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

JORGITO MATIUZZI STOCCHERO

A Network Centric Architecture for Military Command and Control Systems

Thesis presented as a partial requirement for obtaining the degree of Doctor of Computer Science.

Advisor: Prof. Dr. Edison Pignaton de Freitas

Porto Alegre

2023

CIP – CATALOGING-IN-PUBLICATION

Stocchero, Jorgito Matiuzzi
A Network Centric Architecture for Military Command and Control Systems [manuscript] / Jorgito Matiuzzi Stocchero. – 2023.
115 f.:il.
Orientador: Edison Pignaton de Freitas.
Tese (Doutorado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação. Porto Alegre, BR – RS, 2023.
Network Centric Operations. 2. Software Defined Networks. 3. Information Centric Networks. 4. Delay Tolerant Networks. I. Freitas, Edson Pignaton. II. Título.

AGRADECIMENTOS

A minha esposa, Cíntia, agradeço especialmente pela motivação, suporte e paciência durante esses anos em que perseveramos juntos nos meus estudos de doutorado. Teu apoio foi fundamental para que eu conseguisse chegar até aqui, gerenciando a nossa família para que me dedicasse na tese. Contei também com a ajuda e compreensão dos meus filhos, Diego e Bruna, minha família que tornou essa jornada possível. Sem vocês nada disso valeria a pena.

Aos meus pais, Roque e Nelci, pelo amor e carinho, especialmente nos momentos de dificuldades. A minha amada avó, Araci, que nos deixou ano passado, agradeço pelo amor incondicional e pelo exemplo de pessoa maravilhosa, a quem dedico essa tese *in memorian*.

Aos meus amigos e colegas Iulisloi Zacarias, Lauro de Souza Silva, Carlos André Silva, Márcio Antônio Lawisch, Andre Dexheimer Carneiro, que participaram de partes dessa tese com ideias e apoio para a superação dos desafios, incluindo a co-autoria de artigos. Muito obrigado.

Ao professor e orientador Edison Pignaton de Freitas, agradeço pelos conselhos e orientações oportunas durante esse trabalho. O professor Pignaton foi muito mais que um orientador, mas um amigo cujo suporte foi o grande habilitador desse trabalho. Agradeço também aos professores Luciano Gaspary, Juliano Wickboldt e Julio César S. dos Anjos pelas contribuições oportunas.

Enfim, agradeço a Deus, por tudo.

ABSTRACT

Current multi-domain military operations require flexible and adaptable forces that must be able to work with civilian agencies and coalitions in open terrain and urban scenarios. These challenges require informed decision making not only at headquarters but also at the edge of the battlefield, along with network-enabled capabilities and agile command and control (C2) systems. The C2 system should ensure the timely and secure exchange of information originating from networked devices such as radars, unmanned aerial vehicles (UAVs), sensors, armored fighting vehicles, and wearable devices. These networked sensors and effectors form the Internet of Battle Things (IoBT). In this work, a network strategy to achieve C2 agility in the heterogeneous IoBT environment by coupling the application and network in a multi-layer architecture is investigated. The proposed architecture leverages the Software-Defined Networking (SDN) paradigm to orchestrate network services using semantics-oriented data from technologies such as Information Centric Networks (ICN) and Delay-Tolerant Networks (DTN), and applies security mechanisms at the hardware, network, and application layers. Results from simulated scenarios show significant improvement in network metrics as well as network resilience to cyberattacks.

Uma Arquitetura Centrada em Redes para Sistemas de Comando e Controle Militares

RESUMO

As operações militares multi-domínio requerem tropas flexíveis e adaptáveis, que devem ser capazes de cooperar com agências civis e coalizões, tanto em terreno aberto como em cenários urbanos. Tais desafios requerem decisões informadas não apenas nos quartéis generais, mas também na frente do campo de batalha, combinadas com capacidades centradas em rede e sistemas de Comando e Controle (C2) ágeis. O Sistema de C2 deve garantir a disseminação de informação de forma oportuna e segura, obtida de dispositivos interconectados, como radares, veículos aéreos não tripulados, sensores, veículos de combate blindados e dispositivos "vestíveis". Esses sensores e atuadores interconectados constituem a Internet das Coisas do Campo de Batalha. Este trabalho investiga uma estratégia de rede para obter agilidade de C2 para esse ambiente heterogêneo, acoplando a aplicação à rede em uma arquitetura com múltiplas camadas. A arquitetura concebida utiliza o paradigma de Redes Definidas por Software (SDN) para a orquestração dos serviços de rede, por meio de tecnologias orientadas a dados, como Redes Centradas em Informação (ICN) e Redes Tolerantes a Retardo (DTN), usando mecanismos de segurança nos níveis de hardware, rede e aplicação. Resultados preliminares mostraram uma melhoria significativa nas métricas de rede, bem como resiliência para suportar ataques cibernéticos.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

AI	Artificial Intelligence
AoI	Area of Interest
ADR	Allocation of Decision Rights
API	Application Program Interface
BFT	Blue Force Tracking
BN	Battlefield Network
C2	Command and Control
CCN	Content Centric Networks
COP	Common operational Picture
COTS	commercial off-the-shelf
CS	Cache Storage
DI	Distribution of Information
DoS	Denial of Service
DTN	Delay Tolerant Networks
FANETS	Flying Ad Hoc Networks
FIB	Forward Information Base
GIS	Geographical Information System
HMOD	Head Mounted Optronic Device
ICN	Information Centric Networks
IoBT	Internet of Battle Things
IoT	Internet of Things
IPN	Interplanetary Networks
IRTF	Internet Research Task Force
MANETS	Mobile Ad Hoc Networks
MBN	multi-bearer network
MDO	Multi Domain Operations
MEC	Mobile Edge Cloud
NATO	North Atlantic Treaty Organization
NCW	Network Centric Warfare
NDN	Name Data Network
NDO	Named Data Object

ONOS	Open Network Operating System
OODA	Observe Orient Decide Act Loop
OOTW	Operations Other Than War
PI	Patterns of Interaction
PIT	Pending Interest Table
PSO	Particle Swarm Optimization
PUF	Physical Unclonable Function
QoS	Quality of Service
REE	Rich Execution Environment
RTT	Round Trip Time
SDN	Software Defined Networks
SDR	Software Defined Radio
SPoF	Single Point of Failure
TEE	Trusted Execution Environment
TEN	Tactical Edge Networks
TTL	Time to Live
UAV	Unmanned Air Vehicle, sometimes referred as drone
VANETS	Vehicular Ad Hoc Networks
WSN	Wireless Sensor Networks

SUMMARY

1 INTRODUCTION

Armed conflict has changed significantly over time due to societal change and technological advances in conducting military operations. The battlefield is where military operations take place. It is a complex system in which various combat functions perform a range of activities to achieve desired objectives: Movement and Maneuver, Reconnaissance, Fire, Protection, Logistics and Command and Control (C2) (BRASIL. EXÉRCITO BRASILEIRO., 2017).

C2 is the combat function responsible for planning, directing, coordinating, and controlling military operations and linking other combat functions at multiple levels. It brings together many disciplines (theory, philosophy, systems, risk, technology, and intuition) and is a major area of research in military science that is guiding the transformation of armed forces. Network Centric Warfare (NCW) (ALBERTS, 2011) is a key component of military transformation and relies on computer equipment and networked communications technology to create a shared awareness. It is supported at the technical level for a battlefield network (BN) whose technological solution must keep pace with research trends in military science.

Current battlefield networks must be flexible to adapt to the ever-changing conditions on the battlefield: the lack of infrastructure, heterogeneous equipment, dynamic topology, and the inherent chaos of combat scenarios. Because they must ensure operational situational awareness to support decision making, net-centric C2 systems must share tactical information in real time between soldiers and commanders, which is critical for future multi-domain operations (MDO) (FEICKERT, 2021), where military activities permeate all domains of the battlefield: land, sea, air, space, and cyber.

The ongoing war between Russia and Ukraine (GILLIAM; VAN WIE, 2022) (PETIT, 2022) is an important example. Despite the Russian army's formidable firepower, it failed to achieve its tactical objectives in the early months of the war. Russian decision-making has become difficult due to the Russian Army's inability to synchronize its actions across domains. The slow *tempo* of the invasion is indicative of the deficient Russian command and control structure. Transmissions over unsecured lines, the lack of a networked communications infrastructure, and over reliance on officers have revealed problems with tactical C2. The Ukrainians have been adept at targeting Russian generals (the high number of generals killed in this war is rare) and lower-ranking officers to create chaos at the tactical level.

Another example is the 2020 war in Nagorno-Karabakh between Armenia and Azerbaijan (PETIT, 2022), which is considered the first war in which drones were used extensively on the battlefield. The modernized Azerbaijani forces had new sensors that, in concert with legacy systems, enabled effective and lethal targeting of Armenian formations. Armenian command and control systems were often ineffective, which contributed to defeat.

These examples demonstrate the importance of a net-centric C2 system. Large-scale combat operations require disciplined, well-trained tactical leaders who can operate in a dispersed environment without explicit direction from their chain of command. Battlefield networks must address these challenges at the routing and QoS levels to provide robust and secure data network services suitable for this new battlefield environment. The network service must meet the characteristics, requirements, and constraints of both resource-rich nodes, such as C2 centers served by high-capacity strategic networks, and resource-constrained nodes at the edge of the tactical network with low (and fluctuating) capacity.

C2 center operators must have battle planning and command and control capabilities supported by shared situational awareness across all domains (air, land, sea, space, and cyber), including updating resource locations, intelligence information, warnings, and so on. They are at the heart of the Combat Cloud (KISER *et al.*, 2017), whose continuity of operations depends on a distributed system architecture with no single point of failure (SPoF). The Combat Cloud must disseminate relevant information in a secure manner, with full sensor connectivity from the edge of the network. Peer-to-peer C2 applications can be distributed across the network, from elastic applications such as file transfer and text messaging to real-time non-elastic applications such as video streaming, as well as a more efficient way of scheduling tasks (PENG; LU; WU, 2021) (PENG; LU; WU, 2021).

The solutions currently used by the industry do not fully meet the requirements of these applications. Technology is advancing and bringing higher capacity networks to the edge, but at the same time, the need for data and video transmission is increasing. In this situation, distances between nodes, topography, and radio frequency (RF) characteristics pose additional network capacity challenges that cannot be efficiently addressed with current (non-flexible) deployments. Heterogeneity is another critical factor, as network segments may have different performance levels. Routers need to be aware of radio capacity in order to differentiate application traffic according to its capacity.

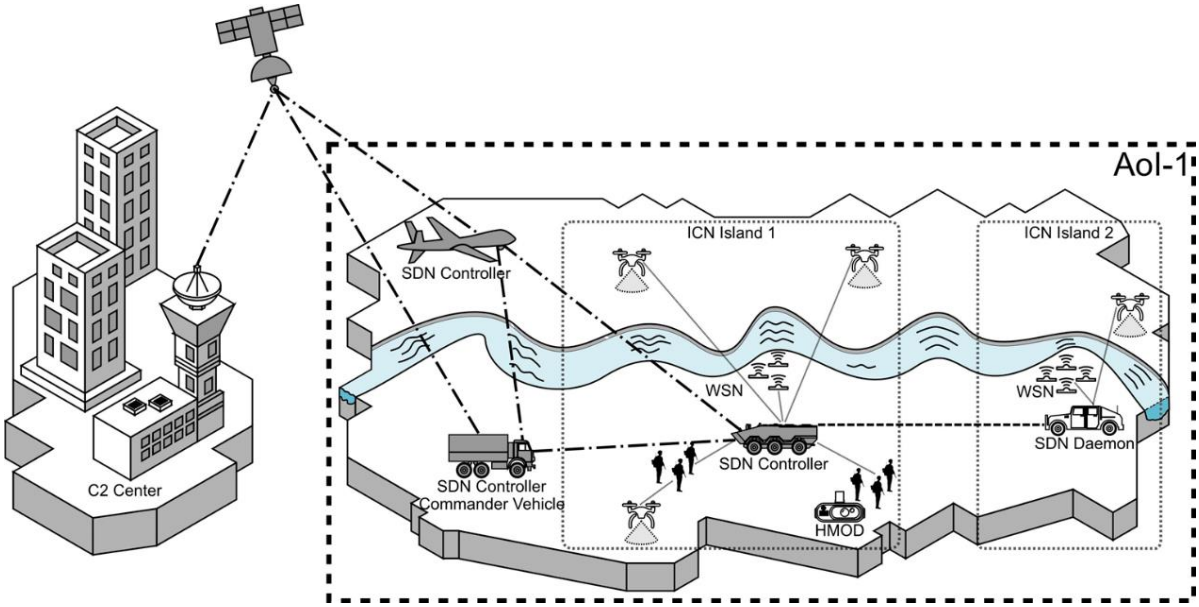
To achieve the levels of performance and efficiency critical to effective C2 systems under the NCW concept, C2 applications could benefit from the integration of new network technologies into edge computing. (ZENG *et al.*, 2021) suggested that these new technologies

move cloud computing and services to the edge of the network, creating a flexible network that can handle limited and fluctuating node capacity and heterogeneity while protecting the network from hostile attacks. These networked edge nodes include wireless sensors deployed in the field, radios carried by special operations forces, drones for image capture, and other applications that form an **Internet of Battle Things (IoBT)**, a type of network in which people interact with "things" to achieve operational goals (KOTT; ALBERTS, 2017) and (RUSSELL; ABDELZAHER, 2018).

Like its civilian counterpart (i.e., the IoT), the IoBT is a network of physical devices for collecting and/or processing data that can be exchanged over the network. The information generated enables the commander to take full advantage of the MDO concept by quickly finding and switching between different options (in multiple domains) to address a particular challenge.

An example scenario is shown in *Figure 1*, which is examined in detail in Section 4.1 of this thesis.

Figure 1 - IoBT Operational Scenario Example



Source: (STOCCHERO *et al.*, 2023)

In such a scenario, forces dispersed in the Area of Interest (AoI) form the IoBT, in which a local commander assigns missions and tasks to his resources (people or things) according to instructions from a C2 headquarters far from the AoI. The dynamics of this environment and

the actions of the enemy require quick decisions on the ground, as there is no time to follow the entire chain of command.

The local commander or even a soldier must have decision-making authority; even things like drones require some degree of autonomy to perform their tasks and interact with other things (other drones and sensors) and with foot troops. The IoBT must support different patterns of interaction and reconfigure the flow of data as circumstances and priorities change. For example, in the decision phase, the commander has the highest priority for receiving video from drones. In the execution phase, the acting troops receive the highest priority.

To enable this decision-making process, an efficient C2 approach must be in place, supported by an information dissemination mechanism to ensure that the right actor receives the needed information at the right time to make informed decisions. Therefore, the network must be flexible to prioritize data flows based on circumstances and change traffic priorities and node hierarchy as needed. Rapid decision-making is a critical differentiator in MDO, especially when hostile forces are attacking a specific target. This underscores the value of the ***C2 agility*** concept, as illustrated by the example above from the Russia-Ukraine war.

A North Atlantic Treaty Organization (NATO) Task Group has produced a detailed report (T. G. SAS-085, 2014) outlining a conceptual model for C2 agility, defined as the ability to select procedures and adapt the C2 approach during military operations. The report categorizes C2 strategies in the following ways: how decision rights are assigned, how different actors interact, and how information is disseminated. Thus, C2 approaches range from highly centralized hierarchies to loosely coupled networks.

Achieving C2 agility in the context of IoBT is challenging because, as mentioned earlier, the currently deployed C2 systems have many technological shortcomings. From a network perspective, C2 agility presents challenges to the information and communications infrastructure in terms of latency and throughput, as well as network configuration and control. Current military C2 implementations, mostly based on vendor-specific top-down systems with conventional IP routing, lack flexibility and need to improve the way available resources are used depending on the mission and circumstances (RUSSELL; ABDELZAHER; SURI, 2019) and (STEPHEN J. TOWNSEND; ARMY, 2018).

To overcome these challenges, one must be able to make dynamic changes in three variables (**Allocation of Decision Rights - ADR**, **Patterns of Interaction - PI** and **Distribution of Information - DI**) in response to changes that occur in the execution of the operation. These variables are examined in detail in chapter 2.

Data semantics-oriented technologies such as information-centric networks (ICN), delay-tolerant networks (DTN), and software-defined networks (SDN) can play a key role as the foundation for such system implementations.

SDN has been used as a means to provide a flexible control plane for dynamic networks such as BN (POULARAKIS; IOSIFIDIS; TASSIULAS, 2018), while using a traditional IP-based paradigm. It orchestrates network services, facilitates end-to-end interactions, defines node hierarchy and permissions, and supports the use of edge computing resources to process the vast amounts of data generated on the battlefield. By processing data in close proximity, SDN avoids unnecessary routing of data and minimizes channel congestion. SDN also facilitates route reconfiguration to match application requirements with network capacity.

However, SDN is not able to efficiently distribute data to relevant actors in the network and deal with disruptions or interrupted communication channels. To address these gaps, ICN and DTN can be added as network services to the SDN orchestrator. DTN deals with interruptions by temporarily storing data in intermediate nodes and forwarding it to the destination when opportunistic connections occur (FALL; FARRELL, 2008). This feature is extremely useful for nodes with high mobility that can connect to and disconnect from different network segments (islands), such as the drones shown in *Figure 1*.

The ICN paradigm can provide an efficient mechanism for data distribution by shifting network logic from the traditional address-based (host-centric) approach to a network architecture that is information-centric (data-centric). The flexibility inherent in the ICN forwarding strategy is suited for service discovery in highly dynamic scenarios (ZHANG *et al.*, 2021). In the scenario of *Figure 1*, ICN uses named data to enable efficient matching between data producers and consumers so that video and sensor data are quickly available to commanders and soldiers regardless of data source or location, reducing end-to-end delays.

1.1 Research Questions

With these novel network paradigms, commanders can execute the decision cycle at multiple levels in a way that is self-adaptive to the environment and operational changes across multiple applications, and maintain a high level of security for data transmissions.

To enable such a network solution, the following research questions are raised:

- How to design a network architecture solution to handle the high-level variables of network-centric warfare (ADI, PI, and DI)?

- How can network services for battlefield networks be orchestrated to establish and modify interactions between nodes and dynamically change the assignment of decision rights?
- Is it beneficial to use different network paradigms simultaneously to deal with the common problems of disruptions or interrupted communication channels and data dissemination in C2 networks?

1.2 Objectives, Methodology and Contributions

Considering the research questions, the main objective of this thesis is to propose a network architecture solution to achieve secure C2 agility in the IoBT by connecting the application to the network through orchestrated network services, while respecting the principles of NCW. SDN orchestrates data flows using different paradigms:

- DTN is used to support intermittent links, such as communication between UAVs and wireless sensor networks (WSNs), which occurs only when a UAV is flying in the communication range of the WSN, or to connect different ICN islands;
- ICN is used for content distribution, delivering valuable tactical information (on threats, for instance) to all authorized nodes, regardless of the source; and
- IP is used whenever the source-destination is relevant, such as voice communication between soldiers, as well as to support legacy systems.

The methodology chosen to achieve the objective is:

- The study and development of a network solution to deliver higher-level user requirements for NCW, enabling C2 agility in tactical edge networks (TEN);
- The development of an application specific use case of ICN/DTN over SDN for wireless Mobile Ad Hoc Networks (MANETs) for military C2 applications.

Information distribution leverages the ICN opportunistic caching mechanism, while intermittent connections among different network segments employ DTN. At the same time, network health is maintained by an SDN controller, which filters non-relevant data and establishes priorities. The SDN Controller tunes the ICN forwarding mechanism according to the C^2 application requirements and network current capacity, establishing a QoS mechanism. It also integrates ICN with the rest of the IP BN.

Some expected contributions of this research are:

- A review of networked-centric military operations, mapping possible network solutions to enable C2 agility, as presented in *Table 1*;
- The establishment of an architectural network solution to support network-centric military operations, by coupling tactical applications with network services protected by an efficient cyber-security mechanism;
- The development of network services that combine DTN, ICN, and SDN technologies providing a suitable quality of service, regulating the flow of traffic and improving effective bandwidth;
- The design of an example C2 tactical application that leverages from the provided network services.

1.3 Outline of the Thesis Organization

Chapter 2 focuses on examining the current state of military science research with an emphasis on command and control in support of NCW and MDO. The concepts of C2 agility and IoBT are discussed and linked to potential network solutions, including security considerations.

Chapter 3 provides an overview of the network paradigms that served as the basis for this thesis, focusing primarily on their military applications presented in Chapter 2. DTN is examined in terms of its advantages for intermittent links demonstrated in preliminary work. ICN is presented as a main candidate for data distribution in BN, with promising results. Finally, SDN is analyzed as a core component that provides the required flexibility and programmability to meet the current requirements for tactical networks.

Chapter 4 presents the architecture as a solution to the concepts and organizations explained in Chapter 2 using the network paradigms discussed in Chapter 3. Examples of operational scenarios and practical deployments are described to validate the operational requirements and better understand the benefits. The architecture is described in detail in all dimensions: the application, orchestration and cybersecurity dimensions are described and discussed.

Chapter 5 presents an implementation of the architecture with selected specific technologies as a proof of concept. To evaluate the feasibility and appropriateness of the concept, Chapter 6 describes experiments conducted in various published works with promising results and Chapter 7 compares them to other research. Finally, Chapter 8 presents the conclusions and future work.

2 NETWORK CENTRIC COMMAND AND CONTROL

Modern military operations range from typical war scenarios to operations other-than-war (OOTW), conducted simultaneously in multiple domains. In both situations (shown in *Figure 1* and *Figure 6*), soldiers must work with civilians from governmental and nongovernmental organizations in urban and rural, with adversaries and noncombatants present simultaneously. Commanders must deal with uncertainty and increasingly complex social relationships among the various actors on the battlefield on land, in the air, at sea, in space, and in cyberspace, which increases the importance of command and control.

(KOTT; ALBERTS, 2017) defined command and control as the management of military organizations and endeavors and identified five functions (command, control, sense-making, execution, and situation monitoring) necessary to achieve desired effects on the battlefield. Command establishes the intent (objectives and priorities) and creates the conditions (rules of engagement and other constraints) for the operation to succeed. The intent is a comprehensive description of 'success". It defines what the enemy and friendly forces should look like at the end of a battle. An intent statement encourages initiative and creativity and protects against failure and confusion when things do not go as planned. A statement of intent might read, "*No enemy soldiers will cross the river*," "*Fight hard, but surrender before you are overrun*".

Control includes all other functions repeated in cycles. The faster the control cycle, the more effective and flexible the C2 system, which is supported by efficient computer and network architectures. Sense making (collecting, processing, and sharing information) increasingly uses smart things (sensors and actuators) to generate valuable information for the multidomain C2 system, which according to (RUSSELL; ABDELZAHER, 2018) is the brain of modern warfare. They form the **Internet of Battle Things (IoBT),** which are the eyes and ears. The troops are the hands and feet.

The goal of the C2 system is to connect the eyes, ears, hands and feet inseparably to the brain. To do this, it must integrate these intelligent and heterogeneous things with the human actors on the battlefield and ensure that the resulting ensemble is capable of achieving the objectives and priorities of the operation as defined by the command and supporting the decision cycle, seamlessly integrating the IoBT into the Combat Cloud.

The United States Air Force defined the Combat Cloud as a "*meshed network for data distribution and information sharing within a battlespace, where each authorized user, platform, or node transparently contributes and receives essential information and is able to utilize it across the full range of military operations*" (KISER *et al.*, 2017). The information

generated is multi-domain and was created from the ground up by IoBTs in all domains. This allows the commander to take full advantage of the MDO by quickly generating and switching between multiple solutions to a given challenge.

In the remainder of this chapter, the concepts of Internet of Battle Things, multi-domain operations, and C2 agility are explored to explain the design decisions for the architecture.

2.1 Internet of Battle Things (IoBT)

Like its civilian counterpart (i.e., the IoT), the IoBT is a network of physical devices, vehicles and sensors to collect and/or process data that can be shared with other nodes in the network. On the battlefield, these objects are military equipment and vehicles such as weapons, optronic devices, drones, wearable health monitoring systems, sensors and others. They are capable of providing a variety of data on different events of interest to support informed decision-making, such as enemy movements or the health status of soldiers. For example, in the tactical scenario shown in *Figure 1*, all units in this battle area must be interconnected, so there is no single sensor shooter, but rather a network of sensors and effectors.

When a sensor detects a new threat (an enemy armored vehicle), the deployed C2 system must enable the commander to make a decision and engage that specific threat. After receiving the available options from his resource pool, he could select a drone (already in the vicinity of the threat) as a sensor and a friendly armored vehicle as an effector to neutralize the threat. The drone would provide the commander with real-time video of the threat and determine the location of the target for firing by the effector. Depending on the battle situation and changes to the situation received from other sources, the commander should be able to change this assignment and redirect other assets to engage this or other incoming threats.

To enable all of these activities and maintain the information flow required for this example, the system software and network must be reliable, maintainable, secure, and scalable. A reliable tactical data distribution mechanism is needed for updating the location of assets, for messages and alerts, and for planning and assigning missions and tasks. It must provide security through authentication/authorization mechanisms and dynamic authorization management.

IoBT networks consist of heterogeneous devices that can connect to more than one communication network depending on their hardware capabilities. This requires a multilayer network model capable of accommodating the connectivity of such devices at scale. "Things" in this context are generally small control and actuation devices that provide sensory

information and actuation capabilities, rather than systems that are more commonly thought of as "platforms" such as vehicles, power grids, etc.

Node heterogeneity requires the network to understand the behavior of each device type and differentiate data flows according to node capacity to avoid bottlenecks. Combat efficiency and coordinated decision making in war scenarios depend heavily on this data collection for a larger C2 system where social interactions are paramount to reasoning and decision-making. For this to occur, reliable connectivity and a resilient and flexible information dissemination scheme are essential, as there are adversaries who seek to disrupt and distort the process.

Related work (RUSSELL; ABDELZAHER, 2018) has explored some of the requirements for such a system:

1) navigate networks that are simultaneously red (hostile), blue (friendly), and gray (noncombatant);

2) integrate mission-relevant sensing and information capabilities from multiple external sources, such as allies or opportunistically cooperating units; and

3) move away from information models that require a complete understanding of the information space to models that are capable of probing, verifying, and aggregating information.

All these requirements force IoBT to deal with uncertain quality and trustworthiness as it receives information from multiple and heterogeneous sources, volatile connectivity/availability, and potential compromise by attackers.

To address these challenges, another work (FAROOQ; MEMBER; ZHU, 2018) conducted a theoretical study on the design of secure and reconfigurable IoBT networks. Using the theories of stochastic geometry and mathematical epidemiology, the work quantified the information dissemination among the heterogeneous devices of IoBT and optimized the network parameters. Their framework takes into account the perceived threat level from the adversary and the cost of combat equipment. It enables reconfiguration of the network according to periodic assessment of lost connectivity or changes in security requirements to achieve desired mission objectives. The results show that mission objectives can be achieved by changing the deployment density of the nodes or by changing their transmission power, or both, in response to changing mission requirements.

Alternatively, this work proposes a multilayer architecture for the IoBT at the edge of the combat cloud, using its collective behavior to achieve C2 agility and ensure situational awareness. Diverse C2 applications running in the heterogeneous IoBT nodes benefit from a novel network scheme for efficient data dissemination. Communications are insensitive to

disconnects, interruptions, and latency, turning the C2 system into a network orchestrator that can respond to operational issues and evaluate courses of action for the decision-making process. Real-time tactical information such as Blue Force Tracking (BFT) and enemy threats is automatically available to decision makers via the underlying network, along with a cybersecurity layer that addresses the challenges the IoBT faces from adversaries acting against the system.

2.1.1 IoBT Security Aspects

The above-mentioned peculiarities of IoBT pose challenges for network and application design to ensure security. Operating conditions with varying ownership and without appropriate network and node infrastructure (computing power, information storage, and battery resources) in the presence of adversaries require novel security techniques to establish trust and make the system resilient. Proximity to the enemy creates opportunities for sabotage, misuse of captured assets, or information theft, which compromises intelligence collection and planned operations.

A related paper (AGADAKOS *et al.*, 2019) asserts that reliance on potentially untrustworthy IoBT resources can affect force confidence in their ability to accomplish missions. The dynamic and adversarial nature of the battlefield prevents a predetermined framework for trustworthy interactions. Therefore, IoBT concepts must implement secure mechanisms for distributed computing to protect and defend their resources from attackers. Network services must handle sensitive data from distributed assets with potentially varying levels of trustworthiness, creating a tension between confidentiality and performance. Another important challenge for IoBT is adversary deception, where fake information can be generated to mask the nature of military activities, intentions, and missions.

Another work (TOSH *et al.*, 2018) also explored the challenges of building trust in IoBT to increase the credibility of the information collected. The authors proposed the use of blockchain technology as a solution to build trust between heterogeneous IoBT nodes. A distributed ledger system enables traceability of historical information, guarantee of data provenance, guaranteed variability of integrity violations of historical data, and detection of information tampering. These features make the technology suitable for cyber operations, especially in the IoBT context, where a distributed trust and identity management mechanism is critical for identifying, authenticating, and authorizing assets without a mediating entity.

The authors propose a three-layer architecture: a "Battlefield Sensing layer" where sensors collect and disseminate information; a "Network layer" dynamically created between

capable IoBT nodes to process blockchain-related operations; and at the top, the "Consensus and Service layer" that defines individual roles and mechanisms to maintain consistency.

Related Work in (LIOU; LIN, 2021) combined blockchain with Physical Unclonable Functions (PUFs) to build a trust architecture. PUFs are a hardware-based device fingerprint technology that can significantly improve the level of security compared to hard-coded passwords. A PUF is based on subtle physical differences within a tolerable range in the semiconductor manufacturing process and generates private keys that cannot be used for impersonation or theft, giving each device a unique identity.

In their mechanism, devices are authenticated based on smart contracts (programs running on the blockchain) rather than a centralized server, avoiding SPoF and tampering and providing design flexibility and operational security. The mechanism supports cross-service group authentication, which is essential for authenticating assets from different groups, and is effective against a range of attacks such as man-in-the-middle, impersonation, secret attempting, and blockchain tampering.

