Um Mecanismo Adaptativo de Descoberta de Serviços para Internet das Coisas Centrado no Consumo Energético

Leonardo L. M. da Silva[1,2], **Robson G. de Melo**[1,2], **Aldri Santos**[1]

[1]Núcleo de Redes Sem-Fio e Redes Avançadas (NR2) – UFPR

[2]Núcleo de Redes Inteligentes e Sistemas Computacionais (RISC) – UNEMAT

{leoluan,robinho}@unemat.br, aldri@inf.ufpr.br

Abstract. *On the Internet of things, several techniques have been applied to perform the demand for services and guarantee the maintenance of the offered services. One of the popular techniques handles the usage of service repositories, which provide registration, update and service request messages. Though, the number of refresh messages exchanged between devices and repository impacts on the power of the devices, and consequently it decreases the availability of services. This paper introduces a mechanism for service discovery, called MADS, which adjusts the period of initialization of services taking into account the device's energy consumption. The operation routines show MADS reduces the number of service refresh messages required between device and repository. It also rises the service availability, and offers one rating of the availability indicator. In contrast to classic approaches, MADS gets to reduce up to 90% the exchange of refresh messages, and keeps up to 75% more available than others.*

Resumo. *A Internet das Coisas demanda pelo meios do gerenciamento da descoberta de serviços e pela garantia da manutenção dos serviços providos. Uma das possíveis abordagens empregadas na descoberta de serviços consiste no uso de repositórios de serviços, que suportam o registro, atualização e requisição de serviços. Contudo, o número de mensagens de atualização trocadas entre os dispositivos e o repositório causam impacto na carga energética dos dispositivos, e assim diminuem a disponibilidade dos serviços. Este trabalho propõe um mecanismo de descoberta de serviço, denominado MADS, que controla o período de atualização dos serviços de acordo com o consumo energético do dispositivo. As rotinas de execução mostram que o MADS diminui o número de mensagens de atualização de serviço trocadas entre dispositivo e repositório. Ele também tem melhorado a disponibilidade de serviços, além de oferecer uma classificação do indicador de disponibilidade. Em comparações com abordagens clássicas, MADS chega a reduzir até 90% a troca de mensagens de anúncio de serviços e ser até 75% mais disponível que outras soluções.*

1. Introdução

O paradigma da Internet das Coisas (IoT) se popularizou com o crescimento do número de dispositivos conectados à Internet [Atzori et al. 2010]. No ecossistema de IoT, uma grande rede pervasiva conecta uma quantidade sem precedentes de coisas e pessoas [Al-Fuqaha et al. 2015]. Porém, essa grande rede de conexões impõe novos desafios,

como, por exemplo, a descoberta de serviços. A descoberta de serviços é um dos fundamentos do ecossistema da IoT, pois por meio dela, usuários e objetos podem ter acesso aos serviços disponibilizados na rede por objetos até então desconhecidos.

O processo de descoberta de serviços demanda que dispositivos que ofereçam serviços na rede sejam encontrados e registrados de modo que seus serviços possam ser consumidos, e isso deve-se a algumas características dos dispositivos no ecossistema de IoT [Mohiuddin et al. 2014]. Primeiro, a maioria dos dispositivos são energeticamente limitados e, portanto, entram em modo de economia de energia, desligando seus rádios para economizar bateria. Segundo, em razão da mobilidade e dinamicidade da entrada e saída de dispositivos na rede ser grande, esses dois aspectos impõem aos mecanismos de descoberta de serviços a necessidade de atualização periódica da lista de serviços registrados, garantindo que esses serviços estejam sempre disponíveis para seus consumidores. A necessidade de atualização torna-se um problema devido ao grande número de mensagens trocadas no processo de anúncio, e causa grande consumo energético dos dispositivos e interfere na disponibilidade dos serviços registrados.

Os trabalhos na literatura têm empregados diferentes abordagens para esse problema. Em [Albalas et al. 2017] é utilizada uma abordagem baseada na *sequência de Fibonacci*. Nele os dispositivos anunciam seus serviços periodicamente em um *Diretório de Serviços*, que é responsável por registrar os serviços anunciados e atender requisições de busca por esses serviços. Para reduzir o número de mensagens de atualização, o intervalo de tempo entre anúncios é incrementado sempre que a carga energética do dispositivo decresce. Contudo, diante de uma possível recarga energética do dispositivo, o algoritmo estagna em um mesmo período de atualização, desconsiderando a nova condição energética. Outra limitação é a proporção fixa de aumento do período de atualização, não refletindo a situação energética de dispositivos com fonte energética abundante.

As abordagens para descoberta de serviço no ecossistema de IoT devem permitir que consumidores possam encontrar os fornecedores de serviços sem causar uma sobrecarga de mensagens na rede. Essa sobrecarga inviabiliza a utilização dos serviços por diversos fatores como o aumento do congestionamento de pacotes, diminuição do QoS da rede, entre outros [Aziez et al. 2017]. Outro aspecto importante trata da garantia miníma de disponibilidade do serviço, para que a medida que os consumidores desejem utilizar determinado serviços na rede, esses serviços estejam ativos e operantes com condições plenas de atendimento [Gigli and Koo 2011, Gubbi et al. 2013].

Este trabalho apresenta um mecanismo para solucionar o alto número de mensagens de anúncio na rede e combater o desperdício energético, que consiste no uso do *Mecanismo Adaptativo de Descoberta de Serviços (MADS)*, que ajusta periodicamente o tempo de atualização dos serviços de acordo com o consumo energético do dispositivo. O MADS também busca solucionar a falta de um indicador de disponibilidade dos serviços registrados por usar o período de atualização como índice de disponibilidade dos serviços registrados. Na medida em que um serviço apresenta um menor intervalo entre anúncios de atividade, maior será sua disponibilidade. A avaliação foi realizada com a implementação de rotinas de execução que mostram que o MADS diminui o número de mensagens de atualização de serviço trocadas entre dispositivo e repositório. O número de mensagens trocadas alterou de acordo com cenário de variação de carga energética. O MADS também foi eficaz na melhora dos indicadores de disponibilidade de serviços, garantido

que os serviços oferecidos na rede estejam disponíveis no momento de sua descoberta.

