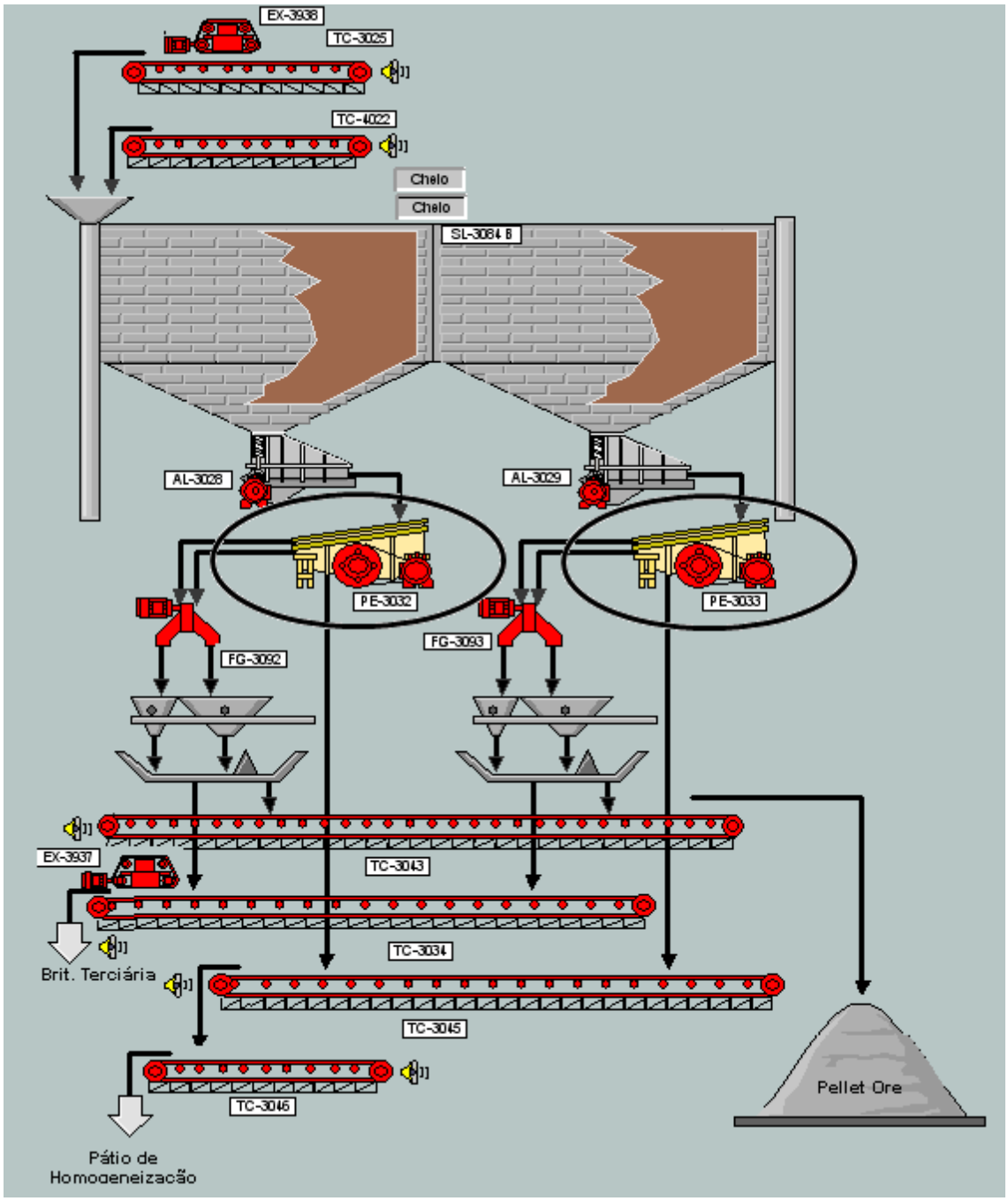

Figura 37 – Visualização do Supervisório do Peneiramento [35]

5 CONCLUSÕES E TRABALHOS FUTUROS

5.1 Conclusões

A área de Gerenciamento de Alarmes possui grande importância no setor industrial, já que sistemas com alarmes mal projetados, com um número muito grande de ocorrências diárias pode tornar a atividade de compreender o comportamento do processo e dos sistemas humanamente inviável para os operadores da planta.

