Disaster-FD: Um detector de falhas para ambientes suscetíveis a desastres

Abadio de Paulo Silva[1], Anubis Graciela de Moraes Rossetto[3], Pierre Sens[2], Luciana Arantes[2], Rafael Pasquini[1], Paulo Coelho[1]

[1]Universidade Federal de Uberlândia – Minas Gerais – Brasil

[2]Universidade de Sorbonne, CNRS, LIP6 – Paris – França

[3]Instituto Federal Sul-rio-grandense – Rio Grande do Sul – Brasil

{abadio,rafael.pasquini,paulocoelho}@ufu.br,

{luciana.arantes,pierre.sens}@lip6.fr,anubisrossetto@ifsul.edu.br

Abstract. *This paper introduces Disaster-FD, a failure detector designed for disaster-prone environments focusing on real-time monitoring of IoT networks. Inspired by Impact-FD, this approach emphasizes active monitoring and assessment of network reliability. The paper explores key elements such as reliability threshold, confidence level, and impact factor. Tests on the IoT-LAB platform demonstrate the functionalities of Disaster-FD in various environments, highlighting its relevance in natural disaster scenarios.*

Resumo. *Este trabalho apresenta o Disaster-FD, um detector de falhas para ambientes suscetíveis a desastres com foco no monitoramento em tempo real de redes IoT. Inspirado no Impact-FD, esta abordagem dá ênfase ao monitoramento ativo e avaliação da confiabilidade da rede. O trabalho explora elementos chaves como limiar de confiabilidade, nível de confiança e fator de impacto. Testes na plataforma IoT-LAB demonstram as funcionalidades do Disaster-FD em ambientes variados, reforçando sua relevância em cenários de desastres naturais.*

1. Introdução

Desastres naturais representam uma das maiores ameaças à sociedade contemporânea, causando impactos devastadores que vão desde o interrompimento de serviços essenciais, como abastecimento de água e energia, até prejuízos econômicos significativos, danos a propriedades públicas e privadas, e o mais grave, a perda de vidas humanas. Esses eventos assolam o mundo e, particularmente no Brasil, a maioria dos desastres naturais têm suas causas relacionadas ao clima.

Conforme relatório da Confederação Nacional de Municípios (CNM) em 2022, cerca de 3,4 mil pessoas no Brasil foram diretamente afetadas por desastres naturais [Janoneda 2022]. Esse contexto desafiador ressalta a necessidade premente de desenvolver estratégias eficazes para o monitoramento e gestão de riscos associados a esses desastres. Uma abordagem promissora para enfrentar este desafio é a utilização de tecnologias de monitoramento remoto baseadas na Internet das Coisas (IoT - *Internet of Things*). A interconectividade de diversos objetos numa rede, proporcionada pelo paradigma IoT, oferece uma oportunidade única para melhorar a resposta a desastres naturais.

No entanto, a eficácia dessa estratégia é frequentemente comprometida em cenários de grandes desastres, onde a infraestrutura de comunicação, incluindo os dispositivos IoT instalados para monitoramento, podem falhar. Tal falha pode subdimensionar a gravidade do desastre pela falta de dados e impedir a emissão de alarmes críticos. Assim, torna-se essencial desenvolver algoritmos robustos capazes de monitorar eficientemente uma rede IoT, assegurando a disponibilidade dos dispositivos e estabelecendo um nível de confiança nos processos originados desse ecossistema. Adicionalmente, é crucial propor melhorias no gerenciamento desses sensores para garantir um monitoramento eficiente e confiável, mesmo em condições adversas.

Neste cenário, emerge a relevância de detectores de falhas eficazes. [Chandra and Toueg 1996], pioneiros no estudo de detectores de falhas não confiáveis, destacaram a importância de propriedades fundamentais como completude e precisão em tais sistemas. Tais propriedades asseguram que os algoritmos que utilizam detectores de falhas mantenham a consistência em suas decisões e não fiquem bloqueados indefinidamente [Chandra et al. 1996].

Este estudo, realizado em cooperação com universidades na França, Uruguai e Chile, visa desenvolver um algoritmo para monitoramento em tempo real de redes de Internet das Coisas. Inspirado no *Impact Failure Detector* [Rossetto et al. 2018], ou *Impact-FD*, o algoritmo proposto adota uma abordagem com algumas particularidades, como o monitoramento ativo e federado de dispositivos IoT.

A plataforma IOT-LAB [Adjih et al. 2015], com mais de 1500 nós de sensores espalhados por várias localidades na França, incluindo Grenoble, Lille, Saclay e Strasbourg, destaca-se como uma dos maiores *testbeds* abertas disponíveis para a comunidade científica internacional. Para facilitar a experimentação em redes IoT complexas, a IoT-LAB oferece suporte a uma variedade de protocolos de comunicação. Adicionalmente, a IoT-LAB facilita a criação de experimentos federados. Isso significa que os dispositivos podem ser programados e gerenciados simultaneamente em múltiplas regiões, proporcionando uma plataforma ideal para testar algoritmos e aplicações distribuídas em uma rede que simula a complexidade da Internet global.

O conceito de monitoramento federado em redes IoT se refere à prática de integrar e gerenciar múltiplas redes de dispositivos IoT autônomos, que são distribuídos geograficamente ou pertencem a diferentes domínios administrativos. Este modelo de monitoramento é projetado para melhorar a eficiência, a segurança e a resiliência de grandes sistemas IoT, permitindo uma supervisão mais efetiva e uma resposta coordenada a incidentes ou falhas. Ao monitorarmos diferentes regiões simultaneamente, é possível identificar rapidamente quando uma região começa a apresentar sinais de falha ou anomalia. Isso é essencial em infraestruturas críticas, como redes elétricas, onde uma falha em uma área pode ter efeito cascata em outras regiões.

Em um sistema federado, cada rede IoT opera de forma independente, mas compartilha informações com outras redes para melhorar a vigilância e a gestão global. Essa abordagem é enfatizada por [Atzori et al. 2010] em seu trabalho sobre o paradigma da Internet das Coisas, onde eles discutem a necessidade de colaboração entre dispositivos e sistemas distribuídos.

O principal objetivo deste estudo é monitorar a disponibilidade dos dispositivos

IoT e estabelecer um nível de confiabilidade para a rede, considerando um conjunto específico de processos monitorados. Embora não seja explorado neste artigo, este estudo também adota o monitoramento do consumo de energia dos dispositivos na rede.