O restante deste trabalho está organizado da seguinte forma: A Seção 2 apresenta brevemente os conceitos e fundamentais sobre a descoberta de serviços em IoT. A Seção 3 apresenta os trabalhos relacionados. A Seção 4 detalha o MADS, bem como ilustra seu funcionamento. A Seção 5 apresenta a metodologia de avaliação do MADS e os resultados obtidos. Por fim, a Seção 6 apresenta as conclusões.

2. Visão Geral de Descoberta de Serviço em IoT

A descoberta de serviços tem por objetivo detectar de modo automático os serviços oferecidos em um escopo de rede [De et al. 2011]. No contexto de IoT, dispositivos chamados de **provedores de serviço** oferecem esses serviços. Os mecanismos de **descoberta de serviços** buscam, armazenam e organizam descrições dos serviços oferecidos na rede, indicando-os quando um solicitação é feita. Os **consumidores de serviços** utilizam essas descrições para terem acesso aos serviços de vários dispositivos desconhecidos. Existem inúmeros trabalhos na literatura que apresentam abordagens relacionadas à descoberta de serviços no contexto de IoT. O trabalho de [Broering et al. 2016] apresenta uma categorização dos mecanismos de descoberta de serviços baseada no escopo da busca *ao redor do nó consumidor*, no *escopo da rede*, e *em diretórios*, como retratado na Figura 1.

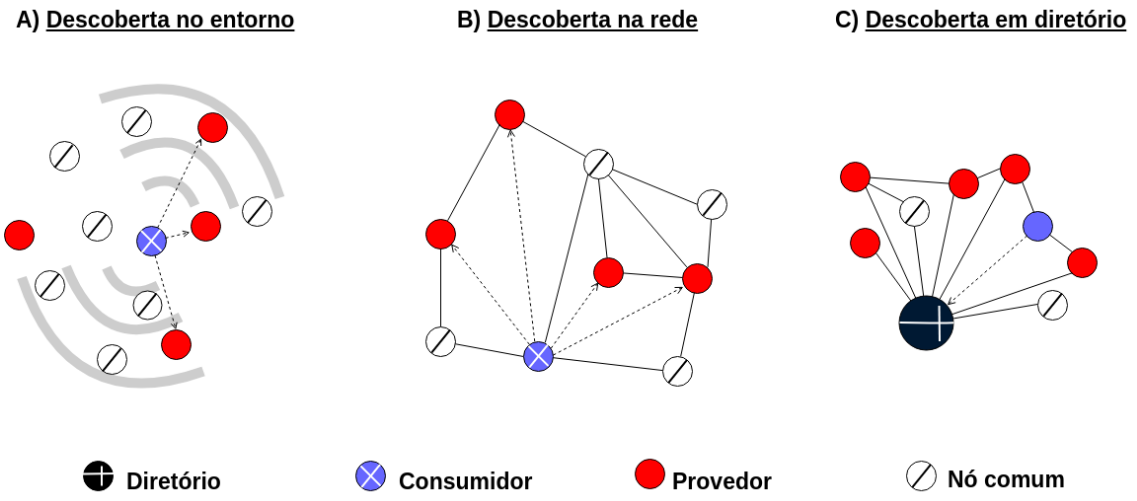


Figura 1. Escopos de descoberta de serviços

A categoria de descoberta de serviços no *entorno do nó consumidor* considera a proximidade espacial dos serviços e, portanto, está relacionado a mecanismos de rede sem-fio, em que o dispositivo consumidor observa em seu entorno como mostrado na Figura 1-A. Na categoria de descoberta de serviços no *escopo de rede*, as técnicas empregadas buscam serviços em toda a rede em que o consumidor está presente. As técnicas dessa categoria geralmente trabalham em escopo local, identificando todos os provedores de serviços da rede local através de mensagens *multicast*, como representado na Figura 1-B. Por fim, a categoria de *busca em diretório*, representada na Figura 1-C. Neste modelo estão classificadas as técnicas de descoberta e registro de serviços em diretórios, que podem ser dispositivos dedicados ou não. Assim, os diretórios de serviços são responsáveis pelos serviços propriamente dito e por receber as requisições de busca por eles.

3. Trabalhos Relacionados

Na literatura diferentes abordagens são apresentadas como forma de debater a descoberta de serviços no contexto de IoT. O trabalho de [Kaiser et al. 2016] aborda a descoberta de serviços usando *mDNS (Mobile DNS) e DNS-SD (DNS-Service Discovery)*. O objetivo é contrapor soluções baseadas em *multicast*, por não serem escaláveis e aprimorar o uso de DNS na descoberta de serviços. Eles propõem uma técnica de descoberta distribuída em *NS (Naming Servers)* sincronizados usando o algoritmo *Raft*. Os *NS* podem assumir a função de diretório de serviços e passam a ser chamados de *SNS (Scope Name Servers)*. Esses *SNS* respondem as requisições de registro e recuperação de serviços. Contudo, em redes maiores, com maior número de nós, o processo de sincronização dos *SNS* gera sobrecarga na rede.

Em [Ishaq et al. 2012], o objetivo consistiu na implantação e descoberta automática de serviços na rede. Eles propõem um mecanismo hierárquico autoconfigurável, em que o principal elemento é o *Gateway* de sensores. Esse *gateway* é responsável por buscar informações sobre todos os nós presentes na rede. Assim, ele envia uma mensagem *multicast* periodicamente a todos os nós da rede, requisitando informações para manutenção da lista de serviços descobertos. A informação é recuperada pelo protocolo *CoAP - Constrained Application Protocol* e os nós que oferecem serviços são registrados em um DNS local. O trabalho também propõe a integração entre *CoAP* e *HTTP* usando outro *gateway*, de mais alto nível, responsável por traduzir as mensagens entre os protocolos. O trabalho não apresenta análise detalhada dos resultados obtidos, porém, adianta que a abordagem *pull-based*, em que o nó detentor do serviço é passivo, tem problemas de escalabilidade, principalmente em relação a abordagem *push-based*, devido ao uso intenso de *multicast*.

