UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

GABRIEL MARTINS LEAL

# Abordagem baseada em Redes Centradas à Informação e Redes Definidas por Software para Suporte de Aplicações Militares Críticas

Dissertação apresentada como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação

Orientador: Prof. Dr. Edison Pignaton de Freitas

Porto Alegre
2020

**CIP — CATALOGAÇÃO NA PUBLICAÇÃO**

Martins Leal, Gabriel

Abordagem baseada em Redes Centradas à Informação e Redes Definidas por Software para Suporte de Aplicações Militares Críticas / Gabriel Martins Leal. – Porto Alegre: PPGC da UFRGS, 2020.

68 f.: il.

Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR–RS, 2020. Orientador: Edison Pignaton de Freitas.

1. Redes Centradas à Informação. 2. Redes Definidas por Software. 3. IoT. 4. IoBT. 5. Redes Militares. I. Freitas, Edison Pignaton de. II. Título.

## RESUMO

A tecnologia de informação e integração de dispositivos conectados em redes vêm crescendo abruptamente, de forma que a complexidade de sua interação também cresce, demandando estudos para prover uma melhor capacidade de atender as necessidades de aplicações que utilizam o conjunto de dados obtidos e transmitidos em diversos ambientes.

Este trabalho tem como o objetivo prover uma medida que melhore e flexibilize a comunicação entre dispositivos de redes através de redes definidas por software (Software-defined Networks) e redes centradas à informações (Information-centric Networks). Com a junção destas duas abordagens foi possível modelar uma arquitetura para auxiliar missões críticas militares em um contexto de internet das coisas (Internet of Things) onde diversos dispositivos trabalham em conjunto para um objetivo. No cenário operacional apresentado foram feitos diversos experimentos partindo da arquitetura exibida neste trabalho que comprova e compara diretamente algumas métricas com redes convencionais IP, e redes IP definidas por software. Com os experimentos apresentados, foi possível mostrar que a rede obteve uma considerável queda em termos de latência e do uso da rede que pode ocasionar sobrecarga, além de apresentar o aumento de confiabilidade e da resiliência da rede.



**Palavras-chave:** Redes Centradas à Informação. Redes Definidas por Software. IoT. IoBT. Redes Militares.

**Approach based on Information-centric Networks and Software-defined Networks to Support Critical Military Applications**

## ABSTRACT

The information technology and integration of networked devices has been growing abruptly, in a way that the complexity of their interaction also grows, demanding studies to provide a better environment to meet the needs of applications that use the data set obtained and transmitted in various environments.

This work aims to provide an approach that improves and eases communication between network devices through software-defined networks and information-centric networks. By bringing these two approaches together, it was possible to model an architecture to aid military-critical missions in an Internet of Things context where multiple devices work together towards one goal. In the presented operational scenario, several experiments were made starting from the architecture shown in this work that directly proves and compares some metrics with conventional IP networks, and software-defined IP networks. With the experiments presented, it was possible to show that the network obtained a considerable decrease in latency and network usage that can cause overload, besides presenting the increase of reliability and network resilience.

**Keywords:** Information-centric Network, Software-defined Network, IoT, IoBT, Mititary Network.

## LISTA DE ABREVIATURAS E SIGLAS

| | |
|---|---|
| SDN | Software-defined Network (Redes Definidas por Software) |
| ICN | Information-centric Network (Redes Centradas à Informação) |
| IoT | Internet of Things (Internet das Coisas) |
| IoBT | Internet of the Battle Things (Internet das Coisas para Campo de Batalha) |
| C2 | Comando e Controle |
| NATO | North Atlantic Treaty Organization |
| BN | Battlefield Network (Redes para Campo de Batalha) |
| NDN | Named-data Network (Redes com Dados Nomeados) |
| CCN | Content-centric Network (Redes Centradas a Conteúdo) |
| AoI | Area of Interest (Área de Interesse) |
| eBPF | Extended Berkeley Packet Filter |

## LISTA DE FIGURAS

## LISTA DE TABELAS

**SUMÁRIO**

# 1 INTRODUÇÃO

Os avanços nas tecnologias de informação e comunicação estão ocasionando o rápido desenvolvimento da Internet das Coisas (IoT). Em vários cenários da sociedade é possível observar que o crescimento da interação entre humanos e dispositivos eletrônicos é promissor. Portanto, várias formas de melhorar a comunicação entre os elementos envolvidos vêm sido estudadas por diversas áreas. Alguns problemas como interpretação de semântica de dados, eficiência em termos de transmissão de dados, otimizações de uso de recursos como bateria, entre diversos outros são alguns dos problemas listáveis.

As "coisas" estão se tornando não apenas capaz de adquirir uma maior quantidade de dados, mas também capaz de funcionar em vários graus de autonomia dependendo de quão "inteligentes" elas são. Neste contexto, os humanos precisarão interagir eficientemente com elas em formas que ainda não são totalmente claras, para que essa junção entre humanos e coisas inteligentes possa produzir um desempenho eficiente, além de confiabilidade, robustez e agilidade, que são características desejadas na maioria das aplicações no mundo real em que se há uma grande interação entre humanos e coisas, como por exemplo em uma grande cidade, ou uma casa inteligente, operações em campos de batalha, e várias outras.

Um grande problema presente em muitas das redes atuais é o crescimento da latência. Embora haja muitas implementações para otimizar a latência através de software, as formas de contornar este problema em sua maioria são orientados à localizações fixas. Tal abordagem limita bastante a flexibilidade de redução do enlace físico a ser percorrido, fazendo com que os dados trafegados em aplicações de tempo real não sejam ágeis o suficiente, principalmente para aplicações críticas. O problema de pesquisa abordado nesse trabalho é justamente a como otimizar a latência em redes de aplicação crítica com os recursos presentes no estado da arte atual.

Grande parte das aplicações exigem requisitos críticos e precisos para a realização eficiente das ações desejadas, como por exemplo aplicações de defesa civil, corpo de bombeiros, e até mesmo soluções militares em campos de batalha. Neste trabalho será explorado principalmente exemplos em campos de batalha, uma vez que a criticidade, confiabilidade e restrições das informações são mais evidentes, o que torna em um bom cenário de exemplificação da necessidade de aprimoramento da comunicação dos dispositivos, tanto das coisas entre si, quanto da comunicação entre coisas e humanos.

No cenário militar, para auxiliar no andamento e melhoria de qualquer operação

no qual está envolvido um campo de batalha com dispositivos inteligentes, é utilizada uma abordagem chamada Comando e Controle (C2). Para um melhor desempenho, o sistema C2 (KOTT; ALBERTS, 2017) precisará administrar e gerenciar interações complexas feitas entre humanos e coisas inteligentes, assim como entender as diferenças entre cognição humana e inteligência de máquinas para criar a abordagem C2 mais apropriada para o conjunto. Por exemplo, coisas inteligentes podem lidar com uma massiva quantidade de dados muito superior ao que humanos podem lidar e fazer decisões instantaneamente, baseado em parâmetros pré-definidos e algoritmos. Por outro lado, humanos são capazes de enxergar diferentes perspectivas em situações críticas e achar soluções baseadas em uma visão não antecipada.

Portanto, o sistema C2 precisa ser construído de forma a ser capaz de aproveitar de forma ótima seus componentes. Para isso, as coisas inteligentes devem ser usadas para coletar e processar grandes quantidades de dados produzidos no campo de batalha, de acordo com sua semântica, além de objetivos e prioridades bem definidos, compartilhando as informações resultantes relevantes com os soldados no campo e tomadores de decisão no *back-end*, de acordo com suas necessidades para fazer decisões efetivas e funcionais.

Devido à dinamicidade do ambiente e da heterogeneidade dos atores participantes das ações, não existe abordagem que condiz com todas as missões e circunstâncias ao mesmo tempo. A abordagem C2 escolhida depende de quão centralizada ou distribuída o sistema C2 opera, levantando diferentes formas de alocar permissões de decisão, interação entre atores e distribuição da informação. Quando as missões e circunstâncias mudam rapidamente, é necessário obter Agilidade C2, que é composta pela mudança de alocação de permissões, padrões de interação e disseminação de informação de acordo com o desejado (ALBERTS; AL., 2013).

Para obter a Agilidade C2 no contexto de Internet das coisas para campos de batalha (*Internet of Battle Things ou IoBT*) que possuem um ambiente de larga escala e heterogêneo, será necessário identificar novos modelos e técnicas de atuação. Do ponto de vista de rede, as tecnologias orientadas à semântica de dados como Redes Definidas por Informação (*Infomation-centric networking ou ICN*) juntamente com Redes Definidas por Software (SDN) têm um papel-chave.

ICN é um paradigma que altera a convencional lógica de rede baseada em endereços (*host-centric*) para uma abordagem em que o ponto-chave é a informação, que pode ser representada por conteúdo ou dados não processados (AHLGREN et al., 2012).

Este paradigma suporta funcionalidades de conexão intermitente, através de ar-

mazenamento de dados dentro da própria rede, em nós intermediários e também dados replicados, que possibilitam caching eficiente, por exemplo. Além disso, é considerada a mobilidade dos nós e acessos múltiplos como algo complementar, ao invés de uma exceção que prejudica o sistema, e também suporta nativamente *anycast*, *multicast*, e disseminação de dados através de *broadcast*.

Essas características possuem o fim de melhorar a performance geral da rede, para prover uma rede mais robusta e escaláve, desacoplando os dados de seus produtores, fazendo a localidade e aplicação dos dados independente, melhorando vários aspectos de acessibilidade de dados. As funcionalidades descritas tornam ICN um forte candidato para redes de campo de batalha (*Battlefield Networks* ou *BN*), como em um ambiente de operações militares a maioria das características mencionadas anteriormente são encontradas, como mobilidade, conexões intermitentes e a necessidade de escalabilidade e robustez da rede. Algumas das características e necessidades mencionadas já foram exploradas em trabalhos anteriores, como em (ZACARIAS et al., 2017), particulamente referente às Redes Tolerantes a Atraso. Apesar de o trabalho (ZACARIAS et al., 2017) apresentar uma abordagem extremamente útil, ele não cobre outros aspectos, como desacoplamento da informação de seus produtores, além do suporte de *anycast* e *multicast*.

Outros designs variantes de ICN também devem ser levantados, como as Redes de dados nomeados (*Named-Data Networking* ou *NDN*) e redes centradas por conteúdo (*Content-Centric Networking* ou *CCN*) (AHLGREN et al., 2012). Ambos implementam os princípios de ICN, mas a utilização em larga escala dessas abordagens isoladamente está longe da realidade pela necessidade da mudança de hardware e software da maioria dos sistemas operando atualmente. É mais provável que ICN seja aplicado em conjunto de outras tecnologias baseadas em IP, como por exemplo o SDN (ZURANIEWSKI et al., 2017).

SDN é um paradigma que visa prover flexibilidade na gerência da rede em geral separando a infraestrutura da rede em diferentes planos (NUNES et al., 2014), o plano físico e o plano de controle. O plano de dados pode ser programado para satisfazer requisitos de aplicações, através da manipulação dos fluxos de rede, uma vez que a rede, através de um controlador, possui visão global e é capaz de efetuar ações inteligentes de forma dinâmica. O SDN pode ser aplicado em praticamente qualquer rede atualmente sem muitas modificações custosas (tanto de infraestrutura quanto de software). Algumas ações feitas na rede são mudanças de roteamento até mesmo melhorias em políticas de segurança da rede. O controlador faz o papel de comunicar com o plano físico, os roteadores

e *switches*, onde este fornece uma interface programável. Trabalhos anteriores já exploraram o uso de SDN em cenários de BN como em (ZACARIAS et al., 2017) e em (NOBRE et al., 2016). O segundo trabalho foca mais em melhorar a comunicação entre nós com abundância de recursos (como por exemplo, navios de guerra e aviões), o trabalho atual foca em nós com recursos limitados de dispositivos utilizados por soldados e tropas no campo de batalha. E enquanto o primeiro utiliza o SDN em uma forma convencional através do IP, a proposta aqui apresentada é utilizar esta abordagem em conjunto com ICN para suportar operações orientadas a dados.

Devido à falta de abordagens centradas à informação no estado da arte endereçando os cenários IoBT, este trabalho propõe a junção das abordagens ICN e SDN para suportar Agilidade C2 neste ambiente operacional. A abordagem proposta se beneficia da programabilidade oferecida pelo SDN e da habilidade de lidar com dados e informações de uma forma distribuída e desacoplada nativa do ICN. Esta abordagem visa em melhorar a capacidade da interação entre humanos e dispositivos inteligentes, além de prover uma solução arquitetural para suportar operações militares.

O objetivo principal deste trabalho é prover otimização na latência para cenários de aplicações de tempo real, e os objetivos específicos são: aumentar a confiabilidade da rede reduzindo perda de dados, prover meios de segurança nos pacotes através de políticas implementadas através da junção das redes ICN e SDN, redução de sobrecarga da rede, de forma a torná-la mais robusta para transporte de dados massivos.

A principal contribuição desse trabalho foi o desenvolvimento de uma arquitetura baseada na combinação de SDN com ICN, com a qual foi possível obter uma melhoria em termos de latência comparada com redes tradicionais IP e redes SDN, além da redução do uso dos enlaces da rede, tornando-a capaz de trafegar uma maior quantidade de dados sem sobrecarga. Também foi possível aumentar a confiabilidade e tolerância à perdas devido ao armazenamento interno da rede através do mecanismo de *caching*.

O trabalho segue uma estrutura onde a seção 2 dá uma breve explicação dos elementos em que o trabalho tomou como base, na seção 3 alguns trabalhos são apresentados que seguem uma abordagem semelhante, porém com algumas diferenças do apresentado nesta proposta, a seção 4 apresenta um cenário direto de aplicação em que é possível observar com facilidade um caso de uso para a arquitetura que é explicada na seção 5, juntamente com todos os detalhes dos elementos, formas de comunicação, e a implementação para a validação do trabalho. A seção 6 utiliza a implementação explicada na seção 5 para mostrar os resultados que foram obtidos a partir da aplicação da arquitetura com-

parando diretamente com outras redes convencionais, e finalmente a seção 7 conclui os pontos importantes e relevantes do trabalho.

# 2 REVISÃO BIBLIOGRÁFICA

Esta seção abrange uma revisão dos conceitos-chave para a compreensão do funcionamento da arquitetura e o domínio de aplicação escolhido para o estudo de caso.

## 2.1 Comando e Controle (C2)

Operações militares modernas estão mudando o conceito de guerra devido aos avanços da tecnologia e da ciência. Estas operações variam desde operações típicas de guerra, como na Síria, até mesmo às operações como as executadas pelas Forças de Paz das Nações Unidas na África e no Haiti, e operações humanitárias para lidar com crises de refugiados na Europa e na América do Sul, assim como operações específicas executadas pela Garantia de Lei e Ordem no governo brasileiro, onde o exército brasileiro é enviado para "pacificar" regiões inteiras de cidades sob o domínio de facções de crime organizado. Nestas situações, os soldados devem agir juntamente com os civis seja de organizações estatais ou não em ambientes urbanos, lidando com um ambiente incerto, o que causa o crescimento de relações sociais complexas destes diversos atores.

Essas mudanças não apenas amplificam a importância do Comando e Controle (*Command and Control* ou *C2*), mas também levantam a necessidade de uma rede eficiente para o sistema C2. Trabalhos anteriores (KOTT; ALBERTS, 2017) definem C2 como a gerência ou governança de organizações militares e esforços, os quais são definidas por cinco funções (*Command*, *Control*, *Sensemaking*, *Execution*, e *Situation Monitoring*) necessárias para alcançar os efeitos desejados no campo de batalha, como é possível. O *Command* é responsável por estabelecer as intenções (isto é, objetivos e prioridades), e de criar condições (regras de combate e outras métricas) para o sucesso da operação. *Control* envolve todas as outras funções executadas repetidas em ciclos. Aqui entra o conceito de agilidade C2. Quanto mais rápido o ciclo de controle é concluído, sem comprometer a qualidade de todas as etapas, mais eficiente é o sistema C2.

Em outras palavras, *Sensemaking* (coletar, processar e compartilhar informações), *Execution* (ações reais no campo de batalha) e *Situation Monitoring* (avaliação em tempo real dos efeitos das operações em capo) devem ser cuidadosamente executados e integrados a uma abordagem C2 que melhor se encaixa para a missão atual. Ao adicionar dispositivos inteligentes e heterogêneas para este cenário com diversos atores humanos fará com que a abordagem escolhida seja capaz de maximizar a performance do que eles

fazem, de ambos humanos e dos dispositivos inteligentes, para que esta integração seja capaz de atingir os objetivos e prioridades da operação, definidos pelo *Command*.

A escolha da abordagem C2 que melhor se encaixa para determinada operação militar não é uma tarefa trivial e, dependendo da dinamicidade da situação, onde as missões e/ou as circunstâncias do ambiente e dos atores variam, a abordagem atual pode precisar ser replanejada. A *North Atlantic Treaty Organization* (NATO) produziu um relatório detalhado (ALBERTS; AL., 2013) onde este problema é discutido e um modelo conceitual de agilidade C2 foi desenvolvido. A NATO concluiu que abordagens C2 podem ser categorizadas em como os direitos de decisão são alocados, como os diferentes atores interagem e como a informação é distribuída. Tais características das diferentes abordagens de C2 podem ser sobremaneira auxiliadas por mecanismos de rede que deem suporte a uma rápida adaptação da rede às constantes mudanças do seu ambiente operacional.

## 2.2 Internet of Battle Things

Seguindo a mesma linha evolucionária que a Internet das Coisas (*Internet of Things* ou *IoT*) segue, as redes focadas para campos de batalha estão focando para a Internet das Coisas de Batalha (*Internet of Battle Things* ou *IoBT*). De acordo com o avanço das tecnologias de comunicação militares, a inteligência em dispositivos embarcados está encaminhando o desenvolvimento da IoBT.

Ao contrário de sua versão civil (i.e. IoT), a IoBT é uma rede de dispositivos físicos, veículos, sensores e qualquer objeto que tem como objetivo adquirir ou processar data que pode ser compartilhado com outros nós da rede. No campo de batalha, estes objetos são equipamentos militares e veículos, como por exemplo, armas inteligentes, dispositivos embarcados, drones, sistemas vestíveis de monitoramento de saúde, entre outros. Eles são capazes de prover uma variedade de dados sobre diferentes eventos e aspectos de interesse para auxiliar decisões baseadas em informações. Por exemplo, eles podem informar a movimentação inimiga em uma determinada região, ou como está a condição da saúde das tropas envolvidas em uma missão.

Como mencionado anteriormente, estas coisas no campo de batalha não são apenas meros sensores adquirindo e provendo dados, mas dispositivos inteligentes capazes de processar dados, gerar informações e tomar decisões por si só. Um exemplo em um cenário operacional de uma grande metrópole, como nas operações de Garantia da Lei e da Ordem brasileira, onde é utilizado drones patrulhando regiões metropolitanas de

difícil acesso, como favelas, para adquirir imagens aéreas. Estas imagens podem auxiliar tropas que estão entrando nesses nichos desestruturados de residências em que possíveis inimigos hostis estão escondidos. Enquanto adquire imagens, os drones processam as mesmas para detectar pessoas segurando armas, por exemplo. Uma vez que tal padrão é detectado e a partir da informação de movimento das tropas em campo, eles são capazes de decidir quais nós na rede eles devem encaminhar dados para que informações críticas temporalmente cheguem nas tropas envolvidas em determinadas ações.

Proporções de larga escala e heterogeneidade também são características importantes da IoBT, da mesma forma que é para a IoT. Entretanto, em cenários tradicionais de campos de batalha, as condições para estas características estão relativamente sob controle, pelo fato de que majoritariamente apenas equipamentos militares irão interagir entre si. Por outro lado, em conflitos em cenários urbanos como mencionados acima, essas características são ainda mais presentes. Isso acontece porque nestes cenários os equipamentos militares IoBT podem interagir com dispositivos IoT de civis, seja para adquirir dados a respeito do estado do ambiente e informações genéricas locais, ou até mesmo para utilizar como meio de encaminhamento de dados. Nesta situação, a rede completa pode se tornar muito maior e com um nível de heterogeneidade que não é possível inferir.

## 2.3 Software-Defined Network

As redes em geral vêm se tornando mais dinâmicas e diversas com o passar dos anos devido às proporções que seu uso alcançou. Os dispositivos de *hardware*, entretanto, se mantiveram atrelados a protocolos fixos em seu *firmware* para executar funcionalidades tais como encaminhamento e gerenciamento dos fluxos, o que segue o caminho contrário do desenvolvimento de ambientes resilientes e de redes que devem se adaptar ao meio de acordo com seus requisitos.

Para contornar essa limitação, surgiram as SDN, que são capazes de atender especifidades em redes isoladas e globais, oferecendo programabilidade e recursos para um melhor gerenciamento de rede. Isso é feito através do desacoplamento do *hardware* do controlador das ações, isso é, separação do plano de dados e o plano de controle, que possibilita o desenvolvimento de diversas funcionalidades complexas em ambientes heterogêneos (NUNES et al., 2014).