2.1.2 Attack vectors

These are just examples of cyberattacks on the IoBT. For a multi-layer SDN architecture, there are several attack options that target the vulnerabilities of each layer, as studied by (J. C. NIKOUE, 2019).

For the application layer, attackers can exploit access control vulnerabilities, such as lack of authentication and authorization. Isolation issues between different applications can lead to inconsistent flow rules.

For northbound communication (applications/controllers), an attacker can inject flow rules because the controller has not verified the application. In addition, weak authentication between applications and controller can lead to spoofing. Furthermore, the control plane itself is vulnerable to denial of service (DoS) attacks, and the faulty implementation or even hijacking of a rogue controller can compromise the entire IoBT.

For southbound communications (controllers/switches), eavesdropping and spoofing attacks can exploit the lack of encryption, and weak authentication can lead to man-in-the-middle attacks. The forwarding layer is itself subject to flooding attacks.

All of these vulnerabilities must be considered during system development. To address these multi-faceted issues, a defense-in-depth approach should be taken, using various

technologies to build trust and resilience, with redundancies that increase the overall security of the system and cover the attack vectors mentioned above.

The proposed architecture benefits from this approach by applying a set of security mechanisms and controls arranged in three layers throughout the IoBT. The hardware layer primarily prevents physical attacks. The application software layer protects against attacks on the application and northbound communications. Finally, the network software layer protects against attacks on the southbound communication, control plane, data plane and against SPoF.

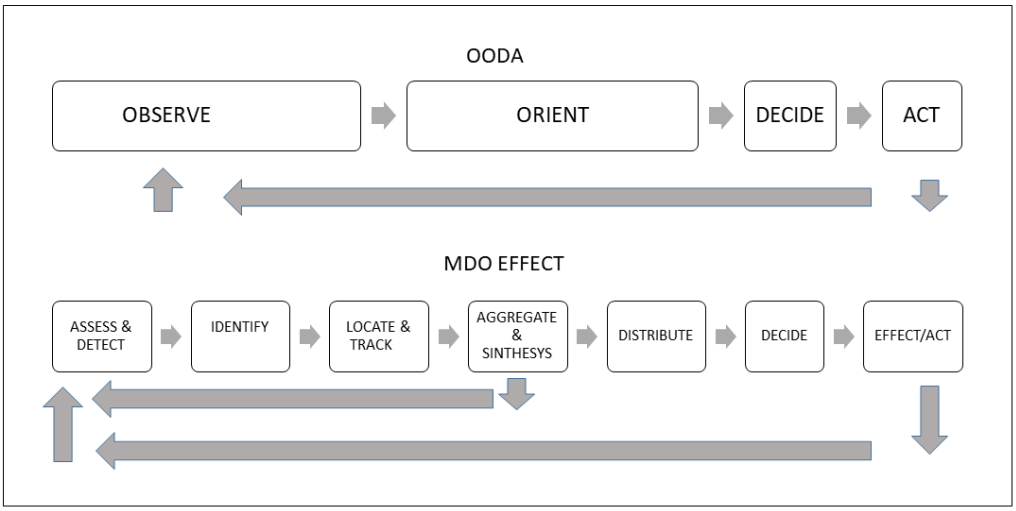
2.2 Multi-Domain Operations

The concept of multi-domain operations stems from the need to conduct military operations that continuously integrate all five domains of warfare: Land, Air, Sea, Space, and Cyberspace. U.S. Army General Stephen Townsend stated in a recent article (STEPHEN J. TOWNSEND; ARMY, 2018) that "*military victories are not defined by battles anymore, but by persistent competition between opposing forces executing integrated operations and campaigning*". These operations occur at varying rates in and out of armed conflict.

The idea behind MDO (FEICKERT, 2021) is to provide commanders with multiple courses of action at both the strategic and tactical levels, taking advantage of surprise and the rapid and continuous integration of capabilities across all domains. Decisive actions should present the adversary with multiple dilemmas to gain physical and psychological advantage, influence, and control of the operational environment. For example, to counter an action in one domain, the adversary becomes more vulnerable in another, expanding the battlefield across domains. The more interconnected forces are to accelerate the decision cycle, the more vulnerable they are to cyber-attacks.

Battlefield information systems supporting operations in multiple physical domains tightly integrate autonomous unmanned platforms with intelligent C2 systems to ensure mission and objective accomplishment (see *Figure 1*). Such systems underscore the importance of IoBT technologies. Since decision speed is a key parameter, the authors propose a shift in the decision cycle from the human-based OODA (*observe-orient-decide-act*) loop to an MDO-effect loop that IoBT devices would follow, whether humans are in the loop or not, as shown in *Figure 2*.

Figure 2 - Decision Loops



Source: Author, adapted from (RUSSELL; ABDELZAHER; SURI, 2019)

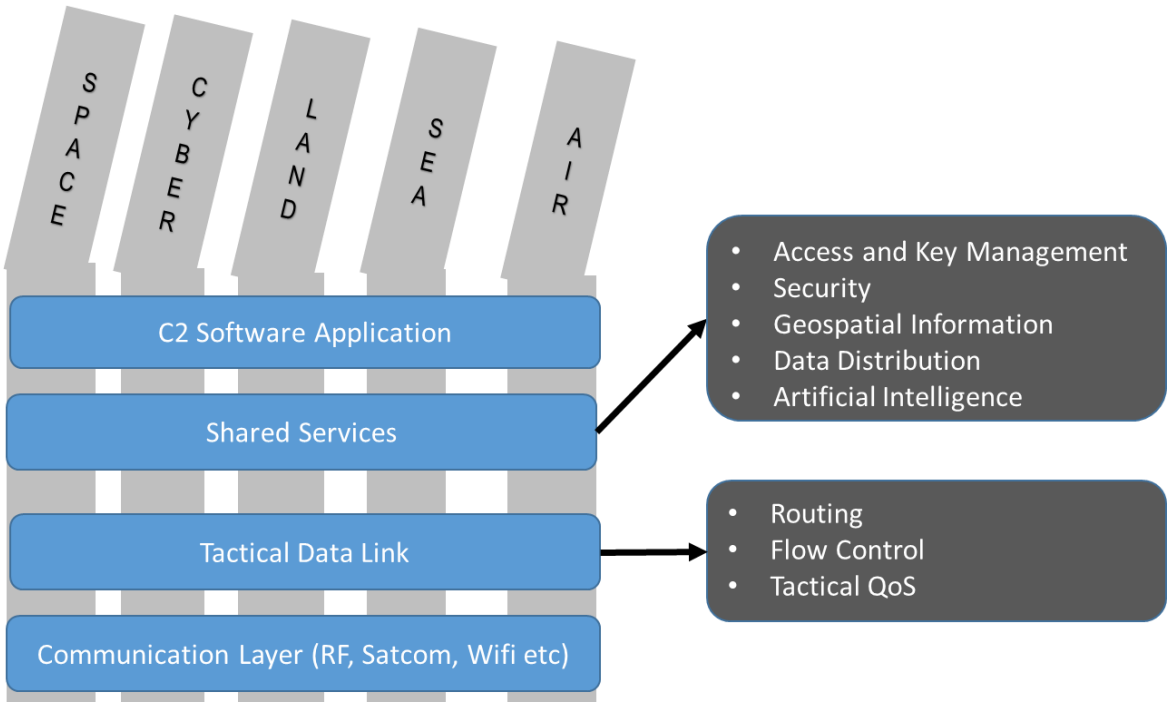
Ideally, human involvement should be limited to the *decision* and *action* phases. The IoBT should take over all the preparatory phases of the loop and transform the *observation* phase of the OODA loop into a sequential, autonomous threat detection, identification, localization, and tracking process delegated to things like drones and smart sensors. The *Orientation* phase would be replaced by network orchestration mechanisms that aggregate information and relay it to the relevant actors, the decision makers at all levels. In addition, the MDO effect loop may provide early feedback to the sensors (when performing aggregation and synthesis), refining data collection to improve quality of information. Implementing such an autonomous process in a use case is precisely the goal of this thesis.

This new autonomous process leads to cross-domain data-driven decision-making capabilities. Here, human intervention is required in most situations, although the things themselves may have some autonomy for routine tasks. The C2 system must satisfy both vertical requirements, i.e., specific requirements to maintain operations in each domain, and horizontal requirements, i.e., conditions that enable cross-domain interactions and data sharing. These requirements lead to an architecture in which different C2 applications (in each domain) are served with information from a common distributed core.

Figure 3 shows a layered approach to sustaining MDO. Fighters in each domain use C2 applications tailored to their needs. To enable MDO, these applications require common

services to ensure interoperability between domains. These services include user access and key management to ensure that only authorized users can participate in the military cloud. Applications must share geospatial data, such as Blue Force Tracking (BFT) and enemy tracks for shared awareness of the situation. These actions are part of the **identification, localization, and tracking** phases of the MDO effects loop.

Figure 3 - Horizontal and Vertical Requirements for MDO



Source: Author, adapted from (NATIONAL ACADEMIES OF SCIENCES, 2018)

The next phases of ***Aggregate & Synthesis*** and ***Distribute*** would benefit from AI and efficient data distribution mechanisms, respectively. All of these shared services run on top of a data link layer that is responsible for routing, flow control, and tactical QoS to handle the channel constraints imposed by the underlying communication layer.

Such a multilayered C2 system guarantees that the desired multidomain effects and actions can be performed jointly by troops and unmanned platforms. The proposed architecture follows these layers and uses SDN to orchestrate network services for the C2 application, providing flow control and a tactical QoS. ICN provides a tactical data distribution (and caching) mechanism. DTN cannot meet strict QoS criteria, but it is used as a data forwarding mechanism for disjoint network segments. For example, in Figure *1*, when two different ICN

segments (islands) come within communication range, DTN is used for data updates. The same is true when a UAV flies in communication range to the WSN and collects its data.

2.3 C2 Agility

Command and Control is the exercise of authority by a commander over subordinate forces to accomplish the mission assigned to him. It enables coordination between the issuance of orders and the acquisition of information on the evolution of the situation and the actions taken. The purpose of command is decision making. The results achieved, especially with the enemy, are the best indicator of its effectiveness. The purpose of control is the effectiveness of the command, i.e. the fulfillment of the mission. It corresponds to the achievement of the desired effects. The effective exercise of C2 by a force is one of the most important factors that contribute to increasing the probability of success of a military operation. Ineffective management of assets, forces, and processes, on the other hand, can lead to failure and defeat.

The following functions are associated with C2:

- Establishing intent (the goal or objective);
- Establishing roles, responsibilities, and relationships;
- Establishing rules and constraints (timelines, etc.); and
- Monitoring and evaluating the situation and progress.

The approach one takes to command and control depends on how these functions are performed. The C2 approach chosen sets the conditions under which military units interact and influences the outcome of the operation. Under the NCW concept, choosing the most appropriate C2 approach for a military operation is not a trivial task, and depending on the dynamics of the situation, if the mission and/or circumstances change, the organization may need to reshape its current approach with consequences for the network configuration. A North Atlantic Treaty Organization (NATO) Task Group (T. G. SAS-085, 2014) categorized C2 according to the variables ADR, PI, and DI.

These variables are explained in detail below (ALBERTS; HAYES, 2006), (HUKILL *et al.*, 2012):

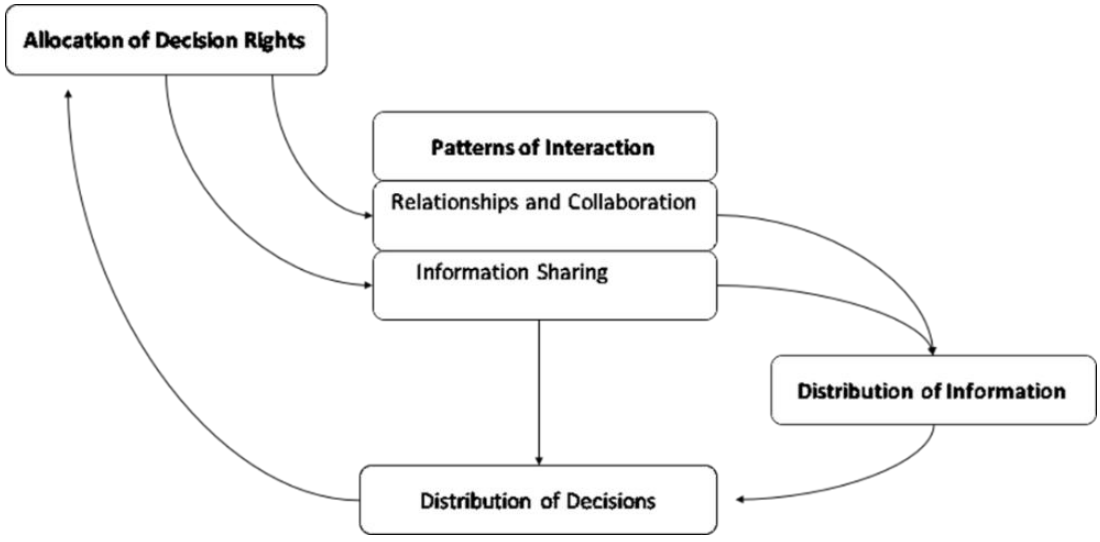
- **ADR (Allocation of Decision Rights)**: Decisions are choices among alternatives. Decision rights belong to authoritative sources on the choices related to a particular topic under some specific set of circumstances or conditions. The allocation of decision rights is their distribution within the military organization across functions, echelons, time, or circumstances. It is a

linear dimension with two logical endpoints. At one end is total centralization (all the rights held by a single actor). At the other end is total decentralization (every actor having equal rights in every decision, or a uniform distribution;

- **PI (Patterns of Interaction)** is defined by three key elements**: Reach (**number and variety of participants), addressing who needs to communicate (e.g., commanders, staffs, and employees); **Richness** (the quality of contents), addressing how they communicate (e.g., face-to-face or by means of video teleconferences or HF instable networks); and **Quality of Interactions,** addressing what types of transactions (e.g., decision, advice, and situational awareness) occur during the communication;
- **DI: (Distribution of Information):** Consists of the various ways and means of sharing information to inform all partners involved in an operation. It includes information sharing across C2 structures of service, joint, coalition, other-government, and nongovernment agencies.

These variables form the key dimensions that make up the C2 approach space and are interdependent as each variable influences the others, as shown in Figure 4. The commander's assignment of decision rights affects the relationships and cooperation among actors and their willingness to share information. These relationships affect the network ability to disseminate information. The right information at the right time enables edge actors to make informed decisions, which may lead the commander to change the current allocation of decision rights.

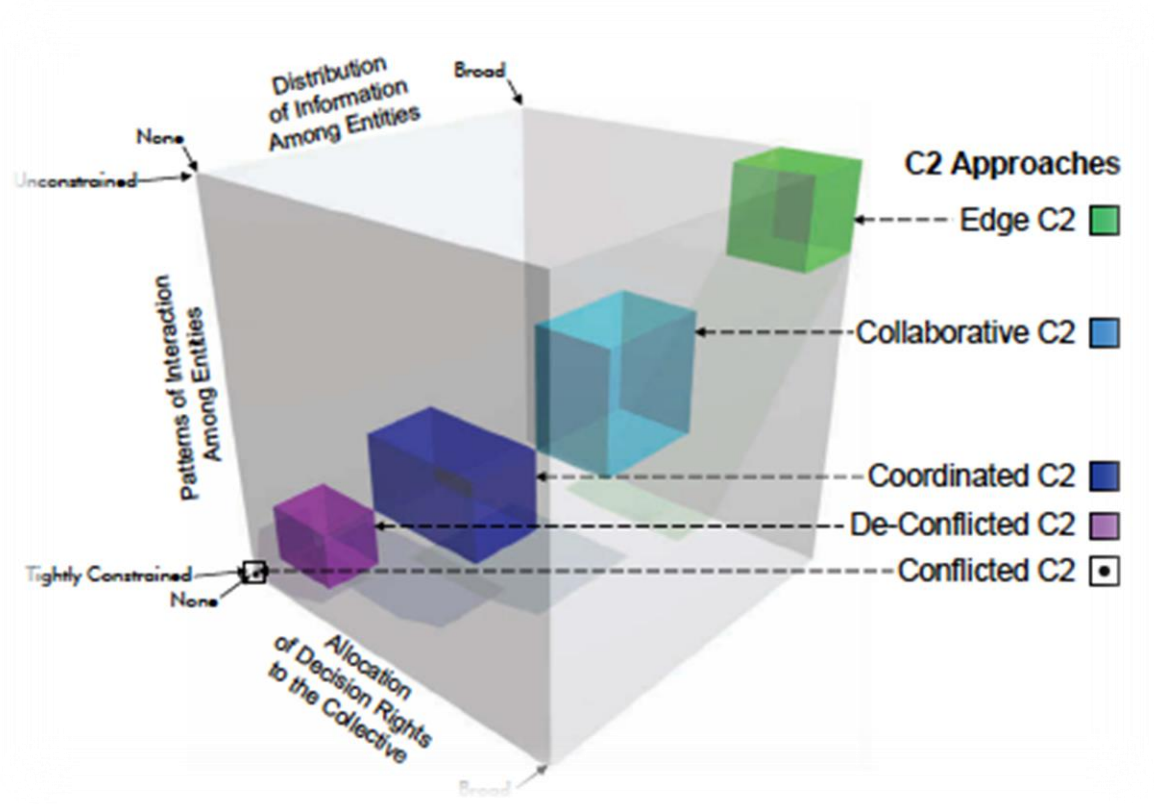
Figure 4 - Relationships among C2 Agility variables



Source: Adapted from (ALBERTS; HAYES, 2006)

Based on the combined use of these variables, the task group identified five distinct regions in this space, as shown in *Figure 5*. These regions range from highly centralized hierarchies (defined as Conflicted C2) to loosely coupled networks (defined by Edge C2). De-Conflicted C2, Coordinated C2 or Collaborative C2 make up the remaining intermediate approaches. In this sense, C2 agility is the ability to change the current C2 approach in response to changes identified during situational awareness to a more appropriate approach that enables the effective and efficient use of system resources in a timely manner.

Figure 5 - C2 Approaches space



Source: (T. G. SAS-085, 2014)

A related work (ZHOU *et al.*, 2020) explored how to reconcile the traditional (conflicted) and edge C2 structures to create a suitable model for an agile C2 organization based on the network information system. Another work (SCOTT *et al.*, 2016) explored the interdependencies of these variables using a framework to test the technical and social layers of tactical networks and then evaluate the performance of different organizations and approaches.

The authors used a command-and-control experimental platform called ELICIT, to draw conclusions about the performance of the social layer (people) in a simulated operation as a function of team organization and C2 approach, considering hierarchical and edge topologies. This work, on the other hand, focuses on the technological aspects that can influence the interdependence of variables in order to propose a feasible system implementation that takes into account the interaction between humans and machines.

The ELICIT platform enables instant sharing and perfect data transfer. To evaluate the technical level of the system in realistic scenarios, they used the network emulation platforms EMANE and CORE along with their own code for message translation and inter-platform communication. At the technical level, they varied the data size, distribution, and channel parameters, such as bandwidth and noise.

The authors claim that their framework provides insight into the interactions and interdependencies of the network technical and social layers and their impact on mission performance by establishing the organizations baseline outcomes of bandwidth, information distribution, and C2 approach. This baseline provides the network designer with useful information for optimizing network parameters in successive mission campaigns.

In contrast, this work proposes a solution to optimize these network parameters for the IoBT during mission execution through SDN-based orchestration of data dissemination protocols such as DTN and ICN. The overall goal is to achieve C2 agility in the IoBT context, which is challenging due to many technological shortcomings in the currently deployed C2 systems. They lack the necessary flexibility to improve the use of available resources.

It is necessary to develop the ability to modify (dynamically) the three variables (ADR, PI and DI) in response to changes during the execution of the operation. Current systems are mostly designed for human-to-human interaction and are not suitable for the context of an ensemble of humans and intelligent things. In response to this need, this work proposes SDN to provide flexible control of the ADR and PI variables, and ICN and DTN as enablers for DI.

Table 1 provides a summary of the C2 agility variables and their mapping to the network paradigms, as well as an overview of which network parameters can be improved by using the features of each paradigm. The variable **ADR** can be dynamically modified by the SDN capability to change node hierarchy and authorization and reconfigure switches in the network accordingly. It can also benefit from the ICN's named data functions, where only authorized nodes can request specific data from the network (these functions are discussed in more detail in Chapter 3).

The variable **PI** benefits from SDN's prioritization of data flows, which governs the **richness** of interactions, while all paradigms improve **reach** through integration with the military cloud, dealing with intermittent connections, and binding data producers and consumers. Similarly, the **DI** variable is enhanced by data filtering techniques orchestrated by the SDN paradigm, as well as by the store-and-forward properties of DTN and content dissemination (situational awareness) from ICN.

The bottom line is that the architecture aims to improve network parameters. SDN enables network orchestration and increases throughput, DTN provides packet loss tolerance for elastic applications, and ICN provides low latency for non-elastic applications.

Table 1 - Network support for C2 Agility

C2 Agility Variables	**Network Paradigm**		
	SDN	**DTN**	**ICN**
Allocation of Decision Rights (ADR)	Node Hierarchy & Authorization		Named Data
Patterns of Interaction (PI)	Data flow Prioritization & Integration to Military Cloud	Intermittent Connections	Integration of data producers and consumers
Distribution of Information (DI)	Data filtering	Store and Forward Data	Intermediate Caching & Content Dissemination
Improved Network Parameters	Network Orchestration & Throughput	Packet Loss Tolerance	End-to-End Delay

Source: (STOCCHERO *et al.*, 2023)

3 UNDERLYING NETWORK TECHNOLOGIES

Chapter 2 discussed the current state of NCW research, focusing on the key challenges to achieving C2 agility in the IoBT context. These challenges lead to software and network requirements that cannot be efficiently met by current implementations. In this chapter, the network paradigms studied in this thesis are discussed along with some design decisions to build the proposed architecture.

The proposed architecture combines different network paradigms to provide shared services and concurrent tactical data connectivity for the C2 applications. SDN-compliant applications use services provided by an SDN controller. The controller orchestrates the network using two different data dissemination paradigms, DTN and ICN.

3.1 Delay Tolerant Networks

DTN is a network architecture approach that addresses the problem of lack of continuous connectivity in dynamic networks where there are extended periods of time when connectivity is not available (FALL; FARRELL, 2008). DTNs enable communication between regions where reliable communication for the implementation of common wireless communication standards is limited by intermittent connectivity, long propagation delays, asymmetric data, high latency, and high error rates. Originally developed to address the problems of an interplanetary network (IPN), DTN data routing is a suitable communication mode in particularly challenging environments such as Vehicular Ad-Hoc Networks (VANETs) (AZZOUG; BOUKRA, 2021) and Flying Ad-hoc Networks (FANETs) (OUBBATI, 2019), as these environments are susceptible to the above problems.

DTN routing is based on finding a store-carry-and-forward node that is better at forwarding data to the destination compared to other potential next-hop nodes. (AMIN *et al.*, 2015) noted that the technology is also being explored for terrestrial applications such as battlefield networks, but the proposed design is limited to the strategic level relying on resource-rich nodes and does not cover the last mile characteristic of IoBT, as well as low-density wireless sensor networks and drone communication systems.

Military MANETs can thus increase the range of communications to ensure compatibility between network segments by picking up interference and translating protocols between networks. Relay nodes are often used in these networks to relay data from source to

destination and to achieve a wider coverage area (including rough terrain and harsh environments) in which they are normally deployed.

DTN stores potentially large amounts of data in intermediate nodes and forwards it to the destination or to the next DTN-enabled node when opportunistic connections occur. However, intermediate nodes may become unavailable for a number of reasons, including lack of power, technical failures, or hostile activity. In addition, due to the high mobility of some nodes (drones and ground vehicles), these networks are susceptible to topology changes that may require recalculation of data routes.

The Internet Research Task Force (IRTF) proposes an architecture and protocols for data exchange over DTN, inserting a "convergence layer" between the transport and application layers. This layer provides DTN services via a bundle protocol for all elastic applications that do not require strict constraints. The DTN bundle protocol (DTNBP) designs the message formats (called bundles) that are transmitted between DTN bundle agents in the bundle communication network leading to the DTN storage and forwarding overlay network.

The store-and-forward method is not based on node-to-node relays, but on star relays, in which a central storage device is contacted separately by both the source and the destination. Each node contains a persistent memory that can store messages indefinitely. The message switching mechanism also provides information such as retransmission bandwidth, buffer memory, and message size to the DTN nodes in the network (or to an SDN controller that can use this information to program the network). These features are particularly useful in the IoBT, where drones can serve as persistent relays to connect different network segments and collect and relay sensor data from the WSN to the rest of the network.

(OUBBATI, 2019) conducted a comparative study of DTN routing protocols in terms of dynamic network density and complexity and identified the advantages and disadvantages of each approach. In general, DTN routing solutions improved delivery rates and reduced packet loss at the cost of higher delays. These results confirm that the use of DTN can compromise QoS for non-elastic applications, while increasing delivery rates for elastic applications when buffering is allowed in intermediate nodes.

Preliminary work (ZACARIAS *et al.*, 2017a) at the outset of this research explored a mechanism for leveraging the DTN paradigm as a last mile solution to IoBT, examining DTN and SDN concepts for handling link failures and network programming, respectively, in support of network-centric military operations. A logically centralized SDN controller manages forwarding devices coordinated by a DTN orchestrator, which schedules packet transmission

between nodes based on data collected by the SDN controller and information gathered by the DTN layer.

The DTN layer follows the concepts of current DTN implementations. The bundle storage stores collected data when the destination node is unreachable. The convergence layer connects the bundle storage to the transport layer and sends the data stored in the bundle to the destination or neighboring nodes via UDP or TCP using an appropriate routing algorithm. There are many routing algorithms for DTNs in the IoBT context, such as geo-routing, multicast, and optimization-based algorithms (AZZOUG; BOUKRA, 2021). The DTN orchestrator selects the most appropriate protocol depending on the network conditions, and the SDN controller facilitates the change of routing protocol while the IoBT continues to run.

The results of this preliminary work show how DTN could be used in the proposed architecture as a suitable mechanism for distributing data between different network segments. Drones can be used as a persistent storage of the DTN mechanism due to their high mobility and ability to connect disjoint network segments such as WSNs. DTN can also be used for intermittent connections between SDN controllers and to coordinate different ICN islands.

3.2 Information Centric Networks

Although current IP networks use ports and hostnames for addressing and forwarding, Internet traffic is primarily about content delivery. The same concept applies to IoBT, where a distributed decision process requires the dissemination of information. Network usage has shifted from endpoint communications to a data-centric model where it is no longer about where (addresses) but what (content). Information Centric Networking (ICN) is a new type of network paradigm that changes the semantics of network services from the traditional address-based (host-centric) approach to a network architecture that focuses on information that can be represented by the content or query of raw data based on named data objects (NDO).

These NDOs can be of two types: INTERESTS or DATA (NOUR *et al.*, 2019). Consumers request data by sending INTERESTS messages (named data to be retrieved), and producers (or intermediate nodes that perform in-network caching and storage) send back matching DATA messages that are signed to guarantee authenticity and integrity.

This cryptographic signature of DATA messages (name-content binding) moves security into the data itself and away from hosts and servers. By decoupling senders and receivers, ICN also provides inherent mobility that is ideal for ad hoc deployment scenarios, and by caching and replicating on the network, data is transported to consumers, making

information distribution more efficient than traditional IP networks in terms of latency and traffic load.

These new features for security, mobility, and in-network caching make ICN a great enabler for IoT and edge networking (BACCELLI *et al.*, 2014). It is possible to use the same namespace for the network and applications, focus on data associated with things rather than devices, and leverage the data path and caching to meet IoT requirements related to energy and bandwidth constraints and content availability with intermittent node activity.

This design can also facilitate IoT deployment for MANETs, FANETs, and VANETs by improving network performance, security, and mobility, and optimizing device energy consumption. IoBT and VANETs have characteristics such as highly dynamic topology, frequent link interruptions, localized communication (GPS), and timing constraints for delay-sensitive and delay-tolerant messages also pose challenges that can be overcome with ICN as the underlying communication paradigm (KHELIFI *et al.*, 2020).

ICN principles are currently implemented through specific concepts (NOUR *et al.*, 2019), e.g., Publish-Subscribe Internet Routing Paradigm (PSIRP/PURSUIT), Content-Centric Networking (CCN), Mobility-First, and Named Data Networking (NDN). These concepts support intermittent connectivity by making storage transparent both on the end host and in the network through caching and replication mechanisms. In addition, ICN considers node mobility and multiple access as the rule rather than the exception, so that anycast, multicast, and broadcast data dissemination are natively supported.

These features aim to increase the overall efficiency of the network, improve its scalability and robustness by decoupling the data from its producers and making the data location and application independent. The characteristics described make ICN well suited for battlefield networks (BN), since most of the aspects mentioned above are found in such military operational environments, e.g., mobility of nodes, broken links, and the need for network scalability and robustness.

In this work, the NDN architecture (ZHANG *et al.*, 2014) was used to model ICN concepts in the IoBT. NDN establishes a request-response mechanism and uses two types of packets: INTEREST and DATA. Both interest and data packets carry the name of the requested content, and routing is achieved by using names. Consumers request data by sending INTEREST messages and producers (or intermediate nodes that handle caching and storage in the network) send back the matching DATA messages. NDN uses hierarchical, human-readable, and structured URL-like names and manages three data structures: cache storage (CS), pending interest table (PIT), and forwarding information base (FIB).

The purpose of NDN is to facilitate finding relevant information on the network in a scalable way. Related work (CAMPIONI *et al.*, 2019) (GIBSON *et al.*, 2018) has explored NDN solutions from the perspective of military applications. NDN is a pull-based architecture where the consumer always asks for information and the network receives it from producers (or intermediate nodes that cache it). NDN offers interesting synergies with IoT applications. Through the clever use of specially designed naming schemes, NDN can implement publish/subscribe communication with support for distributed caching - a communication model that is particularly well suited for the IoT.