O restante do artigo é estruturado da seguinte maneira. A Seção 2 apresenta o referencial teórico e o modelo adotado. O Disaster-FD é detalhado na Seção 3. A Seção 4 analisa metodologias para estimar o tempo de chegada de *heartbeats*. A implementação do Disaster-FD é abordada na Seção 5. Resultados experimentais são discutidos na Seção 6. Trabalhos relacionados são revisados na Seção 7, enquanto a Seção 8 conclui o artigo.

2. Modelo e Definições

Esta seção apresenta o modelo do sistema e definições relacionadas ao escopo do trabalho.

2.1. Modelo do Sistema

Este trabalho considera um sistema distribuído composto de um conjunto finito de processos $\Pi = \{q_1, \ldots, q_n\}$, em que $|\Pi| = n, (n \geq 2)$, e a existência de apenas um processo por nó ou sensor. Cada nó (ou processo), possui um identificador único. Os identificadores são ordenados consecutivamente. Processos podem falhar por parada e não se recuperam. Um processo é considerado correto se não falhar durante toda a execução.

Considera-se ainda um sistema distribuído **assíncrono**, como um sistema onde não existe um tempo de transmissão da mensagem ou tempo de execução da etapa de um processamento, ou seja, não há nenhuma hipótese relacionada à temporização [Cristian and Fetzer 1999, Verissimo and Rodrigues 2012]. Neste tipo de sistema, nenhum mecanismo pode garantir a falha de um processo remoto, uma vez que é impossível diferenciar um processo que falhou de um processo ou comunicação lentos.

O sistema possui canais de comunicação do tipo ***lossy***. De acordo com [Aguilera et al. 2004], um canal do tipo *lossy* satisfaz a propriedade de **integridade**, ou seja, garante que um processo q recebe uma mensagem m de um outro processo p no máximo uma vez, e somente se p enviou m para q previamente. Na prática, isto significa que mensagens não podem ser criadas e que, se uma mensagem m não é perdida, ela é eventualmente recebida em seu destino.

2.2. Detectores de falhas não confiáveis

A detecção de falhas em sistemas distribuídos é um elemento chave para garantir a confiabilidade e a estabilidade desses sistemas complexos, especialmente considerando-se sistemas assíncronos e o uso de detectores de falhas para contornar a impossibilidade de FLP (Fischer, Lynch e Patterson) [Fischer et al. 1985, Chandra et al. 1996].

[Chandra and Toueg 1996] introduziram o conceito de detectores de falhas não confiáveis, definidos por duas propriedades fundamentais: completude e precisão. **Completude** diz respeito à capacidade de o detector identificar corretamente todos os processos que falharam, enquanto **precisão** se refere à capacidade de não classificar incorretamente processos corretos como falhos. Na prática, estes detectores de falhas produzem como saída uma lista de processos considerados suspeitos.

O trabalho proposto neste artigo estende a definição de detectores de falha, utilizando a mesma abordagem definida em [Rossetto et al. 2018]. Neste contexto, existe um

processo $p \in \Pi$ que monitora um subconjunto S de Π. Cada processo em S conecta-se a p por meio de um canal de comunicação.

Deste modo, diferentemente de detectores tradicionais definidos em [Chandra and Toueg 1996], Disaster-FD pode ser definido como um detector de falhas não confiável que oferece uma saída relacionada ao **nível de confiança** nos processos em S. Se o nível de confiança é igual ou superior a um **limiar** definido pelo usuário, o sistema é considerado confiável. Em outras palavras, quando Disaster-FD é invocado no processo p, ele retorna o nível de confiança de p nos processos em S.

3. Disaster-FD

O Disaster-FD é um sistema de detecção de falhas desenvolvido para ambientes propensos a desastres, estendendo o sistema originalmente definido no Impact-FD [Rossetto et al. 2018] para operar em um cenário de múltiplas regiões com monitoramento intra e inter-regiões em tempo real.

Tal extensão introduz o uso de múltiplos processos monitores, cada um implantado em uma região monitorada, conforme exemplo apresentado na Figura 1. Neste cenário, cada região conta com um processo monitor e um conjunto de dispositivos IoT. O monitor de cada região monitora os dispositivos de sua própria região e um subconjunto dos processos em outra região. Tal arranjo permite incrementar a tolerância a desastres, impedindo que a falha completa de uma região passe desapercebida.

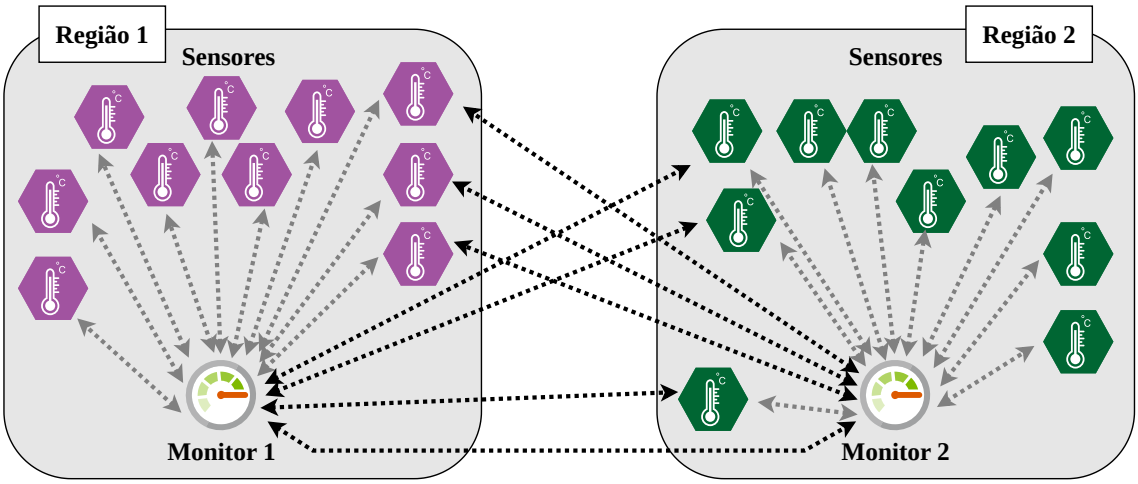

Figura 1. Exemplo de um cenário de monitoramento.