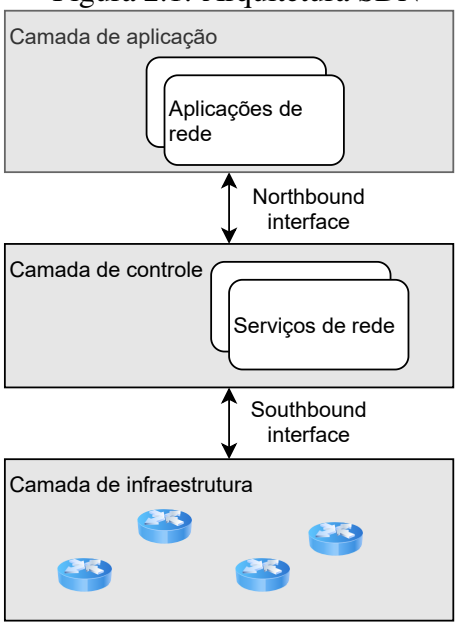
Na abordagem SDN, o controlador é responsável por decidir todos os encaminhamentos de fluxo dentro da rede, fazendo com que os dispositivos de encaminhamento, ou

*switches*, redirecionem os fluxos de acordo com o que o controlador ordenou. Estas características fornecem grandes vantagens em diversos aspectos: i) o controlador possui uma visão global sobre a rede, fazendo com que ele consiga analisar todos os fluxos existentes na mesma, e executar encaminhamentos de forma ótima dentro daquele contexto, o que não é possível sem o SDN, uma vez que a visão da rede dos *switches* é limitada; ii) armazenamento e processamento estatísticos podem ajudar a modificar a rede de acordo com o histórico de fluxos, aplicando aprendizagem de máquina e inteligência artificial para aplicar a abordagem mais eficaz; iii) as redes não permanecem limitadas pelo *firmware* e protocolos estáticos, podendo aplicar diversos protocolos, políticas de segurança, pré-processamento e filtragem de pacotes, entre outras.

A arquitetura SDN pode ser observada na figura Figura 2.1, onde é composta por 3 camadas, a camada de aplicação em que é a aplicação que efetivamente utilizará a rede, que é mediada pela *northbound interface*, que efetua a comunicação entre componentes de rede com as aplicações.

A camada de controle é constituída majoritariamente pelo controlador que consegue efetuar decisões em cima da rede analisando os dados, uma vez que o controlador possui uma visão global da rede. E ela se comunica com a camada de infraestrutura através de uma *southbound interface*, onde a interface mais conhecida é o OpenFlow.

Figura 2.1: Arquitetura SDN




O OpenFlow que opera através de regras de encaminhamentos definidas por *match* e *actions*. É possível definir campos e métricas que os elementos encaminhadores da camada de infraestrutura detectam se o pacote ou fluxo casam, e encaminham caso ele

atenda o requisito de encaminhamento, que é o *match*. Esse requisito pode ser constituído por campos como endereços IP de origem ou destino, endereços MAC, portas de saída dos dispositivos de encaminhamento, tipos de protocolos, tais como TCP, UDP ou ICMP, entre vários outros.

Uma vez que o pacote dá um *match* em alguma regra de encaminhamento, uma *action* é efetuada, de forma que as ações para esse fluxo ou pacote seja aplicada, tal como encaminhar para um host específico, ou descartar o pacote, entre outras funções.

Sempre que um pacote que entra na rede de forma que este não possui uma rota pré-definida em algum encaminhador de pacotes presente na camada de infraestrutura, é invocado um *PACKET IN*, que é uma operação efetuada no controlador de analisar e determinar o caminho que o pacote vai percorrer, de acordo com as políticas e prioridades da rede. Na operação de *PACKET IN*, é possível criar novas regras de encaminhamento e retornar estas regras para o dispositivo encaminhador que não tinha uma rota traçada para o pacote atual.

Alguns exemplos de controladores que possibilitam a implementação do protocolo OpenFlow são o Ryu e o Floodlight. Além do uso de OpenFlow puro existem outras formas de gerenciar a comunicação SDN, como por exemplo a linguagem P4 (BOSSHART et al., 2014) que é uma linguagem de programação de alto nível que trabalha em conjunto com o OpenFlow para implementação de configurações de rede independentes de protocolos e hardware.

Neste trabalho, o foco é o OpenFlow que será utilizado para otimizar vários aspectos da rede que serão apresentados nas próximas seções.

## 2.4 Information-centric Network

Embora as redes atuais baseadas em IP utilizem portas e endereços para mapear e rotear fluxos de dados, a maioria do tráfego é na prática mais focada na entrega de seu conteúdo. O uso da rede migrou de comunicações fim-a-fim para um modelo centrado a dados, focando no quê (conteúdo) ao invés de onde (endereços). Redes centradas à informação ( *Information-centric Network* ou ICN) representam um novo paradigma que muda a semântica dos serviços de rede para permitir que a recuperação dos dados seja baseada em Objetos com Dados Nomeados (*Named Data Objects* ou *NDO*), que são representados por dois tipos distintos, *INTEREST* ou *DATA*, semelhante a um paradigma produtor e consumidor (AHLGREN et al., 2012). Entidades que desejam consumir um

dado envia mensagens de *INTEREST* com o nome do dado para ser recuperado, e quem provê o dado retorna uma mensagem *DATA* que responda adequadamente.

Um dos pontos mais fortes das redes ICN é que os dados estão armazenados de forma distribuída, isto é, vários nós da rede possuem o mesmo dado. Estes dados por definição da própria rede podem possuir a mesma versão ou versões diferentes (AHLGREN et al., 2012). Todo nó é capaz de armazenar e/ou transmitir um dado, possibilitando assim não só a replicação do dado em vários pontos das redes mas também a possibilidade de *caching*, desvinculando um dado de um endereço único, isto é, um dado desacoplado fisicamente.

Para este paradigma funcionar cada nó da rede possui três tabelas: a tabela de roteamento, que traduz um nome para uma interface física para assim encaminhar o dado; a tabela de interesses pendentes, cujo torna este nó capaz de responder qualquer mensagem *INTEREST* armazenado assim que um pacote *DATA* passa por ele. A abordagem de *caching* se torna um ponto-chave em vários aspectos, uma vez que, semelhante às redes de fornecimento de conteúdos (*Content Delivery Networks* ou *CDN*), uma vez que quando um mesmo dado está armazenado em vários pontos distribuídos da rede a latência fim a fim é diminuída, além de diminuir a sobrecarga da rede em geral.

Essa assinatura criptografada das mensagens *DATA* (vinculação nome-conteúdo) melhora a segurança a partir do próprio dado, longe dos *hosts* e servidores. Ao desacoplar remetentes e destinatários, o ICN também provê mobilidade intrínseca, na qual é ideal para cenários operacionais *ad hoc*, além disso, devido ao *caching* e à replicação nos nós da rede, os dados são mantidos mais próximo dos consumidores, fazendo com que a distribuição de informação seja aprimorada comparada com as redes tradicionais IP, em termos de latência e uso de largura de banda.

Essas características potencializam a capacidade de segurança, mobilidade e *caching* dentro da própria rede, tornando a rede ICN uma forte candidata para suportar IoT em larga escala e redes fim-a-fim (BACCELLI et al., 2014).

# 3 TRABALHOS RELACIONADOS

Este trabalho propõe uma arquitetura para implementar as diretivas de Comando e Controle como parte de uma infraestrutura de rede, utilizando a combinação das abordagens SDN e ICN como base para a aplicação C2. O ICN é utilizado para distribuir informação para todos os consumidores de dados na rede, e o SDN responsável por controlar os padrões de interação entre os nós e também como um mecanismo que dispõe da possibilidade da operação de redes ICN em uma infraestrutura convencional IP. Essas diretivas C2 e a implementação de rede proposta pode ser utilizada para diversas implementação para missões críticas, desde típicas operações de Garantia de Lei até operações de recuperação de desastres (OH; LAU; GERLA, 2010) (MELAZZI; CHIARIGLIONE, 2013).

As redes ICN muda a semântica dos serviços de redes para permitir a recuperação dos dados baseados em objetos nomeados. O ICN visa responder o fato de que o tráfego atual das redes está direcionado à entrega de conteúdo. O uso das redes migrou das comunicações fim-a-fim para um modelo centrado à dados, focando principalmente no que (conteúdo) ao invés de onde (endereços). A partir disso, os objetos nomeados podem ser classificados em dois tipos distintos, *INTEREST* e *DATA* (AHLGREN et al., 2012). Os consumidores requisitam dados através de mensagens de *INTEREST* com o nome dos mesmos, e os produtores (ou nós intermediários que efetuam o *caching* dos dados) retornam uma mensagem *DATA* que casa com a requisição.

Assinaturas criptografadas de mensagens de *DATA* (vinculação de nome e conteúdo) incrementam a segurança no próprio dado, diferentemente das redes convencionais que as colocam principalmente nos fins. Ao desacoplar os nós de seus endereços, faz com que o ICN também ofereça uma mobilidade intrínseca a qual é ideal para cenários operacionais *ad hoc*, e ainda devido ao *caching* e a replicação de dados ao longo da rede, ele também move os dados em direção ao consumidor, impulsionando a distribuição de informação.

Essas novas características para segurança, mobilidade, e armazenamento interno na própria rede, fazem do ICN uma abordagem que faz com que a implementação de IoT seja possível, além de possibilitar *edge networking* (BACCELLI et al., 2014). É possível compartilhar o nome espaço de nome na rede e aplicações, focando nos dados associados com coisas, e não nos dispositivos, assim como utilizar como auxílio otimizações de caminho e *caching* para atender requisitos IoT em termos de energia e largura de banda.

Consequentemente, redes ICN atendem a uma boa abordagem para aplicações de missões críticas, onde mobilidade, heterogeneidade e características *ad hoc* devem ser levadas em consideração. Infelizmente as redes ICN não conseguem substituir a infraestrutura IP, e a sua usabilidade mais provável é através de ilhas ICN, que são nichos ICN interconectados por uma rede IP (SIRACUSANO et al., 2018) utilizando uma tecnologia como o SDN.

O SDN (NUNES et al., 2014; WICKBOLDT et al., 2015) por sua vez, foi inicialmente planejado como um novo paradigma aplicado à redes cabeadas para separar a lógica de rede em planos distintos: aplicação, controle, encaminhamento e gerência. Mais tarde, o paradigma SDN introduziu uma entidade central de gerenciamento, chamada controlador SDN, para programar o plano de encaminhamento de acordo com as requisições do plano de aplicação e plano de gerência, da mesma forma que a arquitetura proposta.

Assim sendo, a separação dos planos de controle e dados é mais frequentemente utilizado como uma curta definição do que o SDN realmente é. Os *switches* dependem de instruções recebidas do controlador (logicamente centralizado) para encaminhar tráfegos e fluxos. O OpenFlow é o protocolo mais conhecido e utilizado para executar a comunicação entre os *switches* e o controlador, e consegue suportar uma grande diversidade de protocolos de rede. Infelizmente, o protocolo OpenFlow ainda não dá suporte para redes ICN, dificultando o processo de *switches* presentes em uma rede definida por software reconhecer pacotes com objetos nomeados, uma vez que eles não entendem a sintaxe nem a semântica dos datagramas ICN.

O trabalho (ZURANIEWSKI et al., 2017) criou uma extensão para OpenFlow para que seja possível interpretar pacotes ICN. Os autores projetaram um *framework* que faz o OpenFlow trabalhar em conjunto com o Extended Berkeley Filter (eBPF), tornando possível o encaminhamento de pacotes através dos nomes diretamente nos *switches*, uma vez que os mesmos através do eBPF conseguem reconhecer nomes nos pacotes e efetuar *matches*, sem a necessidade de ter que passar pelo controlador toda vez, assim como um pacote convencional orientado a endereços. O *framework* possibilita o desenvolvimento de programas eBPF parametrizáveis e flexíveis que são capazes de fazerem um *match* com qualquer parte arbitrária de um datagrama, até mesmo de um tão complexo quanto o ICN.

Este trabalho utiliza o *framework* proposto em (ZURANIEWSKI et al., 2017) para desenvolver programas eBPF que conseguem reconhecer os pacotes ICNs que utilizam as diretivas C2, as quais são reconhecidas pelo OpenFlow, estabelecendo comunicação entre diferentes ilhas ICN e entre ilhas e redes IP, e vice-versa.

Outro trabalho (SIRACUSANO et al., 2018) desenvolveu um *framework* diferente para combinar ICN e SDN, também permitindo a introdução de ICN sem a necessidade do desenvolvimento de um novo hardware que suporte esta abordagem. Os autores desenvolveram um ambiente para desenvolver e testar ICN sobre soluções SDN, assim como definir políticas de *caching*. Na abordagem deles, os dispositivos de rede executam as ações de encaminhamento de pacotes seguindo regras instaladas por um controlador SDN, que é bem semelhante à proposta utilizada em (ZURANIEWSKI et al., 2017).

Eles utilizaram regras SDN para definir políticas de *caching*, assim como uma nova lógica de aplicação. No presente trabalho, por outro lado, usam as regras SDN para filtrar datas presentes dentro da rede baseadas nos requisitos da aplicação e das condições da rede. O SDN também foi utilizado para criptografar prefixos ICN, adicionando um adicional para a já existente camada de segurança ICN. O trabalho relacionado (GONZALEZ et al., 2016) utilizou os recursos do SDN para coletar informações dos dispositivos de rede e sua programabilidade para permitir a integração de ferramentas de segurança que podem ser usadas em cenários distribuídos como por exemplo IoT.

Outra maneira de utilizar o SDN como um meio de permitir o uso da abordagem ICN foi apresentada em (LV et al., 2017), onde os autores propuseram um mecanismo de roteamento para ICN incorporando SDN e *Community division* (RISC), desacoplando o plano de controle do plano de dados e dividindo a topologia ICN em diferentes "comunidades". Os autores propuseram um esquema de divisão de comunidades baseados em uma árvore máxima, para auxiliar a entrega de dados e em seguida colocar toda a informação sobre conteúdos e encaminhamentos no controlador. Os autores também projetaram um dispositivo de encaminhamento para “intra-comunidade” (baseado em informações internas) e para “inter-comunidades” baseada na relação entre comunidades.

Neste trabalho, o SDN é utilizado para ajustar a lógica de encaminhamento que o ICN utiliza para melhorar o uso de parâmetros de rede e para permitir um melhor desempenho e maior disponibilidade de dados para nós de alta prioridade. Um outro trabalho (ZHANG; ZHU, 2016) utilizou uma combinação diferente de SDN e ICN para melhorar a Qualidade de Serviço (QoS) para redes sem fio definidas por software. Os autores utilizaram SDN para maximizar a utilização da rede, desacoplando a infraestrutura da rede sem fio do serviço.

Fazendo isto, o SDN permite serviços diferenciados para compartilhar a mesma infraestrutura. Entretanto, isso compromete a transmissão de dados multimídia sensíveis temporalmente devido à latência de tais ambientes. Os autores propuseram uma técnica

de virtualização de redes sem fio centradas à informação onde os usuários de dispositivos móveis requisitam a origem dos dados para o controlador da rede. O controlador encontra esta origem (localizado em um fim ou em um dispositivo com *caching*) e configura o caminho de entrega destes dados para o usuário. Fazendo isto, a arquitetura baseada em SDN constrói dinamicamente um caminho ótimo.

O controlador se mantém ciente dos requisitos de QoS dos usuários e também as atuais condições de rede (e.g., sinais ruidosos), construindo diferentes caminhos ótimos ao mesmo tempo. A ideia é maximizar a capacidade efetiva para um dado requisito de QoS estabelecendo um caminho ótimo entre produtos de dados e consumidores para determinada condição de rede. Neste trabalho, nós pretendemos alcançar um objetivo similar utilizando o módulo de filtro no controlador SDN, definindo caminho de acordo com as condições da rede e dos requisitos de aplicação.

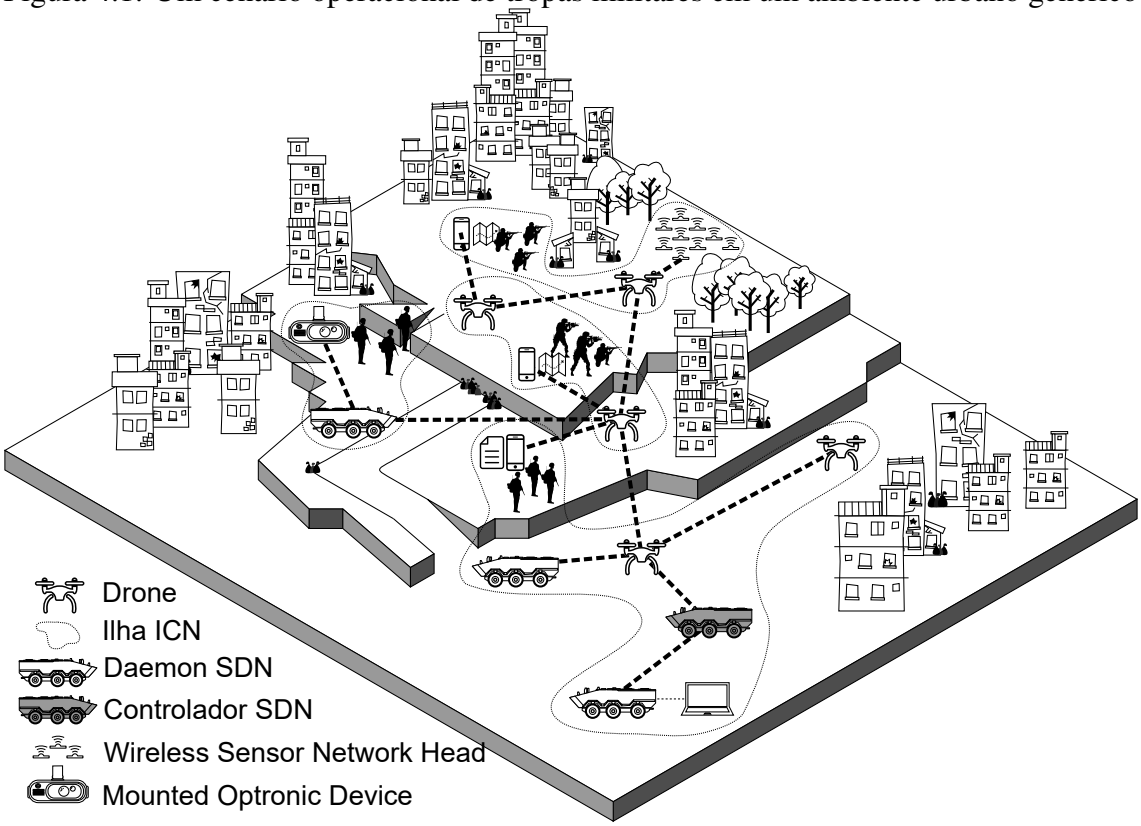
A arquitetura apresentada neste trabalho se diferencia dos trabalhos apresentados anteriormente porque ela apresenta uma aplicação em um específico caso de uso do uso combinado de ICN e SDN. Ele consiste de uma tríade aplicação/SDN/ICN, onde os três componentes agem de forma integrada para atender requisitos de alto nível de usuários para distribuição de informação, padrões de interação e alocação de direitos de decisão, e ao mesmo tempo, mantém a saúde da rede, utilizando um esquema de filtragem de dados.

# 4 CENÁRIO DE APLICAÇÃO

A realidade dos campos de batalha modernos, envolvendo inúmeros dispositivos inteligentes conectados por uma rede, provê meios otimizados para melhorar atividades C2. Essas otimizações atendem as necessidades dos cenários operacionais atuais cujos demandam meios flexíveis e eficientes de orquestrar operações militares, como as citadas anteriormente na seção 2. Um exemplo recente de um cenário operacional é o cenário enfrentado pelo exército no Brasil, nas operações de Garantia de Lei e Ordem. Devido ao caos imposto pelo crime organizado em regiões específicas das grandes metrópoles, como por exemplo o Rio de Janeiro, a aplicação da legislação padrão não é capaz de lidar com o desafio. Portanto, o contexto militar enfrenta um novo cenário operacional urbano, no qual os adversários estão espalhados e escondidos em um ambiente dinâmico de difícil acesso, em que podem facilmente se misturar com civis. A Figura 4.1 mostra um exemplo deste cenário em que as tropas militares ocupam uma região dominada por gangues. Nesta figura é possível observar a presença de diferentes elementos conectados por uma rede. Os veículos militares podem acessar apenas partes limitadas do ambiente devido ao seu tamanho e dificuldade de locomoção. A partir de um determinado ponto, apenas tropas a pé são capazes de acessar e observar os locais críticos. Devido à dificuldade de um progresso seguro em tal território, drones podem dar suporte às tropas oferecendo imagens aéreas procurando por movimentos suspeitos em terra. Esses lugares geralmente possuem uma divisória com florestas urbanas, fazendo com que em muitas situações representem rotas de fuga perfeitas para escape dos indivíduos fora da lei. Na Figura 4.1, uma rede sem fio de sensores (*Wireless Sensor Network* ou *WSN*) é representada monitorando esta rota de evasão.

Ao utilizar os diferentes recursos oferecidos pelos múltiplos dispositivos em campo, o comandante da operação pode rastrear e acompanhar as ações executadas, solicitando vídeos dos drones e dos dispositivos de suporte à visão, carregados pelas tropas (i.e. *head mounted optronic devices* or *HMOD*), por exemplo. Desta posição de observação, localizada em um dos veículos auxiliando a operação, como representado na Figura 4.1, o comandante pode fazer decisões baseadas nas informações obtidas de (re)alocações das tropas, almejando sua melhor performance, enviando ordens diferentes para as próximas ações. Entretanto, devido à grande dinamicidade deste cenário, as tropas de linha de frente podem não esperar por decisões para seus próximos passos, uma vez que estão em uma posição crítica. Portanto, um certo nível de autonomia para decisões locais é necessário,

Figura 4.1: Um cenário operacional de tropas militares em um ambiente urbano genérico




que por sua vez necessita de informações acerca do campo mesmo sem ordens para auxiliar suas missões. O quão decentralizado esta delegação de autonomia será dependente da abordagem C2 escolhida.

