




RESEARCH ARTICLE

Towards a new approach to ensure end-to-end reliability of aeronautical data communications

V. Lala , A. Pirovano  and J. Radzik 

Fédération ENAC ISAE-SUPAERO ONERA, Université de Toulouse, France

Corresponding author: V. Lala; Email: soa-valencia.lala@enac.fr

Received: 11 July 2024; **Revised:** 7 November 2024; **Accepted:** 12 November 2024

Keywords: Data link communication; end-to-end reliability; ATN/IPS; COTP4; TCP

Abstract

The aeronautical telecommunication network (ATN) aims to provide reliable end-to-end communications even for those including the air-to-ground segment and in particular for data link applications. The existing ATN, known as ATN/OSI, is based on OSI protocols since its first deployment. The OSI model implementation in ATN communicating entities causes great complexity in network management, particularly in terms of Internet network interoperability. Therefore, since 2010, the International Civil Aviation Organization (ICAO) proposed a migration to ATN over Internet protocol suite (IPS), called ATN/IPS. Thus, this research work focuses on specifying the reliability mechanisms required for air ground data link applications in future ATN/IPS. To achieve this, the transport protocols performance is assessed based on simulations using an ATN model developed considering the ICAO standards. The modeled legacy application enables to generate traffic based on real controller-pilot data link communications (CPDLC) log files from French area control centre (ACC). The air-to-ground subnetworks are characterised using time series delay induced from previously modeled VDL Mode 2 data link analysis. As proof-of-concept, CPDLC messages exchange from aircraft to controller and future applications that transmits heavier files from ground-to-board are simulated. Transport protocols performance are evaluated with respect to the most constraining requirements. The simulation results highlighted the limitations of both connection-oriented transport protocol class 4 (COTP4) and TCP. This enabled to provide a preliminary overview of a new QUIC-like reliable protocol that should meet the heterogeneous requirements of the legacy and the eventual future ATN/IPS applications.

Nomenclature

AAC	aeronautical administrative communications
ACARS	aircraft communications addressing and reporting system
ACC	area control centre
ADS	automatic dependent surveillance
AIDC	ATS interfacility data communication
AIMD	additive increase and multiplicative decrease
AMHS	ATS message handling service
ANSP	air navigation service provider
AOC	airline operational control
APCs	aeronautical passenger communication
ARINC	aeronautical radio incorporated
ATC	air traffic control
ATM	air traffic management
ATN	aeronautical telecommunication network
ATS	air traffic service
AVLC	VHF link control

CLNP	connectionless network protocol
CLNS	connectionless network services
CM	context management
CN	core network
CNS	communication, navigation, and surveillance
CoAP	constrained application protocol
COCR	communications operating concept and requirements for the future radio system
CONS	connection-oriented network services
COTP4	connection-oriented transport protocol class 4
CPDLC	controller-pilot data link communications
CSMA	carrier sense multiple access
ECN	explicit congestion notification
FAA	Federal Aviation Administration
FANS	future air navigation system
FASOR	fast-slow RTO
FCI	future communication infrastructure
FIS	international civil aviation organization
ICAO	aeronautical administrative communication
IPS	Internet protocol suite
IPS DS	IPS dialogue service
ISO	international standards organization
MPLS	multiprotocol label switching
NextGen	next generation air system
OS	operating system
OSI	open systems interconnection
PORT	pilot operational response time
RAN	radio access network
RASP	reliable aeronautical services protocol
RCMP	required communication monitored performance
RCP	required communication performance
RCTP	required communication technical performance
SANDRA	seamless aeronautical networking through the integration of data links, radios, and antenna
SESAR	single European sky for ATM research
TCP	transmission control protocol
UDP	user datagram protocol
VDL	VHF data link

Symbol

<i>cwnd</i>	sender congestion window
<i>MTU</i>	maximum transmission unit
<i>RTO</i>	retransmission timeout
<i>RTT</i>	round-trip time
<i>rwnd</i>	receiver window
<i>SMSS</i>	sender maximum segment size
<i>ssthresh</i>	slow start threshold
$TT_{95\%}$	RCP nominal transaction time

1.0 Introduction

In the early 1930s, the air traffic control (ATC) and pilots started to communicate mainly using analog voice radio transmissions. The air traffic growth increased the requirements on information exchange between pilots and controllers, which has led to congestion of voice channels. Hence, in 1970s, the aeronautical radio incorporated (ARINC) developed the first data link system to provide connectivity services between commercial aircraft and airlines. Basic flight information from en-route aircraft started

to be exchanged using the aircraft communications addressing and reporting system (ACARS). Due to the system operational utility, ATC began using ACARS for flight clearances [1]. In 1983, ICAO created the future air navigation system (FANS) special committee, which is responsible for developing the future air traffic management (ATM) operational concepts, by exploiting the digital technology availability. In the 1990s, Boeing and Airbus developed their FANS systems, known as FANS-1 and FANS-A, respectively. Eventually, these two systems were combined to create FANS-1/A [2]. Meanwhile, the data link communication system has been enhanced by substituting ACARS with a more effective VHF data link, known as VDL Mode 2. This later one led into a broader use of the aeronautical data link communication by ATC via controller-pilot data link communications (CPDLC) applications. This CPDLC is a well-known data link application for exchanging instructions and information between air traffic controllers and pilots. This data link communication system is actually included in a global communication infrastructure called aeronautical telecommunication network (ATN).

Since their deployment, the existing ATN and air-to-ground communication links are based on OSI protocols, which was developed by the International Standards Organization (ISO) in the early 1990s. Therefore, this ATN/OSI technology uses packet switching concepts to transport information [3]. From 2010, the International Civil Aviation Organization (ICAO) has started to encourage the migration to ATN over IP protocol suite [4]. This upcoming migration aims to ensure the better interoperability between ATN and ground networks, which are usually based on the IP protocol. The second edition of Document 9896, published by ICAO in 2015, provides the technical requirements for applications, transport and network protocols supporting ATN/IPS [5].

The main purpose of the current paper is to provide a preliminary overview of the global approach to end-to-end reliability mechanisms that meets the heterogeneous requirements of the legacy, FANS and ACARS existing applications, and future ATN/IPS applications. A study based on the existing ATN/OSI applications can determine the legacy application reliability requirements particularly in terms of required communication performance (RCP) metrics [6]. Moreover, it is also necessary to consider the eventual needs of future applications.

This research work identifies the requirements through performance analysis of connection-oriented transport class 4 (COTP4) and transmission control protocol (TCP) New Reno version for legacy and future ATN/IPS applications, respectively. COTP4 is the basic transport protocol for ATN/OSI. TCP is the baseline for reliable data transfer in the Internet. In this study, the performance assessment of these transport protocols is based on the simulation results from an ATN model specifically developed on OMNET++ software. This aeronautical network model includes traffic generators to represent CPDLC and future applications. The CPDLC traffic is based on a statistical analyses of log files provided by a French area control centre (ACC). COTP4 and TCP New Reno are modeled in the transport layer to transmit the generated data between ATN end systems. The VDL Mode 2 link delay of previously developed model is used to illustrate the typical behaviour of the ATN air-to-ground subnetworks.

2.0 Literature review

The communication standard on ATN, known as ATN/OSI, is based on the seven-layer OSI model. Hence, according to the Doc 9880 standard [4], the ATN communicating entities implement the seven layers of the model. This specificity is at the origin of great complexity in network management, particularly in terms of Internet network interoperability. Therefore, since 2010, ICAO proposed a migration of the ATN to the Internet protocol suite (IPS) [4]. This migration intends to address communication, navigation and surveillance/air traffic management (CNS/ATM) capacity issues while increasing aviation safety. To achieve these objectives, Eurocontrol for Europe and Federal Aviation Administration (FAA) for US supported various ongoing research programmes for data link communication, as respectively in the case of the single European sky for ATM research (SESAR) and next generation air system (NextGen) projects [7, 8].

This section presents the operating principles of ATN/OSI, ATN/IPS and our research contributions.

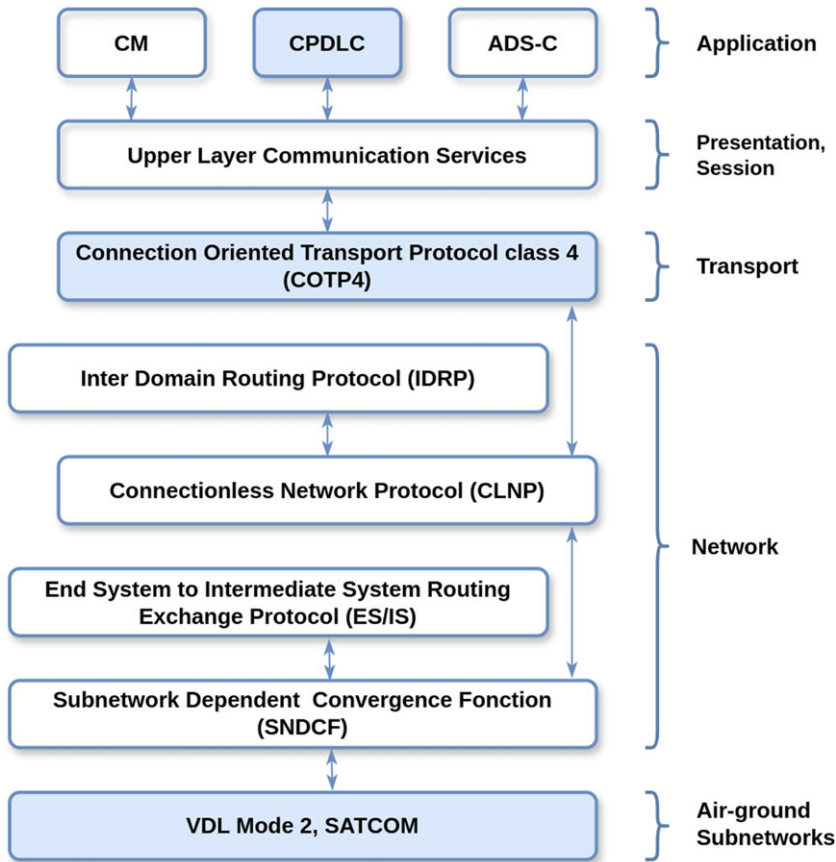


Figure 1. ATN/OSI protocol stack.

2.1 ATN/OSI principle

To provide a reliable end-to-end communications service, ATN/OSI mainly involves air-to-ground and ground-to-ground subnetworks taxonomy similar to radio access network (RAN) and core network (CN) in mobile networks. The air-to-ground subnetwork is defined by the radio access subnetworks including HF and VHF (VDL Mode 2) bands to provide coverage in continental airspace and also satellite, known as SATCOM in oceanic airspace. The ground-to-ground subnetwork is represented by the ATN core network, which interconnects the air-to-ground subnetworks to the ground end systems. Figure 1 describes the ATN/OSI protocol stack [4].