O trabalho de [Helal and ElMougy 2015] apresenta uma solução distribuída e energeticamente eficiente utilizando *P2P-DHT* e *CoAP*. O objetivo é aprimorar a abordagem centralizada usando *gateway* dedicado para evitar a existência de um único ponto de falha. Uma arquitetura hierárquica onde alguns dos nós comuns da rede atuando como *Cluster Heads* mantêm registro de serviços disponibilizados por nós vizinhos. Nesta abordagem híbrida, nós comuns e *gateway* coexistem e dividem responsabilidades no processo de descoberta, resultando em uma arquitetura capaz de diminuir o consumo energético da rede. No entanto, os resultados apontam para uma dependência dos *gateway* dedicados para comunicação de um salto, sendo que ausência deles o consumo energético é maior e a taxa de resposta à requisições diminui.

Em [Kim et al. 2017], os autores propõem o uso de *DNS-SD* para a descoberta de serviços a fim de aprimorar o uso do DNS no processo de descoberta de serviços abrindo mão do *mDNS*, muito usado em conjunto com o *DNS-SD*. A solução apresentada no trabalho inclui a utilização da mensagem de resposta usando protocolo DHCP ao *gateway* da rede, com o nome do dispositivo que oferece o serviço e a lista de serviços oferecidos, que são registrados em um servidor DNS. Os dispositivos consumidores obtém a lista de dispositivos registrados do servidor DNS. Com isso, deve escolher um dispositivo específico e solicitar ao servidor DNS a lista de serviços oferecidos pelo dispositivo escolhido. Ao obter a lista de serviços do dispositivo, o consumidor faz uma requisição *CoAP* ao dispositivo servidor solicitando um de seus serviços. Essa proposta tem resultados conceitualmente interessantes devido ao modelo de interação entre dispositivos e na utilização do protocolo *DNS-SD*. Porém, apresenta limitações na manutenção de serviços,

dado que os dispositivos consumidores precisam escolher um dispositivo sem nenhuma descrição de seus serviços além do valor semântico do nome do dispositivo servidor.

O trabalho de [Barreto et al. 2017] apresenta uma abordagem considerando o contexto da aplicação. Para diminuir o número de dispositivos que oferecem um determinado serviço, considera o contexto dos serviços disponíveis. Esse contexto se refere à localização geográfica e o histórico de uso. Eles partem da hipótese que, ao diminuir o número de dispositivos resultantes do processo de descoberta, reduziria-se o número de mensagens trocadas entre o consumidor e os possíveis provedores. Embora a hipótese provou-se verdadeira, visto que os resultados apontam para uma diminuição no número de mensagens trocadas, o tempo de resposta aumentou devido ao refinamento da descoberta pelo mecanismo de contextualização baseado na localização e no histórico de uso do consumidor.

Um mecanismo de descoberta de serviços deve ser eficiente na troca de mensagens com os dispositivos provedores para evitar o gasto desmedido de energia com registros e atualizações. Isso deixa de lado o primeiro objetivo do serviço: ser útil aos clientes. Da mesma maneira, esse mecanismo deve garantir a disponibilidade dos serviços registrados, pois o oferecimento de serviços que não estão disponíveis também causam desperdício energético, além de tornar o mecanismo de descoberta ineficiente. Por isso, um número elevado de mensagens trocadas durante o processo de anúncio do serviço é um sério problema no ecossistema de IoT [Meshkova et al. 2008].

O trabalho de [Albalas et al. 2017] trata o número elevado de mensagens com um algoritmo para adaptação do período de atualização adequado a cada dispositivo usando uma abordagem baseada na *algoritmo de Fibonacci*. Assim, os provedores de serviço aumentam o período de atualização ao longo da operação do dispositivo à uma taxa de diminuição baseada na sequência de *Fibonacci*. Espera-se que o período de atualização aumente em decorrência da redução da carga energética, logo, reduzindo o número de mensagens trocadas para o anúncio de serviços oferecidos na rede [Albalas et al. 2019]. No entanto, essa solução só pode ser considerada adaptativa no sentido de realizar ajustes em tempo de operação no período de atualização, sem considerar variáveis importantes [Khalil et al. 2018]. No caso de uma possível recarga energética do dispositivo, o algoritmo ficaria estagnado em um mesmo período de atualização, desconsiderando a nova condição energética do dispositivo. Outra questão é a proporção fixa de aumento do período de atualização, o que não reflete a situação energética de dispositivos com fonte energética abundante.

4. MADS: Mecanismo Adaptativo de Descoberta de Serviços em IoT

Esta seção descreve o *Mecanismo Adaptativo de Descoberta de Serviço* (MADS) para o ecossistema de IoT. O MADS atua com dois objetivos principais: *i)* diminuir o número de mensagens trocadas na descoberta de serviços e *ii)* fornecer um índice de disponibilidade de cada serviço descoberto. A exigência de atualização periódica gera um aumento no número de troca de mensagens, um alto consumo energético e consequentemente reduz a disponibilidade do serviço, tendo em vista que aumentam as chances do dispositivo provedor de serviço ficar indisponível por falta de energia.

No ecossistema de IoT, o processo de descoberta de serviço acontece de maneira diferente do tradicional [Cirani et al. 2014]. Os dispositivos não têm acesso a fonte energética, ou seja, tem carga energética limitada e por conta disso, entram em modo de eco-

nomia de energia quando ficam ociosos ou quando a carga energética está acabando. Esse comportamento diferencia o processo de descoberta de serviços dos métodos aplicados por ferramentas de busca tradicionais, por exemplo, que têm acesso ilimitado a energia e aos provedores de serviço, já que esses não possuem restrição energética. Contudo, na IoT, a maioria das soluções possuem uma abordagem diferente, na qual os dispositivos que oferecem um serviço são responsáveis por anunciar periodicamente o serviço oferecido na rede ou em um diretório específico. Esses dispositivos são chamados de *Agentes de Serviços (AS)*, e o diretório específico é denominado *Diretório de Serviços (DS)*. No *MADS*, o *AS* sempre anuncia seus serviços em um diretório específico e o *DS* mantém uma lista com todos os serviços e o *AS* responsável por cada serviço registrado.