The use of NDN enables the adaptation of application behavior to the current network performance. More specifically, by properly setting the transmission rate of interests, a node can attempt to mitigate node mobility issues and manage traffic flow according to available network resources. However, there are many challenges for NDN implementation in BN. The naming scheme and security are some of the most important ones. The network strategy is also important, as the flexibility introduced by this protocol allows the forwarding policy of interests and data to be tailored to application-specific requirements.

The three key performance metrics are delivery rate, latency, and bandwidth utilization - all measured in the context of disseminating three different types of information objects - blue-force data, sensor data, and documents. For these objects, NDN provides routing, caching, mobility, and security mechanisms suitable for dealing with the heterogeneity and ad hoc nature of mission-critical applications. This research has tested this concept in another preliminary work (LEAL *et al.*, 2019), using a use case for the IoBT in an urban context.

In this use case, ICN was used in the context of SDN for data dissemination within an SDN-controlled network segment. Just as DTN was evaluated for connecting disjointed network partitions, ICN was used in this case to improve delivery rate and latency within network segments, with the corresponding DATA responding to the C2 application INTERESTS with relevant information for decision making.

However, C2 applications also require node-specific data to support various data objects such as voice and video conferencing, so a large-scale network deployment using only NDN, especially at BN, is still far from reality and cannot yet replace IP infrastructure. The most likely ICN use is based on ICN islands, where another technology, such as SDN, must be used to interconnect the islands and integrate them with the IP infrastructure, as proposed by (ZURANIEWSKI *et al.*, 2017) and (SIRACUSANO *et al.*, 2018).

3.3 Software Defined Networks

The SDN paradigm was originally envisioned as a technology that could be deployed in wired networks with the goal of decoupling the hardware from the controller, which greatly simplified the development and exploration of new protocols, network policies, and management (NUNES *et al.*, 2014). In the SDN approach, the controller is responsible for all forwarding processing, while the switches only reroute the data streams according to the controller's instructions.

To achieve this, the controller must communicate with all switches in some way, with OpenFlow being the most commonly used protocol. This protocol uses an approach (the duple <**match, actions**>) where the match field identifies the packets (through fields such as Internet Protocol, port number, protocol type, and others) that should perform the actions. The match field can use packet header information, the ingress port, and metadata.

The research community and industry have quickly adopted it and extended its use to other types of (wireless) networks, such as satellite networks (BERTAUX *et al.*, 2015) and wireless sensor networks (WSN) (LUO; TAN; QUEK, 2012).

The paradigm is intended to provide flexibility to network management by dividing the network infrastructure into different planes, separating the data forwarding (or data plane) from the network control logic (control plane). Network equipment in the forwarding plane consists of simple packet forwarding devices, while a centralized unit, the SDN controller, implements the control logic of the network. Each plane can be programmed to meet specific application requirements, which means significant changes to the network itself or the application it supports. For example, legacy or commercial off-the-shelf applications (COTS) can be supported without requiring major changes to their behavior (NUNES *et al.*, 2014).

It should be noted that centralized control does not mean that there is a physically centralized entity. Rather, single points of failure (SPOF) should be avoided, and redundancy in the control plane is highly desirable. The clear separation between network layers and the abstraction of network control logic from distributed hardware facilitates the implementation of new network management strategies. An external application occupying the application layer can interact with the network through the SDN controller (WICKBOLDT *et al.*, 2015).

Examples of external applications include load balancers, network orchestration frameworks, and business functions. The SDN architecture enables interaction between planes by defining protocol-neutral interfaces. External applications use the northbound interface to communicate their requirements to the SDN controller or to obtain information about the

general state of the network. The SDN controller translates the high-level specifications from the application plane into low-level specifications via the northbound API and applies them to the data plane via the southbound API.

The controller also centralizes changes to the network itself, ranging from routing configurations to network security policy enforcement. The currently known and industry-adopted standard protocol used in the southbound interface is OpenFlow (LUO; TAN; QUEK, 2012) .

3.4 SDN Architectures for the Battlefield

All of the flexibility that SDN provides makes the paradigm suitable for BN, as it facilitates end-to-end interactions and supports the use of edge computing resources to process nearby data. However, there are still some challenges related to security and interoperability between different heterogeneous, dynamic, intermittent, and data packet-based technologies, such as the Multi-Bearer Network (MBN), that need to be resolved to take advantage of SDN in tactical environments.

Previous work investigated the use of SDN in the BN scenario, such as in (NOBRE *et al.*, 2016), focusing on improving communications between resource-rich nodes on the battlefield (e.g., battleships and aircraft). In this research, the focus is on resource-constrained devices used by troops in the field by combining SDN with DTN and ICN.

Recently, a systematic literature review on SDN for military applications was published (GKIOULOS; GUNLEIFSEN, 2018), classifying current research according to its application domain (coalition, military, mission critical, tactical, etc.), its contribution (architectures, assessment, services, control systems, etc.), and the extent to which it focuses on security.

A related work (PHEMIUS *et al.*, 2016) explores a framework for traffic management in tactical networks based on the Mobile Edge Cloud (MEC) architecture and its close interaction with Software Defined Networking (SDN) at the platoon edge. Their solution provides services such as Blue Force Tracking (BFT), multimedia content delivery and remote sensor control using multiple wireless technologies with two main objectives: Ensuring quality of service (QoS) for various tactical applications (voice, data and multimedia) and maximizing radio utilization, a valuable resource in tactical networks.

They use a MEC controller as a transversal module that bridges three layers of the architecture: the waveform layer, the SDN layer, and the application management layer. Their experiments confirmed that the SDN layer could guarantee QoS while ensuring efficient use of

wireless resources. The SDN controller enables the reconfiguration of the data flow at the switch interfaces based on the decisions received from the MEC controller. The network operator deploys applications according to end-user requirements in the application management layer. Each application can be considered as a single purpose block with a well-defined functionality.

Another work (CHEN *et al.*, 2018) proposed an SDN architecture for military aviation that is also organized into three different planes (data, control, and application). The data plane is responsible for forwarding (e.g., routing, traffic splitting), transmitting (e.g., channel access, radio switching), and processing (e.g., information fusion, signal processing) data and becomes a communication resource pool that abstracts logical resources from physical resources through virtualization techniques. The control plane is divided into three layers: the swarm control layer (SCL), the group control layer (GCL), and the platform control layer (PCL), creating a hierarchical set of SDN controllers that link mission planning with network configuration. The application plane manages a variety of aviation applications for each subnet. Requests from a network application are translated into rules by the Northbound Interfaces (NBIs) of the SD - ANoAS controllers, which in turn are interpreted into instructions for the underlying devices.

Recent work (MAHMUD *et al.*, 2021) aimed to review current research initiatives in SDN-enabled tactical networks. They propose a system model and taxonomy for SDN orchestration, particularly from the perspective of tactical networks, relying on a framework consisting of four planes: Application, Control, Forwarding, and Orchestration.

The application plane consists of SDN-enabled or legacy applications with different QoS and security requirements. The SDN-enabled applications communicate directly with an SDN controller, while legacy applications simply send data through the network. The control plane consists of multiple specialized SDN controllers that can communicate either on a peer-to-peer basis or through an orchestrating controller with a global, cross-network view.

The forwarding plane consists of network nodes capable of forwarding packets based on the routing policies implemented by the SDN controllers. Finally, the orchestration plane spans all layers and is responsible for monitoring and aggregating data that is used in a meaningful way to support efficient network orchestration in terms of controller management, service resiliency, interoperability, and policy enforcement.

3.5 Research Challenges and Opportunities

All these characteristics make SDN an interesting approach for battlefield networks, especially with respect to multi-domain operations where heterogeneous access networks need to serve different applications. The BN requires a flexible middleware to tune the network parameters and achieve the desired quality of service for each application. SDN can be used as such a middleware for C2 agility through its control and data planes.

However, the challenge of disseminating relevant data for decision making cannot be met by SDN alone, given the harsh conditions in military edge networks. A sophisticated mechanism for data dissemination must be developed to meet the conflicting requirements of real-time data delivery and intermittent links. *Table 2* summarizes the state of the art for each of the network technologies and the challenges of integrating them into a network solution.

In general, both DTN and ICN can be used to disseminate information. However, the operational circumstances that may result from their use are different. While ICN contributes to the achievement of QoS by reducing latency and bandwidth utilization, DTN jeopardizes it by increasing delays. Therefore, the goal of this research was to determine which application would benefit from each paradigm and to develop a coordination mechanism for the simultaneous use of both paradigms in the architecture.

This was only possible by SDN-driven middleware that translates application requirements into individual data transfer protocols. Within contiguous segments (islands), the controller uses ICN, while DTN is chosen for integrating disjointed segments to increase AOI coverage.

Of course, SDN tends to increase the traffic overhead and processing power of the nodes hosting the controllers. This leads to another challenge in such resource-constrained networks, namely the placement of the controllers and how they can interoperate for a distributed (albeit logically centralized) control plane. All these aspects of the individual technologies used are discussed in detail in Chapter 4.

Table 2 - Network Technologies in this study - Research Summary

	SDN	DTN	ICN
Operational gain	✓ Translates application requirements to network devices ✓ Manages traffic flow & network resources ✓ Integrates new protocols	✓ Absorb disruptions in different network segments ✓ Increase area coverage using relay nodes	✓ Disseminates operational data through caching and replication ✓ Decouple data consumers to producers
Strengths	✓ Guarantee quality of service ✓ Efficient usage of the wireless resources	✓ Enhance delivery ratio ✓ Reduce packet losses	✓ Reduce latency, bandwidth use and traffic load. ✓ data location and application independent
Weakness	✓ Complex implementation of decentralized control ✓ Processing and traffic overhead	✓ Higher delays	✓ Not efficient for end-to-end voice and video
Research Challenges	✓ interoperability among several heterogeneous network technologies ✓ Placement of SDN controllers	✓ QoS for non-elastic applications	✓ Integration with IP network segments

Source: The author

4 A C2 ARCHITECTURE FOR THE IoBT

Chapter 2 discussed new paradigms in Command and Control research, featuring key aspects of multidomain operations and their relevance to current military victories. In addition, C2 agility is essential to achieve an improved decision cycle in such a context. The drawbacks of current C2 systems also emphasize the need for innovative application and network technologies. Current research on novel network paradigms that could fill the gaps in C2 systems was discussed in Chapter 3, which summarized the link between each technology and its application in the military domain.

In this chapter, an innovative solution is presented that links these two different research areas - military command and control systems and computer networks. The idea is to use new paradigms in network protocols to address emerging trends in command and control technology. Since research in both areas is still evolving, few systemic implementations leverage these concepts. This research developed several use cases as proofs of concept. First, the use scenarios and requirements are described. Then, the theoretical principles are demonstrated in practical scenarios using various deployment examples. Finally, the architecture is presented in the last part of the chapter.

4.1 Operational Scenarios and Requirements

Current military operational scenarios range from war-like situations, as shown in *Figure 1*, to operations other than war (OOTW involving actors other than the military (state and non-state civilian agencies), as shown in *Figure 6*. Both scenarios represent the diversity of requirements for deploying effective C2 systems: limited and dynamic capacity, heterogeneity, and mobility, with mesh networks forming and dissolving as mounted and dismounted nodes move around the battlefield.

The scenario depicted in *Figure 1* represents a platoon on a reconnaissance or surveillance mission scouting an area of interest (AOI) to detect and locate enemies or identify strategic features and report to C2 headquarters. The platoon consists of armored and mechanized vehicles with an IoBT solution that includes unmanned aerial vehicles (UAVs), wireless sensor networks (WSNs), and intelligent equipment for foot troops (e.g., portable electro-optical systems) to provide enhanced surveillance capability.

The goal of the IoBT solution is to cover a larger area and reduce the number of military personnel required to accomplish the mission. At the same time, the IoBT solution increases

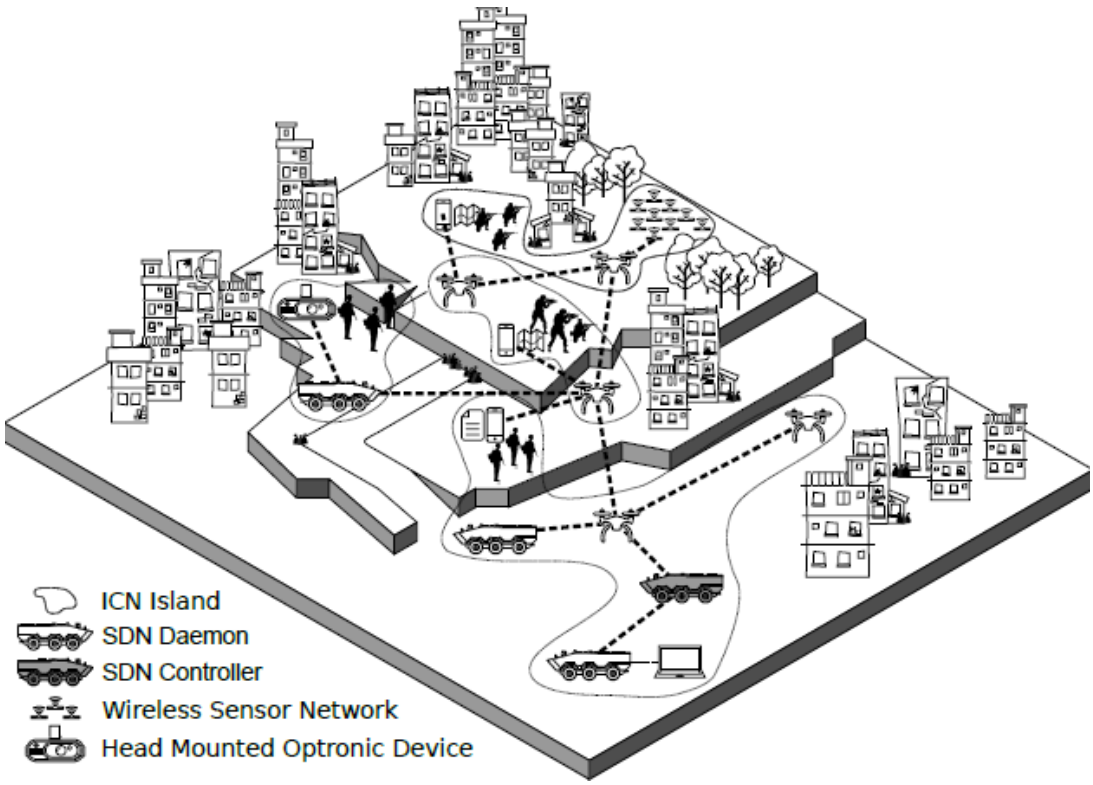
the safety of the vehicle' crew as the unattended sensors take over the riskier surveillance tasks. Two vehicles representing different squads are shown, a 6x6 and a 4x4, each consisting of a crew and equipment such as quadcopter UAVs and WSNs.

UAVs capable of autonomous flight and threat detection can monitor the area across the river and send images/video back to the vehicle. WSNs can detect people crossing the river or other threats in areas of particular interest. A soldier on foot may also have a head-mounted optronic device (HMOD) to locate a target. All squads involved in the mission may have the same assets and must find the best way to accomplish the mission together and efficiently.

The second type of scenario (OOTW in *Figure 6*) is increasingly urban in nature and is often faced by the Brazilian military in operations to maintain public order. Due to the chaos wrought by organized crime in certain regions of megacities such as Rio de Janeiro, regular law enforcement agencies are unable to meet the challenge. The adversaries are scattered and hidden in an environment that is difficult to access and dynamic, and they can easily blend in with the civilian population.

.

Figure 6 - OOTW Operational Scenario



Source: (LEAL *et al.*, 2019)

The presence of various networked elements can be seen in the figure. Military vehicles have access to only a portion of this operational environment. Beyond a certain limit, only foot troops are able to detect enemies hiding in this maze-like environment. Since it is difficult to advance safely in such an area, drones can assist troops by providing aerial images of the movements of suspicious individuals on the ground. These locations are usually adjacent to urban forests, which in many cases are perfect escape routes for criminals monitored by wireless sensor networks (WSNs).

By leveraging the various resources offered by multiple devices on the ground, the mission commander can monitor what is happening in both situations, for example, by requesting video footage from drones and soldier-worn Head Mounted Optronic Devices (HMOD). From his observation position, the commander can make informed decisions about the (re)allocation of troops and issue new instructions for the next actions. However, due to the dynamic nature of such an operational scenario, troops deployed to the front cannot wait for decisions on each of their next steps. Therefore, some degree of autonomy is required for local decisions, which in turn require data/information to support them.

For both scenarios, the corresponding C2 system must account for the variables **ADR** (decision rights of actors involved in the operation), PI (who communicates with whom and how), and **DI** (information flows to C2 centers and to actors at the edge of the network). The IoBT nodes shown in these operational scenarios are located at the edge of the military cloud, at the lowest tactical levels (from platoon up to battalion), where there are usually limitations in communications, especially in terms of connectivity, bandwidth and throughput. In this sense, the IoBT consists of a tactical edge network at the last mile (TEN) (TORTONES*I et al*>, 2013).

Preliminary work in this study explored solutions to the challenges of these deployment scenarios using SDN principles, such as dynamic self-configuration of data flow parameters that handle the variable PI. The work in (ZACARIAS *et al.*, 2017b) (ZACARIAS *et al.*, 2018) addressed the network throughput required for video streaming in IoBT and presented an analysis of how SDN principles can provide the required quality of service for surveillance. In this case, an SDN-enabled network was used with multiple UAVs as data providers.

In addition to SDN, DTN and ICN principles can be used to deal with another variable: The distribution of information. The work in (ZACARIAS *et al.*, 2017a) added DTN principles to investigate further solutions for IoBT in terms of intermittent connections, as mentioned in Section 3.1. Later in the research, (LEAL *et al.*, 2019) explored how ICN can be fused with SDN to provide C2 agility in provisioned networks. ICN provides more efficient data

distribution within "ICN islands" in the military IP network, while SDN enables integration of ICN with the rest of the IP network and control the patterns of interaction between heterogeneous IoBT nodes, enhancing ICN capabilities.

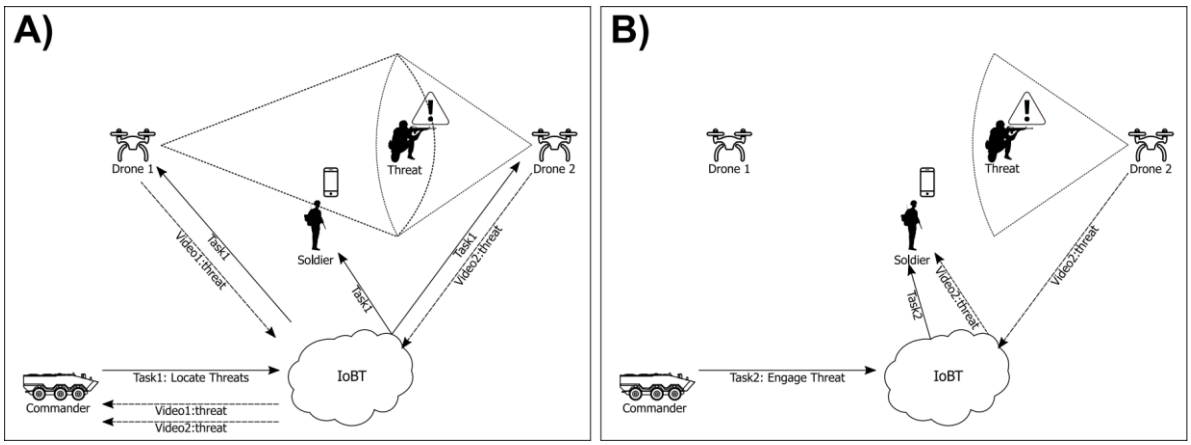
The network solution initiated in (LEAL *et al.*, 2019) and studied in depth in (STOCCHERO *et al.*, 2023) uses such technologies to optimize IoBT communication parameters such as latency, channel bandwidth, interruption, and node failure, and to deal with topology changes due to node mobility. SDN orchestrates the network by establishing different network slices and optimizing data flows based on network (and node) state (bandwidth, data size, channel latency, channel availability, etc.). Within each slice, ICN functions improve information distribution and human-thing interaction.

By controlling which node is allowed to send/receive data, SDN can act on the third dimension of C2 space, namely the allocation of decision rights. During a deployment, when the SDN controller receives input from the deployment's commander through the C2 application, it can change the hierarchy, permissions, and data flow rules, providing the technical means to implement C2 agility in the network.

4.2 Deployment Examples

To illustrate the operational benefits of this combination of SDN, DTN, and ICN in such operational scenarios, this section presents the example situations shown in *Figure 7*, *Figure 8* and *Figure 9*. The first example focuses on ICN. This example highlights four nodes and illustrates how the combination of SDN and ICN capabilities can improve data dissemination and make the interaction between nodes more effective, contributing to C2 Agility.

Figure 7 - Deployment Example: SDN and ICN



Source: (STOCCHERO *et al.*, 2023)

Part A shows the commander's vehicle initiating *Task 1*, which engages the network to **locate threats**. He is the highest-ranking node at this point, and the responses to this task are the highest priority data messages. In response to *task 1*, drones 1 and 2 identify a threat and send videos 1 and 2. The network caching and forwarding configuration (set by the SDN controller) gives priority to sending these videos to the commander. This phase corresponds to the distribution of information. The soldier also receives the videos, but with lower priority and therefore possibly with higher latency.

Using the feedback received from the network (*videos 1 and 2* and location of all assets, the latter from the controllers status messages), the commander assesses the situation and decides that *Soldier 1* is the best effector and *Drone 2* is the best sensor to neutralize the threat. With this decision, he initiates phase two, shown in part B of *Figure 7*. It sends *task 2* to the network with the highest priority. The IoBT (SDN controller) understands that *Soldier 1* and *Drone 2* are now responsible for this threat and updates the forwarding and caching policies for the network devices. Data flow to/from these nodes now has priority.

All these changes in the network configuration are possible because the SDN controller constantly receives status messages from all nodes (with their current capabilities and location) and tasks from authorized nodes, which can change the hierarchy and priorities. However, it is important to note that when a particular node is disconnected from the controller, the NDN service maintains the data plane using previous rules (hierarchy, priority, and filters) until the connection with the controller is restored and the rules are updated.

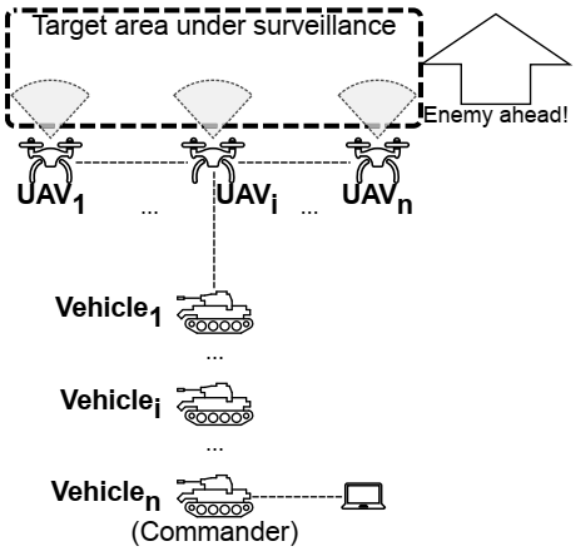
In the IoBT context, things may also need information to support their activities. For example, in an urban operational scenario (*Figure 6*), the platoon commander or a police officer would need images of possible escape routes of suspicious persons. The most obvious producers of such information would be drones. Since drones are intelligent things, they would try to determine the likely places where they could fly and capture such images. Therefore, they would need information about motion detection. When motion is detected, the WSN could send that information, along with the approximate location of the suspected motion, to the drone without human intervention. The nearest drone would move to the area in question, capture the image, and respond to the commander.

In this example, the drones themselves request data from other nodes, in this case the WSN, to fulfill a request made by a human. By doing this at the network level, ICN enables the implementation of different interaction patterns between humans and things. These interactions will support a much better and more flexible control cycle where Sensemaking (detecting

movement and collecting images), Execution (flying to the right zone), and Situational Awareness (detecting enemy evasive maneuvers) are performed without the need to address a specific host/node, but the required information itself.

In the second example, however, the DTN is used to respond to a different situation in the same scenario, as shown in *Figure 8*. Here, the problem is a channel disruption that results in a packet loss in communication with a group of small UAVs launched from one of the vehicles to explore the terrain ahead of them. As they fly far ahead of the vehicles (to protect the platoon from enemy attack), the UAV group forms its own network, which disconnects from the launch vehicle for a period of time due to interference or range limitations. In this situation, the SDN controller sets DTN as the communication protocol for the mission. Once a UAV closer to the ground vehicle reconnects, the DTN mechanism will use it to transmit the video captured by the swarm.

Figure 8 - Deployment Example: SDN and DTN



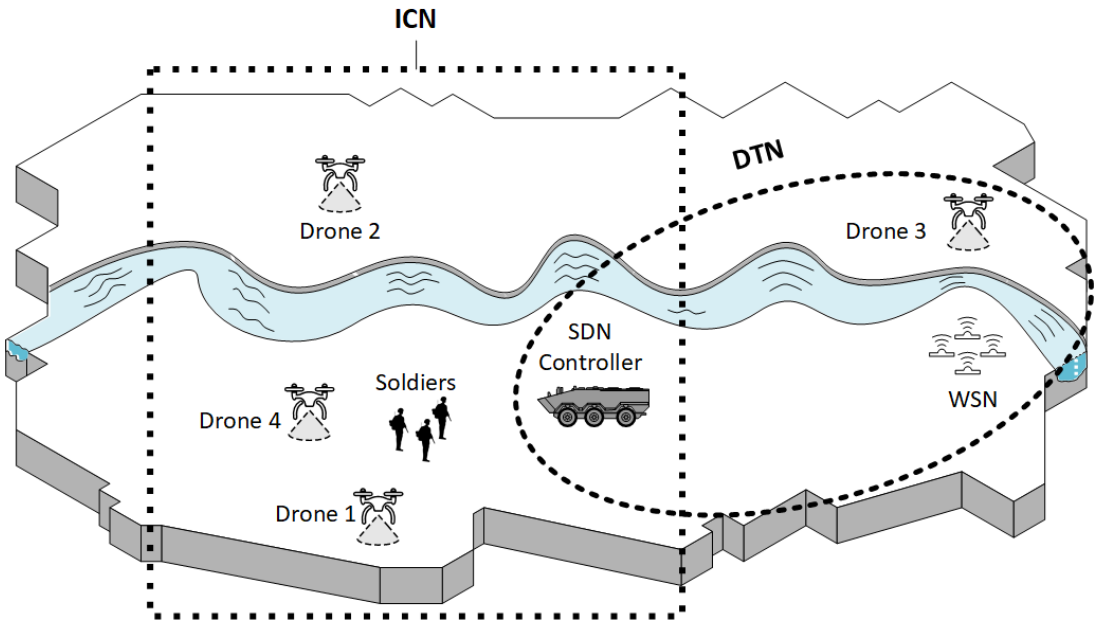
Source: (ZACARIAS *et al.*, 2018)

The third example, shown in *Figure 9*, illustrates the combined use of the two paradigms. The WSN, deployed to detect enemy movement in a specific area along the river, separates from the rest of the IoBT, which continues to move to the left to survey another area. Drone 3 flies at the rear of the moving IoBT to protect the troops. When it comes within communication range with the WSN, it is used as a DTN relay to transmit the collected data to

the commander. At the same time, the other nodes (drones 1, 2, 4 and soldiers) use ICN to disseminate the collected information in the network.

These examples, drawn from published work in this research, illustrate use cases of ICN and DTN to improve network services in the operational scenarios discussed in Section 4.1. The SDN controller configures network policies to create ICN and DTN slices (see *Figure 10*). While ICN is used as the main mechanism for data dissemination, DTN is used to connect different network segments, such as synchronization between vehicles or WSN data collection. Both paradigms enable two variables of C2 agility: DI and PI. Unfortunately, they cannot currently replace IP networks, but only as islands within a larger IP infrastructure (ZURANIEWSKI *et al.*, 2017).

Figure 9 - Deployment Example: SDN combining DTN and ICN



Source: The author

In this work, SDN therefore integrates multiple ICN or DTN islands into the IoBT. Within each island, ICN (or DTN) functions would leverage the data plane for efficient **information distribution** and **human-thing interaction**. SDN also optimizes data flow based on network (and node) status (bandwidth, data size, channel latency, channel availability, etc.) and improves security with the mechanisms described in 4.6.

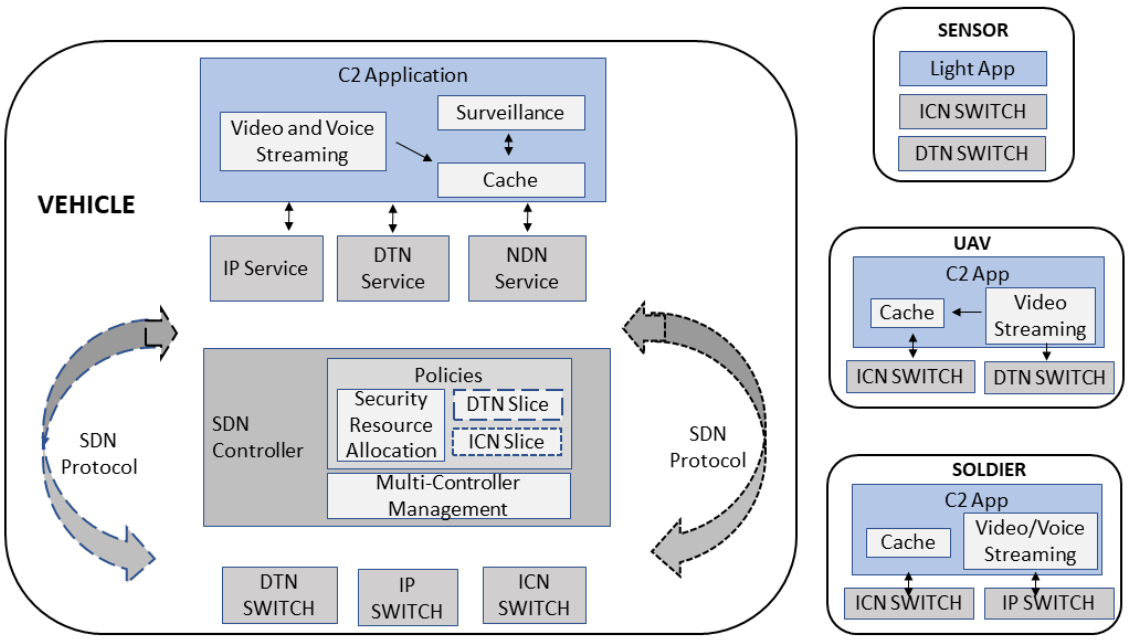
By controlling which node is allowed to send/receive data, SDN can act on the third dimension of C2 space, namely the **allocation of decision rights**. For example, the SDN controller can determine who is allowed to change the drone's flight parameters (e.g., the

commander, any troop member, only troop members near the drone, another drone). Another aspect is that the network administrator can change this assignment during mission execution to respond to changes in mission or circumstances, providing the technical means to implement C2 agility in the network.