The air traffic service (ATS) and airline operational control (AOC) are the two primary safety-related traffic categories in the ATN. The ATS traffic relates to air traffic control, aeronautical and meteorological information, position sharing, etc. The AOC concerns departure, flight planning and other control on aircraft. The ATN also handles other non-safety-related types of traffic such as aeronautical administrative communications (AACs) and aeronautical passenger communications (APCs) [9].

2.1.1 Illustration of ATN/OSI applications

The ATN applications are categorised into two types including air-to-ground [10] and ground-to-board applications [11, 12]. The air-to-ground applications include the context management (CM), which establishes a link between two entities, CPDLC, automatic dependent surveillance (ADS), flight information service (FIS), and so on. For ground-to-ground category, there are ATS interfacility data communication (AIDC) and ATS message handling service (AMHS).

COTP4 protocol is the basic transport protocol for ATN/OSI. It operates over a connectionless network protocol (CLNP). The link and physical layer protocols are defined according to the physical links concerned [13].

2.1.2 COTP4 transport protocol

The COTP4 main purpose is to provide a reliable transport service by detecting the presence of bottlenecks or congestion likely to limit overall performance, and by managing time constraints during packet transmission. COTP4, designed to be deployed at the transport layer, can operate with connection-oriented network services (CONS) and connectionless network services (CLNS). In CLNS mode, COTP4 offers various features including flow control between peer transport entities, detection and recovery from packet loss, duplication, corruption and out-of-sequence delivery via segment and acknowledgment numbering. Furthermore, COTP4 relies on the retransmission timeout mechanisms to detect packet loss and uses the Go-Back-N protocol when retransmitting packets. For air-ground sub-networks, COTP4 protocol implementations must support configurable values for protocol parameters and timers [3].

Congestion on network nodes can cause delays in data transmission and result in data loss. Therefore, to provide end-to-end reliability, the congestion control is one of the key features of transport protocols. In ATN/OSI, the router adds an explicit congestion notification (ECN) to the packet when experiencing congestion. The COTP4 receiving entities rely on ECN to detect congestion. The sending rate, known as advertised window size, is defined and updated by the receiving entity. This advertised window size determines the amount of segments a sender can transmit before waiting for acknowledgment. It is initially set to 1 and updated when the total number of segment received corresponds to the possible advertisement of a new window size. It should be incremented by adding 1, if the number of segments received with ECN is less than 50% of the total number of the segments received. Otherwise, the advertised window size is reduced by multiplying it with a window decreasing factor that ranges from 0.75 to 0.95 [3]. Then, the receiving entity sends the updated window size in the acknowledgment to the sending entity. The advertised window size is used to send the subsequent segments. It is worth to note that the choices made in COTP4 are the opposite of those made in the Internet with the TCP protocol: explicit congestion notification versus implicit congestion detection, determination of the transmission window by the receiver versus calculation of this window by the transmitter.

2.2 Operating principle of ATN/IPS

The ATN/IPS aims to provide a reliable and effective network infrastructure for safety aeronautical communications, specifically ATS and AOC communications. It is currently under development, and it is expected to be the next generation of aeronautical network, enabling the interconnection of all concerned entities. The first edition of the standard defined by ICAO [4], presents the specifications of a system using Internet protocols. The second edition [5] specifies the system for legacy applications and states transport layer requirements when using TCP or user datagram protocol (UDP).

The future communication infrastructure (FCI) [14] for the ATN/IPS will include different sub-networks. The aircraft will be classified as a mobile subnetwork hosting airborne end-systems (A-E), an airborne router (A-R) for the link selection and management of inter-technology handovers, as well as airborne radios (AR) facilitating communication via air-to-ground links. The access networks will be the subnetwork that provides the air-to-ground connection between the aircraft and the ground systems. It includes radio communication links to reach the aircraft (VDL Mode 2, SATCOM, LDACS, . . .), the access ground router (AC-R), which is the last hop router for the aircraft, and the air-to-ground router (A/G-R) to connect the access network to the ground ATN/IPS internetwork. The applicative service providers such as ATC centre, air navigation service provider (ANSP) and AOC are considered as ground end-systems. As depicted in Fig. 2, the ground ATN/IPS internetwork is the core network that keeps all of the ATN/IPS subnetworks connected, and potential future air-to-ground subnetworks are also considered.

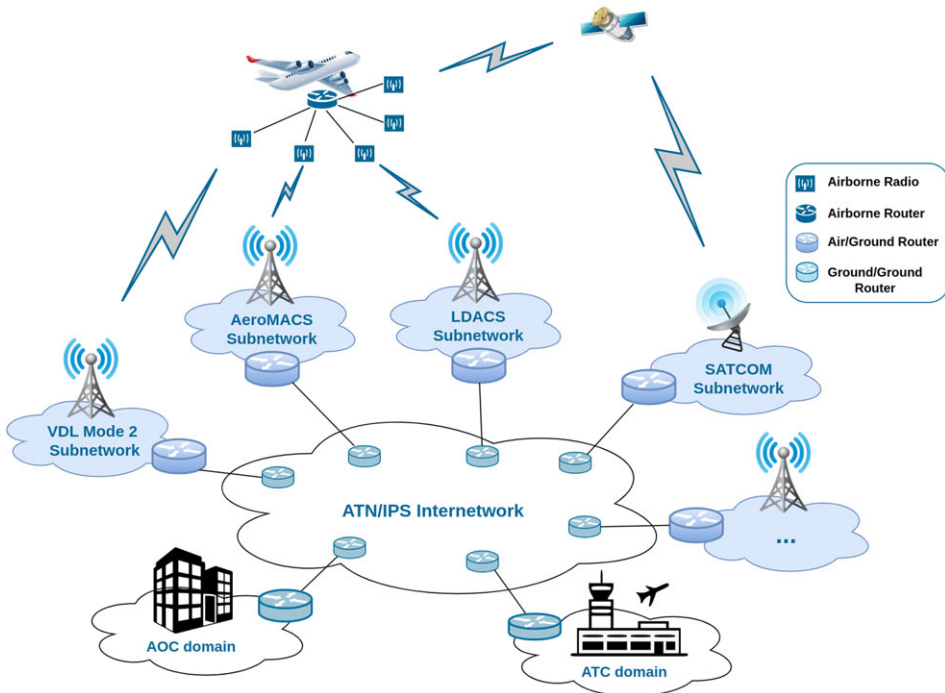


Figure 2. Future communication infrastructure.

2.2.1 ATN/IPS transport layer requirements

Currently in the standardisation phase, the ICAO 9896 document [5] specifies some considerations for the IPS architecture. It includes the connection-oriented TCP per RFC 793 [15] and RFC 1323 [16], and the connectionless UDP per RFC 768 [17] as transport protocols. TCP ensures reliability, whereas UDP is connectionless and lightweight. The primary benefit of TCP over UDP is that it guarantees the delivery of all data. However, TCP, on the other hand, does not offer any control over delivery time delays. The primary drawback of TCP compared to UDP is overhead. The UDP header is 8-bytes long and for TCP the header is between 20 to 60B.

The ICAO document [5] also requires the use of IPS Dialogue Service (IPS DS) on top of the transport protocols for the existing applications in ATN/OSI, as shown in Fig. 3. The IPS DS is an IP adaptation layer which helps the legacy applications to use TCP and UDP without any necessary modifications. However, the IP DS is not required for native IP applications to operate with TCP and UDP. The general IP internetworking for ATN/IPS is based on IPv6 per RFC 2460 [18].

2.2.2 TCP congestion control mechanisms

TCP is the second transport protocol proposed in 1981 [15]. TCP is one of basic protocols of the Internet protocol suite to provide reliable communications in connection-oriented mode. Since its release, different TCP variants were introduced by aiming to improve the congestion control mechanism. This later is one of the most important transport protocol roles. These TCP variants include RENO, New Reno, CUBIC, BBR and etc [19]. What these transport protocols have in common is the use of the Karn and Jacobson algorithms, introduced in RFC 6298, to compute the retransmission delay. The New Reno version enhances the TCP congestion control mechanism presented in RFC 5681 by improving its fast recovery algorithm, as described in RFC 6582. Moreover, CUBIC [20] and QUIC [21] protocols can be perceived as TCP enhancements using congestion control algorithm based on New Reno. Due to its

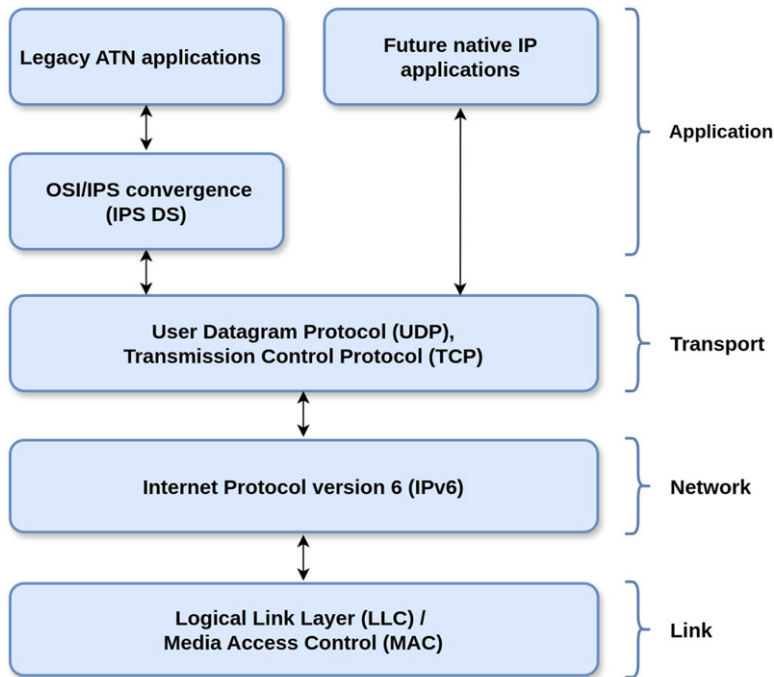


Figure 3. ATN/IPS protocol stack.

wide deployment and representativeness of how congestion control should operate, this work focuses on the evaluation of the TCP New Reno congestion control algorithm for ATN/IPS.

