An Architecture for Software Defined Drone Networks

Mohannad Alharthi
School of Computing
Queen's University
Kingston, Ontario
harthi@cs.queensu.ca

Abd-Elhamid M. Taha
Electrical Engineering Department
Alfaisal University
Riyadh, Saudi Arabia
ataha@alfaisal.edu

Hossam S. Hassanein
School of Computing
Queen's University
Kingston, Ontario
hossam@cs.queensu.ca

***Abstract*—Drones or Unmanned Aerial Vehicles (UAVs) are utilized in a wide range of applications, as they are considered flexible and cost-effective. Novel applications have been recently explored, such as providing communications and Internet coverage where ground infrastructure is lacking or in temporary situations. In this paper, we propose a drone-based network architecture enabled by Software Defined Networking (SDN) to provide dynamic and flexible networking capabilities, suitable for different types of drone applications and deployments, while we discuss associated challenges related to SDN in done networks.**

I. Introduction

Unmanned Aerial Vehicles (UAVs), aka drones, are considered key instruments in emergency situations such as search and rescue, surveillance, and various scientific and civilian applications. The goal of using drones is reducing the cost of missions and eliminating associated risks of sending human personnel to conduct risky or costly tasks, especially in the case of natural disasters and tasks in difficult to reach areas. As drones became more cost-effective and capable of carrying communication technologies, they became an attractive solution to deploy as a fleet of cooperating drones to cover larger geographical areas and relay information to remote targets. This also enabled more advanced applications as currently explored in the literature, e.g., as flying base stations (BSs) in 5G networks, and as flying networking and computing infrastructure to support drone missions and communication networks. Drone-based networks can offer great flexibility to satisfy dynamic and unpredictable service demands, while reducing the cost as an alternative to deploying fixed ground infrastructure.

The use of multiple drones instead of a single one can be motivated by cost reduction and service enhancements [1]. Generally, using multiple small connected drones can cost less than using a single large drone [2], in addition to extending the coverage area and possibly complete tasks faster. While equipped with communication capabilities, a key advantage of such systems is that drone missions can still operate even if some drones fail. Benefits also include the ability to scale out the network by adding more drones as needed. Additionally, connectivity between drones enables creating a network in the sky that relay information over a large area, while drones' computing resources can provide some network functions and application-specific processing, so the network can be creatively applied in a wide range of scenarios.

Our vision for the drone network is that it can be reusable for a variety of applications. Thus, it is practical to develop an architecture that is flexible and cost-effective for doing so. Software Defined Networking (SDN) enables fast adoption of new innovations in networking technologies by moving the behavior of networking devices to a logically centralized controller. The decoupling of the control and the data planes allows for central network management that is unified and flexible regardless of the underlying hardware. As well, it enables implementing adaptive network solutions on the go for the network according to current state and environment. Additionally, SDN can virtualize the network for multiple tenants simultaneously, and isolate unrelated application traffic.

In this paper, we propose an architecture for an SDN-based drone network that we believe is suitable for implementing a wide range of drone deployment scenarios. It is useful for use-cases such as network coverage, and sensing and scanning, and is able to operate where no access to networking infrastructure is available. It can be utilized by law enforcement agencies, mobile network providers, and scientific researchers. We also discuss challenges in designing SDN-based drone networks.

In the following section, we offer some background material. In Section 3, we discuss some related work, and in Section 4 we present the challenges and requirements of our architecture and its design. Finally, in Section 5, we discuss use-cases of our architecture.

II. Background

A. Software Defined Networking (SDN)

In SDN [3], networking devices (specifically switching devices) are turned into simple but flexible devices that expose their functionality through a programming interface. Instead of operating independently with hardcoded functionality, devices are controlled by logically centralized controllers that implement the control logic and dynamically direct how packet flows are routed via well-defined API. Internal flow tables in switches are manipulated by instructions received from the network controller. The API let the controller insert forwarding rules based on packet headers and associated forwarding and processing actions. The controller maintains a global view

978-1-5386-8088-9/19/$31.00 ©2019 IEEE

of the topology of the network, and exposes an API for higher level applications to implement specific aspects of the network, while the controller implements their logic in underlying switches. Applications can implement network protocols, some middle-box functions, and network management tasks. The API exposed by network devices or switches are known as the *South-bound Interface* (SBI), for which OpenFlow is the de facto standard. The API exposed by the SDN controller to high-level applications is called the *North-bound Interface* (NBI), and it is not standardized, as many SDN controller platforms exist.

