

Atribuições Cooperativas de Tarefas de Sensoriamento Baseada em Consenso Relacional para Redes IIoT

Carlos Pedroso¹, Yan Uehara¹, Michele Nogueira¹, Aldri Santos¹

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

{capjunior, yumoraes, michele, aldri}@inf.ufpr.br

Resumo. *Entre os atuais serviços IoT destacam-se os voltados a indústrias, que muitas vezes exigem que os objetos desempenhem mais de uma tarefa. Entretanto, o crescimento exponencial dos objetos na IoT impõe o desafio de distribuir e gerenciar a alocação de tarefas entre os objetos. Um dos principais objetivos da alocação de tarefas é melhorar a qualidade das informações e a maximização de tarefas a serem executadas. Embora existam abordagens que otimizem e gerenciem a dinamicidade dos nós, nem todas consideram a relação de similaridade a alocação distribuída sobre o serviço de agrupamento. Este trabalho propõe um mecanismo chamado CONTASKI para alocação de tarefas em redes IIoT, a fim de distribuir as tarefas entre os objetos. Ele baseia-se em estratégias de consenso relacional para alocar as tarefas e de similaridade de capacidades para determinar quais objetos podem atuar na realização dessas tarefas. O CONTASKI foi avaliado no NS-3 gerenciando um ambiente IIoT e obteve uma eficiência de cerca de 98% de tarefas alocadas na maioria dos cenários avaliados e mais de 80% dos agrupamentos IIoT aptos a executá-las.*

1. Introdução

A Internet das Coisas (IoT) é uma rede heterogênea, onde os objetos que a compõem possuem várias características como identidade, atributos físicos, capacidade computacional e de sensoriamento. Eles usam interfaces inteligentes para estabelecer comunicação entre si, além de poderem apresentar alguma forma de mobilidade [Gubbi et al. 2013, Botta et al. 2016]. Para a grande maioria dos objetos da IoT, torna-se importante reduzir o consumo de energia com a comunicação ou execução de determinadas tarefas. Os objetos devem compartilhar seus recursos e cooperar de forma equilibrada para melhorar o desempenho da rede [Khalil et al. 2019]. Dessa forma, objetos que executem a mesma função podem colaborar entre si para alocar diferentes tarefas [Ghanbari et al. 2019]. A IoT está diretamente ligada à evolução de vários domínios de aplicação complexos como, processos industriais, manufatura, logística, saúde inteligente e cidades inteligentes. Nesse contexto, o paradigma da Internet das coisas Industriais (IIoT) vem recebendo maior atenção nos últimos anos [Xu et al. 2018]. A IIoT trata da conexão de objetos com inúmeras capacidades dentro de uma indústria, possibilitando que todos trabalhem de forma sincronizada e organizada na execução de tarefas.

A alocação de tarefas entre os objetos da IoT torna-se uma alternativa para lidar com a heterogeneidade dos objetos que podem executar mais de uma tarefa de sensoriamento [Qin et al. 2020]. Isso acontece pelo fato de que as tarefas podem ser alocadas avaliando as capacidades que cada objeto é capaz de executar dentro da rede [Korsah et al. 2013]. Entretanto, a IoT é dinâmica e normalmente conta com objetos móveis e fixos, além disso, a infraestrutura varia conforme as interações entre

os objetos. Parte dos objetos têm recursos limitados, pouca energia, baixa capacidade de processamento e armazenamento, além de sofrerem perdas nos links de conexão [Borgia 2014, Qiu et al. 2018]. Assim, lidar com essas variações impõem desafios quanto à qualidade dos dados gerados e disseminados pelos objetos. A má qualidade na disponibilização das informações dificulta sua interpretação pelas aplicações [Evangelista et al. 2016]. A utilização de agrupamentos de objetos contribui para organização da rede, diminuição do consumo de recursos e a fluidez com que as informações são disseminadas entre os objetos [Gielow et al. 2015]. Devido ao tamanho e as configurações que uma rede IoT pode alcançar, o problema de alocação de tarefas não é trivial e demanda atenção de pesquisadores [Colistra et al. 2014, Khalil et al. 2018].

Particularmente, lidar com o problema de alocação de recursos para a execução de tarefas entre diferentes objetos da IoT têm se mostrado um grande desafio [Aazam et al. 2018]. Fatores como a escassez de recursos, heterogeneidade de objetos e dinamicidade do ambiente têm dificultado a gerência de distribuição de tarefas. A alocação de recursos torna-se importante na justiça da distribuição de tarefas entre os participantes da rede IoT, sem ela diversos objetos podem consumir recursos de forma desnecessária ou mesmo vir a falhar em momentos cruciais para definições de papéis [Ghanbari et al. 2019]. Uma má distribuição de tarefas acarreta problemas na distribuição de recursos e na disponibilização das informações que são geradas pelos objetos. Além disso, a eficiência da alocação de tarefas deve ser preservada para que os objetos garantam a qualidade dos serviços alocados [Fang et al. 2018]. Nas indústrias inteligentes com alta demanda de múltiplos serviços, as capacidades de sensoriamento dos objetos IIoT são fundamentais para garantir a distribuição de tarefas de modo otimizado, visto que diariamente os objetos realizam múltiplos serviços [Zhong et al. 2017]. Assim, uma gerência inteligente otimiza os recursos dos objetos IIoT nas diversas tarefas, ao permitir os objetos intercalarem a execução de acordo com as necessidades da aplicação.

A alocação de tarefas tem sido estudada extensivamente em redes de sensores sem fio (RSSF), que normalmente a trata como alocação de recursos. Particularmente, a alocação de tarefas emprega-se visando melhorar o tempo de vida da rede [Pilloni and Atzori 2017]. Com o rápido avanço das tecnologias para redes IoT, as soluções para alocação de tarefas passaram também a fazer parte deste escopo de pesquisa [Ghanbari et al. 2019]. Dentre os trabalhos que buscam lidar com o problema de alocação de tarefas, muitos o consideram através do uso de virtualização de objetos em grupos de tarefas [Khalil et al. 2019] e consenso distribuído [Pilloni and Atzori 2017]. A virtualização de objetos utiliza-se na atribuição de tarefas de acordo com as competências que cada objeto dispõe e é capaz de executar. Seu principal objetivo é a otimização de tarefas com o intuito de economizar recursos. O consenso distribuído aplica-se na distribuição igualitária de recursos com base nas características de cada grupo de objetos presentes na rede, essa abordagem emprega-se em redes com um grande número de participantes [Colistra et al. 2014]. Entretanto, essas soluções não consideram a relação de similaridade entre os objetos e as tarefas a serem executadas, bem como a definição de alocação com base nas características do ambiente onde os objetos estão inseridos. Logo, a IoT demanda soluções de alocação capazes de disseminar tarefas entre os objetos IoT ao considerar suas relações e capacidades, e seus consensos colaborativos, para uma gerência mais equilibrada dos recursos disponíveis da rede.