O *MADS* utiliza uma abordagem adaptativa a fim de diminuir o número de mensagens trocadas durante o processo de descoberta de serviços. Nessa abordagem, o período de atualização de um serviço é ajustado em função da variação de carga energética do *AS* responsável, em relação ao último anúncio realizado. Quando o período para o próximo anúncio é ajustado em função da variação de carga energética, o processo de descoberta considera as condições do *AS*, garantindo maior tempo de operação do serviço. Caso o agente receba muitas requisições de serviço, a variação de carga energética é maior e, o período para o próximo anúncio é maior, reduzindo o número de troca de mensagens.

Em relação ao fornecimento do índice de disponibilidade de serviços, o *DS* realiza um procedimento de classificação dos serviços registrados. Essa classificação é feita através da ordenação crescente, do período de atualização informada em cada anúncio realizado pelo *AS*. O *DS* mantém os registros de serviços ordenados para que o solicitante da lista de serviços seja capaz de decidir qual serviço requisitar, de acordo com o índice de disponibilidade do serviço. Desse modo, evita-se que o solicitante faça requisição de serviços que estejam indisponíveis ou desatualizado. No *MADS*, a probabilidade de um serviço estar disponível é maior quando o período de tempo até a próxima atualização é menor. Isso se deve ao fato de que o período de tempo é definido em relação a variação de carga energética do *AS* responsável pelo serviço. Da mesma maneira, quanto maior é o período de tempo para o próximo anúncio, maior foi o gasto energético do *AS* e maior é a possibilidade dele se tornar indisponível.

4.1. Arquitetura do MADS

A arquitetura do MADS compreende dois principais módulos: O **Módulo de Análise Energética (MAE)** e o **Módulo de Manutenção de Serviços (MMS)**, como ilustra a Figura 2. O Módulo de Análise Energética é executado no *AS*, enquanto o Módulo de Manutenção de Serviços opera no *DS*. Esses módulos se interagem por meio do *índice de disponibilidade* do serviço (**ID**) que é encaminhado do **MAE** para o **MMS** a partir da mensagens de anúncio. A estrutura arquitetural do MADS é sucinta e facilmente realizável no ecossistema de IoT. As regras para o funcionamento do mecanismo se baseiam nas especificações do protocolo *CoAP* e do padrão *CoRE Resource Directory* [Barreto et al. 2017]. Ambos protocolos são especificados pela IETF (*Internet Engineering Task Force*) e amplamente adotados como padrão da indústria da IoT.

O **MAE** é responsável por determinar o **ID** com base no cálculo da variação de carga energética do *AS* que é obtida a partir da entrada do valor da carga atual e da a carga anterior. Só então é realizado o ajuste do período do próximo anúncio de atualização do

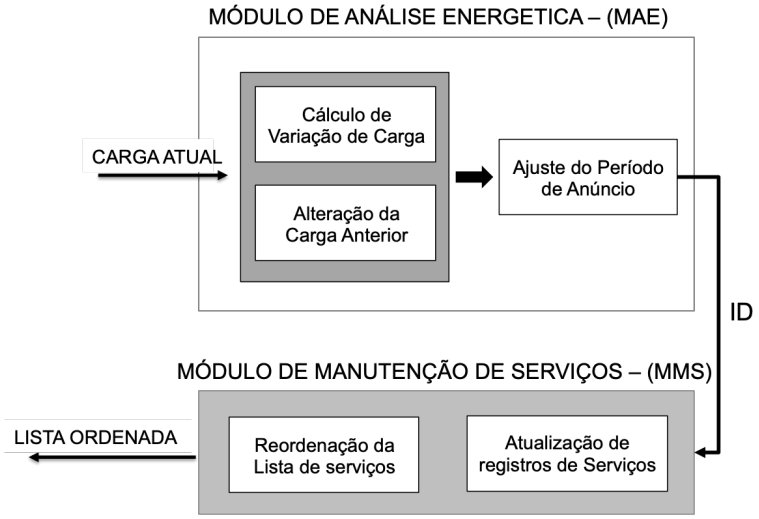


Figura 2. Arquitetura MADS

AS, usado para indicar o índice de disponibilidade. As variáveis para a operação desse módulo são:

- *Carga Energética Máxima* $(CMAX)$: corresponde ao capacidade máxima de carga energética do *AS*;
- *Carga Energética Imediata* (CI): corresponde ao valor da carga energética atual do *AS*;
- *Carga Energética Anterior* (CA): corresponde ao valor anterior da carga energética do *AS*;
- *Variação de Carga Energética* (ΔC): corresponde a diferença entre o *CA* e o *CI* de energia do *AS* como observado na Equação 1.

$$\Delta C = CA - CI \tag{1}$$

- *Prazo de Atualização* (PA): corresponde ao prazo da próxima atualização definida pela Equação 2.

$$PA = MomentoAtual + \Delta C \tag{2}$$

A operação do **MAE** pode ser dividida em duas etapas. A primeira consiste em calcular o valor da *CI* realizando uma medição da carga energética do dispositivo. Já segunda etapa realiza os procedimentos demonstrados no Algoritmo 1. Nele, a linha 2 realiza uma verificação do valor da variável *CA* e, caso esse valor ainda não estiver definido, trata-se do primeiro anúncio, que é o anúncio de registro no *DS*. Caso o valor da *CA* não esteja definido, este recebe o valor máximo de carga energética *(CMAX)* do *AS* (linha 3). Isso ocorre porque não existe anúncio anterior. Esse valor corresponde ao total de unidades de carga energética ou valor máximo de carga energética do *AS* (linha 3). O valor de *CMAX* é relativo e depende da capacidade de carga energética do *AS*.

