



FFI-RAPPORT

19/00808

Autonomous Sensors and Communications Infrastructure

Erlend Larsen
Kim Mathiassen
Lars Landmark
Martin Vonheim Larsen
Øivind Kure

Autonomous Sensors and Communications Infrastructure

Erlend Larsen
Kim Mathiassen
Lars Landmark
Martin Vonheim Larsen
Øivind Kure

Keywords

Bildesensorer
Kommunikasjonsnettverk
Maskinl ring
SDN (Software Defined Networking)
Tr dl s kommunikasjon

FFI-rapport

19/00808

Project number

1372, 1371, 1400, 1367

ISBN

E: 978-82-464-3207-6

Approvers

Andr  Pettersen, *Director of Research*
Lorns Harald Bakstad, *Research Manager*

The document is electronically approved and therefore has no handwritten signature.

Copyright

  Norwegian Defence Research Establishment (FFI). The publication may be freely cited where the source is acknowledged.

(U) Summary

The future digital battlefield will employ a vast number of sensors, both manned and unmanned. Identifying solutions for efficient interconnection between decision-makers, sensors, and effectors is very important to enable joint operations. Allowing the users to enforce more control over the behavior of the communications infrastructure is part of these solutions. So is also taking control over the network topology, especially exploiting the potential of unmanned platforms to suit an external communications requirement.

The report aims to shed light on the challenges and possibilities for autonomous observation systems in tactical communications networks through four research questions:

1. What specific requirements do autonomous observation systems put on communications?
2. How can autonomous observation systems contribute to improve the performance of the communications infrastructure?
3. How can the communications infrastructure meet these requirements?
4. What is the gap between the existing communications infrastructure and these requirements?

To investigate these research questions, a joint experiment campaign was initiated, where several research activities at the Norwegian Defence Research Establishment (FFI) were focused at working together and enabling better efficiency of new technology.

This report describes multiple experiments performed on a scenario consisting of a sensor, two unmanned relays and an information cell. The experiment was a collaborative effort between several research programs at FFI and Kongsberg Defence Systems as an industry partner. The effort included experiments from an application perspective and experiments from a communications perspective.

The results from the experiment campaign show that there is a need for more interaction and understanding between the users and the communications infrastructure, especially in a demanding tactical mobile environment, where the difference between the volume of data that a sensor can produce, and the small volume that a modern tactical mobile radio system network can move, is extremely large.

(U) Sammendrag

Fremtidens digitale kamparena vil bruke et enormt antall sensorer, både bemannede og ubemannede. Det er svært viktig å identifisere løsninger for effektiv sammenkobling av beslutningstakere, sensorer og effektorer for å muliggjøre fellesoperasjoner. En del av disse løsningene er å gi brukere muligheten til å ta mer kontroll over oppførselen til kommunikasjonsinfrastrukturen. Det samme er også muligheten til å ta kontroll over nettverkstopologien, spesielt det å utnytte ubemannede plattformer for å tilfredsstille eksterne kommunikasjonskrav.

Rapporten tar mål av seg til å belyse utfordringer og muligheter for autonome observasjonssystemer i taktiske kommunikasjonsnettverk gjennom fire forskningsspørsmål:

1. Hvilke spesifikke krav til kommunikasjon stiller autonome observasjonssystemer?
2. Hvordan kan autonome observasjonssystemer bidra til å forbedre ytelsen til kommunikasjonsinfrastrukturen?
3. Hvordan kan kommunikasjonsinfrastrukturen møte disse kravene?
4. Hva er gapet mellom den eksisterende kommunikasjonsinfrastrukturen og disse kravene?

For å undersøke disse forskningsspørsmålene, ble det initiert en felles eksperimentkampanje hvor flere forskningsaktiviteter ved Forsvarets forskningsinstitutt (FFI) ble fokusert inn mot å arbeide sammen og effektivisere bruken av ny teknologi.

Denne rapporten beskriver en rekke eksperimenter gjennomført på et scenario som bestod av en sensor, to ubemannede reléer og en informasjonscelle. Eksperimentrekken var et samarbeid mellom flere forskningsprogrammer ved FFI og Kongsberg Defence and Aerospace som industrisamarbeidspartner. Arbeidet inkluderte både eksperimenter fra et applikasjonsperspektiv og fra et kommunikasjonsperspektiv.

Resultatene fra eksperimentene viser at det er behov for mer samhandling og forståelse mellom brukerne og kommunikasjonsinfrastrukturen, spesielt i et krevende taktisk mobilt miljø, hvor det er ekstremt store forskjeller mellom det store volumet av data som en sensor kan produsere og det lille volumet et moderne taktisk mobilt radiosystem kan overføre.

Contents

(U) Summary	3
(U) Sammendrag	4
Preface	6
1 Introduction	7
1.1 Background	7
1.2 Research questions	9
1.3 Methods	9
1.4 Report structure	9
2 System description	10
2.1 Application	11
2.1.1 Sensor platform configuration	11
2.1.2 Sensor data processing	11
2.1.3 Operator's interface	14
2.2 Application and network communication	14
2.3 The communications infrastructure	20
2.3.1 Kongsberg TacLAN radios	20
2.3.2 Kongsberg Multi-role radios	22
2.3.3 SDN and OpenFlow	22
3 Experiments	24
3.1 Application	25
3.2 Communication	25
4 Results	28
4.1 Application	29
4.2 Communication	30
5 Discussion	34
5.1 Application performance	34
5.1.1 Practical issues affecting the performance	34
5.2 Applications use of the communications infrastructure - ROS bridge	35
5.3 Serving the application from a communications infrastructure perspective	36
5.4 How to contribute to improving the performance of the communications infrastructure	37
6 Conclusion and future work	38
References	42

Preface

At FFI, the activities on sensors, unmanned ground vehicles, autonomy for unmanned systems and communications infrastructure are organized into separate research programs. However, several of the programs employ the same individual researchers, thus making collaboration and interaction easier.

The experiment campaign described in this report was a collaborative effort between the FFI-projects 1372 "Autonomi for ubemannede systemer", 1371 "Ubemannede kjøretøy for Forsvaret", 1400 "Situasjonsforståelse og aktiv beskyttelse for stridskjøretøy", and 1367 "Taktisk mobil kommunikasjon for Forsvaret" through its activity in Coalition Network for Secure Information Sharing (CoNSIS) II.

The FFI-project 1372 "Autonomi for ubemannede systemer" investigates different aspects of autonomy for unmanned systems, and is tasked with providing output to the whole range of unmanned platform projects at FFI. The interdisciplinary collaboration needed for the experiment described in this report have been easier to implement because of the contacts generated through this project.

The FFI-project 1371 "Ubemannede kjøretøy for Forsvaret" investigates the potential for employing Unmanned Ground Vehicles (UGVs) in the Norwegian Armed Forces (NAF). This project addresses the use of UGV and cooperation with intelligent sensors. In the experiment, the UGV project used the communication infrastructure to transfer information from an intelligent camera sensor, via relays including a UGV, to an information cell. The UGV project also collaborates with Kongsberg Defence Systems (KDS) wherein KDS supports the UGV project with radios and implementation of software to bridge information exchange between the sensor and the information cell over the radio communication infrastructure.

The FFI-project 1400 "Situasjonsforståelse og aktiv beskyttelse for stridskjøretøy" has an activity exploring different aspects of future situational awareness systems for armored vehicles. This involves multi modal sensor systems as well as autonomous sensor processing. The Argus sensor demonstrator platform later described in this report was developed as part of this project.

The CoNSIS is a multi-national research project, combining the efforts of the industry and research facilities. It is based on a Memorandum of Understanding (MoU) between the collaborating nations, funding respective national defense industry Research & Development (R&D). Phase I of CoNSIS started in 2008 and ran through 2012, while phase II started in 2015 and ran for two years through 2016. FFI-project 1367 "Taktisk mobil kommunikasjon for Forsvaret" have participated in CoNSIS phase II Task 1 "Communication Infrastructure", where there has been a collaboration with KDS on developing several new functions in their TacLAN broadband radios.

This report will be useful for sensor application developers and communication infrastructure developers, both in the procurement agency, the Norwegian Defence and at FFI.

The authors would like to thank KDS for providing equipment and participation through CoNSIS II, in addition to programming for the Robotic Operating System (ROS) bridge.

Kjeller, 11 June 2019

Erlend Larsen, Kim Mathiassen, Lars Landmark, Martin Vonheim Larsen, Øivind Kure

1 Introduction

The high rate of development with regards to sensor technology should be exploited by the Norwegian Armed Forces (NAF). However, as new sensor technology enables the generation of vast volumes of data, transferring these between actors in the defence structure, between sensors at the edge and the decision makers connected to a wired high bandwidth infrastructure separated by large distances, becomes a defining challenge in order to exploit the sensors. Several new techniques must be used, in order to enable the effect of powerful sensors at the edge, for instance machine learning. Another aspect is autonomous platforms that can be controlled by the network to support needed network resources through physic repositioning and deployment. This report presents an experiment campaign performed at Rygge in March 2017. The campaign was a practical approach to researching the integration of new technology in a potential future tactical communications infrastructure for the NAF.

1.1 Background

The future digital battlefield will employ a vast number of unmanned intelligent sensors, to reduce the need for humans in jobs that are dirty, dull and/or dangerous. These sensors will increase the situational awareness in the battlefield, enabling moving inside the enemy's Observe-Orient-Decide-Act (OODA)-loop. Such sensors may be basic digital sensors continuously producing a set rate of data, or be equipped with intelligence to limit the need for data capacity, transmitting only when specific events occur. A camera, for instance, can either produce a continuous video stream or generate video or still pictures on-demand, upon detecting a target of interest. There is already a great gap between the amount of data that a single sensor can produce and the capacity of tactical mobile radio systems available for procurement today, and this gap will only increase, due to the development of improved sensor technology. A 4 megapixel camera having 6 frames per second will produce 576 Mbps, and for change detection purposes these data needs to be uncompressed before they are processed. In our experiment we have five such cameras and they could potentially generate up to 2.5 Gbps of data, while the maximum link rate is 2.5 Mbps, creating a virtual gulf between the sensor and the decision maker in terms of data transfer. However, the data rate can be significantly reduced by processing the sensor data locally before sending aggregated data back to the operator.

Along with the fast developing intelligent sensors, unmanned platforms are soon to be introduced in the NAF, in all domains (airborne, ground-based and sea-based). These sensors and platforms will increase the demand for tactical communication resources.

To achieve efficient use of the sensors and the unmanned vehicles, they are expected to incorporate a large degree of autonomy. Autonomy in this respect may be both for positioning the unit and for intelligent use of the communication resources.

Delivering the sensor information to decision makers will be essential to take advantage of the sensors. In a tactical environment, the communications infrastructure is challenged with a number of problems, ranging from capacity to stability. The need for transfer capacity has historically been low in tactical mobile networks. For the most part, the NAF tactical mobile network infrastructure consists of low capacity narrowband systems. These radio systems offer long range communication (>30 km). They are mainly used for voice communication and Battle Management System (BMS) information. The future capacity need will presumably increase. Thus, systems with more capacity must supplement these narrowband systems. Wideband radio systems can provide this capacity

increase, at the expense of range. Higher radio frequencies curve less around obstacles in the terrain. They are also more dependent on Line-of-Sight (LoS), due to propagation limitations. Across the ground, the range is thus reduced with increasing radio frequencies. This makes it difficult to completely replace narrowband communications with wideband systems. More likely, the two types of radio networks will be used in combination, where wideband communications supplement the current narrowband communications, and support the sensor traffic with high capacity requirements. However, the support will be varying, depending on the configuration of the participating nodes and the communications infrastructure. In such a case, both the adaptiveness of the applications and the means to configure the network depending on the environment will impact the end result, i.e. the ability to exchange and utilize information. Understanding and mitigating the communications infrastructure limitations will allow for efficient communications for the future sensor-saturated battlefield.