Neste contexto, cada processo q dentro do conjunto $S \subset \Pi$ recebe um fator de impacto atribuído. Esse fator, um valor real positivo, reflete a importância relativa do processo dentro do sistema. No cenário da Figura 1, por exemplo, pode-se atribuir um fator de impacto maior aos monitores em comparação ao fator de impacto dos sensores, expressando que a falha de um monitor tem um peso maior que a falha de um sensor.

Cada processo monitor p invoca o Disaster-FD e realiza o cálculo do nível de confiança para o conjunto S, estabelecendo assim a confiabilidade geral do sistema. Esse nível de confiança é determinado pela soma dos fatores de impacto dos processos que, naquele instante, não são considerados falhos.

Diferentemente do Impact-FD, os valores dos fatores de impacto são escolhidos para garantir que o conjunto de processos acompanhados pelo monitor em cada região coloque o nível de confiança abaixo do limiar caso a região monitorada deixe de responder.

3.1. Formalização

Esta seção formaliza as definições e os conceitos apresentados. Disaster-FD segue a mesma nomenclatura adotada em [Rossetto et al. 2018], expandindo os conceitos para um cenário de multi-monitoramento ou monitoramento federado. Esta incorporação melhora a eficiência, precisão e adaptabilidade do sistema, qualidades indispensáveis para enfrentar as incertezas e rápidas mudanças nos cenários de desastre.

O **Fator de Impacto** atribuído a cada processo corresponde a um valor inteiro positivo que indica sua importância relativa no sistema. O fator de impacto de cada processo monitorado i, (I_i), juntamente com o identificador único do processo, compõem o conjunto $S \subset \Pi$ monitorado. Deste modo, os valores no conjunto S correspondem ao conjunto $\{\langle id_1, I_1\rangle, \langle id_2, I_2\rangle, \ldots, \langle id_k, I_k\rangle\}$ para cada processo $i \in S, 1 \leq i \leq k$.

Para cada conjunto S monitorado, o subconjunto $T_p^S(t)$ representa os processos que não são suspeitos pelo monitor p no instante t. Complementarmente, o conjunto $F_p^S(t)$ representa os processos considerados faltosos pelo monitor p no instante t.

O **Nível de Confiança**, ou ***Trust Level***, indica o nível de confiança do processo monitor p no conjunto de processos em S em um determinado instante, calculado por $TL_p^S(t)$. Representa a soma dos fatores de impacto dos processos não falhos, ou seja, $TL_p^S(t) = \sum_i (I_i), \forall i \in T_p^S(t)$

Cada monitor pode acompanhar subconjuntos diversos de processos, com diferentes níveis de confiança e fatores de impacto individuais. O conjunto S^* abrange os m subconjuntos únicos monitorados, indicado por $S^* = \{S_1, S_2, S_3, \ldots, S_m\}$.

Por fim, o **Limiar**, ou ***Threshold***, define o limite mínimo de confiabilidade para cada conjunto em S^*, matematicamente representado por $\{Th_1, Th_2, \ldots, Th_m\}$, no qual cada Th_i está relacionado ao nível mínimo de confiança necessário para um subconjunto de processos S_i.

O Th_S é usado pelo monitor para verificar a confiança nos processos dos subconjuntos em S^*. Se, para cada um dos m subconjuntos de $S^*, 1 \leq i \leq m$, o $TL_p^i(t) > Th_i$, então S^* é considerado confiável (*trusted*) no tempo t por p; caso contrário, S^* é considerado não confiável (*not trusted*).

Essa classe de algoritmo introduz o conceito de **Propriedade de Flexibilidade**, que denota a capacidade de o detector de falhas de tolerar uma certa margem de falhas ou falsas suspeitas, ou seja, a sua capacidade de considerar diferentes conjuntos de respostas que levam o sistema a estados de confiança.

Tabela 1. Conjunto S_1 com processos q_i e seus respectivos fatores de impacto.

Conjunto S_1 monitorado pelo processo monitor da Região 1
$\langle q_0, 10\rangle, \langle q_1, 10\rangle, \langle q_2, 10\rangle, \langle q_3, 10\rangle, \langle q_4, 10\rangle, \langle q_5, 10\rangle, \langle q_6, 10\rangle,$ $\langle q_7, 10\rangle, \langle q_8, 10\rangle, \langle q_9, 10\rangle, \langle q_{10}, 60\rangle, \langle q_{11}, 20\rangle, \langle q_{12}, 20\rangle, \langle q_{13}, 20\rangle$

A Tabela 1 apresenta um exemplo de conjunto de processos monitorados semelhante ao cenário da Figura 1. O conjunto S_1 do monitor da Região 1 compreende os processos q_0 a q_9 representando os sensores localizados na própria Região 1 (roxo), os processos q_{11} a q_{13} representando sensores remotos em monitoramento (verde), e q_{10} como o monitor da Região 2. O valor máximo de $TL_p^{S_1}(t)$ é 220, $\forall t > 0$, sendo 100 para os sensores locais e 120 para os sensores e monitor remotos. Nesta situação, o limiar escolhido deve refletir o objetivo do monitor. Por exemplo, para garantir que ao menos um processo de cada região sempre responda, tem-se $120 < Th_1 \leq 220$ e $TL_p^{S_1}(t) > Th_1, \forall t > 0$. O monitor da Região 2 pode adotar uma estratégia equivalente.

3.2. Métricas de Qualidade de Serviço (QoS)

A avaliação do Disaster-FD baseia-se no trabalho de [Chen et al. 2002], em que definem um conjunto de métricas para avaliar a qualidade de serviço (QoS - *Quality of Service*) dos algoritmos de detecção de falhas. Essas métricas são centradas em torno de restrições temporais, que se referem ao tempo necessário para detectar uma falha, ao tempo para corrigir uma suspeita incorreta e ao intervalo entre duas suspeitas falsas.