Dentre as abordagens já utilizadas para a identificação dos padrões de alarmes redundantes, para posterior supressão através de lógicas e outras melhorias no sistema, estão a análise de correlação cruzada, sugerida pela norma EEMUA 191, e algoritmos de mineração de sequências como o WINEPI e MINEPI. A correlação cruzada isoladamente é insuficiente para detectar todos os tipos de padrões e os algoritmos como o WINEPI sofrem com o problema da dependência de parâmetros, com forte influência nos resultados. Uma abordagem para resolver o problema foi proposta no algoritmo *cross-effect-test* e descoberta de regras de associação, mas a solução utilizada também depende de um parâmetro similar às técnicas anteriores, além de ser incapaz de detectar todos os tipos de padrões temporais possíveis.

Neste trabalho foi apresentada uma solução inovadora, que consiste em um processo combinando diferentes algoritmos para resolver os problemas das abordagens anteriores e tornar os padrões minerados mais significativos e úteis, do ponto de vista do usuário que faz a interpretação ou análise semântica. Para facilitar esta atividade, a solução acrescenta ainda uma técnica de visualização com modelagem dos padrões minerados através de redes complexas e sugere uma mudança na dinâmica do processo de mineração, na qual os cortes das relações menos significativas ao usuário são feitos depois da descoberta dos padrões (ao invés de iterações alterando o valor de *minsup* e *minconf*), utilizando o auxílio das informações visuais.

Os resultados dos experimentos, realizados em uma base de dados sintética gerada a partir de um processo de caracterização de carga e na própria base de dados real coletada de um processo real de mineração de minério de ferro, apontam para um desempenho superior na capacidade de detecção dos vários tipos de padrões possíveis (com erro menor) em relação à melhor solução existente. Os tópicos a seguir detalham as demais conclusões obtidas:

- Todos os padrões inseridos na base sintética foram detectados pelo algoritmo, incluindo a cascata contendo uma sequência de três alarmes;
- Padrões do tipo YMX (Y *met by* X) são detectados com maior dificuldade tanto pela solução *cross-effect-test*, quanto pela nova abordagem proposta, embora na nova abordagem o erro tenha sido menor;
- Na análise com a base de dados real, todas as relações identificadas pelo algoritmo que foram avaliadas pelo engenheiro de processo representavam relações com algum tipo de dependência semântica no processo;
- A solução de navegação visual nos padrões através de uma rede complexa facilita a identificação dos grupos de alarmes relacionados através da detecção dos componentes conexos. A orientação temporal através da direção da aresta e as anotações dão suporte à atividade de interpretação, tornando o processo mais intuitivo e rápido;
- Os sete tipos de relações entre períodos de duração temporais levantados por Allen são passíveis de ocorrer em processos industriais, razão pela qual a abordagem de KORDIC et al. pode ser considerada pouco abrangente;
- O erro na detecção do padrão YOX na abordagem *cross-effect-test* foi ligeiramente inferior, já que o algoritmo se propõe a detectar especificamente este tipo de padrão. Contudo, na nova abordagem, a combinação com a análise de correlação cruzada permitiu a descoberta das características da relação de fato, com erro zero, portanto;
- Apesar do argumento dos autores no trabalho de KORDIC et al. sobre a eliminação do parâmetro *window size*, um novo parêmetro *minw* foi inserido. Os resultados experimentais confirmam que o parâmetro exerce forte influência nos resultados;
- O algoritmo escolhido na abordagem anterior (*cross-effect-test*) para minerar as transações faz com que a ordenação temporal se perca. Na nova abordagem esta precedência é preservada para os “padrões simples”, que representam a maior parte dos padrões;
- Considerando a dificuldade de acesso e o custo do trabalho dos especialistas do processo industrial, uma abordagem interessante é reduzir o tempo necessário de execução de algoritmos durante a interpretação. Por esta razão, na nova abordagem o algoritmo é executado com um valor de suporte e confiança mínimos baixos, criando a possibilidade de cortes instantâneos