Da perspectiva de auxílio às redes, é importante efetuar rápidas escolhas de rotas para que elas possam implementar corretamente a recepção/entrega relacionados a outros nós específicos, em outras palavras, é necessária uma rápida decisão por alguma entidade (que como explicado mais a frente, se trata do controlador), e fazer uma decisão de rota que otimize o tempo de comunicação entre os nós, necessitando de uma latência otimizada. Além disso, a rede deve também deve prover dados com informações críticas de forma confiável para os nós autorizados que possam precisar dos mesmos, em um intervalo temporal eficiente, aumentando a disponibilidade dos nós mesmo em cenários propensos à falhas frequentes, ou seja, uma alta tolerância à falhas de canais. No cenário estudado, as tropas a pé podem receber imagens de atividades suspeitas ao seu redor, mas não podem receber imagens de locais muito longe de suas respectivas posições. Por outro lado, o comandante receberá dados de todos os meios de comunicação, uma vez que ele precisa ter conhecimento de todo o campo de batalha para arranjar os recursos focando em suas missões. Entretanto, devido à dinamicidade destas operações, a topologia da

rede pode mudar com bastante rapidez, e é possível que os nós fiquem sobrecarregados. Estes nós podem decidir independentemente qual dado encaminhar, e também qual dado descartar ou armazenar localmente para encaminhamentos posteriores. Na situação crítica de contato iminente com grupos inimigos hostis, as tropas a pé são mais importantes de receber imagens de ameaças próximas do que o comandante. Neste e em outros casos particulares, a rede pode agir desta forma: primeiro envia os dados para as tropas, e armazenando os mesmos em seu cache para entrega posterior ao comandante.

No contexto de IoBT, as "coisas" podem também precisar de informações para auxiliar suas atividades. Neste cenário operacional, por exemplo, o comandante ou um soldado pode enviar requisições de *INTEREST* para obter imagens de possível evasão de suspeitos. O produtor de tais informações de forma mais imediata são os drones, porém em uma rede inteligente, eles não trabalham sozinhos. Como dispositivos inteligentes, os drones tentariam identificar os locais mais apropriados para sobrevoar e capturar tais imagens. E assim, os drones enviariam requisições de *INTERESTs* por detecção de movimento para sensores de movimento. Assim que um movimento é identificado, este sensor responde com uma mensagem *DATA* que atende o segundo *INTEREST*, vindo do drone, com uma localização aproximada de movimento suspeito. O drone mais próximo desta localidade então deve fazer uma investigação na área, adquirir imagens, e responder com mensagens *DATA* para os humanos, que atendem o primeiro *INTEREST*.

Neste exemplo, para executar uma ordem enviada por um humano, os drones por si mesmo são capazes de requisitar dados de outros nós, neste caso dos sensores. Fazendo isto em nível de rede, a rede ICN age como uma ferramente capaz de permitir esta implementação de interações complexas e padrões de comunicação entre humanos e coisas. Essas interações implicarão em um controle de ciclo C2 muito melhor e mais ágil, onde a etapa de *sensemaking* (detecção de movimento e obtenção de imagens), etapa de *execution* (voar para a zona correta) e *situation monitoring* (identificar evasão de adversários) são atendidas sem a necessidade de especificar um nó ou endereço específico, uma vez que as informações em si, de forma distribuída ou não, completam este ciclo.

Uma característica interessante é que diferentes tipos de fluxos de dados circulam pela rede, desde pequenas mensagens de textos de poucos *bytes*, como ordens de superiores e mensagens de alarme dos sensores, até mensagens de alto custo de armazenamento, como vídeos providos pelos drones ou de dispositivos de visão utilizados pelas tropas. Diferentemente das tradicionais redes IP, nesta arquitetura é possível tirar proveito do *caching* dentro da própria rede, disponibilizado pelas redes ICN, e conseguir mensagens de

*DATA* de nós vizinhos ou mais próximos, reduzindo o tráfego de transmissão de dados na rede, uma vez que o armazenamento dos dados mais relevantes e críticos temporalmente tendem a estar próximos do consumidor. Através da replicação de dados em múltiplos nós, a rede por si só é capaz de contornar problemas como conexões intermitentes e falhas em dispositivos. Estas características tornam a distribuição de informação, que é outro aspecto chave do C2, muito mais robusto e ágil com ICN.

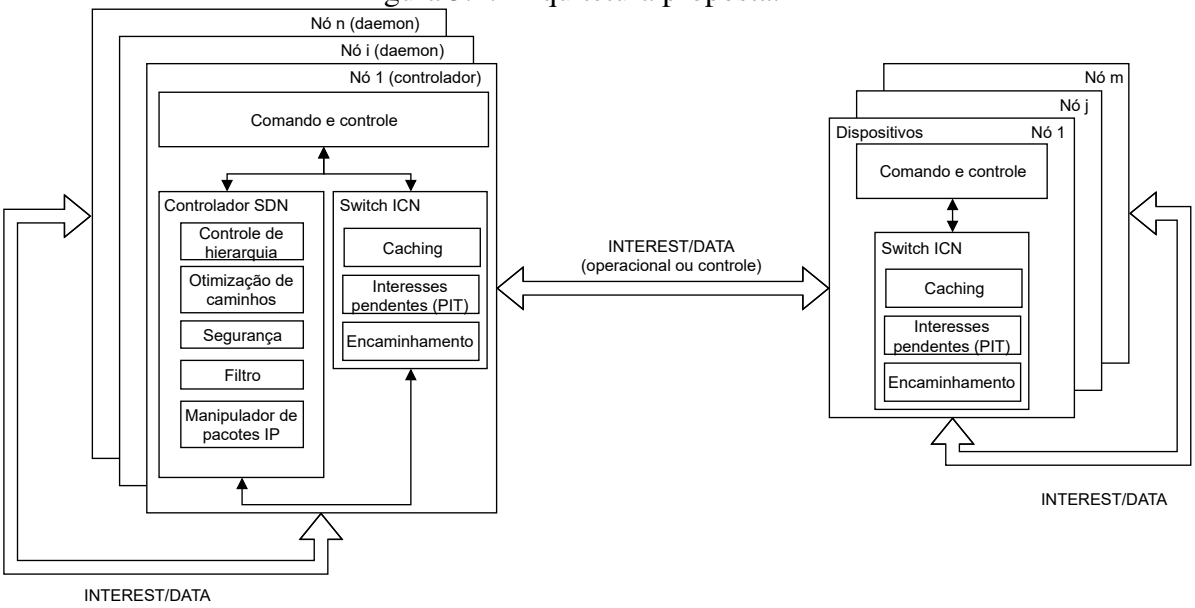
Ao adicionar estas características para as duas dimensões apresentadas, distribuição de informação e padrões de interação, que através de redes IP convencionais seriam muito mais dificilmente e mais custosas de serem atendidas, ICN se encaixa bem para ser testada em cenários operacionais de melhora da Agilidade C2 em nível de rede. Um dos maiores problemas com as redes ICN é que ela atualmente não é capaz de substituir redes IP, entretanto é possível utilizar ilhas ICN interconectadas em uma grande rede infraestrutura IP, como observado em (ZURANIEWSKI et al., 2017). Portanto, como citado anteriormente, uma rede SDN é uma possível candidata para integrar múltiplas ilhas ICN com o IoBT.

Com esta abordagem, dentro das ilhas, as características ICN controlaria o plano de dados para uma distribuição de informação eficiente, e interações entre humanos e coisas, e as redes SDN são responsáveis por duas tarefas: a primeira é por integrar diferentes redes ICN dentro de uma infraestrutura IP, agindo como um tunelamento; a segunda tarefa é controlar o plano de dados ajustando os parâmetros baseados nas propriedades dos fluxos de dados (largura de banda, tamanho dos dados, controles de hierarquias, controle de prioridades, otimização de caminhos, e etc) usado pelo ICN para encaminhar dados e encontrar caminhos. Por exemplo, uma mensagem *DATA* com um tamanho específico pode ser encaminhado apelas para nós com uma largura de banda mínima. Estes parâmetros podem ser modificador pelo administrador da rede durante as operações utilizando SDN para amplificar as entregas da rede.

Ao controlar qual nó pode enviar tipos diferentes de mensagens de *INTEREST*, a rede SDN pode agir em uma terceira dimensão do espaço C2, a alocação de decisões. Por exemplo, o controlador SDN pode estabelecer quem é permitido para modificar parâmetros de voos dos drones (e.g., o comandante, qualquer soldado, apenas soldados perto dos drones, outros drones). Outro aspecto é que esta alocação pode ser modificada pelo administrador da rede, durante a execução da operação, em resposta de mudanças das missões ou de circunstâncias assim provendo meios técnicos de implementar agilidade C2 em uma rede.

# 5 ARQUITETURA PROPOSTA SDN-ICN

Figura 5.1: Arquitetura proposta.




## 5.1 Visão Geral da Arquitetura

Uma rede militar tática pode ser composta pela combinação dos seguintes dispositivos: um sensor sem fio, que pode ser utilizado para detecção de movimento e monitoração de área; veículos armados possuindo *switches* para encaminhar e obter dados; tropas a pé, equipados com equipamentos vestíveis e dispositivos embarcados transportáveis; e drones com o propósito de vigilância geral. Estes elementos são nós primários a ser gerenciados para otimizar um ambiente IoBT.

Este ambiente é extremamente dinâmico devido à frequente mudança geográfica dos nós e da constante entrada e saída de nós da rede. Adicionalmente, a rede IoBT deve ser capaz de lidar com a heterogeneidade destes nós, onde eles são capazes de autenticarem uns ao outros, comunicarem entre si, devem possuir capacidade de auto-adaptação e produzir informações de alta qualidade para auxiliar o sucesso da missão, mantendo alta confiabilidade e disponibilidade.

Como dito na seção anterior, ICN cobre bem estes requisitos devido aos mecanismos de *caching* e da abordagem centrada a dados, onde objetos nomeados não possuem restrições de hardware ou de software para serem disseminados pelos *switches* ICN. Ao

utilizar objetos nomeados, diferentes nós podem se comunicar entre si independentemente de suas características.

O serviço de nomes provido pelas redes ICN torna mais fácil receber dados de nós dinâmicos, uma vez que o consumidor não precisa saber quem é o produtor, mas apenas o dado disseminado pela rede. Portanto, a integração entre canais de terceiros para complementar com informações auxiliar se torna muito mais fácil. Este aspecto é extremamente importante para a visão operacional, porque isso possibilita o aumento da disponibilidade dos nós que já estão no campo (por exemplo, nós servindo aplicações em cidades inteligentes).

A disponibilidade também é aumentada através do *caching* e replicação, mantendo informações até mesmo se o produtor destes dados se tornar inalcançável, adicionando duas melhorias: a) os dados são armazenados em múltiplos nós, o que pode reduzir a latência, uma vez que o dado requisitado pode estar presente em um nó mais próximo, e também se torna mais confiável em caso de falhas de nós, uma vez que qualquer nó replicado pode servir o consumidor; b) uma conexão direta entre produtor e consumidor não é necessária, desacoplando espaço e tempo das requisições.

Por outro lado, em um campo de batalha também é necessário tratar uma quantidade massiva de dados (KOTT; SWAMI; WEST, 2016) e também evitar qualquer encaminhamento desnecessário de dados, minimizando a sobrecarga dos canais de transmissão. Da mesma forma, alguns mecanismos que definem padrões de interação através dos nós da rede podem ser bastante informativos, como a prioridade de um nó para receber determinada informação. Por exemplo, no caso de um congestionamento de rede, o veículo do comandante pode receber maior prioridade de receber determinada informação do que um veículo secundário.

Neste sentido, o processo de filtrar dados desnecessários é importante, tanto no ponto de vista operacional, para evitar que nós que tomam decisões importantes sejam sobrecarregados com informação, assim como para manter a rede estável e funcionando sem gargalos. O processo de filtragem de informações pode ser executado através de técnicas de aprendizagem de máquina e de padrões semânticos (KOTT; SWAMI; WEST, 2016), ou analisando a capacidade dos nós em termos de armazenamento de dados e de largura de banda.

Por exemplo, dispositivos vestíveis normalmente possuem capacidade de armazenamento e de largura de banda limitados, o que torna difícil para eles receber e efetuar *caching* de dados extensos. A abordagem SDN é apropriada para resolver este problema,

sendo capaz de usar condições para encaminhamento de dados. Se uma mensagem *DATA* passa por um caminho em que um nó com limitações muito restritas de armazenamento e de transmissão está presente, este pacote deve ser redirecionado por outro caminho que embora possa ser mais longo, ainda assim será mais benéfico para a rede em geral.

A rede SDN então tem um papel crítico no plano de controle, estabelecendo hierarquia de nós, segurança, otimização de caminhos, e também filtragem de dados de acordo com parâmetros operacionais ou status da rede, enquanto as vantagens da rede ICN se dá no plano de dados. A arquitetura baseada nos princípios descritos anteriormente é agora apresentada.

## 5.2 Elementos da Arquitetura

A arquitetura proposta coloca a abordagem ICN responsável por disseminar informação através da rede e a SDN para conectar as ilhas ICN, e para controlar padrões de interação entre os nós. Cada ilha ICN é representada por um agregado de veículos, tropas a pé, drones e sensores, como mostrado na Figura 4.1.

A arquitetura é composta pela aplicação C2, o controlador SDN e os *switches* ICN. Estes módulos em conjunto definem o comportamento dos dispositivos que são os nós da rede, e suas interações são apresentadas na Figura 5.1.

### 5.2.1 Camada de Aplicação

A camada de aplicação, representada pelo componente comando e controle na arquitetura, é o componente de mais alto nível da arquitetura, onde efetivamente se define os objetivos, tipos de dados e meios da comunicação da rede. Ela é responsável por produzir e requisitar dados, assim como definir os parâmetros para a execução de qualquer tarefa, como permissões, tipos de nós, e aspectos específicos da aplicação desejada. Qualquer aplicação onde se deseja reutilização de dados, eficiência otimizada, maior confiabilidade e maior flexibilidade de parametrização da comunicação pode ser implementada aqui.

Uma das possibilidades de aplicação, a aplicação C2, é responsável por apresentar informações relevantes e atualizadas para os usuários, e capturar as intenções e decisões dos usuários depois de cada ciclo do processo C2. Diferentes aplicações C2 podem rodar em diferentes tipos de nós, de acordo com a capacidade do nó e dos requisitos opera-

cionais. O mais robusto e completo roda nos veículos, uma vez que possui uma maior capacidade e é controlada pelo comandante da missão.

Tropas a pé possuem versões simplificadas em seus dispositivos embarcados, que disponibilizam um mapa com eventos georreferenciados além de imagens e vídeos, e também torna possível que o usuário receba ou envie relatórios através de áudios e textos. Os drones rodam aplicações especializadas em capturar imagens e vídeos que efetuam pré-processamentos e reconhecimentos de padrões.

A aplicação C2 no veículo do comandante define a configuração inicial da missão para os dispositivos. Neste contexto algumas definições devem ser feitas para auxiliar o entendimento de objetivos:

- Missão: é uma atividade planejada para alcançar um objetivo operacional. Ela estabelece os recursos a serem usados (veículos, tropas e drones), as Áreas de Interesse (*Areas of Interest* ou *AoI*), assim como a hierarquia dos nós e as prioridades das AoI. É possível que exista mais de uma missão em execução simultaneamente. Por exemplo, uma ilha ICN pode estar rodando a missão N, enquanto outra ilha ICN pode estar rodando a missão M.
- Tarefa: é uma atividade *ad hoc*, definida pela aplicação C2 que está no andamento de uma missão, para coletar informações ou para efetuar ações através dos dispositivos em ação, em resposta ao processo de decisão provido pelo ciclo C2. A tarefa é especificamente guiada à uma AoI (que está presente no conjunto de todas as AoI de uma missão), designando um nó para executar esta tarefa e também definindo sua prioridade.

A aplicação C2 é o módulo que compreende a situação, se mantendo ciente de todos os dados operacionais. Esta aplicação tem acesso à interface do controlador SDN para realizar mudanças na rede e obter informações estatísticas e do estado da rede através do controlador, enquanto os *switches* ICN são responsáveis por transmitir e receber dados.

Este módulo se comunica com o controlador SDN e com o switch ICN através de mensagens especiais de controle, onde é possível definir os parâmetros de execução do controlador, e também definir como será efetuado o encaminhamento através dos switches, que será melhor detalhado na seção de implementação.

### 5.2.2 Controlador SDN

O controlador SDN é o módulo responsável pelo controle de funções do ciclo C2. Ele também é responsável por conectar múltiplas ilhas ICN, assim como traduzir conexões IP, uma vez que as redes existentes em sua maioria são baseadas em IP, assim toda a comunicação com a *cloud*, que é representada por redes externas (como no exemplo de aplicações em cidades inteligentes descritas anteriormente) é toda traduzida pelo controlador. É possível tratar estas diferentes conexões através da modularização de tratamentos, onde cada módulo trata um tipo diferente isoladamente como visto em (ZURANIEWSKI et al., 2017).

O controlador é composto por três componentes principais:

- Controle de hierarquia: ele define a hierarquia dos nós e prioridade das mensagens baseados nos parâmetros recebidos pela aplicação C2, seguindo a lógica de missão e tarefa. O controlador pode descartar requisições não autorizadas como uma medida de segurança para um melhor controle geral da rede. Esta característica também pode ser utilizada para lidar com casos em que o IoBT cresce abruptamente em quantidade de nós. O controle de hierarquia pode ser implementado através dos nomes dos prefixos dos objetos nomeados, onde o nome dos dados definem de que tipo de nó vem ou vai o dado, e portanto, se esse pode ser encaminhado.
- Segurança: autentica nós de rede válidos e criptografa/descriptografa os nomes ICN para evitar possível ataques intermediários durante as operações.
- Otimização de caminho: Modifica a lógica de encaminhamento baseado na visão global de rede que as redes SDN possuem, definindo regras para encaminhar através de canais que possuem maior disponibilidade de banda, ou pelo menor número de saltos de nós, otimizando a latência e uso de banda.
- Filtro: Através da análise do controle de hierarquia, disponibilidade de cache nos caminhos escolhidos e pacotes que não atendem políticas de segurança, como a criptografia, são descartados pelo filtro, que é um intermediário entre os pacotes que entram e saem do controlador para ser repassado para os switches.

Esses componentes tratam parâmetros operacionais das aplicações C2 (hierarquia, prioridade e segurança). O controlador através da análise dos fluxos das redes que percorrem através das redes ICN e também das redes IP também é capaz de armazenar estatísticas para melhor entender as condições e requisitos atuais da rede.

O controlador SDN também contém gerenciadores de pacotes IP/ICN para tratar e encaminhar apropriadamente todos os tipos de pacotes que passam por ele.

Além disso, como o controlador é um ponto crítico na rede, existem daemons para caso o controlador principal se torne indisponível, um daemon tome seu lugar através de um algoritmo de eleição. Este trabalho não teve como foco o estudo do algoritmo mais eficiente de eleição, e foi utilizado um algoritmo simples de eleger o primeiro daemon que detectar que o controlador se tornou indisponível através de mensagens de *heartbeat* para verificar os status atual da rede.

É também presente na rede um canal de controle com o SDN, que é o canal de comunicação efetuado entre os dispositivos e o controlador. Esses canais são os mesmos para a transmissão de dados de *INTEREST* e *DATA* em geral, e esse caminho até o controlador deve sempre estar presente em todos os dispositivos. Isso é mantido por um *heartbeat* contínuo para saber como chegar ao controlador. Caso o caminho do controlador seja perdido, então é efetuado uma disseminação *multicast* de *INTEREST* para se ter contato do novo controlador.

### 5.2.3 ICN Switches

Os *switches* ICN são responsáveis pela disseminação dos dados. A lógica de encaminhamento é definida de acordo com mensagens de *INTEREST* de controle, definidas pela aplicação C2 e montadas no controlador SDN. A tabela de interesses pendentes (*Pending Interest Table* or *PIT*) e o *caching* seguem as funcionalidades convencionais do ICN, mas a capacidade de armazenamento varia de acordo com o tipo do dispositivo (e.g., veículos tem capacidade de *cache* muito superior comparado a um sensor).

O papel fundamental dos switches é aplicar regras de encaminhamentos providas pelo controlador. Como a rede apresentada neste trabalho não é uma rede convencional IP, os switches devem encaminhar fluxos baseados nos nomes, e não nos endereços físicos tais como endereços IP e MAC, ou seja, o *match* deve ocorrer em cima dos nomes, para assim aplicar *actions* providas pelo controlador. O switch ICN detecta o nome e caso este nome seja desconhecido em sua tabela de encaminhamento, há uma requisição de PACKET IN para o controlador, onde o mesmo faz o processamento em cima da topologia atual da rede e passa para o switch ICN como, e para onde este pacote deve ser encaminhado.

### 5.2.4 Dispositivos

Todos os dispositivos na rede possuem um comportamento genérico que pode ser sumarizado em um diagrama de sequência apresentado na Figura 5.2.

A configuração inicial da missão é definida pela aplicação C2 no veículo do comandante, que associa dispositivos à missão e à AoI, assim como hierarquias e prioridades. Um nó genérico pode carregar sua configuração de missão em sua inicialização, ou ter ela definida ou atualizada ao requisitar um *INTEREST* ao comandante/controlador informando que está pronto e disponível para receber uma missão. O comandante então envia uma mensagem *DATA* com os parâmetros de missão para este nó. Estes parâmetros serão usados pela aplicação C2 para executar a missão genérica. Uma missão possui um contexto fechado, como por exemplo, um drone ficar rondando uma determinada área a procura de qualquer imagem suspeita. O nó permanecerá em estado de andamento de missão e fazendo um papel genérico até que chegue uma mensagem *INTEREST* com uma tarefa específica, no qual o dispositivo executará uma tarefa dentro do contexto da missão dada.