For Unmanned Ground Vehicles (UGVs) there is a need to transmit and receive different types of information. This will depend on the type of mission it is executing and at what stage in the mission the UGV is in. Before the mission it will need to receive the mission plan from the user. While executing the mission the communication needs are largely defined by the mission. In most situations the UGV will transmit its position back to the operator. In an Intelligence, Surveillance and Reconnaissance (ISR) type mission it is likely that the UGV, or an Unattended Ground Sensor (UGS) that is deployed by the UGV, will send images or video back to the operator. This will give other bandwidth requirements to the communication system.

Configuring the network optimally, with regards to traffic flow paths and the allocation of resources, is difficult in an operation that can last for a long time. Sensors will provide information with differing importance and amount of information. The topology will change and adversaries may attempt to disrupt communications. Taking advantage of information which is not directly available to the network protocol when engineering the use of the network may enable a more efficient network use. An example is the deployment of a Unmanned Aerial Vehicle (UAV) as a communication relay. A network protocol would not look at this elevated node any differently than a ground node. However, the links between the ground nodes and the UAV will have other properties than between ground nodes. The UAV will potentially provide more stable links but will interfere with more ground links. I.e., the UAV can be used as a relay if the traffic requires stable transfer, at the cost of less capacity available for the rest of the network.

This report is based on a set of experiments addressing a scenario consisting of a sensor, two unmanned relays and an information cell. The effort included both experiments from an application perspective and from a communications perspective. Through the experiments we explored the duality between the efficient use of communications infrastructure for application purposes and vice versa. The applications need to operate while understanding the limitations of the communications infrastructure, and the communications infrastructure needs to treat application traffic according to the needs of the application. Through the collaboration between projects on communication infrastructure, UGV, and situational awareness, a combination of experiments have been carried out as a large system test:

- The traffic flows of different parts of a sensor application were steered over multiple paths across a small tactical network.
- Several application-focused experiments were performed.

1.2 Research questions

The report aims to shed light on the challenges and possibilities for autonomous observation systems in tactical communications networks. We formulate the following research questions to help address these challenges:

1. What specific requirements do autonomous observation systems put on communications?
2. How can autonomous observation systems contribute to improve the performance of the communications infrastructure?
3. How can the communications infrastructure meet these requirements?
4. What is the gap between the existing communications infrastructure and these requirements?

We do not aim to answer these questions exhaustively and finally in this report, but the work should be considered a contribution to these stated research questions.

1.3 Methods

We have sought to find answers to the questions above through planning and executing an experiment campaign using a setup of autonomous observation systems and an interconnecting communications infrastructure in the form of a heterogeneous tactical radio network. The communications infrastructure was used to transport information from a sensor, through relay nodes with varying characteristics, to a Ground Control Station (GCS). The experiment campaign was considered as a potential catalyst for generating answers to the presented research questions.

This report is limited in that it only addresses the combination of one sensor type and one specific communications infrastructure, investigating one particular method to implement greater flexibility into the communications infrastructure. Nevertheless, we consider the results to be valid for more generic solutions than this specific experiment.

1.4 Report structure

The structure for the rest of the report is as follows: First, we give a description of the experiments' building blocks in Chapter 2. Second, the different experiments are described in Chapter 3. Third, the results of the experiments are presented in Chapter 4. A discussion of the results is given in Chapter 5, and finally the report is concluded in Chapter 6 with a conclusion and potential future work. The chapters 2 through 4 are organized into sections based on the focus areas, while the discussion and conclusion chapters address the research questions from a joint perspective.

2 System description

To seek answers to the questions stated in the previous chapter, we designed a systems test consisting of several experiments where the main purpose of the system was to retrieve information from a remotely placed unmanned sensor, the Unattended Ground Sensor (UGS). The sensor information system utilized a communications infrastructure to interconnect the sensor and the user. The user was situated at another location, the Ground Control Station (GCS). Further, as part of the communications infrastructure, the system included two nodes used primarily for relaying the sensor information and sensor control communication between the sensor and the user. One relay was an Unmanned Ground Vehicle (UGV), both capable of deploying the UGS, and moving throughout the operation area. The other relay was an Unmanned Aerial Vehicle (UAV), more mobile and potentially a more stable relay while on station, but weight limited and with low endurance. Fig. 2.1 shows the four system nodes in one potential radio connectivity situation. The main components were the four system nodes:

UGS Unattended Ground Sensor – the camera sensor with five cameras and advanced local image processing.

GCS Ground Control Station – the information cell where information from the sensor terminated.

UGV Unmanned Ground Vehicle – the mobile ground relay.

UAV Unmanned Aerial Vehicle – the mobile aerial relay.

The UGS consisted of five cameras giving an overview of the monitored scene. In addition there was a Pan-Tilt-Zoom (PTZ) camera that could take close-up pictures of objects of interest and send them to the operator. The UGV was FFI's autonomous unmanned vehicle Olav, a Polaris All Terrain Vehicle (ATV) [1].

The platforms were connected using two different radio systems, one narrowband, utilizing the Very High Frequency (30-300 MHz) (VHF) frequency band and one wideband, utilizing the

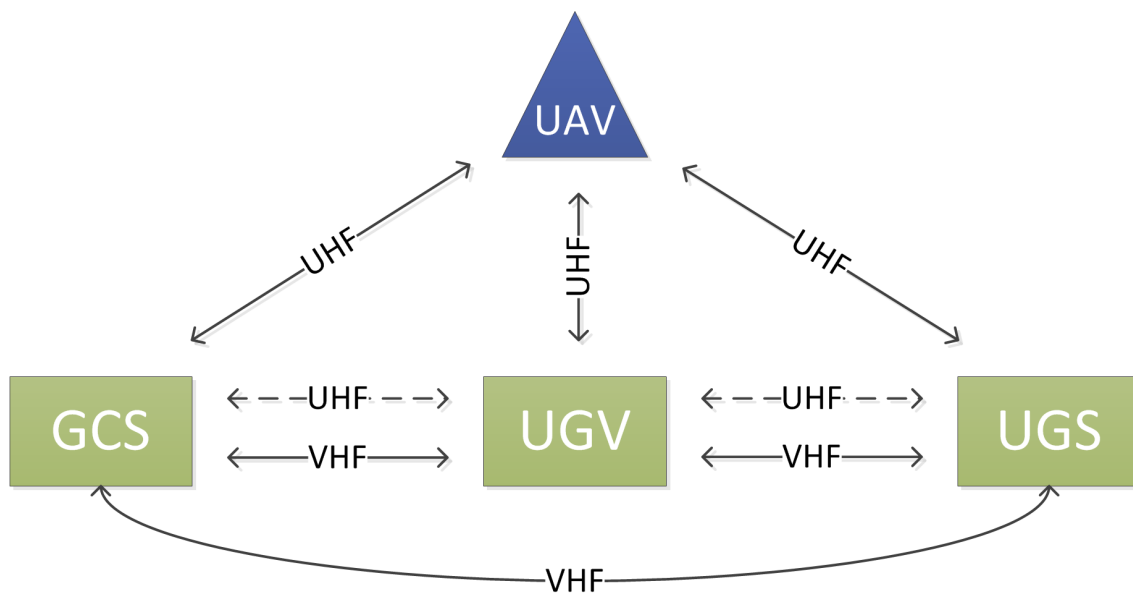


Figure 2.1 The system components in one potential radio connectivity situation. Dotted lines indicate unstable links.

Ultra High Frequency (300-3000 MHz) (UHF) frequency band. These systems had different characteristics that could be exploited through traffic engineering, in particular the difference in range and capacity. Even though the wideband radio system supported much larger transfer rates than the narrowband system (2.5 Mbps vs 19.2 kbps), both radio systems were limited with regards to capacity, compared to the amount of information the UGS was capable of producing. Due to the limited capacity of the communications infrastructure, the transmission of information to the operator had to be limited. This was done through advanced local image processing at the UGS, and through a Robotic Operating System (ROS)-bridge, developed to handle the application's communication between the UGS and the GCS.

The next sections describe the application, the application and network communication, and the components of the communications infrastructure.

2.1 Application

The application used in the experiment is based off of the Argus sensor platform developed by FFI project 1400 *Situational awareness and active protection for armored vehicles*. The installment at hand consisted of a sensor package of day cameras, in addition to automatic sensor-side processing. On top of this we built an application for communicating sensor data from the UGS to the GCS, as well as a simple GUI for the operator at the GCS.

2.1.1 Sensor platform configuration

In the experiments Argus was used in a configuration with five wide-angle day cameras, as well as a PTZ day camera. The wide-angle cameras collectively provide a Field-Of-View (FOV) of just over 300 degrees, while the PTZ-camera provides long-range visual coverage. The sensor platform is shown in Fig. 2.2. In addition to the cameras the UGS consists of a GPU-enabled desktop computer, and UHF and VHF radios, as shown in Fig. 2.1.

The five wide-angle cameras generate 4 megapixel images at 6 frames per second, so each camera generate a stream of 576 Mbps of data. For change detection and tracking these images must be uncompressed, in order to detect small pixel changes in the images. The PTZ day camera is a full HD camera operating at 30 frames per second. This video stream is not used for change detection, thus it can be compressed. The compressed video generates approximately 5 Mbps of data. However, if the change detection and tracking processing is done locally, as done in this experiment, the bandwidth can be reduced significantly. The video stream from the five wide-angle cameras should then only be used by the human operator, and can be down-sized and compressed. In this case we assume that the image size is reduced to one fourth and then the video stream from the five cameras will approximate to 10 Mbps of data. Combined with the PTZ day camera the minimum bandwidth for the network in the ideal case where all video is transmitted would be 15 Mbps. However, this is not available and therefore we have in this experiment chosen an approach where only still images are sent.

2.1.2 Sensor data processing

Processing of sensor data is based on the Argus architecture, consisting of several parallel processing pipelines fused in a generic object tracker. An overview of the processing pipeline is shown in Fig. 2.3.