A variável ΔC é inicializada com o valor 1 (um), ou seja, adota-se uma variação inicial de uma unidade de carga (linha 4). Após a inicialização, a variação de carga energética do *AS* é calculada (linha 6), com os valores de *CA* e *CI*. Nos casos em que

Algoritmo 1: Ajuste do Período de Anúncio

```
   Entrada: CA, CI, ΔC, CMAX
1  início
2      se CA = null então
3          CA ← CMAX
4          ΔC ← 1
5      fim
6      se CA > CI então
7          ΔC ← CA − CI
8      fim
9      se CA < CI então
10         ΔC ← CMAX − CI
11     fim
12     CA ← CI
13 fim
```

o valor de *CA* é maior do que o valor de *CI*, ou seja, a carga energética do dispositivo diminuiu (linha 6), o valor da ΔC é igual a diferença entre os valores de *CA* e *CI* (linha 7). Nas situações em que o valor de *CI* é superior ao valor de *CA* (linha 9), deste modo, a carga energética do *AS* aumentou em relação à última medição, o valor de ΔC é igual à diferença do valor de *CMAX* e o valor de *CI* (linha 10). Assim, a variação será ajustada em situações de recarga e descarga energética. Na linha 12, o valor de *CA* é atualizado com o valor de *CI*, e passa a ser o referencial para o cálculo da variação no próximo anúncio. Essa atualização no valor de *CA* é importante para manter o valor de ΔC relativo à última medição de carga do *AS*. O valor de ΔC encontrado é importante, pois ele representa o período de tempo para o próximo anúncio, e isso define o *Prazo de Atualização (PA)*. Assim, o *AS* responsável pelo serviço têm definido o próximo momento de anúncio. Além de definir o *PA*, o ΔC define o *índice de disponibilidade* (**ID**) do serviço, que é enviado na mensagem de anúncio ao *DS*.

O **MMS** realiza a classificação de serviços com base no índice de disponibilidade recebido na mensagens de anúncio do **MAE**. A mensagem de anúncio do serviço é enviada por um canal seguro de comunicação. A principal função do **MMS** consiste em classificar os serviços registrados e responder às requisições dos dispositivos que solicitam a lista de serviços do repositório. Essa classificação promove os serviços com maior disponibilidade. Nesse módulo, dois procedimentos acontece: *i) atualização do registro de serviço anunciado*, e *ii) reordenação da lista de serviços* de acordo com o índice de disponibilidade do serviço. A primeira etapa de operação do módulo atualiza o registro do serviço no *DS*. Este procedimento ocorre quando o *AS* anuncia o serviço, e o *DS* recebe a mensagem de anúncio que contém o índice de disponibilidade do serviço. O *DS* então encontra o registro do serviço anunciado e atualiza o índice de disponibilidade. Essa etapa é fundamental à manutenção dos serviços registrados. Os registros no *DS* devem estar atualizados para reproduzir a situação do *AS* responsável por cada serviço.

Feita a atualização de registro do serviço anunciado, a lista de serviços mantida pelo *DS* deve ser reordenada. A reordenação da lista é feita em ordem crescente do índice de disponibilidade de cada serviço. Os serviços com o menor valor de índice de disponibilidade têm um período de anúncio menor, e portanto, tiveram menor gasto energético desde o anúncio anterior. Esse menor gasto energético pode ser referente a um menor número de requisições de serviços atendidos, ou referente à maior capacidade de carga energética do *AS* responsável pelo serviço. O resultado da reordenação dos serviços re-

gistrado é a lista ordenada de serviço. Essa lista colabora para a resolução do problema da falta de um indicador de disponibilidade dos serviços registrados. Em posse dessa lista, os consumidores de serviços podem requisitar os serviços no topo da lista, com maior possibilidade de obterem resposta mais confiáveis e disponíveis.

4.2. Fluxo de funcionamento do MADS e do AFT

O funcionamento dos módulos do MADS contribuem para a diminuição do número de mensagens de anúncio e para a geração do *indicador de disponibilidade* de cada serviço. A análise energética realizada no *AS* é importante para definir *períodos de anúncios* compatíveis com o gasto energético do dispositivo e para gerar o contexto de execução de onde é obtido o índice de disponibilidade, fundamental para gerar o indicador de disponibilidade de cada serviço do *DS*. A Figura 3 ilustra um esquema com o fluxo de funcionamento da *MADS* no *AS*, bem como um exemplo de operação do mecanismo *AFT (Adaptive Fibonacci-based Tuning Protocol)* de descoberta de serviços proposto por [Albalas et al. 2017]. O esquema é dividido em 3 camadas. Cada camada representa um fluxo de operação. A camada superior ilustra o fluxo de operação do *MADS*. A camada inferior representa o fluxo de operação do mecanismo *AFT*, por fim a camada central demonstra as operações comuns à ambas as abordagens.