O comportamento descrito acima pode ser observado no diagrama da Figura 5.2, que é representado pelo **Ciclo de missão**, onde o dispositivo produz dados continuamente enquanto a missão está em andamento, até que esta seja abortada. A tarefa interrompe a execução genérica de missão para algo específico, porém possui um contexto determinístico onde quando a tarefa acaba (ou é abortada) o dispositivo volta a operar a missão atual. A missão só acaba quando abortada.

Tarefas possuem prioridades como parâmetros, e uma tarefa pode sobrepor outra caso sua prioridade seja maior, onde as tarefas de menor prioridade ficarão em uma fila de espera ordenada por prioridades.

## 5.3 Definição das Mensagens

As mensagens de *INTEREST* e *DATA* são definidas como na seguinte tabela apresentada em Tabela 5.2 e Tabela 5.1. Um grupo de mensagens é utilizado para controle da rede, enquanto outro grupo é utilizado para a execução de tarefas operacionais e transmissão de dados. As mensagens são textos criptografados em cima das mensagens apresentadas na tabela, com um parser, onde a forma de criptografia e descriptografia é definida pela aplicação C2 do veículo, e sempre que um novo nó entra na rede, é passado o método

de criptografia e descriptografia para estes nós, assim nós que tentem obter informações a respeito do pacote não conseguirão ler o conteúdo, nem o nome do *INTEREST* ou *DATA*.

Tabela 5.1: INTERESTs de Controle

| **INTEREST** | **#1** | **#2** |
|---|---|---|
| **Nome** | /control/setup | /control/status |
| **Descrição** | Qualquer dispositivo envia um interesse de configuração para o controlador para inicializar uma missão | Periodicamente o controlador envia o interesse de status para os dispositivos para constantemente atualizar os status da rede em geral |

Tabela 5.2: INTERESTs Operacionais

| **INTEREST** | **#3** | **#4** | **#5** |
|---|---|---|---|
| **Nome** | /missao/X/tarefa/Y | /missao/X/tarefa/Y/ AoI/Z/ detectarmovimento | /missao/X/tarefa/Y/ AoI/Z/ detectarmovimento |
| **Descrição** | Inicializa uma tarefa em qualquer um dos dispositivos | Um humano solicita por movimentação de drones ou de sensores na área Z | Um drone solicita por movimentação vinda de sensores na área Z |

| **INTEREST** | **#6** | **#7** |
|---|---|---|
| **Nome** | /missao/X/tarefa/Y/ AoI/Z/avancar/Q | /missao/X/[tarefa/Y/] abortar |
| **Descrição** | Solicita para que uma quantidade Q de humanos avancem em determinada área de interesse Z | Solicita para que a missão ou a tarefa seja abortada (campo tarefa opcional) |

## 5.4 Ciclo de vida da rede

A entidade principal no plano de controle é o controlador SDN, presente no veículo do comandante. Como visto na Figura 5.1, existem *daemons* SDN sendo executados em todos os outros veículos. No caso de um veículo de comandante se tornar não operável, um dos *daemons* é então eleito como o novo controlador, uma vez que deve existir um controlador disponível em todo momento para a rede operar com normalidade.

O controlador age baseado nas informações providas pela aplicação C2 e pelo ICN *switch* presente no veículo do comandante. Depois que o comandante completa o planejamento de missão, a aplicação C2 envia os parâmetros das missões para o controlador.

Como visto na Figura 5.2, o controlador gera regras de filtragem com parâmetros de missão e prepara *DATA* #1 para ser entregue para todos os dispositivos que requisitaram uma missão.

O estado da rede é processado utilizando *INTEREST* e *DATA* #2, ilustrado por **INTEREST e DATA (estado atual)** na Figura 5.2. Todos os nós informam seus estados atuais (OK ou falha) e seus parâmetros (bateria disponível, armazenamento, largura de banda, etc.) e o controlador SDN utiliza essas informações para adaptações de encaminhamentos seguindo as prioridades da rede, além de atualizar a topologia da rede e definir novas regras de filtragem de acordo com as mudanças da rede. Esse processo é o **Ciclo de comando e controle** também observado na Figura 5.2, que ilustra a contínua atualização da rede.

Os nós podem estar em um estado de falha. Os estados de falha se classificam em dois tipos: falhas transiente, em que após uma manutenção é possível reestabelecer o nó e o mesmo voltar a operar normalmente, e falha fatal, onde o nó não pode mais operar. A gravidade da falha é obtida a partir dos últimos dados enviados dos nós para o controlador através das mensagens de estado atual. Se um nó pára de responder, um drone é enviado para verificar a viabilidade de recuperar o nó.

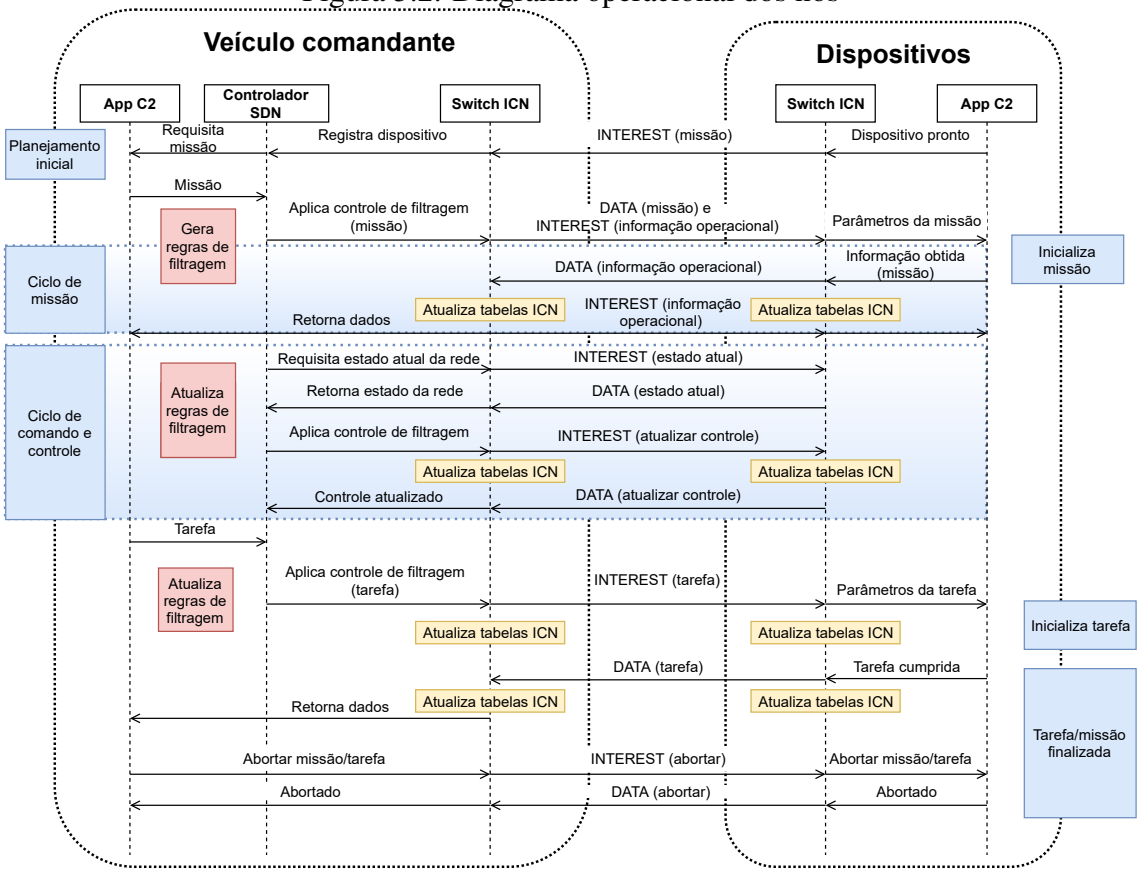
Um ponto importante a se destacar é que embora haja uma intensa comunicação entre todos os nós e o controlador através de INTERESTs de controle, esses interesses não sobrecarregam a rede, uma vez que pacotes de controle são extremamente simples (poucos bytes), pois apenas possuem informações a respeito de parâmetros, definições de prioridades e da missão, que são conteúdos extremamente pequenos, e estes pacotes trafegam apenas em suas respectivas ilhas ICN, que são compostas por uma quantidade não muito grande de nós (normalmente algumas centenas de nós), principalmente no caso de uso militar apresentado.

## 5.5 Detalhes de Implementação

Para a validação da arquitetura previamente apresentada, um protótipo capaz de realizar as funcionalidades descritas foi implementado. Atualmente existem poucos trabalhos para simular uma rede ICN gerida por um controlador capaz de produzir resultados para análise, portanto foi necessário a junção de algumas ferramentas para se tornar possível gerar uma rede SDN e ICN completa de forma acoplada.

Para a simulação de uma rede SDN foi utilizado o simulador mininet (LANTZ;

Figura 5.2: Diagrama operacional dos nós

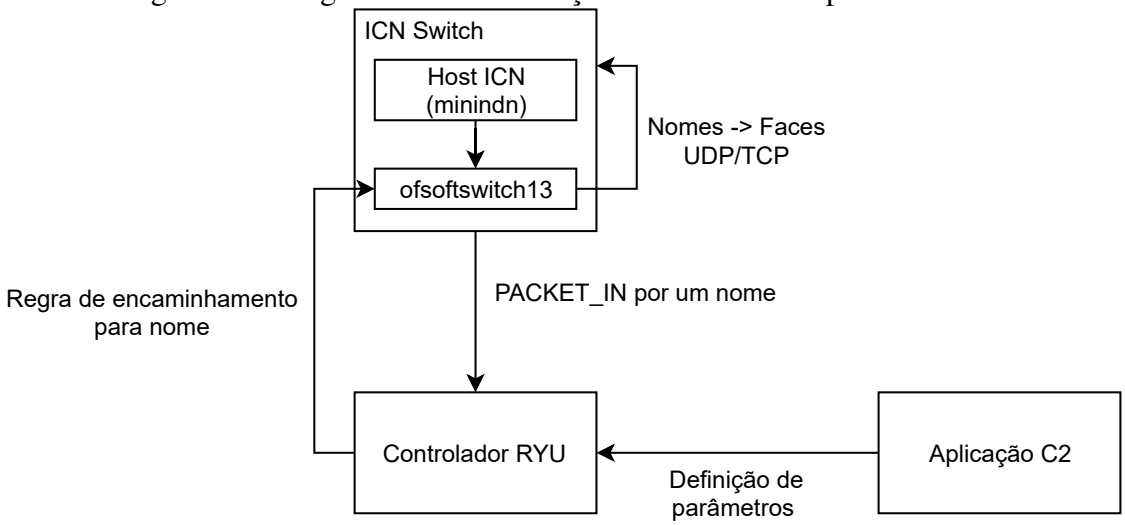



HELLER; MCKEOWN, 2010) para gerar os *assets* que respondem um controlador, que trabalhou juntamente com o controlador Ryu (KHONDOKER et al., 2014), que controla a topologia da rede, encaminhamento e otimização do tráfego da rede, filtra fluxos, controla a hierarquia, além da gerência de pacotes ICN e pacotes IP que trafegam vindo da nuvem. Para a simulação de uma rede ICN foi utilizado o simulador minindn (LANE et al., 2019) que é capaz de trabalhar em conjunto com o mininet adicionando algumas propriedades especiais para os *hosts*, os tornando capazes de reconhecer nomes. É possível observar visualmente como os componentes implementados se comunicam através da Figura 5.3. A implementação é descrita em detalhes a seguir.

### 5.5.1 Switches

Os *switches* formam um ponto chave para a integração entre SDN e ICN, porque embora através do minindn seja possível reconhecer nos *hosts* os nomes dos objetos desejados, o *switch* deve ser capaz de encaminhá-los de forma apropriada, também através do nome. Para atender esse requisito, foi utilizado o ofsoftswitch13 (FERNANDES et al.,

Figura 5.3: Diagrama de comunicação dos módulos implementados




2019), que é um software em espaço de usuário no sistema operacional capaz de simular um *switch* com OpenFlow 1.3.

Este *switch* é capaz de receber especificações de como executar um *match* de forma específica quando um pacote passa por ele, e essa especificação vem do controlador. Ainda assim, o *switch* sozinho não é capaz de reconhecer um nome ICN, uma vez que até a versão mais atual do OpenFlow ainda não executa *match* em cima de nomes ICN, portanto, assim como no trabalho (ZURANIEWSKI et al., 2017), foi utilizado o eBPF (extended Berkeley Packet Filter), que é um pré-processador de pacotes capaz de extrair informações de um pacote assim que este chega na interface de rede, sendo capaz de auxiliar no encaminhamento de pacotes específicos como o ICN de forma otimizada.

Utilizando o ofsoftswitch13 em conjunto com o eBPF podemos reconhecer um nome, pois um programa eBPF pode rapidamente identificar o tipo do pacote, suas informações, e encaminhar de acordo com estes metadados. Neste momento, entra o papel do controlador, que é gerar os programas de *match* em cima de nomes, e repassá-los para os *switches*. O primeiro passo para isso é o controlador repassar os programas eBPF para os *switches* executarem na inicialização da rede, e neste programa deve estar especificado como o *switch* se comporta quando um pacote ICN está trafegando através do mesmo. Ele se comunica com um switch ICN como ilustrado na Figura 5.3.

Uma vez que os *switches* possuem os programas que reconhecem nomes instalados, já é possível de o controlador mandar *matches* específicos em cima de nomes, o que possibilita a integração SDN com ICN.

### 5.5.2 Hosts ICN

Como dito, os *hosts* ICN são gerados utilizando a extensão do mininet chamada minindn que provê um comportamento específico para os *hosts*. Embora os *hosts* passem a reconhecer apenas nomes, ainda assim, como é apenas um simulador, é necessário uma forma de comunicação entre dois *hosts* diferentes em cima de uma rede atual, isto é, redes IP.

Para o minindn contornar esta limitação, ele utiliza as chamadas "faces", que são um mascaramento de comunicação para os nomes passarem através dela. As faces podem ser através de IP, isto é, utilizando protocolos como UDP ou TCP, e também podem ser em nível de sistema operacional, como por exemplo utilizando diretamente sockets do Unix.

Neste trabalho foram utilizadas faces UDP, onde cada *host* ICN possui um endereço IP físico e uma porta correspondente para aquele dado específico. A diferença que torna o ICN possível, é que diferentes faces podem responder a uma mesma requisição para um determinado nome de objeto ICN. Além disso, o minindn torna possível o *caching* nos *hosts* ICN, que dão uma das propriedades mais importantes para este trabalho.

Uma vez que os *hosts* configuram suas faces, eles estão prontos pra comunicar com outros *hosts* ICN. Porém, para efetuar esta comunicação, eles ainda assim precisam expor as suas faces para o controlador, para que este consiga encaminhar os dados para as faces corretas sempre que um nome sofre um PACKET IN.

Foi necessário modificar o código fonte do minindn pois o mesmo não suportava o uso de um controlador para operar os nós, uma vez que ele simulava apenas os nós ICN sem nenhum tipo de definição por software da rede. Para tal, foi necessário modificar para que qualquer pacote que não fosse possível encaminhar fizesse uma requisição de PACKET IN para o controlador Ryu implementado.

O canal de controle no mininet foi utilizando um enlace direto entre cada nó e o controlador, porém como dito anteriormente, as mensagens e a frequência das mesmas para uma quantidade de centenas de nós é extremamente baixa, a nível de bytes, fazendo com que o uso deste enlace seja mínimo.

### 5.5.3 Switch ICN

Para se tornar compatível com a arquitetura, um *switch* ICN deve existir para operar as funcionalidades de *caching*, *pending interests*, e *forwarding*. Neste trabalho um Switch ICN é representado por uma dupla composta por Switch e Host ICN. Isto é, um ofsoftswitch que suporta eBPF, integrado com um *host* minindn que suporta faces, é possível observar na Figura 5.3.

O *caching* é feito no *host* minindn, onde é possível predeterminar a capacidade de *caching* para simular as limitações de capacidade de armazenamento dos tipos de nós (sensores, drones, humanos e veículos). Os interesses pendentes são implementados através da junção do *host* ICN e do *switch*. Uma vez que um interesse é exposto, este permanece aguardando através de uma face até chegar um dado correspondente a este interesse. Esses interesses pendentes são armazenados em uma tabela no *host* minindn, e eles podem ser bloqueantes, ou não bloqueantes (através de *threads* ou processos distintos de outras execuções do *host*). O *switch* faz o papel de receber o dado que responde aquele interesse e encaminha pra face UDP correspondente que atende aquele nome. Finalmente, o *forwarding* é feito totalmente pelo *switch*, onde ele possui o programa eBPF pronto para dar *match* em um determinado nome e ser capaz de encaminhar ou para outro *switch* ICN, ou para a face UDP correspondente a um *host* conectado ao nome de um interesse ou de um dado.

Com isso, temos uma rede ICN pronta para operar segundo as regras que são determinadas por um controlador SDN, que necessita dar ordens para a rede atender todos os requisitos da arquitetura.

### 5.5.4 Controlador

O controlador é a conexão entre todos os outros componentes, cujo determina as políticas de filtragem, encaminhamento, *caching*, reconhecimento de *assets*, entre outras. Os *assets* se comunicam com um controlador através de pacotes de interesse #1 da Tabela 5.1, como mostrado no diagrama de sequência da Figura 5.2. O controlador recebe estes pacotes através de um PACKET IN, e após analisar o nome do pacote e concluir que este é um pacote de controle, ele é diretamente encaminhado para o C2 do *asset* comandante (i.e. um veículo). Uma vez que este interesse é respondido, é obtido dados, como a missão a ser executada, parâmetros de área e objetivos, entre outros detalhes para

completar sua missão. Sempre que um novo *asset* é ativado para começar uma operação, e ele é inicializado no estado pronto para operar.

Como citado no subcapítulo 5.5.1, sempre que um novo *switch* ingressa na rede, o controlador envia um programa eBPF para o mesmo, indicando como os pacotes que passarem por ele devem ser processados, de forma que este programa torne o *switch* capaz de processar nomes, e não apenas IP, MAC e outros parâmetros pré-determinados pelo OpenFlow. Estes programas foram baseados nos programas do trabalho (ZURANIEWSKI et al., 2017).

Quando o controlador retorna a missão para um novo *asset* na rede, ao mesmo tempo, o controlador imediatamente registra esse *asset*, assim como suas coordenadas atuais, para que este seja capaz de redirecionar fluxos de forma otimizada quando estes entram na rede, através do reconhecimento da topologia atual. Como se trata de uma rede sem fio em que os nós estão em constante movimento, é necessário que os nós constantemente atualizem suas posições, e isso é feito através do interesse #2 da Tabela 5.1 que funciona como um *"heartbeat"*. O tempo de atualização desses nós podem ser definidos pelo comandante da missão que está em um veículo com um controlador, visto que cada missão, e também cada nó, pode requerer ter um tempo de atualização do local diferente.

Além das funções descritas, o controlador também é responsável pelo controle de hierarquia, otimização de caminho e da segurança. O controle de hierarquia é definido pelo comandante que opera o controlador, e esta hierarquia é imediatamente aplicada toda vez que um *asset* se registra na rede através do interesse #1, pois se toma conhecimento do tipo do *asset*, assim como a face correspondente para alcançá-lo. Esta comunicação deve ser criptografada, de forma que invasores não consigam registrar um *asset* sem permissão. Uma vez que o *asset* é devidamente registrado, e há um PACKET IN vindo de algum *switch* com alguma operação relacionado com ele, então é imediatamente registrado uma regra de fluxo que força o *switch* encaminhar o interesse deste *asset* apenas para nós que ele possui autorização, caso contrário ele é descartado.

Para a otimização de caminho, existem alguns fatores que devem ser considerados: 1) a distância física entre os nós; 2) o tráfego de fluxo de dados entre os nós; 3) o *caching* correspondente dos nós. Estes três itens são avaliados de forma que o item 2 possui mais peso, uma vez que um caminho congestionado pode comprometer a criticidade de tempo da informação, seguido pelo item 1, visto que uma rede de uma missão militar específica atua dentro de uma área limitada, então a distância física na maioria dos casos não será um fator de peso alto, e finalmente o item 3 determina onde um fluxo pode ou não passar.

Se um determinado caminho não tem *caching* disponível suficiente, os dados não devem passar por ele, a não ser que seja uma mensagem de prioridade alta. Para a otimização de caminho foi utilizado o algoritmo de Dijkstra, que calcula o peso baseado nos fluxos de dados entre os enlaces.

A segurança é controlada através dos status dos nós, da criptografia que apenas o controlador deve ser capaz de reconhecer. O filtro é aplicado utilizando todos estes módulos.

E finalmente, o controlador deve ser capaz de reconhecer tipos diferentes de pacotes, o que realmente acontece de fato. Sempre que houver um PACKET IN o controlador primeiramente irá procurar se há um nome ICN presente no pacote UDP, e se houver, ele detecta a face correspondente àquele nome e encaminhará normalmente dentro da rede ICN, independentemente se vier da rede IP ou ICN. Então, sempre que um pacote passa pelo controlador de alguma forma, este primeiro passa por um reconhecimento para ser devidamente encaminhado.