TCP New Reno is one of loss-based transport protocols using additive increase and multiplicative decrease (AIMD) algorithm for the congestion control [22]. This AIMD algorithm main objective is to either increase or decrease the flow sending rate based on events occurring on the sender entity. TCP New Reno is well known to represent how a congestion control mechanism should work. Therefore, in this study, TCP New Reno is considered for the future ATN/IPS applications. TCP New Reno has three different key features to manage congestion, such as slow start, congestion avoidance, and fast recovery. In this protocol, the sender defines its congestion window ($cwnd$) before sending the numbered segments. The $cwnd$ is initially set based on the sender maximum segment size (SMSS) value. For instance, if SMSS is less than or equal to 1095B, the initial $cwnd$ size should be set to $4 \cdot SMSS$ [23]. In the other side, the receiver indicates its window ($rwnd$) size while establishing the transport connection. Then, the $inflight$ size is defined by the minimum value between $cwnd$ and $rwnd$. The sender entity uses this $inflight$ size to limit the amount of segment can be sent before receiving an acknowledgment (ACK). During the slow start and congestion avoidance phases, the sender analyses the acknowledgments of the previously sent data to detect the network load.

During the slow start phase, the sender increases exponentially its $cwnd$ size when a new data ACK is received. The congestion avoidance phase starts when the $cwnd$ size reaches the slow start threshold ($ssthresh$). In this phase, the $cwnd$ size continues to increase linearly. The three duplicated ACKs receipt is considered as a sign of packet loss which starts the fast recovery phase. In this phase, the oldest in-flight segment is retransmitted and the $ssthresh$ is updated to half of the $inflight$ size. It remains in this phase until the acknowledgment of all segments in flight from the same window size. When this phase ends, $cwnd$ is multiplicatively reduced and the sender enters the congestion avoidance phase. A retransmission timeout (RTO) is considered as a link failure, which updates the $ssthresh$ to half of the $inflight$ size, and resets the $cwnd$ and $inflight$ size to 1 SMSS. Then, the slow start phase increases the $cwnd$ size up to $ssthresh$, as shown in Fig. 4.

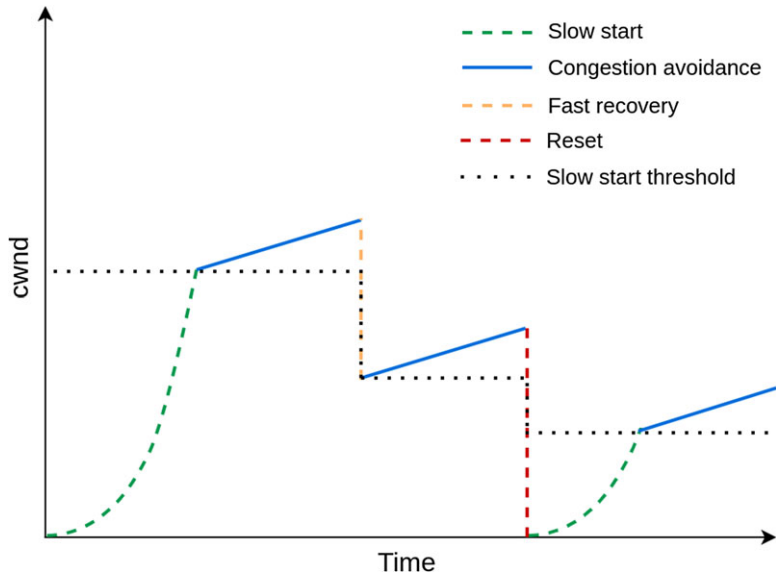


Figure 4. Evolution of cwnd size in TCP New Reno.

2.3 Related work

Air-to-ground communication technology is crucial to the end-to-end air traffic management concept, which necessitates the global integration of existing, future and emerging communication networks. Different research are motivated by the need to develop new data communication protocols and services that meet the safety and performance standards of ATN/IPS. Some of the existing work [24] focuses on safety-critical communications, such as controller-pilot communication for directing aircraft in controlled airspace and aircraft separation. A simulation-based performance assessment of TCP (RFC 793) transport protocol, for data communication in ATN/IPS is conducted in Ref. [25]. The authors categorised the traffic pattern of aeronautical applications to short and some larger messages of bulk data based on ‘communications operating concept and requirements for the future radio system’ (COCR) [26]. TCP performance degrades to a simple stop-and-wait protocol for short messages. On the other hand, for larger messages, considerable overheads and intolerable inefficiencies in terms of performance were observed. Under seamless aeronautical networking through the integration of data links, radios, and antennas (SANDRA) project, message-based reliable transport protocol, namely, reliable aeronautical services protocol (RASP) was proposed in Ref. [27]. This protocol aims to handle safety-related aeronautical communications. RASP is a connectionless protocol that does not provide congestion and flow control. This protocol is designed to be implemented on top of UDP. A simulation-based performance analysis has shown that RASP has lower end-to-end delay average and requires less retransmissions compared to TCP New Reno. However, the lack of flow and congestion control will not be suitable for future ATN applications that will generate larger data flows with low inter-arrival time, as we demonstrate in the current study.

3.0 Methodology

In order to identify the requirements for transport protocol in ATN/IPS a first step is to analyse the performances of regular protocols, i.e. COTP4 and TCP. The identified weakness will help define the requirements for the new IPS protocol stack. This research work presents a simulation-based performance analysis of COTP4 and TCP transport protocols. To achieve this objective, we have developed an ATN network model incorporating different parameters. This section describes the ATN modeling and the performance requirement for CPDLC data link application.

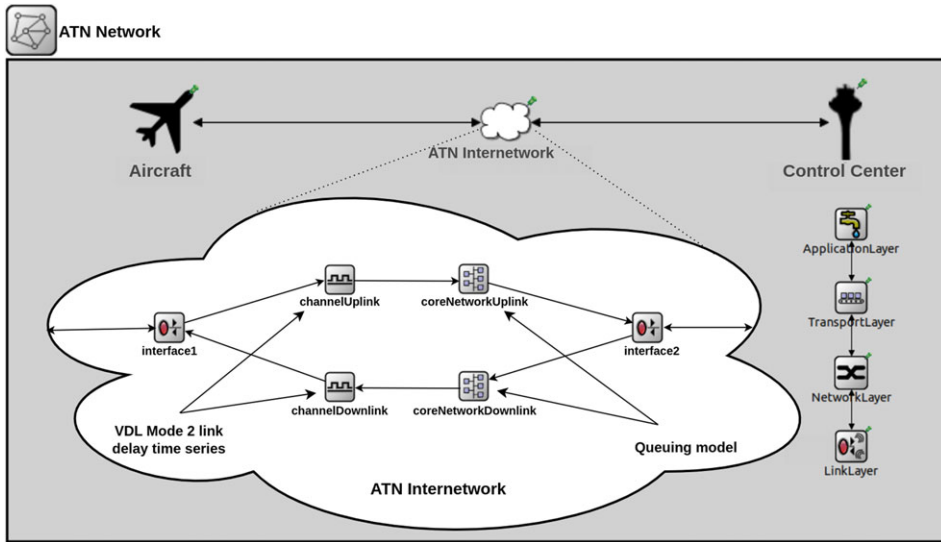


Figure 5. ATN model description.

3.1 ATN modeling description

The modeled ATN network consists of the core network, air-to-ground link, and the end systems representing the aircraft and the control centre as shown in Fig. 5. The end system encompasses different layers such as application, transport, network and link layer. The application layer is modeled to generate traffic. As introduced in Section 2, COTP4 is the basic transport protocol for ATN/OSI, and TCP is the considered transport protocol options for ATN/IPS. Therefore, in the transport layer, COTP4 and TCP New Reno are modeled. The network layer transmits packets and controls the information, as ECN, from the core network to the upper layers. Then, the link layer operates as an interface which transmits frames from/to air-to-ground links.

The air-to-ground link ensures the data communication between aircraft and ground station. As this research work focuses on continental airspace, VDL Mode 2 link is considered based on air-to-ground delay time series. The VDL Mode 2 link delay is obtained from a model developed by ENAC and ISAE/SUPAERO. It contains the end-to-end transaction time delay measured between sender and receiver application layers. This provides the ability to consider the resource sharing effects between the aircraft and the ground base station. VDL Mode 2 link model is characterised by the aviation VHF link control (AVLC) sublayer and carrier sense multiple access (CSMA) p-persistent access mechanism. The AVLC protocol is introduced to establish a logical link between air-to-ground end systems. CSMA is used to limit collisions of data transmissions between the aircraft and ground stations. Therefore, the collected delay time series include the waiting access time induced by CSMA p-persistent, as well as the frame retransmission possibility due to collisions or losses. Figures 6 and 7 display the transaction time delay in the uplink and downlink, respectively.

The ATN core network can experience congestion periods, which are produced by the amount of data flows on the output buffer either of a router or a switch. The congestion is modeled using queuing model as indicated in Fig. 13. During the congestion periods, background traffics are generated, which represent an average of 85% of the output link capacity. As result, transmission delay and random loss rate may occur, based on the bandwidth buffer capacity. Usually, with a conventional IP network, the buffer size is defined by multiplying the targeted end-to-end round-trip time (RTT) by the output link capacity. In this work, the buffer size is set to 200kB, with a targeted RTT of less than 2s and link capacity of 1Mbit/s. Since ATN networks are primarily based on MPLS label switching, a larger buffer size results in a higher RTT target, which enables networks to achieve an extremely low packet loss rate.

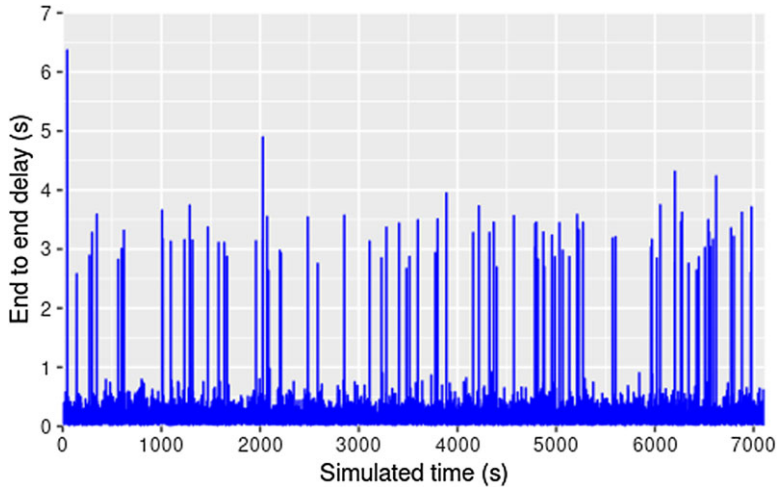


Figure 6. VDL Mode 2 uplink delay time series.