Este trabalho apresenta um mecanismo para apoiar o serviço de alocação de tare-

fas entre os objetos de um rede IIoT, chamado CONTASKI (*CONsensus Collaborative Based TASK Allocation for IIoT*). Ele organiza a rede em agrupamentos baseados na similaridade de capacidade de sensoriamento dos objetos. Para lidar com a distribuição e execução de tarefas, o CONTASKI emprega o consenso colaborativo relacional que possibilita melhor adaptação ao contexto, rapidez nas respostas e tomadas de decisões mais assertivas. A distribuição das tarefas considera as capacidades dos objetos em desempenhá-las para otimizar a sua execução. A avaliação do CONTASKI no simulador NS-3 gerenciando um ambiente IIoT mostrou que ele alocou cerca de 98% de tarefas na maioria dos cenários avaliados e alcançou mais de 80% dos agrupamentos formados aptos para execução. Isso deve-se ao consenso relacional que conseguiu correlacionar as tarefas e as capacidades dos agrupamentos.

O restante do artigo está organizado da seguinte forma: A Seção 2 apresenta os trabalhos relacionados. A Seção 3 apresenta e descreve o mecanismo CONTASKI e seu funcionamento. A Seção 4 descreve o método de avaliação e os resultados obtidos na atuação do CONTASKI. A Seção 5 conclui o artigo e apresenta direcionamentos futuros.

2. Trabalhos Relacionados

A exigência de serviços dinâmicos e distribuídos conforme os recursos e capacidades dos objetos IoT tem sido o foco de diversos trabalhos [Khalil et al. 2019, Pilloni and Atzori 2017, Colistra et al. 2014, Pilloni and Atzori 2011]. Em geral, o uso de gerência de alocação de tarefas viabiliza a maximização de aproveitamento de recursos entre os objetos atuantes na rede. Entretanto, gerenciar a distribuição desses recursos acarreta desafios em sua condução como: (i) avaliar as capacidades dos objetos em relação às tarefas a serem executadas, (ii) organização da rede baseada em líderes e (iii) otimização de recursos e tomada de decisão.

Os autores [Pilloni and Atzori 2017] propuseram um novo algoritmo distribuído para objetos virtuais (OV) em uma rede IoT *Smart health*. Os OVs são capazes de implementar uma estratégia descentralizada para a alocação de tarefas, que negociam entre si, para chegar a um consenso de alocação de recursos. Eles estenderam o modelo de informações que são recebidas pelos objetos para incluir novos recursos em um cenário distribuído, incluindo a qualidade de informação que mede a caracterização das informações. Embora o trabalho tenha seguido uma linha promissora, ele não avalia as questões dos objetos terem diferentes capacidades e tipos de interações e também qual a influência do tamanho da rede. Em [Khalil et al. 2019], os autores desenvolveram um algoritmo evolutivo baseado em uma heurística de reconhecimento de heterogeneidade para garantir maior extensão da estabilidade e dos períodos operacionais das tarefas em uma rede IoT. O modelo cria uma colaboração entre as funções dos objetos IoT com base nas tarefas a serem executadas e nos grupos selecionados. O algoritmo busca selecionar objetos com níveis de energia acima do nível médio dos outros membros do grupo de tarefas. Dessa forma, os objetos virtuais do grupo são capazes de executar as tarefas e também reduzir o seu consumo de energia. Entretanto, o modelo considera que poucos objetos são capazes de executar todas as tarefas definidas pela rede, além disso, contam com apenas dois tipos de tarefas a serem executadas, o que limita o modelo a redes com diversos nós e capacidades.

Em [Colistra et al. 2014], os autores desenvolveram uma abordagem heurística baseada em consenso para alocação de tarefas. O modelo é um método distribuído de alocação de tarefas na IoT com o objetivo principal de tolerância a falhas. O algoritmo

usa o conceito de grupos de tarefas e objetos. Em cada grupo de tarefa, os objetos podem ser selecionados como virtual e vice-virtual. Esse modelo auxilia a melhor divisão de tarefas entre os objetos. Então, quando um objeto virtual esgota suas energias, o objeto vice-virtual assume o dever de ser um objeto virtual e o próximo na lista se torna o objeto vice-virtual do grupo de tarefas correspondente. Entretanto, o modelo exige trocas periódicas de mensagens de *hello*, que representam as mensagens de gerenciamento. Essa troca excessiva de mensagens representa uma sobrecarga tanto de comunicação quanto de computação, armazenamento e energia. Além disso, eles não consideram as capacidades dos nós da rede, o que influencia diretamente na distribuição de tarefas. Em [Kim and Ko 2015], os autores propuseram lidar com o problema de alocação de tarefas na IoT assumindo que todos os objetos não possam interagir diretamente um com o outro. Eles utilizam um serviços de *gateways* para ser responsável por gerenciar essa interação. Assim, transformaram o problema de alocação de tarefas em um problema de integração com uma variante de grau mínimo com o objetivo de restringir o problema e, dessa forma, conseguem aplicar um algoritmo genético para reduzir o tempo necessário para alocar as tarefas. Porém, a utilização do gerenciador de interação acaba por limitar as relações entre os nós, além disso, esse modelo de centralização de informações pode gerar um gargalo de comunicação dependendo do tamanho da rede.

Em [Pilloni and Atzori 2011], os autores propuseram um algoritmo que decompõe as tarefas dos sensores em tarefas distribuídas. O algoritmo avalia o consumo de energia para execução de cada tarefa e, dessa forma, avalia a viabilidade da solução na atribuição das tarefas entre os sensores. No entanto, o modelo considera apenas uma entidade centralizadora na distribuição das funções, não considerando uma organização da rede baseada em líder para distribuição de tarefas entre os nós. Embora efetivo em uma rede dinâmica como a IoT, pode não ser viável, tendo em vista os cenários autônomos. Seguindo a mesma linha [Jin et al. 2011], apresentaram um algoritmo que utiliza um mapeamento adaptável de tarefas em sensores. Ele trabalha de maneira paralela com um agendamento baseado em um algoritmo genético em que ambos trabalham em tempo real. Os algoritmos visam estender a vida útil da rede através de um balanceamento de carga de trabalho entre os sensores. Porém, ao centralizar toda a distribuição de tarefas, eles sobrecarregam o canal de transmissão, além de inserir um atraso na entrega das mensagens, que acabam por comprometer a sincronização da execução das tarefas. Em [Chen et al. 2016] utilizou-se a similaridade do cosseno para calcular a similaridade entre dois objetos quanto ao seu contexto social, mas neste trabalho será utilizada para determinar a similaridade de capacidade de sensoreamento entre os nós.