4.3 High-Level Architecture

The proposed architecture aims to optimize IoBT communication parameters such as latency, channel bandwidth, interruption and node failure, and topology changes while providing a mechanism for cybersecurity of the solution. As explained in the previous sections, the idea is to orchestrate the network with SDN as the core and use its programmability to select the optimal network protocols for the given current operational requirements and use its flexibility to change them according to the operational and network status. The general architecture is shown in *Figure 10*. Using the taxonomy stipulated by (MAHMUD *et al.*, 2021), the architecture has three dimensions: *Network Orchestration*, *Application*, and *Security*.

Figure 10 - High Level Architecture



Source: The author

The heart of the *orchestration dimension* is the SDN controller. This dimension must provide scalability and fault tolerance. Due to the *ad hoc* nature and mobility of battlefield networks, sophisticated placement of SDN controllers is required. In this work, the SDN controllers run in the vehicles, the nodes that meet the processing and performance requirements. Thus, they are physically distributed to provide scalability and fault tolerance. In the first published papers of this research, replicated controllers (daemons) were used to remove the single point of failure. However, this method had many drawbacks, such as the use of passive controllers that become active only when the main controller fails.

To cope with these constraints, logically centralized multi-controller management is currently used. All controllers have the same responsibilities. They share the load equally to improve network performance, but for the layer below, they are considered as a single controller. They are aware of changes in the network and share the information thanks to network synchronization. A specific implementation is discussed in Chapter 5.

This physically distributed and logically centralized controller programs network behavior with policies and characterizes the traffic to which the policy applies through traffic identifiers (DTN and ICN slices). These policies are used to enforce the required quality of service through resource allocation and to protect the network from security threats. The controller transforms the high-level policies (received from the C2 application) into simple SDN flows by evaluating the stored mapping of high-level traffic identifiers (from the application) and known traffic flows (DTN or ICN), while periodically updating the data flow rules according to new conditions.

This orchestration enables concurrent IP, NDN, and DTN flows, establishing logical islands with different network protocols depending on the C2 application requirements. For example, two different applications running on the same node have different network requirements that can be better met by different protocols (as seen in the deployment examples). In this way, DTN, ICN or IP can be used in the same nodes.

The *orchestration dimension* provides network services to the *application dimension*, which in turn provides C2 services to users in the form of operation planning and monitoring. The application dimension consists of SDN-enabled (for surveillance applications) or legacy applications (video and voice streaming) with different QoS and security requirements. In *Figure 10*, legacy applications use the network services provided by the IP stack, while SDN-enabled applications use differentiated SDN services via a controller.

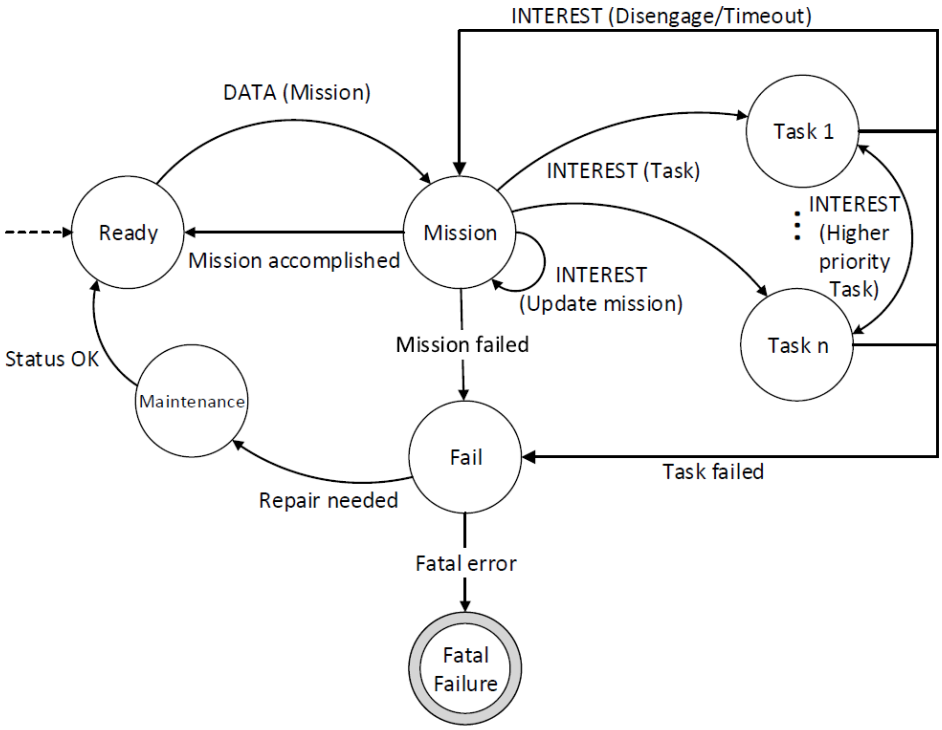
The *cybersecurity dimension* permeates the other dimensions with the concept of defense in depth, which prevents vulnerabilities in one dimension from affecting another. Three

layers of security are used: *hardware, network software* and *application software*. User access to the system is authenticated, while communication between the SDN controller and the application is encrypted, protecting the application dimension. Flow control on the switches (managed by the controllers) prevents unauthorized requests from being forwarded to the receivers, protecting the orchestration dimension.

4.3.1 Architecture Dynamics

All assets in the network have a general behavior that can be summarized in the state diagram shown in *Figure 11*. In the application dimension, the commander sets the mission, assets and AoI, as well as hierarchies and priorities. He can also create tasks and update mission parameters. A general node can be in the states MISSION, TASK, READY, MAINTENANCE, and FAIL. MISSION can be configured in advance or set by the controller. The node remains in the state MISSION until it receives an INTEREST message with a TASK.

Figure 11 - Assets State Diagram



Source: (STOCCHERO *et al.*, 2023)

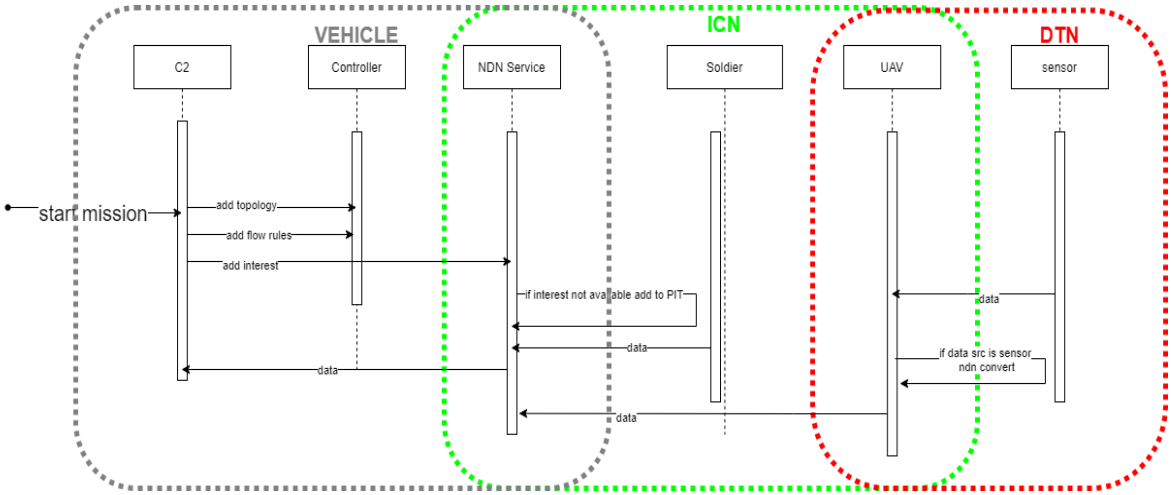
The parameters of TASK update the C2 application running on the node by transitioning to a new state specified by TASK. The system remains in this state until a higher priority TASK

is received or the current task is completed (as in the example shown in *Figure 7*). Depending on the will of the commander, two cases can occur: 1) it forces the interruption of the current activity in favor of the recently arrived one; 2) it places the current task in a priority queue that continues after the highest priority task is completed.

The nodes may also be in a failed state. In this case, the SDN controller must track the network state or attempt to recover overloaded nodes. After a maintenance state, a failed node may return to the ready state. In the event of a fatal failure, the node is considered lost.

The architecture must be able to handle both high-end nodes, such as the vehicles running the SDN controller, and low-end nodes, such as dismounted soldiers, UAVs, and sensors. The interactions of these heterogeneous nodes are shown in *Figure 12*, where the controller orchestrates both the ICN and DTN data services to meet application requirements.

Figure 12 - Node Interactions Sequence Diagram



Source: The author

To this end, an SDN controller running in the platoon commander vehicle can orchestrate all communications in the IoBT. It also connects the IoBT to the rest of the combat cloud, such as remote ground support or even a larger and more capable drone flying nearby. The controller acts as an enabler for quick route adjustments to properly implement the authorization to receive/transmit data from/to other specific nodes.

In the scenarios examined, troops on foot can receive imagery of threats near them, but not imagery from remote locations. The commander, on the other hand, receives data from all locations. However, due to the dynamic nature of the mission, the network topology can change very quickly and some nodes can become overloaded. The controller implements a network policy that determines which data should be forwarded and which should be discarded or stored

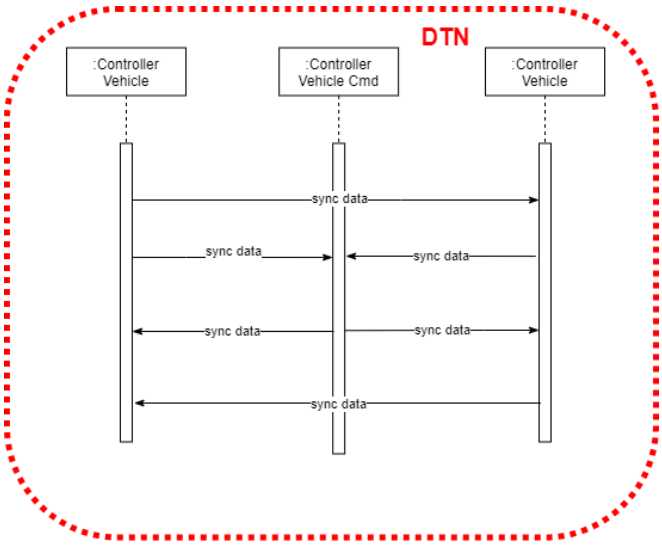
locally for later transmission. In the particular situation of imminent contact with hostile groups, it is more important for soldiers on foot to receive images of the incoming threat than for the commander. In this case, the network may need to switch quickly by sending the data to the soldiers first and caching it for later transmission to the commander (*Figure 7*).

ICN (NDN service) increases the efficiency of data transmission by caching data in nodes closer to consumers to reduce delays and increase availability, while avoiding a direct connection with the data producer, since each replicated node can serve the consumer. Efficient forwarding strategies can be used to locate services in such dynamic scenarios and provide flexible routing schemes (ZHANG *et al.*, 2021).

However, it is important to note that the NDN service keeps the data plane in operation when the node disconnects from its controller. The C2 application uses its direct connection with the ICN switch to send INTERESTs and receive DATA messages using the previous rules (hierarchy, priority, and filter) until the connection with a controller is restored and the rules are updated. In addition, the ICN caching, and replication mechanism should ensure that the information is preserved in the network even if the source node is disconnected, taking into account the expiration time set by the C2 application. This scheme is suitable for the contested BN scenario (which is prone to various spectrum attacks) because it does not require a stable connection to the controller all times and uses the ICN caching mechanism for data transfer.

DTN is the mechanism for integrating disjointed network segments, such as the WSN deployed in the field, whose data is collected by the UAVs. *Figure 13* shows the communication between the vehicles. In this case, DTN is used because the vehicles are not always connected due to their distance from each other.

Figure 13 – Inter Vehicle synchronization

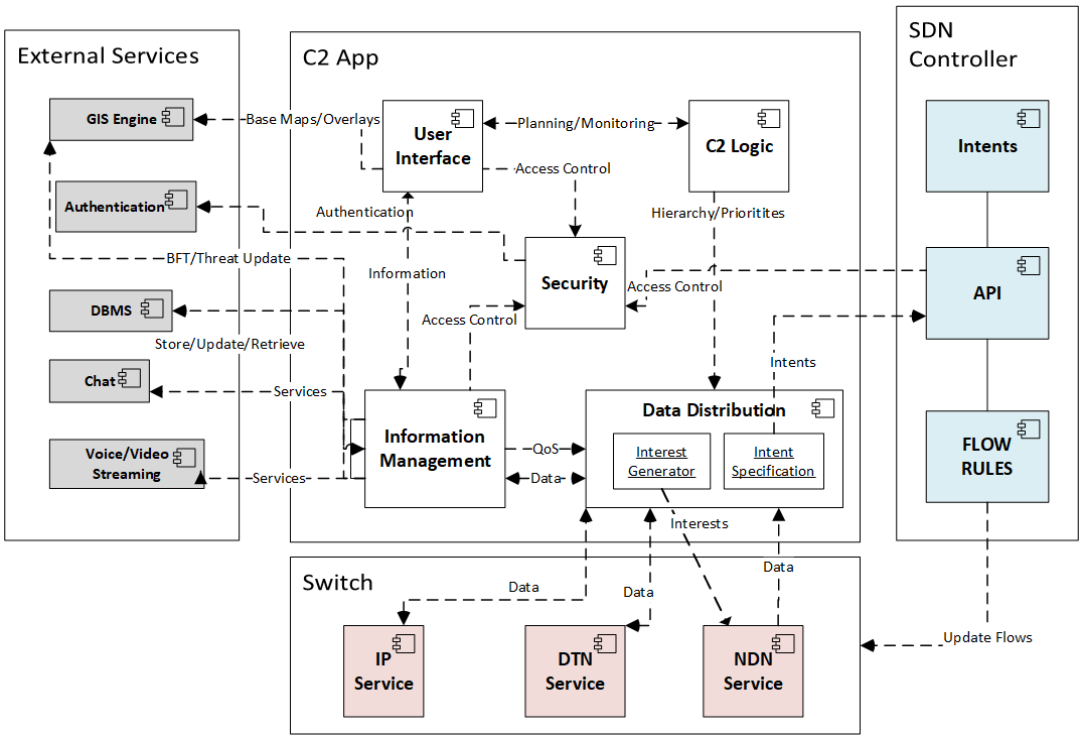


Source: Author

4.4 Application Dimension

The primary objective of a C2 application is to assist the commander in managing his personnel and available resources to accomplish his assigned tasks of organizing, coordinating, directing, and supervising the force. The application dimension of the proposed architecture is an ensemble of the C2 application and open source external standard services such as the geographic information system (map service), authentication service, database management system, and chat, voice, and video streaming services. The C2 application itself consists of several components that provide the necessary functions and interfaces with the SDN controller and asset´s switches. These main components deal with C2 logic, information management, data distribution, security, and user interface, as shown in *Figure 14*.

Figure 14 - C2 Application Components



Source: The author

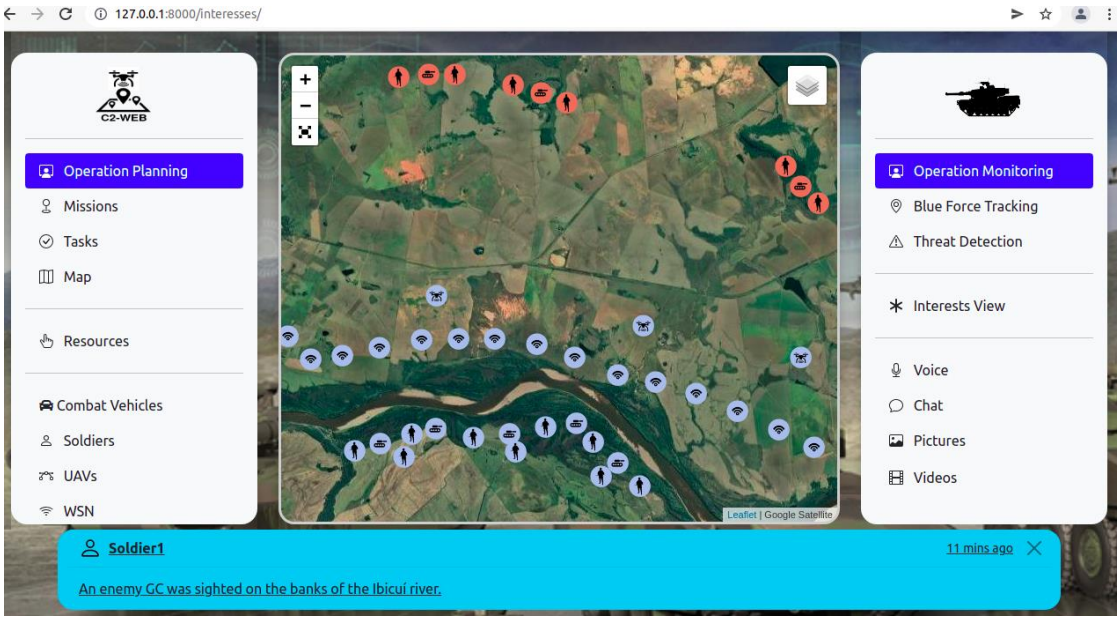
4.4.1 User Interface

The user interface provides the user with all necessary services and interprets his input for the other components:

- it determines the operating mode of the ***C2 logic*** component, and the functionality for the user changes depending on the operating mode;
- access control for the user is defined by interaction with the ***security*** component;
- it operates based on base-maps provided by the external GIS engine. GIS also provides overlays with the common operational picture (COP) that displays BFT and threat updates; and
- Operational input from the user is sent to the ***information management*** component, which sends back updated information from the network, as well as external services such as chat and video streaming;

Figure 15 shows the user interface of the proposed C2, which relies on information received from underlying services by the information management component and base-maps and overlays from the geographic information system (Figure 14).

Figure 15 - Application Graphical User Interface



Source: The author

4.4.2 C2 Logic

The C2 logic defines the business-level relationships required for the mission, depending on the mode of operation: *Planning* or *Monitoring*.

4.4.2.1 Operation Planning Mode

In the *planning* mode, the commander determines the assets, their priority and hierarchy, the communication patterns (who sends/receives data from whom), and the areas of interest. All these aspects are handled by the C2 Logic component using the following data entities:

- MISSION the planned activity to achieve an operational objective, defining the resources to be used, the Areas of Interest (AoI), the hierarchy of nodes', and the priorities of the AoI;
- TASK ad hoc activity defined by the C2 application as part of a MISSION. Typically, it is used to acquire specific targets or, if necessary, to act on enemy;
- Map: Displays the map of Area of operations;
- Resources: personnel, facilities, equipment, and materiel under the control of a commander. The following C2 System resources are shown in *Figure 15*:
    - Combat vehicles: armored (tracked or wheeled) and light reconnaissance vehicles, depending on MISSION type, terrain, and enemy combat power, among other aspects;
    - Soldiers: Mounted (crew depending on vehicle type) or dismounted;
    - UAVs: quantity and model depend on the mission planning;
    - WSN: to trigger alarms to the C2 application. These nodes are connected to the C2 application via a DTN by UAVs flying in the vicinity.

4.4.2.2 Operation Monitoring Mode

Operation Monitoring implements the services necessary for Situation Awareness. It keeps track of:

- Blue Forces Tracking (BFT): is updated when the application receives the status of the deployment assets: the icons of the deployment assets' appear in different colors depending on their status, according to the standards of the doctrine;
- Threat detection: The C2 application receives alerts, reports, images and videos and displays them to the user. When the user identifies and categorizes a threat, this information is distributed over the network;
- Interest view, with data services for end users:
    - Voice: VoIP in real time with troops on the battlefield;
    - Chat: Interaction with troops via text messages;
    - Image: Retrieval of images of a specific asset;
    - Videos: Retrieval of streaming videos from a specific asset.

4.4.3 Information Management

One of the most challenging aspects of the C2 application is *information management*. The application must handle both time sensitive and elastic data elements:

- Time-sensitive
  - Chat, Voice and video streaming;
  - BFT, Threats, Alerts and status messages (predefined messages)
- Elastic data
  - Transmission of image/video files;
  - Tasks and Reports

Therefore, application logic must coordinate these various data elements with network services to provide the required level of QoS and security. This component receives requests from the ***user interface*** for external chat and streaming services, while also performing BFT and threat updates for the GIS engine. Since all this data is time-sensitive, but the user also needs a good experience with elastic data, this component is responsible for setting the QoS levels for the ***Data Distribution*** component and receiving the corresponding data. It is also responsible for performing all external database operations. After receiving data from the network, this component commits it to the database. The information available to the user interface always comes from the updated database.

4.4.4 Data Distribution

The *data distribution* component receives the desired QoS levels from ***Information Management*** and hierarchies and priorities from ***C2 Logic***. It communicates with the SDN controller through its Northbound API to specify high-level network policies defined as **intents** (HAN *et al.*, 2016). These are common concepts and terminologies that can be interpreted by the application and underlying network service providers. The SDN controller receives the intent specifications and converts them into actionable operations in the network environment via intent compilation. In the event of network failures, the intent is automatically recompiled, and the network reconfigured to restore the desired connectivity without any intervention by the application or the user.

The application can set up and change intents for node hierarchy and data priority depending on operational circumstances. The controller will also transparently handle changes in the underlying network to comply with the higher-level policy. The data distribution also

communicates directly with network device switches for data-driven services using flow rules set by the controller. To receive updated content from the network, it generates INTERESTS and receives DATA messages to/from the NDN service. It also processes DTN and IP packets according to the QoS parameters set by the information management component.

4.4.5 Security

The security component in the application dimension implements access control policies using an external authentication service. It is a part of the security dimension of the architecture embedded in the application. Local authentication is required to access the application that accesses the controller. This component prevents attacks on the application and northbound communication (between the application and the controller). Such attacks aim to discover vulnerabilities in access control and application isolation (AGADAKOS *et al.*, 2019), either by allowing unauthorized people to access information or by sending inconsistent flow rules to the controller. The security aspect is discussed in more detail in Section 4.6.

4.4.6 External Services

The application uses external services with an open standard. A GIS engine and a DBMS handle all data-related operations for the ***information management*** and ***user interface*** components. A standard chat and voice/video streaming service also support these components with time-critical data, while authentication services are provided for the ***security*** component.

4.4.6.1 GIS Engine:

- Base-maps: uploaded before operation, consisting of multiple layers such as terrain, rivers, cities, as well as satellite imagery, etc;
- Real-time information layers: updated information from network nodes (alerts, videos and photos) and their current location.

4.4.6.2 DBMS:

- Resources: updated parameters for each resource (ammunition type and quantity, available weapons and sensors, etc.)
- Threats: updated data on known parameters (combat power, etc.);

4.5 Orchestration Dimension

As mentioned earlier, the core of the orchestration dimension is the SDN controller running in the vehicles that meet the processing and performance requirements. The master controller runs in the commander vehicle and controls all resources (drones, WSN, and soldiers) within range, forming an SDN-controlled ICN island. The SDN controllers in the other vehicles are slaves running an SDN daemon component that continuously monitors their current network segment to detect active SDN controllers.

If no SDN controller is found, the SDN daemon starts a leader election algorithm for the new (and disconnected) ICN island in case the master controller is not connected or fails. A simple election algorithm was used to determine the first daemon that detects the unavailability of the commander' through heartbeat messages as the new leader. A more efficient algorithm could be developed in future work. In the extreme case, an isolated vehicle would control only its own resources within range. A specific implementation of the orchestration dimension is presented in Chapter 5.

The Controller performs three main functions:

- Hierarchy control: sets the node hierarchy and message priority based on the parameters received from the C2 application (MISSION). It can filter unauthorized requests from nodes with lower hierarchy as a security measure. This feature can also be used for network scalability if the number of nodes in the IoBT suddenly increases. Hierarchy control can be implemented by associating the ICN device type with named data prefixes that it can request;
- Security: implemented as the network software level, as seen in section 4.6;
- Path Optimization: Modifies ICN and DTN forwarding logic based on the global SDN vision of the network and defines rules for forwarding packets over links with better parameters and lower costs, such as available bandwidth and delay.

These functions specify the required QoS mechanisms. QoS rules set the behavior of individual switches and the priorities between flows from different facilities. Such rules are based on any field of the incoming packet and are used to create different queues for each traffic category. Each queue can have a different transmission rate depending on the priority level.

Path optimization requires the analysis of several factors: 1) the distance between nodes, 2) the existing data flows between nodes, and 3) the caching capability of each node. The Dijkstras algorithm was used to calculate the best path based on these weighted factors. The second factor is given a higher weight in the evaluation because a congested path can affect the

time-critical nature of operational information, and the caching capability defines where a particular data flow can or cannot pass. If a node in a path does not have enough cache for the data flow, it must be sent to another location.

The SDN controller connects to the C2 application via an API (see *Figure 14*) to obtain high-level network policies (hierarchy, priorities, and security) via *intents*. The controller converts these intents into flow rules for NDN, DTN, and IP stacks and prioritizes traffic according to the application's requirements. The flow rules written in the P4 language are used as pipelines for the switches in the other nodes. Unlike Openflow, which uses fixed functions, P4 programs activate only the functions that are really needed in resource-constrained nodes.

Using the most recently rules, the pipeline implemented in P4 remains active even in situations where a node loses connectivity to its controller, so the data plane continues to function. Once connectivity is restored and the controller has changed rules, they are updated. For example, nodes out of range of their controller continue to forward packets with the current priorities until they enter the controller's communication range and receive updated priorities.

UAV-to-vehicle communications rely on NDN for content transmission, i.e., dissemination of tactical threat information, while unattended ground sensors use DTN for communications when a drone is flying nearby, since in this case, connection failures are expected to occur by design. Voice communications between troops on foot and vehicles depend on the IP stack, as this situation is not suitable for NDN or DTN and requires end-to-end node real-time connections. The P4 language defines a pipeline treatment in the data plane for each data type, either NDN, IP, or DTN. This allows the three different paradigms to interoperate using the same infrastructure:

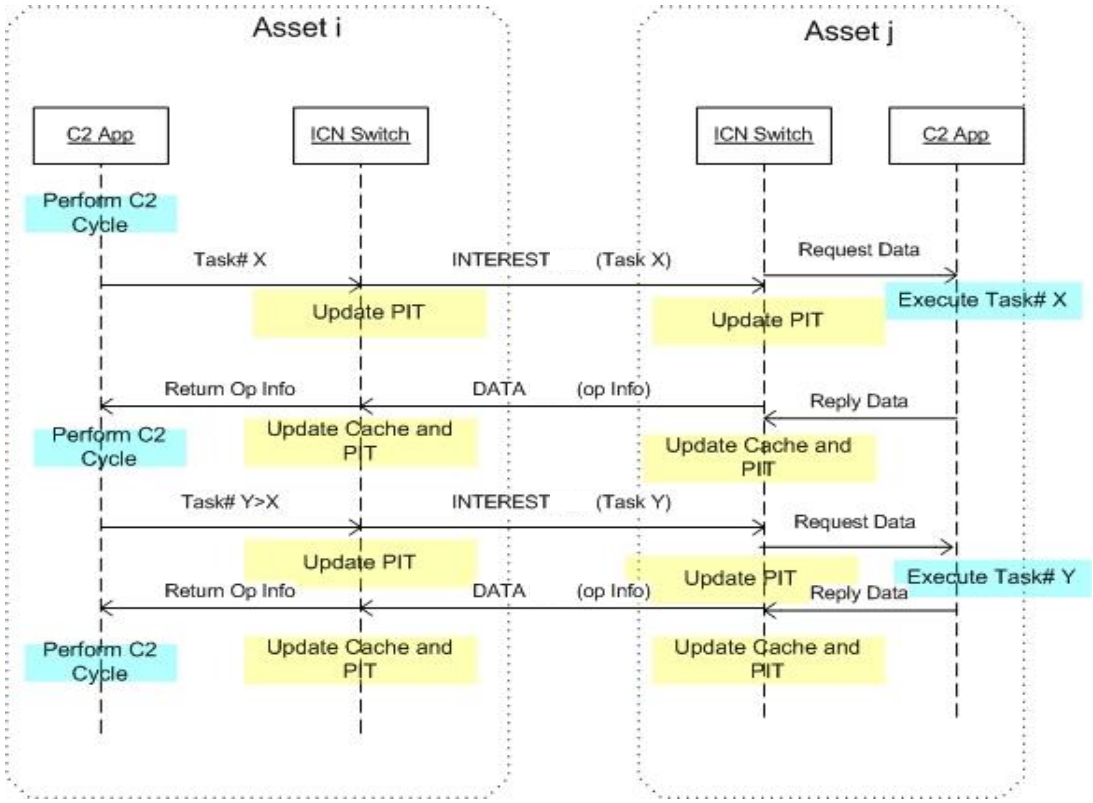
- IP flows: used when real-time end-to-end communications are required, especially for soldier-to-soldier and vehicle-to-soldier VOIP communications, highest priority;
- DTN flows: used when node disruptions are expected to synchronize data between vehicles (other controllers) as well as UAVS as they approach wireless sensor nodes scattered around the battlefield; and
- NDN flows: for dissemination of data, e.g., UAV data from surveillance cameras, as well as data produced by soldiers, such as text, photos, and videos.