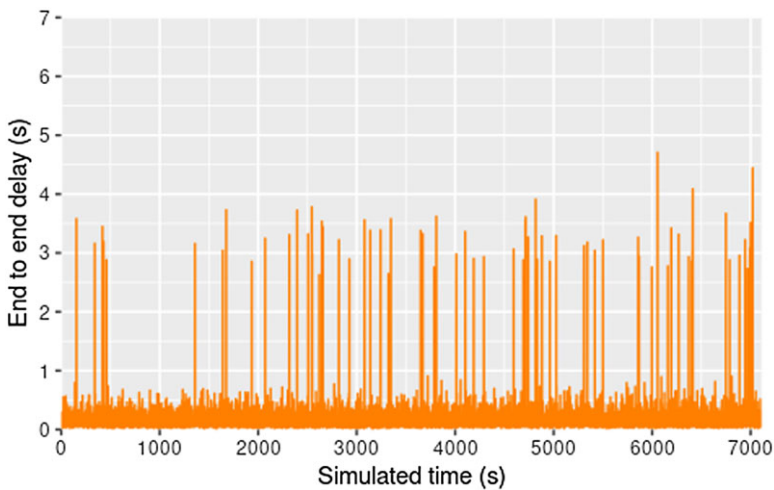


Figure 7. VDL Mode 2 downlink delay time series.

3.2 RCP requirements for CPDLC

The end-to-end requirements for RCP specifications are allocated to various delay factors such as initiator time, required communication technical performance (RCTP), pilot operational response time (PORT) and required communication monitored performance (RCMP). Figure 8, adapted from Ref. [6], describes a performance definition for a controller-initiated communication, which ends when the controller receives the response message. The overall round-trip time from the beginning to the end of a communication represents the transaction time, which includes the initiator time and the RCMP.

The initiator delay is the amount of time required for an initiator to review an outgoing and to recognise the corresponding incoming response message. The RCMP is the delay between the timestamp of the outgoing message and receipt of its response. It encompasses PORT and RCTP, as shown in Fig. 8. PORT is the time that the pilot needs to process and respond to a message. RCTP represents the round-trip time from when a controller sends a message to when it receives a corresponding response, excluding initiator time and PORT.

Table 1. Performance requirements for CPDLC [6]

RCP type	Parameters	Transaction time (Sec)	RCMP (Sec)	RCTP (Sec)
RCP 130	<i>ET</i>	130	120	32
	<i>TT</i> _{95%}	67	60	20
RCP 240	<i>ET</i>	240	210	150
	<i>TT</i> _{95%}	210	180	120
RCP 400	<i>ET</i>	400	370	310
	<i>TT</i> _{95%}	350	320	260

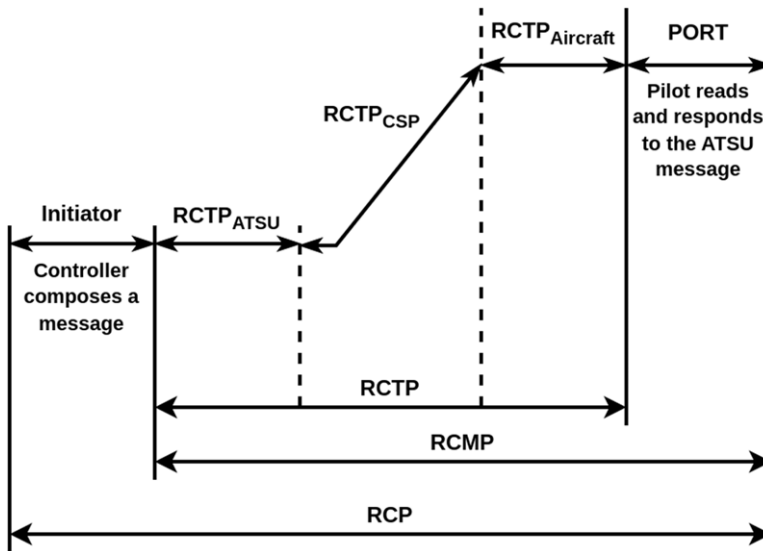


Figure 8. Performance allocation for controller-initiated communications.

Table 1 summarises the performance and time allocations of different RCP types for CPDLC requirements. The RCP expiration time (*ET*) refers to the maximum time for the completion of the transaction after which the initiator is required to revert to an alternative procedure. The RCP nominal time (*TT*_{95%}) corresponds to the maximum nominal time within which 95% of transactions is required to be completed. The choice of RCP depends on the airspace environmental characteristics. According to Ref. [6], RCP 240 is eligible for oceanic area and RCP 130 for continental area with high traffic airspace.

3.3 Previous results

The resulting problem, which is the subject of our case study, concerns the end-to-end reliability of applications with heterogeneous requirements on ATN/IPS data link communication. In order to achieve this, we started by analysing the performance of COTP4 of ATN/OSI and TCP New Reno protocol in Ref. [28]. In that work, we considered the legacy application, CPDLC and a generic file generator as a future ATN/IPS application.

4.0 Simulation results

4.1 Simulation parameters

The main objective of the performed ATN simulation is to identify the transport protocol mechanisms required by the legacy and the future ATN applications. To achieve this objective, the COTP4 and TCP

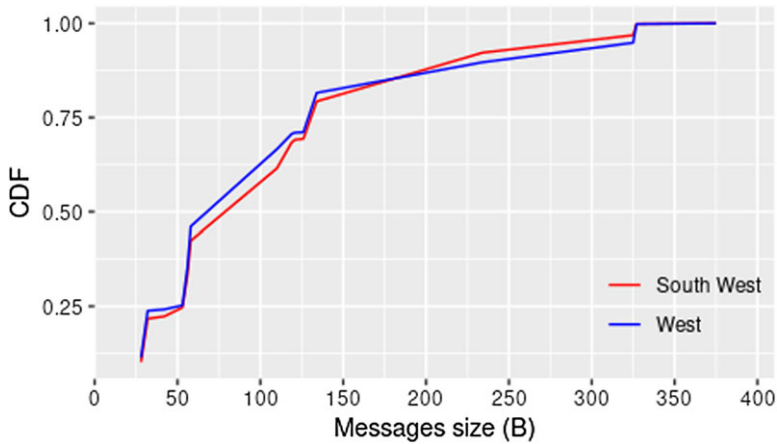


Figure 9. CDF of the messages size from South West and West ACC of France.

New Reno performance comparison is conducted in terms of end-to-end application delay, i.e. the time required for data to be transmitted from source to sink. As traffic source, two different applications were modeled, such as CPDLC as legacy application and a file generator as future heavier application.

The recent study [28] demonstrated that CPDLC does not benefit from New Reno congestion control features because of its low frequency of message generation. The performance analysis is based on RCP 130, which is the most constraining requirement, with transaction time (TT95%) of 10s. As this RCP requirement is applied in terms of transaction time or end-to-end delay (TT/2), it has no restrictions in terms of generation time interval. Therefore, in this study, we set the message generation time interval to 10s. Then, in order to obtain higher end-to-end delay samples for COTP4 and TCP performance analysis, the packet flow is increased by using the block-wise mechanism in the CPDLC application. This technique consists in dividing the messages into smaller blocks in the sending entity, and the receiving entity reassembles the blocks to reconstitute the original message.

The maximum block size was defined based on the CDF of CPDLC messages size obtained from ACC West and South West of France. Figure 9 shows that, the largest CPDLC message size is 375B and the smallest is 25B. Furthermore, it demonstrates that 80% of the messages size are less than 200B. For the simulation, CPDLC generates messages with variable size between 25 to 375B; then, the maximum block size was set at 10 and 50B, respectively. A value of 10B divides all CPDLC messages into smaller blocks, and a value of 50B fragments only messages exceeding this size.

The considered second traffic source is modeled to generate files that may represent the future ATN/IPS application, for instance those that will be used to transmit up-to-date weather information map to pilots, in real time. The weather information includes air temperature, moisture and pressure, as well as wind speed and direction, etc. It can be updated periodically and also when unexpected weather conditions are encountered between the regularly schedule.

For the simulation, the file generation time interval and size are variable. In order to conduct an in-depth analysis of COTP4 and TCP New Reno performance, two different cases were considered. As the transport protocols performance is evaluated on the basis of end-to-end application delays, using the variation of time interval between files generation ranging from 1 to 10s increases the packet flow, resulting in higher end-to-end samples. Therefore, in the first case, the time between generated files has a random value based on a uniform distribution of 1–5s, which is referred to as high-rate file generation. In the second case, the uniform distribution is between 6 and 10s, which is regarded as medium-rate file generation. In both cases, each generated file has a random size uniformly distributed between 5kB minimum and 10kB maximum sizes.

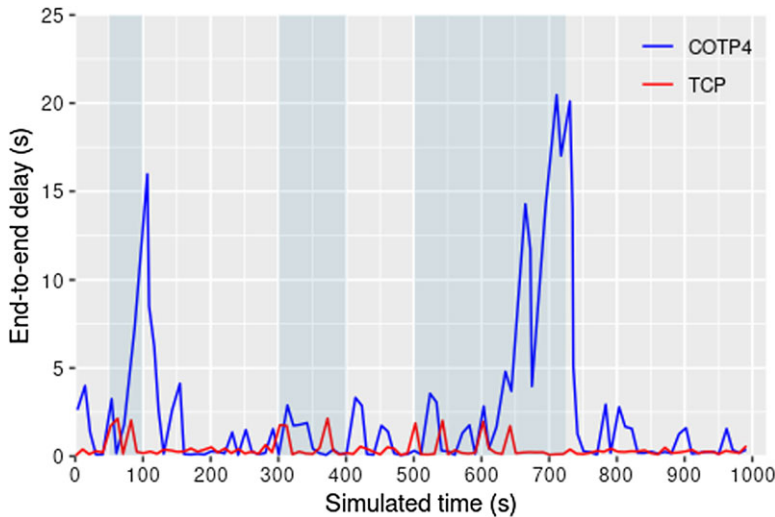


Figure 10. Messages end-to-end delay for block size of 10B.

For each scenario, the transport protocol can be COTP4 or TCP. The congestion periods time intervals are modeled with bandwidth-delay buffer size which results in random losses. In the ATN core network, three different congestion periods were introduced such as a short-period of 50s between 50 and 100s, 100s for the medium-period from 300 to 400s, and 225s for the long-period between 500 to 725s of the simulation time. In the simulation results, the congestion periods are highlighted by the blue background. Furthermore, each scenario considers a persistent connection that keeps the end systems connected during data transmission.