No programa Ryu implementado, sempre que um nó entra na rede este se registra no controlador através de seu IP, e então é mantida uma lista de IPs de cada tipo de dispositivo. O controlador também armazena todos os pacotes DATA que os nós podem produzir, além de armazenar sempre que um dispositivo possui esse pacote DATA em *caching* para fazer o redirecionamento da melhor forma possível. Sempre que um PACKET IN ocorre essas listas são percorridas, de forma que é criada uma nova regra de encaminhamento a partir do cálculo de melhor caminho, com um *timeout* de regra de caminho especificada pela aplicação C2.

### 5.5.5 Aplicação de Controle (Command and Control)

A aplicação de controle define os parâmetros a serem executados na rede, tais como definição da hierarquia dos *assets*, elementos que operam com prioridade, e outras configurações. A aplicação também é responsável por gerar os pacotes INTEREST e DATA para disseminá-los pela rede, de forma que este define o que cada nó efetivamente fará no campo de batalha.

Na aplicação implementada cada nó é um produtor e também um consumidor, de forma que as requisições ICN são atendidas pelas informações geradas. Para as aplicações serem construídas foi utilizada a biblioteca **ndn-cxx** provida pelo projeto NDN (PAPADOPOULOS et al., 2019).

A aplicação em si gera dados de tamanhos pré-determinados em bytes para que aplicações e testes possam ser feitos. Uma vez que não conseguimos reproduzir um campo de batalha e o tempo de processamento de cada operação utilizando um simulador, o tempo de gerar os dados dos produtores são fixos.

O programa C2 é um programa em C++ que roda em qualquer dispositivo presente na rede em que é possível setar parâmetros e definir o que cada dispositivo consegue responder e gerar através de pacotes DATA. Na implementação os programas C++ geravam dados com bytes aleatórios para simular dados diversos que trafegam pela rede.

# 6 EXPERIMENTOS E RESULTADOS

Para ilustrar a aplicação da arquitetura SDN-ICN proposta auxiliando uma operação militar, os seguintes experimentos foram feitos para mostrar alguns resultados imediatos. #1: Medição da latência fim-a-fim de comunicações entre os nós; #2: Taxa de entrega de pacotes na rede; #3: Uso da largura de banda de acordo com o uso da rede.

Para executar estes experimentos, foi utilizada a implementação apresentada na seção 5.5 a qual é capaz de criar uma rede completa de forma que as abordagens ICN e SDN trabalhem em conjunto. Para a medição de cada resultado foram utilizadas algumas topologias com um número variado de nós, de forma a ser capaz de observar o impacto direto da abordagem apresentada neste trabalho na performance, largura de banda, latência e outras métricas.

Em todos os experimentos, para se obter uma noção direta de melhoria pelas abordagens foram feitos experimentos para redes exclusivamente IP, redes IP com SDN, e a abordagem apresentada neste trabalho, redes ICN em conjunto com SDN. Para todos os casos os experimentos foram efetuados em topologias o mais semelhantes possíveis.

Para redes exclusivamente IP e redes IP com SDN, foram vinculados o tipo de *asset* com os IPs, de forma que quando uma informação fosse desejada de um deles através do nome, fosse possível resgatar diretamente no produtor.

A tabela 6.1 apresenta a configuração de rede para os nós. Foram feitos experimentos para 20, 40, 60, 80 e 100 nós para analisar a escalabilidade da rede nos diferentes cenários testados.

O número de delay dos enlaces foi escolhido fixo e relativamente baixo de forma a mostrar que independente da taxa de delay, o que determina a alta variação do atraso de entrega nas redes, são os algoritmos de encaminhamento, isto é, a quantidade de hops que devem ser percorridos, e também enlaces que estão sobrecarregados.

A capacidade de caching também foi escolhida com capacidades possíveis de serem aplicados em um cenário real de acordo com as capacidades de armazenamento atuais.

## 6.1 Inicialização da Rede Para os Experimentos

Em todos os experimentos que serão apresentados, foram disseminados fluxos na rede para medir e avaliar a reação da rede como um todo. Para a disseminação desses

Tabela 6.1: Parâmetros padrões para os experimentos

| **Nós totais** | **20 nós** | **40 nós** | **60 nós** | **80 nós** | **100 nós** |
|---|---|---|---|---|---|
| **Sensores** | 4 | 10 | 18 | 25 | 31 |
| **Drones** | 8 | 14 | 22 | 25 | 38 |
| **Humanos** | 6 | 12 | 16 | 25 | 25 |
| **Veículos** | 2 | 4 | 4 | 5 | 6 |
| **Delay dos enlaces** | 2ms | | | | |
| **Cache disponível** | Sensor : 100MB<br>Drone: 1GB<br>Humano: 4GB<br>Veículo: 5 TB | | | | |
| **Tipo de pacote** | UDP | | | | |

fluxos é necessário primeiro uma pré-configuração da rede, de forma que os nós fiquem prontos para atender as requisições e fluxos desejados. Três configurações distintas foram feitas, uma para cada um dos tipos de rede testada (IP, IP+SDN, ICN+SDN).

### 6.1.1 Redes IP

Na inicialização da rede as aplicações que propagam o fluxo recebem uma lista de IPs que respondem por um tipo de *asset*, por exemplo, como apresentada na tabela 6.1 para 20 nós, existiria uma lista de drones que apontaria para 4 IPs distintos. Em uma rede IP não é possível a nível de rede manter os status atuais de um nó, apenas é possível construindo uma aplicação *heartbeat* que constantemente envia uma mensagem para obter o status do nó. Como todos os nós teriam que fazer isso para todos os outros nós, foi simplificado de forma que em cada requisição de uma informação para um tipo de *asset* fosse percorrida toda a lista de IPs, até que alguém responda a requisição. Cada nó então possui uma aplicação de produtor de *background* para responder requisições, assim como de qualquer nó é possível disparar uma aplicação de consumidor para requisitar uma informação dos outros nós.

### 6.1.2 Redes IP+SDN

Na inicialização da rede um nó indica para o controlador seu tipo (sensor, drone, humano ou veículo), de forma que a lista de nós de um determinado tipo fiquem armazenados no controlador. Assim, em todas as aplicações nos nós é possível deixar apenas um IP para um tipo de nó, de forma que quando este IP chega no controlador é processado,

e logo após é verificado qual o melhor nó para atender esta requisição. Uma vez que o controlador tem uma visão global da rede é possível ver o nó com menos hops dentro da lista de IPs, além de também ser possível otimizar em perspectiva de largura de banda disponível em cada canal.

### 6.1.3 Redes ICN+SDN

Para os experimentos funcionarem em uma rede simulada, os pacotes precisam ser reconhecidos pelo controlador SDN em forma de nomes, e como explicado na seção 5.5 isso pode ser feito através de um *match* especial onde os *switches* ICN são capazes de encaminhar os pacotes através de nomes. Sempre que um novo nó entra na rede, ele envia um INTEREST #1 como apresentado no diagrama de sequência na Figura **??**, de forma que o controlador sabe exatamente onde um nó se encontra através do seu nome, além de manter um *hearbeat* com o INTEREST #2 periodicamente. A principal diferença entre ICN e IP com a abordagem SDN se dá no *caching*, onde sempre que um nó armazena um dado através do *caching* este também avisa ao controlador que possui esta informação, de forma que sempre que uma requisição por esta informação passar pelo controlador, ela poderá ser recuperada tanto em um dos produtores diretos estabelecidos com o INTEREST #1 na inicialização de todos os nós, como por algum dos nós que tem esta informação armazenada em *caching*.

Com estas configurações de inicialização de rede é possível enfim disseminar informação entre os diversos nós que estão presentes na rede.

## 6.2 Caso de Estudo #1: Otimização de Latência

Para comprovar e verificar o uso de *caching* e da otimização de caminhos através do trabalho conjunto de SDN e ICN, foram utilizados fluxos fim-a-fim para determinar o impacto causado. Para isto, alguns pacotes DATA foram criados de forma que todos os nós respondessem esses pacotes DATA dependendo de seu tipo. Os pacotes DATA criados foram: DATA 1- produção de dados aéreos do drone; DATA 2- produção de dados da saúde do soldado por equipamentos vestíveis em humanos; DATA 3- obtenção de movimento de sensores. Estes pacotes são respondidos de acordo com uma requisição INTEREST que chega de outro nó.

O objetivo deste caso de estudo é mostrar como o *round-trip time* (RTT) e o uso da largura de banda podem ser otimizados através do uso de *caching*. É natural que este resultado seja esperado, uma vez que os dados tendem a estar mais próximos dos nós que requisitam alguma informação. Embora o *caching* consiga obter algumas otimizações, também é levado em conta a disponibilidade de cache de cada dispositivo da rede. Alguns dispositivos como sensores e drones possuem um cache mais limitado, então os mesmos não podem ser utilizados para armazenamento de informações muito custosas como grandes arquivos de multimídia em geral.

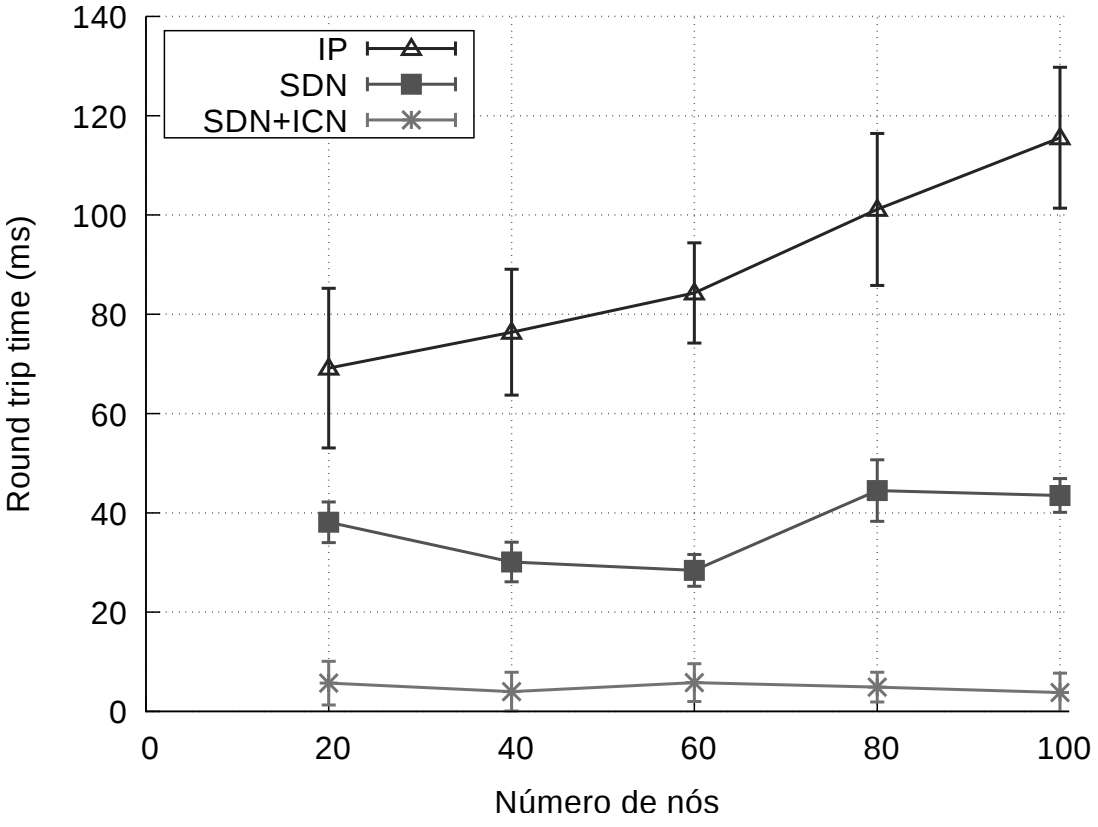
Para este experimento, o tempo de produção da informação foi considerada nula (isto é, a duração da atividade para a produção daquela informação), uma vez que o interesse em questão é o tempo de propagação dos dados pela rede, e não a execução da tarefa. Uma vez que estes pacotes DATA foram estabelecidos, todos os drones são produtores do DATA tipo 1, todos os humanos produtores do DATA tipo 2, e os sensores responsáveis pelo DATA tipo 3.

Como dito anteriormente, foi testado para os 3 tipos de rede, onde para:

- Redes exclusivamente IP: foram disseminadas requisições de consumidores para os produtores, e como não é possível nativamente pela rede obter se o nó está vivo, foi criada uma lista dos nós que produzem determinada informação, como por exemplo, para o tipo DATA 1 existe uma lista de IPs vinculadas, onde estes nós são requisitados de qualquer um destes IPs.
- Redes IP com SDN: através do controlador é possível manter estatísticas, assim como os nós vivos e também fazer otimização de caminho utilizando informações de largura de banda da rede e número de saltos necessários para chegar de um nó a outro, os quais foram utilizados para a definição do melhor caminho. O controlador, para cada pacote DATA distinto, armazenou uma lista de IPs vinculados e sempre que uma requisição por um tipo de DATA fosse feita o controlador calcula o melhor nó nesta lista de nós baseado nas estatísticas da rede para direcionar a requisição.
- Redes ICN com SDN: utilizando a mesma abordagem que redes IP com SDN, os nós de interesse que são capazes de responder a um *INTEREST* foram calculados de acordo com o número de hops e largura de banda na rede. A grande diferença é que os nós que possuem esta determinada informação em armazenada em cache (este fato é conhecido pelo controlador, que sabe quais dispositivos estão guardando o que em cache) também entram no cálculo de rota para achar o nó que consiga responder no menor período de tempo.

Uma vez determinado como os caminhos são obtidos para a resposta de cada requisição *INTEREST*, fluxos foram disseminados na rede. **Um milhão de requisições** foram injetadas na rede de forma aleatória onde cada nó poderia requisitar um dos 3 tipos de dados produzidos pelos nós na rede. Foi escolhida essa quantidade de requisições, pois após uma grande quantidade de requisições a rede já possui a grande maioria dos dados disseminados em caching, possibilitando a melhoria dos resultados, uma vez que haverá uma redução de procura em diversos pontos da rede. É possível observar a redução de latência de acordo com o crescimento dos nós no gráfico da Figura 6.2. Os resultados obtidos estão apresentados na Figura 6.1. É possível observar que devido à otimização de caminhos as redes IP com SDN se sobressaem sobre as redes IP, e através do *caching* a rede ICN com SDN ultrapassa ambas as abordagens anteriores.

Figura 6.1: Latência na disseminação de pacotes na rede




A abordagem ICN com SDN tem sua melhoria graças ao *caching*, mas dependendo das circunstâncias da disponibilidade de *caching*, tempo de uso e frequência de pacotes, essa eficiência varia. Por exemplo, se o *caching* é limitado ao ponto de não conseguir armazenar nenhuma informação, a sua eficiência fica diretamente equivalente à abordagem de IP com SDN, já que os *switches* apenas encaminharão os pacotes sem

efetuar qualquer tipo de armazenamento intermediário. Em outro caso, se houver *caching* disponível porém se os pacotes sempre forem diferentes uns dos outros, a usabilidade do cache não será tão eficiente uma vez que os dados não serão reutilizados.

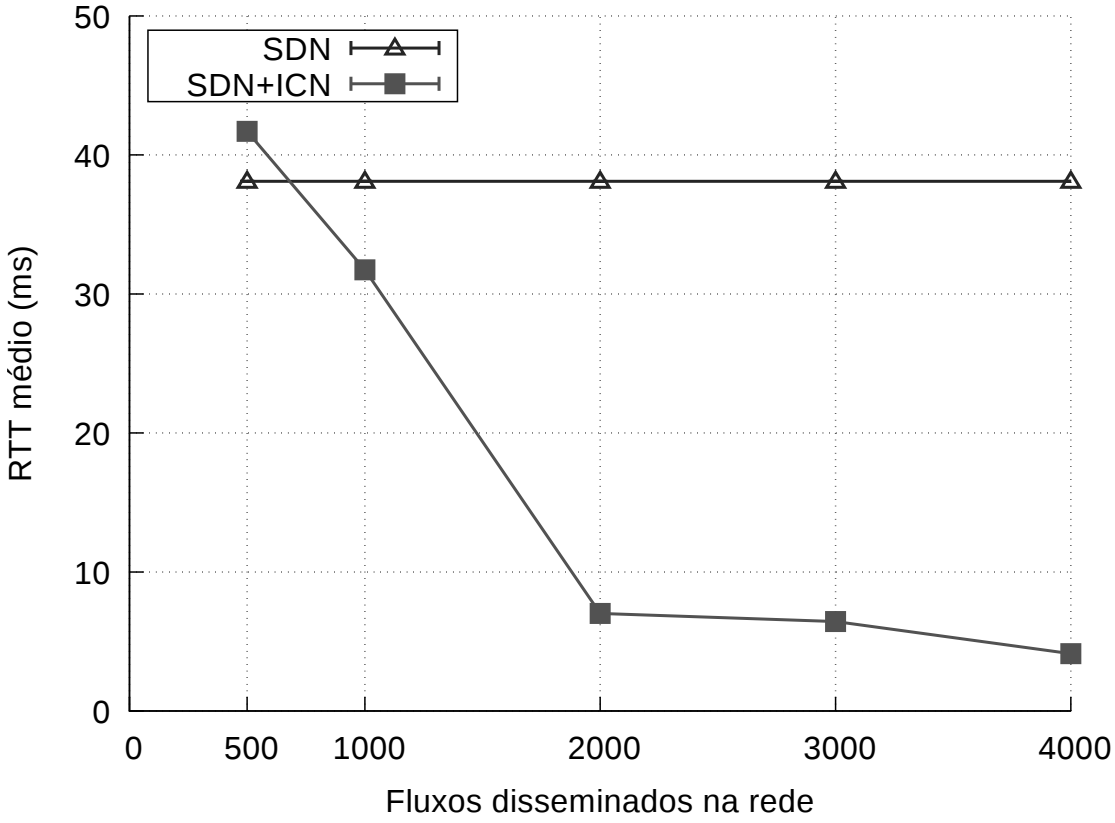
A Figura 6.2 mostra a otimização da rede quando 100 pacotes distintos são requisitados aleatoriamente na rede por diferentes nós, com o tempo esses pacotes ficam espalhados e mais perto do consumidor, reduzindo o tempo para alcançar os dados desejados. Este experimento foi feito em cima da topologia de 20 nós, onde esses pacotes foram disseminados lentamente pela rede aleatoriamente, de forma que é possível observar no eixo x a quantidade de pacotes disseminados ao decorrer do tempo, e também foi exposta uma constante de referência do tempo gasto pela abordagem SDN+IP no gráfico para termos de comparação. É possível observar que quando poucos fluxos foram disseminados a performance de ICN+SDN e IP+SDN são extremamente semelhantes, mas de acordo com o uso da rede o ICN acaba tendo uma grande melhoria em termos de tempo, uma vez que os pacotes ficam distribuídos pela rede.

Para este e para os próximos experimentos é possível observar que os resultados para redes IP são muito piores, devido à falta de conhecimento da topologia e quais nós estão disponíveis, então a comunicação é baseada em tentativa e falha a partir da lista de IPs vinculados aos nomes, assim a variação de latência também cresce, uma vez que os IPs estão distribuídos pela rede e não é mantido uma estatística na infraestrutura da rede para saber qual o produtor mais perto, ou com menor latência, além de que o protocolo de roteamento é fixo, então uma abordagem para melhoria deveria ser implementado a nível de aplicação, o que reduz bastante a eficiência do encaminhamento e faz com que os resultados fiquem extremamente ruins comparados às soluções aplicadas em nível de rede.

Também para validar o quanto a disponibilidade de *caching* impacta na otimização, foram feitos experimentos limitando a quantidade de cache disponível a partir das configurações apresentadas na Tabela 6.1, de forma que apenas uma porcentagem dele fosse disponibilizado.

Os resultados podem ser observados na Figura 6.3. É possível observar que quanto maior o *caching* disponível, menor será o RTT médio. Isso acontece porque com uma quantidade de *caching* maior é possível armazenar mais pacotes fazendo com que não haja substituição de pacotes no *caching* ao sobrecarregar sua capacidade de armazenamento, assim a grande maioria dos pacotes ficam armazenados mais perto de seus consumidores reduzindo o tempo de busca por esses dados em geral.

