*Unisanta Science and Technology*, **2026, 10, September**
Published Online 2026 Vol.15 Nº1 http://periodicos.unisanta.br/index.php/sat

UNISANTA
*Science and Technology*
**ISSN 2317-1316**

# Monitoramento de Redes Industriais com LPA2v: Comparação contextual com *Triggers* de limiar inspirados na plataforma de monitoramento *Zabbix*

**Alexandre de Carvalho[1], Arnaldo de Carvalho Junior[1,2], João Inácio da Silva Filho[1]**

[1]UNISANTA–Universidade Santa Cecília - Grupo de Lógica Paraconsistente Aplicada - Rua Oswaldo Cruz, 277 - Boqueirão - Santos/SP
[2]IFSP- Embedded Artificial Intelligence Laboratory (EAILab), Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, Rua Maria Cristina, 50, Jardim Casqueiro – Cubatão/SP.

**E-mail***:* alexandre@stratustelecom.com.br
*Received July*, 2026

**Resumo:** Em uma sala de operação, nem todo alarme significa falha. Um *backup* pode elevar a CPU; uma política de rede pode bloquear o ICMP (Protocolo de Mensagens de Controle da Internet); e uma interrupção breve pode desaparecer antes de afetar a produção. O desafio é separar esses casos de uma degradação real sem esconder informações importantes do operador. Este trabalho compara um modelo tradicional de monitoramento por *triggers* e limiares, inspirado em configurações usuais da plataforma de monitoramento Zabbix, com um modelo baseado em LPA2v. A comparação foi realizada no Scilab com 50 ativos simulados por 12 h e a mesma janela de avaliação de cinco minutos para os dois modelos. No traço avaliado, o modelo LPA2v elevou a precisão de 35,2% para 73,1%, manteve o recall entre 78,8% e 95,0% e reduziu os episódios de problema de 8 para 5. Os resultados são específicos do cenário simulado, mas mostram que o uso explícito de contexto pode ajudar a reduzir alertas indevidos e destacar situações que exigem confirmação humana.
**Palavras-chave:** LPA2v; Zabbix; Monitoramento de Redes; *Triggers*; Alarmes; Redes Industriais.

# Industrial Network Monitoring with PAL2v: Contextual comparison using threshold triggers inspired by the Zabbix Monitoring Platform

**Abstract:** In an operations room, not every alarm indicates a failure. A backup may increase CPU utilization, a network policy may block ICMP (Internet Control Message Protocol), and a short interruption may disappear before affecting production. The challenge is to distinguish these conditions from a genuine degradation without hiding relevant information from the operator. This work compares a traditional trigger-and-threshold monitoring model, inspired by common Zabbix Monitoring Platform configurations, with an PAL2v-based model. The comparison was implemented in Scilab with 50 simulated assets over 12 hours, using the same five-minute evaluation window for both methods. In the evaluated trace, the PAL2v model increased precision from 35.2% to 73.1%, kept recall between 78.8% and 95.0%, and reduced problem episodes from 8 to 5. These results are specific to the simulated scenario but indicate that explicit context can reduce undue alerts and highlight situations requiring human confirmation.
**Keywords:** PAL2v; Zabbix; Network Monitoring; Triggers; Alarms; Industrial Networks.

## 1. INTRODUÇÃO

Quem acompanha uma rede industrial convive com uma situação aparentemente simples, mas desgastante: o mesmo sinal pode ter significados muito diferentes. Uma unidade central de processamento (*Central Processing Unit* - CPU) elevada durante um *backup* costuma ser esperada; a mesma CPU elevada, acompa-nhada de lentidão e falhas de comunicação pode anunciar um problema que merece atenção imediata. Quando o monitoramento trata as duas situações da mesma forma, o resultado são notificações em excesso e perda de confiança nos alarmes [1],[2]. Plataformas de monitoramento como o Zabbix trabalham com itens, *triggers*,


Copyright © 2026 ISESC. 

eventos, severidades e correlação. Em uma configuração comum, uma expressão de *trigger* passa a indicar problema quando uma métrica ou uma condição definida pelo administrador se torna verdadeira [3].