Especificamente, são adotadas as mesmas métricas de QoS utilizadas pelo Impact-FD. Essa escolha se deve ao fato de que o Disaster-FD herda conceitos fundamentais do Impact-FD, o que torna essas métricas particularmente relevantes para a nossa análise. Ao aplicar estas métricas ao Disaster-FD, busca-se não apenas manter a consistência com os métodos estabelecidos, mas também avaliar a eficácia do Disaster-FD dentro do mesmo quadro teórico. As métricas escolhidas são:

- **Tempo Médio de Detecção** (TD): Afere a rapidez e eficiência do sistema na detecção de falhas. O TD mede o período desde a falha de um processo q até que o detector de falhas em p comece a suspeitar continuamente de q. Esta métrica é crítica para entender como o sistema responde rapidamente a incidentes.
- **Taxa Média de Erros** (μR): Representa a frequência com que o detector de falhas comete erros por unidade de tempo, servindo como um indicativo da confiabilidade do detector. Esta métrica é particularmente importante para avaliar a propensão do sistema a falsos positivos ou falsos negativos.
- **Probabilidade da Acurácia** (PA): Avalia a probabilidade de que as saídas do detector de falhas sejam corretas em um momento aleatório, fornecendo uma medida da precisão geral do sistema ao longo do tempo, obtido a partir da duração total do período de falso positivo em relação ao tempo total em análise.

4. Estimativa de chegada de *heartbeats*

O mecanismo básico de monitoramento consiste no recebimento de mensagens periódicas dos processos monitorados, comumente denominadas *heartbeats* (HB). O método proposto por [Chen et al. 2002] para estimar a chegada do próximo *heartbeat* (EA_{k+1}) é baseado no histórico dos tempos de chegada dos *heartbeats* anteriores e inclui uma margem de segurança (β).

No cálculo de EA_{k+1}, o processo p considera uma janela deslizante com as w mensagens de *heartbeat* mais recentes recebidas do processo q representadas por $m_1, m_2, ..., m_w$. Os valores $T_1, T_2, ..., T_w$ são os respectivos tempos de recepção dessas mensagens, de acordo com o relógio local de p. Assim, conforme definido

em [Chen et al. 2002], tem-se $EA_{k+1} = \frac{1}{w}\sum_{i=k-w}^{k}(T_i - \Delta_i \times i) + (k+1) \times \Delta_i$, em que Δ_i corresponde ao intervalo de envio de dois *heartbeats* consecutivos.

Deste modo, o tempo esperado de chegada do *heartbeat* $k+1$, denominado τ_{k+1}, é definido como $\tau_{k+1} = EA_{k+1} + \beta$, e o não recebimento de um *heartbeat* até o tempo τ_{k+1} caracteriza o processo q como suspeito.

4.1. Comparação com o Impact-FD

Embora tanto o Disaster-FD quanto o Impact-FD utilizem a equação da Seção 4 para calcular o tempo estimado de chegada do próximo *heartbeat*, denominado EA_{k+1}, as duas propostas utilizam abordagens ligeiramente distintas na interpretação e implementação da equação.

A implementação do protocolo Disaster-FD utiliza o identificador de número de sequência da mensagem de *heartbeat* para calcular a diferença entre o tempo real de chegada e um tempo de chegada teórico, o qual definem como produto do identificador pelo intervalo fixo entre *heartbeats* consecutivos (Δ_i). O protocolo Impact-FD, por sua vez, não utiliza diretamente o número de sequência do *heartbeat*, empregando um índice incremental para calcular a diferença entre o tempo de chegada de cada *heartbeat* e o tempo de chegada esperado, baseado no Δ_i.

A metodologia proposta por Chen baseia-se no ajuste da estimativa de tempo de chegada usando o histórico dos tempos de chegada das w mensagens anteriores, considerando a diferença entre os tempos de chegada reais e os esperados, com base no intervalo regular entre os *heartbeats*.

Sendo assim, o Disaster-FD está mais alinhado com a teoria de Chen, pois ele incorpora diretamente o conceito de sequencialidade dos *heartbeats* (através do número de sequência), refletindo a abordagem de Chen de ajustar as estimativas com base nas diferenças entre os tempos de chegada reais e os esperados. Já o Impact-FD, embora similar em estrutura, não captura a sequencialidade de forma tão direta, utilizando um índice incremental que pode não representar com precisão a sequência real dos *heartbeats*.

Na prática, conforme observado ao analisar os registros de experimentos do Impact-FD, isto significa que o Impact-FD tem mais dificuldade em lidar com "buracos" na sequência de *heartbeats*, o que ocorre quando algumas mensagens são perdidas. O efeito disso é que a implementação do Impact-FD necessita de um tempo superior para detectar um falso positivo, ou seja, perceber que suspeitou erroneamente de um processo correto.

5. Implementação

O Disaster-FD foi implementado em Java e utiliza a biblioteca Californium [Kovatsch et al. 2014]. Ele se destaca por seu monitoramento federado e multi-protocolo, empregando requisições CoAP (*Constrained Application Protocol*) para monitoramento de dispositivos IoT e o protocolo ICMP (*Internet Control Message Protocol*) para monitorar nós monitores de regiões federadas vizinhas. Essa abordagem dual fornece flexibilidade e uma análise mais completa do estado da rede.

As solicitações CoAP do tipo "CON" (*Confirmable*) aumentam a confiabilidade na comunicação, pois cada mensagem prevê uma resposta do dispositivo receptor. Além

disso, o sistema implementa um mecanismo de *timeout* para estas solicitações, assegurando que o monitoramento continue eficiente mesmo quando um dispositivo não responde dentro do tempo esperado.

O Disaster-FD utiliza a biblioteca Californium para tratar as respostas das requisições CoAP de forma assíncrona, com tratamento tanto dos casos de retorno com sucesso da mensagem de *heartbeat* quanto de erros, tais como estouro do tempo calculado para o recebimento do próximo *heartbeat* e problemas de conectividade, permitindo manter o fluxo contínuo de monitoramento e análise ininterruptos.

5.1. Implementação do Rastreamento de Mensagens CoAP

Inicialmente, cada dispositivo IoT é acessível por meio de uma URI (*Uniform Resource Identifier*) única, baseada em seu endereço IPv6 específico. Utilizando essa URI, o cliente CoAP na implementação do Disaster-FD deriva um identificador único de dispositivo, utilizado na geração do valor inicial do número de sequência das mensagens, ou MID (*Message* ID).

O MID é gerado ou recuperado para cada dispositivo e incrementado a cada nova requisição, assegurando unicidade e rastreabilidade das mensagens. Nas requisições CoAP do tipo GET e CON (Confirmable), o MID é explicitamente definido no cabeçalho, permitindo correlacionar precisamente as requisições enviadas e as respostas recebidas.