Figura 6.2: Latência de acordo com o uso da rede

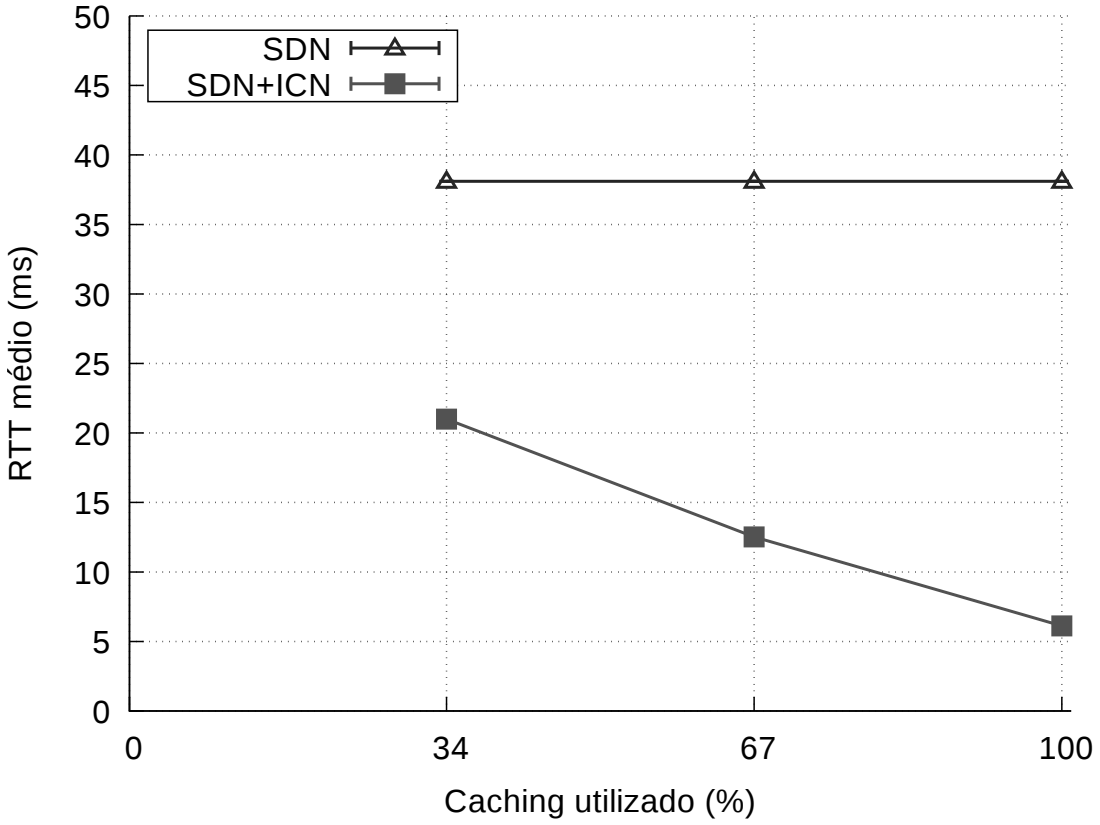



No cenário ideal todos os pacotes utilizados na rede estariam armazenados em *caching*, porém isto em uma rede real é inevitavelmente falso devido à limitação de armazenamento. Uma forma de otimizar é aplicando algoritmos de substituição, tais como manter em *caching* o pacote mais usado, ou manter o mais recente, entre outros, que pode gerar trabalhos futuros para melhorias da arquitetura apresentada neste trabalho.

## 6.3 Caso de Estudo #2: Minimização do Uso dos Enlaces na Rede

Neste experimento é medido o impacto direto do *caching* e da otimização de caminho presente na rede, os quais reduzem o uso dos enlaces da rede em geral, o que ajuda a manter a estabilidade da rede para grandes quantidades de informações. Este estudo de caso também lida com a necessidade de flexibilidade da configuração da rede para gerenciar padrões de interação (como os nós se comunicam), alocação de direitos de decisão (quem pode modificar o itinerário dos drones, por exemplo) e disseminação da informação (filtragem dos fluxos de dados desnecessários ou não permitidos), além da capacidade de adaptar a diferentes cenários de disponibilidade dos nós graças ao limite de *caching*

Figura 6.3: Latência de acordo com o caching disponível




presente nos mesmos.

Como apresentado no cenário de aplicação, o controlador SDN pode estabelecer qual nó pode enviar determinado tipo de mensagem para outro nó. Este aspecto é implementado utilizando filtros que permitirão que pacotes *INTEREST* e pacotes *DATA* consigam atender as configurações especificadas pelo administrador da rede. Esses parâmetros caracterizam a abordagem C2 utilizada. Alterando as configurações também modifica o plano de execução das ações para uma característica mais centralizada ou descentralizada de acordo com as circunstâncias.

Assim sendo, o controle de hierarquia e a filtragem de pacotes tem um papel importante para os resultados aqui apresentados, de forma que pacotes não autorizados são imediatamente descartados antes de possivelmente utilizar largura de banda desnecessariamente, visto que é um pacote que viola as regras estabelecidas para a arquitetura. Cada um destes aspectos é melhor explicado a seguir:

- Controle de hierarquia: alguns dispositivos não possuem permissão de requisitar objetos nomeados que contém informações fora de seu escopo permitido estabelecido pelo controle da missão. Quanto mais cedo tais requisições não permitidas

forem detectadas, mais efetivo será o filtro, uma vez que ele pode prevenir que fluxos de dados desnecessários adentrem a rede.

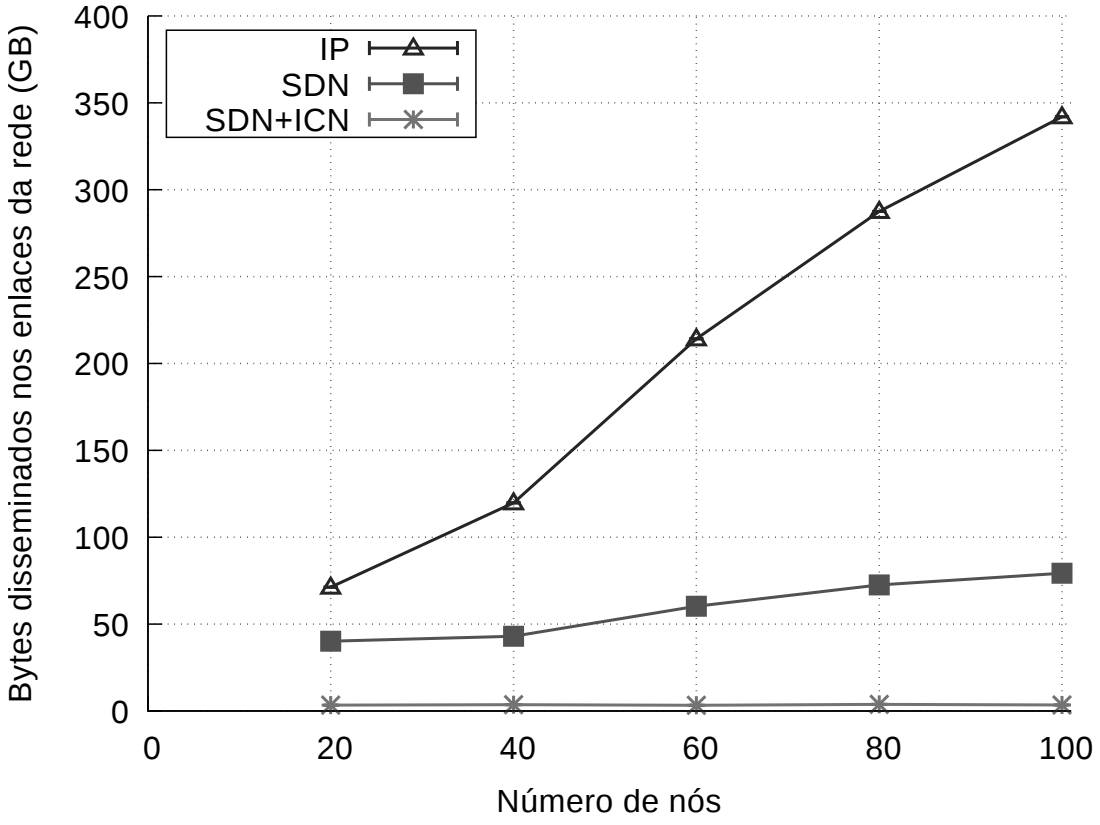
- Tamanho do *caching*: alguns nós possuem capacidade de armazenamento interno limitado e podem ser facilmente sobrecarregados com objetos nomeados muito grandes (como por exemplo vídeos), reduzindo sua capacidade de manter informações importantes ou de encaminhar informações de maiores prioridades. Isto pode ser evitado por um policiamento de *caching*, descartando pacotes de menores prioridades, porém armazenamento e largura de banda seriam desnecessariamente utilizados, uma vez que estas informações seriam descartadas apenas quando chegasse em um nó bloqueante em termos de disponibilidade de *caching*. Ao invés disso, como a rede SDN provê visão global da rede incluindo disponibilidade de *caching* em geral que é atualizada periodicamente utilizando o INTEREST de *heartbeat* apresentado na seção 5.3, ela é responsável por descartar essas informações com menos impacto na rede mais perto da borda da rede, minimizando qualquer impacto de um pacote que possa causar congestionamento, ou até mesmo propagar este fluxo por um caminho alternativo onde há disponibilidade de armazenamento em todos os nós, mesmo que este caminho seja maior.
- Otimização de caminho: Para reduzir a latência, o caminho com a maior largura de banda disponível, ou até mesmo o que houver menos saltos entre nós será escolhido, e também o *caching* disponível de cada nó deve ser avaliado para verificar se o caminho escolhido é utilizável.

Para obter os resultados deste experimento foram disseminados fluxos de tamanhos diversos (variando pacotes entre 100KB e 500MB) pela rede de forma que fosse detectado os dados que passaram em cada enlace. Os mesmos pacotes na mesma ordem foram disseminados em todos os três tipos de rede.

Com a disseminação dos fluxos, diversos canais da rede foram utilizados para distribuir a informação entre os nós. O controlador SDN consegue capturar as informações de uso da largura de banda em cada enlace, e o somatório da quantidade de dados foi medido e avaliado para observar a diferença entre as abordagens testadas. Os resultados podem ser observados na Figura 6.4.

É possível observar que os fluxos IP são os que mais utilizaram largura de banda, o que já era esperado, uma vez que esta abordagem não possui nenhum mecanismo de controle dos pacotes tal como os filtros e controle de hierarquia a nível de rede, assim como os caminhos até o destino não são dinâmicos observando o comportamento da rede,

Figura 6.4: Bytes disseminados na rede após a disseminação de fluxos




fazendo com que muitas vezes percorram um caminho mais longo, utilizando mais enlaces para transmitir a informação.

Em seguida vem a abordagem IP+SDN que já possui otimização a níveis de melhoria de encaminhamento de pacotes, o que já diminui bastante o uso dos enlaces, uma vez que os caminhos utilizados para a transmissão dos dados são pré-calculados de forma que se ache o menor e/ou mais disponível.

Os melhores resultados se deram nas redes ICN+SDN, que além da otimização de caminhos, também otimiza através do *caching*, onde as informações tendem a ficar ainda mais perto podendo estar presente até mesmo no próprio nó o que torna o uso da largura de banda da rede nula, além dos mecanismos de filtragem por tamanho de pacote dependendo da capacidade de *caching* de cada nó e também das hierarquias pré-estabelecidas pelo C2.

## 6.4 Caso de Estudo #3: Minimização de Perda de Pacotes e Aumento de Confiabilidade da Rede

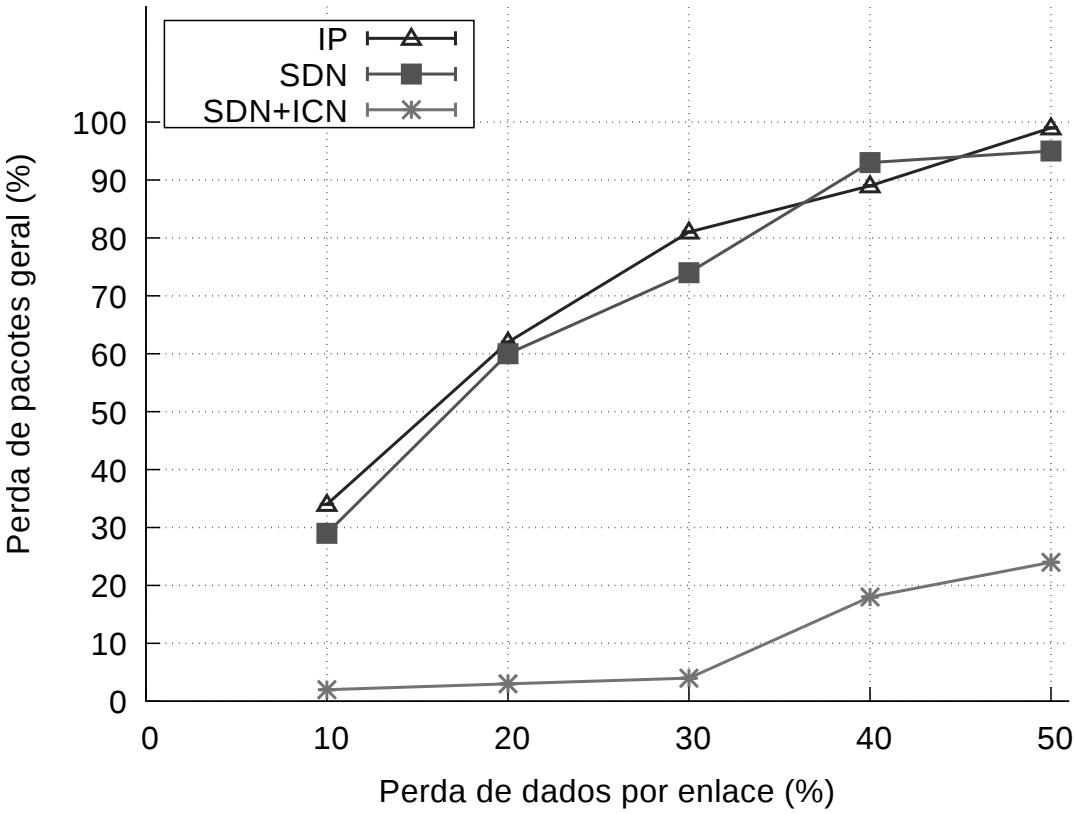
A confiabilidade em uma rede militar é crítica, onde algumas informações devem ser mantidas de forma confiável. Um dos maiores problemas com o campo de batalha é que os nós estão propensos a sofrer ataques e assim se tornarem indisponíveis, impossibilitando o acesso a algumas informações.

Com a capacidade de ICN de replicação e *caching* dos dados essa característica tem uma tremenda melhora, uma vez que além de a informação ficar armazenada mais perto de quem requisita fazendo os pacotes passarem por menos canais que podem estar sobrecarregados ou ter sofrido um ataque, os pacotes também estão distribuídos pela rede, fazendo com que seja possível recuperar a informação desejada em mais de um local desejado.

Para este experimento, foi inicializada uma rede de forma que os enlaces estão instáveis e possuem uma porcentagem de perda de pacotes. A porcentagem foi incrementada para analisar como os diferentes tipos de redes se comportam diante de uma rede instável e com alta incidência de perda de pacotes. Os resultados deste experimento pode ser observado na figura Figura 6.5.

A diferença entre as redes IP com e sem SDN é que os caminhos são mais curtos, para SDN, uma vez que este possui uma otimização de caminho, então é esperado assim como observado no gráfico que quase sempre SDN possua uma menor perda de pacotes. Nas redes ICN por outro lado já possui uma grande queda na perda de pacotes pelas argumentações previamente apresentadas, o pacote em sua maioria será retornado de um *caching* em algum nó extremamente próximo ou até mesmo do próprio nó que está requisitando determinado pacote DATA.

Figura 6.5: Taxa de perda de pacotes na rede



# 7 CONCLUSÃO E TRABALHOS FUTUROS

Os cenários de campo de batalha modernos demandam mudanças importantes na forma em que o comando e controle é efetuado. Para cobrir estas necessidades introduzidas, com os avanços da tecnologia da informação e da comunicação novos mecanismos estão surgindo para cobrir os novos cenários.

Observando os avanços nos paradigmas ICN e SDN, este trabalho propõe uma combinação explorando suas melhores características para endereçar este complexo cenário militar. A arquitetura proposta provê meios de mapear as necessidades de alto nível necessárias para a Agilidade C2 para o nível de rede. Este mapeamento oferece uma otimização de latência, uso de largura de banda, e uma maior confiabilidade em perda de pacotes, de acordo com a configuração da rede e os recursos disponíveis. Entretanto, além dos benefícios apresentados neste trabalho, ainda existem alguns desafios em aberto.

Os controladores possuem daemons secundários para tomar o lugar caso este se encontre indisponível. Um possível estudo seria avaliar o melhor algoritmo de eleição possível para este cenário, além de abordar como seria efetuada a autenticação e formas de autenticar os controladores, daemons e dispositivos, de forma que um dispositivo não autorizado acabe tomando o lugar do controlador e assim possuindo controle sobre toda a rede. Também a partir deste último ponto, é possível desenvolver estudos para verificar se qualquer um dos dispositivos se tornou um nó malicioso, de forma que atue como um buraco negro ou corrompa a rede de alguma forma.

Outra questão muito importante é a respeito da mobilidade dos drones e como a variação do posicionamento geográfico afeta os resultados obtidos. No cenário de aplicação apresentado há um território delimitado com uma extensão não muito grande, que se trata de um ambiente civil bem determinado. Porém para ambientes mais variáveis e extensos deve ser obtido o impacto direto da variação do posicionamento dos dispositivos. Além da mobilidade, muitos dos dispositivos possuem capacidades limitadas a respeito de bateria, e tempo em que é possível operar em campo, o que abrange mais um tema de estudo possível.

A taxa frequência da comunicação dos dispositivos com o controlador é mais um ponto que deve ser observado cuidadosamente, pois embora as comunicações apresentadas neste trabalho são efetuadas em quantidade a nível de bytes, a tornando praticamente desprezível, para cenários com maior quantidades de nós, ou que é necessária atualização muito frequente e com dados mais complexos e grandes, a comunicação com o contro-

lador pode crescer e afetar diretamente no uso dos enlaces, congestionando a rede. Este é outro tema que pode ser estudado e observar a variação dos resultados de acordo com a variação desta frequência. Além disso, também devem ser feitas pesquisas com variação da taxa de delay, uma vez que o atraso das respostas desses pacotes podem afetar o desempenho da rede em geral.

A segurança dos nós da rede podem sofrer uma melhoria através de uma autenticação mais rígida, de forma que mantenha a integridade e privacidade dos dados. O gerenciamento do *caching* é outro ponto que pode ser melhorado, uma vez que quando um dispositivo cache fica cheio, o pacote com maior relevância deve continuar, o que pode ser feito com algoritmos que analisam a frequência de uso de cada pacote, ou aprendizagem de máquina, entre outros. Algoritmos que otimizam o caminho baseado no *caching* atual da rede também podem ser desenvolvidos, de forma que as informações se tornem distribuídas de forma mais uniforme possível, onde os dados ficam facilmente acessíveis de qualquer ponto da rede, sem ter que efetuar muitos saltos por nós da rede. A melhoria da comunicação entre SDN e ICN é outro ponto que merece destaque.

Embora este trabalho tenha apresentado a abordagem modelada em um ambiente militar, é possível expandir para redes mais genéricas para obter benefícios semelhantes, variando do tipo da aplicação. Grandes cidades podem se beneficiar da reutilização de redes, uma vez que grande parte dos serviços atuais já utilizam redes de fornecimento de conteúdo, onde arquivos que possuem uma alta taxa repetição entre diferentes clientes são armazenados em servidores de *caching* centrais. Com a proposta apresentada os benefícios seriam muito mais facilmente obtidos. Cenários de casos mais específicos e reais, a partir de redes já atualmente implementadas também podem ser estudadas, de forma a aplicar esta arquitetura e comparar o impacto direto com estudos já feitos em cima de um caso de uso já prontamente estudado e com experimentos disponíveis para comparação.

Boa parte das melhorias obtidas neste trabalho se dão em cima do *caching*, portanto o aprofundamento em como o *caching* é utilizado, métodos de substituição das informações armazenadas, versionamento, e outros fatores também podem ser melhor observados.

A implementação dos protocolos que integram completamente a proposta aqui apresentada é também um grande desafio que requer esforço tanto dos meios acadêmicos quanto dos meios industriais.

# REFERÊNCIAS

AHLGREN, B. et al. A survey of information-centric networking. **IEEE Communications Magazine**, IEEE, v. 50, n. 7, 2012.

ALBERTS, D.; AL. et. Sas-085 final report on c2 agility. **NATO Research and Technology Organization**, NATO Research and Technology Organization, 2013. Available from Internet: <http://www.dodccrp.org/sas-085/>.

BACCELLI, E. et al. Information centric networking in the iot: Experiments with ndn in the wild. In: **Proceedings of the 1st ACM Conference on Information-Centric Networking**. New York, NY, USA: ACM, 2014. (ACM-ICN '14), p. 77–86. ISBN 978-1-4503-3206-4. Available from Internet: <http://doi.acm.org/10.1145/2660129.2660144>.

BOSSHART, P. et al. P4: Programming protocol-independent packet processors. **ACM SIGCOMM Computer Communication Review**, ACM, v. 44, n. 3, p. 87–95, 2014.

FERNANDES, E. L. et al. The road to bofuss: The basic openflow user-space software switch. **arXiv preprint arXiv:1901.06699**, 2019.

GONZALEZ, C. et al. Sdn-based security framework for the iot in distributed grid. In: **2016 International Multidisciplinary Conference on Computer and Energy Science (SpliTech)**. [S.l.: s.n.], 2016. p. 1–5.

KHONDOKER, R. et al. Feature-based comparison and selection of software defined networking (sdn) controllers. In: IEEE. **2014 World Congress on Computer Applications and Information Systems (WCCAIS)**. [S.l.], 2014. p. 1–7.

KOTT, A.; ALBERTS, D. S. How do you command an army of intelligent things? **Computer**, IEEE, v. 50, n. 12, p. 96–100, 2017.

KOTT, A.; SWAMI, A.; WEST, B. J. The internet of battle things. **Computer**, IEEE, v. 49, n. 12, p. 70–75, 2016.

LANE, A. et al. **MiniNDN**. MiniNDN Developers, 2019. Available from Internet: <https://github.com/named-data/mini-ndn>.

LANTZ, B.; HELLER, B.; MCKEOWN, N. A network in a laptop: rapid prototyping for software-defined networks. In: ACM. **Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks**. [S.l.], 2010. p. 19.

LV, J. et al. Risc: Icn routing mechanism incorporating sdn and community division. **Computer Networks**, Elsevier, v. 123, p. 88–103, 2017.