durante a interpretação sem a necessidade de re-execução com valores diferentes de suporte e confiança.

Sobre estas conclusões, pode-se afirmar que o presente trabalho gerou as seguintes contribuições:

- Os padrões que a solução é capaz de detectar são mais abrangentes, se adequando a qualquer tipo de processo industrial, e não somente a processos químicos ou petroquímicos, como na solução anterior. Esta abrangência foi determinada a partir dos axiomas temporais de Allen, explorados pela primeira vez neste contexto;
- Os resultados foram validados utilizando uma base de dados sintética representativa para o problema de mineração de alarmes redundantes: houve uma atividade de caracterização de carga feita a partir de uma base de dados real e padrões artificiais abrangentes foram inseridos. Isto pode ser dito como sendo uma abordagem inovadora, consistindo em uma solução que permitiu também a quantificação dos falsos positivos;
- A influência dos parâmetros nos resultados foi reduzida, sendo o parâmetro “tamanho da janela deslizante” (*sliding window*) eliminado e a especificação de *minsup* e *minconf* pelo usuário facilitada pela modelagem visual;
- A solução utilizada na visualização é inovadora neste contexto e cria novas possibilidades e facilidades para o processo de mineração, incluindo a detecção dos componentes conexos (e, portanto, de padrões conexos) mais facilmente;
- O uso combinado de algoritmos, proposto de maneira original, reconhece que, para várias situações, a análise de correlação é a melhor alternativa e que a decisão sobre a melhor medida de interesse para cada caso é do usuário que conhece do domínio do problema;
- O uso de Regras de Associação Mínimas Não Redundantes (MNR) para a mineração dos padrões complexos cria uma forma mais sucinta de representação das regras de associação sem perda de informação em relação à abordagem com *FP-growth*;
- Em função dos resultados, esta nova abordagem proposta constitui uma solução efetivamente útil em termos práticos, na descoberta de padrões de alarmes redundantes, já que trata de todas as etapas, do tratamento inicial e limpeza da base de dados à interpretação visual dos padrões minerados.

5.2 Trabalhos Futuros

Este trabalho teve um foco muito grande no entendimento do problema e soluções existentes, enfatizando a capacidade da solução em detectar os tipos de padrões temporais que efetivamente podem ser encontrados em sistemas de alarmes industriais. Como continuação do trabalho, espera-se atuar nas seguintes oportunidades de melhoria:

- Utilizar apenas um algoritmo na mineração das transações, modificando, se possível, o ZART para preservar a ordenação temporal;
- Gerar regras de associação complexas também com ordenação temporal;
- Explorar outras métricas e recursos do software de análise de redes complexas na atividade de interpretação visual;
- Melhorar a capacidade de detecção dos padrões YMX;
- Utilizar outros atributos da base de dados, como a criticidade do alarme e hora do reconhecimento do operador, com o objetivo de revelar relações em outros aspectos;
- Realizar um trabalho de caracterização de carga em bases de dados relativas a outros tipos de processos para identificar diferenças que possam justificar uma modificação no gerador de cargas para tais tipos de processos;
- Criação de um software para uso em aplicações práticas capaz de agrupar as funções estatísticas normalmente utilizadas na racionalização, com a capacidade e potenciais descritos neste trabalho.