4.2 End-to-end delay analysis

4.2.1 Scenario 1: CPDLC message exchange from pilot to controller

This scenario focuses on pilot sending messages to controller with maximum block size of 10 and 50B. The SMSS of the transport protocol is set to 500B. Figures 10 and 11 present the message end-to-end delay with block size of 10 and 50B. With the block size of 10B, the data flow increases because each message is fragmented. Figure 10 indicates that the medium-period of lossy congestion has less effect on the delay for both COTP4 and TCP. However, during the short-period of lossy congestion, the end-to-end delay increases up to 17 and 21s in long-period congestion for COTP4. Contrarily to COTP4, for TCP, the lossy congestion periods does not highly affect the end-to-end delay which increases to around 2.3s. When the data flow decreases with the maximum block size of 50B, during the lossy short and medium congestion periods, the end-to-end delay slightly increases to 2 and 1.9s for COTP4 and TCP, respectively. During the long period of lossless congestion, the end-to-end delay for COTP4 increases up to around 5.2s, as shown in Fig. 11.

When the block size is set to 10B, we have the overhead ratio plotted in Fig. 12. It can be seen that the header is between 45% to 90% of the COTP4 segment size against 69% to 95% with TCP transport protocol. However, with 50B block size, the payload size is larger than the overhead for both COTP4 and TCP, as illustrated in Fig. 13. In this case, the header ratio decreases to 15% with COTP4 and 29% with TCP.

4.2.2 Scenario 2: File transmission from ground-to-board

According to the files size considered in this scenario, the SMSS is set to 1kB for the transport protocols. Figures 14 and 15 display the end-to-end delay of files transmission with COTP4 and TCP while using medium-rate file generator, respectively. The important data flow leads from packet loss to each

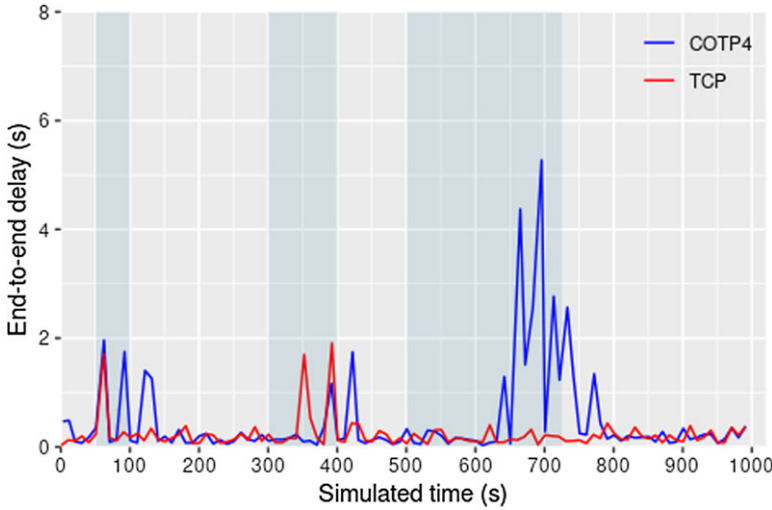


Figure 11. Messages end-to-end delay for 50B block size.

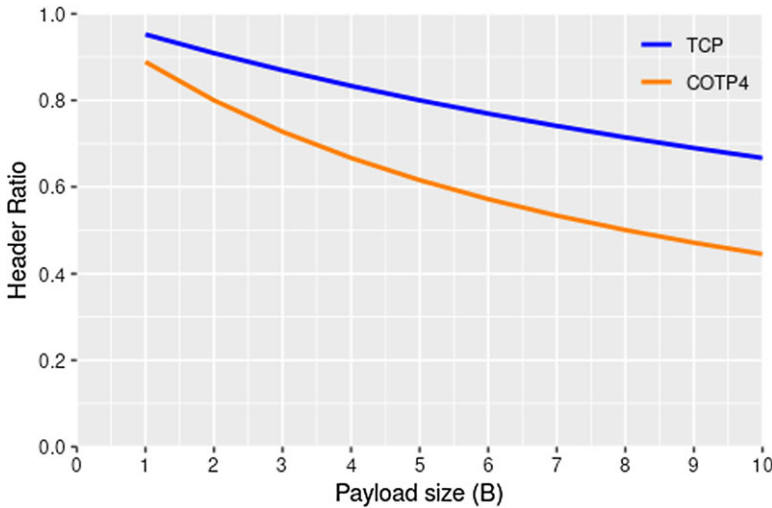


Figure 12. Overhead analysis for 10B block size.

congestion period in this scenario. As shown in Fig. 14, the end-to-end delay increases to around 6s when packet loss occurred during the short and medium congestion periods. Then, the long-period of lossy congestion highly affects the end-to-end delay, which increases up to around 125s. It keeps increasing even when the congestion period is over. For TCP, Fig. 15 demonstrates that, the end-to-end delay increases slightly during the short and medium lossy congestion periods. The delay increases to 10.8s during the long-period of lossy congestion.

With the high-rate file generator, the end-to-end delay increases continuously before, during and even when the congestion period is over with COTP4, as shown in Fig. 16. Contrarily to COTP4, with TCP, the short and the medium lossy congestion periods lightly affects the end-to-end delay, which increases to around 7s, as indicated in Fig. 17. Then, it increases up to 19.5s during the long-period lossy congestion. However, the delay decreases to 3s when the congestion period is over.

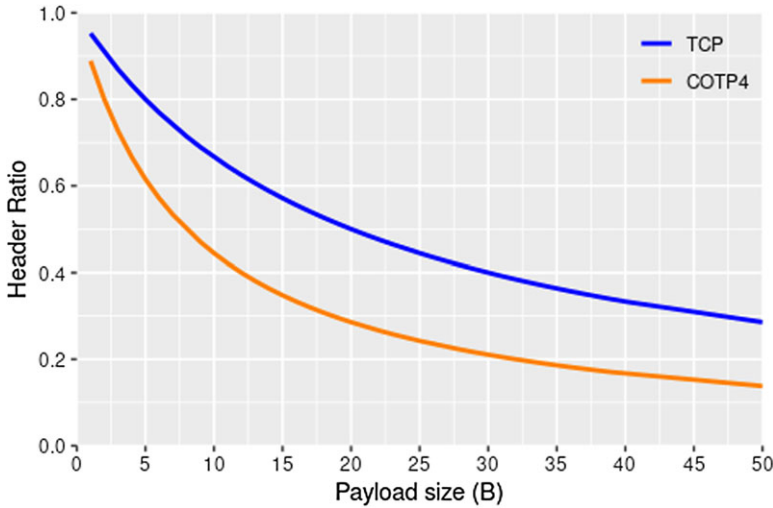


Figure 13. Overhead analysis for 50B block size.

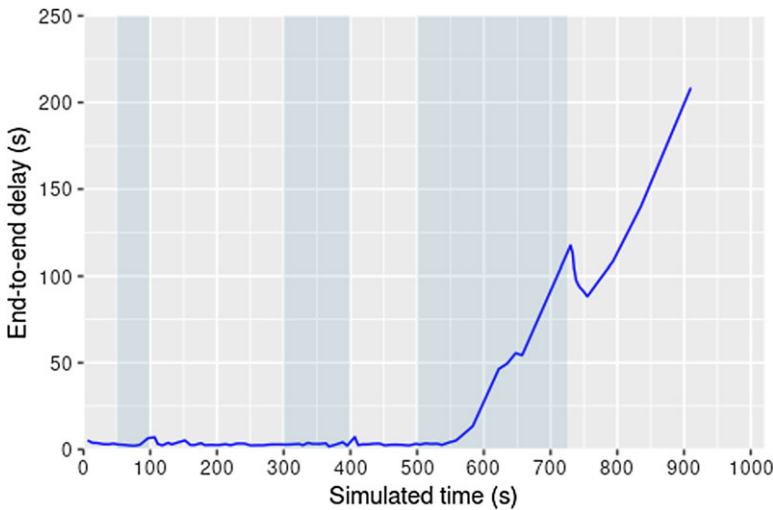


Figure 14. Files end-to-end delay with COTP4 for medium-rate generation.

For the case of files size between 5 and 10kB, and SMSS set to 1kB, we have the header ratio plotted in Fig. 18. It is noteworthy that the header ratio versus the payload size varies from 1 to 1kB. Then, it decreases from 90% to 5% with TCP against 47% to 1% for COTP4.

4.3 Performance analysis of COTP4 and TCP

4.3.1 Performance analysis for CPDLC messages exchange

Figures 19 and 20 show how COTP4 control the congestion when the maximum block size is set to 10 and 50B. During the congestion periods, the window size decreases because of the received packet amount with ECN in the receiving entity. A packet loss occurred at 52.14s of simulated time, as indicated in Fig. 19, the end-to-end delay increases to 3s. Because the window size corresponds to 8 segments and the in-flight packets are greater than 1 segment, the lost packet is retransmitted when new acknowledgment received. The retransmission timer T1 is rescheduled so there is no retransmission timeout.

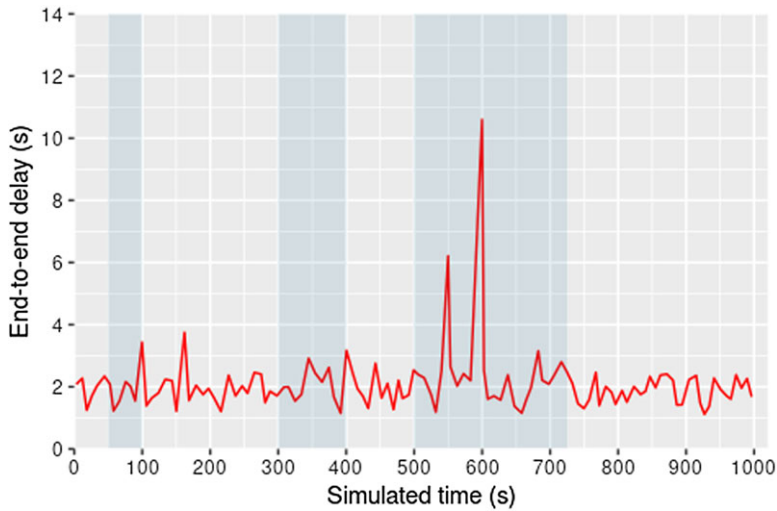


Figure 15. Files end-to-end delay with TCP for medium-rate generation.