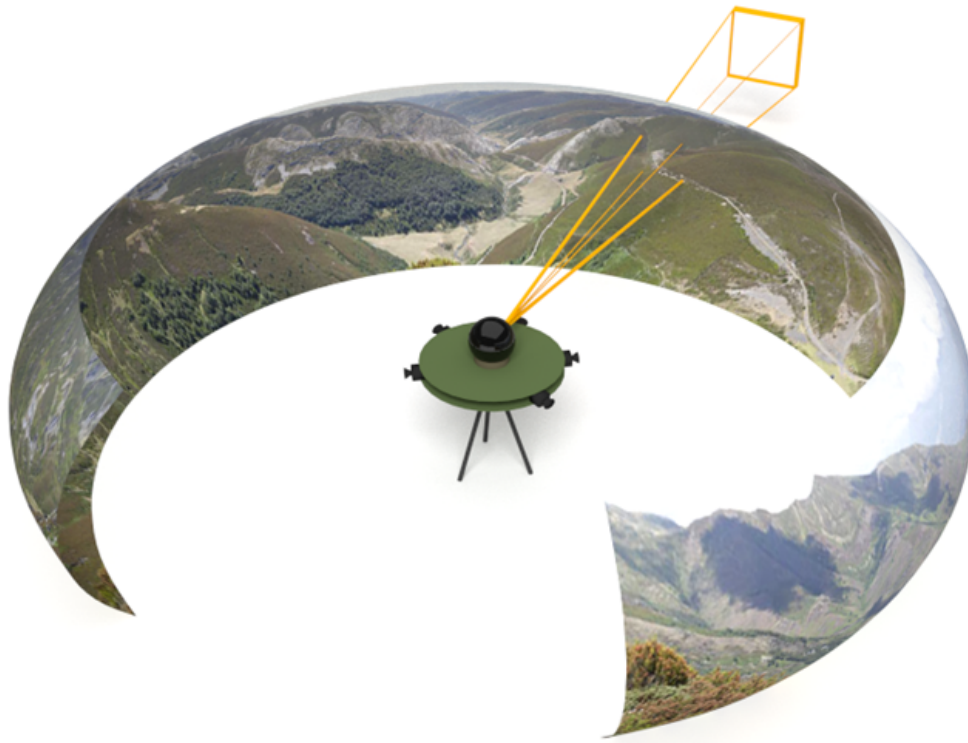


Figure 2.2 Argus sensor platform, as used in the experiment. Five wide-angle day cameras provide near 360 degree short to medium range coverage, whereas the PTZ-camera in the center provide long-range coverage.

Tracker

The *tracker* component is responsible for compiling the high level situational awareness provided by Argus. It does this by associating information across time and sensors, deducing which detections that stem from the same real-world objects, and which are false-alarms. The result of this is a list of object descriptions with position- and velocity estimates, classification and associated image highlights.

Short range detection

Short range detection of vehicles and personnel is done directly on the 360-images using Deep Neural Networks (DNNs). In the experiments an off-the-shelf open source implementation [2] of the Single Shot MultiBox Detector (SSD) [3] was used. For each image a set of detections is extracted and later fed to the tracker. This is high quality information which provides the tracker with both position estimates, object classification and confidence estimates.

In order to conserve computational resources the detection efforts are intensified in preconfigured areas of the 360-images. This configuration is later referred to as the *SSD configuration*. Using the cameras at hand, this model has an effective detection range of up to just over 100m against

personnel and up to approximately 300m against larger vehicles.

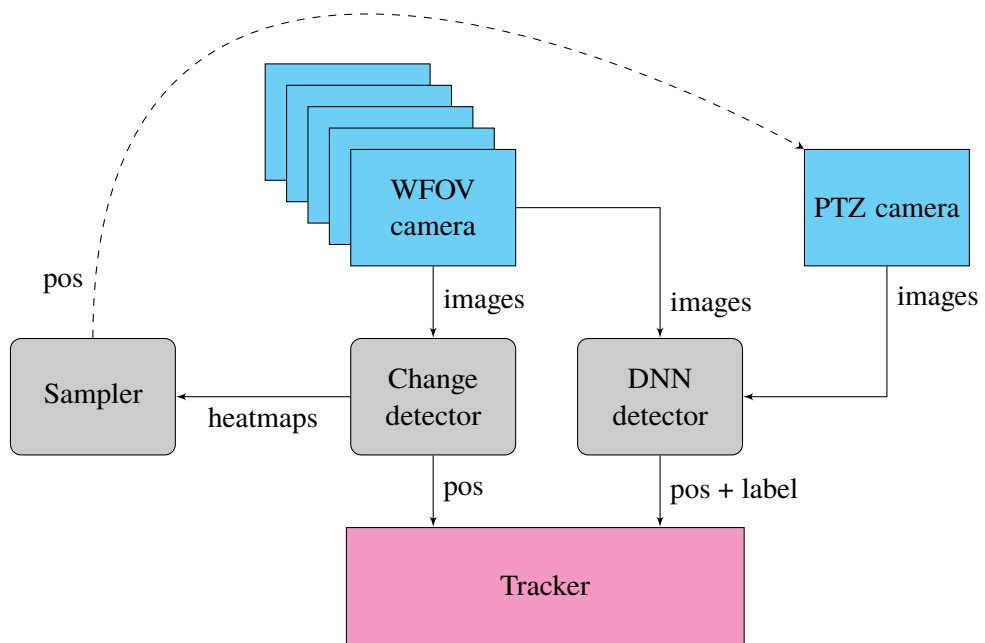


Figure 2.3 Overview of the Argus processing pipeline.

Medium range detection

Medium range detection (more than 100m, less than 1000m) is more involved, and requires a multistep process leveraging both the 360-cameras, the PTZ and the DNN detector. The key idea is to use the 360-cameras to queue the PTZ, which via the DNN detector produces high quality detections for the tracker. Simultaneously, the 360-cameras produce low quality detections, which enables the tracker to update established tracks while the PTZ is pointing in another direction.

The process starts with detecting pixel changes on the 360-images. This is done using the mixture model background subtraction implemented in OpenCV [4] based on [5] and [6]. This method produces a foreground mask, as shown in Fig. 2.5, which then is used both for queuing the PTZ as well as producing low quality detections.

Through a simple decaying sum the foreground masks are temporally fused, thus producing an activity heatmap of the surroundings, as shown in Fig. 2.6. This heatmap is consumed by the *sampler*, which is responsible for selecting the next position the PTZ should be moved to. The sampler selects regions with significant activity, while optimizing for PTZ round-trip efficiency and area coverage. Once the PTZ has moved to a new position, it captures a high resolution image of the target. As for the images from the 360-cameras, these narrow-FOV images are fed through the DNN detector, producing high quality detections for the tracker, as shown in Fig. 2.7.

Each contiguous area of foreground is merged and used into a *detection*, which are fed directly to the tracker. As can be seen from Fig. 2.5, there is not enough information in the wide-FOV images at these ranges to give a robust classification or range estimate. The position of each foreground blob does however help the tracker update the relative angular position of the tracked objects.

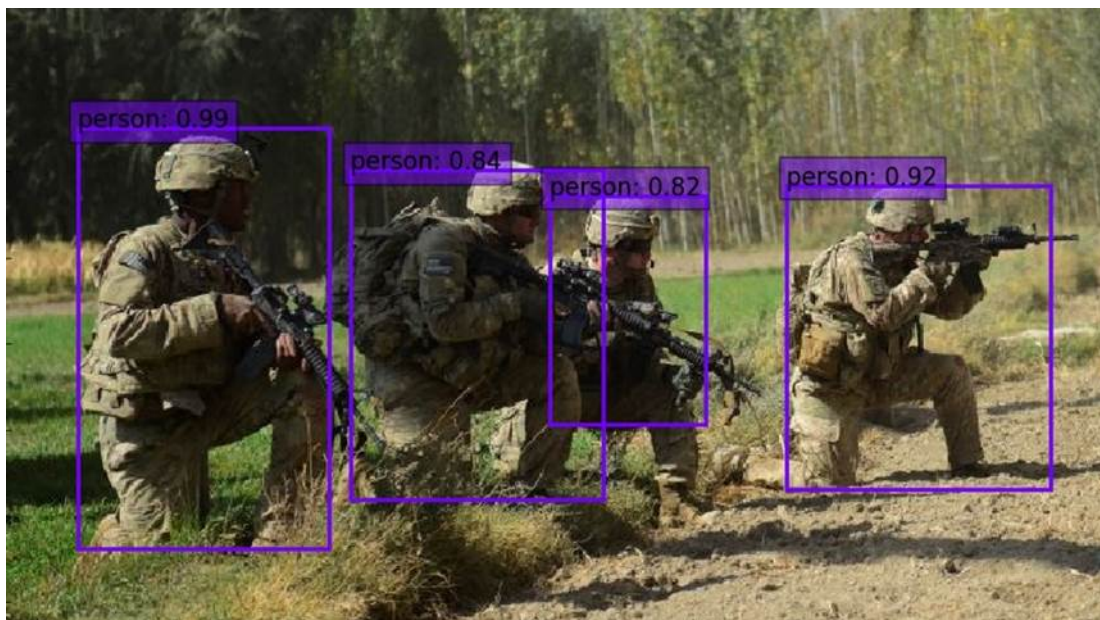
2.1.3 Operator's interface

The display node receives information from the monitoring and tracking node and displays it for the operator. The display is shown in Fig. 2.8. It shows a large window with an overview image of the scene. Each object of interest is shown with a cross on that image, and an image of the object of interest are shown in an additional window. In addition one window shows a map around the sensor, and observation is shown in that map.

2.2 Application and network communication

The application is written using the distributed framework ROS. ROS is widely used in robotics projects world wide, and is used here to have the software in the same ecosystem as the software of the UGV. The application utilizes the publish/subscribe model implemented in ROS, and the application is divided into standalone components which are termed *nodes* in ROS. The application consists of two nodes, a producer node and a display node. The producer node is responsible for monitoring the scene and tracking objects of interest, and produces data for the display node. The display node visualizes the data from the producer node. Both nodes have a communication component that keeps track of the data state and retransmissions, and this will be described later in this section.

In ROS there is a central component named the *ROS master* which manages the connection between the nodes. This creates network traffic that we do not control. ROS uses the Transmission Control Protocol (TCP) transport protocol. For node-internal communications and communications



(a)



(b)

Figure 2.4 Detection of personnel and vehicles using DNNs, in particular [3]. Out-of-the-box these have low false-alarm rates and good detection rates, but struggle with small objects which are rare in the original training set.



(a)



(b)

Figure 2.5 Foreground extraction on the Argus 360-images for medium-range detection. A section of the 360-image is shown in Fig. 2.5a. The corresponding foreground mask is shown as a red overlay in Fig. 2.5b, highlighting two cars and some activity at the construction site.

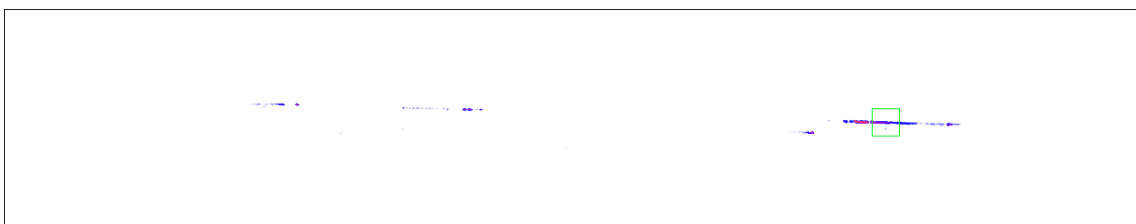


Figure 2.6 Activity heatmapping on the Argus 360-images for medium-range detection. Foreground masks are extracted as shown in Fig. 2.5b, and then temporally fused resulting in the heatmap above. White indicates no activity, blue low activity, purple medium activity and red high activity. The green rectangle indicates the region the PTZ camera is about to be directed to.



Figure 2.7 DNN detector run at range on a typical high resolution image from the PTZ camera.