REFERÊNCIAS

[1] EEMUA - The Engineering Equipment & Materials User's Association Publication 191: Alarm Systems - a Guide to Design, Management and Procurement. 2007, 2nd edition.

[2] ANSI – American National Standards Institute: ANSI/ISA-18.2-2009, Management of Alarm Systems in the Process Industries. Washington – DC, 2009.

[3] Health and Safety Executive, 'The explosion and fires at the Texaco Refinery, Milford Haven, 24 July 1994: A report of the investigation by the Health and Safety Executive into the explosion and fires on the Pembroke Cracking Company Plant at the Texaco Refinery, Milford Haven on 24 July 1994',ISBN 0 7176 1413 1, 1997.

[4] HOLLIFIELD, Bill R., HABIBI, Eddie. Alarm Management: Seven Effective Methods for Optimum Performance. ISA – Instrumentation, Systems and Automation Society.

[5] BOYER, Stuart A. Supervisory Control and Data Acquisition, 2nd edition. ISA – International Society of Automation, 1999.

[6] HARLE, A., GARDNER, Robert D.. "Fault Resolution and Alarm Correlation in High-speed Networks using Database Mining Techniques". In: Proceedings of the 2002 Network Operations and Management Symposium (NOMS2002), pp 405-419, 2002.

[7] SRIKANT, R. AGRAWAL, R. Mining sequential patterns: generalizations and performance improvements. In: Proceeding of the 5th international conference on extending database technology (EDBT'96), Avignon, France, pp 3–17, 1996.

[8] AGRAWAL, R. SRIKANT, R "Mining sequential patterns". In: Proceedings of the 1995 international conference on data engineering (ICDE'95), Taipei, Taiwan, pp 3–14, 1995.

[9] MANNILA, H. TOINONEN, H. and VERKAMO, A. I. Discovery of frequent episodes in event sequences. Data Mining and Knowledge Discovery, 1:259–289, 1997.

[10] ZAKI, Mohamed. LESH, Neal. OGIHARA, "PLANMINE – Predicting Plan Failures Using Sequence Mining". Mitsnunori. 1998.

[11] HAN, Jiawei. CHENG, Hong. XIN, Dong, YAN, Xifeng. "Frequent pattern mining: current status and future directions". Springer Science+Business Media, LLC 2007.

[12] JENSEN, David, COHEN, Pault R, et al. "Family of Algorithms for Finding Temporal Structure in Data". In: 6th Intl. Workshop on AI and Statistics, Fort Lauderdale, Florida, USA, 1997.

[13] KORDIC, Savo. LAM, Peng. XIAO, Jitian. LI, Huaizhong. "Analisys of Alarm Sequences in a Chemical Plant". ADMA 2008, LNAI 5139, pp. 135-146. 2008.

[14] LESKOVEC, Jure. KRAUSE, Andreas, FALOUTSOS, Christos, et al. "A. Cost-effective Outbreak Detection in Networks". ACM KDD'07.

[15] BIKHCHANDANI, S. HIRSHLEIFER, D. and WELCH, I. "A theory of fads, fashion, custom, and cultural change as informational cascades". J. of Polit. Econ., (5), 1992.

[16] CORMEN, Thomas H., LEISERSON, C. E., RIVEST, Ronald L., STEIN, Clifford. "Introduction to Algorithms 2nd ed." The MIT Press: pp. 370-405, 2001.

[17] LESKOVEC, Jure. MCGLOHON, Mary, FALOUTSOS, Christos et al. "Cascading Behavior in Large Blog Graphs Patterns and a model".

[18] KLEINBERG, Jon. WATTS, Duncan. KOSSINETS, Gueorgi. "The Structure of Information Pathways in a Social Communication Network". SIGKDD 2008.

[19] LAMPORT, L. "Time, clocks, and the ordering of events in a distributed system. *Comm. ACM*, 21(7):558–565, 1978.