Assim, há a demanda de uma gerência de alocação de tarefas, capaz de atuar de forma colaborativa e distribuída entre os objetos, atrelando as tarefas de acordo com as capacidades de cada objeto na rede IIoT. Essas soluções, ao preservar os recursos dos objetos, não devem gerar sobrecarga adicional, além de atender normalmente as características da rede IoT e do ambiente onde ela está inserida.

3. Um Mecanismo para Alocação de Tarefas Colaborativa

Esta seção apresenta o mecanismo CONTASKI (*CONsensus Collaborative Based TASK Allocation for IIoT*) para gerenciamento de alocação de tarefas entre objetos (nós) de uma rede IIoT. Ele organiza a rede em agrupamentos baseados na similaridade de capacidade de sensoriamento dos objetos. Para lidar com a distribuição e execução de tarefas, o CONTASKI emprega o consenso colaborativo relacional. A distribuição das ta-

refas considera as capacidades dos nós em desempenhá-las tal que sua execução ocorra de forma otimizada. Assim, obtém-se uma otimização de recursos, organização no tráfego de informações na rede, além de melhorar a rapidez com que as tarefas são disponibilizadas.

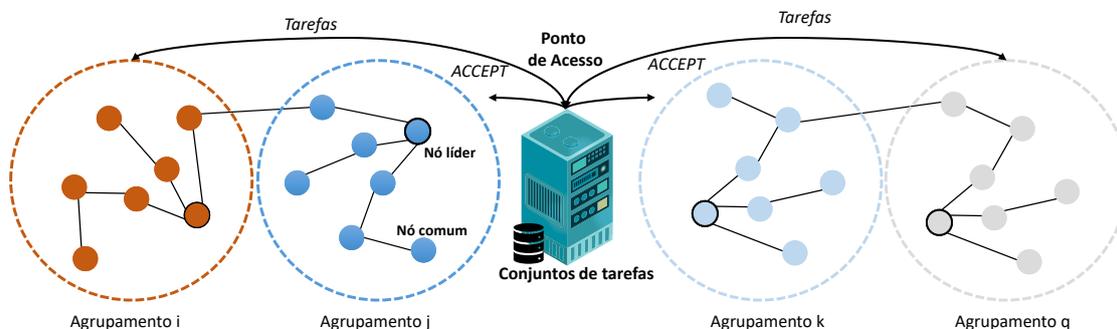


Figura 1. Modelo de rede IIoT

Modelo de Rede IIoT: O modelo de rede pode ser observado na Figura 1 consiste de uma rede IIoT N , representada em uma área (X_x, Y_y) . A rede N é composta por conjunto de objetos OB denotados por $N = \{ob_1, ob_2, ob_3, ob_n\}$, onde $ob_i \in N$. Cada objeto ob é equipado por um identificador único Id que o identifica na rede. Os objetos diferenciam-se pelos seus conjuntos de capacidades $C = \{c_1, c_2, c_3, \dots, c_n\}$, processamento, energia e memória. Os objetos são estáticos e distribuídos aleatoriamente em uma área da N com as coordenadas $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$. Por tratar-se de uma rede IIoT, considera-se que os objetos normalmente não sofrem de restrição energética.

Modelo de Tarefa: Cada tarefa representa uma requisição de sensoriamento que exige capacidades atribuídas aos objetos IIoT. Uma tarefa T é um conjunto $\{T_{id}, c, \tau, q\}$, onde c é o conjunto de capacidades necessárias para desempenhar a tarefa, τ , o tempo necessário para completá-la e q , a quantidade de nós necessários. A tarefa T é distribuída através do Ponto de Acesso (AP), que mantém a lista de tarefas a serem executadas e seus estados, que podem ser pendentes e despachadas.

Modelo de Comunicação e Mensagens: A comunicação ocorre através do meio sem fio através de um canal assíncrono. Considera-se que as conexões são confiáveis e, portanto, os objetos não apresentam falhas de comunicação. O modelo compreende cinco tipos de mensagens trocadas entre os participantes da rede: mensagens de disseminação de capacidade dos objetos (*CapabilityDissemination*), enviadas aos dispositivos da rede para a formação dos agrupamentos; Os líderes enviam mensagens de registro (*LeaderRegister*), para o AP para anunciar como líder; mensagens de despacho de tarefas (*TaskDispatch*) são enviadas pelo AP aos líderes; mensagens de aceite de tarefa (*TaskAccept*) são enviadas ao AP quando os líderes aceitam uma tarefa; mensagens de execução de tarefa (*LeaderToCluster*) são enviadas aos membros do agrupamento quando uma tarefa é aceita.

3.1. Arquitetura CONSTASKI

A arquitetura do CONTASKI é composta por dois módulos, **Coordenador de Agrupamentos** e **Controle de Alocação de Tarefas**, como ilustrado na Figura 2. Ambos operam de maneira conjunta para garantir a distribuição de tarefas entre os nós da rede IIoT. O módulo coordenador de agrupamentos organiza a rede em agrupamentos e o módulo controle de alocação de tarefa trata da gerencia e distribuição das tarefas entre os nós participantes da rede conforme suas capacidades.

O Módulo **Coordenador de Agrupamentos (COA)** controla a formação e manutenção dos agrupamentos dentro da rede. Ele utiliza um limiar de similaridade de capacidade dos nós que estão próximos para avaliar se estão aptos a participarem dos mesmo agrupamento. Assim, ao receber uma mensagem do tipo *CapabilityDissemination* ele verifica a identificação, as capacidades e o número de vizinhos. O COA possui o componente *Disseminação de Capacidades (DC)*, encarregado de disseminar as mensagens *CapabilityDissemination* com seu identificador, capacidades e número de vizinhos. A *Verificação de Similaridade (VS)* é responsável por receber e interpretar as mensagens trocadas entre os nós da rede e calcular a similaridade entre os nós. A *Gerência de Agrupamentos (GA)* trata de formação dos agrupamentos a partir da relação de similaridade entre os nós, ele também é responsável pela eleição dos líderes. Logo, todos os nós são capazes de saber qual agrupamento eles integram.