NDN flows disseminate information between assets according to flow rules that are constantly updated by the controller to optimize data transmission. However, the SDN controller does not intervene in the actual data exchange process. As mentioned earlier, the NDN service keeps the data plane running when the node disconnects from its controller.

The C2 application running in any asset (except for the sensors, which are extremely passive nodes) can generate TASKS, which propagates through the network as INTERESTS, as shown in *Figure 16*. The application receives operational information back in the form of detections (sensors), images and videos (UAVs/troops), pattern reports (UAVs), and operational reports (troops).

To deal with topology changes due to mobility, this work uses a particle swarm optimization (PSO) based approach described in (DAPPER E SILVA *et al.*, 2019). More efficient techniques may be explored in future works. Moreover, the modularity of the proposed architecture allows the adoption of alternative approaches for a given deployment. However, it is important to note that IoBT is about collecting current information from specific areas of interest. If the asset moves out of the area, its contents are no longer needed (but any cached data is still useful during the validity of its timestamp). Another node entering the area could be a new source, and the SDN controller tracks all assets in the AOI by their coordinates.

Figure 16 - Data Plane: NDN Flows



Source: The author

4.6 Cyber Security Dimension

This work implements cybersecurity at three levels using the concept of Defense in Depth. The architecture creates sequential barriers to prevent unauthorized access and protect assets. Even in the event of an attack, the protection mechanism enables responsible parties to take the necessary measures and act accordingly. The levels of defense are described as follows:

4.6.1 Hardware Level

The hardware level prevents theft or tampering of information when a device is hijacked. A secure area stores the most important data that can be handled by trusted software. Two different environments are created: the trusted execution environment (TEE), with hardware components that process sensitive data and are accessible only to trusted software (private keys are stored here); and the rich execution environment (REE), an unreliable area with the rest of the hardware components accessible to untrusted software (GONZALEZ *et al.*, 2016).

4.6.2 Network Software Level

This level protects the Control Plane, Southbound Communication, and Data Plane (J. C. NIKOUE, 2019) and prevents unauthorized access to the IoBT. Due to the larger attack surface, three techniques have been implemented: Anti-DDoS attacks in the switches to prevent them from damaging the data plane and control plane; fingerprint technology to prevent attacks on the southbound communication and unauthorized access to the network; and a distributed ledger mechanism to prevent SPoF, taking into account sabotage or compromise by adversarial activities.

4.6.3 Application Software Level

The final layer of security will act at the application level. This is done through authentication at login in the software, in the relationship between the application and the network controller, and within the controller through flow rules implemented in the switches. The goal of implementing this layer is to prevent some attacks that target the application itself and the northbound communication (between the application and the controller). According to (J. C. NIKOUE, 2019), attacks on the application target access control and application isolation

flaws. The former can allow unauthorized individuals to access information, while the latter can result in inconsistent data flow rules being sent to the controller.

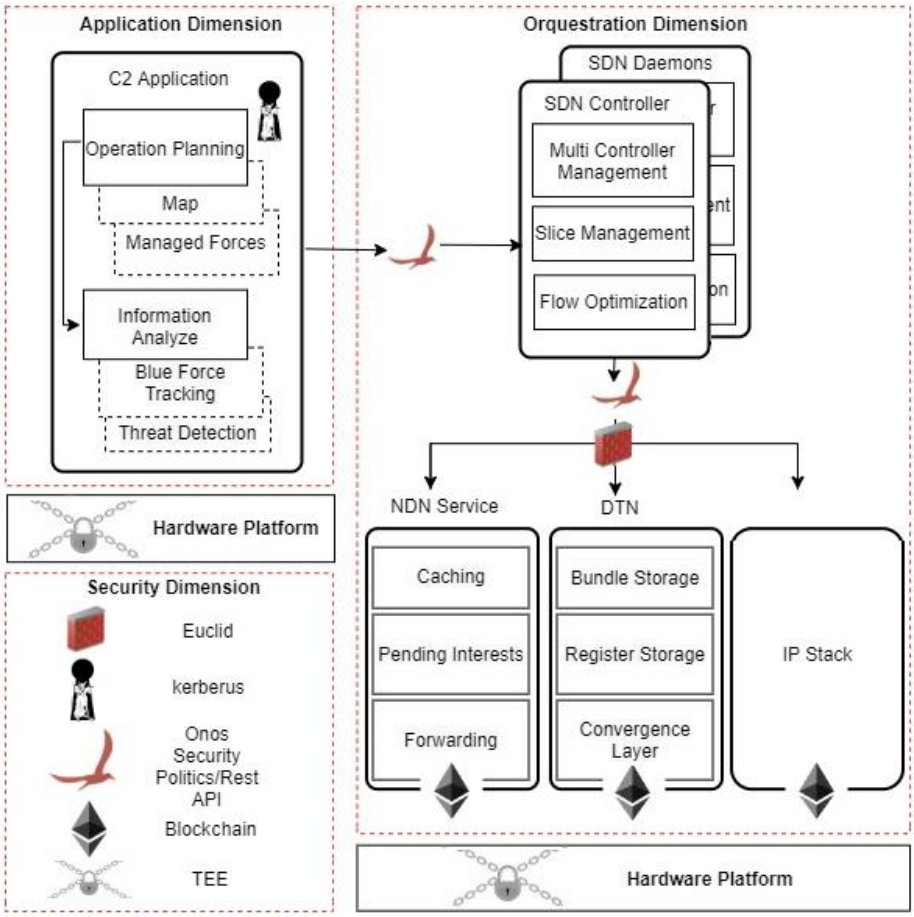
Also according to (J. C. NIKOUE, 2019) , attacks on northbound communication target weak or non-existent authentication between the controller and the application, which can lead to spoofing. In this case, the proposed solution was to use an application programming interface of the SDN controller. This interface provides secure authentication between the application and the controller. In addition, the flow rules implemented by the controller in the switches make it impossible for unauthorized network elements to make requests to the controller.

Chapter 5 explains an implementation of the security mechanisms in detail.

5 AN EXAMPLE IMPLEMENTATION OF THE ARCHITECTURE

This chapter presents an example implementation of the architecture using specific technologies for the orchestration and security dimensions, as shown in *Figure 17*.

Figure 17 - Architecture Implementation



Source: (STOCCHERO *et al.*, 2022)

The Open Network Operating System (ONOS) (VACHUSKA, [*s. d.*]) is used for *network orchestration* because it is a fully distributed network operating system ideal for multi-domain wide-area networks (WAN). ONOS provides scalability, fault tolerance with a distributed core that maintains a global view of each controller instance running on different servers (vehicles), and optimizes data flow for different forwarding protocols such as DTN, ICN, and IP. This dimension provides network services to the *application dimension*, which in turn provides C2 services to users in the form of operation planning and monitoring. A multi-layer *cybersecurity dimension* crosses these dimensions.

The *cybersecurity dimension* permeates the other dimensions with the concept of defense in depth, which prevents vulnerabilities in one dimension from affecting another. In

Figure 17, user access to the system is authenticated with Kerberos, while communication between ONOS and the application is encrypted, protecting the application dimension. All network access is via Physical Unclonable Functions (PUF) keys, while flow control on switches (managed by controllers) prevents unauthorized requests from being forwarded to receivers, protecting the orchestration dimension. Blockchain Ethereum secures inter-group communication and avoids Simple Point of Failure (SPoF).

5.1 Orchestration Dimension

The ONOS SDN controller can operate as a cluster system (multiple controller instances), providing fault tolerance if one instance fails and real-time updates to the network without affecting network traffic. ONOS connects to the C2 application via an API (Representational State Transfer - REST), as shown in *Figure 18* to obtain high-level network policies via *Intents*.

ONOS translates these intents into flow rules for NDN, DTN, and IP stacks and prioritizes traffic according to application requirements. ONOS then deploys the pipelines written in P4 for switches with the STRATUM operating system on the other nodes.

ONOS enables the following key features (ADDAD *et al.*, 2018), as seen in *Figure 19*:

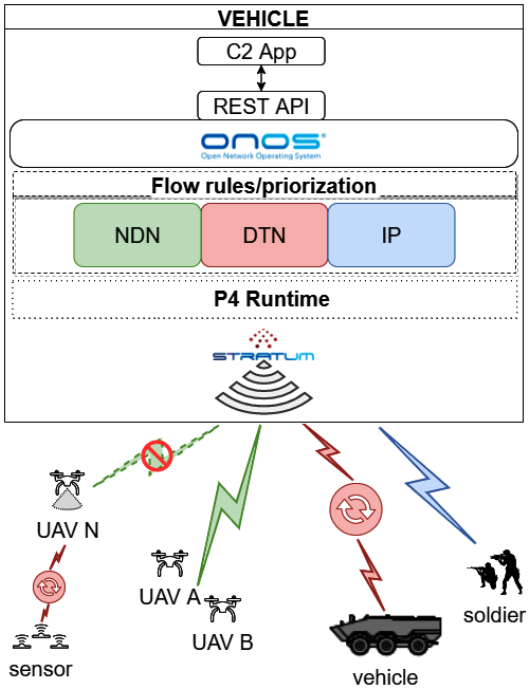
- **Distributed core** for maximum scalability and modularity, managing the control plane in a multi-instance cluster. Due to the distributed nature of its core, ONOS does not need an election mechanism to select cluster leaders;
- **Southbound abstractions** that provide protocol plugins for communicating with network devices and isolate the core from communication protocols; and
- **Northbound abstractions** that provide configuration and management services for C2 application development.

The Northbound interface connects to the C2 application to receive intents. ONOS Intent Framework translates them into descriptions of **network resources** and **constraints**, as well as **criteria** and **instructions for traffic selection** (ALALMAEI *et al.*, 2020) via *Intent Compilation*, into *Installable Intents*, which are Actionable Operations to ONOS.

These actions are then executed by the Intent installation process, which results in a set of flow rules being installed on one or more selected switches in the network as pipelines written in P4. The ONOS core is responsible for maintaining network state, interacting with network devices through the southbound APIs, and providing services to the application

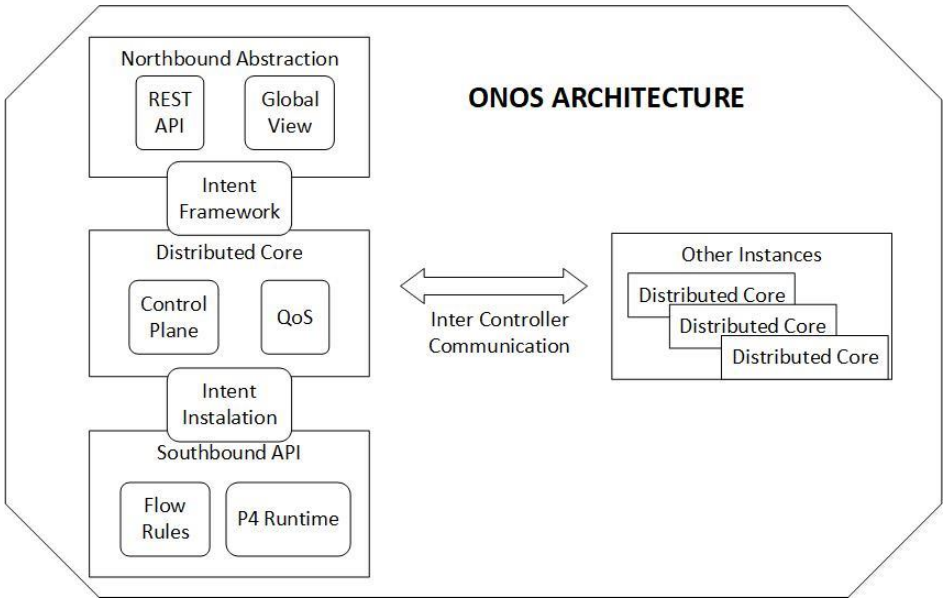
through the northbound APIs: flow rules abstract the protocol to configure forwarding logic in the devices, and the intent framework abstracts high-level application policies.

Figure 18 - ONOS Orchestration



Source: (STOCCHERO *et al.*, 2022)

Figure 19 - ONOS Architecture



Source: The author

5.1.1 Northbound Abstraction

Application data can be exchanged via ONOS REST APIs using the JavaScript Object Notation (JSON) format, which is an easy-to-read language for both humans and machines, similar to calling a web service with a specific request over the HTTP protocol. This web service forwards this request to the ONOS core after converting it into a request in a query format that the core can understand.

It is divided into two main parts: the *global network view*, which provides the C2 application with up-to-date information about the network (hosts, switches, and links), and the *intent framework*, which allows the C2 application to set network policies and determine under what circumstances they apply.

5.1.2 Intent Framework

The intent framework provided by ONOS (SANVITO *et al.*, 2018) receives high-level policy specifications from the application and translates them into the corresponding low-level device details. ONOS defines the intent as an immutable model object that is requested by the application and used to modify the behavior of the network with respect to its operation.

After an intent is submitted, it is immediately (but asynchronously) moved to the compile phase, then to the install phase, and finally to the installed state. An intent can also be withdrawn if it is no longer needed by the application. The intent can also be transparently recompiled in the event of topology events (e.g., network failures) to achieve the desired goal. The compilation process takes into account the network state and the behavior required by the C2 application to create one or more flow rules that depend on the application requirements, node type, and interaction pattern and aim to optimize the network and prioritize messages.

Intents may be bound to a specific traffic subset (expressed as a set of specific values for packet header fields) and associated with a treatment (a set of actions applied to all packets processed by the intent). There are several types of intents: Built-in ONOS intents are *point-to-point*, *single-to-multi-point*, *multi-to-single-point* and *host-to-host*.

In addition, intents can specify a number of constraints that limit the results of compilation. For example, the resulting paths must traverse a certain number of nodes or reserve a certain amount of bandwidth. Upon receiving a high-level intent, the system computes the configuration for the devices on the network without requiring any further request to the user.

To achieve this flexibility and independence, ONOS maintains network topology information and uses a path calculation engine to determine the best path through the network to satisfy the application request. Once the path is computed, a generic device-centric abstraction is created for each element in that path. This device policy is translated into a specific configuration that depends on the capabilities of the device, as shown in 5.1.4.

5.1.3 ONOS Distributed Core

The distributed core is responsible for managing network resources and satisfies SDN control plane requirements. The core enables consistency among multiple controller instances in a cluster and implements network requirements for agility, resiliency, and fault tolerance based on the C2 application and bandwidth requirements. In this architecture, each vehicle has an embedded SDN controller that is logically connected to neighboring vehicles in a full mesh (on a peer-to-peer basis) and uses a specific TCP port (9876) for interaction.

The system uses a flat architecture (RAVURI *et al.*, 2022), where controllers use an east-west bound interface to communicate with their peers. The intent service uses this communication to set up data flows. The intent subsystem of the first controller publishes the event of establishing a new flow to the distributed core of ONOS. All other controllers that have access to the distributed core configure their part of the network accordingly.

Network applications within the operating system assist controllers in handling traffic flow and maintaining a global view of the network (ADDAD *et al.*, 2018). In addition, the operating system supports dynamic updating of applications without interrupting traffic flow. ONOS, like other network operating systems, is modular and fault-tolerant. Moreover, the extension and consolidation of control planes based on network operating systems are comparatively easier and less time-consuming. However, such control planes must be deployed locally for synchronization, which may not be practical for military use cases requiring cross-network communication.

5.1.4 Southbound Abstraction

The southbound abstraction is the communication channel between the data plane and the control plane. It consists of protocols that specify how the SDN controller should forward information to the data plane, the installation of flow entries, and the insertion of forwarding rules into switches.

In this work, ONOS categorizes the data streams according to different rules for each source through the intent installation process. NDN, DTN, and IP data streams have a specific pipeline defined in the P4 language (P4.ORG, [*s. d.*]) that is installed and deployed on switches or APs through P4Runtime, the SDN control plane interface for controlling forwarding behavior at runtime. This mechanism enables the creation of network islands with different priority levels. P4Runtime populates the forwarding tables and manipulates packet-processing behavior based on a P4 program in a hardware-independent manner.

This is achieved by translating the device-specific high-level concept (intent) (input port 1, output port 2) into device-specific payloads (P4Runtime table entries). This translation could also be done for Open Flow Rules or OpenConfig-based configurations, since the Intent APIs are protocol and transport independent. These payloads define matching criteria for stream metadata. Combined, these rulesets enable different forwarding strategies based on specific actions applied to each flow. This ensures that data streams are optimized for network conditions and operational requirements and prioritized accordingly.

In request-response communication protocols such as those used in this work, a data plane node (IoBT sensor) that requires a command from the control plane (vehicle) sends a request to the corresponding instance of the distributed ONOS core. In response, the controller sends the required instructions to the requesting IoBT node. The opposite is the case when the vehicle needs state information from the data plane.

Protocol oblivious forwarding makes the format of a packet transparent to data plane nodes. In this case, data plane nodes extract and assemble key features from the packet header to perform flow table lookups based on controller instructions. This gives the data plane the flexibility to support new protocols and forwarding requirements.

5.2 Cyber Security Dimension

As explained in Chapter 4, the proposed architecture uses security mechanisms at three levels. An implementation of these mechanisms is shown in *Figure 20* and *Figure 21*.

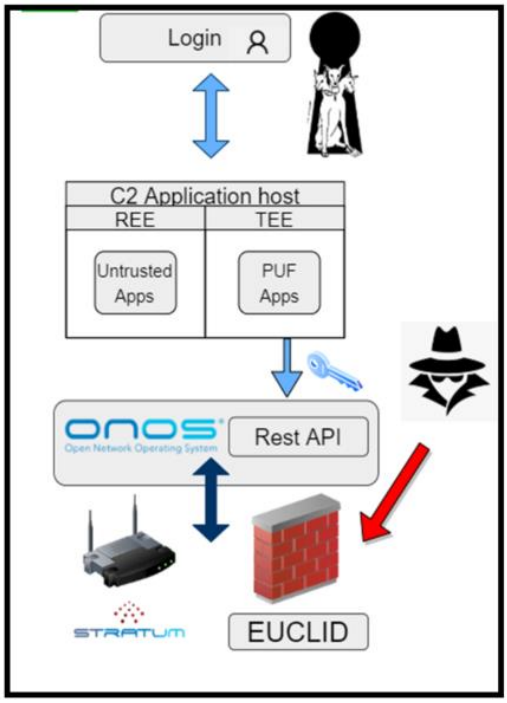
5.2.1 Hardware Level

As can be seen in the middle of *Figure 20*, two different environments are created. The trusted execution environment (TEE), with hardware components that process sensitive data and are only accessible to trusted software (private keys are stored here). And the rich execution

environment (REE), an unreliable area with the rest of the hardware components accessible to untrusted software (GONZALEZ *et al.*, 2016).

To implement this solution, the ARM TrustZone was used to allow the definition of a dynamic security area defined by CPU, memory, and peripherals that can be modified at runtime. Due to the memory limitations of the IoBT nodes, only private keys were stored. They were only used for authentication and to verify the authenticity of the devices.

Figure 20 - Cybersecurity Mechanisms (Hardware/Application Software Levels)



Source: (STOCCHERO *et al.*, 2022)

5.2.2 Network Software Level

This level protects the Control Plane, Southbound Communication, and Data Plane and prevents unauthorized access to the IoBT. Due to the larger attack surface, three techniques have been implemented: EUCLID (ILHA *et al.*, 2021), PUFs (LIOU; LIN, 2021), and Ethereum blockchain.

The EUCLID project uses custom counting sketches that perform computationally intensive mathematical operations using a memory-optimized longer-prefix match table (LPM) to prevent DDoS and flooding attacks directly from the switches and prevent them from damaging the data and control plan, as shown in *Figure 20* below.

PUFs are a hardware-based technology for device fingerprinting. Two PUF architectures are used to verify the authenticity of nodes to prevent attacks on southbound communications and unauthorized access to the network. The first architecture is the Arbiter PUF, which is used for bidirectional verification and is suitable for high-end nodes such as radars, UAVs, wearables, and armored vehicles. The Arbiter PUF offers several advantages over other existing architectures. It has a more robust PUF architecture than the alternatives and is partially resistant to machine learning attacks.

The second architecture is the ring oscillator PUF, which is used for unidirectional verification of low-end devices such as unattended ground sensors. Only the high-end nodes connected to them (e.g., UAVs flying over them to collect their data) verify the low-end devices. With these structures in place, it is possible to both avoid attacks on southern communications and prevent unauthorized nodes from gaining access to the network.

The last technique, shown in *Figure 21*, is directly related to the previous one and uses the Ethereum blockchain to replicate node credentials and avoid SPoF, taking into account sabotage or compromise by adversarial activities. As in (LIOU; LIN, 2021), groups are formed. Trust ID (group ID) and credentials are stored in the blockchain through smart contracts. In case of an attack or prolonged communication, nodes from one group can join other groups if they are registered in a blockchain database, which enables cross-group communication between nodes and avoids SPoF.

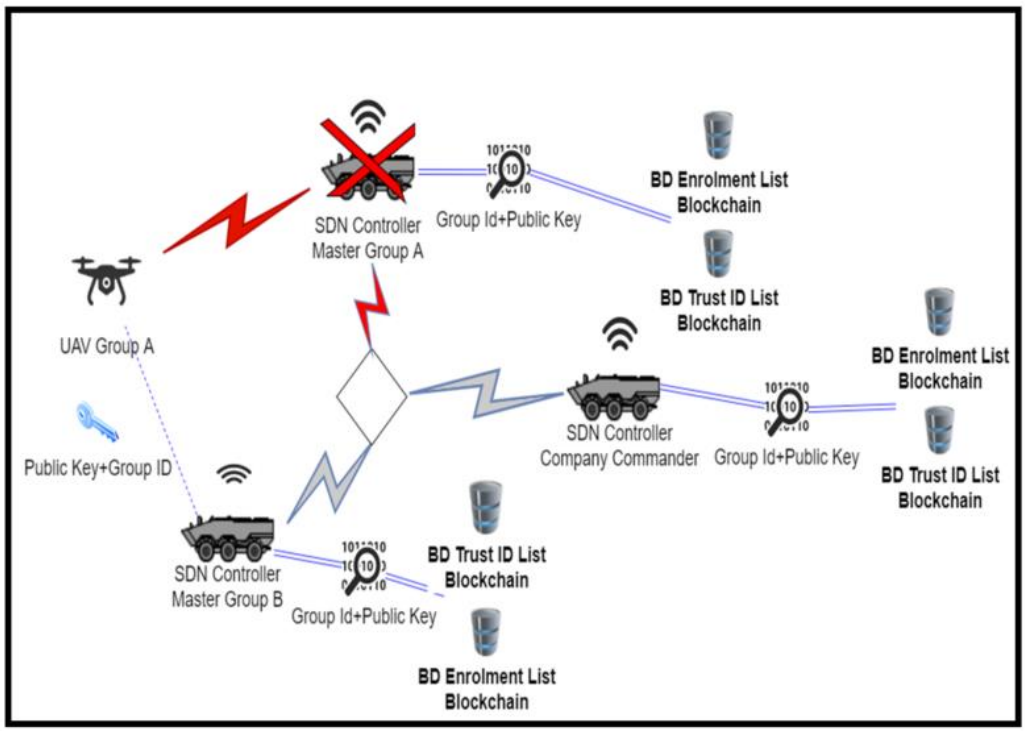
The reasons for choosing Blockchain, and Ethereum in particular, are as follows: a) it is open source; b) its utility properties facilitate customization to the intended system; c) the immutability of the blockchain can prevent data tampering; and d) the inherent decentralization (public consensus) of the blockchain, which benefits system availability, avoids the single point of failure (SPoF) problem. Ethereum also offers PoS (proof-of-stake) consensus algorithms that support a higher number of transactions per second and require fewer resources (ASIF; GHANEM; IRVINE, 2021).

5.2.3 Application Software Level

To prevent unauthorized people from accessing information, the login authentication in the proposed software uses Kerberos software. In this approach, users of the software make authentication requests to the company commander's vehicle where the Kerberos server is installed. This prevents elements that are more exposed on the battlefield from storing

credentials, thus preventing possible credential theft. This authentication only needs to be performed at system startup, so bandwidth is not constantly consumed.

Figure 21 - Cybersecurity Mechanisms (Network Software Level)



Source: : (STOCCHERO *et al.*, 2022)

To prevent attacks on the northbound communication (controller and application), an application programming interface of the ONOS controller called Rest API was used. This interface provides secure authentication between the application and the controller. In addition, the flow rules implemented by ONOS in the switches make it impossible for unauthorized network elements to make requests to the controller.

6 APPROPRIATENESS AND FEASIBILITY EVALUATION

As mentioned in the previous chapters, the proposed architecture has been evaluated in several use cases published in different papers, depending on the maturity of the research. This chapter presents these evaluations and discusses the appropriateness of the proposed combination of network technologies for the deployment scenarios, as well as the feasibility of the architecture with implications for potential failures.

6.1 General Experiment Configuration

Although each published paper had a specific configuration, the experiments generally used network emulators to test the desired functionality. All interference was mitigated by the proper choice of scenarios so that the simulations ran without bottlenecks in terms of CPU and memory.

6.1.1 Emulation Layer

Among SDN emulators, Mininet (LANTZ, BOB; HELLER, BRANDON; MCKEOWN, 2010) is the most popular because it is lightweight, boots faster, and offers higher scalability. In the literature (ARAHUNASHI; SCHOLAR, 2019), Mininet has been used to evaluate SDN controllers, improve controller adaptability, manage data flow, and create integrated SDN environments.

(ZHAO *et al.*, 2019), for example, used Mininet to create an integrated software-defined battlefield network testbed. Their goal was to provide an emulation environment for tactical scenarios and for evaluating the corresponding communication performance to demonstrate the effectiveness of their proposed solution. The testbed supports common traffic generators such as iPerf, D-ITG, MGEN, and a distributed SDN controller model.

Due to the characteristics of IoBT, evaluations in this work were performed using the MininetWifi software-defined wireless network emulator (FONTES *et al.*, 2015), which is a fork of Mininet. It adds wireless emulation capabilities and integrates mobility models used to simulate the movement of IoBT nodes to emulate realistic wireless scenarios through virtual wireless stations and access points (APs).

In addition, the NDN Forwarding Daemon (NFD) Library (AFANASYEV *et al.*, 2015) was used to implement all functions related to *name-based* communication. NFD enables the forwarding of INTERESTs messages until a matching DATA message is found, either in the

local cache of an intermediate node or in the producer itself. This library runs on each node and maintains point-to-point connections with other hosts through interfaces called faces, which are types of a communication mask to enable *name* forwarding.

To integrate these ICN switches with the SDN controller, content-based rules were mapped as host-based rules by making minor adjustments to Openflow. Scripts allowed the controller to set up forwarding rules for the switches using its global view of the network. The host-switch pair enables all NFD functionality directly, through point-to-point connections with the addresses configured for each prefix (name) in the forwarding tables. If a rule for a particular name is not found in the FIB, the switch sends a PACKET _ IN to the controller. The controller filters the data streams and manages the node hierarchy based on the parameters defined by the C2 application. The controller also manages the translation of ICN, DTN, and IP packets to and from the Military Cloud.

6.1.2 SDN Controller

SDN configuration was demonstrated by integrating an external SDN controller into the emulation environment. **Ryu** (COMMUNITY, [*s. d.*]), a component-based software defined networking framework, and **Floodlight** (VEDHAPRIYAVADHANA; FRANCY IRUDAYA RANI; THEEPA, 2018) were used for the initial experiments. Ryu enables the creation of new network management and control applications and supports several protocols for managing network devices, including OpenFlow (MCKEOWN *et al.*, 2008). OpenFlow is the most commonly used protocol to implement the SDN paradigm and has several libraries and features that facilitate the process of topology discovery and installation of flows on switches.

The Ryu controller used Openflow 1.3 with a modified version of Simple Switch 13 to build a centralized spanning tree and define each switch's forwarding table. To initialize each ICN switch with the controller and achieve connectivity between all devices, the responsibility for forwarding packets to the access points in each device was transferred to the NFD library. To continuously update the NDN forwarding tables, the Named Data Link State Routing Protocol (NLSR) was used.

However, the Ryu framework creates new management and network applications through logically centralized control and APIs, while this work envisions a distributed system. Therefore, the Ryu controller was later replaced by ONOS, which is ideal for distributed multidomain networks and provides fault tolerance and scalability for the entire military cloud.

6.2 Published results and discussion

This section presents several proofs of concept that have been performed to increase the maturity of the architecture and evaluate its benefits and potential drawbacks. These results have been compiled in several published papers, listed in Table 12 and summarized below.

6.2.1 SDN for Tactical Edge Networks

The first network paradigm considered in this research was SDN, even before the official start of the PhD study. Given the mobility patterns and ad hoc nature of military networks, especially at the edge, network configuration and management must be dynamic enough that manual configuration is not a viable solution to operational challenges. For this reason, SDN was explored as a possible middleware between the applications and the carrier networks, allowing the network to configure itself to provide the required quality of service.

The initial experiments used a scenario with ground vehicles running surveillance applications that require video streaming to identify possible enemy threats from a fleet of UAVs, as shown in *Figure 8*. The commander should receive video captured by the UAVs and use other vehicles as relays if necessary. Since there is a risk of losing important events about enemy movements, this transmission must meet strict requirements.

To guarantee QoS in this multi-hop network, SDN (Ryu Controller with Openflow) was used to select optimal connection paths for forwarding data, avoiding congested connections and choosing paths with fewer hops. The results were published in (ZACARIAS *et al.*, 2017b) and (ZACARIAS *et al.*, 2018). Experiments used multiple simultaneous video streams, starting with one video stream and increasing the number of streams according to the parameters in *Table 3*. To measure the performance of the network, several metrics were collected, such as the *video playback start time* (corresponding to the time it takes for the player to start playing), the *number of interruptions* (playback is temporarily stopped), and the *total duration of interruptions* (sum of the duration of all interruptions). The collected measurements were used to predict the subjective experience quality indicator (Mean Opinion Score - MOS).