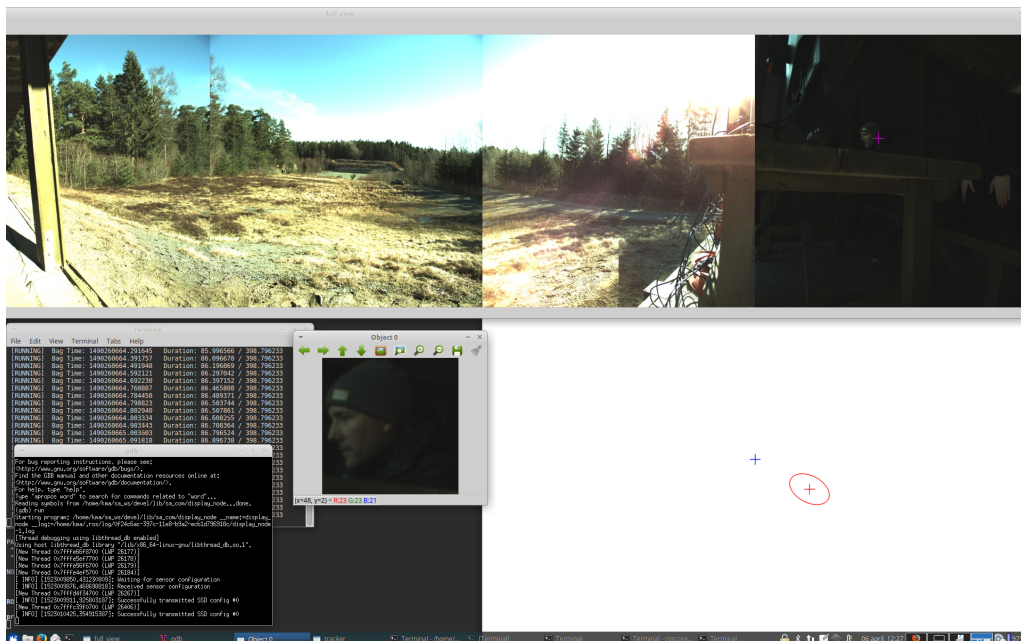


Figure 2.8 The display of sensor data for the operator. The overview image is in the upper part of the picture, the map with objects is shown in the lower right part, and an object of interest is shown in the lower left part.

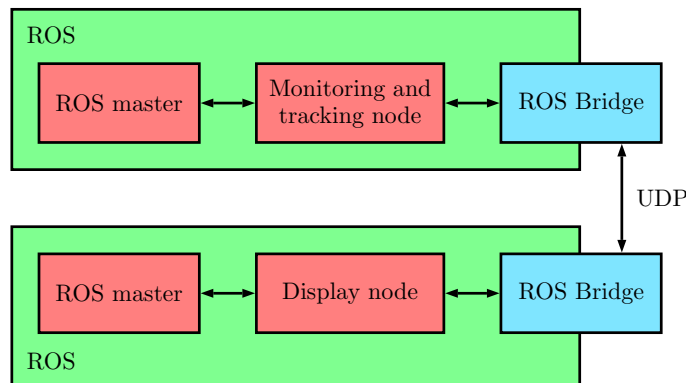


Figure 2.9 Data is transmitted in the application layer using ROS publish/subscribe model. The two nodes “Tracking and monitoring” and “Display” are deployed in two different ROS environments that communicate through a ROS bridge.

over wired networks, TCP is well suited, as it supports both congestion control and retransmissions, allowing applications developers to view the TCP-connection as a byte-stream transfer. However, TCP, a protocol developed for wired networks during the early years of the Internet, encounters several challenges [7] in wireless networks. To avoid using TCP over the wireless links, a ROS-bridge consisting of two separate ROS deployments, one at the sensor side with the producer node, and one at the operator side with the display node, was set up. We then have two ROS masters, and we use a bridge between these two ROS deployments in order for the two to communicate with each other. The bridge consists of the ROS nodes, one on each side, and forwards ROS messages using User Datagram Protocol (UDP). This is illustrated in Fig. 2.9. The ROS bridge manages connections between the two ROS environments, and is configured by the user. The configuration includes which topics that should be made available by the the bridge, and which topics that are wanted from the other bridge. It is also possible to specify a different Differentiated Services Code Point (DSCP) value for each topic. The DSCP is a 6-bit tag in the IPv4 header that can be read by the forwarding nodes in the network, allowing packets with different DSCP to be treated differently. One ROS bridge node informs the other bridge node that it should send messages across only when a node in the ROS network has started to subscribe to that specific topic. Thus, the ROS bridge nodes hold a state for each of the topic they have available and want, depending on which topics that are subscribed to in their ROS environment and what the other ROS bridge node requests. The bridge was provided by Kongsberg Defence Communications.

In the communication between the producer node and the display node there are two phases. First there is an initial phase where configuration parameters are shared. After this phase a continuous operation phase starts. This is shown in Fig. 2.10. In the initialization phase the producer node first reports which sensors are available to the display node. Next, the display node transmits the SSD configuration which the producer node should use. In both cases the sender keeps sending the message until an acknowledgment is received from the receiver.

In the continuous operation phase there are three streams of data: a 360 overview image, zoom images of objects of interest and tracker information. The 360 overview image is constructed using the five cameras on the sensor. The image gives the operator an overview of the surroundings of the sensor, and makes it easier to visualize where observations are made. The image is scale down to one fourth of its original size, but is still approximately 5 megapixels which is so large that it

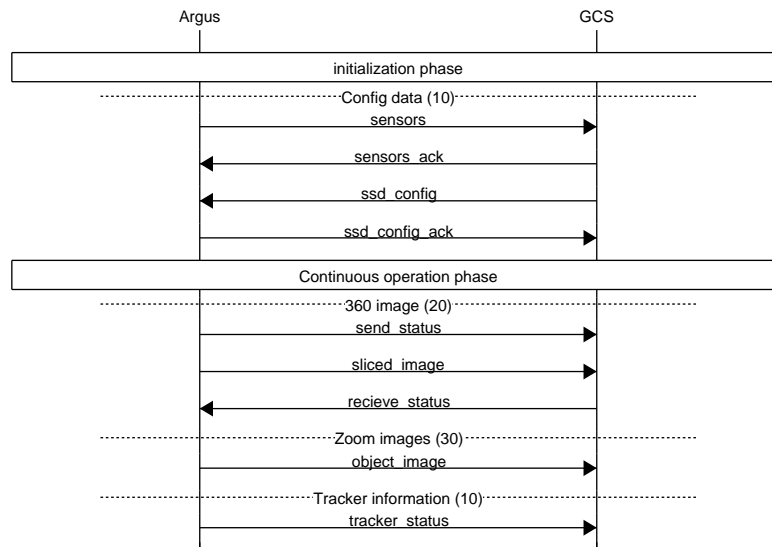


Figure 2.10 Data exchange between the producer node Argus and the display node GCS

cannot be transmitted as one full image over the wireless links. The slice size i configurable, but is typically between 32x32 to 128x128 pixels with 3 8-bit channels. Image compression is used to lower the size of the slices, but in most cases the slice will be larger than the Ethernet MTU size of 1500 bytes. This means that the slice is divided into two or more packets, and if one of these packets are lost the whole slice is lost.

First the producer nodes iterates over all the slices, and tries to send them one by one with a delay between each send attempt. This is the `sliced_image` stream in Fig. 2.10. The `send_status` stream contains information about the image that is sent, such as the width, height and number of slices. The `receive_status` stream contains a map of which slices that are received or not. After the produce node has sent one of each slice it checks whether a slice was received or not, and resends the slice if it was not received. When all slices are received the producer node starts over again and begins to send all the slices. Keep in mind that the overview image might change slightly over time and should be updated continuously.

The second stream is the zoom images of the objects of interest. These are sent periodically to the display node. There may be more than one object that is tracked but the number of images sent per second is constant, meaning that if there are many objects each image will have a lower update frequency. The last stream is the `tracker_status` stream. This stream contains information about the objects of interest found by the monitoring and tracking software. The information is position, observation type (person or vehicle) and time of last observation.

The streams have different priorities and classes, which is indicated by the number in parenthesis in Fig. 2.10 (10, 20 and 30). The lower the number the higher priority. Through traffic engineering it is possible to have some classes of streams use a different set of links than another class of streams. The streams marked with 10 require a low bandwidth, but are crucial for the operation of the system. The streams marked with 20 are involved in the 360 overview image, and the stream marked with 30 is the zoom images. These also important to be transmitted, but we can permit a longer delay on these streams. Also they require a higher bandwidth.

SR600

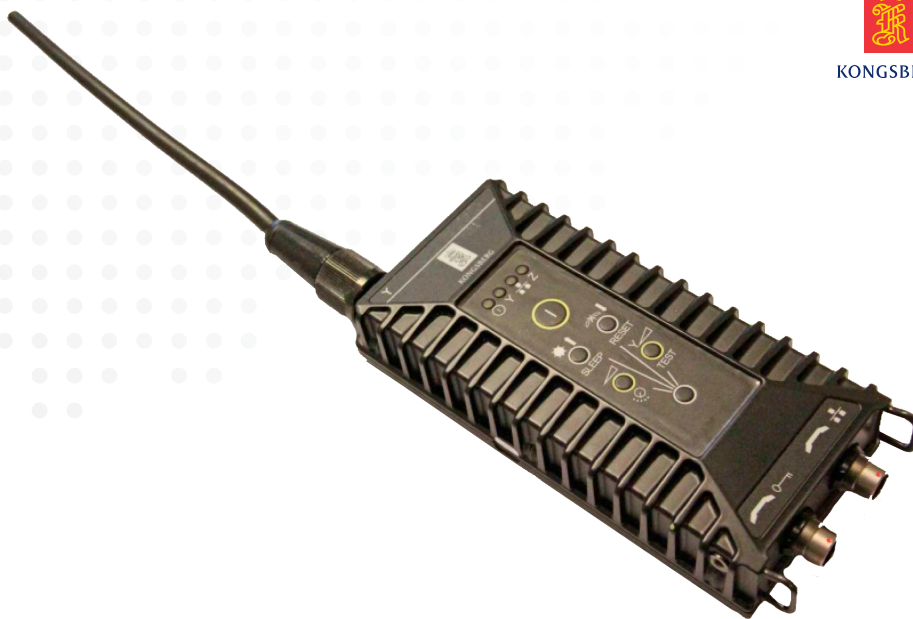


Figure 2.11 The Kongsberg TacLAN SR600, from publicly available data sheet [8].

2.3 The communications infrastructure

The communications infrastructure consisted of two different types of mesh radios. One was the Kongsberg Multi-Role Radio (MRR) radio, a narrowband VHF radio. The other was the Kongsberg TacLAN radio suite of on-air compatible wideband UHF radios. As these two technologies were utilized in one network, this constituted a heterogeneous network. The main differences between the two radios were the longer range and the lower capacity for the narrowband radio, compared to the wideband radio.

2.3.1 Kongsberg TacLAN radios

Kongsberg TacLAN radios come in three different models, SR600 (Fig. 2.11), WM600 (Fig. 2.12) and UM600 (Fig. 2.13), that are on-air compatible. The models differ mainly in their physical appearance, or form factor. The SR600 is the lightest of the radios, at 695 g plus batteries, developed to be used as a hand-held soldier radio. The WM600 is a vehicle-mounted size radio, and the UM600 is about the same size as the WM600, but specifically formed to fit in a vehicle-mounted rack together with the Kongsberg MRR radio.