Esse mecanismo é essencial e amplamente utilizado, entretanto, o resultado prático depende do conhecimento com que os limiares, as janelas de tempo, as manutenções e as exceções são configuradas [4].

Este artigo não pretende medir o desempenho do produto Zabbix nem reproduzir todos os seus recursos. O objetivo é mais restrito: comparar um conjunto limitado de *triggers* por limiar, inspirado nesse modo de operação, com um modelo de lógica paraconsistente anotada por anotação de 2 valores (LPA2v) que recebe as mesmas medições e acrescente contexto operacional à decisão.

A LPA2v pertence a família de lógicas não clássicas paraconsistentes [6],[7] e a intenção dessa pesquisa é investigar se quando há sinais conflitantes, essa segunda abordagem ajuda o operador a entender melhor o que está acontecendo e assim tomar decisões consideradas mais assertivas [5].

Monitoramento por Limiares e LPA2v

No modelo tradicional da simulação, uma média de cinco minutos é comparada com limites para CPU, memória, erros de verificação de redundância cíclica (*Cyclic Redundancy Check* - CRC), *broadcast*, *drops*, latência e tempo de resposta. A manutenção programada é suprimida, pois o Zabbix também oferece recursos de manutenção e correlação de eventos [3].

Ainda assim, o modelo por limiar não diferencia sozinho um *ping* bloqueado de uma indisponibilidade completa quando a regra foi criada apenas para o protocolo de mensagem de controle de entre redes (*Internet Control Message Protocol* - ICMP).

A LPA2v trabalha de outro modo. Para a hipótese apresentada na forma de uma proposição P= "há falha operacional", o modelo recebe uma evidência favorável ($\mu$) e uma evidência desfavorável ($\lambda$). O grau de certeza ($G_C$) é calculado conforme Eq (1). Já o Grau de Contradição ($G_{CT}$) é calculado pela Eq. (2) [7].

$$G_C = \mu - \lambda \tag{1}$$

$$G_{ct} = \mu + \lambda - 1 \tag{2}$$

Assim, uma CPU elevada aumenta $\mu$; backup ativo e serviços respondendo aumentam $\lambda$. Quando as duas evidências são relevantes ao mesmo tempo, o resultado pode ser "inconsistente", isto é, um pedido de verificação, e não uma declaração automática de falha [4,5].

## 2. MATERIAIS E MÉTODOS

A Figura 1 apresenta a arquitetura hierárquica de decisão proposta, baseada em LPA2v. As evidências são agrupadas em: Disponibilidade, *Performance*, Rede Industrial, Aplicação e Contexto Operacional, dados simulados da plataforma aberta de comunicação (*Open Platform Communications* - OPC) [8], sistema de supervisão, controle e aquisição de dados (*Supervisory Control and Data Acquisition* - SCADA) [9], além de dados dos equipamentos via SNMP.