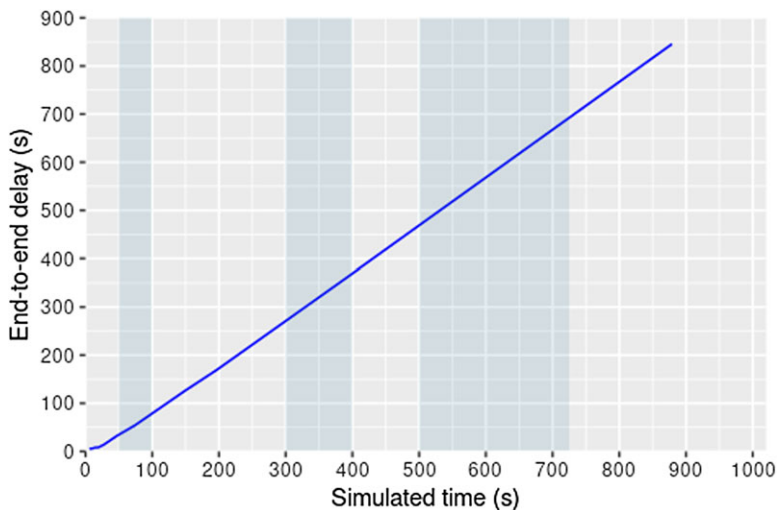


Figure 16. Files end-to-end delay with COTP4 for high-rate generation.

A second packet loss occurred at 81.66s, the window size is continuously decreasing and the lack of acknowledgment causes timer T1 to expire. Then, the window size is blocked into 1 segment until the end of the congestion period which prevents to benefit from the Go-Back-N mechanism in COTP4. This congestion increases promptly the end-to-end delay up to 17s. Furthermore, Fig. 20 demonstrates that during a lossless congestion period, the ECN receipt decreases the window size continuously. As result, the end-to-end delay increases because of window size blocked into 1 segment and not because of packet loss.

During congestion period analysis different cases were identified in Fig. 21 when using TCP as the transport protocol. The first case consists of delayed transmission at the start of the congestion period, which causes the RTO timer to expire at 51s before the packet reaches its destination, as seen in Fig. 21. Then, the end-to-end delay increases, as shown in Fig. 10, and the *ssthresh* decreases to half of the *inflight* size as stated in Fig. 21. The second case is that a packet loss occurred at 61.02s, the RTO

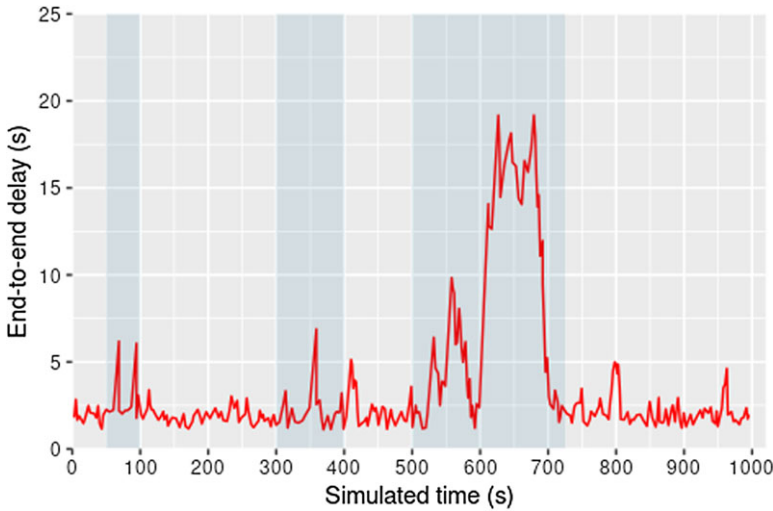


Figure 17. Files end-to-end delay with TCP for high-rate generation.

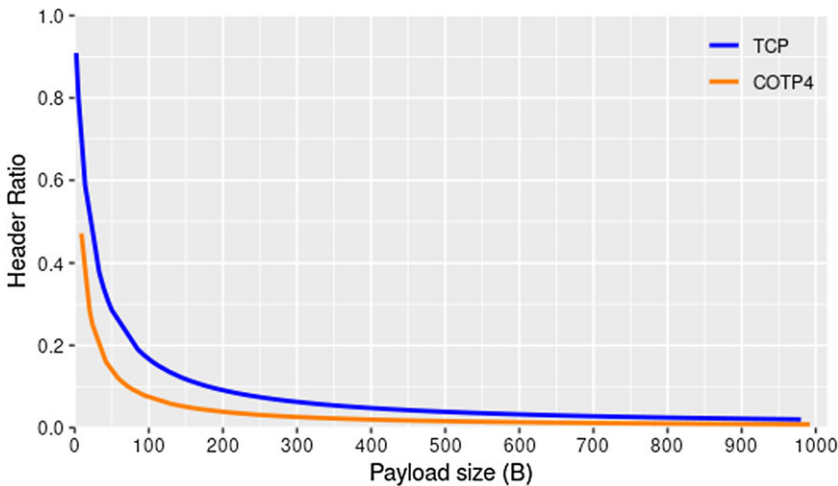


Figure 18. Overhead analysis with COTP4 and TCP packets.

timer expires at 62.12s before receiving any duplicate ACK, which increases the end-to-end delay. Last case, after receiving three duplicate ACKs, TCP enters the fast retransmit phase. However, the reception of duplicate ACKs and the packet retransmission during the fast retransmit phase do not satisfy the conditions for restarting the retransmission timer and leading to an RTO. Figure 22 also shows that the packet retransmission relies only on RTO expiration. In these different cases, TCP considers delayed transmission and isolated loss as link failures. This leads to a degradation of TCP performance, which is inline with the studies presented in Refs [25, 27].

In the continental airspace, with the RCTP of RCP 130 presented in Table 1, the required end-to-end delay corresponds to 10s (TT/2). The COTP4 performance respects the expected delay with the maximum block size of 50B but not with 10B. However, despite the overreaction during the congestion control, TCP New Reno respects the required end-to-end delay with both block size of 10 and 50B.

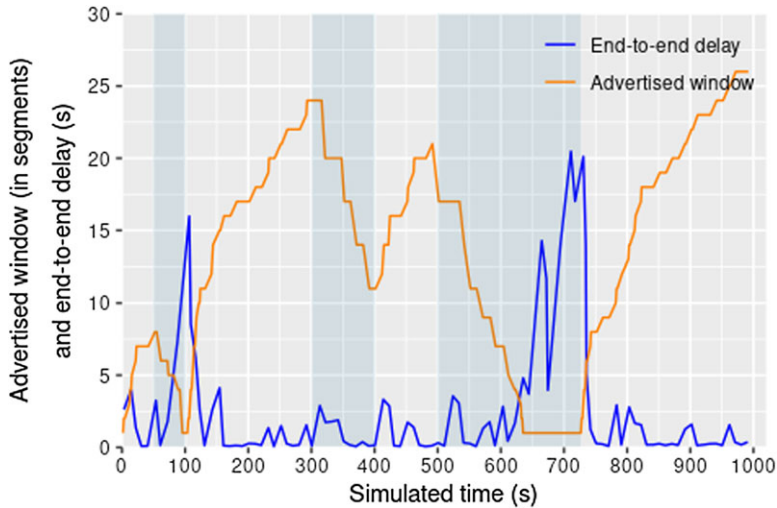


Figure 19. COTP4 congestion control analysis for 10B block size.

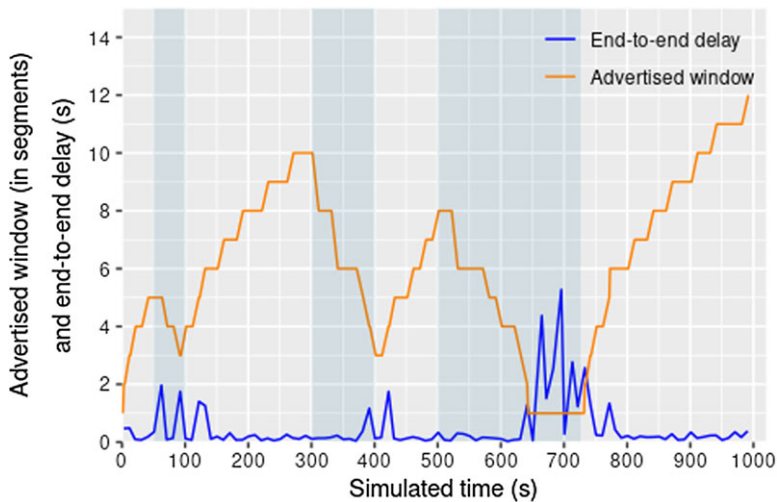


Figure 20. COTP4 congestion control analysis for 50B block size.

4.3.2 Performance analysis for files transmission

In terms of congestion control, Figs 23 and 24 illustrate the behaviour of COTP4 and TCP with medium-rate file generator. For COTP4, the advertised window size increases normally until the start of the short-period congestion, as shown in Fig. 23. Then, the window size decreases because of the receipt of ECN in the receiving entity, and when it reaches the value of 1 segment, the end-to-end delay increases. With the medium-period of lossy congestion, the end-to-end delay is slightly affected because the advertised window size is still large enough for COTP4 to benefit from the Go-Back-N mechanism. However, when the window size is stalled into 1 segment, the end-to-end is dramatically affected, as displayed in Fig. 14. Figure 24 demonstrates the efficiency of TCP New Reno congestion control mechanism while operating with medium-rate file generator compared to COTP4. When isolated loss occurred during the short and the medium congestion periods, the receipt of duplicated ACK decreases the *ssthresh* to half of the *inflight* size. Then, when the sender receives three duplicated ACKs, it enters the fast recovery phase. The *ssthresh* is updated to half of the *inflight* size and the *cwnd* is set to $ssthresh + 3$ SMSS.

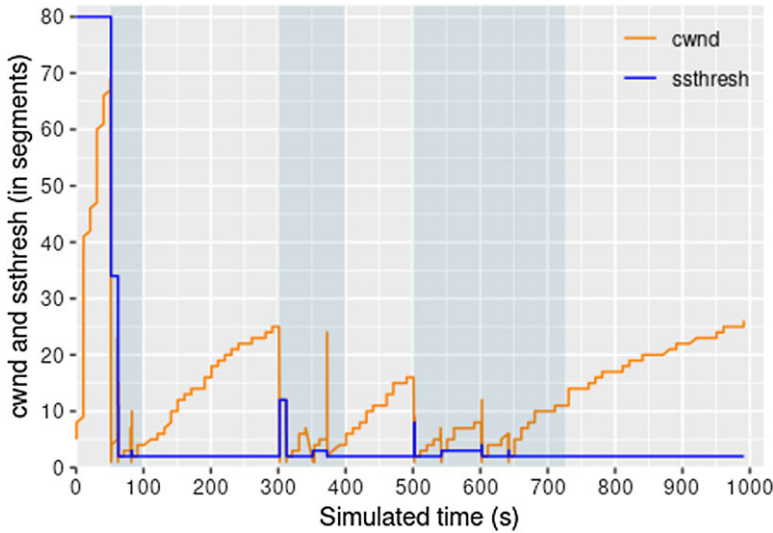


Figure 21. TCP congestion control analysis for 10B block size.