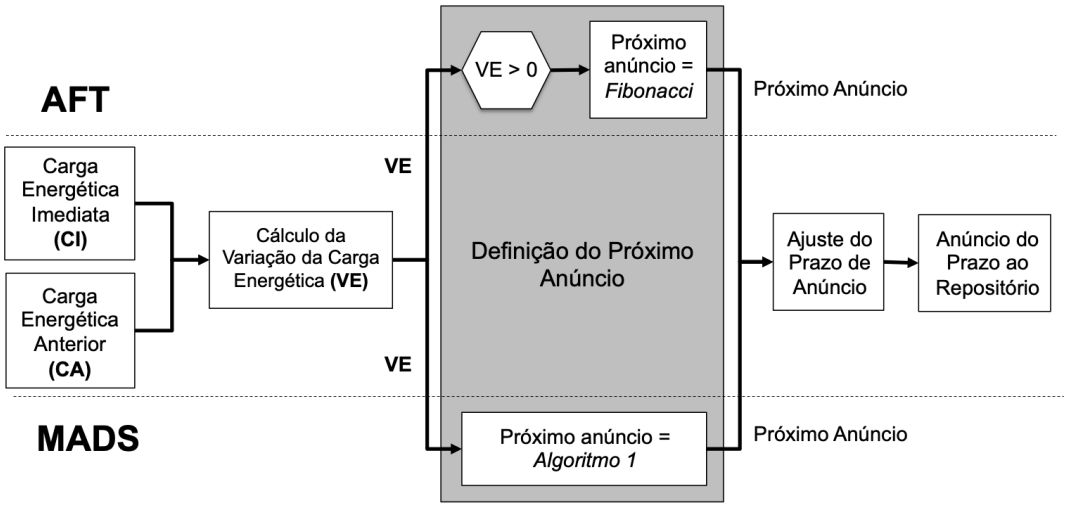


Figura 3. Fluxo de funcionamento - MADS x AFT

As duas primeiras operações determinam os valores de *CA* e *CI*. A terceira operação realiza o cálculo da valor de ΔC, correspondente a variação de carga. Essas 3 operações são comuns nos dois mecanismos. A seguir, na fase de definição do próximo anúncio, o fluxo de operações se diferenciam em cada mecanismo. Para definir o próximo anúncio, o mecanismo *AFT* verifica se houve variação na carga energética e, caso positivo, calcula um novo período de anúncio de serviço usando a *Sequência de Fibonacci*. Nessa fase, o valor da variação de carga energética é utilizado somente para verificar se houve diminuição na carga energética do dispositivo. Durante a mesma fase, no *MADS*, o período para o próximo anúncio do serviço é definido diretamente pelo valor da variação da carga. A principal diferença de operação dos dois mecanismos é a maneira como o valor da variação de carga energética é utilizado. No *MADS*, o período de anúncio de serviço é ajustado em razão do valor de variação de carga energética.

Definido o período para o próximo anúncio de serviço, o fluxo converge novamente na operação de ajuste do prazo de anúncio. Essa operação consiste no ajuste do novo prazo que o dispositivo tem para enviar uma nova mensagem de anúncio ao repositório de serviços. A última operação, também comum aos dois mecanismos, é o anúncio do prazo de atualização ao repositório de serviços. Nessa operação, o dispositivo envia uma mensagem de atualização ao repositório, contendo o novo prazo para atualização.

5. Avaliação

Esta seção descreve a metodologia de implementação e avaliação de desempenho do do MADs para validar a sua eficácia. Também é feita uma análise comparativa com o mecanismo AFT (*AFT: Adaptive Fibonacci-based Tuning Protocol*), proposto por [Albalas et al. 2017]. Para realizar a comparação, rotinas foram implementadas utilizando a linguagem de programação *Python* na versão 3.44. A implementação foi feita representando o cenário de troca de mensagens de anúncio durante períodos de observação de 5, 10, 30 e 60 minutos, conforme parâmetros descritos na Tabela 1. Esses períodos de observação foram escolhidos de forma arbitrária, e representam o intervalo entre o registro e o desligamento do serviço.

Tabela 1. Configuração da simulação

Parâmetros	Valores
Intervalos de observação	5, 10, 30 e 60 minutos
Taxa de variação de carga energética	1-5, 6-10, 11-15 e 16-20 unidades de carga
Mensagens por anúncio	1

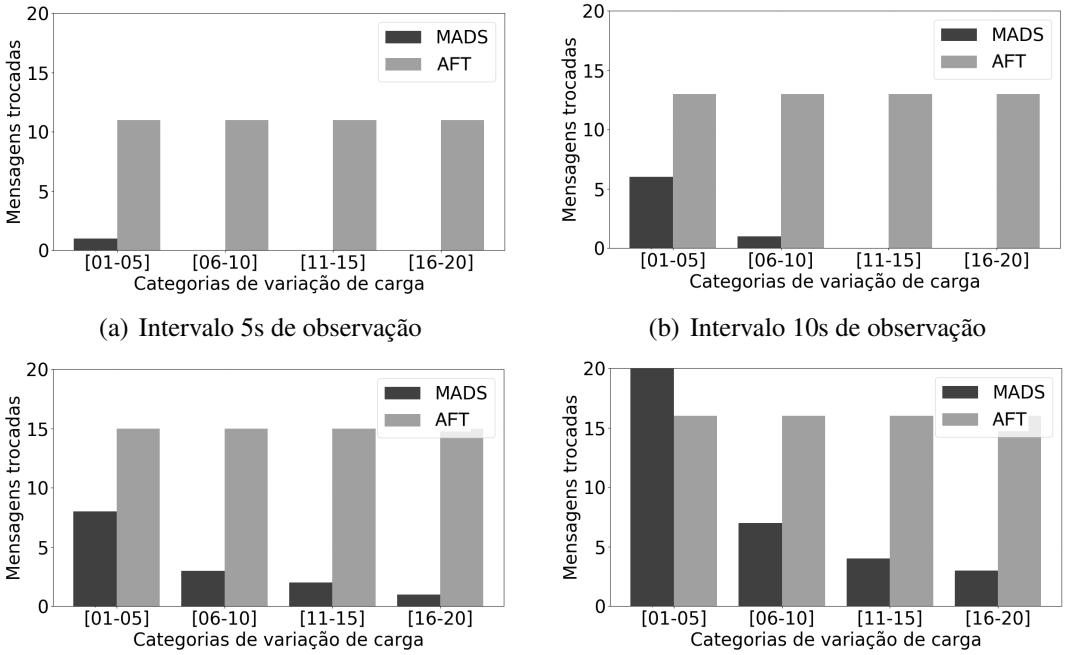
Durante a simulação, foram aplicadas taxas de variação de carga energética de 1-5, 6-10, 11-15 e 16-20 unidades por minuto. As unidades de carga energética são uma abstração do algoritmo para unidades de medida de carga energéticas reais, geralmente representadas em miliampére-hora (mAh). Foram executadas 30 rodadas de simulação, variando a carga de acordo com os valores anteriormente descritos.

Na análise do desempenho de ambos os sistemas, foram consideradas duas métricas: **número de mensagens trocadas (NMT)** e o **índice de disponibilidade dos serviços (IDS)**. O **NMT** representa as mensagens trocadas durante o processo de descoberta de serviços, o que consiste em toda a transação realizada com o solicitante de um serviço na rede e o provedor de serviço registrado no diretório de serviço associado. Já o **IDS** corresponde ao indicador de disponibilidade considerando que quanto menor o intervalo entre os anúncios de um provedor de serviço maior será sua disponibilidade devido sua demostração de atividade na rede.