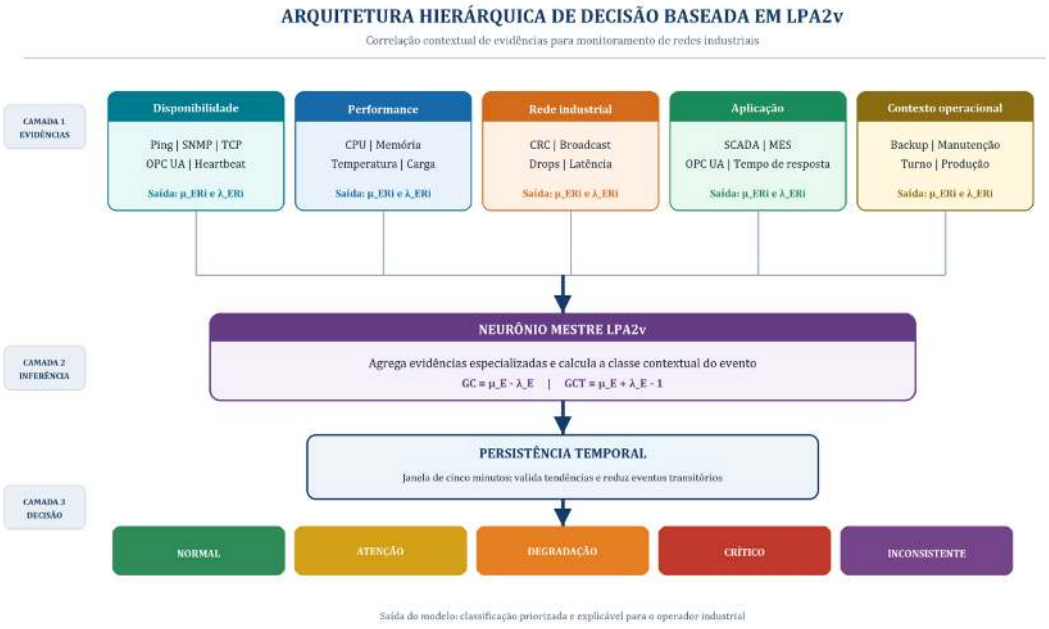



**Figura 1.** Fluxo de decisão do modelo LPA2v: evidências especializadas, agregação, persistência temporal e classes finais.



Copyright © 2026 ISESC 

A simulação foi implementada no Scilab 2026.0.1 e reproduziu 12 h de telemetria para 50 ativos. Os dois modelos receberam a mesma série temporal e utilizaram uma janela de cinco minutos. O traço contém operação normal e sete situações representativas:

a) *Backup* previsto;
b) Manutenção planejada;
c) Bloqueio de ICMP com protocolo simples de gerenciamento de rede (*Simple Network Management Protocol* - SNMP) e protocolo de controle de transmissão (*Transmission Control Protocol* - TCP) ativos;
d) *Loop* Ethernet progressivo;
e) Uma rajada transitória de tráfego;
f) Lentidão de supervisório, e
g) Indisponibilidade real de um Controlador Lógico Programável (CLP).

O modelo tradicional gerou problemas quando as médias ultrapassaram os limites pré-definidos. O modelo LPA2v combinou disponibilidade, desempenho, qualidade da rede, resposta da aplicação e contexto de *backup* ou manutenção. Ainda de acordo com a Figura 1, A saída foi classificada como NORMAL (verde), ATENÇÃO (amarelo), DEGRADAÇÃO (laranja), CRÍTICO (vermelho) e INCONSISTENTE (púrpura).

Para evitar reações a um único ponto de coleta, o grau de certeza também foi calculado em uma janela de cinco minutos.

No *backup* previsto, a CPU permanece elevada durante alguns minutos.

A avaliação utilizou as seguintes métricas: precisão, *recall*, F1-score, minutos classificados incorretamente como problema e número de episódios de problema.Como se trata de dados sintéticos, os valores não devem ser generalizados para outras plantas ou configurados como metas de desempenho. Eles servem para tornar explícita a diferença entre decidir somente pelo limiar e decidir com contexto.

## 3. RESULTADOS E DISCUSSÃO

A Figura 2 apresenta os resultados da simulação. No traço simulado, os *triggers* por limiar apresentaram precisão de 35,2%, recall de 78,8% e F1-score de 48,6%. O modelo LPA2v apresentou precisão de 73,1%, *recall* de 95,0% e F1-score de 82,6%. Foram observados 116 minutos de falsos positivos no modelo por limiar e 28 no modelo LPA2v; os episódios de problema passaram de 8 para 5. A diferença não deve ser interpretada como superioridade universal da LPA2v, pois os pesos e os cenários favorecem a análise de contexto. Ela indica, contudo, que o contexto tem efeito mensurável na série construída.