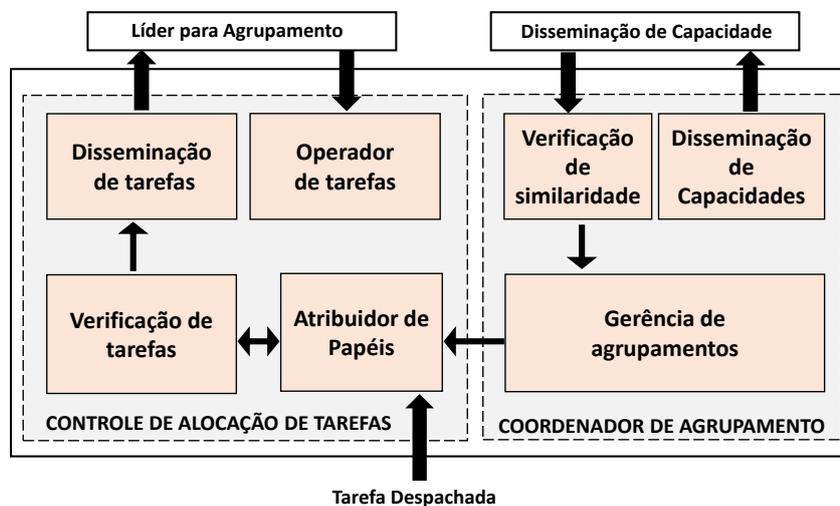


Figura 2. Arquitetura CONTASKI

O Módulo **Controle de Alocação de Tarefas (CAT)** gerencia e distribui as tarefas entre os nós da rede IoT de modo a maximizar e preservar os recursos dos nós. Ele é formado pelos componentes *Verificação de Tarefas (VT)*, *Atribuidor de Papéis (ADP)*, *Distribuidor de Tarefas (DT)* e *Operador de Tarefas (OP)*. O componente VT supervisiona quais são as tarefas a serem executadas e quais são as características necessárias para realizá-la. O componente ADP monitora as tarefas que serão atribuídas aos nós, ele emprega a técnica de consenso colaborativo para avaliar quais tarefas devem ser alocadas de acordo com as capacidades de cada nó. O consenso relacional envolve a concordância e uniformidade de opiniões que os nós estabelecem por meio de troca de informações entre eles. Essas informações estão relacionadas as capacidades disponíveis em cada nó e são associadas com as capacidades necessárias para executar uma tarefa. O componente DT atua na distribuição das tarefas requisitadas respeitando as capacidades dos nós presentes na rede. O componente OP atua no recebimento e execução das tarefas disseminadas pelo líder do agrupamento. Desta forma, a alocação de tarefa torna-se mais justa e equilibrada, além de não sobrecarregar os nós participantes da rede IIoT.

3.1.1. Formação dos agrupamentos

A rede IIoT consiste de uma grande diversidade de nós com diferentes capacidades e implicando um desafio gerenciar e controlar todas as interações. O módulo COA ge-

rencia a organização da rede em agrupamentos, tendo por base os líderes, para criar uma infraestrutura de disseminação de tarefas. Os nós iniciam a formação dos agrupamentos trocando mensagens *CapabilityDissemination* que transportam o identificador (*Id*), capacidades (*MyCapabilities*) e número de vizinhos (*NeighborhoodSize*) do emissor.

Algoritmo 1: Estabelecimento dos Agrupamentos

```

1  procedure WARMUP
2      while 150 seconds has not elapsed do
3          BROADCASTCAPABILITIES()
4          RECEIVECAPABILITIES()
5  end procedure

6  procedure BROADCASTCAPABILITIES
7      Broadcast(MyId, MyCapabilities, NeighborhoodSize)
8      WaitInterval()
9  end procedure

10 procedure RECEIVECAPABILITIES
11     Id ← GetId()
12     NeighList ← NeighList ∪ Id
13     NeighCapabilities[Id] ← GetCapabilities()
14     NeighSize[Id] ← GetNeighborhoodSize()
15 end procedure

16 procedure SIMILARITYCALCULATION
17     foreach neighbor in NeighList do
18         NeighborCapabilities ← NeighCapabilities[neighbor]
19          $sim = \frac{|MyCapabilities \cap NeighborCapabilities|}{\sqrt{|MyCapabilities| * |NeighborCapabilities|}}$ 
20         if sim ≥ Threshold then
21             cluster ← cluster ∪ neighbor
22     end procedure

23 procedure SELECTLEADER
24     LeaderCandidates ← GetNeighborsGreatestNeighborhoodSize(cluster)
25     Leader ← GetNeighborLargestCapabilitiesSet(LeaderCandidates)
26     if Leader == MyId then
27         SendMessageToAP(MyId)
28 end procedure

```

O Algoritmo 1 detalha o funcionamento do coordenador de agrupamento. Durante um tempo inicial (*warm-up*) (l.1-l.5), cada nó anuncia em *broadcast* seu identificador (*MyId*), suas capacidades (*MyCapabilities*) e a quantidade de vizinhos (*NeighborhoodSize*) e aguarda um intervalo de tempo até o próximo anúncio (l.6-l.9). Ao receber um anúncio, o nó obtém a origem (*GetId()*), as capacidades (*GetCapabilities()*) e a quantidade de vizinhos (*GetNeighborhoodSize()*) do nó anunciador e atualiza as informações recebidas inserindo-as em uma estruturas de vizinhança (*NeighList*) e capacidades (*NeighCapabilities*) (l.10-l.15). A verificação de similaridade das capacidades considera as estruturas de vizinhança e capacidades do nós e ocorre logo após o *warm-up*. A relação é calculada utilizando a similaridade do cosseno, que considera as capacidades individuais de cada nó (*MyCapabilities*) em relação a sua vizinhança (*NeighborCapabilities*). A partir de um limiar de similaridade, o nó que respeita esse valor integra o vizinho (*neighbor*) em seu agrupamento (l.16-l.22). Esse processo ocorre dinamicamente em cada nó da rede, garantindo que todos possam manter sua estrutura de agrupamento atualizada. O processo de seleção de líder ocorre após a verificação de similaridade e considera o número de vizinhos e as capacidades individuais dos nós

para escolher o líder do agrupamento. Após a seleção, o líder envia uma mensagem de registro ao AP, informando que ele é o líder (l.23-l.28). A definição do líder garante toda a comunicação entre os nós e o AP, e uma melhor organização hierárquica dentro da rede, permitindo assim uma melhor fluidez da disseminação dos dados sensorizados coletados.

A Equação 1 verifica o quão similares são as capacidades entre os nós e baseia-se no trabalho de [Chen et al. 2016]. Em sua composição são atribuídos a similaridade (sim), as capacidades do nó (C_{ob1}) e as capacidades do vizinho (C_{ob2}). Na parte superior da equação verifica-se o módulo da interseção (\cap) de capacidades entre os nós. Na parte inferior realiza-se a multiplicação dos módulos das capacidades e retirada a raiz quadrada desse valor. Ao final, é aferido se a relação de similaridade é respeitada entre os nós, caso sim, eles passam a fazer parte do mesmo agrupamento.

$$sim(ob_1, ob_2) = \frac{|C_{ob1} \cap C_{ob2}|}{\sqrt{|C_{ob1}| * |C_{ob2}|}} \quad (1)$$

A relação de similaridade é calculada com base nas capacidades entre dois nós. Ela considera um valor, que varia entre 0,65 a 1,00. Quanto mais próximo a 1, mais forte o nível de similaridade entre dois nós. A Figura 3 ilustra as relações de atribuições de similaridade em níveis relacionais entre $S_1 =$ Fraco, $S_2 =$ Médio e $S_3 =$ Forte. Essa relação pode variar de acordo com o tamanho do conjunto de capacidades pré-estabelecidas antes da rede IIoT ser implantada.