The Kongsberg TacLAN radios are software-defined and offers an IP interface. The frequency range is 225-440 MHz with a bandwidth of 5 MHz. It has a selectable RF output of 10 mW to 1 W. The radio offers communication at several different link rates spanning from 128 kbps to 2.5 Mbps. The transmitting link rate is set static per radio.

WM600



Figure 2.12 The Kongsberg TacLAN WM600, from publicly available data sheet [8].

UM600



Figure 2.13 The Kongsberg TacLAN UM600, from publicly available data sheet [8].



Figure 2.14 The Kongsberg vehicle mounted MRR, from publicly available data sheet [8].

2.3.2 Kongsberg Multi-role radios

The Kongsberg MRR (Fig. 2.14) is a VHF radio with a very high range (30 km), specially developed for ground-ground communications. It offers data rates up to 19.2 kbps. The MRR applies a technology that gives the possibility of reception in negative signal to noise ratios and resistance to multipaths, resulting in improved range. The radio supports IP-packet transmission and packet radio with automatic routing. The radio combines voice, data and packet radio traffic on the same service integrated net.

2.3.3 SDN and OpenFlow

The routing of traffic through a heterogeneous network can either be done statically or using a routing protocol. The routing protocol could utilize hop count or static cost per link, or utilize routing metrics suited to choose between links with different performance. In our case, Kongsberg had implemented support for Software Defined Networking (SDN) on the TacLAN radios through our CoNSIS II-based collaboration with Kongsberg. Kongsberg Defence Systems (KDS) implemented support for SDN on their TacLAN radios in 2016. Thus, instead of only utilizing the internal radio routing protocol, we were also able to take advantage of the implemented support for SDN to perform traffic control.

SDN is an approach to computer networking where the control plane and forwarding plane is decoupled, in contrary to traditional networking. The control plane is where the routing protocol is running, and the forwarding plane is where the actual traffic is forwarded within a network node. In SDN, the control plane of a router/switch can be moved out in a new node and further be able to control more than one router/switch in a centralized node called a controller. Thus, the controller may control more than one network node/switch at the same time.

One of the benefits with SDN is the ability of a centralized controller, controlling multiple switches at the same time. SDN, especially through the popular implementation OpenFlow [9] has gained popularity in the wired network as a tool to enable network virtualization and policy enforcement. Such functionality would be beneficial also to mobile wireless networks.

Our network equipment consisted of a combination of SDN-enabled hardware and legacy hardware not supporting SDN with limited computational capability. Hence, one central controller in our scenario was seen as infeasible, although our network was of limited size. We therefore designed a network with multiple controllers.

3 Experiments

To address the questions presented in Chapter 1, we employed the system as described in Chapter 2 on a scenario that was designed to strain the applications and the communications infrastructure through several properties:

- Forcing the need for relaying information over the UHF-radios
- Introduce two relays with different performance:
 - Varying availability
 - Varying stability
 - Varying play-time
- Coordinating a long-range operation
- Utilizing a heterogeneous network

The experiments were planned for execution at Rygge Aerodrome with the nodes positioned as shown in the map (Fig. 3.1). The UGS was positioned beyond the UHF radio range of the GCS. Thus, communications via the UHF-radio would require relaying via either the UGV or the UAV. Further, the UGV was expected to represent a communications performance limitation, due to unstable UHF links between the UGV and the two traffic end-points, i.e., the GCS and the UGS.

The communications infrastructure was set up several weeks before the execution of the experiments in a testbed at FFI, and the application traffic was tested with traffic classification. During this test period, several bugs and misunderstandings were identified and corrected, both with regards to the OpenFlow scripts and assumptions on part of the application and communications infrastructure developers.

In the next sections, we describe in detail the experiments that the system test consisted of.

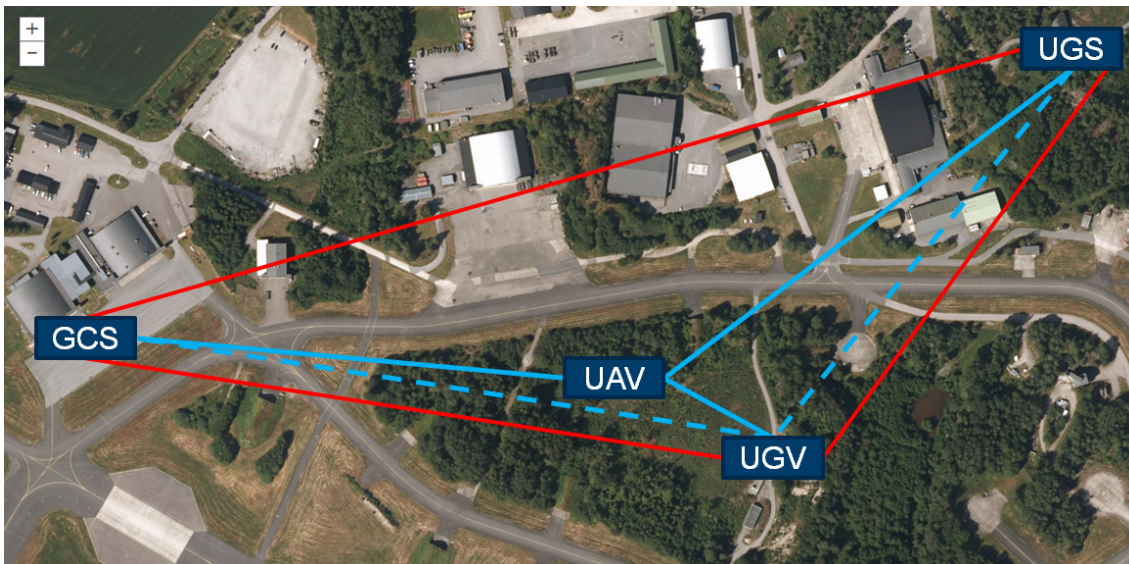


Figure 3.1 Air photo of the experimentation area at Rygge Aerodrome with the planned positions of the units and the corresponding links.

3.1 Application

In order to find the effect of different parameters in the application and the effects on the application with different communication configurations a number of experiments were performed. For each experiment the application was started and run for some minutes. All messages sent from the UGS side to the GCS side and the other way around was logged. This way we can determine which messages that were lost through in transmission and which one got through to the other side. The zoom images of the tracked objects of interest had a send rate of 0.2 Hz throughout all experiments.

The first set of experiments was conducted to check the reliability and throughput difference when using one intermediate nodes compared to having a direct link. Each of the two network configurations was tested with two different updated frequencies for sending overview image slices (0.1 Hz and 0.5 Hz).

The second set of experiments was comparing TCP and UDP connections to transport the messages. Here two experiments were conducted using a TCP connection and a low update frequency for the image slices of the overview image. One experiment used one intermediate nodes and the second used a direct link. For the experiments using TCP the ROS bridge was skipped, and only one ROS master was used. Thus both ROS messages and ROS meta data was transmitted over the network. Four experiments with UDP were conducted, using one intermediate node and varying update frequencies of the image slices (two with 0.1 Hz and two with 0.5 Hz).

The third and last set of experiments it was experimented with the size of the slices. In all previous experiments the slice size was 128x128 pixels. Three experiments were conducted using slices of 64x64 pixels and 32x32 pixels. A UDP connection over a direct link was used, and the following three experiment configurations were performed: 64x64 pixels slices and 0.5 Hz update frequency, 32x32 pixels slices and 4 Hz update frequency, and 64x64 pixels slices and 16 Hz update frequency.

3.2 Communication

The SDN-support implemented by KDS for the TacLAN radios provided us with the opportunity to employ SDN-functionality for traffic engineering in tactical networks. The traffic flow paths can be controlled on a per-policy basis based on several metrics (route selection, protection of low capacity links). The main goal of the communications infrastructure experiment was to demonstrate that traffic flows can be controlled by using SDN. This was broken down into two smaller goals:

- Directing priority traffic over stable links
- Supporting the use and protection of low capacity links

The network consisted of two different platform configurations: one elevated configuration and one ground configuration equal for all three ground nodes. The ground nodes operated with two radio networks (UHF and VHF), while the elevated node only operated over UHF, due to weight and antenna size constraints. A close-up of the experiment communications infrastructure is shown in Fig. 3.2. Each of the ground nodes included a computer running an SDN controller and an OpenFlow capable switch. A simple learning switch interconnected the controller PC, the terminals and the radios. Furthermore, each of the UHF-radios ran an OpenFlow-capable switch.

Fig. 3.3 shows our communication configuration for the ground platforms. The application computer was a computer performing application tasks outside of the network, such as collecting, processing and exchanging sensor information. The application computer was configured with a

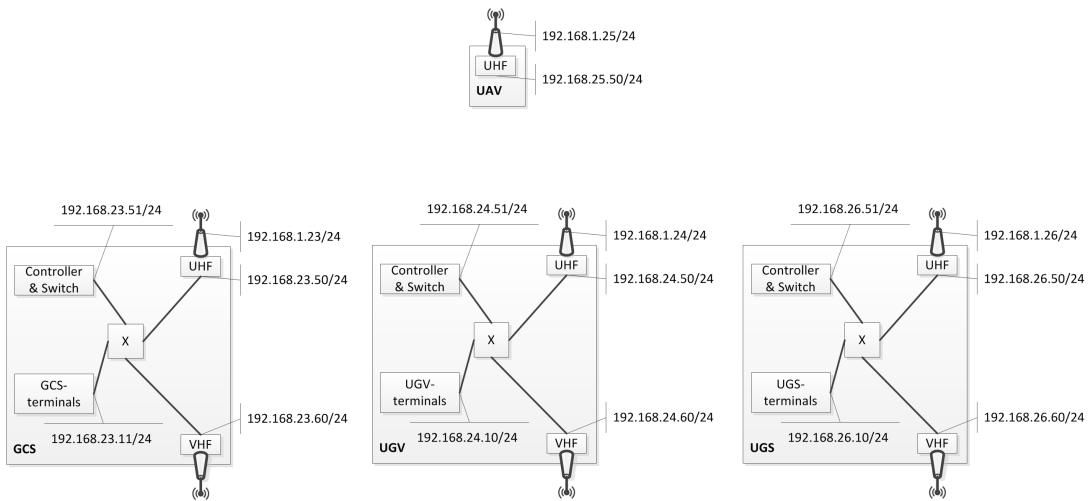


Figure 3.2 The network architecture.