6. Resultados

O presente estudo implementa e testa o sistema de detecção de falhas Disaster-FD no ambiente de Internet das Coisas (IoT) denominado FIT-IoTLAB [Adjih et al. 2015], monitorando duas regiões interconectadas, Grenoble e Strasbourg, em uma configuração similar ao exemplo da Figura 1. A escolha por apenas estas 2 regiões justifica-se por problemas de conectividade entre as demais regiões da infraestrutura, provavelmente relacionados a configurações de *firewall*.

O experimento foi estruturado em torno do monitoramento de 14 processos pelo monitor em cada região, sendo 10 sensores localizados na própria região do monitor e 3 sensores na região vizinha, além do processo monitor na região vizinha. Cada monitor utiliza o protocolo CoAP para os sensores e ICMP para acompanhar outros monitores.

As requisições aos sensores foram feitas utilizando o método GET do protocolo CoAP a um intervalo de 5.000ms. Esta taxa foi cuidadosamente selecionada para equilibrar a eficiência na detecção de falhas, a minimização da carga na rede e o consumo de energia nos dispositivos. A escolha da margem β no cálculo do tempo estimado do próximo *heartbeat* também foi definida com base em testes de latência de envio e recebimento de mensagens para os dispositivos nas duas regiões escolhidas. Após 24 horas de teste, escolheu-se o valor que corresponde à média dos tempos de resposta, acrescidos do desvio padrão calculado, correspondendo a um valor de 1.500ms.

Este estudo visa não apenas testar a eficácia do Disaster-FD em detectar falhas em tempo real em um ambiente IoT, mas também explorar as interações entre dispositivos de rede em regiões federadas.

A configuração dos fatores de impacto em cada região foi definida de maneira similar àquela apresentada na Tabela 1, a saber: os 10 sensores que localizam-se na área

local de cada região foram estabelecidos com um fator de impacto de 10, enquanto os 3 sensores na região vizinha receberam um fator de impacto de 20. Adicionalmente, o monitor na região vizinha tem um fator de impacto atribuído de 60, refletindo sua importância no monitoramento da região. Consequentemente, o nível de confiança para este conjunto de processos pode chegar no máximo a 220 ($10 \times 10 + 3 \times 20 + 60$).

6.1. Erros Acumulados por dispositivo - Região Strasbourg

A Figura 2 mostra a quantidade acumulada de erros ocorridos por dispositivo monitorado durante as 24 horas de monitoramento na região de Strasbourg. Os dispositivos de 0 a 9 correspondem a sensores locais, os dispositivos 11 a 13 correspondem a sensores na região de Grenoble e o dispositivo 10 representa o monitor na região de Grenoble.

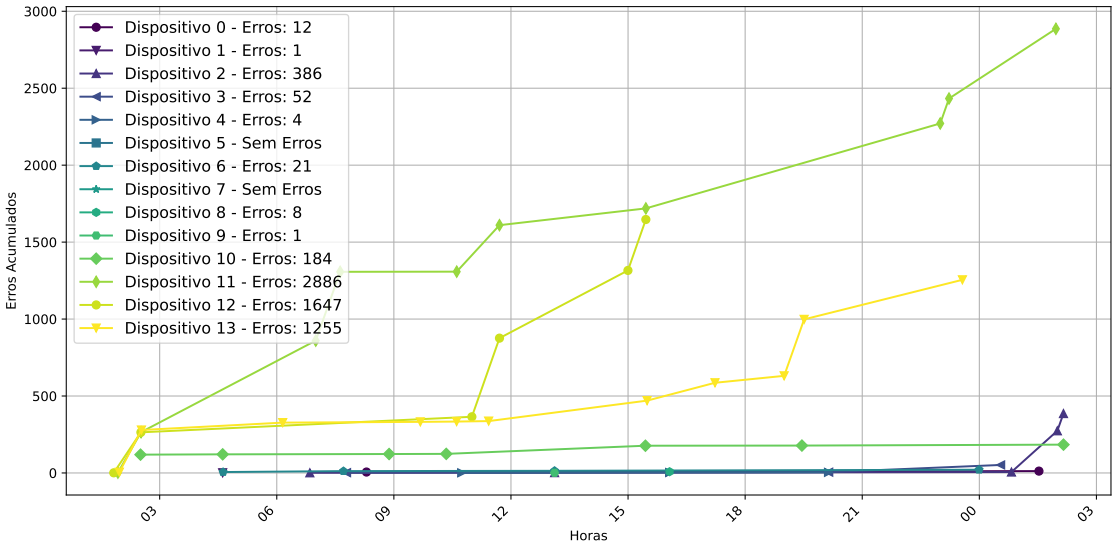

Figura 2. Erros Acumulados por dispositivo na Região de Strasbourg.

O detector não registrou falhas para os dispositivos 5 e 7, enquanto os dispositivos IoT (0, 1, 2, 3, 4, 6, 8 e 9) também apresentaram poucos erros, o que reflete a estabilidade da região de Strasbourg. Em contraste, na região vizinha de Grenoble, tanto os dispositivos IoT monitorados (11, 12 e 13) quanto o monitor (dispositivo 10) apresentaram um número significativo de erros, indicando instabilidade nessa área.

Notavelmente, o dispositivo 10, que representa o monitor na região de Grenoble, registrou vários erros. Este dispositivo atua também como o nó central ou roteador de borda e é fundamental para a gestão do tráfego de rede em Grenoble. Os erros observados no dispositivo 10 sugerem problemas na conectividade, comprometendo a comunicação e a eficácia operacional da rede IoT da região. A instabilidade do dispositivo 10 é, portanto, um fator crítico que pode afetar a performance da rede de Grenoble, causando interrupções no funcionamento e falhas na resposta às requisições enviadas pelo monitor Disaster-FD.

6.2. Erros Acumulados por dispositivo - Região Grenoble

De maneira similar à seção anterior, a Figura 3 mostra a quantidade acumulada de erros ocorridos por dispositivo monitorado durante as 24 horas de monitoramento na região de

Grenoble. Os dispositivos de 0 a 9 correspondem a sensores locais, os dispositivos 11 a 13 correspondem a sensores na região de Strasbourg e o dispositivo 10 representa o monitor na região de Strasbourg.