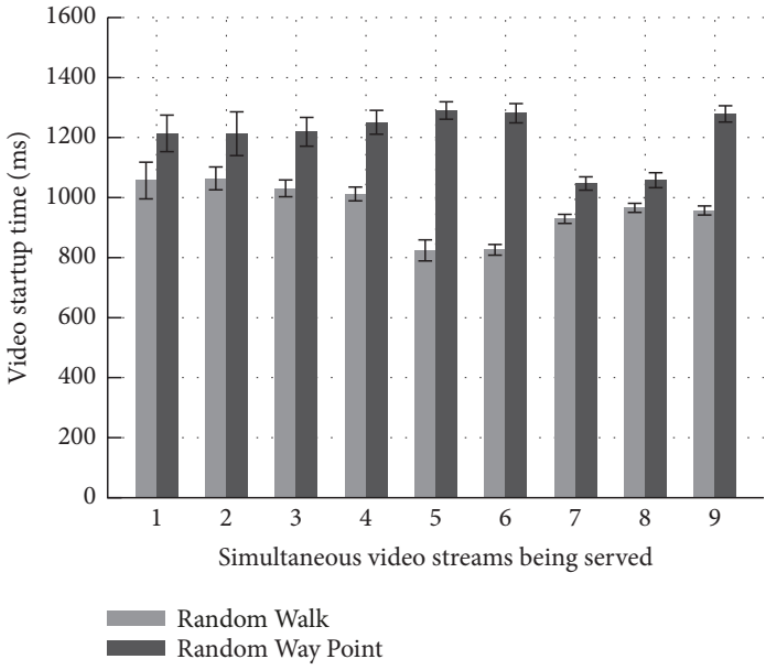
The results show that programmable networks can be successfully applied to heterogeneous networks, interference-prone networks, and networks with opportunistic connections common in military applications. In all experiments conducted, *the video playback startup time* was maintained at an acceptable value for a video surveillance application, starting within an acceptable time interval even in the worst case, as shown in Figure 22.

Table 3 - SDN Experiments parameters

Parameter	Used Value
Number of UAVs	9
Number of ground vehicles	9
Total simulation area	4 km x 4 km
UAV moving area	4 km x 1 km
UAV communication radius (R_u)	360 m
Ground communication radius (R_v)	650 m
Mobility models	Random Walk, Random Waypoint
Video Size	960x540 pixels
Video frame rate	30 fps
Video codec	H.264
Video length	60 seconds
Number of runs per number of streams been served	33 runs

Source: (ZACARIAS *et al.*, 2017b)

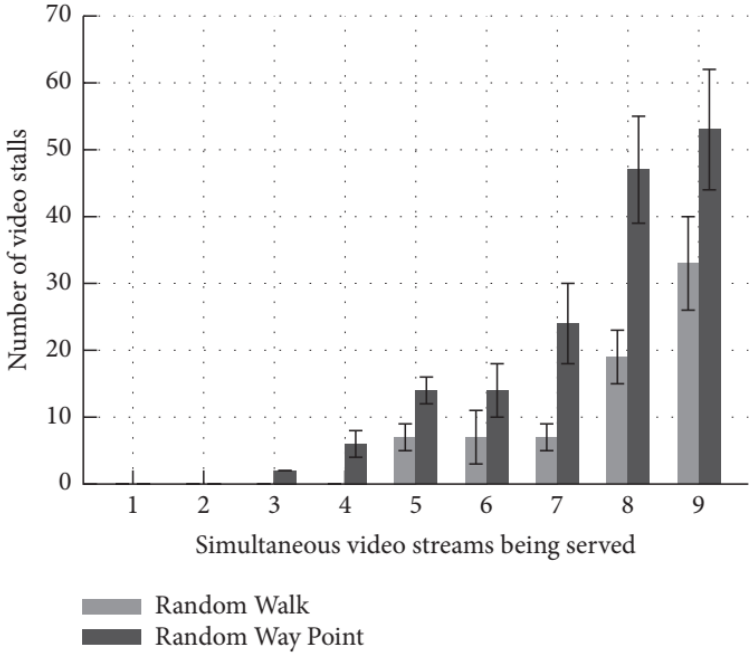
Figure 22 - Video playback start time measured in the experiment



Source: (ZACARIAS *et al.*, 2018)

The number of video interruptions (stalls) was strongly influenced by the number of simultaneous video streams, as shown in Figure 23. The different streams competing for the available bandwidth resulted in resource contention, especially in the context of the wireless network environment. The simplest mobility model, i.e., *Random Walk*, resulted in more stable wireless links. The total stall length for the *Random Walk* mobility model shown in Figure 24 has slightly better values compared to the *Random Waypoint* mobility model, but even for the latter the total stall length for up to eight simultaneous video streams is satisfactory.

Figure 23 - Number of interruptions measured in the experiment



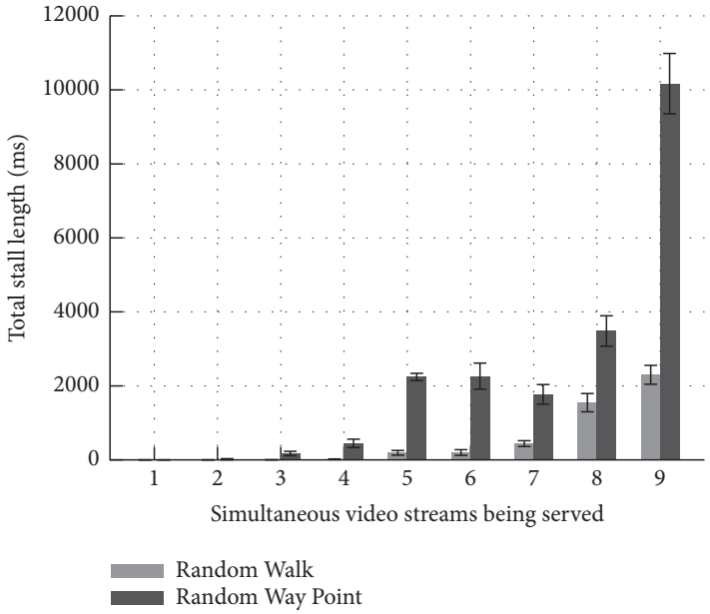
Source: (ZACARIAS *et al.*, 2018)

Figure 25 shows the average length per video frame, i.e., the time the image remains frozen per video frame. Even in the worst case (nine video streams competing for network bandwidth), the time of each video stand is less than 1 second, which is an acceptable value for surveillance and reconnaissance missions that use video streaming.

From the user perspective, according to the predicted MOS values in Figure 26 , good results were observed when using up to four simultaneous video streams with the *random walk* mobility model. Of note is the video frame rate used in the experiment (~30 fps). Normally, video surveillance does not require such a high frame rate, and with a lower video frame rate, the streams should be less resource intensive. So a better user experience is expected, leading

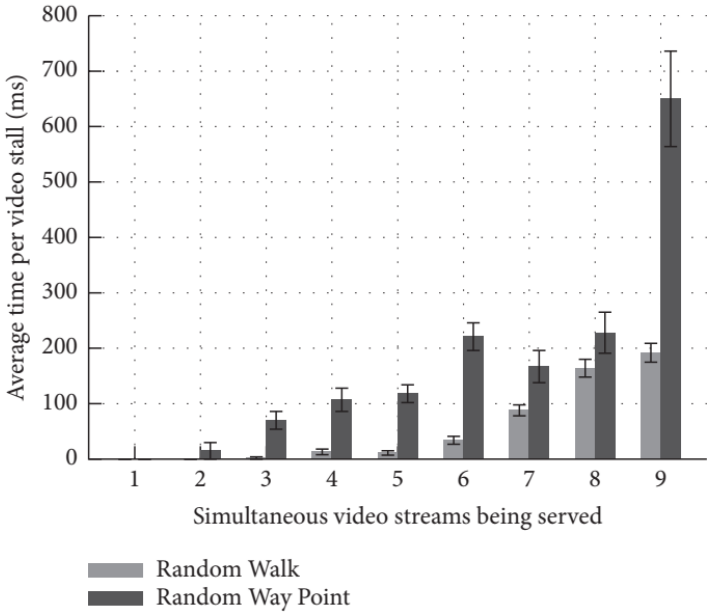
to a higher MOS. The same is true for the video resolutions chosen for the simulations. The video size chosen is in the range of values used in the military context. However, smaller video sizes are used in many military applications.

Figure 24 - Total duration of interruptions in the experiment



Source: (ZACARIAS *et al.*, 2018)

Figure 25 - Average time per video stall measured in the experiment

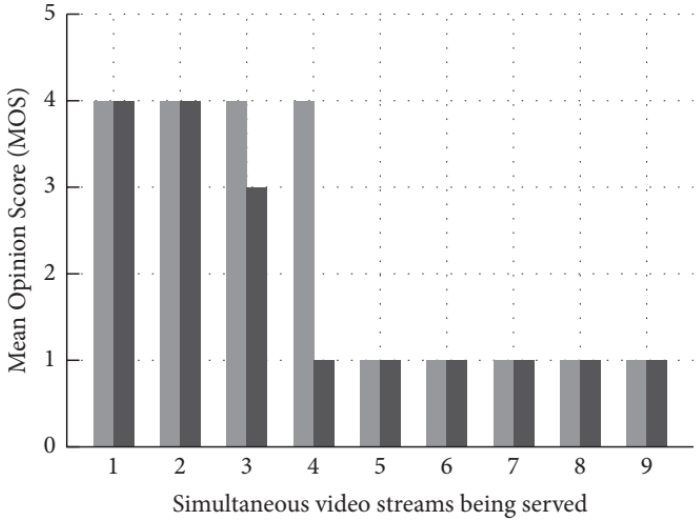


Source: (ZACARIAS *et al.*, 2018)

These results indicated a future path for the continuation of the research, as the insertion of an SDN controller into the relay network also enables the implementation of DTN protocols. As mentioned earlier in this work, such protocols leverage the SDN controller's global knowledge of the network to optimize the transmission and routing of packets on opportunistic links created by the movement of UAVs.

Information Centric Networking (ICN) could also be considered in conjunction with SDN so that mechanisms for caching video in intermediate nodes could be explored to enable faster delivery of video to multiple requesting nodes.

Figure 26 - Predicted mean opinion score (MOS) perceived by the user



Source: (ZACARIAS *et al.*, 2018)

6.2.2 SDN integration with DTN

As the next step to increase maturity in the network, the software-defined approach of the previous experiments was combined with a delay-tolerant approach to optimize network performance, using the ability of DTN to handle link outages.

The operational scenario studied reflected the IoBT, with heterogeneous nodes and data flows with different QoS requirements, depending on the applications (file transfer, video streaming, short sensor messages, etc) being elastic or non-elastic. The IoBT consists of several small networks (partitions) that occasionally and opportunistically interconnect with each other.

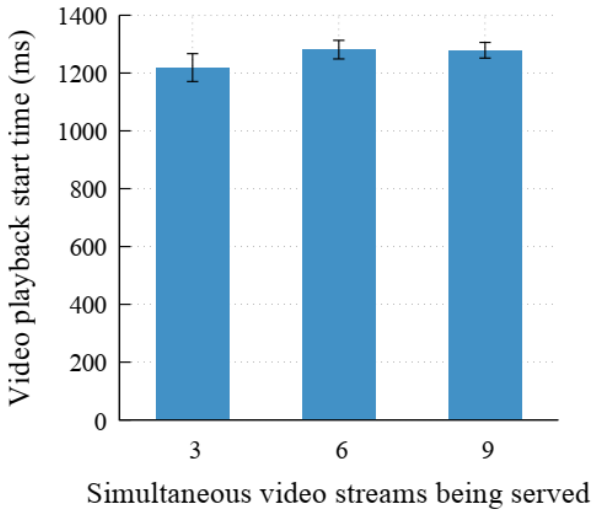
The experiments in this new development were published in (ZACARIAS *et al.*, 2017a), with the same parameters in *Table 3* (in this case only the random waypoint mobility

model was used) . Multiple SDN controllers work in master/slave configuration, exchanging link information with the DTN orchestrator, which schedules data transmission between the DTN nodes and the most suitable routing protocol, in accordance with internal decision algorithms and the data exchanged with the SDN controller. The flexibility provided by SDN makes it easier to change the routing protocol while the network remains running.

Two use cases for intra and inter network partitions were simulated. The first handles simultaneous video streaming in an intra-network partition scenario, where some nodes need to act as relay, due to the distance between source and destination. There is a need for these intermediate nodes to share bandwidth among many data flows from different video sources.

In this use case, the SDN Controller select appropriate paths to forward video flows that comply with the C2 application requirements and at the same time employs mechanisms to ensure fairness among concurrent flows or evenly distribute the data streams among redundant links. Similar metrics as in 6.2.1 were used and results (*Figure 27*, *Figure 28*, *Figure 29*) showed that the SDN Controller selected the best routes and the degradation observed was due to the inherent limits of the used physical layer. Without the controller, the degradation was observed with just one stream and, for many simultaneous streams, completely unacceptable.

Figure 27 - Playback start time considering simultaneous video streams

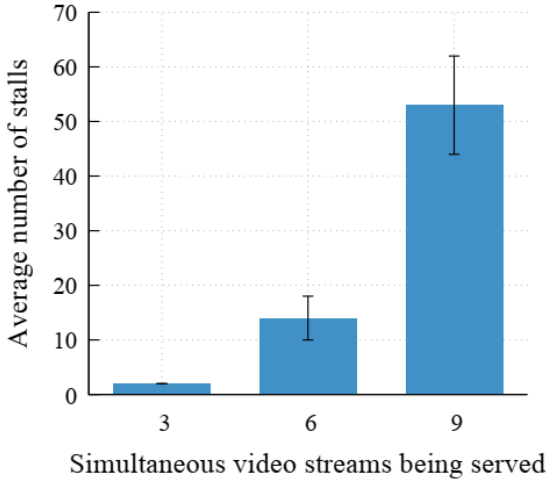


Source: (ZACARIAS *et al.*, 2017a)

The second use case add the DTN paradigm to the network. It deals with communication between heterogeneous nodes in the IoBT, spread across different partitions. The WSN generates data belonging to an elastic application in one isolated partition. When an UAV flies

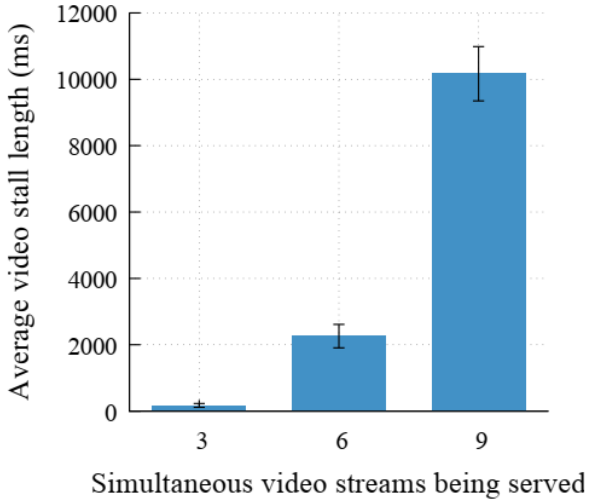
nearby and establishes a link with the WSN, the SDN controller notifies the DTN orchestrator about the data being received and optimizes the data transfer between the DTN nodes and the UAV, by selecting links that are capable of proceeding with the data transfer. The same process is repeated across multiple partitions until the destination node is reached. Results show that the proposed approach is able to successfully deliver 100 percent of the data sent by the WSN. On the other hand, the same network without the proposed SDN-DTN approach has results with a success rate for delivered data ranging from 44 to 55 percent.

Figure 28 - Average number video stalls considering simultaneous video streams



Source: (ZACARIAS *et al.*, 2017a)

Figure 29 - Average video stall length considering simultaneous video streams



Source: (ZACARIAS *et al.*, 2017a)

6.2.3 SDN Integration with ICN

After the initial experiments with SDN and DTN, this research aimed to investigate ICN as the main mechanism for IoBT data distribution. The operational scenario used for this experiment is shown in *Figure 6*, where troops face the challenge of locating adversaries in a widely dispersed and dynamic urban environment. In this situation, the IoBT should be able to distribute data in a timely manner over a congested network.

The initial experiment and its results were published in (LEAL *et al.*, 2019). The architecture studied consisted of three main components: the C2 application, the SDN controller, and the ICN switches. The C2 application generates missions and tasks that are distributed to the ICN switches. The SDN controller defines the node hierarchy, priorities, and optimizes paths according to the requirements of the application.

Two use cases with the parameters shown in *Table 4*. The first dealt with the data filtering mechanisms implemented by the SDN controller to avoid congestion while maintaining network uptime. The second use case dealt with ICN.

Table 4 - SDN combined with ICN experiment parameters

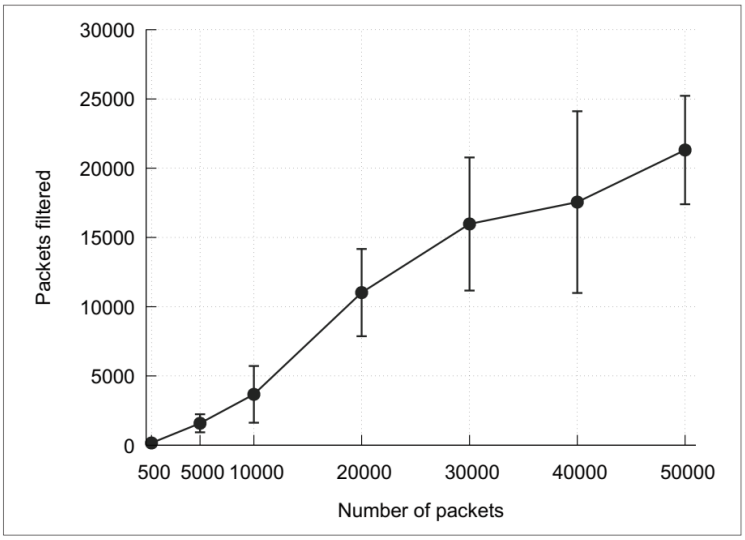
Parameters	**Use Case # 1 Filtering**	**Use Case #2 Caching**
Number of Nodes	3 vehicles 5 drones 10 soldiers 20 sensors	1 vehicle 18 drones 1 soldier
Connection Type	UDP	INTEREST/DATA
Data Rate	Up to 54 Mbps	Up to 54 Mbps
Sensor Max data size	10 KB	-
Number of runs	100	100
Simulator	Mininet	Mininet/Mini-NDN
Controller	Floodlight	-
SDN Protocol	Openflow 1.3	-
Link Delay	-	5ms

Source: ((LEAL *et al.*, 2019)

The first use case results are shown in *Figure 30*. They show that the proposed mechanism scales and is able to filter out irrelevant packets based on hierarchy control (which

nodes are more valuable to the network) and packet type. Both mechanisms resulted in a significant reduction in useless traffic.

Figure 30 - Number of packets filtered for different numbers of packets sent in



Source: ((LEAL *et al.*, 2019)

In the second use case, the commander sends INTERESTS for videos of threats to which the producers (UAVs) respond with DATA. The objective is to evaluate the ability of the ICN caching mechanism to provide the required data for follow-up requests (for the same data). Performance was evaluated in terms of latency compared to a non-ICN network, and the results showed a significant improvement in latency (about 35%).

Based on these initial results, the architecture was studied in detail and the combination of SDN and ICN was explored. The results were recently published in (STOCCHERO *et al.*, 2023). In this work, the operational scenario shown in *Figure 1* was investigated. The C^2 application runs on any device of the network and defines the asset hierarchy and priorities. It also generates the INTEREST (defined in Table 5) and DATA (defined in *Table 6*) packets, which are processed by the SDN controller and propagated through ICN. Each node is a producer and consumer of DATA, which are generated based on period, payload, and Time to Live (TTL). The first two are control messages with greater frequency and reduced TTL and payload, while the last four responses are to operational INTERESTS.

The TTL field in the INTEREST and DATA message is important to prevent cache aging. This parameter is populated based on the operational value of the generated data and determines how long the DATA messages are stored in the local cache and how long the INTEREST is valid on the network.

To evaluate the proposed SDN-ICN architecture, several simulation experiments were performed in two use cases, according to the parameters listed in *Table 7*. In the first set of experiments, the performance of the current architecture was compared with a SDN-IP approach and an IP-only approach. In the second use case, the simulations were performed in a dynamic scenario to check the topology management. Nodes are initially positioned in the simulated environment according to a random uniform distribution and follow a random mobility pattern (with uniform distribution). As an example of the network topology, *Figure 31* shows a snapshot of the network with 40 nodes.

Table 5 - ICN Operational INTERESTS

INTEREST	INTEREST#3	INTEREST #4	INTEREST #5	INTEREST #6
Name	/mission/X/ movement	/task/Y/situation/ V	/task/Y/situation/H	/task/Y/engage/ human/z

Source: (STOCCHERO *et al.*, 2023)

Table 6 - ICN Data Messages

Type	DATA#1	DATA#2	DATA#3	DATA#4	DATA#5	DATA#6
Period	30s	30s	60s	150s	300s	300s
TTL	5s	30s	60s	150s	300s	300s
Payload	1KB	5KB	100KB	1MB	5MB	10MB

Source: (STOCCHERO *et al.*, 2023)

The experimental simulations were configured using modules in the form of classes and Python files, as seen in *Figure 32*.

The **Topology** class is the central processing component responsible for the most important functions in the experiments:

- reading and interpreting topology files, generating commands to create scenarios in MininetWifi;
- discovering the best paths between nodes and filling the forwarding tables of all NFD applications, since this class has a global view of the network; and
- Performing all configurations related to device caches in the NFD and creating priority levels in the SDN controller Class

The **Data Manager** class creates and reads the data queue to be transferred during the experiment, and arranges the data to be transferred by timestamp to facilitate processing. The **Experiment** class deals with the execution of the experiment. It contains the list of devices

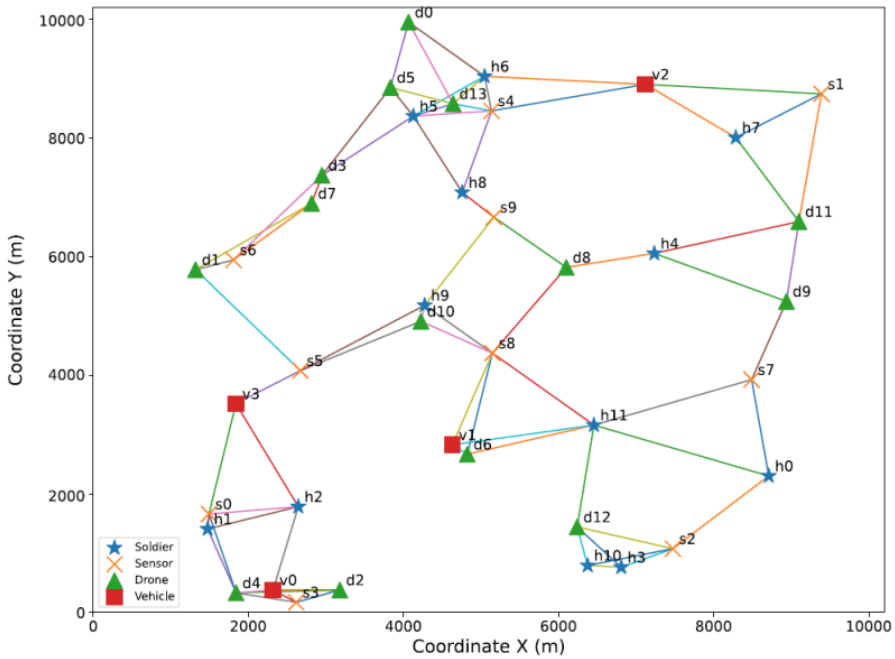
and all the data to be sent. From these files, it reads the queue and creates instances of all producers and consumers. The **main** component connects all the other components by setting some parameters such as the topology file, the number of runs and the network type (IP, SDN-IP and SDN-ICN) and determines the appropriate controller for each type of network.

Table 7 - SDN combined with ICN Experiment parameters

Total Nodes	**10 nodes**	**20 nodes**	**30 nodes**	**40 nodes**
Sensors	2	4	8	10
UAVs	4	8	12	14
Humans	3	6	9	12
Vehicles	2	4	4	5
Link Delay	2ms			
Available cache	Sensor: 468.65 MB UAV: 312.5 MB Human: 312.5 MB Vehicle: 625 MB			

Source: (STOCCHERO *et al.*, 2023)

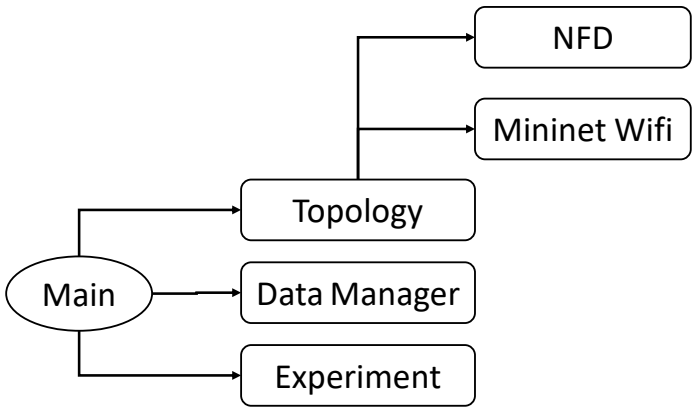
Figure 31 - Network Topology Snapshot Example (40 nodes)



Source: (STOCCHERO *et al.*, 2023)

In the first use case, the performance of the proposed architecture SDN-ICN was evaluated against a SDN-IP approach and an IP-only approach using similar network topologies with a variable number of nodes (10 to 40). Experiments (20 to 30 repetitions) were performed for each configuration to measure end-to-end delay and network load.

Figure 32 - SDN ICN Experiment Architecture



Source: (STOCCHERO *et al.*, 2023)

The IP-only configuration does not have a cache and is supported by a logically decentralized controller that defines routing tables individually without having a global view of the network. The SDN-IP approach also does not have a local cache, but does have a logically centralized controller. This controller determines routes using a spanning tree algorithm based on the network topology. The SDN-ICN configuration has the same controller as SDN-IP and a local cache in the devices with the capacity shown in *Table 7*. The cache capacity was determined based on the maximum capacity supported by the NFD library (corresponding to 625 MB) for the vehicles, 75% for sensors, and 50% for drones and soldiers.

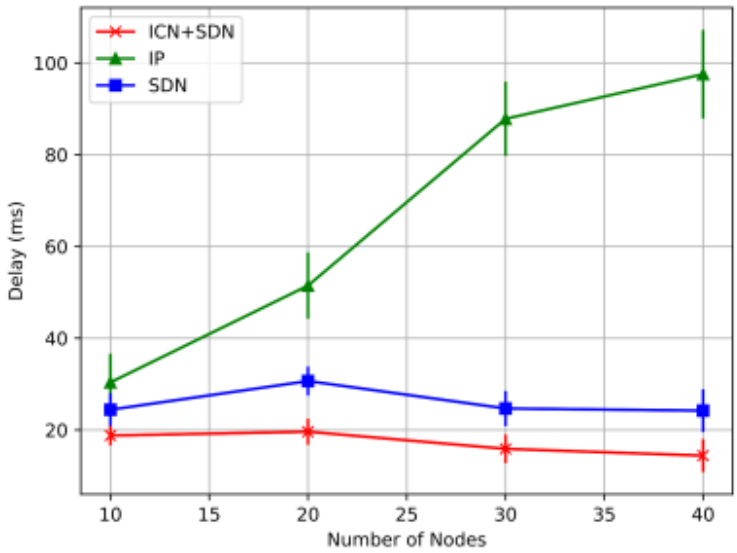
The results for the end-to-end delay metric are shown in Figure 33 and *Figure 34*.

Figure 33 shows the different performances due to the centralized controller and the caching mechanism. It can be observed that SDN-IP performs better than IP-only due to path optimization through the controller's global view of the network, and SDN-ICN performs significantly better than both approaches due to the caching mechanism, as expected.

However, the network performance may vary depending on the availability of the cache and the network load. When no cache is available, the efficiency of SDN-ICN is reduced

because in this case the switches only forward the packets without caching them. Also, if the cache is available but the packets are different, they are not cached. To verify the impact of cache availability on delay, experiments were conducted with the cache proportions listed in Table 7 (25%, 50%, 75%, and 100%) considering the 20-node topology.

Figure 33 - End-to-End Delay

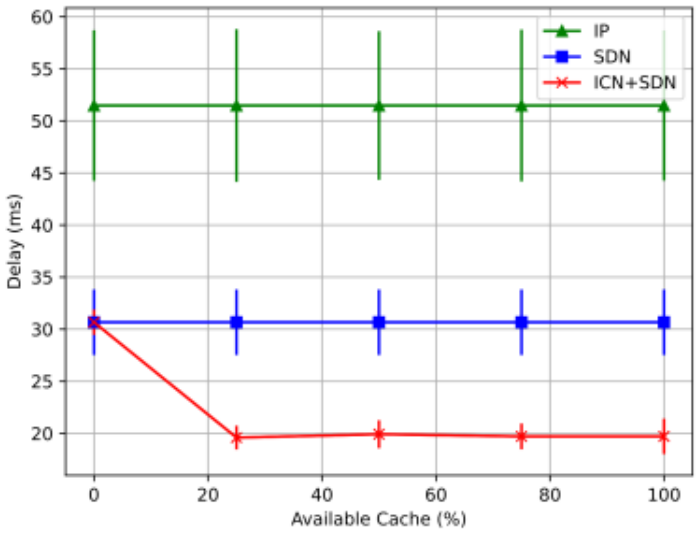


Source: (STOCCHERO *et al.*, 2023)

In *Figure 34*, the delay in the IP configuration is higher than the SDN-IP configuration due the data filtering techniques enabled by SDN. The SDN-ICN approach further improves this result because of the local cache. Both SDN-IP and SDN-ICN have the same results when there is no cache, but from 25% cache availability, the average RTT (delay) is the lowest for SDN-ICN because it avoids the frequent exchange of packets stored in intermediate nodes, which reduces the time to retrieve the data.