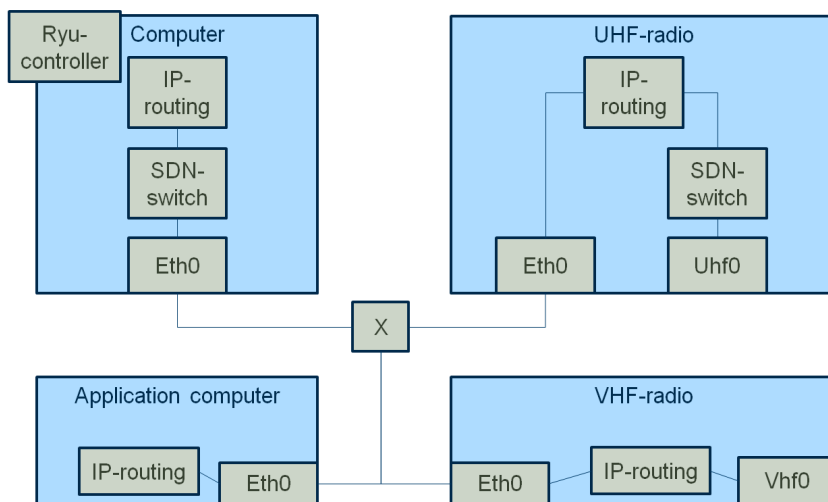


Figure 3.3 The ground node communication architecture.

default route entry to the computer. The computer was operating as the platform SDN-controller and it controlled two switches: one in the computer (onboard) and one in the UHF radio. The onboard switch was responsible for switching traffic between the UHF and VHF networks. The UHF switch was set to traffic engineer traffic over UHF radio network. The UHF radio network was running Wireless Open Shortest Path First (WOSPF), i.e., traditional distributed routing. Our goal by running SDN along with IP routing was to steer marked traffic over the UAV in cases where the routing protocol selected equal distance path at ground. SDN was only used for traffic engineering (i.e., changing the traffic paths) on the stationary ground platforms, since there was no point in changing the route for traffic that had already reached the UGV on its way either towards either the UGS or the GCS.

Wireless mobile networks have many characteristics that are not shared with the wired network counterpart. Mobile wireless networks often experience topology changes, and links are prone to error. Thus, when employing SDN in our experimental radio network, some design choices had to be made that were different from those best-practices for wired networks. An in-depth look at the design choices has been documented in [10] and the code for the experiment has been documented in an internal report [11].

To achieve the traffic engineering, the applications marked the packets that needed different service in coordination with the mechanisms in the communications infrastructure:

DSCP=10 Low capacity requirement, but essential for the operation of the application. Forwarded over MRR.

DSCP=20 Normal traffic to be forwarded over the UHF radios.

DSCP=30 Important traffic to be forwarded over the UHF radios. To be forwarded over the UAV when available.

The goal of the communications infrastructure experiment was to confirm that the traffic generated with DSCP value of 10 was forwarded over the MRR radio network, and further to establish that the traffic marked with a DSCP of 30 was routed over the UAV when the UAV was available.

4 Results

The executed experiment scenario differed some from the planned scenario. Due to problems related to the compass and control system of the UAV, we were not able to run the experiments as planned. Due to the UAV platform problems, we had to improvise the stable link that was going to be provided by the UAV. In practice, this was sought solved in two ways during our experiment campaign: first by elevating the SR600 on a flag pole, and second by placing it on a tripod on a small hill right by the flag pole. We continue in this report to refer to the SR600 intended for use as an elevated relay as "the UAV" for recognition purposes, and since the main function of the UAV was to provide a stable relay, as opposed to the UGV's expectedly more unstable relay function. The selected position of the UAV, albeit very close to the UGS, was at a small hill around 10 m higher than the position of the UGS, making it a suitable location for a relay. We argue that although the SR600 relay would have provided better link conditions in an elevated position, the most important effect of this relay was providing a stable relay. The chosen position and placement attempted to make this possible, even though it was on the ground. The resulting positions of the network nodes are seen in Fig. 4.1.

The GCS was conveniently set up in a hangar that we had at our disposal. In the hangar, there was power and a large space to set up all needed equipment for communication and application programming, as well as a staging area for the UGV. The antennas at the GCS were positioned on a pallet just outside the hangar doors, and after some testing, the pallet was lifted up on a mobile airstair. To achieve a multihop topology via the UGV or the UAV, the airstair had to be moved close to the wall, to take advantage of a building shadowing the LoS to the UGS.

The UGV had the antennas mounted on the cargo door. The UGV itself moved along most of the horizontal taxiway as seen in Fig. 4.1 when mobile. When static, the UGV was mainly located in the position showed in Fig. 4.1.

The UGS was set up at a shooting range at the aerodrome. Seen from the GCS, the shooting range was positioned just behind a small hill, which was expected to block the link between the GCS and the UGS. Fig. 4.2 shows the setup of the UGS with antennas at the shooting range. The antennas were placed on a pallet, placed on top of two boxes, bringing it about 1 m above the ground.



Figure 4.1 The positions of the network nodes in the final experiment setup.



Figure 4.2 The UGS sensor setup with the UHF and VHF antennas on the left.

4.1 Application

The result for the first set of experiments are given in Table 4.1. Here the number of messages sent from the UGS is reported, along with the number of received messages at the GCS.

For the second set of experiments there were some problems. For the TCP experiments no data came across the network when using two intermediate nodes. With only one intermediate node some tracker state messages came through before the network was saturated, and no messages came through. For the UDP experiments there were some problems with the connectivity of the UHF links, and no image messages came through, only tracker state messages as these were sent over the VHF links.

Table 4.1 Transmitting data over UDP using one or two intermediate nodes (or jumps), and using either 0.1 Hz send frequency for image slices (low freq.) or 0.5 Hz (high freq.).

Jumps	Message	Low freq.			High freq.		
		Sent	Recv	%-loss	Sent	Recv	%-loss
1	Image slice	20	18	10	145	91	37.2
	Zoomed image	18	10	44.4	52	22	57.7
	Tracker state	37	37	0	170	145	14.7
2	Image slice	19	0	100	82	1	98.8
	Zoomed image	10	0	100	32	1	96.9
	Tracker state	30	30	0	81	80	1.2

Table 4.2 Transmitting date using varying frequency and image slice size

Message	64x64 slices, 0.5 Hz			32x32 slices, 4 Hz			64x64, 16 Hz		
	Sent	Recv	%-loss	Sent	Recv	%-loss	Sent	Recv	%-loss
Image slice	111	99	10.8	842	402	52.3	2180	1583	27.4
Zoomed image	10	0	100	42	0	100	1	0	100
Tracker state	111	111	0	111	110	0.9	109	93	14.7

The results for the third set of experiments are given in Table 4.2.

4.2 Communication

During the experiments, tcpdump [12] was used to capture the data packets at the GCS and at the UGS, running at the computer. After the experiment, the log-files were explored using Wireshark [13], a very versatile program for examining network traffic. Using Wireshark, we were able to confirm that the traffic generated with DSCP value of 10 was forwarded over the MRR radio network. In practice we did this by asserting that the logs showed that packets received on either the GCS or the UGS with a DSCP value of 10 were forwarded via the MRR radio, and that the packets with a DSCP value different from 10 were forwarded via the WM600 radio.

Due to problems obtaining a topology with two simultaneous reachable relays, we were not able to confirm the behavior of consistently routing packets marked with DSCP of 30 over the UAV, while the other traffic was routed over the UGV. However, we did confirm that traffic with DSCP of 30 was routed over the UAV. The Fig. 4.3 shows simultaneous traffic going over VHF and UHF. This traffic was correctly forwarded according to the DSCP field value.

We observed other topologies than expected. These were probably a product of the lack of an elevated node, leading to elaborate "tuning" of the radio parameters, where the expected relays (the UAV and the UGV) had to be configured with a more robust link-rate than the end nodes (the UGS and the GCS). The most important (and problematic) link was that of the direct link between the UGS and the GCS. This impacted the routing protocol on the UHF-radios, resulting in periodic link loss and rerouting with a considerable timeout (~40 s). One such event is observed in Fig. 4.4.

When studying the log files after the conclusion of the experiment, strange behavior was discovered, where the controller computer at the UGS had received numerous packets destined to the UGS application computer. The computers at the UGS were interconnected using a simple Medium Access Control (MAC)-switch. This switch is a learning switch which maps the received packets' MAC-source address with the respective incoming port. Based on the normal Address Resolution Protocol (ARP) discovery of MAC-clients, it is thereby able to forward the MAC-packets to the proper local destination. However, the SDN code that was implemented for forwarding traffic from the sensor application computer to the MRR-radio had a bug in it, wherein it did not change the MAC-source address for packets relayed from the controller computer to the MRR-radio. The result was that every time the controller computer at the UGS forwarded a packet to the MRR-radio, the simple switch updated its MAC-address mapping to point to the controller computer for the sensor application computer MAC address. In a way, the controller computer became a black hole for traffic coming from the radios to the sensor application computer. The consequence was that the handshake behavior of the application encountered a major challenge. However, all packets that were to receive UHF-radio forwarding would correct the switch mapping, since these packets were routed normally by the controller computer, including setting a new MAC-source address.

It turned out that the Kongsberg TacLAN radios were fitted with a software version optimized for a single-sender scenario. It was made to work better with video transfer. Therefore, the MAC re-transmissions had been switched off. Furthermore, the radio was configured so that it discarded the packet ready for transmission if the medium was found to be "busy". Both in a relay situation and in a bidirectional transmission situation, this behavior could be devastating. In a relay situation with traffic bursts, for example due to large packets fragmented to fit the Maximum Transmission Unit (MTU), the second fragment from the source would likely coincide in time with the relay's transmission of the first fragment. With the current radio MAC behavior, one of the fragments would be discarded, leaving the destination lacking this fragment. Without an end-to-end transmission protocol that takes responsibility for re-transmission, the entire packet would be lost. Even if the transmission protocol were to re-transmit the packet, this would mean re-transmitting also fragments that were successfully transmitted earlier.

On recommendation from KDS, a fix was implemented which delayed the transmissions from the expected relay nodes by $100\text{ ms} \pm 50\text{ ms}$. In the lab, this implementation gave the wanted effect, in that the packet loss was reduced. However, a side effect of this delay implementation was that the maximum transmission rate was reduced to 10 packets per second. Furthermore, the problem of discarding packets if the medium was found "busy" was not handled. Any performance tests were thus discarded, as they could not be trusted to give any significant performance improvement when selecting a stable relay over an unstable direct link or when comparing the performance of two relays, one stable and one unstable.