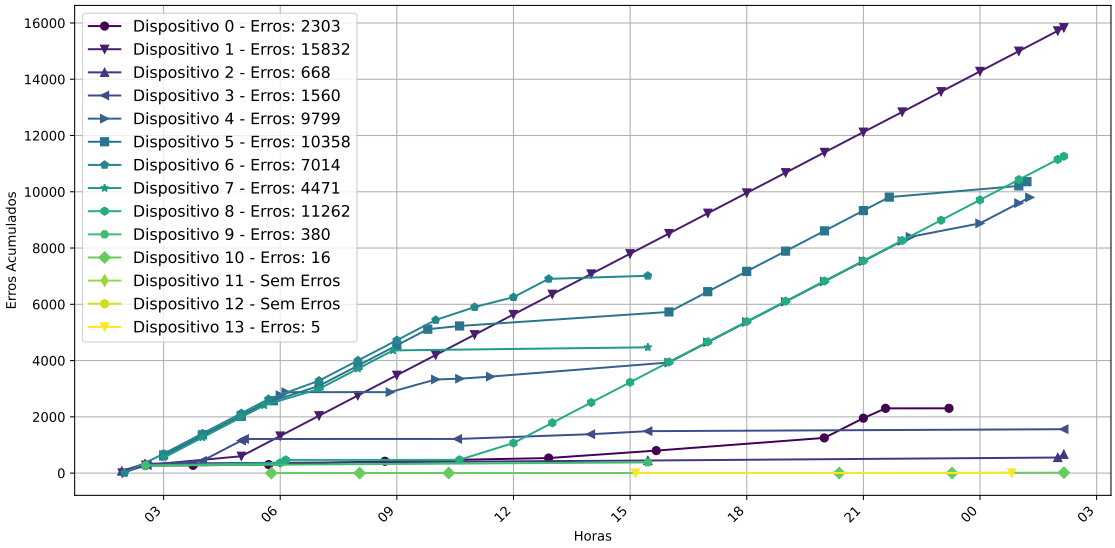

Figura 3. Erros Acumulados por dispositivo na Região de Grenoble.

Os resultados fornecidos pelo detector de falhas Disaster-FD reforçam o observado na seção anterior e na Figura 2, especialmente com relação à instabilidade da rede na região de Grenoble. Enquanto o monitor de Strasbourg identificou instabilidade em Grenoble, o monitoramento em Grenoble também detectou uma alta incidência de erros em seus próprios dispositivos (dispositivos 0 a 9) na Figura 3 e poucos em Strasbourg (dispositivos 10 a 13) na Figura 3, corroborando a percepção mútua de desempenho da rede entre as duas regiões.

6.3. Tempos efetivo *vs.* Tempo Estimado de chegada

A Figura 4 fornece uma análise do comportamento do dispositivo 5, situado em Strasbourg, durante o período de monitoramento de 24 horas. A escolha deste dispositivo para análise justifica-se pela sua robustez operacional, como evidenciado por sua performance estável, que também é destacada na Figura 2.

Conforme discussão na Seção 6, o monitor adotou uma margem de segurança de 1500 milissegundos e intervalos de 5000 milissegundos para o envio de requisições. Uma análise comparativa entre os tempos estimados (em vermelho) e os tempos reais de chegada (em azul) dos *heartbeats* revelou estabilidade, demonstrada pela pequena variação nos intervalos de tempo entre a estimativa e a recepção efetiva dos heartbeats. Além disso, a prevalência de pontos vermelhos sobre os pontos azuis indicam que os *heartbeats* chegaram ao monitor antes do tempo estimado.

6.4. Estatística da Rede em Strasbourg

O experimento foi conduzido sob um valor estabelecido de *threshold* de 160, que serve como um limiar de segurança para a rede nas regiões de Grenoble e Strasbourg, conforme

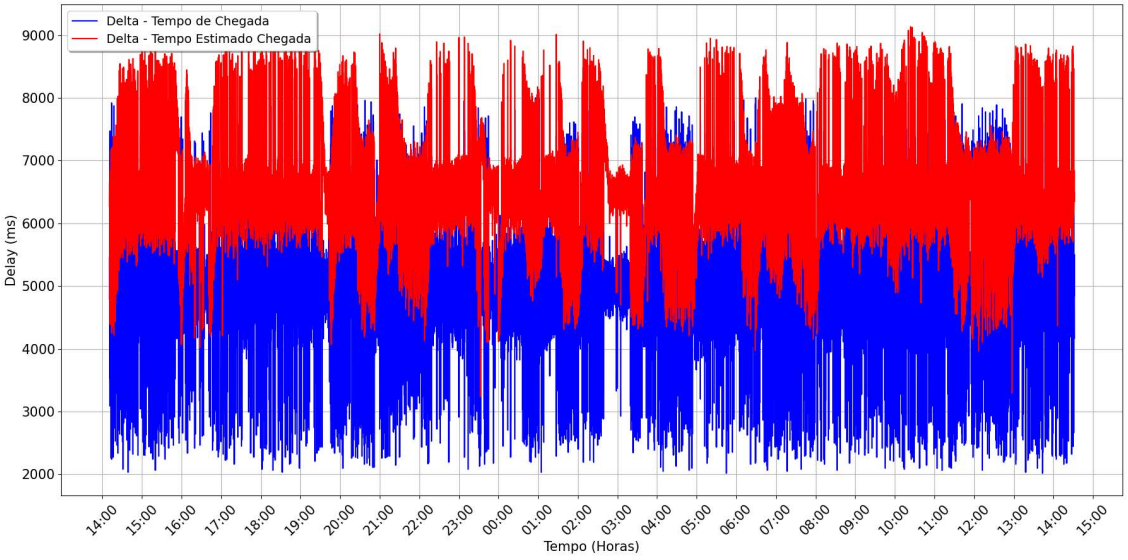

Figura 4. Tempo de chegada e Tempo Estimado para Dispositivo 5 (Strasbourg).

detalhado na discussão sobre os valores da Tabela 1 na Seção 3.1. Este valor de 160 corresponde, por exemplo, a situações em que todos os nós de Strasbourg não são suspeitos e pelo menos o monitor em Grenoble está respondendo.

A análise, retratada na Figura 5, proporciona uma perspectiva estatística sobre o comportamento da rede com respeito ao nível de confiança ($TL_p^S(t)$) (curva vermelha). Este é um indicador composto, refletindo a soma dos impactos individuais de cada dispositivo na confiabilidade da rede.