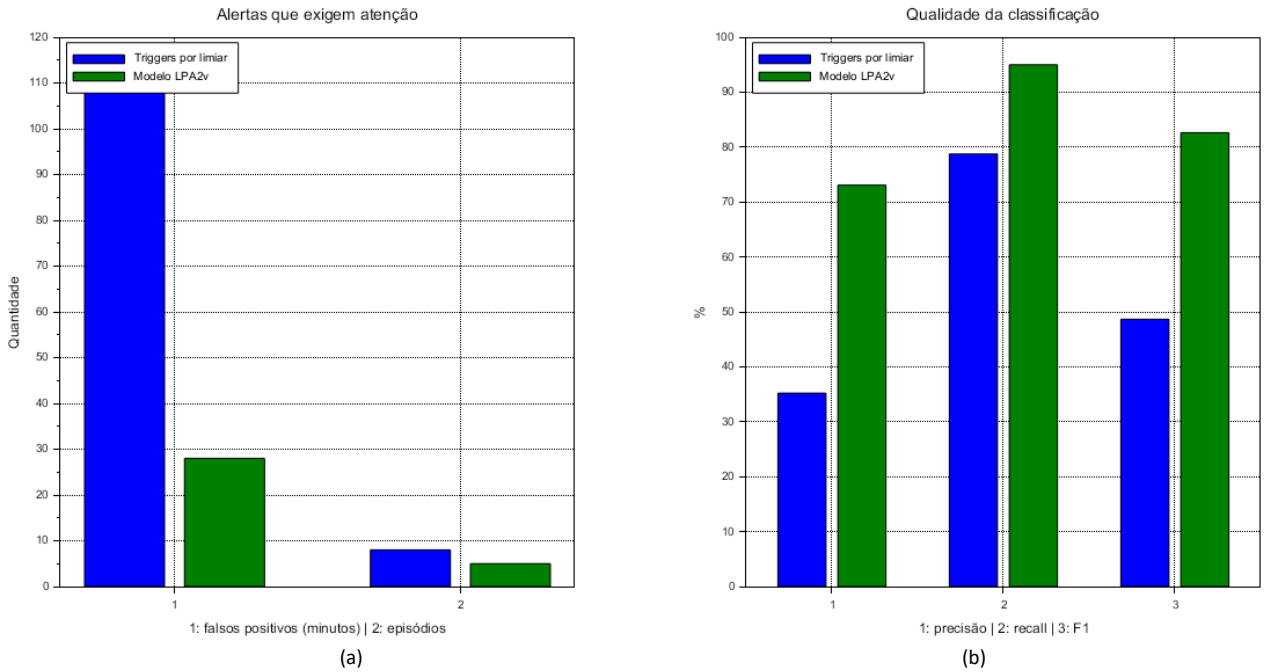



**Figura 2.** Gráficos comparativos. a) Comparação no cenário simulado: minutos de falsos positivos. (b) Episódios de problema, precisão, recall e F1-score.


Copyright © 2026 ISESC

A Figura 3 mostra duas situações em que o valor isolado é insuficiente. O *trigger* por limiar abre problema, enquanto o modelo LPA2v considera a evidência de *backup* e mantém a condição fora das classes de degradação e crítico.

No bloqueio de ICMP, o trigger interpreta a ausência de resposta como problema; a LPA2v identifica a contradição porque o SNMP e TCP continuam disponíveis e classifica o caso como inconsistente.

Essa classe não significa ignorar o evento: significa solicitar confirmação por outras fontes antes de abrir um incidente de indisponibilidade.

A Figura 4 apresenta o *loop* Ethernet progressivo, que foi incluído para mostrar que contexto não serve apenas para reduzir notificações, pois à medida que CRC, *broadcast*, *drops* e latência sobem conjuntamente, o $G_C$ do modelo LPA2v cresce.

Na série avaliada, o modelo identificou a degradação um minuto antes do instante marcado como início da falha, enquanto o *trigger* por limiar reagiu no próprio início.

Esse ganho é pequeno e deve ser visto como indicativo; em uma implantação real ele depende dos limites, da frequência de coleta e da qualidade dos dados.