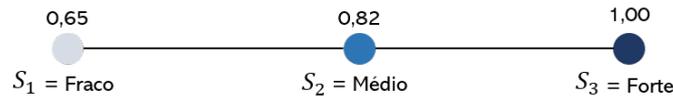


Figura 3. Relação de similaridade

3.1.2. Coordenação da Alocação de tarefas

A disponibilização das tarefas ocorre através do ponto de acesso, que mantém uma lista de tarefas a serem executadas e vai despachando-as de acordo com a sua demanda. O envio das tarefas ocorre por meio de mensagens *TaskDispatch* direcionadas aos líderes dos agrupamentos. Para identificar quem são os líderes o AP monitora as mensagens *LeaderRegister* e mantém sempre atualizada sua lista de líderes. Logo, o CAT atua considerando que a infraestrutura de redes está estabelecida pela formação dos agrupamentos. Assim, ele trabalha garantindo uma maximização de recursos, ou seja, permitindo a distribuição de tarefas de acordo com as capacidades dos nós em cada agrupamento.

O Algoritmo 2 descreve o funcionamento do controle de alocação de tarefas entre os nós da rede e como as entidades relacionam-se para a execução das tarefas encaminhadas pelo AP. Inicialmente, o AP, distribuidor de tarefas, tem um lista de tarefas de sensoriamento pendentes *TaskList* a serem realizadas pelos nós da rede. Para enviar uma tarefa, ele seleciona uma tarefa pendente e a envia através de uma mensagem *TaskDispatch* direcionada aos líderes dos agrupamentos informando a tarefa T a ser executada. Após o envio, o AP aguarda *WaitConfirmation()* até receber a confirmação de quais líderes são capazes de executar a função. Assim que é recebida a confirmação, ele retira a tarefa da lista de tarefas pendentes *TaskList* (l.1-l.6). Ao receber uma tarefa $T = (T_{id}, c, \tau, q)$,

os líderes verificam se suas capacidades *MyCapabilities* são compatíveis com as capacidades *c* necessárias para executá-la e se o número de nós no agrupamento é maior ou igual do que o quorum *q* necessário para realizar o serviço. Apenas os líderes dos agrupamentos que aceitem a execução, enviam uma mensagem *TaskAccept* ao AP e disseminam as tarefas aos seus membros através de mensagens *LeaderToCluster* (l.7-l.11). O consenso relacional entre os líderes e os seus membros viabiliza que a alocação de tarefas ocorra apenas naqueles agrupamentos com capacidades suficientes à execução.

Algoritmo 2: Distribuição da Alocação de Tarefas

```

1 procedure SENDTASK
2   for  $T \in TaskList$  do
3     SendTaskDispatch( $T$ )
4     if WaitConfirmation() then
5        $TaskList \leftarrow TaskList - T$ 
6 end procedure
7 procedure RECEIVETASK( $T_{id}, c, \tau, q$ )
8   if  $c \subseteq MyCapabilities$  and  $|cluster| \geq q$  then
9     SendTaskAccept(AP)
10    SendLeaderToCluster( $T$ )
11 end procedure

```

À medida que a gerência de alocação de tarefas atua de forma dinâmica e colaborativa, juntamente com a formação dos agrupamentos, chega-se a uma melhor distribuição de tarefas entre os participantes da rede. Esse modelo facilita o fluxo de informações trocadas entre os nós e ao mesmo tempo garante que as tarefas sejam executadas com rapidez pelos participantes da rede assegurando a otimização de recursos. Entretanto, ressalta-se que o funcionamento distribuído dentro da rede depende da colaboração consensual entre todos os participantes.

3.2. Funcionamento

Considere uma rede IoT industrial de manufatura onde os objetos estão embarcados nos maquinários e a formação dos agrupamentos atuando de maneira dinâmica em cada objeto. As interações entre objetos IIoT realizam-se sob grandezas de espaço e tempo e as mensagens de são trocadas pelos objetos que estejam dentro do raio de transmissão do objeto emissor. A Figura 4 ilustra como o CONTASKI atua na formação dos agrupamentos dos objetos IIoT, eleição de líderes e alocação de tarefas. Os sinais (*wi-fi*) indicam os objetos que estão dentro do raio de transmissão um do outro e estão aptos a trocarem mensagens de controle. Cada objeto conta com seu identificador, conjunto de capacidades e número de vizinhos. As capacidades c_1 , c_2 , c_3 e c_4 correspondem ao sensoriamento de temperatura, umidade, presença e iluminação. Além disso, emprega-se um limiar de similaridade, conforme a Figura 3, variando entre 0,65 (fraco) e 1 (forte), de acordo com as capacidades dos objetos da rede IIoT para formação dos agrupamentos.

Cada instante de tempo I_t corresponde a atuação do CONSTASKI. No instante I_{t1} , o conjunto de objetos formados por (ob_1, ob_2, ob_3, ob_4) trocam informações de capacidade e vizinhança e, em seguida, cada objeto realiza o cálculo de similaridade conforme a Equação 1. Os quatro objetos calculam e obtém os seguintes valores de similaridades entre eles: $sim(ob_1, ob_2) = sim(ob_1, ob_4) = sim(ob_2, ob_4) = \frac{3}{\sqrt{3*3}} = 1$ e $sim(ob_1, ob_3) = sim(ob_2, ob_3) = sim(ob_3, ob_4) = \frac{3}{\sqrt{3*4}} = 0,87$. A menor similaridade

medida no exemplo foi 0,87 e a máxima similaridade 1. Isso mostra que a menor similaridade obtida é uma $S_2 = \text{m\u00e9dio}$. Assim, os objetos dentro desse intervalo s\u00e3o agrupados e o agrupamento resultante representa os objetos com as capacidades C_1, C_2, C_3 .