Especificamente, o nível de confiança é influenciado pelos parâmetros de estabilidade da rede e, durante o período de testes, a precisão do sistema Disaster-FD mostrou uma melhoria progressiva à medida que os dispositivos monitorados mantiveram operações estáveis.

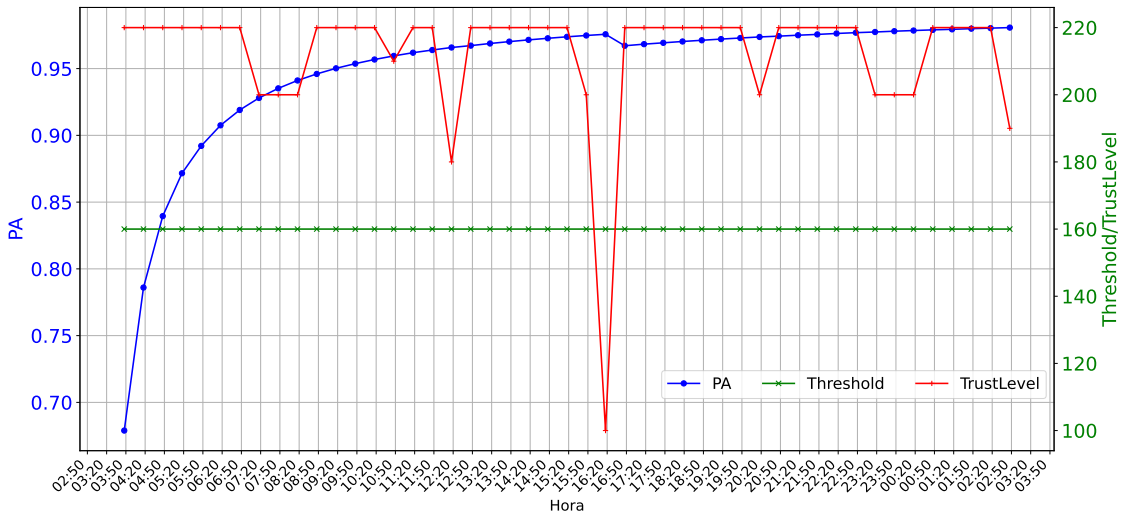

Figura 5. Estatística da rede em Strasbourg.

A Figura 5 mostra ainda que a Probabilidade de Acurácia (PA), identificada pela curva azul, permanece acima de 95% a partir 8:50, realçando a capacidade do monitor em indicar com precisão o estado do sistema.

Por fim, ressalta-se, pela análise da mesma figura, que as regiões em análise foram consideradas não confiáveis no intervalo entre 15:50 e 16:50, período em que o nível de confiança caiu abaixo do limiar estabelecido. Em um cenário real, este fato poderia indicar um possível desastre, disparando ações de emergência para as regiões afetadas.

Complementarmente, a Tabela 2 apresenta dois pontos críticos extraídos dos registros do monitor em Strasbourg em que o nível de confiança (*trust level*) registrou um valor de 100, ou seja, abaixo do limiar estabelecido. Notavelmente, a sequência de "Vetor de dispositivos" 'TTTTTTTTTTFFFF' indica uma falha (valor 'F') nos quatro dispositivos da rede vizinha em Grenoble, dispositivos 10 a 13, com fatores de impacto de 60, 20, 20 e 20, respectivamente.

Tabela 2. Extrato dos *logs* para os monitores em Strasbourg e Grenoble.

Região	Nível de Confiança	Dispositivos	P.A.	Hora
Strasbourg	100	TTTTTTTTTTFFFF	0.8061	04-01-2024 03:17
Strasbourg	100	TTTTTTTTTTFFFF	0.9756	04-01-2024 16:17
Grenoble	120	FFFFFFFFFFTTTT	0.9963	04-01-2024 03:17
Grenoble	120	FFFFFFFFFFTTTT	0.9676	04-01-2024 16:17

6.5. Estatística da Rede em Grenoble

A Figura 6 corrobora os resultados apresentados na Figura 3, indicando que a rede da região de Grenoble experimentou períodos de instabilidade. Esta conclusão é evidenciada pela análise do nível de confiança (*Trust Level*, em vermelho), que em diversos momentos aproximou-se do limiar de segurança estabelecido para a rede (em verde).

Além disso, conforme exposto na Tabela 2, houve falhas nos dispositivos locais de Grenoble, resultando em uma classificação de rede não confiável. Isso pode ser observado pelo valor do "Vetor de dispositivos" complementar, exibido na Tabela 2 para o mesmo horário na região de Grenoble, ou seja, os 10 dispositivos em Grenoble são percebidos como faltosos ('F') pelo monitor na própria região.

7. Trabalhos Relacionados

Esta seção apresenta um resumo do levantamento bibliográfico focado na Internet das Coisas (IoT), com ênfase especial em tecnologias para detecção de falhas, monitoramento e interação entre dispositivos.

Impact-FD [Rossetto et al. 2018] é uma solução que propõe um novo detector de falhas denominado Impact FD, que fornece uma saída expressando a confiança do detector de falhas em relação ao sistema (ou conjunto de processos) como um todo. A confiança é configurada pelo fator de impacto, permitindo ao usuário definir a importância de cada nó dentro de uma margem aceitável de falhas. Além disso, são definidas algumas propriedades de flexibilidade que caracterizam a capacidade do Impact FD de tolerar uma certa margem de falhas ou suspeitas. Disaster-FD estende o Impact-FD, com foco na