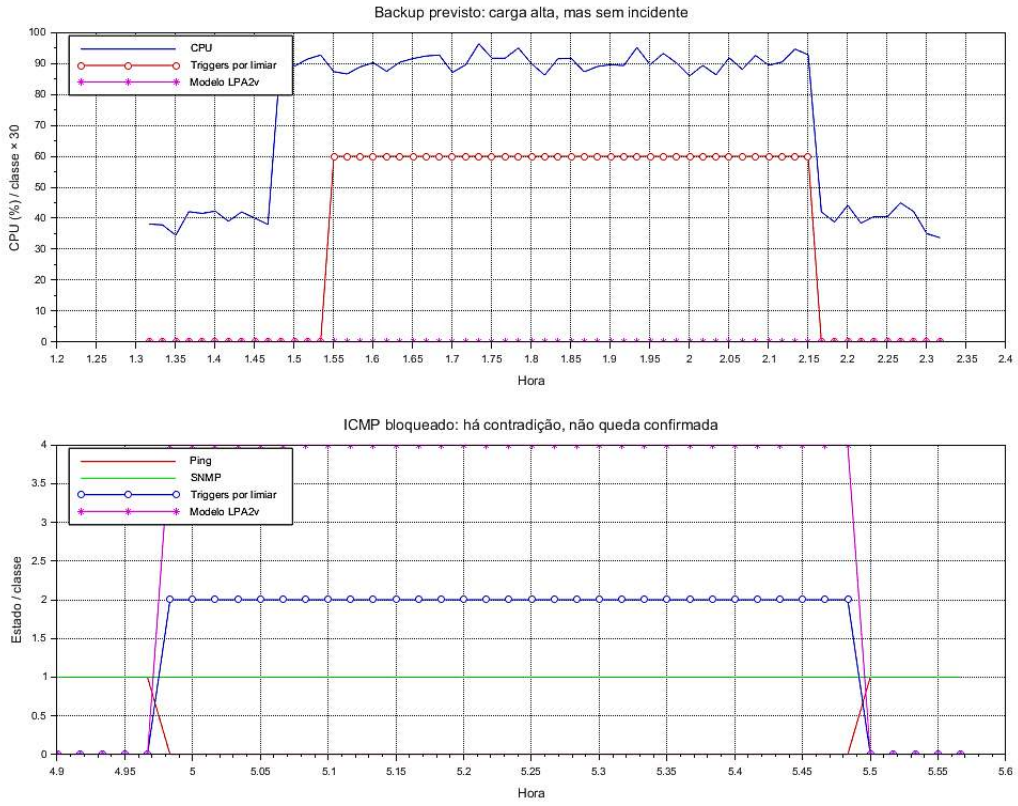



**Figura 3**. Casos contextuais: backup previsto e ICMP bloqueado com serviços ainda disponíveis.


Copyright © 2026 ISESC

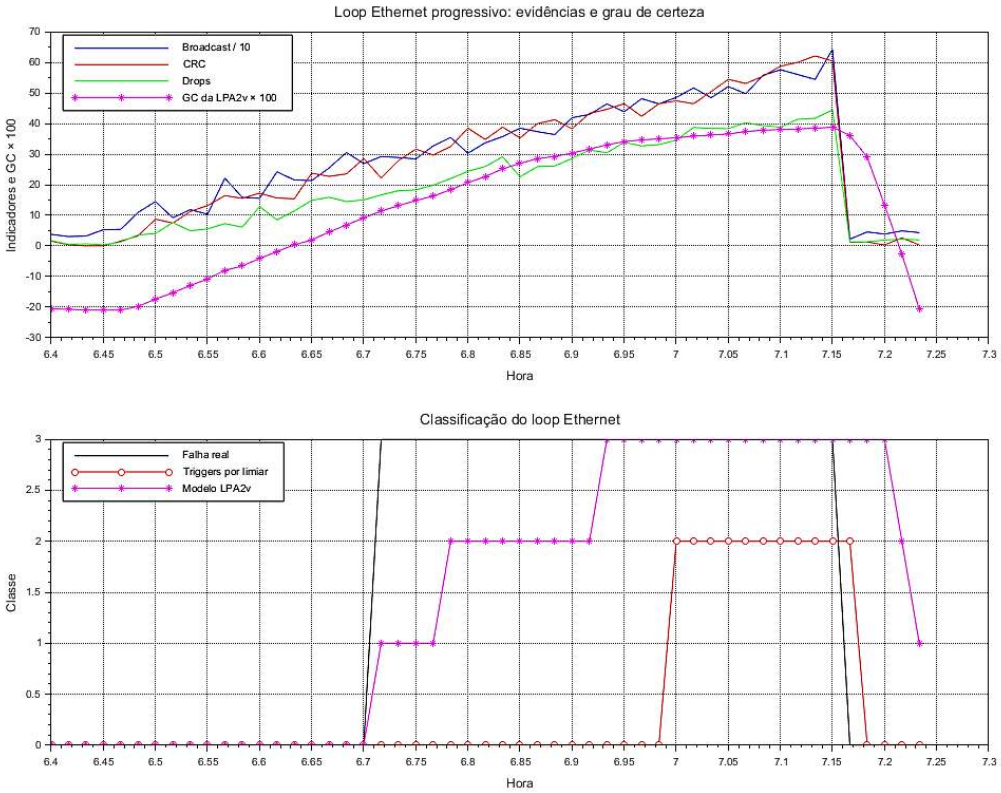



**Figura 4.** Loop Ethernet progressivo: evolução dos indicadores, grau de certeza e classes atribuídas pelos dois modelos.

## 4. CONSIDERAÇÕES FINAIS

A pesquisa mostrou que o monitoramento tradicional por *triggers* continua importante: ele é simples, transparente e rápido de configurar. O modelo LPA2v não precisa substituí-lo. A proposta é usá-lo como uma camada de decisão quando o evento não pode ser explicado apenas por um número acima do limite.

No cenário estudado, essa camada reduziu alertas indevidos e manteve a capacidade de identificar falhas reais. O resultado mais útil não é uma promessa de eliminação de falsos positivos, mas a possibilidade de dar um significado diferente a eventos ambíguos: esperado, crítico ou inconsistente. Para o operador, essa distinção pode ser mais valiosa do que simplesmente receber mais uma notificação.

Como continuidade, o modelo deve ser avaliado com telemetria real, exportada de uma instalação Zabbix ou de outra plataforma de monitoramento.

Essa etapa permitirá calibrar $\mu$ e $\lambda$ com dados observados, definir políticas para a classe inconsistente e verificar se os ganhos permanecem em ambientes com diferentes equipamentos, janelas de manutenção e perfis de tráfego.