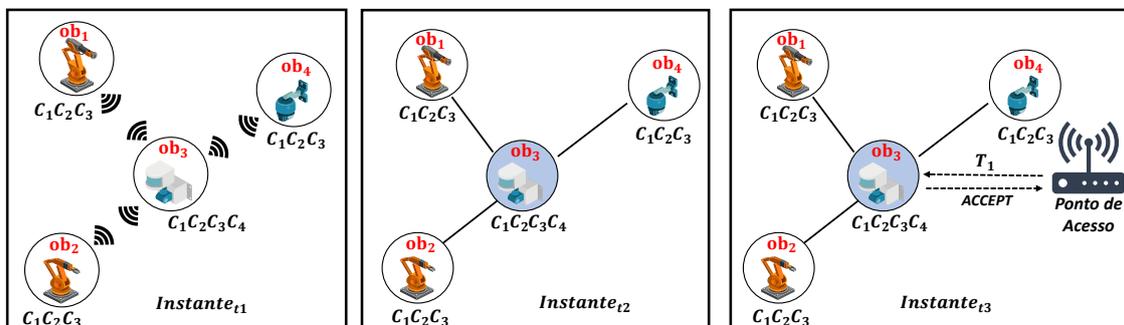


Figura 4. Opera\u00e7\u00e3o na forma\u00e7\u00e3o dos agrupamentos e distribui\u00e7\u00e3o de tarefas

No instante I_{t2} , ocorre a elei\u00e7\u00e3o do l\u00edder, e o objeto com maior n\u00famero de vizinhos \u00e9 o ob_3 , que ent\u00e3o \u00e9 eleito l\u00edder. Com o coordenador de agrupamentos operando desta forma, cada objeto mant\u00e9m suas informa\u00e7\u00f5es de vizinhan\u00e7a e capacidades atualizadas atrav\u00e9s da troca de mensagens. Essa estrutura determina quais objetos da vizinhan\u00e7a espacial s\u00e3o vistos como membro do mesmo agrupamento, al\u00e9m de assegurar uma melhor escalabilidade \u00e0 rede, visto que uma hierarquia baseada em l\u00edderes auxilia na qualidade das informa\u00e7\u00f5es trafegadas. Ela tamb\u00e9m facilita a distribui\u00e7\u00e3o de tarefas entre os objetos da rede, pois cada l\u00edder representa o pr\u00f3prio agrupamento. No instante I_{t3} o AP envia uma mensagem com a tarefa T_1 a ser executada pelo agrupamento O ob_3 , l\u00edder do agrupamento, responde ao AP com a mensagem *TaskAccept* que confirma que seu agrupamento \u00e9 capaz de executar a tarefa enviada e em seguida a executa. Entretanto, como essa \u00e9 uma vis\u00e3o parcial da rede, caso exista algum agrupamento que n\u00e3o seja capaz de realizar a tarefa, basta n\u00e3o enviar a mensagem *TaskAccept* da tarefa em quest\u00e3o.

4. Avalia\u00e7\u00e3o

Esta se\u00e7\u00e3o apresenta a avalia\u00e7\u00e3o de desempenho do mecanismo CONTASKI. O mecanismo foi implementado no simulador NS-3, vers\u00e3o 3.29, permitindo representar um ambiente IIoT distribu\u00eddo com as caracter\u00edsticas de um modelo pr\u00f3ximo ao real. O cen\u00e1rio analisado cria um ambiente similar a de uma ind\u00fastria de manufatura, onde os diversos n\u00f3s IoT est\u00e3o embarcados nos objetos industriais de uma rede IIoT. Esses objetos podem variar conforme o tipo de ind\u00fastria avaliada e seu tipo de fun\u00e7\u00e3o dentro do ambiente. A escolha das capacidades dos n\u00f3s baseia-se nas fun\u00e7\u00f5es de sensoriamento que um objeto IIoT \u00e9 habilitado a desempenhar e segue as fun\u00e7\u00f5es descritas por [Haverkort and Zimmermann 2017].

Os cen\u00e1rios avaliados consistem de 50, 75 e 100 n\u00f3s fixos, distribu\u00eddos uniformemente em uma \u00e1rea retangular de $200m \times 200m$ operando por $750s$. A comunica\u00e7\u00e3o entre os n\u00f3s ocorre atrav\u00e9s do protocolo IPv6, sendo estabelecida uma rede *ad-hoc* no padr\u00e3o IEEE 802.15.4. A Tabela 1 sintetiza os par\u00e2metros aplicados na simula\u00e7\u00e3o. Assume-se que n\u00e3o h\u00e1 perda de pacotes, durante a transmiss\u00e3o, causadas por ru\u00eddo. A inser\u00e7\u00e3o de um tempo de 2 milissegundos \u00e0s mensagens trocadas entre os n\u00f3s, busca evitar a perda de pacotes e gargalo na rede. Al\u00e9m disso, as 10 tarefas de sensoriamento s\u00e3o encaminhadas aos l\u00edderes atrav\u00e9s de um *Access Point* (AP), localizado no centro da rede e equipado

Tabela 1. Configuração das simulações

Parâmetros	Valores
Número de nós	50, 75 e 100
Tarefas despachadas	10
Tempo de execução de tarefas	60s
Tempo de aquecimento	150s
Tempo de simulação	750s
Área da rede	200m x 200m
Protocolo de camada física	IEEE 802.15.4
Protocolo de rede	IPv6
Protocolo de transporte	UDP

Tabela 2. Sensoriamentos

c_1	→ Temperatura
c_2	→ Umidade
c_3	→ Presença
c_4	→ Luz
c_5	→ Sincronização de Maquinário
c_6	→ Movimento
c_7	→ Pressão

com forte sinal de internet para alcançar todos os nós presentes. Assume-se também que o AP é uma entidade sempre disponível. Cada nó possui um conjunto de capacidades de sensoriamento, sendo que as c_1 , c_2 , c_3 são comuns a todos. Além das capacidades comuns, outras são tomadas aleatoriamente entre as demais da Tabela 2, num conjunto final entre 4 e 7 capacidades. O parâmetro de similaridade varia de 0,65 a 1, quanto mais próximo a 1 maior o grau de similaridade entre dois nós. Na avaliação da eficiência usamos métricas similares ao trabalho de [Khalil et al. 2019]. As métricas para mensurar a eficiência do CONTASKI foram: **Número de Agrupamentos (NA)**, **Agrupamentos Aptos à Realizar as Tarefas (AAT)**, **Agrupamentos Ociosos (AO)**, **Número de Tarefas Alocadas (NTA)**, **Número de Tarefas não Alocadas (NTN)**, **Latência do Tempo de Aceite (LTA)** e **Grau de Similaridade (GS)**. Os gráficos apresentados correspondem a média dos resultados obtidos a partir de 35 simulações com um intervalo de confiança de 95%. Uma análise comparativa não foi realizada pois o trabalho mais próximo é de [Khalil et al. 2019], entretanto eles dividem a rede apenas em dois agrupamentos e os objetos executam um tipo de tarefa.