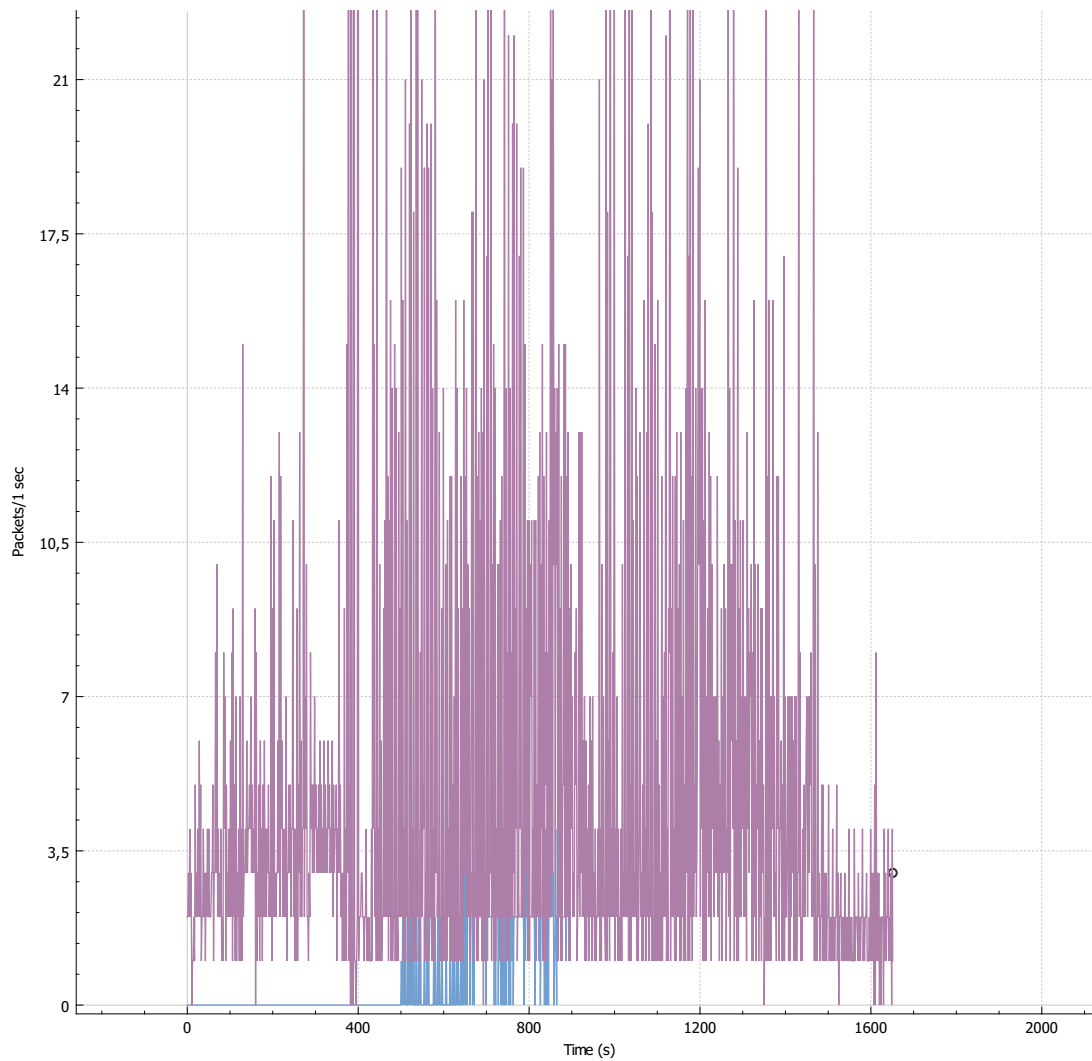


Figure 4.3 Packets received over VHF (blue) and UHF (purple) simultaneously.


```
Terminal - labuser@tough1: ~/mgen
File Edit View Terminal Tabs Help
rtt min/avg/max/mdev = 12.919/20.659/57.922/16.668 ms
labuser@tough1:~/mgen$ ping 192.168.23.50
PING 192.168.23.50 (192.168.23.50) 56(84) bytes of data:
64 bytes from 192.168.23.50: icmp_seq=1 ttl=62 time=56.9 ms
^C
--- 192.168.23.50 ping statistics ---
2 packets transmitted, 1 received, 50% packet loss, time 1001ms
rtt min/avg/max/mdev = 56.945/56.945/56.945/0.000 ms
labuser@tough1:~/mgen$ ping 192.168.23.50
PING 192.168.23.50 (192.168.23.50) 56(84) bytes of data:
64 bytes from 192.168.23.50: icmp_seq=9 ttl=63 time=20.3 ms
From 192.168.26.50 icmp_seq=10 Destination Host Unreachable
From 192.168.26.50 icmp_seq=11 Destination Host Unreachable
From 192.168.26.50 icmp_seq=12 Destination Host Unreachable
64 bytes from 192.168.23.50: icmp_seq=18 ttl=62 time=55.7 ms
64 bytes from 192.168.23.50: icmp_seq=19 ttl=62 time=12.4 ms
64 bytes from 192.168.23.50: icmp_seq=20 ttl=62 time=12.5 ms
64 bytes from 192.168.23.50: icmp_seq=21 ttl=62 time=12.5 ms
64 bytes from 192.168.23.50: icmp_seq=22 ttl=62 time=12.3 ms
^C
--- 192.168.23.50 ping statistics ---
22 packets transmitted, 6 received, +3 errors, 72% packet loss, time 21064ms
rtt min/avg/max/mdev = 12.317/20.999/55.793/15.826 ms, pipe 3
labuser@tough1:~/mgen$
```

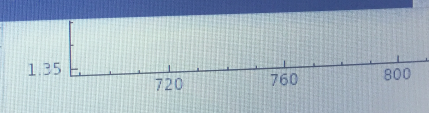


Figure 4.4 Observed rerouting with considerable timeout in Ping-application.

5 Discussion

This chapter discusses various aspects and lessons learned through the preparations and execution of the experiments. Several of the issues that arose were focused around the interaction, or lacking interaction, between the application and the communication network. This discussion chapter also provides the deeper reasoning behind the contributions to the research questions put forward in Chapter 1, that we present in the conclusion chapter.

5.1 Application performance

When looking at the results in comparing a direct link between two radios with the case where there is one intermediate node in Section 4.1 we see that the loss rate is much higher in the latter case. Interestingly, only “Image slice” and “Zoomed image” have high loss ratios, and “Tracker state” messages has a low loss ratio. The first type of messages are transmitted through the UHF radio, while the latter is transmitted through the VHF radio. This indicates that something happens when the messages need to be forwarded through the UHF radio. Also, it is likely that the VHF messages are transmitted directly to the destination, because of the extended range of that radio.

The second experiment was comparing the throughput by using UDP or TCP. Our hypothesis that TCP would have lower throughput than UDP was confirmed, as no packages made it through using TCP. There is another aspect of this experiment than just comparing UDP with TCP. ROS messages are transmitted through TCP, and this means that a bridge between the ROS networks are required, as the radio network is not able to transmit TCP packages effectively.

In the last experiment assessing the application performance we compared different sizes of the slices sent over the network. When comparing the high frequency column in Table 4.1 with the first column in Table 4.2, these data are sent with the same frequency. We see that for a smaller image slice (64x64) there is less loss. We saw during the experiment that it was beneficial to have the messages lower than the MTU size (1500 bytes), as this increased the likelihood for the messages to come through. When we further decrease the image slice to 32x32 we see that the loss ratio increases. The packages in this case might become too small for efficiently utilizing the bandwidth of the radios, and this might cause higher loss ratio. Then we try to use 64x64 slices and increase the frequency. This case has a loss ration in between the two fist cases, but has the highest throughput of image slices.

5.1.1 Practical issues affecting the performance

While involuntarily, the unexpected challenges for the communications infrastructure proved a point with regards to the need for applications to better understand the performance of the underlying communications infrastructure, and also be able to adapt traffic in the situation where the communications infrastructure does not perform as expected. The application performance suffered, due to several problems with the communications infrastructure:

- Missing UAV
- TacLAN relay problems
- SDN-code bug

Due to the missing UAV, the topology did not consist of the expected stable paths between the GCS and the UGS. Even though the UAV-radio was placed in what was expected to be an optimal location, forming the topology to fit with the planned experiment proved difficult to achieve, at

least for any longer periods of time. At the same time, observing the topology over any extents of time with direct feedback to the users was not implemented, leading to the observed application performance not to be evaluated directly against the current communications infrastructure topology.

The TacLAN relay fix created a bottleneck for the traffic going over the UHF-radios. Even though the application had implemented mechanisms to perform rate control, the low UHF-radio rate reduced the application performance unnecessary.

Finally, the SDN-bug made the application initialization problematic, since the setup over the UHF-radios was hampered.

Part of the challenge with the communications infrastructure was the code being a quickly developed prototype. This meant that both the communications infrastructure and the application adaption with regards to the use of DSCP to choose which traffic should go over which paths and radio networks where tailored to the experiment. In an updated configuration, the support for applications in the communications infrastructure should be more dynamic, with easier configuration and propagation of the changes throughout the network. Further, implementing a test framework to test the performance of the communications infrastructure, and whether it handles the traffic in the expected way, should be implemented.

5.2 Applications use of the communications infrastructure - ROS bridge

The concept of implementing an application in ROS and then use a bridge to connect the components in the application though a radio network is very elegant. Unfortunately it becomes more complicated in practice and the application developer have to keep in mind the bandwidth limitations when creating the application.

When beginning to develop the application no thought was given to the bandwidth constraints of the network. The application was developed and tested without using the radio network, and was working when both parts of the application was in the same ROS network. However, when we did the initial tests using radios the communication failed. We had to redo parts of the application to take into account the low bandwidth of the radio network. What we learned from this is that it is important as an application developer to think of the bandwidth constraints of the communication already when starting to design the application.

When starting to develop the application we had the hypothesis that using ROS for inter process communication would be beneficial, and a bridge between two ROS networks could mask the underlying network. In this case the application could be used with different network configurations, and the ROS bridge could be used when necessary. After finishing the experiment we are not so sure about this anymore. The fundamental problem is that you must design you application with bandwidth limitation in mind.

There are several types of traffic between the two components in the application (the sensor platform and display). In a traditional ROS network, without using a bridge, this does not matter since TCP is used over a reliable high capacity network (wired Ethernet). However when sending over a unreliable low capacity link this matters. The streams that send all available data in one packet, and consecutive packets are just updated measurements, can be filtered by reducing the frequency they are transmitted. Also no re-transmission is required since it is better to just receive a newer message. Configuration messages that should be sent once, but must be received by the other part needs to be re-transmitted. In our case this was implemented in the two components in the application.

We found that a better approach would have been to increase the functionality of the bridge so the bridge could be responsible for re-transmissions. However, there are certain limitations in having a bridge since the application developer would have the underlying network abstracted away. In our opinion it would be better to develop the communication between the two components of the application directly using UDP. Since this would have given the application developer more freedom to choose appropriate schemes for different type of stream, and given the the application developer more sense of responsibility of how the network communication would be. This is especially the case for the zoomed images. It is very hard for the bridge to know which zoomed image to drop. Ideally one zoomed image of every object should be received by the display component before new zoomed images are sent. This was very hard to achieve by the current configuration.

5.3 Serving the application from a communications infrastructure perspective

Applications are limited in understanding and affecting the performance delivered by the communications infrastructure. This is actually due to the successful network layer model that is the basis for IP-based networks. However, allowing the users¹ to enforce more control over the behavior of the communications infrastructure is important, in order to optimize the use of the available network resources to serve the applications and through this the users' experience.

In the communications infrastructure experiment, we showed that the applications could receive a more refined service, depending on the available resources, through network mechanisms implemented through SDN. These mechanisms were used to direct the traffic for some of the applications over the UAV when available, and if no end-to-end high capacity radio was available at all, the application traffic with high capacity demands was stopped before entering the network, to avoid starving the application traffic that after all was both high priority and consuming so little data rate that it would receive acceptable service over the low capacity network links. This protected the network from abrupt deterioration when changes occurred, and applications users who knew the limitations during the experiment were also able to receive better service from the communications infrastructure than if these limitations were not known. The applications were also signaling the needed treatment from the communications infrastructure by using the DSCP code field in the IP packets.

However, the results from the experiments showed that even in our limited configuration, understanding the current state of the network, and regulating the applications is a complex task. Defining the topology and positioning the network nodes in positions that would create the wanted topology was harder than expected, and better visualization tools for understanding the current network topology were left desired. The complexity generated by a heterogeneous network in a small configuration of four nodes with two of these communicating, was high. More sensors will only add to this complexity.