## REFERÊNCIAS

[1] PILLO-GUANOLUISA, Darwin et al. Real-Time Network Infrastructure Monitoring: A Zabbix and Telegram Integration Case Study in the Energy Sector (EMELNORTE SA). In: 2025 IEEE Ninth Ecuador Technical Chapters Meeting (ETCM). IEEE, 2025. p. 1-6.

[2] LEE, Sihyung; LEVANTI, Kyriaki; KIM, Hyong S. Network monitoring: Present and future. Computer Networks, v. 65, p. 84-98, 2014.



Copyright © 2026 ISESC 

[3] ZABBIX LLC, "Triggers and trigger-based event correlation," Zabbix Documentation 7.4, 2026. [Online]. Available: https://www.zabbix.com/documentation/7.4/en/manual/config/triggers. Accessed: Aug. 30, 2026.

[4] INTERNATIONAL SOCIETY OF AUTOMATION, ANSI/ISA-18.2-2016: Management of Alarm Systems for the Process Industries. Research Triangle Park, NC, USA: ISA, 2016.

[5] EEMUA, EEMUA Publication 191: Alarm Systems - A Guide to Design, Management and Procurement, 3rd ed. London, U.K.: EEMUA, 2013.

[6] DA COSTA, Newton CA. On the theory of inconsistent formal systems. Notre dame journal of formal logic, v. 15, n. 4, p. 497-510, 1974.

[7] DE CARVALHO JUNIOR, Arnaldo et al. A comprehensive review on paraconsistent annotated evi-

dential logic: Algorithms, Applications, and Perspectives. Engineering Applications of Artificial Intelligence, v. 127, p. 107342, 2024.

[8] OPC FOUNDATION, "OPC Unified Architecture," 2026. [Online]. Disponível em: https://opcfoundation.org/about/opc-technologies/opc-ua/. Acessado em: Agosto 30, 2026.

[9] IEC, IEC 62443-3-3:2013, Industrial communication networks - Network and system security - Part 3-3: System security requirements and security levels. Geneva, Switzerland: IEC, 2013.



Copyright © 2026 ISESC 