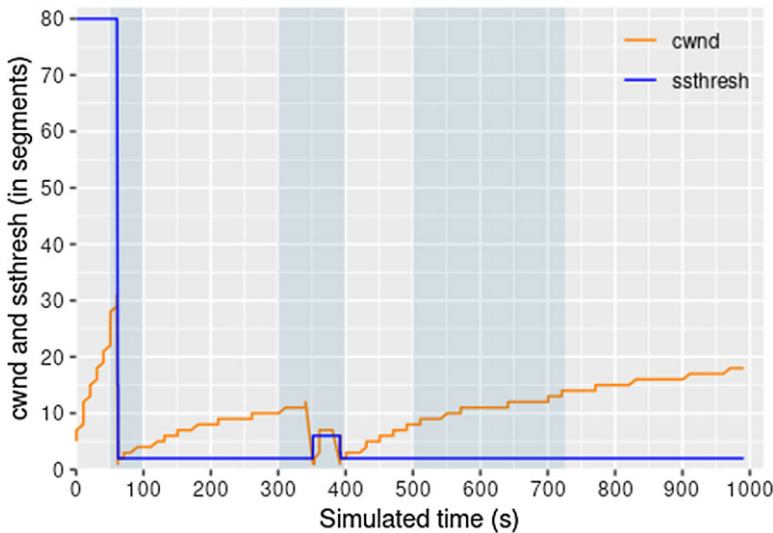


Figure 22. TCP congestion control analysis for 50B block size.

As result, the end-to-end is lightly affected by these packet losses. However, during the long-period of lossy congestion, TCP experiences three RTOs timeouts at 551, 592, and at 600s of the simulated time followed by the slow start phases. These RTOs timeouts are caused by the loss of two packets belonging to the same *cwnd*. The packets are retransmitted during the fast recovery phase; nevertheless, the receipt of duplicated ACKs and this retransmission type do not restart the retransmission timer. As result, RTO expires before new packet is acknowledged. The end-to-end delay increases for the second packet loss due to the time required to receive its duplicated ACK. This is normal behaviour for TCP New Reno.

Figure 25 shows an abnormal congestion control of COTP4 with high-rate generator. The advertised window size is rarely updated compared to the supposed flow of files generated. The window size increases normally before the short-period lossy congestion starts. During the congestion periods, the window size is blocked to one value for a while then it decreases or increases. It results the drastic

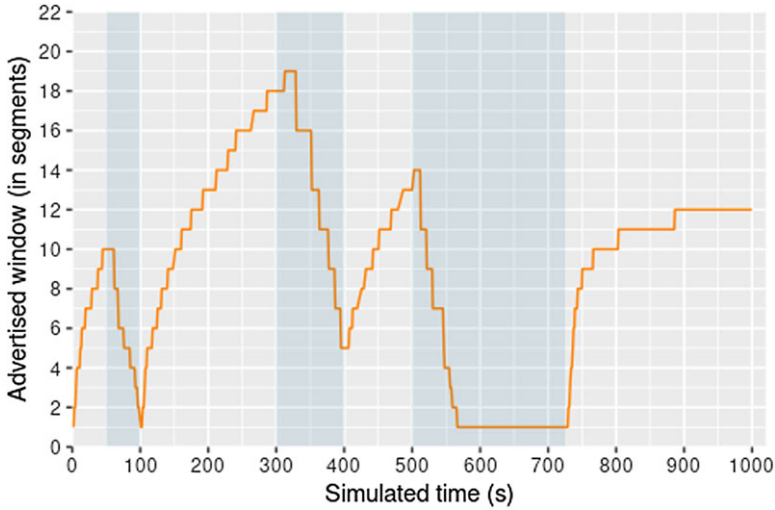


Figure 23. COTP4 congestion control analysis for medium-rate generator.

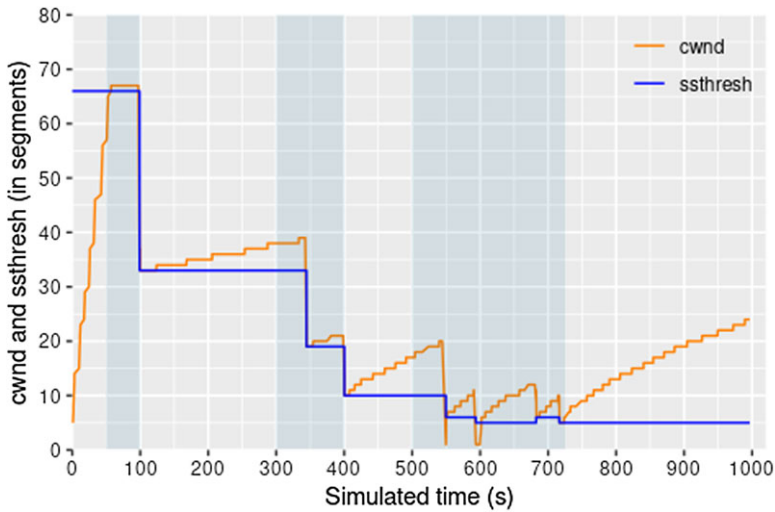


Figure 24. TCP control analysis for medium-rate generator.

increase of the end-to-end delay in Fig. 16. For TCP, similarly to the result with medium-rate generator, the congestion control in Fig. 26 displays its efficiency.

The packet losses during the short and medium congestion periods lightly affect the end-to-end performance because of the fast recovery feature. The RTO timeouts occurred because of two-packet loss from the same *cwnd*. This is handled in the exact identical way as the packet loss in medium-rate generator.

This files transmission analysis demonstrated that the congestion control mechanism used in COTP4 is unsuitable for applications transmitting flows of data. However, the TCP New Reno congestion control mechanisms can efficiently manage massive and high-rate data transmission for the future ATN/IPS applications. In contrast to the approach proposed in the RASP protocol [27], these results demonstrated the necessity of flow and congestion control in handling the data flows of future ATN applications.

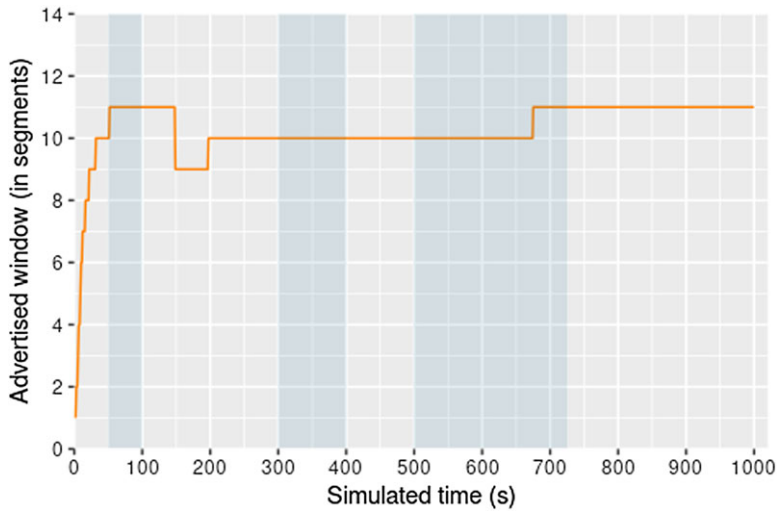


Figure 25. COTPA congestion control analysis for high-rate generator.

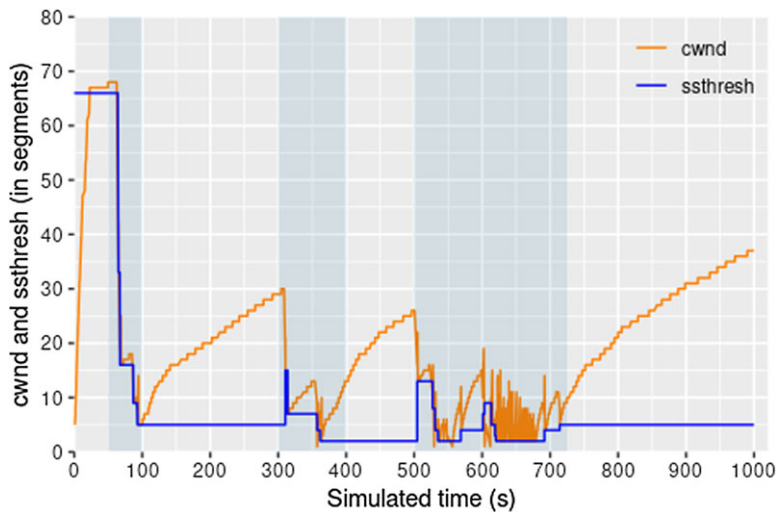


Figure 26. TCP control analysis for high-rate generator.

5.0 Concepts of the new transport protocol

Legacy and the future ATN applications have different properties and may have to meet different requirements. For instance, the performance analysis in Subsection 4.2 shows two application aspects that are:

- In the one hand, CPDLC requires retransmission time control or timer-oriented protocol due to its sporadic flow of messages and the RCP requirements. The use of block-wise allowing to increase the flow. Then, it was not suitable to address this retransmission time issue with TCP New Reno as transport protocol. Moreover, CPDLC could not benefit from the fast retransmit and fast recovery.
- In the other hand, the other applications that have higher data flow and larger size to transmit require the use of protocol with congestion control mechanism which considers the capacity and

adapted RTT estimation algorithm. As shown earlier, this issue can be handled by using TCP variant.

For this reason, we propose to adopt a protocol stack inspired by QUIC design. QUIC is a multiplexed, encrypted and low-latency transport protocol designed by Google to improve transport performance for HTTPS traffic [29]. QUIC was introduced in 2017 due to the rapid transition from insecure to secure Internet traffic, as well as the enormous growth in Google secure web traffic from 2012 to 2016, which causes delay. To achieve this low-latency objective, QUIC shares cryptographic and transport information with a single handshake. Once a client knows the initial keys of the server, new QUIC connections can be established with zero round trip time (0-RTT) data delay. Reducing latency and overheads caused by head-of-line blocking delay is also one of its objectives. To provide an end-to-end reliability service, QUIC allows a sender to unilaterally choose a different congestion control algorithm, although the New Reno algorithm introduced RFC 6582 [30] is the most widely used, as presented in RFC 9002 standard.

In contrast to TCP transport protocols, which are commonly implemented in the operating system (OS) kernel, QUIC does not require changes to the OS as it is designed to be implemented in the user space. It is therefore easier to deploy in a variety of applications and enables iterative changes during application update times. As the reliability mechanisms are provided at the application layer, QUIC uses UDP as transport protocol to take advantage of its low latency [31].