SDN has been applied in different real-world applications such as in [4]. While a large portion of these efforts focused on wired networks and data centers, the application of SDNs is a popular trend in mobile and wireless networks research, more recently in the core network of LTE and 5G architectures [5] [6]. The use of SDN in drone networks enables utilizing those advantages, contributing to the cost reduction and flexibility gained by using drones.

B. Drones

Drones are manufactured in various sizes and capabilities. They are generally classified based on their flying capabilities, altitude, range and payload size [2]. Drones with rotary wings, such as quad-copters, are small in size, have less endurance, and fly at lower altitudes than other high altitude and high endurance types. The payload can be customized to carry specialized equipment such as cameras, sensors, and communication hardware. Such drones are also known as Low-Altitude Platform (LAP), which is the type of drones we consider suitable for this work, due to flexibility in cost and deployment.

III. Related Work

A few works aimed at utilizing SDN and related Network Function Virtualization (NFV) in drone-based networks. In [7], authors provide a software defined UAV-based network architecture for robust end-to-end connectivity. In this architecture, UAVs are equipped with multiple network interfaces such as Wi-Fi and LTE, and act as SDN switches. The SDN controller implements a multi-path disjoint routing protocol utilizing the multiple interfaces in each UAV. The routing component of the controller actively computes multiple disjoint paths between UAVs while considering the quality of links between drones with to provide alternative paths in case of link failures.

In [8], authors introduce an SDN/NFV drone-based Flying Ad-hoc Network (FANET) for rural zone monitoring. It is realized as a Video Monitoring as a Service platform that utilizes cameras on the ground and on drones. Drones serve as a backbone network for the platform, where end users (consumers) monitor a certain rural area by viewing broadcasted video streams. The architecture is comprised of backbone drone nodes that form a mesh network, in which drones are equipped with NFV to provide virtual network functions (VNFs) that implement video transcoding and streaming. The network is managed by a master orchestrator drone that controls all virtualization aspects. The platform specifies the VNF chains that realize video recording, storage, and streaming.

Both of aforementioned works presented an SDN, or NFV-enabled drone network applications. However, practical challenges of enabling SDN in drone networks are not addressed. We discuss them in the next section.

IV. Design of Software Defined Drone Network Architecture

In this section, we outline our proposed design of a software defined drone network. First, we discuss the challenges encountered in SDN-enabled drone networks. These challenges influence the design and requirements of the architecture.

A. Design Requirements and Challenges

While drones combined with SDN gives us a lot of flexibility in terms of the mobility, programmability and reconfiguration, the design of the architecture faces some challenges. The common challenges in drone networks are the possible dynamic change of nodes locations, drone and link failures, and capacity constraints (computing, power, etc.), so the design of a drone network must address these challenges. The key design requirements are resilience, reconfigurability, reusability, and energy awareness. The central control and management of SDN can help in solving these challenges through programmability. We discuss these requirements below.

- Resilience: The network should adapt in the face of network and drone failures. This is possible because of the relative ease of deploying a new drone, or reconfiguring an already deployed one compared to a ground base station. If a certain link fail, traffic can be rerouted through alternative paths.
- Reconfigurability: Drones will need to adjust the network topology as well as node configuration as network hosts to meet service requirement or adapt to drones entering or leaving the network. For instance, when the network needs to extend coverage, some node might need to be reconfigured to act as a relay node to stay reachable to an end user.
- Reusability: The network should be flexible to implement various deployment applications. For example, a drone deployment might implement a search and rescue mission, as well as providing Internet connection to end users on the ground and have the drone network as a backhaul using the same physical drones.
- Energy-awareness: The network should be designed to be energy aware because power sources of drones can be limited. Using SDN, traffic flows can be dynamically balanced to conserve energy.

SDN introduces additional challenges when applied to drone networks. Those challenges exist in wired networks, but they can be more challenging given the wireless and mobile nature of the network. Network devices require reliable control plane links in order to communicate with the SDN controller. So in addition to providing reliable data plane links, reliable