Tabela 3. Equações das métricas de avaliação

$NA = AAT + AO$	$NTN = NTA - AO$	$GS = \frac{ C_{ob1} \cap C_{ob2} }{\sqrt{ C_{ob1} * C_{ob2} }}$
$AAT = NA - AO$	$AO = NA - AAT$	$LTA = \sum_{i=0}^{NA} (TempoDespacho - TempoAceite_i)$

4.1. Análise

Os gráficos da Figura 5 apresentam o desempenho do CONTASKI. O gráfico Aptos x ociosos mostra a quantidade de agrupamentos formados (NA), agrupamentos aptos (AAT) e agrupamentos ociosos (AO). O **NA** está associado diretamente com o parâmetro de similaridade alcançado pelos dispositivos, pois como os nós são configurados com diferentes capacidades essa relação mantém-se forte; visto que todos os nós têm o mínimo de capacidades similares. A precisão na formação dos agrupamentos ocorre pelo fato do cenário não apresentar nenhum tipo de mobilidade, o que facilita as interações entre os dispositivos. Além disso, considera-se uma quantidade não determinista de dispositivos por agrupamento. O total de **AAT** em alguns casos chega a ser de 80% maior que os ociosos devido a relação de similaridade de capacidade entre os nós ao formarem os agrupamentos onde um líder, ao receber uma tarefa, verifica qual o quorum necessário

com base no consenso colaborativo aplicado a todos os participantes do seu agrupamento. Dessa forma, as tarefas são executadas apenas pelos agrupamentos aptos. Este processo ocorre ao inverso quando um líder verifica que as capacidades dos nós membros do seu agrupamento não condizem com aquelas exigidas pela tarefa. Assim, eles preservam seus recursos para a próxima rodada de tarefas.

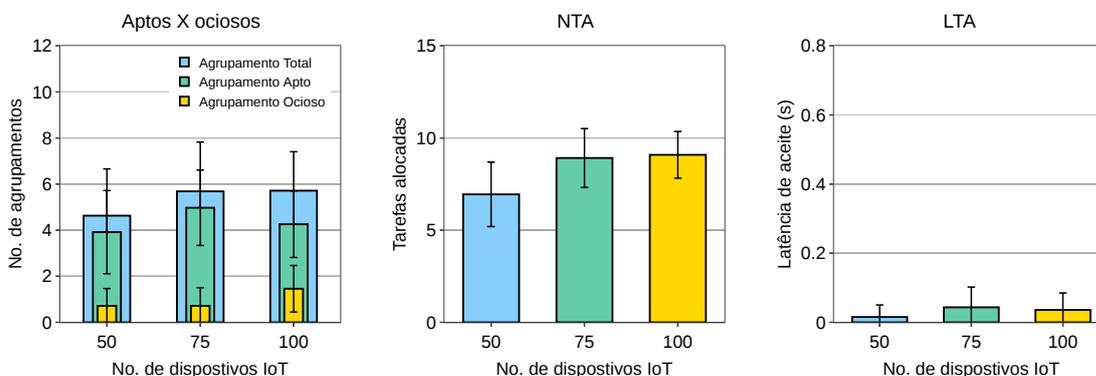


Figura 5. Agrupamentos, tarefas alocadas e latência de aceite pelos líderes

O gráfico da Figura 5 mostra o número de tarefas alocadas (NTA) disponibilizadas pelo AP, a quantidade de tarefas não variam com a quantidade de nós presentes na rede. O valor está relacionado ao tempo de execução de cada tarefa, que é de 60s. Além disso, uma mesma tarefa pode ser executada mais de uma vez. Neste caso mudando apenas a quantidade de tarefas. No cenário com 75 e 100 nós os agrupamentos realizaram cerca de 98% das tarefas despachadas. A distorção no cenário com 50 nós deve-se a configuração aleatória das capacidades e ao número reduzido de nós na rede. O gráfico apresenta os resultados referentes à Latência do tempo de aceite (LTA), que refere-se a diferença entre o tempo de envio da tarefa e o último tempo de aceitação, conforme visto pelo AP. Isso está relacionado à infraestrutura hierárquica baseada em líderes. Como os recursos dos líderes são compatíveis com os participantes do agrupamentos, o líder é responsável por avaliar se o agrupamento pode executar a tarefa e informar o AP sobre a aceitação. A LTA varia de acordo com o número de agrupamentos existentes e agrupamentos aptos para executar uma determinada tarefa. Essa variação observa-se nos diferentes cenários, com 100 nós, a latência média foi de 35ms, sendo menor que aquele com 75 nós, que alcançou 43ms. A menor latência média observada foi de 15ms, com 50 nós. Isso decorreu pela localização do nós líderes presentes na rede, visto que os mais próximos ao AP apresentam um tempo de resposta menor para o aceite das tarefas.

O gráfico da Figura 6 (a) mostra os agrupamentos aptos (AAT) e o ociosos (AO) ao longo do tempo para cada despacho de tarefa pelo AP. Cada ponto no gráfico indica que uma tarefa está sendo despachada e executada pelos agrupamentos aptos, as linhas pontilhadas significam AO e as linhas sólidas os AAT a realizar a tarefa. No instante 0s a 150s ocorreu o *warm up* da rede, a primeira tarefa foi despachada no instante 150s e durou 60s, a próxima tarefa foi despachada no instante 210s, e assim sucessivamente até a última tarefa despachada. Observa-se que os AO agrupamentos ociosos são significativamente maiores que os AAT. Isso ocorre porque nem todos os agrupamentos necessitam realizar determinada tarefa, e assim eles aguardam para quando forem requisitados estejam prontos. Logo, todas as 10 tarefas foram despachadas até 690s, sendo que a última tarefa termina até 750s e todas tiveram ao menos um agrupamento apto realizando-as. O