Another observation from the experiments was the need to monitor the current available capacity and provide the user and/or the application with this information.

¹Users here meaning either human or autonomous entities.

5.4 How to contribute to improving the performance of the communications infrastructure

The users of the communications infrastructure, (users, platforms and applications) may also contribute to improving the performance of the communications infrastructure, to benefit all users of the communications infrastructure. The network topology may be modified and optimized through the behavior of the mobile network nodes with regards to other mobile and stationary network nodes. For instance, unmanned platforms could adapt their trajectories to provide for a better communications infrastructure, instead of only positioning themselves to optimize their own use of the communications infrastructure and their primary objectives.

With more advanced autonomous vehicles, the amount of information transferred to and from these vehicles may impact the performance of the vehicles. In some cases, reducing the information flow to one vehicle may improve the information flow to another vehicle, and the result could be a combined better performance from both vehicles. Finding ways to express the value of information to allow the network to strike the best balance between the information flow to multiple units could be beneficial.

One rule of thumb is to expect no guarantees from the communications network. Packet loss will occur, and the available transfer rate could be much lower than expected. Understanding what happens if overloading the network will also be beneficial to the users, since many link and network technologies have problems maintaining maximum performance when faced with congestion. Instead, pushing too much traffic into the network can lead to detrimental effects for the performance.

The users and applications will also help the performance of the communications infrastructure by measuring how well the traffic has fared through the network, and scaling back the traffic load if congestion seems to be indicated. However, some technologies encourage pushing all traffic one can muster onto the network, such as Demand Assigned Multiple Access (DAMA). This is a behavior which will have negative impact in a network such as ours.

Evaluating the need to re-transmit data should be part of any application's use of the network. The timeouts before re-transmissions should be tuned, and re-transmissions should be restricted to important data, and less important data should not be re-transmitted unnecessarily.

Finally, while the network should have measures to limit the amount of traffic entering the network to what resources are available, the users and applications may be better off by avoiding to overload the network.

6 Conclusion and future work

The report has aimed to shed light on the challenges and possibilities for autonomous unmanned sensors in tactical communications networks through several experiments using autonomous platforms. We have done this through addressing the following questions:

1. What specific requirements do autonomous observation systems put on communications?
2. How can autonomous observation systems contribute to improve the performance of the communications infrastructure?
3. How can the communications infrastructure meet these requirements?
4. What is the gap between the existing communications infrastructure and these requirements?

In the paragraphs below, we present our conclusions for the research questions based on our experiences as described in this report.

The requirements that autonomous observation systems put on communications can be very high, based on the potential data production that can take place in such a system. Our experience indicates that the system would benefit from knowing more about the underlying network and the availability of resources in the network. Then the sensor would know which streams of data that can be sent with a lower frequency or stopped. There is also a need to cope with different types of streams, which is challenging, since the network may not know what packets that could be dropped without consequence and which that should be kept. This information is only known by the application. Enabling the exchange of this information between the application and the communications infrastructure would be beneficial and should be researched.

The autonomous observation systems can contribute to improve the performance of the communications infrastructure through several measures. The systems should allow local control of the network resources. I.e., all carriers should be made available for the communications infrastructure, and stovepipe systems where one carrier is used for one application purpose should be avoided. Further, the autonomous observation systems can also be allowed to reposition themselves if possible to improve the communication links, either for themselves or for other platforms that can achieve better communications, and the systems must by itself evaluate whether a potential degradation in sensory performance can be defended because of the improved communication service for the other platform.

The communications infrastructure can meet the requirements put by the autonomous observation systems through several measures and mechanisms. Two clear suggestions are allowing more dynamic use of the resources, for instance through traffic control, and having an interface towards the applications to signal available resources

We have established that there is a large gap between the employed communications infrastructure and the requirements of the autonomous observation systems used in the performed experiments. First and foremost, the limited communications capacity put a strain on the usability of the autonomous observation systems. Secondly, the flexibility of the communications using the standard mechanisms available is very low. In the performed experiments, the flexibility was increased through SDN-based traffic engineering. The methods showed a high potential, even though it was not fully realized in the experiment campaign.

Evaluating the autonomous sensors and communications infrastructure requirements through experimentation provided a valuable arena where application developers and communications infrastructure developers became more acquainted with each others possibilities and challenges when connecting data producing sensors at the edge of the communications infrastructure.

From the experience gained from the performed experiments, future work to address the problem statements of this report further could consist of more combined efforts to adjust expectations, and improve the effectiveness and efficiency of the communications infrastructure as a tool to interconnect sensors and decision makers in a joint warfare concept.

Abbreviations

ARP	Address Resolution Protocol
ATV	All Terrain Vehicle
BMS	Battle Management System
CoNSIS	Coalition Network for Secure Information Sharing
DAMA	Demand Assigned Multiple Access
DNN	Deep Neural Network
DSCP	Differentiated Services Code Point
FOV	Field-Of-View
GCS	Ground Control Station
ISR	Intelligence, Surveillance and Reconnaissance
KDS	Kongsberg Defence Systems
LoS	Line-of-Sight
MAC	Medium Access Control
MoU	Memorandum of Understanding
MRR	Multi-Role Radio
MTU	Maximum Transmission Unit
NAF	Norwegian Armed Forces
OODA	Observe-Orient-Decide-Act
PTZ	Pan-Tilt-Zoom
SSD	Single Shot MultiBox Detector
R&D	Research & Development
ROS	Robotic Operating System
SDN	Software Defined Networking
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
UAV	Unmanned Aerial Vehicle
UGS	Unattended Ground Sensor

UGV	Unmanned Ground Vehicle
UHF	Ultra High Frequency (300-3000 MHz)
VHF	Very High Frequency (30-300 MHz)
WOSPF	Wireless Open Shortest Path First

References

- [1] K. Mathiassen, M. Baksaas, L. E. Olsen, M. Thoresen, and B. Tveit, “Development of an Autonomous Off-Road Vehicle for Surveillance Missions,” in *Proceedings of IST-127/RSM-003 Specialists’ Meeting in Intelligence & Autonomy in Robotics*. Bonn, Germany: NATO Science and Technology Organization, Oct. 2016.
- [2] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, “Caffe: Convolutional architecture for fast feature embedding,” *arXiv preprint arXiv:1408.5093*, 2014.
- [3] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “SSD: Single shot multibox detector,” in *ECCV*, 2016.
- [4] G. Bradski, “The OpenCV Library,” *Dr. Dobb’s Journal of Software Tools*, 2000.
- [5] Z. Zivkovic, “Improved adaptive gaussian mixture model for background subtraction,” in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, vol. 2. IEEE, 2004, pp. 28–31.
- [6] Z. Zivkovic and F. Van Der Heijden, “Efficient adaptive density estimation per image pixel for the task of background subtraction,” *Pattern recognition letters*, vol. 27, no. 7, pp. 773–780, 2006.
- [7] E. Larsen, “TCP in MANET,” Forsvarets forskningsinstitutt, Rapport 12/01289, 2012.
- [8] Kongsberg Defence Communications. Accessed 2019-03-07. [Online]. Available: <https://www.kongsberg.com/en/kds/products/defencecommunications/>
- [9] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, “OpenFlow: Enabling Innovation in Campus Networks,” *SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, Mar. 2008. [Online]. Available: <http://doi.acm.org/10.1145/1355734.1355746>
- [10] L. Landmark, E. Larsen, and Ø. Kure, “Traffic control in a heterogeneous mobile tactical network with autonomous platforms,” FFI, FFI-rapport 18/00904, 2018.
- [11] ———, “Traffic control in a heterogeneous mobile tactical network using SDN: code listings,” FFI, FFI-internnotat 17/16390, 2018.
- [12] tcpdump. Accessed 2019-03-12. [Online]. Available: <https://www.tcpdump.org/>
- [13] Wireshark. Accessed 2019-03-12. [Online]. Available: <https://www.wireshark.org/>

About FFI

The Norwegian Defence Research Establishment (FFI) was founded 11th of April 1946. It is organised as an administrative agency subordinate to the Ministry of Defence.

FFI's MISSION

FFI is the prime institution responsible for defence related research in Norway. Its principal mission is to carry out research and development to meet the requirements of the Armed Forces. FFI has the role of chief adviser to the political and military leadership. In particular, the institute shall focus on aspects of the development in science and technology that can influence our security policy or defence planning.

FFI's VISION

FFI turns knowledge and ideas into an efficient defence.

FFI's CHARACTERISTICS

Creative, daring, broad-minded and responsible.

Om FFI

Forsvarets forskningsinstitutt ble etablert 11. april 1946. Instituttet er organisert som et forvaltningsorgan med særskilte fullmakter underlagt Forsvarsdepartementet.

FFIs FORMÅL

Forsvarets forskningsinstitutt er Forsvarets sentrale forskningsinstitusjon og har som formål å drive forskning og utvikling for Forsvarets behov. Videre er FFI rådgiver overfor Forsvarets strategiske ledelse. Spesielt skal instituttet følge opp trekk ved vitenskapelig og militærteknisk utvikling som kan påvirke forutsetningene for sikkerhetspolitikken eller forsvarsplanleggingen.

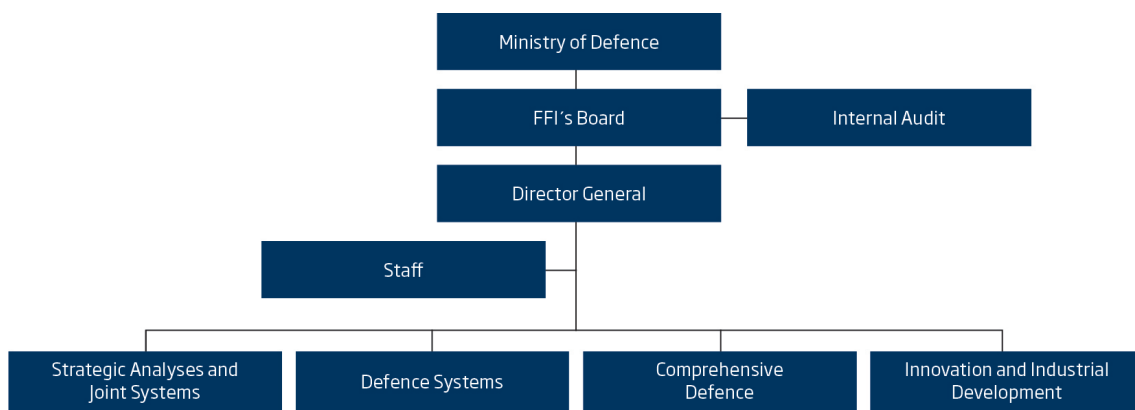
FFIs VISJON

FFI gjør kunnskap og ideer til et effektivt forsvar.

FFIs VERDIER

Skapende, drivende, vidsynt og ansvarlig.

FFI's organisation



Forsvarets forskningsinstitutt
Postboks 25
2027 Kjeller

Besøksadresse:
Instituttveien 20
2007 Kjeller

Telefon: 63 80 70 00
Telefaks: 63 80 71 15
Epost: ffi@ffi.no

Norwegian Defence Research Establishment (FFI)
P.O. Box 25
NO-2027 Kjeller

Office address:
Instituttveien 20
N-2007 Kjeller

Telephone: +47 63 80 70 00
Telefax: +47 63 80 71 15
Email: ffi@ffi.no