[20] LESKOVEC, Jure. HORVITZ, Eric. “Planetary-Scale Views on a Large Instant-Messaging Network”. Carnegie Mellon University. Microsoft Research 2008.

[21] AGRAWAL, R, IMIELINSKI T, SWAMI A (1993) “Mining association rules between sets of items in large databases”. In: Proceedings of the 1993ACM-SIGMODinternational conference on management of data (SIGMOD’93), Washington, DC, pp 207–21.

[22] ALLEN, James F. FERGUSON, George. “Actions and Events in Temporal Logic”. University of Rocherster, New York. Journal of Logic and Computation, Special Issue on Actions and Processes, 1994.

[23] NING, Pang. Tan. KUMAR, Vipin. SRIVASTAVA, Jaideep. “Selecting the Right Interestingness Measure for Association Patterns”. ACM SIGKDD 2002.

[24] AGRAWAL R, SRIKANT R (1994). “Fast algorithms for mining association rules”. In: Proceedings of the 1994 international conference on very large data bases (VLDB’94), Santiago, Chile, pp 487–499.

[25] JAIM, Raj, WILEY, John & Sons. “The Art of Computer System Performance Analysis: Techniques for Experimental Design, Simulation and Modeling”. 1991, ISBN: 0-471-50336-3.

[26] PEI J. HAN J. MORTAZAVI-Asl B. PINTO H, CHENG Q. DAYAL, U. “PrefixSpan: mining sequential patterns efficiently by prefix-projected pattern growth”. In: Proceeding of the 2001 international conference on data engineering (ICDE’01), Heidelberg, Germany, pp 215–224, 2001.

[27] Han J, Pei J, Yin Y (2000) “Mining frequent patterns without candidate generation”. In: Proceeding of the 2000 ACM-SIGMOD international conference on management of data (SIGMOD’00), Dallas, TX, pp 1–12.

[28] KRYSZKIEWICZ, M. “Representative association rules and minimum condition maximum consequence association rules”. In Proceedings of Second European Symposium on Principles of Data Mining and Knowledge Discovery (PKDD), 1998, LNCS, volume 1510, Springer-Verlag, Nantes, France, pages 361–369, 1998.

[29] LASZLO Szathmary. “Méthodes symboliques de fouille de données avec la plate-forme Coron”. Thése pour l'obtention Du Doctorat de l'université Henri Poincaré, Nancy. Departement de formation doctorale en informatique, UFR STMIA. Nancy, Italy. 2006.

[30] LASZLO Szathmary, Amedeo Napoli, and Sergei O. Kuznetsov . “ZART: A Multifunctional Itemset Mining Algorithm”. Fifth International Conference on Concept Lattices and Their Applications CLA 2007.

[31] BASTIDE, Y., N. Pasquier, R. Taouil, L. Lakhal, and G. Stumme. “Mining minimal non-redundant association rules using frequent closed itemsets”. In Proceedings of the International Conference DOOD’2000, LNAI, volume 1861, Springer-Verlag, London, UK, pages 972–986, July 2000.

[32] NOOY, W. de, A. Mrvar, V. Batagelj: “Exploratory Social Network Analysis with Pajek”. CUP, January 2005; ESNA page.

[33] LESKOVEC, Jure, SINGH, Ajit, KLEINBERG, Jon. “Patterns of Influence in a Recommendation Network”. School of Computer Science Carnegie Mellon University.

[34] DUFFIN, Joseph. DEVITT, Ann. MOLONEY, Robert. “Topographical Proximity for Mining Network Alarm Data”. SIGCOMM 2005.

[35] ALMEIDA, Ludmila V. M., JOTA, Fabio G.; “Proposta de Melhorias para o Sistema de Alarmes de Uma Planta de Beneficiamento de Minério de Ferro”. Monografia de Conclusão de Curso de Graduação em Engenharia de Controle e Automação – UFMG. Belo Horizonte, MG, 2009, p. 77-91.