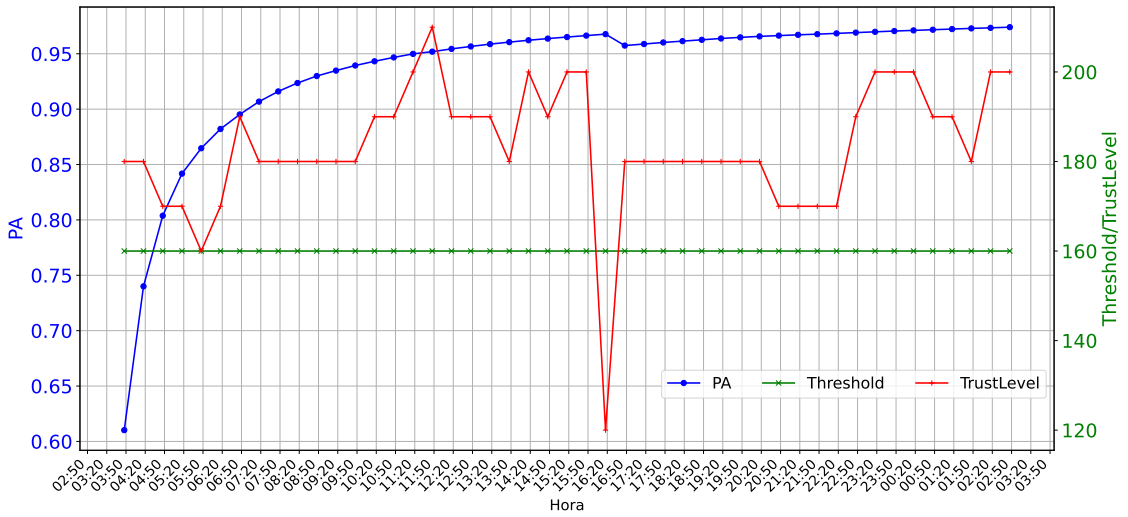

Figura 6. Estatística da rede Grenoble.

detecção de falhas e na avaliação da confiabilidade da rede em ambientes IoT, e ênfase no monitoramento em tempo real. Além disso, o Disaster-FD realiza monitoramento federado abrangendo diferentes regiões, levando em consideração aspectos adicionais como o consumo de energia dos dispositivos.

Medley [Yang et al. 2019] é uma solução descentralizada para detecção de falhas em IoT operando em redes ad-hoc, empregando seleção espacial para enviar mensagens de *ping*, priorizando nós próximos. Utiliza múltiplos votos de dispositivos diversos para avaliar a condição de um nó para reduzir falsos positivos. O protocolo SWIM, integrado ao Medley, assegura a detecção de falhas de forma escalável e tolerante. Disaster-FD concentra-se no monitoramento em tempo real e análise de redes IoT, especialmente em situações de desastre, priorizando a avaliação da confiabilidade do sistema.

Por fim, Stab-FD [Sens et al. 2024] é uma solução que propõe um detector de falhas para sistemas distribuídos, especialmente adequado para redes de área ampla (WAN), que se ajusta dinamicamente dentro de uma margem de segurança para adaptar-se às variações na qualidade dos links de comunicação. Além disso, o Stab-FD possui uma versão cooperativa, onde os nós trocam informações sobre a estabilidade dos links e a lista de nós suspeitos [Sens et al. 2024]. Disaster-FD se destaca em ambientes IoT, especialmente em cenários de desastre, utilizando um algoritmo para avaliar a confiabilidade da rede. Técnicas propostas no Stab-FD para adaptação e redução de falsos positivos são ortogonais e poderiam ser aplicadas ao Disaster-FD.

8. Conclusão

Disaster-FD demonstrou resultados satisfatórios na detecção de falhas para redes IoT. Através da incorporação de funcionalidades do Impact-FD, do desenvolvimento de capacidades de monitoramento em tempo real e avaliação contínua da confiabilidade da rede, mostrou ser uma ferramenta valiosa. Os testes realizados na plataforma IoT-LAB confirmaram sua capacidade de operar de forma confiável em cenários desafiadores, tornando-o uma ferramenta potencial de apoio na detecção de desastres. Trabalhos futuros prevêem a comparação com outros detectores de falhas no mesmo ambiente.

Agradecimentos

Esta pesquisa recebeu apoio financeiro da CAPES, através da chamada STIC/AmSud, registrada no processo número 88881.368742/2019-01.

Referências

Adjih, C., Baccelli, E., Fleury, E., Harter, G., Mitton, N., Noel, T., Pissard-Gibollet, R., Saint-Marcel, F., Schreiner, G., Vandaele, J., et al. (2015). FIT IoT-LAB: A large scale open experimental IoT testbed. In *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*, pages 459–464. IEEE.

Aguilera, M. K., Delporte-Gallet, C., Fauconnier, H., and Toueg, S. (2004). Communication-efficient leader election and consensus with limited link synchrony. In *Proceedings of the twenty-third annual ACM symposium on Principles of distributed computing*, pages 328–337.

Atzori, L., Iera, A., and Morabito, G. (2010). The internet of things: A survey. *Computer networks*, 54(15):2787–2805.

Chandra, T. D., Hadzilacos, V., and Toueg, S. (1996). The weakest failure detector for solving consensus. *Journal of the ACM (JACM)*, 43(4):685–722.

Chandra, T. D. and Toueg, S. (1996). Unreliable failure detectors for reliable distributed systems. *Journal of the ACM (JACM)*, 43(2):225–267.

Chen, W., Toueg, S., and Aguilera, M. K. (2002). On the quality of service of failure detectors. *IEEE Transactions on computers*, 51(5):561–580.

Cristian, F. and Fetzer, C. (1999). The timed asynchronous distributed system model. *IEEE Transactions on Parallel and Distributed Systems*, 10(6):642–657.

Fischer, M. J., Lynch, N. A., and Paterson, M. S. (1985). Impossibility of distributed consensus with one faulty process. *Journal of the ACM (JACM)*, 32(2):374–382.

Janoneda, L. (2022). A cada desastre natural no brasil, em média, 3,4 mil pessoas são afetadas. Acessada: 2022-01-09.

Kovatsch, M., Lanter, M., and Shelby, Z. (2014). Californium: Scalable cloud services for the internet of things with coap. In *2014 International Conference on the Internet of Things (IOT)*, pages 1–6. IEEE.

Rossetto, A. G. d. M., Geyer, C. F., Arantes, L., and Sens, P. (2018). Impact fd: An unreliable failure detector based on process relevance and confidence in the system. *The Computer Journal*, 61(10):1557–1576.

Sens, P., Arantes, L., Rosseto, A. G. D. M., and Marin, O. (2024). Stab-fd: A cooperative and adaptive failure detector for wide area networks. *Journal of Parallel and Distributed Computing*, 186.

Verissimo, P. and Rodrigues, L. (2012). *Distributed Systems for System Architects*, volume 1. Springer Science & Business Media.

Yang, R., Zhu, S., Li, Y., and Gupta, I. (2019). Medley: A novel distributed failure detector for IoT networks. In *Proceedings of the 20th International Middleware Conference*, pages 319–331.