To verify the second metric (network load), three experiments were conducted. The first assessed the impact of the number of flows in the delay. Only one DATA type (500 KB) was used with 5 seconds period, using the 20 nodes topology. The results are presented in *Figure 35*. As the number of consumer rises, so does the number of flows. In the SDN-ICN configuration, as the number of flows increases, local cache gets packets stored closer to the consumer, reducing RTT. For a reduced number of flows, performance is similar to SDN-IP, since the cache does not present much improvement, but reduction in RTT is considerable for a higher number of flows. The IP-only configuration present higher but constant RTT for all flows, as expected.

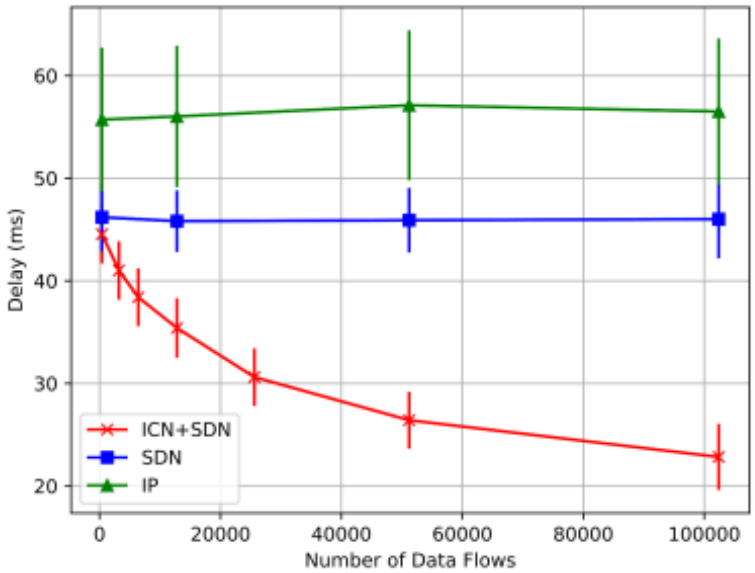
Figure 34 - Delay according to cache availability



Source: (STOCCHERO *et al.*, 2023)

The second experiment aimed to evaluate the overhead of INTEREST messages propagated through the network by the SDN-ICN configuration, which is an additional data volume as intermediate nodes propagate INTERESTS until the data is found. In this experiment, one consumer node originated INTERESTS, and the ICN amplification for an increasing number of nodes in the network is shown in *Figure 36*.

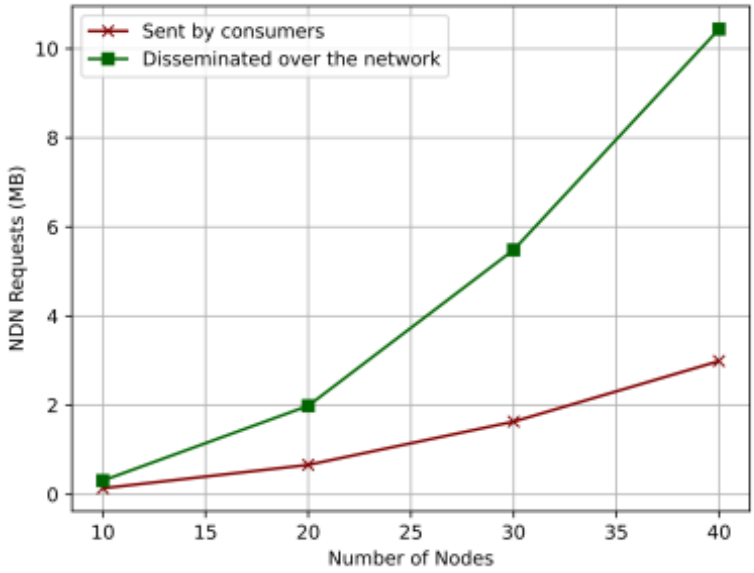
Figure 35 - Delay according to the network load (number of data flows)



Source: (STOCCHERO *et al.*, 2023)

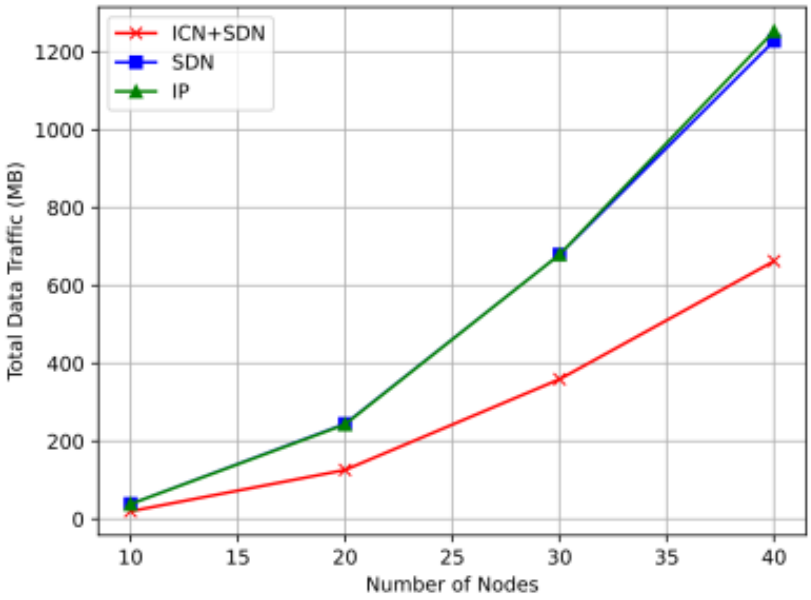
The third experiment evaluated the tradeoff between the amplification of INTEREST messages and the resulting reduction in the volume of DATA messages achieved by the SDN-ICN configuration. Using the same INTEREST messages generated in the second experiment, the resulting DATA messages are shown in *Figure 37*.

Figure 36 - Interest overhead



Source: (STOCCHERO *et al.*, 2023)

Figure 37 - Data Volume



Source: (STOCCHERO *et al.*, 2023)

Since INTEREST messages are much smaller (94 bytes on average) than DATA messages, the increase in traffic shown in the second experiment is greatly offset by the reduction in traffic volume from DATA because the cache reduces bandwidth usage and network congestion when the data is found near the consumer.

The second use case evaluated dynamic topology management. 70 independent simulation runs (number of runs sufficient to achieve 95% confidence interval for measured averages) were performed in a scenario with up to 20 randomly moving (5 to 10 m/s) nodes in the AOI (10 x 10 km2) and one controller node. Five nodes are data providers/consumers and an increasing number of intermediate nodes (3, 9, and 15) act only as relay nodes. The communication range between nodes is 1 km. DATA Messages are transmitted at intervals of 100 to 200 ms, CONTROL messages at intervals of 250 ms, and the controller reconstructs the topology at intervals of 500 ms.

The reliability of the network was measured in terms of packet loss and latency for messages CONTROL and DATA, as well as connectivity between data providers and consumers. The results are shown in *Table 8*. As expected, packet loss is significantly reduced as the number of relay nodes increases. This is due to the larger number of connections and the smaller required link length when more nodes are available as relays.

It can also be observed that the latency for message delivery decreases when the number of relay nodes increases, since they are closer together and there are more available nodes for shorter hops. The final metric is connectivity, which is measured as the ratio of the time it takes for data provider and consumer nodes to communicate with each other. It was observed that the network kept the data provider/consumer nodes connected for a longer period when more relay nodes were available.

Table 8 - Dynamic Topology Evaluation

Metric	**Number of Nodes**		
	8	**14**	**20**
CONTROL Packet Loss (%)	13.58 +/- 1.16	7.94 +/- 0.53	6.97 +/- 0.46
DATA Packet Loss (%)	41.4 +/- 3.33	28.65 +/- 2.72	25.19 +/- 1.92
CONTROL Latency (ms)	12.32 +/- 0.87	12.43 +/- 0.31	11.13 +/- 0.13
DATA Latency (ms)	17.56 +/- 0.34	12.85 +/- 0.26	11.28 +/- 0.09
Connectivity (%)	48.79 +/- 3.95	68.12 +/- 4.81	70.57 +/- 4.49

Source: (STOCCHERO *et al.*, 2023)

6.2.4 SDN Integration with ICN and DTN

The last proof of concept addressed the integration of SDN with ICN and DTN, as presented in Chapter 5. The results were published in (STOCCHERO *et al.*, 2022). An SDN orchestrator manages network services for the C2 application and enables concurrent IP, DTN, and ICN flows, each for a different application and QoS requirements. Security mechanisms were added to the architecture, traversing all layers of the architecture.

The system was verified through simulations based on two experiments that addressed information dissemination (NDN and IP) and resistance to distributed denial of service (DDoS) attacks, using a custom Mininet configuration that combines Mininet WiFi with MiniNDN modules. The default operating system for switches and APs was also replaced with Stratum.

The first experiment focused on the evaluation of IP × NDN. It represents the information exchange between an armored vehicle and a drone flying at the edge of the network communication range (SNR is weakest). When comparing the performance of the two paradigms, the C2 application in the vehicle makes a REQUEST (IP) or signals a INTEREST (NDN) for available data to the UAV. This scenario results in intermittent connections and fluctuates between frequent connections and disconnections. The environmental configuration consists of these two nodes (the drone and the vehicle) and an intermediate node (e.g., another vehicle or a soldier) with a stable connection (one-hop or multi-hop) to the ONOS installed in the vehicle and operating as a network AP to the drone. The connection provided by AP has a delay of 10ms and a data rate of 54Mb/s.

This simulation also evaluates the packet overhead caused by the continuously controlled ONOS traffic sent to the network and the overhead caused by the security mechanisms deployed with the SDN controller. During the experiment, there were 10 rounds of 100 transmissions each, for a total of 1000 transmissions for each of the two files responding to IP REQUEST and NDN INTEREST, with data payloads of different sizes: small (120.81 kB) and large (1113.53 kB). The overhead caused by the ONOS security mechanisms was measured by repeating the experiment in the IP paradigm, adding 20 calls to the REST API authentication mechanisms in each round.

The results are shown in *Table 9* and *Table 10*. NDN was more efficient than IP in all simulations in terms of packet losses. Although losses due to frequent disconnections are still observed, the volume of data delivered with NDN is much larger than with IP, showing the advantages of NDN as a data distribution mechanism. The advantage is greater for larger

payloads (a total of 648 MB of NDN versus 43.5 MB of IP), but the performance improvement is still significant for smaller payloads (a total of 86.2 MB of NDN versus 23.5 MB of IP).

At higher payloads, ONOS caused a slight increase in packet loss, slightly limiting NDNs ability to deliver data. ONOS also caused overhead for both IP and NDN protocols, due to the higher number of packets required to transmit the data (ONOS packets had lower payloads) and the security measures implemented. This overhead did not cause significant performance issues and is offset by the intangible benefits of ONOS, such as global view of the network, scalability, and real-time control of the data plane, which are important for the required network flexibility and security. Packet retransmissions occurred only when using IP, with and without ONOS, without significant performance degradation.

Table 9 - Performance and overhead: IP flows

		Without ONOS		With ONOS	
Total Payload Volume (MB)	Source Packets	Delivered Packets	Delivered Payload Volume (MB)	Delivered Packets	Delivered Payload Volume (MB)
120	793	563 +/- 28	23.5	595 +/- 24	23.5
1113	2540	1550 +/- 83	43.5	1584 +/- 69	43.5
Received Retransmissions in Number of Packets					
120		3		3	
1113		14		10	
Overhead in Number of Packets (Volume)					
120		235 +/- 58 (0.07MB)		214 (0.05MB)	
1113		252 +/- 57 (0.08MB)		214 (0.05MB)	

Source: (STOCCHERO *et al.*, 2022)

The second experiment simulated the use of Euclid (ILHA *et al.*, 2021) for the security dimension in a scenario representing a military network with two different configurations, one with and one without Euclid protection. A DDoS tool called Low Orbit Ion Cannon (LOIC) attacked both with a virtual machine as the attacker. Additional attacker nodes were stimulated by increasing the speed of the attacker's network interface card (NIC) from 2000 kbit/s (two attacker nodes) to 10000 kbit/s (ten attacker nodes), while the rest of the network remained at 1000 kbit/s.

The results are shown in *Figure 38* (without EUCLID protection) and *Figure 39* (with EUCLID). The network experiences significant degradation after the Low Orbit Ion Cannon

tool launches packet injection attacks. However, with the Euclid defense, there is about a 50% improvement in system responsiveness, reducing the average network delay from 0.65 to 0.35 seconds even for severe attacks (NIC speed = 10000 Kbps, injection of ~1721 attack packets/second of simultaneous messages in burst mode). This result is evidence that the C2 system can maintain quality of service for time-critical applications.

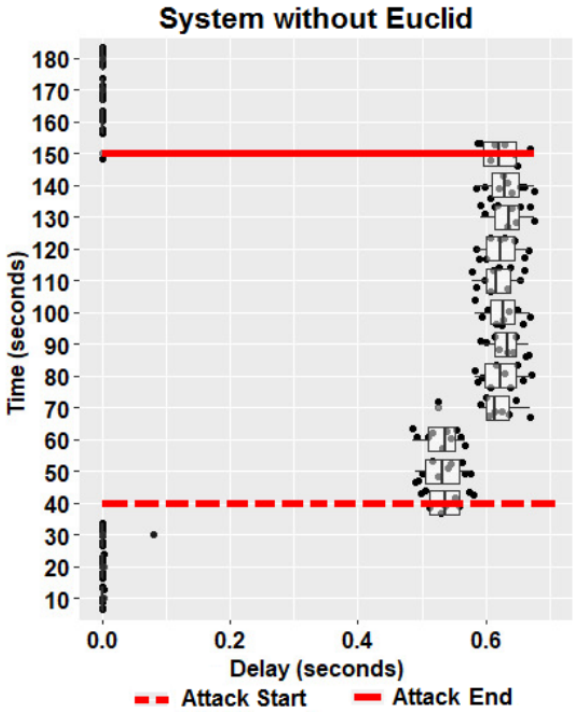
Table 10 - Performance and overhead: NDN Flows

		Without ONOS		With ONOS	
Total Payload Volume (MB)	Source Packets	Delivered Packets	Delivered Payload Volume (MB)	Delivered Packets	Delivered Payload Volume (MB)
120	143	108 +/- 5	86.2	116 +/- 33	86.2
1113	2605	1047 +/- 45	648	931 +/- 59	643.2
Received Retransmissions in Number of Packets					
120		3		3	
1113		14		10	
Overhead in Number of Packets (Volume)					
120		158 +/- 32 (0.03MB)		214 (0.05MB)	
1113		201 +/- 45 (0.04MB)		214 (0.05MB)	

Source: (STOCCHERO *et al.*, 2022)

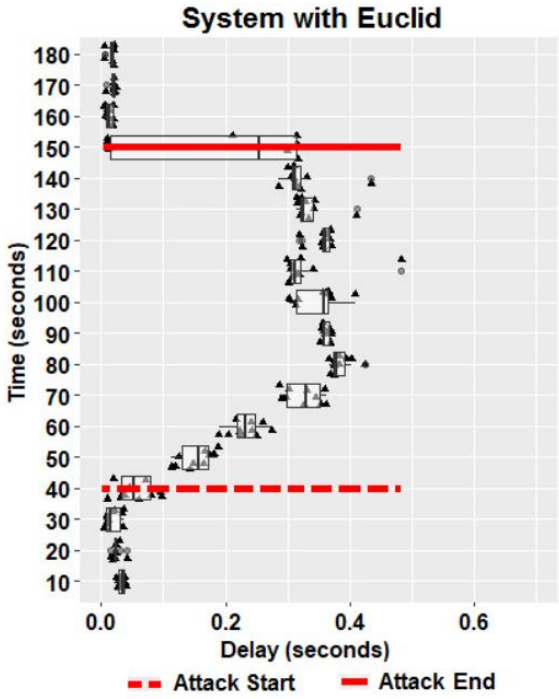
The computational overhead was checked by sampling the percentage processor utilization and the number of processes for four minutes in three configurations: 1) Baseline (without Euclid); 2) Secure (with Euclid); and 3) Secure under attack (with Euclid during a ten nodes attack). The results show that Configuration 2 increased the number of processes by 25 and the usage of CPU by 20.23% compared to Baseline. Configuration 3 increased the number of processes by 61 and the usage of CPU by 24.51% compared to the baseline.

Figure 38 - Security Evaluation without Euclid



Source: (STOCCHERO *et al.*, 2022)

Figure 39 - Security Evaluation with Euclid



Source: (STOCCHERO *et al.*, 2022)

7 COMPARISON WITH RELATED SOLUTIONS

The proposed architecture leverages several emerging network technologies. This complex combination may result in a less than ideal system that needs further development to be mature enough for deployment. This section discusses the implications for both operation and management of such an architecture and compares it to current research in the field.

A related work (NOBRE *et al.*, 2016) proposed an architecture that integrates SDN into Battlefield networks. In that work, the authors define an SDBN architecture that adapts SDN technology to BN and enables SDN-based applications, methods, and policies. Their approach is designed to support different types of wired and wireless networks in a military context by using a hierarchical distribution of controllers. Forwarding nodes are grouped around specific controllers, which in turn can be controlled by a higher-level controller. The authors evaluated their architecture using theoretical use cases without experimentation.

Another work (TARIQ; REHMAN; KIM, 2020) investigated the benefits of combining SDN controller, NDN, and IoT and their impact on overall network performance. The authors proposed an energy-efficient priority-based forwarding (EPF) strategy in SDN-enabled NDN-IoT to deal with the broadcast storms caused by the NDN' wireless channel. Similar to this work, the efficient flow management of the SDN controller was used to control the broadcast storm and forward the priority-based packets. The controller also uses Dijkstras algorithm to calculate the best possible and shortest path between the consumer and the producer.

In contrast to this work, they used an energy threshold mechanism to control energy consumption and improve overall energy efficiency. However, they did not consider node mobility in their work. They compared their system with the traditional flooding mechanism and Geographic Interest Forwarding (GIF) and obtained the best results in terms of total number of interests and retransmissions, end-to-end delay, total number of priority interests, energy consumption, and network lifetime.

Comparing their results with this work, there are similarities in terms of end-to-end delay reduction and number of retransmissions. However, the results of this work show slightly better performance when the number of nodes increases and a 20% reduction in end-to-end delay when the network size increases to 30 and 40 nodes. On the other hand, the delay in their simulations increased by 5% when the number of consumers/producers increased.

The combined use of ICN and SDN was also studied in (SIRACUSANO *et al.*, 2018). Here, the authors show that ICN can be supported by various SDN caching policies and present a platform for further experimentation and development. They also conclude that the adoption

of ICN in production networks is facilitated when implemented with SDN. The authors have tested and evaluated their proposal with Mininet and other emulators such as OFELIA and CONET.

Another work (GUO et al., 2021) addressed the broadcast storm in high-density networks (MANET) in a different way by improving the NDN routing protocol without using SDN. They implemented NOLSR (Novel Optimized Link State Routing Protocol), which is based on OLSR for NDN-MANET. Their protocol avoids the problem of unidirectional links that degrade network performance in a wireless environment by improving neighbor discovery.

The authors implemented NOLSR based on NFD, similar to the NLSR used in this work. They simulated different network sizes using a mini-NDN and found that the number of packets forwarded in the network and the average number of packets forwarded by each node were better compared to traditional flooding. Compared to this work, using the filtering capabilities of the SDN controller proposed here contributes to a lower number of packets forwarded in the network, especially when the number of nodes is higher. For example, in a network with 40 nodes, a 50% reduction in the total data volume was observed, while this reduction is not expected in their approach.

Another related solution (NAKAMURA, 2021) proposed a different mechanism for integrating ICN into DTN. The authors proposed an information-centric delay-tolerant network protocol (ICDTN). Instead of the host addresses used in conventional DTN, ICDTN resolves content requests and delivers content based on keywords, i.e., content descriptors, contained in the request message and the response message. An entity enters a content request with keywords into a network, and then the content request is passed through intermediate nodes in the network using store-carry-forward communication in DTN.

ICDTN provides pull-based communication, while DTN provides push-based communication. With ICDTN, an entity can retrieve available content at any point in time, which is beneficial in a disaster situation or battlefield network, for example. The authors conducted a preliminary evaluation of the protocol through simulations in simple scenarios. They focused on the similarity between a request message and a response message and measured the time required for a node to retrieve the response message whose similarity to the request message is greater than or equal to a certain constant value. They found that the average delay in retrieving a response message is significantly different depending on its similarity.

Although ICDTN seems to be a promising approach, the inclusion of ICN in conjunction with DTN causes delays in time-critical applications. The proposal to include ICN functions in

DTN might be a good idea to replace the traditional DTN approach used for the connections between UAVs and WSNs in the architecture proposed in this work.

These related works characterize current research efforts in this area and are summarized in *Table 11*, which shows some of the major contributions of this work compared to other works in the same research direction. One limitation of this work that is common to all research in this area concerns the integration of SDN with the physical (PHY) and medium-access control (MAC) layers, focusing more on network layer routing and traffic flow optimization.

Table 11 - Comparative Assessment

Work	Type	Paradigm	Scope	Main purpose	Validation	Target Network
(AMI*N et al.*, 2015)	Ad Hoc	DTN	Military	Mitigate periodic link outages on ship-to-shore networks	Simulation Satellite Tool Kit	High-end
(NOBRE *et al.*, 2016)	Infra-structure	SDN	Military	Increase flexibility and programmability in NCW	Theoretical Case Study	Hybrid
(DAPPER E SILVA *et al.*, 2019)	FANET	SDN	Military	Establish a topology management for Flying Ad Hoc Networks based on SDN	Simulation OMNet++	Low-end
(SIRACUSANO *et al.*, 2018)	Mesh	ICN, SDN	Civilian	Evaluate the performance of ICN over SDN	Emulation OFELIA Mininet CONET	High-end
(TARIQ; REHMAN; KIM, 2020)	IoT	SDN, ICN	Civilian	Establish an energy efficient protocol for IoT	Simulation ndnsim	Low-End
(GUO *et al.*, 2021)	MANET	ICN	Civilian	Improve NDN protocol	Emulation Mini-NDN	Low-End
(NAKAMURA, 2021)	MANET	ICN DTN	Civilian	Develop a protocol combining ICN and DTN	Simulation ICDTNSim	Low-End
This work						
	MANET	SDN DTN ICN	Military	Develop a network solution for BN	Emulation Mininet	Hybrid

Source: The author

All simulations used WiFi as the wireless medium, which is not common in military networks. Instead, tactical radios are typically used, with Software Defined Radio (SDR) being the most common technology. As a result, some parameters that affect key network performance metrics have not been studied in detail, such as transmission rate, transmit power, coding rate, channel (frequency/timeslot) allocation (for channelized systems), channel access (for random access), scheduling, and queue management.

Since this problem is common in SDN research, as seen in *Table 11*, (SUN et al., 2015) wrote a survey paper on cross-layer optimization that extends the scope of SDN to the flexible and fast adaptation of PHY and MAC layers offered by SDR. The authors aimed to combine the centralized network control inherent in SDN and the distributed nature of MANET with the additional features of SDR to optimize parameters and achieve global network objectives. The author's goal was to examine SDN and SDR as separate domains of wireless networks and present the lessons learned, with an eye toward the challenges associated with SDN-SDR interaction that would facilitate cross-layer optimization, i.e., the creation of fine-grained performance metric expressions for networks with dynamically adjusted link preferences.

The authors do not provide a solution, but rather suggest some possible research paths. SDR has access to various parameters of the radio environment, while SDN can use these results to set network policy. SDN switches could monitor the parameters of SDR channels. If necessary, the switch would send messages to the SDN controller. The controller, receiving messages from switches across the network, could change forwarding rules and orchestrate reconfiguration of SDR channels. These actions would optimize the network at both levels: the forwarding rules at the network layer, and the SDR parameters at the PHY and MAC layers.

Another limitation is the network emulator used in the proofs of concept. Although emulations and simulations are widely used in the research community to evaluate proposed architectures, and Mininet is widely used in the research community (and was selected for this reason), an improved testbed is needed for more robust experimentation in real operational scenarios. Unfortunately, such a testbed requires a large development effort, including interfaces with real military radios for more efficient coupling with lower layers, as mentioned earlier.

In this context, our research group at UFRGS has proposed to the Brazilian Army a research project in this area, including the development of mobility and propagation models suitable for creating realistic network scenarios. The Army is very interested in such collaboration, but has been unable to find funding. Conducting such a project would increase

the maturity of the proposed architecture in terms of scalability and feasibility in realistic military scenarios, but also bridge the gaps between SDN and SDR research.

On the other hand, this study is already part of a research project supported by the Brazilian Army (project S2C2, ref. 2904/20) to investigate a system of C2 systems. The contributions to S2C2 include the study of ad hoc network technologies to support distributed applications in command and control networks, including the specification of network requirements and the design to support multi-agents.

These limitations, combined with efforts to develop comprehensive research programs, highlight the complexity of designing and implementing C2 systems. (HOUGHTON, 2020) examined the nature of the challenges in managing, developing, and transforming C2 capabilities and sought to identify some of the reasons for the difficulties in transforming C2 practice relative to other areas of defense.

Because C2 is a socio-technological system, the authors identified social challenges (management of C2 capabilities, resistance to change, and organizational learning) and technological challenges (development and evaluation of C2 capabilities). The authors acknowledge that the challenge for C2 development is to build into our C2 organizations and systems what might be called "adaptability," which is precisely the goal of this research with the concept of implementing a technical solution for the NATO concept for C2 Agility.

The authors also acknowledge that C2 development requires a relatively high degree of match between human and technological components. In their view, C2 adaptability requires the implementation of solutions that are both stable and flexible, and that a new wave of adaptive and autonomous systems is yet to come. They also point to the need for greater alignment of organizational and technological development. C2 concepts and organizations must become key drivers of the latter.

The architecture proposed in this work is a possible solution to these challenges, as shown by the various published proofs of concept, although it is still at an early stage of maturity given its current limitations. As proposed by the authors, the development of the architecture derives from the C2 concepts and requirements (with a focus on the social aspects of a C2 system) outlined in Chapter 2.

8 CONCLUSION AND FUTURE WORK

The modern battlefield requires a significant change in the way command and control is conducted. Although advances in information and communications technology provide new mechanisms for designing military networks, current C2 systems are not yet capable of meeting the challenges of this new reality. In this context, the development of C2 systems requires a relatively high degree of alignment between human and technological components. The IoBT can be an answer to the agility requirements of C2 systems, but some challenges still need to be overcome. Considering the advances in ICN, DTN, and SDN paradigms, this work proposes a combination that leverages their best features to address these challenges.

This scheme enabled the use of differentiated services through an SDN-enabled C2 application. The project was designed for devices with low computational and energy resources, which is a characteristic of the IoBT. The simulations showed encouraging results, but future work should investigate other factors to improve the maturity level achieved in this work.

8.1 Contributions and conclusions

The proofs of concept presented in Section 6.2 represents relevant contributions.

In the work presented in 6.2.1, the main contributions are:

- Application of SDN in UAV-based military surveillance scenarios;
- The demonstration of the feasibility of an OpenFlow-compatible solution that enables the efficient use of commercial off-the-shelf equipment (COTS) in military networking applications; and
- An analysis based on quality of experience indicators and experimental results to support the proposed approach.

The results showed that SDN can be successfully applied to heterogeneous networks with opportunistic connections. In addition, they directed the research to investigate the use of DTN in conjunction with the SDN-based solution for handling disjoint partitions of the IoBT.

The work presented in 6.2.2 defined an architectural solution to support IoBT by combining SDN and DTN technologies. There were still some challenges to be solved, such as solutions to preserve the network topology in highly dynamic situations and mechanisms to migrate the SDN controller between nodes connected by wireless links with limited bandwidth. In addition, the integration of the proposed architecture with existing military network solutions needs to be further explored, especially with respect to network security.

The next proof of concept, presented in 6.2.3, had the following contributions:

- Definition of an architectural solution to support the IoBT by adapting and combining SDN and ICN;
- Modification of existing SDN and ICN protocols to support flexible C2 approaches; and
- Implementation and simulation in a tactical, realistic scenario for evaluation against IP-based and pure SDN implementations using metrics such as latency and network load.

This proof of concept has shown that ICN could be an interesting paradigm for data distribution in the IoBT, but open research questions remain, such as the study of cybersecurity mechanisms to cope with adversarial attacks and the development of C2 applications coupled to the network design to optimize the quality of service. Another important direction is to explore SDN orchestration capabilities to virtualize the underlying network protocols (ICN, DTN, and IP) for C2 applications.

Most of these issues were addressed in the work presented in 6.2.4, including an architectural solution for linking tactical C2 applications to network services protected by an efficient cybersecurity mechanism. The main contributions were:

- The development of network services that combine DTN, ICN, and SDN technologies to provide adequate quality of service, regulate traffic flow, and improve effective bandwidth;
- The definition of a multilayer cybersecurity mechanism for the solution.

8.2 Accomplishments and future work

Based on the results obtained in (STOCCHERO *et al.*, 2022) and (STOCCHERO *et al.*, 2023), this work contributed to answering the research questions raised and stimulated advances in the state of the art in C2 systems research, as follows:

- How to design a network architecture solution to handle the high-level variables of network-centric warfare (ADI, PI, and DI)?
    - The proofs of concept show that the designed architecture is capable of handling the high-level variables of NCW (ADI, PI and DI);
- How can network services for battlefield networks be orchestrated to establish and modify interactions between nodes and dynamically change the assignment of decision rights?

    - Experiments have also shown that the SDN organization proposed in this work can orchestrate network services for battlefield networks by setting and changing the parameters of node interactions;
    - In addition, this work has also identified the key vectors for cyberattacks on the IoBT and developed a layered security mechanism to mitigate their impact.
- Is it beneficial to use different network paradigms simultaneously to deal with the common problems of disruptions or interrupted communication channels and data dissemination in C2 networks?
    - The deployment examples and experiments have demonstrated the advantages of using DTN to maintain data flows where QoS is not mandatory and ICN for data dissemination;

Despite these achievements, and given the challenges in C2 research discussed in Chapter 6, future work should pursue the following lines of investigation:

- At the application dimension, mechanisms for mapping high-level user policies (node hierarchy, data priorities, etc.) into network intents that are later used by the SDN controller to orchestrate network services should be thoroughly investigated;
- At the application dimension, develop the data distribution component to create a complete set of interests and receive the appropriate data messages from the NDN switches;
- At the orchestration dimension, an efficient strategy must be developed to convert the intents received from the application into flow rules that are distributed by the SDN controller to switches throughout the network;
- At the security dimension, the use of blockchain as the basis for a decentralized security mechanism should be further explored, as should the trade-off between any planned security mechanisms and their cost to network metrics;
- Develop a realistic military testbed to replace the generic network emulator used in this thesis and conduct robust experiments to improve architecture development, including realistic propagation and mobility models for military scenarios; and
- Integration of SDR technologies into the architecture to replace the currently used WLAN standard and increase its technological readiness level.