MELAZZI, N. B.; CHIARIGLIONE, L. The potential of information centric networking in two illustrative use scenarios: mobile video delivery and network management in disaster situations. **E-LETTER**, 2013.

NOBRE, J. et al. Toward software-defined battlefield networking. **IEEE Communications Magazine**, IEEE, v. 54, n. 10, p. 152–157, 2016.

NUNES, B. A. A. et al. A survey of software-defined networking: Past, present, and future of programmable networks. **IEEE Communications Surveys & Tutorials**, IEEE, v. 16, n. 3, p. 1617–1634, 2014.

OH, S. Y.; LAU, D.; GERLA, M. Content centric networking in tactical and emergency manets. In: **2010 IFIP Wireless Days**. [S.l.: s.n.], 2010. p. 1–5. ISSN 2156-9711.

PAPADOPOULOS, C. et al. **NDN Project**. NDN Project Participants, 2019. Available from Internet: <https://named-data.net/>.

SIRACUSANO, G. et al. A framework for experimenting icn over sdn solutions using physical and virtual testbeds. **Computer Networks**, Elsevier, v. 134, p. 245–259, 2018.

WICKBOLDT, J. A. et al. Software-defined networking: management requirements and challenges. **IEEE Communications Magazine**, IEEE, v. 53, n. 1, p. 278–285, 2015.

ZACARIAS, I. et al. Combining software-defined and delay-tolerant approaches in last-mile tactical edge networking. **IEEE Communications Magazine**, IEEE, v. 55, n. 10, p. 22–29, 2017.

ZHANG, X.; ZHU, Q. Information-centric network virtualization for qos provisioning over software defined wireless networks. In: IEEE. **Military Communications Conference, MILCOM 2016-2016 IEEE**. [S.l.], 2016. p. 1028–1033.

ZURANIEWSKI, P. et al. Facilitating icn deployment with an extended openflow protocol. In: ASSOCIATION FOR COMPUTING MACHINERY, INC. **4th ACM Conference on Information-Centric Networking, ICN 2017. 26 September 2017 through 28 September 2017, 123-133**. [S.l.], 2017.

# Empowering Command and Control through a Combination of Information-Centric Networking and Software Defined Networking

Gabriel Martins Leal, Iulisloi Zacarias, Jorgito Matiuzzi Stocchero, and Edison Pignaton de Freitas

The authors seek to merge information-centric networking (ICN) with software-defined networking (SDN) to meet the high-level operational requirements for "C2 agility" within deployed networks. ICN provides a more efficient data distribution within "ICN islands" in the military IP network, while SDN enables ICN to be integrated with the rest of the IP network, as well as control the patterns of interaction among the IoBT heterogeneous nodes, thus enhancing ICN features.

## Abstract

Current operational scenarios are raising new challenges for the deployment of armed forces. New command and control (C2) approaches are emerging to address these problems, pushing responsibility to the edge of the battlefield in what is called network-centric C2 operations. Advances in network warfare have also led to the development of the Internet of Battle Things (IoBT), which combines everything on the battlefield that can support informed decision making. However, these new dynamics require new network paradigms that can be adapted to efficient network operations. In view of the lack of technical solutions, this work seeks to merge information-centric networking (ICN) with software-defined networking (SDN) to meet the high-level operational requirements for "C2 agility" within deployed networks. ICN provides a more efficient data distribution within "ICN islands" in the military IP network, while SDN enables ICN to be integrated with the rest of the IP network, as well as control the patterns of interaction among the IoBT heterogeneous nodes, thus enhancing ICN features. After reviewing key concepts in C2, IoBT, SDN, and ICN, the architecture was modeled in a specific application, and the first experiments have shown promising results.

## Introduction

Advances in information and communication technology (ICT) are paving the way for the rapid development of an Internet of Battle Things (IoBT) [1]. In this context, the "Things" on the battlefield are now not only able to acquire a larger amount of data, but also capable of functioning with varying degrees of autonomy. In light of this, people must efficiently interact with them in ways that are not yet well understood, which is evidence of the need for a reliable, robust and agile command and control (C2) system to support challenging military operations [2]. To be effective, the C2 system must use the intelligent things to collect and process data and share the information with decision makers. A suitable C2 approach should be chosen, from extremely network-centric, with distributed decision rights allocated to the "edge", to a very centralized system. This approach emerges from how decision rights are allocated, actors interact, and information is shared. When missions and circumstances change rapidly, it is necessary to adjust these variables accordingly, and this defines the concept of C2 agility [3]. New models and technical systems are required to achieve the needed C2 agility in the context of a large-scale and heterogeneous IoBT. From the standpoint of network management, data semantic oriented technologies such as information-centric networking (ICN) combined with software-defined networking (SDN) can play a key role.

ICN is a new paradigm that shifts the networking logic from host-centric to data-centric applications [4]. It supports intermittent connectivity, node mobility, and multiple access. These features are suitable to meet the requirements of battlefield networks (BNs). SDN provides network management with flexibility by separating the network infrastructure into distinct planes [5]; this allows considerable changes to be made in the network, ranging from routing configurations to security policy enforcement. Additionally, it can function as an ICN enabler by converting ICN packets to IP and vice versa [6].

In light of the lack of information-centric approaches in the literature addressing IoBT, this study adopts a joint ICN/SDN approach to support C2 agility, using ICN (enabled by SDN) for data dissemination and to control the patterns of node interactions in the network. The system benefits from both the programmability offered by SDN and the ability of ICN to handle data/information-oriented communication. The key idea of the combined SDN-ICN proposed approach is that the SDN controller programs the ICN switches forwarding logic in accordance with changes in operational requirements and in the network status.

Two key contributions to the literature can be highlighted in this article: first, a comprehensive review of C2 agility in the IoBT, with possible network solutions using ICN and SDN; and second, the definition of an architectural framework to support network-centric military operations in the reality of a challenging battlefield by adapting and combining ICN and SDN technologies, and describing an application scenario and two experimental use cases with their results.

Digital Object Identifier:
10.1109/MCOM.2019.1800288

*The authors are with the Federal University of Rio Grande do Sul (UFRGS).*


0163-6804/19/$25.00 © 2019 IEEE

## Background

### Agile Command and Control

Modern military operations are evolving the concepts of warfare. They range from conventional war scenarios, like Syria, to operations other than war, like the United Nations Peacekeeping missions in the Congo and Haiti, as well as the very specific Brazilian case of the Guarantee of Law and Order, where the army is sent to "pacify" regions of cities dominated by organized crime. In these situations, soldiers must act together with civilians in urban environments, dealing with both the uncertainty and increasing complexity of the situation.

These changes have not only increased the importance of C2, but have also confirmed the advantage in using a network-centric approach to the C2 system, to provide necessary data in time to support decentralized decision making. Previous studies [2] have defined C2 as the management or governance of military organizations and identified five functions (command, control, sense-making, execution, and situation monitoring) that are necessary to achieve the desired effects on the battlefield. Command establishes intent (goals and priorities) and creates the conditions (rules of engagement and constraints) for a successful operation. Control involves all the other functions repeated in cycles. Here, the concept of agility is applicable: the faster the control cycle operates, without impairing quality, the more effective the C2 system will be. Sensemaking (collecting, processing, and sharing information), execution (real action on the battlefield), and situation monitoring (real-time assessment of the operational effects on the environment) should all be strategically planned and integrated into a C2 approach that best suits the mission in hand.

The choice of the most suitable C2 approach is not trivial and depends on the dynamics of the situation (i.e., the current approach may need to be redesigned). The North Atlantic Treaty Organization (NATO) has produced a detailed report [3] resulting in a conceptual model of C2 agility. NATO concluded that C2 approaches can be categorized by how i) decision rights are allocated, ii) the different actors interact, and iii) information is distributed. These variables form the key dimensions of the C2 approach space — allocation of decision rights (ADR), patterns of interaction (PIs), and distribution of information (DI)). They are interdependent and form the C2 approach (ranging from highly centralized hierarchies to loosely coupled networks). C2 agility can thus be defined in terms of its capacity to dynamically alter the current C2 approach in response to operational changes.

### Internet of Battle Things

Following the same trend as the Internet of Things (IoT) [1], ground network warfare is moving toward the development of IoBT [1], consisting of a network of physical devices, vehicles, sensors, and any object designed to acquire, process, and/or share data. These things are able to provide a wide range of data of interest (enemy movements, resource status, etc.) to support informed decisions.

On the battlefield, these things are not just mere sensors that acquire and provide data, but also intelligent devices capable of both supplying information and making decisions. One example in a megacity operational scenario is the employment of drones used to acquire aerial images of urban regions that are difficult to access, such as slums. These drones may also run internal algorithms to process these images and trace significant operational patterns. They may also be able to decide to which C2 node(s) these images should be forwarded first.

High scalability and heterogeneity are also important features of IoBT. Military IoBT equipment can interact with civilian IoT devices, either to acquire data from the environment or even possibly to forward data. Thus, the overall network can become much larger and more heterogeneous.

### Software-Defined Networking

Computer networks expanded from a few nodes exchanging data in the early age of the Internet to a digital world where almost everything is interconnected. However, the vertical integration of data and control planes constrains the ability to introduce innovations to the current network model. For instance, the update of IPv4 to IPv6 is still incomplete after years of frustrating effort.

SDN was initially planned as a new paradigm that could add flexibility to wired networks by dividing the logical network into distinct planes — application, control, forwarding, and management — and defining a set of application programming interfaces (APIs) to exchange information between them. Furthermore, the SDN paradigm introduced a centralized management entity, called an SDN controller, to program the forwarding plane responding to requests from the application and management planes [5], [7].

These new features introduced by SDN provide for easier design, simpler testing strategy, and faster deployment of new protocols and technologies, as well as other systems and paradigms. ICN represents one of these paradigms that can benefit from these SDN features, as highlighted in [6].

### Information-Centric Networks

Although current IP networks use ports and hostnames for addressing and routing, most Internet traffic is concerned with content delivery, which is an important characteristic of IoT-based applications. Thus, network traffic is shifting from endpoint communications to a data-centric model, with a focus on "what" (content) rather than "where" (addresses). In this way, ICN is a novel network paradigm that changes the semantics of the network to enable data retrieval services based on named data objects (NDOs), which can be of two types: INTERESTs or DATA [4]. Consumers request data by sending INTEREST messages (with the named data to be retrieved), and producers (or intermediate nodes executing in-network caching and storage) return the matching DATA messages.

The cryptographic signature of DATA messages (name binding) puts security into the data itself and away from hosts and servers, contributing to the handling of security concerns (i.e., by providing data integrity). By decoupling the senders

Modern military operations are evolving the concepts of warfare. They range from conventional war scenarios, like Syria, to operations other-than-war, like the United Nations Peacekeeping missions in the Congo and Haiti, as well as the very specific Brazilian case of the Guarantee of Law and Order, where the army is sent to "pacify" regions of cities dominated by organized crime.

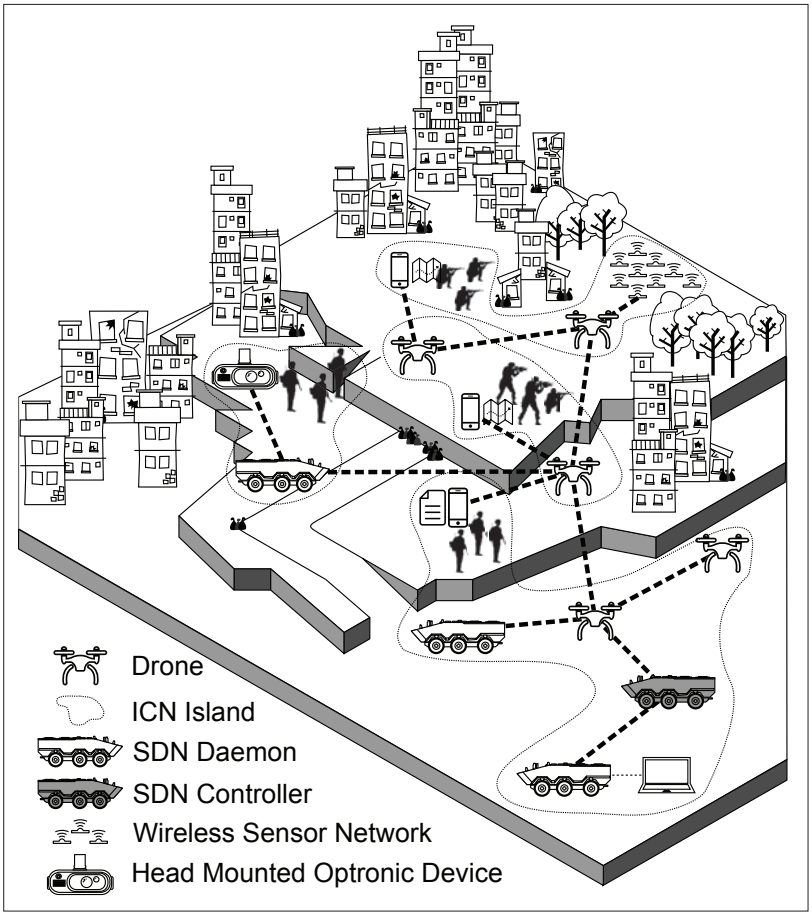



**Figure 1.** An operational scenario of military troops in an urban environment.

and receivers, ICN also provides intrinsic mobility, which appears ideal for ad hoc operational scenarios, and through caching and replication along the network, it is able to convey the data to the consumers, and thus enhance information distribution. These new features for security, mobility, and in-network caching make ICN a suitable enabler for IoT and edge networking [8].

## Related Work

Previous work [9–11] has applied SDN to BNs. While the latter paper focused on improving communication between resource-rich nodes (e.g., battleships and airplanes), the first two also targeted resource-constrained devices, while still adopting a conventional IP-based approach. ICN-specific designs, such as named data networking (NDN) and content-centric networking (CCN), are covered in [4]. Both adhere to the principles of a data-centric paradigm, but a large-scale ICN deployment, particularly in a BN, is far from reality. However, it is likely that ICN can be employed in conjunction with other IP-based networking technologies, such as SDN [6].

Another research study [12] has investigated the interdependencies of the C2 variables (ADR, PI, and DI) by using a simulated framework to test the technical and social layers of tactical networks. The simulations established baseline results in terms of bandwidth, information distribution, and the C2 approach. In this current work, on the other hand, an architecture is devised to provide the means to support C2 agility in the IoBT by using SDN to adjust the network parameters for node interaction and ICN to maximize the distribution of information.

## The Application Scenario

The armed forces are facing new operational challenges in complex urban scenarios, in which the adversary is hidden and difficult to locate in a widely dispersed, dynamic, heterogeneous, and cluttered environment. Figure 1 shows an example of this kind of setting in which there are military troops occupying a region dominated by criminal gangs, and where different networking elements cooperate. Military vehicles can only gain access to a part of this environment. After a given point in the terrain, only troops on foot are able to progress further and pursue criminals hidden in this maze-like environment. Since it is difficult to ensure safe progress in this environment, drones are often used to assist troops by providing aerial images that capture the movement of suspects on the ground. These places also generally border on urban forests, which can provide outlaws with a perfect escape route. In Fig. 1, a wireless sensor network (WSN) is shown that monitors these escape routes (on the upper right).

By using the different resources provided by multiple devices in the field, the operational commander can follow events as they unfold, and request video images from drones to the head-mounted optronic devices (HMODs) worn by soldiers. From his/her position inside an armored vehicle, as represented in Fig. 1, the commander can make informed decisions about the repositioning of the troops. However, due to the operational dynamics of the situation, the troops engaged in the front may also not be able to wait for these decisions. Thus, a certain degree of autonomy for local decisions is required, which, in turn, requires timely data/information support.

From the standpoint of the network, there is a need for rapid adaptation so that the authorization for receiving/sending data (C2 ADR dimension) can be implemented correctly. Moreover, the network must also provide reliable data for authorized nodes in a timely and efficient way. In the scenario under study, the soldiers may receive images from suspect activities in their surroundings, but not from locations that are far from them. In contrast, the commander receives data from all the sites. However, due to the dynamics of the operation, the network topology may change very fast, and it is possible that the nodes might become overloaded. These nodes can decide autonomously which data to forward, and which to discard or store locally. In a real situation where imminent contact is expected with hostile groups, it is more important for dismounted soldiers to receive images about the incoming threat than the commander. In this case, the network should first send the data to these soldiers and cache it for later delivery to the commander.

In this IoBT context, *things* may also require information to support their activities. For instance, the commander may send INTERESTs for images of the possible escape of suspects, and drones are the best candidates to produce this information. As intelligent things, drones will be able to determine the best places to fly, capture

the desired kind of images, and send INTERESTs for (human) movement detection. Upon detecting movement, the WSN will reply with DATA messages that match this second INTEREST and give information about its location. The nearest drone will head to that area, acquire the image, and respond with DATA messages that contain the image to the first INTEREST sent by the commander. Thus, before responding to a human order, drones require data from other nodes (WSN). This example requires the support of extreme patterns of interactions between humans and things (C2 PI dimension) at the network level.

Different types of data flows through this network, from short (a few bytes) text-based messages (e.g., WSN alarms) to very resource-demanding ones (e.g., drone videos). Unlike traditional IP networks, this scenario requires in-network caching to get DATA messages (especially videos/images) from neighbor nodes. By replicating data in multiple nodes, the network can protect itself from intermittent connections. These factors make the DI, one of the key C2 dimensions, more robust and agile.

## SDN-ICN Support for IoBT

The environment described in the application scenario is extremely dynamic, as mobile nodes are constantly entering and leaving the network. In addition, the IoBT must be able to cope with the problems of node heterogeneity: WSNs, armored vehicles, dismounted soldiers equipped with wearable and handheld network devices, and surveillance drones. These nodes must authenticate each other, communicate, self-adapt, and produce high-quality information to accomplish the mission by maintaining high levels of reliability and availability

ICN is a suitable means of meeting these requirements because of its caching mechanisms and data-centric approach. The naming service makes it easier to receive incoming data from dynamic nodes, since the consumer does not need to know the data producer. The caching mechanism keeps providing information even if the data producer becomes unreachable, as well as improving availability (in case of failure) and reducing latency (since the data is cached near the consumer).

In view of this, ICN is a promising network response to the IoBT requirements for information distribution. However, in the battlefield it is also necessary to handle a huge volume of data [1], avoid unnecessary data forwarding, and reduce channel overload. Likewise, some kind of mechanism that could define patterns of interaction between the network nodes would be helpful, such as node priority. For instance, if there is network congestion, the commander's vehicle could be assigned higher priority than subordinate vehicles or dismounts.

With regard to this, data filtering plays a very important role, both from an operational standpoint, to avoid overwhelming decision makers with a huge amount of information, and to maintain the health of the network. Filtering can be carried out through machine learning techniques and semantic patterns or analysis [1], or by analyzing node capacity in terms of data storage and bandwidth.

For example, wearable devices usually have limited storage and bandwidth, which makes it difficult for them to receive and cache large amounts of data. The SDN approach is an appropriate paradigm to solve this issue, enabling the use of data forwarding control conditions. If a DATA message matches an INTEREST from a constrained node, but it is too large to be handled (according to the pre-defined parameters set by the SDN controller), the ICN switches should not forward it. Hence, a mechanism for ICN-SDN interaction should be devised, where ICN would play a key role in the data plane and SDN in the control plane.

This work proposes a kind of dynamic mechanism that can take advantage of the information distribution features of ICN and use SDN to overcome some of ICN's shortcomings. The SDN controller is responsible for a set of nodes that is considered an ICN island, similar to the islands of nodes discussed in [9]. While ICN plays a major role in the data plane, SDN acts in the control plane and has two key roles: to act as a gateway between the ICN islands, and also between the island and the rest of the military IP network. In this role, SDN must convert ICN packets into IP and vice versa [6]. Each ICN island is represented by an aggregate of vehicles, dismounted soldiers (with HMOD and handheld devices), drones, and/or sensors, as shown in Fig. 1.

In the second role, the SDN controller establishes and controls the patterns of interaction of all the network nodes, by means of status INTEREST and DATA messages exchanged within the island, which are very small in comparison to the operational DATA messages and add little overhead. By receiving DATA messages with node status, the SDN controller is able to "sense" network degradation and filter large DATA messages before a congestion/interruption occurs, creating path optimization. In comparison, standard ICN uses negative acknowledgments (NACKs) to inform downstream nodes after a failure has occurred.

The controller also establishes a node hierarchy (based on the parameters received from the C2 application) and enacts security measures (e.g., node authentication and ICN-prefix encryption). These operational (node hierarchy and security) and network (path optimization) parameters are used to set up the ICN switches forwarding logic. In exchanging operational messages, ICN switches use this logic until a change is required by the SDN controller in response to operational demands or network circumstances. In theory, this integration should be smooth. However, a number of challenges must be overcome:

- The mapping of host-based rules into content-based rules
- The implementation of a protocol capable of handling ICN packets, establishing a network traffic policy and applying these rules to the entire network
- The modeling of content-based network nodes rather than address-based ones, where the optimization algorithms and forwarding/security policies can operate in different ways

To overcome these challenges, the architecture must consist of three main components: the C2 application, the SDN controller, and the ICN

The environment described in the application scenario is extremely dynamic, as mobile nodes are constantly entering and leaving the network. In addition, the IoBT must be able to cope with the problems of node heterogeneity: WSNs, armored vehicles, dismounted soldiers equipped with wearable and handheld network devices, and surveillance drones.