5.1. Resultado: Número de mensagens de anúncio trocadas

O número de mensagens trocadas (NMT) para anunciar um serviço é importante no ecossistema de IoT. Uma maior quantidade de mensagens trocadas provoca maior consumo de energia dos dispositivos, devido ao uso frequente do rádio para realizar o envio dos pacotes. Os resultados obtidos nas experimentações são mostrados nos gráficos da Figura 4.

O gráfico 4(a) apresenta o desempenho do MADS comparado ao AFT no número de mensagens trocadas. O desempenho do MADS é superior na comparação durante



(a) Intervalo 5s de observação

(b) Intervalo 10s de observação

(c) Intervalo de 30s de observação

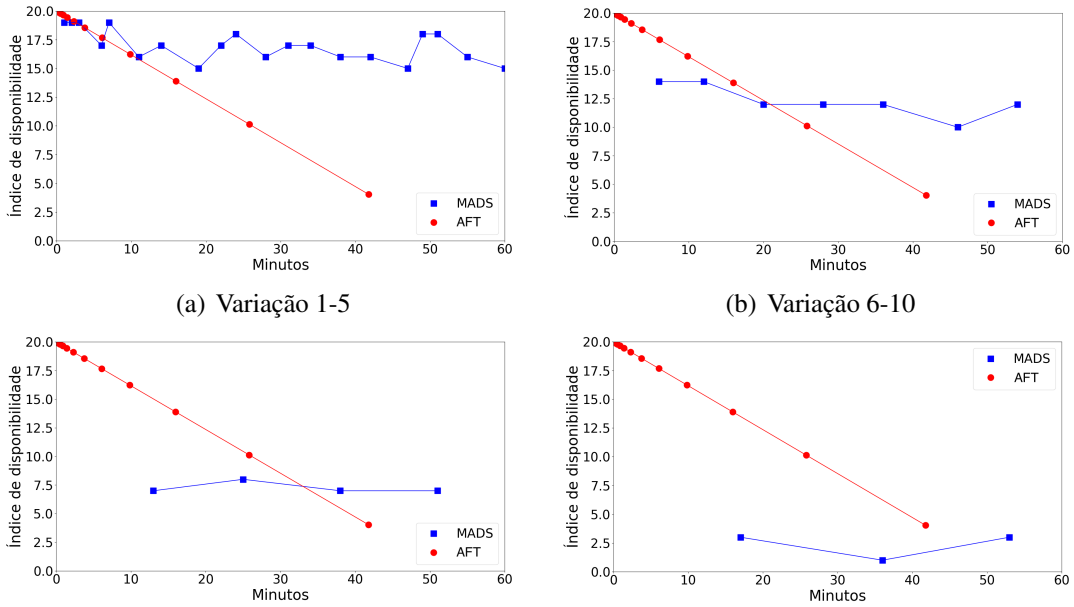
(d) Intervalo de 60s de observação

Figura 4. Avaliação de Mensagens de Anúncio Trocadas

5 segundos de anúncio do dispositivo, trocando uma mensagens somente na categoria de variação de carga energética entre 1 e 5. Nesse mesmo período, o AFT troca 11 mensagens em todas as taxa de variação. No gráfico 4(b), o desempenho do MADS continua superior em todas categorias de variação de carga energética, trocando mensagens somente nas categorias de variação entre 1 e 5, e 6 e 10. O número máximo de troca de mensagens durante 10 segundos atinge 6, enquanto o mecanismo AFT troca 13 mensagens em todas as categorias de carga energética. No gráfico 4(c), o MADS troca 8 mensagens com uma variação de carga energética entre 1 e 5, e apresenta decréscimo no número de mensagens trocadas nas outras categorias. Já o AFT troca 16 mensagens em todas as categorias de variação de carga energética. Já no gráfico 4(d), o desempenho do MADS na categoria de variação de carga energética entre 1 e 5 é inferior, trocando 20 mensagens, 2 a mais que o AFT. Porém, nas outras categorias de variação, o número de mensagens trocadas pelo MADS decresce, enquanto do AFT se mantém fixo em 18 mensagens. O desempenho do MADS em comparação ao mecanismo AFT, no geral, é superior. Nos 16 cenários de comparação, o número de mensagens trocadas pelo MADS é sempre inferior.

5.2. Resultado: Índice de disponibilidade do serviço

O índice de disponibilidade diz respeito à uma métrica importante na seleção de um serviço no ecossistema IoT. Nesse ecossistema, a mobilidade de um dispositivo e a falta de uma fonte energética ilimitada influenciam diretamente na disponibilidade de um serviço, dado que o dispositivo provedor pode se desconectar a qualquer momento. A disponibilidade de um serviço é relacionada, nesse trabalho, com a periodicidade com que um serviço realiza anúncios de atualização de serviço, ou seja, de sua atividade na rede. Na medida em que um serviço apresenta um menor intervalo entre anúncios, maior sua disponibilidade, dado que está mais ativo na rede.



(a) Variação 1-5

(b) Variação 6-10

(c) Variação 11-15

(d) Variação 16-20

Figura 5. Avaliação do Índice de Disponibilidade

Na Figura 5 são apresentados quatro gráficos comparativos entre o comportamento do índice de disponibilidade do MADS e do mecanismo AFT durante 60 minutos de observação. O gráfico 5(a) mostra o comportamento dos índices de disponibilidade baseados em variações entre 1 e 5 unidades de carga energética por minuto. O comportamento do MADS, apesar de irregular no sentido da variação do índice, apresenta um índice superior enquanto o valor do índice do mecanismo AFT diminui com o passar do tempo. No gráfico 5(b), o índice de disponibilidade do MADS cai, porém apresenta comportamento mais regular, e continua superior ao AFT na maior parte do tempo observado. No gráfico 5(c), a regularidade do índice do MADS melhora, porém o índice de disponibilidade cai novamente, e passa a ser inferior durante a maior parte do tempo, contudo apresenta um comportamento mais estável que o AFT, que se mantém em queda. Já no gráfico 5(d), o índice de disponibilidade do MADS é inferior durante todo o período de observação, sendo o valor mínimo alcançado bem próximo a zero.