9 APPENDIX A – PUBLISHED PAPERS

Table 12 lists all published papers related to this research. The most important ones, which form the core of the study, are the last two, for which I am the first author. The papers published up to 2018 served as the basis for starting this research. At that time, I was a special student at UFRGS and my role in the publications was to link network technologies and operational requirements for C2 systems development. In 2019, I officially started the PhD program, and the first paper with the basic concepts for this research was published in IEEE Comm Magazine. Although I am not the first author, my role in this paper was critical as I contributed to much of the research, with the exception of the simulations.

Table 12- List of Publications

Title	Publication	Qualis	Date
Combining Software-Defined and Delay-Tolerant Approaches in Last-Mile Tactical Edge Networking	IEEE Communications Magazine (ZACARIAS *et al.*, 2017a)	A1	2017
Enhancing Mobile Military Surveillance based on Video Streaming by Employing Software Defined Networks	Hindawi Wireless Communications and Mobile Computing (ZACARIAS *et al.*, 2018)	A2	2018
Employing SDN to control video streaming applications in military mobile networks	IEEE 16th International Symposium on Network Computing and Applications (ZACARIAS *et al.*, 2017b)	B1	2018
Empowering Command and Control through a Combination of Information Centric Networking and Software Defined Network	IEEE Communications Magazine (LEAL et al., 2019)	A1	2019
Secure Command and Control for Internet of Battle Things Using Novel Network Paradigms	IEEE Communications Magazine (STOCCHERO *et al.*, 2022)	A1	2022
Combining Information Centric and Software Defined Networking to Support Command and Control Agility in Military Mobile Networks	Peer To Peer Networks And Applications (STOCCHERO *et al.*, 2023)	A4	2023

10 APENDICE B – RESUMO EXPANDIDO EM PORTUGUÊS

O campo de batalha é um sistema complexo no qual diversas funções de combate buscam atingir os objetivos desejados. Nesse contexto, o Comando e Controle (C2) militar é a função de combate responsável por planejar, dirigir, coordenar e controlar operações militares e vincular outras funções de combate em vários níveis (BRASIL. EXÉRCITO BRASILEIRO, 2017). O conceito de Guerra Centrada em Redes (ALBERTS, 2011) é uma componente chave da transformação militar e depende de dispositivos computacionais e tecnologias de comunicação para criar uma consciência compartilhada, o que é crítico para futuras operações multidomínio (MDO) (FEICKERT, 2021), onde as atividades militares permeiam todos os domínios do campo de batalha: terra, mar, ar, espaço e cibernético.

O Conceito MDO requer tropas flexíveis e adaptáveis, que devem ser capazes de cooperar com agências civis e coalizões, tanto em terreno aberto como em cenários urbanos. Esses desafios demandam uma rede de combate que atenda simultaneamente os quartéis generais, ricos em recursos, e os elementos empregados na frente do campo de batalha, normalmente com severas restrições em capacidade computacional e comunicações.

Esses nós localizados na frente de combate incluem sensores sem fio, dispositivos vestíveis, veículos blindados de combate e pequenos Veículos Aéreos Não Tripulados (VANTs), entre outros equipamentos militares, formando uma Internet das Coisas de Batalha (IoBT), um tipo de rede descrita em (KOTT; ALBERTS, 2017), onde os humanos interagem com "Coisas" para alcançar objetivos militares.

A IoBT é uma rede de dispositivos físicos para coletar e/ou processar dados pela rede. As informações geradas permitem que o comandante aproveite ao máximo o conceito de MDO, encontrando e alternando rapidamente entre diferentes opções (em vários domínios) para enfrentar uma ameaça específico. Um cenário de exemplo é mostrado na Figura 1. Em tal cenário, forças dispersas na Área de Interesse (AoI) formam a IoBT, na qual um comandante local atribui missões e tarefas a seus recursos (pessoas ou coisas), de acordo com instruções de um quartel-general longe da AoI. A dinâmica desse ambiente e as ações do inimigo exigem decisões rápidas no terreno, pois não há tempo para acompanhar toda a cadeia de comando.

Para permitir este processo de tomada de decisão, o Sistema de C2 deve permitir um mecanismo eficiente de disseminação de informações, priorizando fluxos de dados com base nas circunstâncias e alterando dinamicamente as prioridades e a hierarquia dos nós da rede. A tomada de decisão rápida é um diferencial crítico no conceito MDO, especialmente quando forças hostis estão atacando um alvo específico, impulsionando o conceito de Agilidade de C2.

Um Grupo de Trabalho da Organização do Tratado do Atlântico Norte (OTAN) (T. G. SAS-085, 2014) definiu Agilidade de C2 como a capacidade de selecionar procedimentos e ajustar a abordagem de C2 durante operações militares. O relatório categorizou as estratégias de C2 em função de como os direitos de decisão são alocados, como os diferentes atores interagem e como a informação é disseminada. Assim, as abordagens C2 variam de hierarquias altamente centralizadas a redes fracamente acopladas.

Alcançar a Agilidade C2 no contexto da IoBT é um desafio, pois existem muitas deficiências tecnológicas nos atuais Sistemas. É necessário fazer alterações dinâmicas em três variáveis (Alocação de Direitos de Decisão, Padrões de Interação e Disseminação de Informação) em resposta às mudanças que ocorrem enquanto a operação é realizada. Tecnologias orientadas à semântica de dados, como redes centradas em informações (ICN), redes tolerantes a atrasos (DTN) e redes definidas por software (SDN) podem desempenhar um papel fundamental como base para tais implementações de sistemas.

O Paradigma SDN (POULARAKIS; IOSIFIDIS; TASSIULAS, 2018) orquestra serviços de rede, definindo dinamicamente hierarquia e permissões de nós e permitindo o processamento de dados nas proximidades da sua geração. Evita assim o roteamento desnecessário de dados e minimiza congestionamentos na rede. O SDN também facilita a reconfiguração de rotas para casar os requisitos da aplicação com as capacidades da rede.

O DTN lida com interrupções armazenando temporariamente dados em nós intermediários e encaminhando-os quando ocorrem conexões oportunistas (FALL; FARRELL, 2008). Esse recurso é extremamente útil para nós com alta mobilidade que podem se conectar e se desconectar de diferentes segmentos de rede. O paradigma ICN (ZHANG *et al.*, 2021), por sua vez, pode fornecer um mecanismo eficiente para a distribuição de dados, mudando a lógica de rede da abordagem tradicional baseada em endereço (centrada no nó de origem ou destino) para uma arquitetura de rede centrada em informações (centrada em dados).

O objetivo principal desta tese é propor uma solução de arquitetura de rede para obter Agilidade de C2 na IoBT, conectando a aplicação à rede através da orquestração de serviços de rede. A organização de alto nível da arquitetura pode ser visualizada na figura *Figure 10*.

O SDN orquestra fluxos de dados usando diferentes paradigmas:

- DTN é usado para suportar enlaces intermitentes, como a comunicação entre VANTs e redes de sensores sem fio (WSNs), que ocorre apenas quando um VANT está voando na faixa de comunicação do WSN, ou para conectar diferentes ilhas ICN;

- ICN é usado para distribuição de conteúdo, fornecendo informações táticas valiosas (sobre ameaças, por exemplo) para todos os nós autorizados, independentemente da fonte; e
- IP é usado sempre que a origem-destino é relevante, como a comunicação de voz entre soldados, bem como para suportar sistemas legados.

Algumas contribuições desta pesquisa são:

- Uma revisão das operações militares centradas em rede, mapeando possíveis soluções de rede para permitir a agilidade de C2;
- O estabelecimento de uma solução arquitetônica de rede para apoiar operações militares, acoplando aplicações táticas com serviços de rede protegidos por um mecanismo eficiente de segurança cibernética;
- O desenvolvimento de serviços de rede que combinem tecnologias DTN, ICN e SDN proporcionando uma qualidade de serviço adequada, regulando o fluxo de tráfego e melhorando a largura de banda efetiva; e
- O projeto de um aplicativo tático de C2 que aproveita esse serviços de rede.

A arquitetura proposta otimiza parâmetros de comunicação como latência, largura de banda do canal e mudanças de topologia, fornecendo um mecanismo de segurança para a solução. A ideia é orquestrar a rede tendo SDN como núcleo de forma a programar as seleções de protocolos de rede ideais para atender requisitos operacionais correntes. Ao mesmo tempo, a flexibilidade SDN permite para alterá-los, de acordo com a situação da operação e da rede. A arquitetura consiste de três dimensões: Orquestração de Rede, Aplicativo de C2 e Segurança.

O controlador SDN é o orquestrador, fornecendo escalabilidade e tolerância a falhas. Ele programa o comportamento da rede por meio de políticas para estabelecer a qualidade de serviço necessária, aplicando-as conforme o tipo de tráfego (fatias DTN, IP e ICN). O controlador SDN fornece serviços de rede para a dimensão de aplicativo de C2, que, por sua vez, fornece serviços de planejamento e monitoramento da operação para os usuários.

A dimensão segurança permeia as demais dimensões por meio do conceito de defesa em profundidade, o que evita que vulnerabilidades em uma dimensão afetem outra. Três camadas de segurança são usadas: hardware, software de rede e software de aplicativo. O acesso do usuário ao sistema é autenticado, enquanto a comunicação entre o controlador SDN e o aplicativo é criptografada, bem como o fluxo de dados nos switches (gerenciado pelos controladores) é controlado, evitando solicitações não autorizadas.

Várias provas de conceito foram realizadas para aumentar a maturidade da arquitetura e avaliar seus benefícios e potenciais inconvenientes. Esses resultados foram compilados em vários artigos publicados, listados na Tabela 12. Embora cada artigo publicado teve uma configuração específica, os experimentos geralmente usavam emuladores de rede para testar a funcionalidade desejada. Todas as interferências foram mitigadas pela escolha adequada dos cenários para que as simulações fossem executadas sem gargalos em termos de CPU e memória.

As provas de conceito mostram que a arquitetura projetada é capaz de lidar com as variáveis promotoras da Agilidade de C2 (Alocação de Direitos de Decisão, Padrões de Interação e Disseminação de Informação). Os experimentos também mostraram que a organização SDN proposta neste trabalho pode orquestrar serviços de rede para o campo de batalha, definindo e alterando os parâmetros de comunicação. Os exemplos e experimentos de implantação demonstraram as vantagens do uso de DTN para manter fluxos de dados onde QoS não é obrigatório e ICN para disseminação de dados.

Os experimentos realizados mostraram resultados animadores, mas trabalhos futuros devem investigar outros fatores para melhorar o nível de maturidade alcançado neste trabalho:

- Na dimensão da aplicação, investigar em detalhe mecanismos para mapear políticas de usuário de alto nível (hierarquia de nós, prioridades de dados, etc.) para o controlador SDN orquestrar serviços de rede;
- Na dimensão da aplicação, desenvolver o componente de distribuição de dados, definindo a configuração ideal dos switches ICN;
- Na dimensão de orquestração, uma estratégia eficiente deve ser desenvolvida para converter as intenções recebidas da aplicação em regras de fluxo a serem distribuídas pelo controlador SDN para switches em toda a rede;
- Na dimensão de segurança, o uso de blockchain como base para um mecanismo de segurança descentralizado deve ser mais explorado;
- Desenvolver um emulador militar realista para substituir o emulador de rede genérico utilizado neste trabalho, incluindo modelos realistas de propagação e mobilidade para cenários militares; e
- Integração de tecnologias RDS (Rádio Definido por Software) na arquitetura para substituir o padrão WLAN atualmente utilizado e aumentar seu nível de prontidão tecnológica.

11 REFERENCES

ADDAD, R. A. *et al.* Benchmarking the ONOS Intent interfaces to ease 5G service management. *In*: , 2018. **2018 IEEE Global Communications Conference (GLOBECOM)**. [*S. l.*]: IEEE, 2018. p. 1–6.

AFANASYEV, A. *et al.* NFD Developer ' s Guide. [*s. l.*], p. 1–56, 2015.

AGADAKOS, I. *et al.* Security for Resilient IoBT Systems : Emerging Research Directions. *In*: , 2019. **IEEE INFOCOM 2019 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)**. [*S. l.: s. n.*], 2019. p. 1–6.

ALALMAEI, S. *et al.* SDN Heading North : Towards a Declarative Intent-based Northbound Interface. *In*: , 2020. **16th International Conference on Network and Service Management (CNSM)**. [*S. l.: s. n.*], 2020. p. 1–5.

ALBERTS, D. S. Network Centric Warfare. **Networks**, [*s. l.*], p. 30, 2011. Disponível em: http://www.dodccrp.org/html4/research_ncw.html.

ALBERTS, D. S.; HAYES, R. E. **UNDERSTANDIGN COMMAND AND CONTROL**. [*S. l.: s. n.*], 2006.

AMIN, R. *et al.* Design Considerations in Applying Disruption Tolerant Networking to Tactical Edge Networks. [*s. l.*], v. 53, n. 10, p. 32–38, 2015.

ARAHUNASHI, A. K.; SCHOLAR, P. G. Performance Analysis of Various SDN Controllers In Mininet Emulator. *In*: , 2019. **4th International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT)**. [*S. l.*]: IEEE, 2019. p. 752–756.

ASIF, R.; GHANEM, K.; IRVINE, J. Proof-of-puf enabled blockchain: Concurrent data and device security for internet-of-energy. **Sensors (Switzerland)**, [*s. l.*], v. 21, n. 1, p. 1–32, 2021.

AZZOUG, Y.; BOUKRA, A. Bio - inspired VANET routing optimization : an overview. **Artificial Intelligence Review**, [*s. l.*], v. 54, n. 2, p. 1005–1062, 2021. Disponível em: https://doi.org/10.1007/s10462-020-09868-9.

BACCELLI, E. *et al.* Information Centric Networking in the {IoT}: Experiments with {NDN} in the Wild. *In*: , 2014. **Proc. 1st ACM Conf. on Information-Centric Networking**. [*S. l.*]: ACM, 2014. p. 77–86.

BERTAUX, L. *et al.* Software defined networking and virtualization for broadband satellite networks. [*s. l.*], v. 53, n. 3, p. 54–60, 2015.

BRASIL. EXÉRCITO BRASILEIRO. **Manual de Campanha Operações**. 5th. ed. [*S. l.*]: Comando de Operações Terrestres, 2017. *E-book*. Disponível em:

http://bdex.eb.mil.br/jspui/handle/1/848.

CAMPIONI, L. *et al.* Experimental Evaluation of Named Data Networking (NDN) in Tactical Environments. **Proceedings - IEEE Military Communications Conference MILCOM**, [*s. l.*], v. 2019-Novem, p. 1–8, 2019.

CHEN, K. *et al.* A scheme for improving the communications efficiency between the control plane and data plane of the SDN-Enabled airborne tactical network. **IEEE Access**, [*s. l.*], v. 6, p. 37286–37301, 2018.

COMMUNITY, R. S. F. **Ryu component-based software defined networking framework**. [*S. l.*], [*s. d.*]. Disponível em: https://ryu-sdn.org/. Acesso em: 11 set. 2022.

DAPPER E SILVA, T. *et al.* STFANET: SDN-Based Topology Management for Flying Ad Hoc Network. **IEEE Access**, [*s. l.*], v. 7, p. 173499–173514, 2019.

FALL, K.; FARRELL, S. DTN: an architectural retrospective. [*s. l.*], v. 26, n. 5, p. 828–836, 2008.

FAROOQ, M. J.; MEMBER, S.; ZHU, Q. On the Secure and Reconfigurable Multi-Layer Network Design for Critical Information Dissemination in the Internet of Battlefield Things (IoBT). **IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS**, [*s. l.*], v. 17, n. 4, p. 2618–2632, 2018.

FEICKERT, A. Defense Primer : Army Multi-Domain Operations (MDO). **In Focus, Congressional Research Service**, [*s. l.*], 2021.

FONTES, R. R. *et al.* Mininet-WiFi: Emulating software-defined wireless networks. *In*: , 2015. **2015 11th International Conference on Network and Service Management (CNSM)**. [*S. l.: s. n.*], 2015. p. 384–389.

GIBSON, C. *et al.* Opportunities and challenges for named data networking to increase the agility of military coalitions. **2017 IEEE SmartWorld Ubiquitous Intelligence and Computing, Advanced and Trusted Computed, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People and Smart City Innovation, SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI 2017 -** , [*s. l.*], p. 1–6, 2018.

GILLIAM, J. B.; VAN WIE, R. C. Interim Security Insights and Implications From the First Two Months of the Russia-Ukraine War. **Brookings Institution**, [*s. l.*], n. May, p. 1–22, 2022.

GKIOULOS, V.; GUNLEIFSEN, H. A Systematic Literature Review on Military Software Defined Networks. **Future Internet**, [*s. l.*], v. 10, n. 9, p. 88, 2018.

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.*]: University of Split, FESB, 2016. p. 1–5.

GUO, X. *et al.* A new solution based on optimal link-state routing for named data MANET. **China Communications**, [*s. l.*], v. 18, n. 4, p. 213–229, 2021.

HAN, Y. *et al.* An Intent-based Network Virtualization Platform for SDN. *In*: , 2016. **12th International Conference on Network and Service Management (CNSM)**. [*S. l.*]: IFIP, 2016. p. 353–358.

HOUGHTON, P. C2 Capability – Why do we fail to improve it ? Exploring Barriers to the Development of C2 Capability . *In*: , 2020. **25th ICCRTS**. [*S. l.: s. n.*], 2020.

HUKILL, J. *et al.* **Air Force Command and Control: The need for Increased AdaptabilityAir Force Command and Control**: The Need for Increased Adaptability. [*S. l.*]: Air University Press, 2012. Disponível em: http://www.jstor.org/stable/resrep13756.7. .

ILHA, S. *et al.* Euclid : A Fully In-Network , P4-Based Approach for Real-Time DDoS Attack Detection and Mitigation. **IEEE Transactions on Network and Service Management**, [*s. l.*], v. 18, n. 3, p. 3121–3139, 2021.

J. C. NIKOUE, S. B. and Y. M. Security Evaluation Methodology for Software Defined Network Solutions. *In*: , 2019. **International Conference on Platform Technology and Service (PlatCon)**. [*S. l.*]: IEEE, 2019. p. 1–6.

KHELIFI, H. *et al.* Named Data Networking in Vehicular Ad Hoc Networks: State-of-the-Art and Challenges. **IEEE Communications Surveys and Tutorials**, [*s. l.*], v. 22, n. 1, p. 320–351, 2020.

KISER, A. *et al.* The Combat Cloud: Multidomain Command and Control Across the Range of Military Operations. **Air Command and Staff College**, [*s. l.*], n. March, 2017.

KOTT, A.; ALBERTS, D. S. How Do You Command an Army of Intelligent Things?. [*s. l.*], v. 50, n. 12, p. 96–100, 2017.

LANTZ, BOB; HELLER, BRANDON; MCKEOWN, N. A network in a laptop: rapid prototyping for software-defined networks. *In*: , 2010. **Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks**. [*S. l.: s. n.*], 2010. p. 1–6.

LEAL, G. M. *et al.* Empowering Command and Control through a Combination of Information-Centric Networking and Software Defined Networking. **IEEE Communications Magazine**, [*s. l.*], v. 57, n. 8, p. 48–55, 2019.

LIOU, W.; LIN, T. T-Auth : A Novel Authentication Mechanism for the IoT Based on Smart Contracts and PUFs. *In*: , 2021. **IEEE International Conference on Communications Workshops (ICC Workshops**. [*S. l.*]: IEEE, 2021. p. 1–6.

LUO, T.; TAN, H. P.; QUEK, T. Q. S. Sensor OpenFlow: Enabling Software-Defined Wireless Sensor Networks. [*s. l.*], v. 16, n. 11, p. 1896–1899, 2012.

MAHMUD, R. *et al.* Software-Defined Multi-domain Tactical Networks: Foundations and Future Directions. **Mobile Edge Computing**, [*s. l.*], p. 183--227, 2021. Disponível em: http://arxiv.org/abs/2010.10509.

MCKEOWN, N. *et al.* OpenFlow : Enabling innovation in campus networks OpenFlow : Enabling Innovation in Campus Networks. **ACM SIGCOMM Computer Communication Review**, [*s. l.*], v. 38, n. April, 2008.

NAKAMURA, R. Proposal of Keyword-Based Information-Centric Delay-Tolerant Network. *In*: , 2021. **2021 IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR 2021)**. [*S. l.*: *s. n.*], 2021.

NATIONAL ACADEMIES OF SCIENCES, E. and M. Multi-Domain Command and Control: Proceedings of a Workshop in Brief (2018). *In*: , 2018. **multi-domain command and control (MDC2)1 in the U.S. Air Force (USAF).** [*S. l.*]: The National Academies Press, 2018. Disponível em: https://doi.org/10.17226/25316.

NOBRE, J. *et al.* Toward software-defined battlefield networking. [*s. l.*], v. 54, n. 10, p. 152–157, 2016.

NOUR, B. *et al.* A survey of Internet of Things communication using ICN: A use case perspective. **Computer Communications**, [*s. l.*], v. 142–143, p. 95–123, 2019. Disponível em: https://www.sciencedirect.com/science/article/pii/S0140366418309228.

NUNES, B. A. A. *et al.* A survey of software-defined networking: Past, present, and future of programmable networks. [*s. l.*], v. 16, n. 3, p. 1617–1634, 2014.

OUBBATI, O. S. Routing in Flying Ad Hoc Networks : Survey , Constraints , and Future Challenge Perspectives. **IEEE Access**, [*s. l.*], v. 7, p. 81057–81105, 2019.

P4.ORG. **P4Runtime**. [*S. l.*], [*s. d.*]. Disponível em: https://p4.org/p4-spec/p4runtime/main/P4Runtime-Spec.html. Acesso em: 9 jul. 2022.

PENG, P.; LU, R.; WU, L. A Review of The Development of Distributed Task Planning in Command and Control Domain. *In*: , 2021. **IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)**. [*S. l.*: *s. n.*], 2021. p. 659–663.

PETIT, B. S. Finding Order in Chaos : Conceptualizing Resistance Command and Control Approaches. **Journal on Baltic Security**, [*s. l.*], v. 8, n. 1, p. 131–149, 2022.

PHEMIUS, K. *et al.* Bringing SDN to the edge of tactical networks. *In*: , 2016. **MILCOM 2016 - 2016 IEEE Military Communications Conference**. [*S. l.*]: IEEE, 2016. p. 1047–1052. Disponível em: http://ieeexplore.ieee.org/document/7795468/. Acesso em: 7 mar.

2019.

POULARAKIS, K.; IOSIFIDIS, G.; TASSIULAS, L. SDN-enabled Tactical Ad Hoc Networks: Extending Programmable Control to the Edge. **IEEE Communications Magazine**, [*s. l.*], v. 56, n. 7, p. 132–138, 2018.

RAVURI, H. K. *et al.* A Scalable Hierarchically Distributed Architecture for Next-Generation Applications. **Journal of Network and Systems Management**, [*s. l.*], v. 30, n. 1, 2022.

RUSSELL, S.; ABDELZAHER, T. The Internet of Battlefield Things : The Next Generation of Command , Control , Communications and Intelligence (C3I) Decision-Making. *In*: , 2018. **MILCOM 2018 - 2018 IEEE Military Communications Conference (MILCOM)**. [*S. l.*]: IEEE, 2018. p. 1–6.

RUSSELL, S.; ABDELZAHER, T.; SURI, N. Multi-Domain Effects and the Internet of Battlefield Things. *In*: , 2019. **IEEE Military Communications Conference (MILCOM)**. [*S. l.*]: IEEE, 2019. p. 724–730.

SANVITO, D. *et al.* ONOS Intent Monitor and Reroute service : enabling plug & play routing logic. *In*: , 2018. **4th IEEE Conference on Network Softwarization and Workshops (NetSoft), 2018**. [*S. l.*]: IEEE, 2018. p. 272–276.

SCOTT, L. *et al.* Exploring dependencies of networks of multi-genre network experiments. *In*: , 2016. **2016 IEEE Military Communications Conf.** [*S. l.: s. n.*], 2016. p. 576–581.

SIRACUSANO, G. *et al.* A framework for experimenting ICN over SDN solutions using physical and virtual testbeds. **Computer Networks**, [*s. l.*], v. 134, p. 245–259, 2018. Disponível em: https://doi.org/10.1016/j.comnet.2018.01.026.

STEPHEN J. TOWNSEND; ARMY, U. Accelerating Multi-Domain Operations - Evolution of an Idea. **Military Review Special Edition**, [*s. l.*], v. September-, n. AUGUST, p. 1–3, 2018. Disponível em: https://www.armyupress.army.mil/Journals/Military-Review/English-Edition-Archives/September-October-2018/Townsend-Multi-Domain-Operations/.

STOCCHERO, J. M. *et al.* Combining information centric and software defined networking to support command and control agility in military mobile networks. **Peer-to-Peer Networking and Applications**, [*s. l.*], n. 0123456789, 2023. Disponível em: https://doi.org/10.1007/s12083-022-01443-z.

STOCCHERO, J. M. *et al.* Secure Command and Control for Internet of Battle Things Using Novel Network Paradigms. **IEEE Communications Magazine**, [*s. l.*], p. 1–7, 2022.

SUN, S. *et al.* Integrating network function virtualization with SDR and SDN for 4G/5G networks. **IEEE Network**, [*s. l.*], v. 29, n. 3, p. 54–59, 2015. Disponível em: http://ieeexplore.ieee.org/document/7113226/. Acesso em: 7 mar. 2019.

T. G. SAS-085. **Command and Control (C2) AgilityNATO Science and Technology Organisation**. [*S. l.*: *s. n.*], 2014.

TARIQ, A.; REHMAN, R. A.; KIM, B. S. Epf—an efficient forwarding mechanism in sdn controller enabled named data iots. **Applied Sciences (Switzerland)**, [*s. l.*], v. 10, n. 21, p. 1–22, 2020.

TORTONESI, M. *et al.* Enabling the Deployment of COTS Applications in Tactical Edge Networks. [*s. l.*], v. 51, n. 10, p. 66–73, 2013.

TOSH, D. K. *et al.* Blockchain-Empowered Secure Internet-of-Battlefield Things (IoBT) Architecture. *In*: , 2018. **MILCOM 2018 - 2018 IEEE Military Communications Conference (MILCOM)**. [*S. l.*]: IEEE, 2018. p. 593–598.

VACHUSKA, T. **Open Network Operating System**. [*S. l.*], [*s. d.*]. Disponível em: https://opennetworking.org/onos/. .

VEDHAPRIYAVADHANA, R.; FRANCY IRUDAYA RANI, E.; THEEPA, M. Simulation and performance analysis of Security issue Using Floodlight controller in Software Defined Network. **2018 International Conference on Emerging Trends and Innovations In Engineering And Technological Research, ICETIETR 2018**, [*s. l.*], p. 1–6, 2018.

WICKBOLDT, J. *et al.* Software-Defined Networking: Management Requirements and Challenges. [*s. l.*], 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**, [*s. l.*], v. 55, n. 10, p. 1–8, 2017.

ZACARIAS, I. *et al.* Employing SDN to control video streaming applications in military mobile networks. *In*: , 2017. **2017 IEEE 16th International Symposium on Network Computing and Applications (NCA)**. [*S. l.*]: IEEE, 2017. p. 1–4. Disponível em: http://ieeexplore.ieee.org/document/8171390/. Acesso em: 3 out. 2018.

ZACARIAS, I. *et al.* Enhancing Mobile Military Surveillance Based on Video Streaming by Employing Software Defined Networks. **Wireless Communications and Mobile Computing**, [*s. l.*], v. 2018, p. 1–12, 2018.

ZENG, D. *et al.* Convergence of Edge Computing and Next Generation Networking. **Peer-to-Peer Networking and Applications**, [*s. l.*], v. 14, p. 3891–3894, 2021.

ZHANG, M. *et al.* Comparative analysis of probabilistic forwarding strategies in ICN

for edge computing. **Peer-to-Peer Networking and Applications**, [*s. l.*], v. 14, n. 6, p. 4014–4030, 2021.

ZHANG, L. *et al.* Named data networking. **ACM SIGCOMM Computer Communication Review**, [*s. l.*], v. 44, n. 3, p. 66–73, 2014. Disponível em: http://dl.acm.org/citation.cfm?doid=2656877.2656887.

ZHAO, Q. *et al.* An Integrated Software-Defined Battlefield Network Testbed for Tactical Scenario Emulation. *In*: , 2019. **MILCOM 2019 - 2019 IEEE Military Communications Conference (MILCOM)**. [*S. l.*]: IEEE, 2019. p. 373–378.

ZHOU, W. *et al.* A reconciliation model of agile C2 organization based on converged networks. *In*: , 2020. **Proceedings - 2020 6th International Conference on Big Data and Information Analytics, BigDIA 2020**. [*S. l.: s. n.*], 2020. p. 58–65.

ZURANIEWSKI, P. *et al.* Facilitating {ICN} Deployment with an Extended Openflow Protocol. *In*: , 2017. **Proc. 4th ACM Conf. on Information-Centric Networking**. [*S. l.*]: ACM, 2017. p. 123–133.

ZURANIEWSKI, P. *et al.* Facilitating ICN deployment with an extended openflow protocol. **Proceedings of the 4th ACM Conference on Information-Centric Networking - ICN '17**, [*s. l.*], p. 123–133, 2017. Disponível em: http://dl.acm.org/citation.cfm?doid=3125719.3125729.