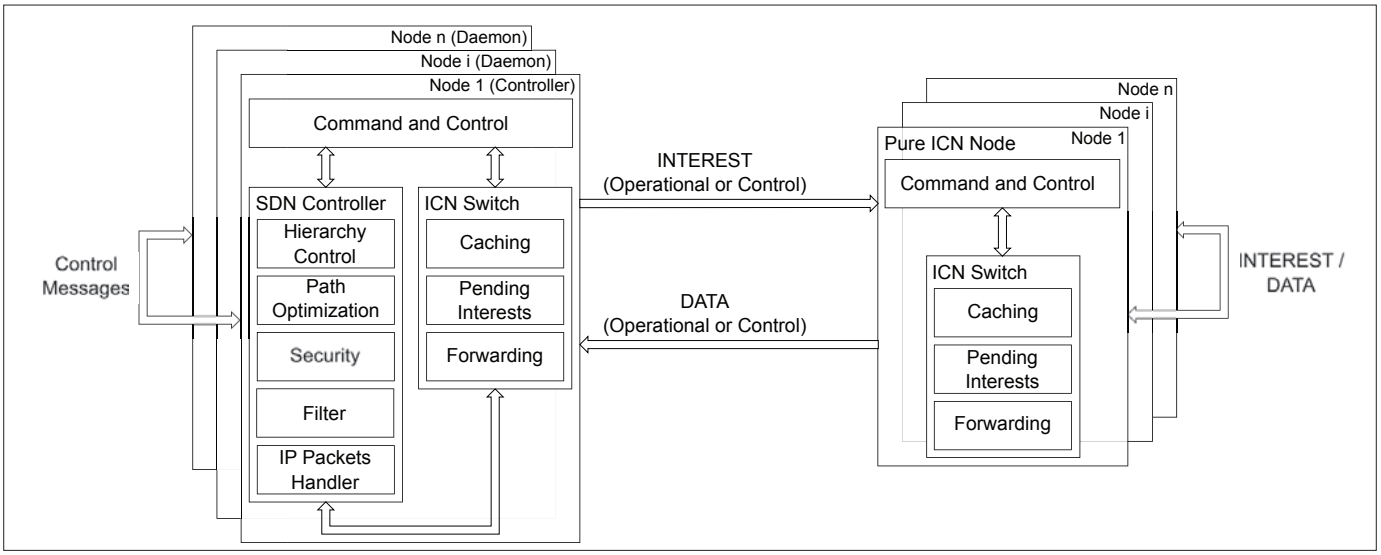



**Figure 2.** Proposed architecture for the SDN-ICN network nodes.

switches, as shown in Fig. 2. The C2 application is the highest-level component of the architecture. It is responsible for supplying the user with relevant and updated information and noting the user's aims/decisions after each cycle of the C2 process. Different C2 applications may run on different node types, depending on the node capacity and operational requirements.

The C2 application running in the commander's vehicle configures the initial mission. In this context, two key terms need to be defined.

**MISSION:** This is a planned activity designed to accomplish an operational goal. It establishes the resources required, the areas of interest (AoIs), as well as the node hierarchy and AoI priorities for receiving information. There might be more than one MISSION running simultaneously.

**TASK:** This is an ad hoc activity defined by the C2 application during the course of a MISSION to gather information or to send orders in response to a C2 decision process. The TASK should specify the AoI (a subset of the mission AoI), designate the node to carry it out, and define its priority.

The sequence diagram shown in Fig. 3 shows the interactions between the architecture components based on a two-step process in the control plane. The C2 application sets the initial MISSION parameters for the SDN controller (node relevance, area priority, etc.). The controller uses this information to set filter rules on the ICN switches, which initiate the INTEREST/DATA dissemination. If the mission parameters do not change, users continue receiving the information initially

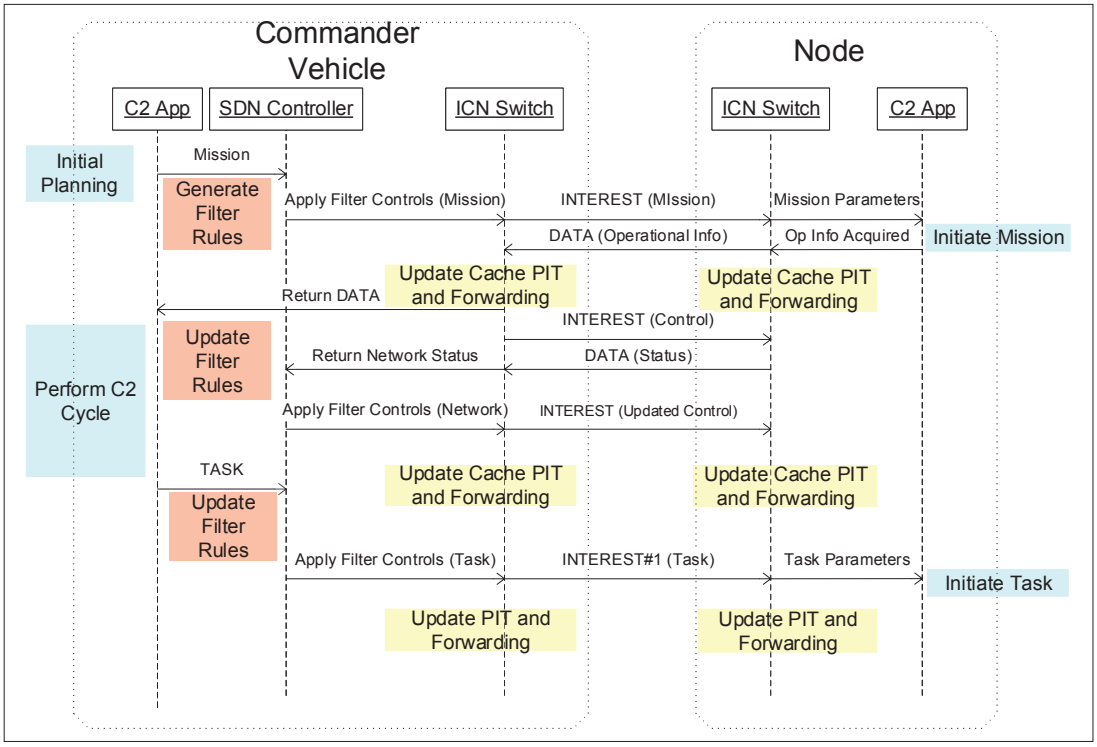



**Figure 3.** Interactions among architecture components.

requested. In the second step, The SDN controller acts on the ICN switches to adjust the data flows on the basis of the changing network parameters (node status: battery depletion, lower bandwidths, etc). The ICN switches then adjust their internal forwarding logic in accordance with these new filter rules. Finally, if the operational requirements change after the situational assessment, the user can adjust the mission parameters by sending TASKs to the controller, which will update the forwarding logic in the ICN switches and the C2 application in the specified nodes.

There is only one active SDN controller running in the commander's vehicle. The other vehicles run SDN daemons and follow the same principle outlined in [9]. Daemons act as secondary (inactive) controllers, and each can be chosen as the new controller if the primary one is unavailable. The controller consists of three main components.

**Hierarchy Control:** This sets the node hierarchy and message priority on the basis of the parameters received from the C2 application. It can filter unauthorized requests from lower hierarchy nodes as a security measure for better control of the network.

**Security:** This authenticates valid network nodes and encrypts ICN prefixes to avoid the possible risk of man-in-the-middle attacks.

**Path Optimization:** This adjusts the forwarding logic on the basis of SDN's global vision of the network by defining rules to forward packets through links with more available bandwidth. It also verifies multiple paths to the object and forwarding through the less expensive path, providing latency and bandwidth optimization.

The treated parameters are then sent to the filter component to be converted into INTEREST messages that are used to define the forwarding logic of all ICN switches in the controlled island. The ICN switches are responsible for data dissemination to all the nodes in the island. The forwarding logic is set according to special INTEREST messages (control messages) that are configured by the SDN controller. The pending interest table and cache have conventional ICN functions, but the storage capacity varies depending on the device type (e.g., vehicles have much more cache than soldiers).

## Applying the Proposed SDN-ICN Approach to the Studied Scenario

Two case studies have been undertaken to illustrate the application of the proposed approach. The first highlights some of the SDN features that are needed to configure the network for a desired C2 approach. The second complements the first by describing the use of ICN to support data dissemination. Figure 4 illustrates these two use case scenarios, and Table 1 provides a summary of the main parameters used in the performed simulations.

### Use Case #1: SDN Providing a Flexible C2 Approach

This case study examines the need for network flexibility to manage PIs, ADR, and DI (filtering data flows). As in the application scenario, the SDN controller can determine which node is allowed to send a given type of message. This feature is implemented by means of filters that only allow the flow of INTERESTs and DATA that match the specified parameters and characterize the chosen C2 approach. If they are changed, it will also alter the approach and make it either more centralized or decentralized, depending on the circumstances.

Apart from network configuration flexibility, SDN filters can also reduce data traffic by dropping unauthorized packets. The Mininet network simulator was used [13] to validate this capacity. The following criteria were defined for dropping the packets.

**Hierarchy Control:** Some devices do not have permission to request NDOs that contain information beyond the scope of the devices. Four node types were used: armored vehicles, soldiers, drones, and sensors, in this hierarchical order.

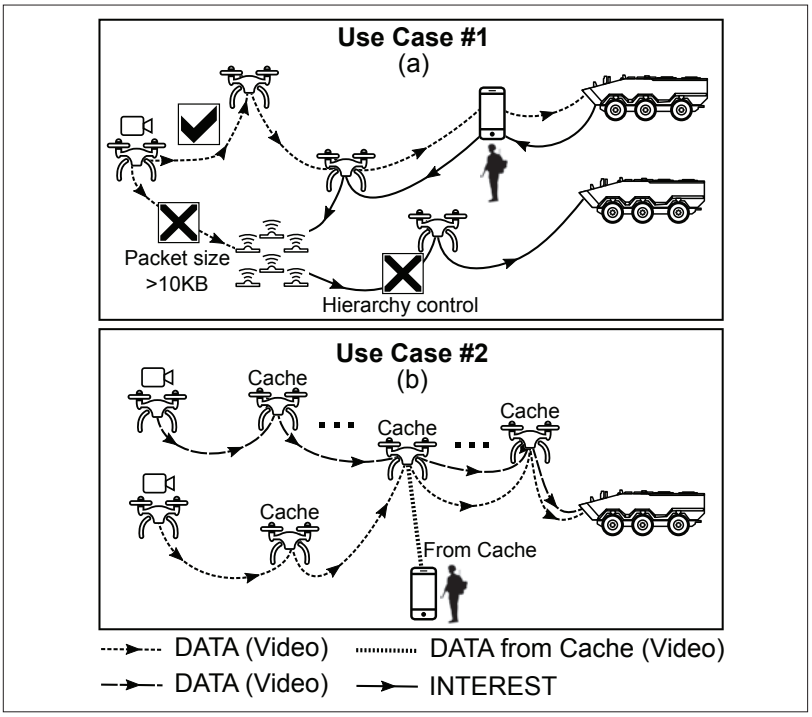



**Figure 4.** Illustration of case study scenarios where the SDN-ICN approach was applied.

| Parameters | Filtering use case #1 | Caching use case #2 |
|---|---|---|
| Number of nodes | 3 ground vehicles<br>5 drones<br>10 soldiers<br>20 sensor nodes | 1 ground vehicle<br>18 drones<br>1 soldier |
| Connection type | UDP | DATA/INTEREST |
| Data rate | Up to 54 MB/s | Up to 54 MB/s |
| Sensor maximum data size | 10 kB | – |
| Number of runs | 100 | 100 |
| Simulator | Mininet | Mininet with Mini-NDN |
| Controller | Floodlight | – |
| SDN protocol | OpenFlow 1.3 | – |
| Link delay | – | 5 ms |

**Table 1.** Parameters used in the experiments.

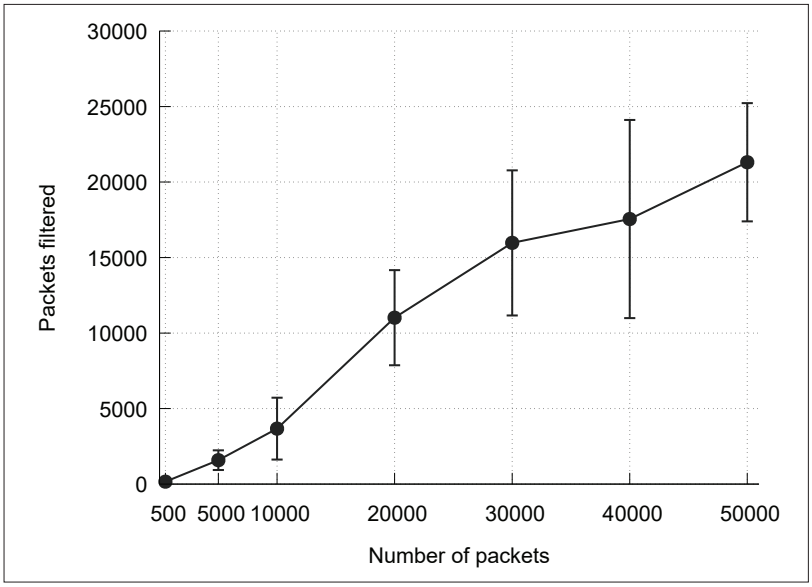



**Figure 5.** Number of packets filtered for different numbers of packets sent in the network (error bars: +/– 1 SD).

**Packet Size:** Some nodes have limited internal storage and can be easily overloaded with larger NDOs (e.g., video). This can be avoided by means of caching policies, but this would mean that the bandwidth would be spent on transporting data that would be later discarded. Instead, as SDN provides a network global vision, this information could be removed with less impact on the network.

Since there is no SDN standardized protocol such as OpenFlow for ICNs [6], this concept was simulated in a host-centric network in which each ICN node is represented by a switch-and-host setting. A node topology was created to implement the filters, and four lists were compiled with a view toward binding the ICN nodes to their hierarchy level. For instance, if a sensor made a request to a vehicle, it was immediately blocked by PACKET_IN processing. With ICN, this is easily done by binding some prefix with the hierarchical feature (e.g., /military nodes/vehicles/), and if any ICN node requests information from an NDO with a disallowed prefix, it is dropped, in a similar manner as presented in [14], which underlines the need for a possible extension of this study to address security issues. Figure 4a illustrates this scenario.

The experiment used UDP packets. Every IP host runs a UDP server that is waiting for requests. Random traffic was disseminated through the network, with an increasing number of packets. The parameters used in these experiments are shown in Table 1. The effects of the filtering can be seen in Fig. 5. As can be seen, the proposed mechanism scales, being able to filter out the unauthorized packets. This filters all packets it has to drop, with a mean rate of 42.21 percent of the total number of packets (31.13 percent due to size filtering and 11.08 percent due to hierarchy control), thus substantially reducing the amount of useless traffic. The high variation in the results reflects the wide range of randomly selected packets that were dropped. The results are positive, since they address the criteria for both hierarchy control and packet size (packets with a payload bigger than 10 kB are directed to avoid cache-constrained nodes).

## Use Case #2: ICN Enhancing Distribution of Information for a Given C2 Approach

When the commander sends INTERESTs for images of the possible escape of suspects, drones are triggered to capture videos and send a second INTEREST for these suspects' movements, which are replied to by DATA from the WSN. When a drone finally reaches the location indicated by the WSN, it starts sending videos to the requesting nodes. At this point, more than one node may be interested in the video, for example, the soldiers for an immediate reaction and the commander to monitor the operation. Thus, aside from the PI defined by the C2 approach, ICN also plays a significant role for the DI.

The goal is to evaluate the ability of the ICN caching mechanism to provide the necessary data available for follow-up requests (for the same data). The performance of the proposed architecture is assessed in terms of latency, and the results are compared to a non-ICN network. Table 1 summarizes the parameters used in the simulations that are used for the video parameters [15], while Fig. 4b illustrates the scenario from the moment the drones start sending the videos. The experiment used three NDOs that were produced and sent to the network, and all the nodes sent INTERESTs for them. The results achieved with the proposed ICN show a significant improvement in latency over non-ICN networks. For example, mean latency to access the requested data was 53.13 ms in the host-centric network but 34.56 ms with ICN, which represents a reduction of approximately 35 percent in latency.

## Conclusion

Modern battlefield scenarios require important changes in C2 activities, and advances in ICT are providing mechanisms for new military communications network designs. In view of this, IoBT is an important asset, but challenges must still be overcome to make it run smoothly and provide agility for C2.

By noting the advances made in ICN and SDN paradigms, this study combines both by exploring their best features to address the complexity of current military operations. The proposed architecture provides the means to map high-level C2 agility needs at the operational level of the network. However, despite the value of this proposal, there are still open issues that have to be addressed by future investigations, such as how to enhance security given the resource-constrained nodes of IoBT. Another important area for future work is how to integrate the SDN and ICN in terms of network control. There is also a need to address the issue of smooth SDN controller migration while including the data-centric paradigm.


### Acknowledgment

This work is partially supported by CAPES.


### References


[1] A. Kott, A. Swami, and B. J. West, "The Internet of Battle Things," *Computer*, vol. 49, no. 12, 2016, pp. 70–75.

[2] A. Kott and D. S. Alberts, "How Do You Command an Army of Intelligent Things?" *Computer*, vol. 50, no. 12, 2017, pp. 96–100.

[3] D. Alberts *et al.*, "SAS-085 Final Report on C2 Agility," NATO Research and Technology Organization, 2013; http://www.dodccrp.org/sas-085/
[4] B. Ahlgren *et al.*, "A Survey of Information-Centric Networking," *IEEE Commun. Mag.*, vol. 50, no. 7, July 2012, pp. 26–36.
[5] B. A. A. Nunes *et al.*, "A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks," *IEEE Commun. Surveys & Tutorials*, vol. 16, no. 3, 3rd qtr. 2014, pp. 1617–34.
[6] P. Zuraniewski *et al.*, "Facilitating ICN Deployment with an Extended Openflow Protocol," *Proc. 4th ACM Conf. Information-Centric Networking*, 2017, pp. 123–33.
[7] J. Wickboldt *et al.*, "Software-Defined Networking: Management Requirements and Challenges," *IEEE Commun. Mag.*, vol. 53, no. 1, Jan. 2015, pp. 278–85.
[8] E. Baccelli *et al.*, "Information Centric Networking in the IoT: Experiments with NDN in the Wild," *Proc. 1st ACM Conf. Information-Centric Networking*, 2014, pp. 77–86.
[9] I. Zacarias *et al.*, "Combining Software-Defined and Delay-Tolerant Approaches in Last-Mile Tactical Edge Networking," *IEEE Commun. Mag.*, vol. 55, no. 10, Oct. 2017, pp. 22–29.
[10] K. Poularakis, G. Iosifidis, and L. Tassiulas, "SDN-Enabled Tactical Ad Hoc Networks: Extending Programmable Control to the Edge," *IEEE Commun. Mag.*, vol. 56, no. 7, July 2018, pp. 132–38.
[11] J. Nobre *et al.*, "Toward Software-Defined Battlefield Networking," *IEEE Commun. Mag.*, vol. 54, no. 10, Oct. 2016, pp. 152–57.
[12] L. Scott *et al.*, "Exploring Dependencies of Networks of Multi-Genre Network Experiments," *Proc. 2016 IEEE MILCOM*, 2016, pp. 576–81.
[13] B. Lantz, B. Heller, and N. McKeown, "A Network in a Laptop: Rapid Prototyping for Software-Defined Networks," *Proc. 9th ACM SIGCOMM Wksp. Hot Topics in Networks*, no. 19. 2010, pp. 19:1–19:6.
[14] Q. Li *et al.*, "Mandatory Content Access Control for Privacy Protection In Information Centric Networks," *IEEE Trans. Depend. Sec. Comp.*, vol. 14, no. 5, 2017, pp. 494–506.
[15] J. Nightingale *et al.*, "Reliable Full Motion Video Services in Disadvantaged Tactical Radio Networks," *Proc. 2016 Int'l. Conf. Military Communications and Information Systems*, 2016, pp. 1–9.

## Biographies

Gabriel Martins Leal (gabrielmsleal@gmail.com) is an M.Sc. student in computer networks at the Federal University of Rio Grande do Sul (UFRGS). He achieved his Bachelor's degree from the Federal University of Goias, Brazil, 2016. Currently, his research interests are related to software-defined networks, ad hoc networks, and the Internet of Things.

Iulisloi Zacarias (iuli.zacarias@gmail.com) is a Ph.D. student at Halmstad University, Sweden. He received his M.Sc. degree in computer networks from UFRGS in 2018. He received his Bachelor's degree in information systems from the Federal University of Santa Maria, Brazil, 2016. His current research interests are wireless networks, SDN, and IoT.

Jorgito Matiuzzi Stocchero (jstocche@gmail.com) has an M.Sc. degree in electrical engineering from COPPE/UFRJ (2004), an M.B.A. in politics and strategy from FGV/RJ (2016), a postgraduate degree in military sciences from ECEME (2016 and 2012), a communication engineering degree from IME (1996), and a Bachelor's degree in military sciences from AMAN (1990). Currently, he is working on the development of scientific cooperation between the Brazilian Army and UFRGS.

Edison Pignaton Freitas (epfreitas@inf.ufrgs.br) has a Ph.D. in computer science and engineering from Halmstad University (2011), an M.Sc. in computer science from UFRGS (2007), and a computer engineering degree from the Military Institute of Engineering (2003). Currently he holds an associate professor position at UFRGS, developing research mainly in the following areas: computer networks, real-time systems, unmanned aerial vehicles, IoT, and smart (autonomous) systems.