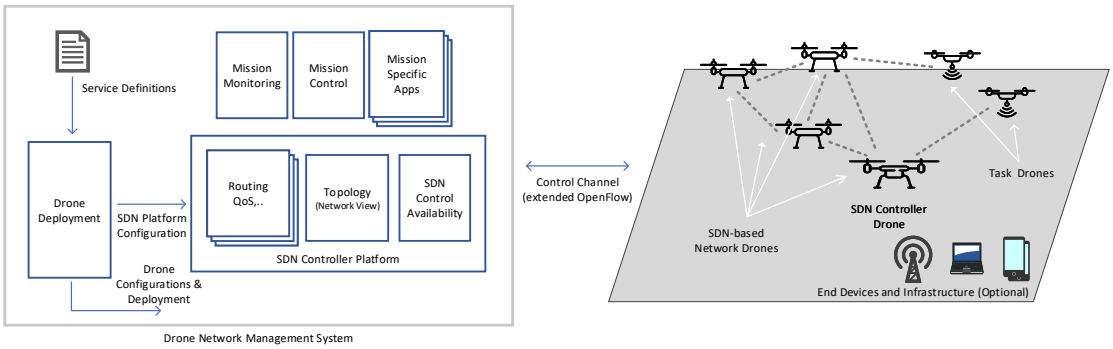

Fig. 1. Software Defined Drone Network Architecture

wireless control plane links need to be maintained. As network control decisions are centralized, the controller availability to nodes becomes critical. The topology formation for drones can consider its proximity to the controller, but that can limit the range of where drones can be deployed. In such cases, additional controller instances can be deployed as drones. Optimal placement can be applied to drone-based networks, however, that can add to the cost of deployment, so special considerations need to be recognized, such as the limited communications and computing capacity in drones, as well as mobile nature of drones.

Moreover, the SDN control plane introduces computing and communication overhead, a result of control messages exchanged between nodes and the SDN controller. The SDN controller requires computing resources to maintain a current global view of the network and dynamically make data plane decisions. This may not be a challenge if the controller mainly resides in a ground station or in the cloud. However, we consider applications where the network may need to operate independently from a ground infrastructure, where the SDN controller is a drone. So considering the imposed overhead by using SDN is especially important where such resources are expensive such in the case of drone networks. As for communication overhead, the controller monitors network nodes continuously to maintain its view, however the frequency of such updates need to be minimized to conserve communication resources.

B. Design Overview

The architecture provides a reconfigurable and reusable network for different services or applications. Drones represent the nodes of the SDN-based network, performing networking and mission-related functions. The network is managed by a ground management system that carries out the planning and deployment of drones, and the monitoring, management and control of the network. Operators of the drone network specify their service requirements by supplying mission location, network policy descriptions and application implementations needed for the drone deployment to carry out its mission. The management system deploys the software components on drones, and the specified policies and routing of network traffic are enforced by the SDN controller located within the management system. The management system also monitors drones resources and capitalizes on its global network view to dynamically reconfigure drones and network topology as needed.

C. Architecture Components

The network architecture components, shown in Figure 1, consist of a Drone Network Management System (DNMS), located on the ground, and drones with heterogeneous communication, computing and storage capacities. These components are described below. Drones represent the nodes of the network, and they form a wireless multi-hop mesh network utilizing multiple network technologies.

1) Drones: Drones form the nodes of the multi-hop network. Each drone can be different in terms of its capabilities. A typical drone would be equipped with:

- One or more wireless interfaces with OpenFlow support.
- Generic computing and storage elements.
- A GPS unit, and optionally a set of sensors if needed.
- An operating system with container-based virtualization to host task applications.

Drones can have varying roles depending on their capabilities, their geographic location and service requirements. The two main roles are network drones and task-specific drones. The task of network drones is the provision of SDN-based wireless connectivity, managed by the OpenFlow protocol. Task-specific drones perform the actual tasks related by the mission, such as sensing, monitoring, or providing wireless access to ground end users. Network drones can be separate from task drones, and optionally a capable drone can carry out both roles provided the needed components and resources. Network drones also have the following additional configurable roles in the network:

Relay Drones: Some drones can act as relay drones to extend the coverage of the network. For example, some drones can reach a remote area inaccessible by other nodes. A relay node can position itself between two task sites or end-points such as an end device or the management system.

Controller Drones: Controller drones are flying SDN controllers that are deployed with the drone network to optimize and distribute the logically centralized SDN control functions. This is also beneficial when the network needs to be deployed to a remote location away from the management system,

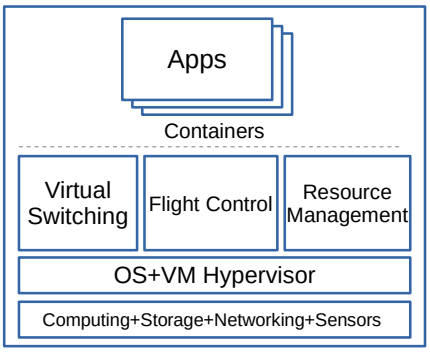

Fig. 2. Drone Architecture

and where communication infrastructure is inaccessible to the mission.