No geral, na métrica da disponibilidade do serviço, o MADS é superior. Isso deve-se à abordagem adotada para definir os períodos de anúncio. O mecanismo AFT apresenta diminuição constante do índice de disponibilidade, e como pode ser observado em todos os gráficos da Figura 5, o último aviso dentro do período de observação acontece entre 40 e 50 minutos, longe do limite de 60 minutos, o que demonstra a chance crescente de que o serviço esteja indisponível no momento de uma requisição de serviço.

6. Conclusão

A descoberta de serviços é fundamental no ecossistema de IoT pois disponibiliza as funcionalidades, bem como os próprios serviços demandados pelos dispositivos conectados à rede. Para realizar a descoberta de serviço, várias proposta foram apresentadas na literatura. Na maioria dessas propostas, os dispositivos conectados a rede realizam anúncios periódicos de atualização dos serviços oferecidos ao diretório de serviços. A necessidade

de realização de atualização periódica gera um aumento no número de troca de mensagens, um alto consumo energético e consequentemente reduz a disponibilidade do serviço. Este trabalho apresentou o Mecanismo Adaptativo de Descoberta de Serviços (MADS) para diminuir o número de mensagens de atualização durante o processo de descoberta de serviço e para fornecer um indicador de disponibilidade de serviço.

As análises realizadas durante a execução das rotinas do mecanismos proposto avaliaram as métricas: *i) número de mensagens trocadas* e; *ii) valor do índice de disponibilidade*. Os resultados mostram que o MADS diminui o número de mensagens trocadas no processo de atualização e fornece um indicador de disponibilidade dos serviços. Além disso, o mecanismo contribui para a redução do consumo de carga energética dos dispositivos e auxilia na seleção do serviço mais disponível, colaborando para a melhora do processo de descoberta de serviço na IoT. Trabalhos futuros incluem o desenvolvimento de mecanismo para garantia de segurança durante a troca de mensagens de anúncio de serviços entre o provedor e o diretório de serviços, e o desenvolvimento de um procedimento de remoção de registros de serviços indisponíveis no diretório de serviços.

Referências

Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., and Ayyash, M. (2015). Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys & Tutorials*, 17:2347–2376.

Albalas, F., Mardini, W., and Al-Soud, M. (2017). Aft - adaptive fibonacci-based tuning protocol for service and resource discovery in the internet of things. *IEEE Second International Conference on Fog and Mobile Edge Computing (FMEC)*.

Albalas, F., Mardini, W., and Yaseen, M. A.-S. Q. (2019). A topology–based performance evaluation for an adaptive tuning protocol for service and resource discovery in the internet of things. *Computing and Communication Workshop and Conference (CCWC) 2019 IEEE 9th Annual*.

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

Aziez, M., Benharzallah, S., and Bennoui, H. (2017). Service discovery for the internet of things: Comparison study of the approaches. *IEEE 4th International Conference on Control, Decision and Information Technologies (CoDIT)*.

Barreto, F. M., de S. Duarte, P. A., Maia, M. E. F., de C. Andrade, R. M., and Viana, W. (2017). Coap-ctx: A context-aware coap extension for smart objects discovery in internet of things. *IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)*.

Broering, A., Datta, S. K., and Bonnet, C. (2016). A categorization of discovery technologies for the internet of things. *ACM IoT'16 Proceedings of the 6th International Conference on the Internet of Things*, 6:131–139.

Cirani, S., Davoli, L., Ferrari, G., Léone, R., Medagliani, P., Picone, M., and Veltri, L. (2014). A scalable and self-configuring architecture for service discovery in the internet of things. *IEEE Internet of Things Journal*, 1:508 – 521.

De, S., Barnaghi, P., Bauer, M., and Meissner, S. (2011). Service modelling for the internet of things. *IEEE Federated Conference on Computer Science and Information Systems*, page 949–955.

Gigli, M. and Koo, S. G. M. (2011). Internet of things: Services and applications categorization. *Advances in Internet of Things*, 1:27–31.

Gubbi, J., Buyya, R., Marusic, S., and Palaniswami, M. (2013). Internet of things (iot): A vision, architectural elements, and future directions. *Elsevier Future Generation Computer Systems*, 29:1645–1660.

Helal, R. and ElMougy, A. (2015). An energy-efficient service discovery protocol for the iot based on a multi-tier wsn architecture. *IEEE 40th Local Computer Networks Conference Workshops (LCN Workshops)*, 40:862–869.

Ishaq, I., Hoebeke, J., Rossey, J., Poorter, E. D., Moerman, I., and Demeester, P. (2012). Facilitating sensor deployment, discovery and resource access using embedded web services. *IEEE Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS)*.

Kaiser, D., Waldvogel, M., Strittmatter, H., and Haase, O. (2016). User-friendly, versatile, and efficient multi-link dns service discovery. *IEEE International Conference on Distributed Computing Systems Workshops*, 36:146 – 155.

Khalil, K., Elgazzar, K., and Bayoumi, M. (2018). A comparative analysis on resource discovery protocols for the internet of things. *Global Communications Conference (GLOBECOM) 2018 IEEE*.

Kim, S., Lee, K., and Jeong, J. P. (2017). Dns naming services for service discovery and remote control for internet-of-things devices. *IEEE International Conference on Information and Communication Technology Convergence (ICTC)*.

Meshkova, E., Riihijärvi, J., Petrova, M., and Mähönen, P. (2008). A survey on resource discovery mechanisms, peer-to-peer and service discovery frameworks. *IEEE Computer Networks*.

Mohiuddin, J., Bhadram, V., Palli, S., and Koshy, S. S. (2014). 6lowpan based service discovery and restful web accessibility for internet of things. *IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI)*.