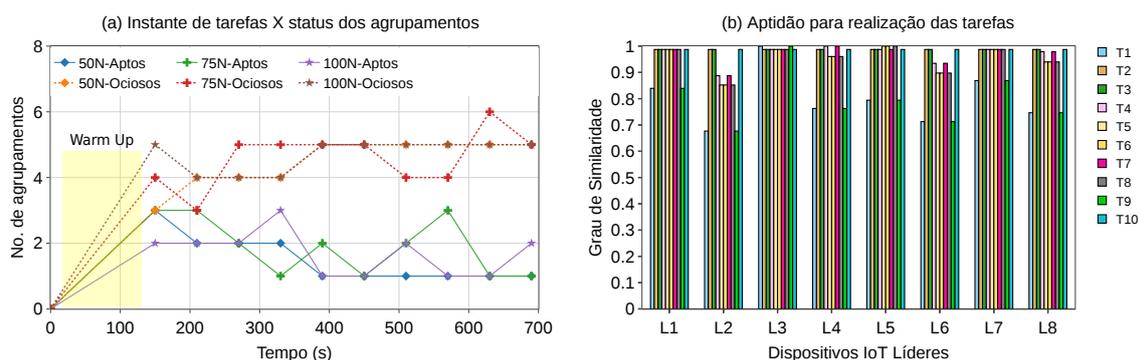


Figura 6. Agrupamentos aptos e ociosos e similaridade entre tarefas e líderes

gráfico da Figura 6 (b) apresenta o grau de similaridade (GS) entre tarefas a serem realizadas e os líderes dos agrupamentos na sétima das trinta e cinco rodadas com 50 nós. Esta rodada exibiu recursos dos nós semelhantes aos recursos da tarefa e maior quantidade de líderes e explicita a existência do consenso colaborativo entre os nós, já que apenas os agrupamentos aptos executam as tarefas, permitindo aos outros agrupamentos economizar seus recursos. As barras coloridas representam cada tarefa que o AP despachou ao longo da simulação. Os líderes verificam a similaridade entre as capacidades e as tarefas exigidas e aceitam e executam a tarefa cuja similaridade é igual a 1.

5. Conclusão

Este trabalho apresentou o mecanismo CONTASKI para apoiar o serviço de alocação de tarefas em objetos de uma rede IIoT. Ele organiza a rede em agrupamentos com base na similaridade das capacidades dos dispositivos e nas capacidades dos dispositivos vizinhos. O mecanismo aplica o consenso colaborativo relacional para gerenciar e distribuir a tarefas entre os agrupamentos, considerando as capacidades por eles informadas e as tarefas a serem executadas. Essas estratégias conjuntas conseguem manter a rede organizada hierarquicamente permitindo que os participantes atuem de acordo com suas capacidades. Os resultados demonstram a eficácia do CONTASKI considerando a relação entre capacidade e tarefas a serem executadas pelos objetos IIoT e possibilitou um melhor aproveitamento de recursos, aumentando a qualidade das informações disponibilizadas pelos objetos à aplicação. Como trabalhos futuros, pretende-se avaliar a alocação de múltiplas tarefas simultâneas entre os objetos, diferentes contextos de redes IoT e com diferentes tipos de mobilidades. Além disso, uma avaliação do consumo de energia e a comparação com outros mecanismos de alocação de tarefa presentes na literatura.

Referências

- Aazam, M., St-Hilaire, M., Lung, C.-H., Lambadaris, I., and Huh, E.-N. (2018). Iot resource estimation challenges and modeling in fog. In *Fog Computing in the Internet of Things*, pages 17–31. Springer.
- Borgia, E. (2014). The internet of things vision: Key features, applications and open issues. *Computer Communications*, 54:1–31.
- Botta, A., De Donato, W., Persico, V., and Pescapé, A. (2016). Integration of cloud computing and internet of things: a survey. *Future generation computer systems*, 56:684–700.
- Chen, I.-R., Guo, J., and Bao, F. (2016). Trust Management for SOA-Based IoT and Its Application to Service Composition. *IEEE Transactions on Services Computing*, 9(3):482–495.

- Colistra, G., Pilloni, V., and Atzori, L. (2014). Task allocation in group of nodes in the iot: A consensus approach. In *2014 IEEE International Conference on Communications (ICC)*, pages 3848–3853. IEEE.
- Evangelista, D., Mezghani, F., Nogueira, M., and Santos, A. (2016). Evaluation of sybil attack detection approaches in the internet of things content dissemination. In *2016 Wireless Days (WD)*, pages 1–6. IEEE.
- Fang, W., Zhang, Q., Liu, M., Liu, Q., and Xia, P. (2018). Earning maximization with quality of charging service guarantee for iot devices. *IEEE Internet of Things Journal*, 6(1):1114–1124.
- Ghanbari, Z., Navimipour, N. J., Hosseinzadeh, M., and Darwesh, A. (2019). Resource allocation mechanisms and approaches on the internet of things. *Cluster Computing*, pages 1–30.
- Gielow, F., Jakllari, G., Nogueira, M., and Santos, A. (2015). Data similarity aware dynamic node clustering in wireless sensor networks. *Ad Hoc Networks*, 24:29–45.
- Gubbi, J., Buyya, R., Marusic, S., and Palaniswami, M. (2013). Internet of things (iot): A vision, architectural elements, and future directions. *Future generation computer systems*, 29(7):1645–1660.
- Haverkort, B. R. and Zimmermann, A. (2017). Smart industry: How ict will change the game! *IEEE internet computing*, 21(1):8–10.
- Jin, Y., Jin, J., Gluhak, A., Moessner, K., and Palaniswami, M. (2011). An intelligent task allocation scheme for multihop wireless networks. *IEEE Transactions on Parallel and Distributed Systems*, 23(3):444–451.
- Khalil, E. A., Ozdemir, S., and Bara'a, A. A. (2019). A new task allocation protocol for extending stability and operational periods in internet of things. *IEEE Internet of Things Journal*.
- Khalil, E. A., Ozdemir, S., and Tosun, S. (2018). Evolutionary task allocation in internet of things-based application domains. *Future Generation Computer Systems*, 86:121–133.
- Kim, M. and Ko, I.-Y. (2015). An efficient resource allocation approach based on a genetic algorithm for composite services in iot environments. In *2015 IEEE International Conference on Web Services*, pages 543–550. IEEE.
- Korsah, G. A., Stentz, A., and Dias, M. B. (2013). A comprehensive taxonomy for multi-robot task allocation. *The International Journal of Robotics Research*, 32(12):1495–1512.
- Pilloni, V. and Atzori, L. (2011). Deployment of distributed applications in wireless sensor networks. *Sensors*, 11(8):7395–7419.
- Pilloni, V. and Atzori, L. (2017). Consensus-based resource allocation among objects in the internet of things. *Annals of Telecommunications*, 72(7-8):415–429.
- Qin, W., Chen, S., and Peng, M. (2020). Recent advances in industrial internet: insights and challenges. *Digital Communications and Networks*, 6(1):1–13.
- Qiu, T., Chen, N., Li, K., Atiquzzaman, M., and Zhao, W. (2018). How can heterogeneous internet of things build our future: A survey. *IEEE Communication Surveys & Tutorials*, 20(3):2011–2027.
- Xu, L. D., Xu, E. L., and Li, L. (2018). Industry 4.0: state of the art and future trends. *International Journal of Production Research*, 56(8):2941–2962.
- Zhong, R. Y., Xu, X., Klotz, E., and Newman, S. T. (2017). Intelligent manufacturing in the context of industry 4.0: a review. *Engineering*, 3(5):616–630.