The internal components of drones are described below. The components, shown in Figure 2, are based on the hardware foundation of the drone, including its computing resources, sensors and specialized equipment, which are managed by an operating system with a hypervisor for virtualizing the underlying hardware, to enable easier deployment of applications into drones. The rest of the components are:

- *Resource Management*: The resource management component performs monitoring of local computing, storage and power resources. It keeps track of resources utilized by applications, and how much resources are available, so it can be acquired by control components.
- *Flight Control*: This component is responsible for flying the drone. This includes local flight control to follow a preconfigured path or to dynamically avoid obstacles during flight. It also configured instructions from the DNMS to adjust its path or change location. The DNMS can gather information from drones or external sources about some areas in a demand of the particular service that is been offered by the drone network. The new paths or location targets are forwarded the drones to adjust their movement accordingly.
- *Virtual Switching*: Virtual switching is required to direct traffic through the multiple network interfaces and configure and manage the data plane behavior as instructed by the SDN controller. It provides the programmability needed for the network, as it is the programming interface for network nodes.
- *Applications*: Task-specific applications implement the functionalities and services intended for the deployed drone network. These run on virtualized drone computing resources. Such applications include any specialized programs that manages task specific equipment such as sensors and cameras. Applications run on virtualized containers, and have virtualized access to the drone hardware, such as cameras and sensors, in order to capture, process and collect sensor data. They act as a driver for the mission by initiating tasks and service provision.

2) Drone Network Management System: The Drone Network Management System (DNMS) is a ground network control system that communicates with drones and exchange control and state information during the operation of the network over a control channel. It consists of several components that are collectively responsible for the monitoring of drones' computing and network resources, and the deployment of applications and network configurations on drones. The DNMS is based on an SDN controller platform, with associated SDN applications that controls the data plane of the network. On top of it, sit applications that implement mission monitoring and control. The DNMS, depicted in Figure 1, consists of the components described below.

Drone Network Deployment Component (DNDC): The DNDC is responsible for configuring the DNMS and network nodes for the mission after receiving deployment requirements as *service definitions* from the drone network operators. The definitions include information needed to implement the functionality of the drone network, and they include:

- the geographical locations of deployment,
- node topology,
- data plane policies,
- the drones used, their roles and specifications,
- drone application implementations,
- and mapping of suitable drone and application pairings.

The service definitions specify where the drone network should be deployed geographically along with the required network topology. The DNDC configures the initial flight path and formation of drones. The flight path and formation are assumed to be initially determined based on prior knowledge of the coverage area known to the mission operators. Geographic locations can also be updated during the mission. Moreover, service definitions contain a listing of all drones and their capacities. The listing also specifies drones with special equipment needed for special roles, such as camera drones and drone access points. The data plane policies are essentially high-level SDN applications that define data plane policy, such as quality of service (QoS) and forwarding paths. These high-level applications are translated to data plane configurations or forwarding rules. More complex SDN applications can be specified, such as applications that implement adaptive routing protocols.

The DNDC is also assumed to have a physical location where drones have their before-flight payload installed and set up, as well as performing drone maintenance and charging. Drones also return to this point in order to be recharged or replaced by other drones.

SDN Control Component (SDNC): This component consists of the SDN controller to control the data plane of the drone network. The SDNC is configured by the DNDC with the configuration and SDN modules required for the network as defined in the service definition. The SDNC can be realized using extensible off-the-shelf SDN controller platform such as OpenDayLight or ONOS. The SBI (OpenFlow) is extended to facilitate control and monitoring related messaging performed by the different SDN applications. Characteristically, the SDNC keeps track of the network topology and maintains a global view of the network and an information base. It computes the routing and forwarding of traffic flows, and

continuously installs them on the network drones. The SDNC may integrate additional modules that perform other functions, such as QoS, security, and virtualization. The SDN Control Availability module ensures reliable control-plane links between network drones and the SDNC. It monitors and deploys additional drone controllers to ensures their availability. This is especially key when the network is deployed in a remote area not in the range of the DNMS.

Mission Monitoring and Control: The MMC components are responsible for the high-level monitoring of the overall mission, related to drones' locations, power and networking resources and flight path and topology formation. It works on top of the SDNC, utilizing its NBI to access and program drones. While the MMC keeps track of these resources, it informs the SDNC so that its data plane control can reconfigure the network according to available resources by, for instance, reducing the load on a certain drone or by finding alternative routes for traffic through less utilized drones. The formation of drones is also adjusted by the MMC. The mission may require its own additional components. For instance, a certain mission may require custom adaptive routing protocol or implement certain control mechanisms related to the mission.

3) End Devices and Ground Infrastructure: End devices are optional components of the network. They represent equipment used by end users to access the network to collect data gathered by drones or to communicate with other devices through the drone network, depending on the deployment application. Examples include ground users with mobile devices equipped with a compatible network technology to connect to the network. The functionality of end devices is governed by the type of the network and the applications deployed on the drones. As well, ground infrastructure can be optionally utilized by incorporating drones that can connect to the LTE or 5G if required to connect to external networks and the Internet.

V. Use Cases

A network system based on this architecture is suitable for operators that require limited deployments or others that require larger scale and multi-purpose deployments. Below we describe example use cases of our architecture.

A. Small-Scale Scenarios

This type of deployments suits small and recurrent single applications. A matching example is monitoring and scanning of natural disaster areas, such as hurricanes and earthquakes. A drone network system based on our architecture can be utilized by agencies that handle such situations. The agency maintains a fleet of drones with SDN-enabled components and management system, as well task-specific equipment such as thermographic and/or regular cameras. Service definitions are formulated to achieve mission objectives. Definitions include the drone descriptions, the network policies, and for each mission, the geographical location of drones and the target area. Applications are also provided for capturing and encoding photos and video from the drone camera, and forwarding them to end devices for monitoring purposes. Applications specific for monitoring along with the SDN platform and associated SDN applications can be updated regularly to improve the operation of the mission, the networking and management without requiring too frequent updates to physical drone components.

B. Large-Scale Scenarios

This type of deployments is ideal for a large service provider that deploys a flying networking infrastructure that can be utilized for several purposes by multiple tenants. An already flying network covers an area that is not properly covered by ground infrastructure. A telecom provider might be one tenant by supplying their own service definitions and task-specific drones that connect to the drone network and provide services to ground users. Local law enforcement can be another tenant that monitor a certain area for security purposes. The SDN-based network can provide a communication backbone for those tenants and thanks to SDN, provide traffic isolation. The whole network can be managed and monitored efficiently, and network policies can be dynamically adjusted accordingly.

Ideally, given a stable infrastructure of drones with powerful computing and networking capabilities, the compute and networking infrastructure can be virtualized using a cloud computing and NFV platform similar to [8]. The SDN-enabled network enables the virtualization of the underlying network, while the compute resources can provide virtualized network functions that can be deployed and instantiated as needed. However, that will introduce additional challenges that require more efficient orchestration of resources on the mobile and wireless infrastructure.

VI. Conclusion

In this paper, we presented an architecture for a software defined drone network that utilizes SDN to implement different drone mission scenarios. We also described use cases that demonstrate the reconfigurability and reusability benefits of our architecture.

References

[1] İ. Bekmezci, O. K. Sahingoz, and Ş. Temel, "Flying ad-hoc networks (fanets): A survey," *Ad Hoc Networks*, vol. 11, no. 3, pp. 1254–1270, 2013.

[2] M. Mozaffari *et al.*, "A Tutorial on UAVs for Wireless Networks: Applications, Challenges, and Open Problems," *ArXiv e-prints*, Mar. 2018.

[3] D. Kreutz *et al.*, "Software-defined networking: A comprehensive survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, Jan 2015.

[4] S. Jain *et al.*, "B4: Experience with a globally-deployed software defined wan," in *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM*, ser. SIGCOMM '13. New York, NY, USA: ACM, 2013, pp. 3–14.

[5] A. Basta *et al.*, "A virtual sdn-enabled lte epc architecture: A case study for s-/p-gateways functions," in *2013 IEEE SDN for Future Networks and Services (SDN4FNS)*, Nov 2013, pp. 1–7.

[6] R. Trivisonno *et al.*, "Sdn-based 5g mobile networks: architecture, functions, procedures and backward compatibility," *Transactions on Emerging Telecommunications Technologies*, vol. 26, no. 1, pp. 82–92.

[7] G. Secinti *et al.*, "Sdns in the sky: Robust end-to-end connectivity for aerial vehicular networks," *IEEE Communications Magazine*, vol. 56, no. 1, pp. 16–21, Jan 2018.

[8] C. Rametta and G. Schembra, "Designing a softwarized network deployed on a fleet of drones for rural zone monitoring," *Future Internet*, vol. 9, no. 1, 2017.