To benefit from the ease of deployment, we adopted the QUIC philosophy by adding the reliability mechanism on top of UDP. As mentioned earlier, QUIC was designed specifically for Internet traffic, which has a far higher rate than legacy ATN/OSI applications such CPDLC. Adopting QUIC as it stands will not be appropriate for data exchange in aeronautical communication field. Furthermore, these applications will not benefit from 0-RTT mechanism due to the differences in link characteristics between the Internet and the aeronautical air-ground subnetwork. The use of New Reno algorithm for legacy ATN/OSI applications would result into the same issues as those described in Section 4 for CPDLC. Nevertheless, New Reno provides adequate performance for future ATN/IPS applications. This entails the needs of adding different reliability mechanisms based on the targeted application.

Figure 27 shows the new protocol stack with a non-exhaustive list of the reliability mechanisms. The main property of this protocol is the relation between each application and a set of mechanisms that has to be used to meet the expected requirement. Table 2 depicts this relationship for the applications covered in our study. And certainly, such a way to adopt a relevant set of reliability mechanisms should be used for any new future ATN/IPS applications.

5.1 CPDLC required features

5.1.1 Loss detection and flow control

For CPDLC application, the sender attributes a sequence number for each outgoing segment. The receiving entity sends an acknowledgment for the received segment with the next expected segment number. In terms of flow control, a stop and wait mechanism is appropriate due to the low rate of messages exchange with CPDLC. This mechanism relies on RTO timer to detect UDP datagram loss. The RTO expiration requires a packet retransmission, with a maximum retry of three times. The segmentation and reassembling are necessary for packet with size greater than 500 B.

5.1.2 Congestion control mechanisms

CPDLC is a time-sensitive application and to respect the RCP requirements, the use of RTT-based congestion control algorithm would be efficient. Due to robust RTT estimation and RTO back off handling, Karn's algorithm [32] has prevented unnecessary retransmission from turning into link failure for decades for TCP. However, the RTO retransmission timer, which only uses measured RTTs when there are no retransmissions, as required by the Karn's algorithm, results in unnecessary retransmissions. The binary exponential back off, $RTT * 2$, used in TCP does not correctly handle unnecessary retransmissions when the RTT is greater than the RTO, as in the case with CPDLC.

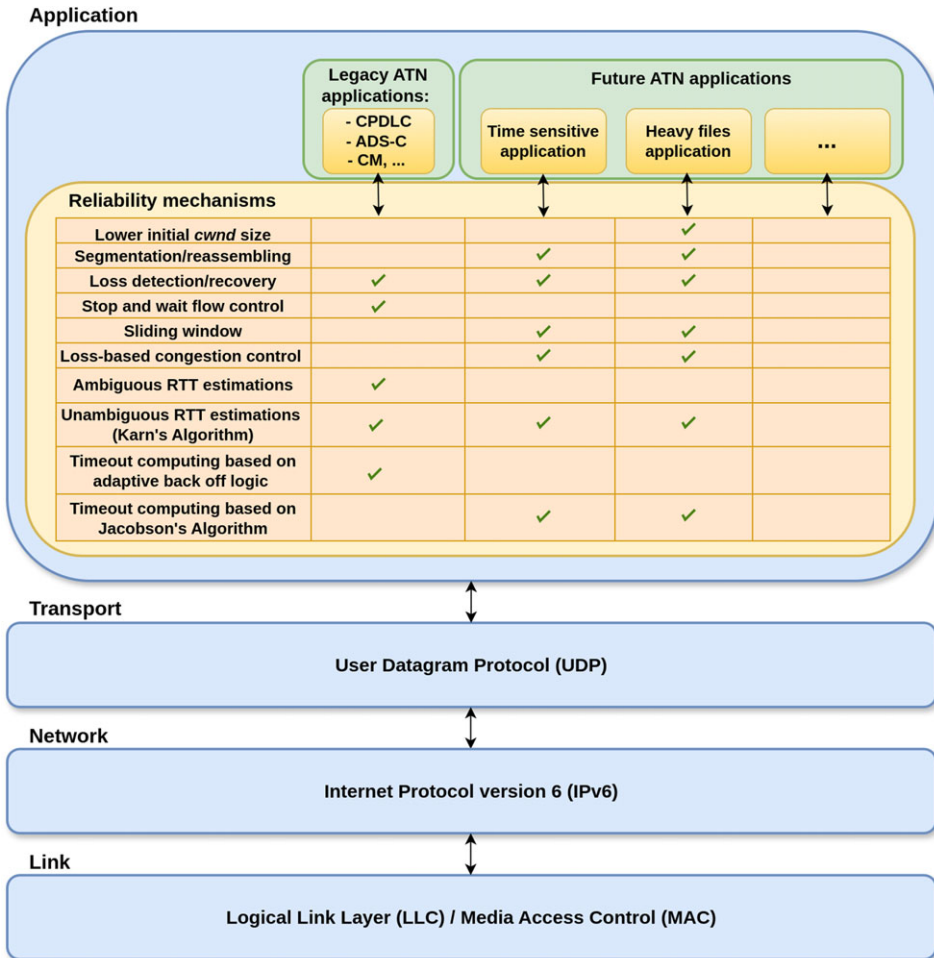


Figure 27. Proposed end system protocol stack for ATN/IPS.

Due to the similarity of data flow in CPDLC with the IoT field, a standardised IoT approach will be used for the RTT estimation. In IoT, various congestion control algorithms were introduced to improve the end-to-end reliability mechanism for constrained application protocol (CoAP). Fast-Slow RTO (FASOR) is one of the proposed algorithms and is in standardisation phase, last updated in 2023 [33]. FASOR aims to provide a trade-off between aggressive and conservative retransmissions logic using a self-adaptive retransmission timer. This technique avoids introducing extra delay with unnecessary retransmissions and reduces flow completion time under link errors. To achieve these objectives, FASOR considers the unambiguous and ambiguous RTTs to update its retransmission timer. The unambiguous RTT is measured from packet that receives an ACK without retransmissions. While an ambiguous RTT is derived from the original transmission of the packet until the receipt of the ACK, regardless of the number of retransmissions.

5.1.3 RTO estimation

To correctly infer losses and avoid making unnecessary retransmissions of data still existing in the network, the RTO must be longer than the actual RTT. FASOR, as specified in Ref. [33], is composed of three key components including FastRTO computation, SlowRTO and innovative retransmission timer

Table 2. ATN applications required features

Parameters	CPDLC	High and medium-rate
	Low-rate Generator	Files Generator
Loss detection & flow control	One-to-one ACK Retransmission timer + maximum retry count	Segmentation/reassembling Cumulative ACK Retransmission timer + maximum retry count Go-Back-N
Congestion control mechanisms	RTT-based congestion control	Loss-based congestion control
RTO estimation	Based on ambiguous & unambiguous RTT estimator Adaptive back off logic	Karn’s algorithm for RTT estimation Jacobson’s algorithm Binary exponential back off
Transport protocol	UDP	UDP

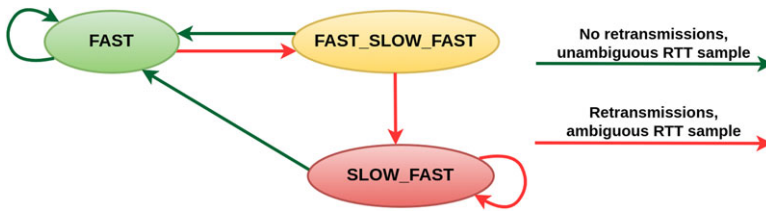


Figure 28. FASOR back off logic state machine.

back off logic. The FastRTO calculation is based on the TCP RTO computation (RFC 6298) [34], with an upper bound of 60s and without minimum RTO bound, using unambiguous RTT sample. Allowing RTO values lower than 1s can improve latency for low RTT environments. SlowRTO has been developed to ensure that only one copy of the message is transmitted before at least one RTT has elapsed. SlowRTO is computed by multiplying the ambiguous RTT by a factor, set to 1.5 by default. To reduce the impact of SlowRTO in cases where RTOs occur for reasons unrelated to congestion, FASOR uses retransmission timer back off logic.

FASOR back off logic has three states that are from the least conservative to most conservative such as FAST, FAST_SLOW_FAST, and SLOW_FAST, as shown in Fig. 28. After receiving ACK without retransmissions, the FASOR sender always stays in or transits back to the FAST state. If an ACK is received only after retransmissions, FASOR switches to a more conservative state of FAST_SLOW_FAST until it reaches the SLOW_FAST state, where it remains as long as completing a message exchange requires retransmissions.

In this way, situations with link errors will be improved by using the back off with the unambiguous RTT-based RTO, and heavy congestion cases will benefit from the systematic use of ambiguous RTT-based RTO [33].

5.2 Required features for future applications

5.2.1 Loss detection and flow control

The future ATN/IPS applications are expected to send large data flows. Hence, for data larger than the SMSS, segmentation and reassembling must be performed. The SMSS value should be set in accordance with the maximum transmission unit (MTU) of the network, which excludes the TCP/IP headers and

options. Similar to CPDLC, each outgoing segment must have a sequence number to enable the recognition of its respective acknowledgment. The receiver uses the sequence number to reorder segments as well as detect out of range segment. To notify the sender of an out-of-range receipt, the receiving entity sends an ACK with next expected sequence number for the missing segment until it is delivered. As result, the sender entity receives duplicate ACKs, which are then used to detect loss.

In terms of flow control, the sliding window protocol will be used along with cumulative ACK. The RTO expiration is considered as an indication of link failure. The segments will then be retransmitted using the Go-Back-N protocol. The maximum number of segment transmissions is limited to three.

5.2.2 Congestion control mechanisms

The performance analysis in Subsection 4.2 demonstrates that the loss-based congestion control mechanism adopted by AIMD transport protocols like TCP New Reno is suitable for the considered future application. As described in Ref. [23], this TCP protocol considers segment loss as sign of network congestion.

The initial congestion window size should be defined based on the SMSS value and the *cwnd* update must meet the standards outlined in Ref. [23]. The high-rate data transmission expected in the future applications can benefit from New Reno features. It includes the slow start phase, the congestion avoidance phase, and the fast recovery mechanism in the event of isolated segment loss [30].

5.2.3 RTO estimation

When the sender does not get an acknowledgment from the receiver, it should rely on the retransmission timer to ensure data delivery. The retransmission timer, called RTO timer, should be computed using Jacobson's algorithm, as outlined in Ref. [35]. The RTT samples used to calculate RTO should meets the requirements of Karn's algorithm [32]. It only considers RTT samples derived from segments that were not retransmitted. The retransmission timer should be managed based on the algorithm defined in Ref. [34]. It avoids the retransmission of a segment in less than one RTO after its previous transmission. When the RTO timer expires, segments that have not been acknowledged should be retransmitted. Then, the sender must update the RTO timer using the binary exponential back off value of $RTO * 2$. A maximum value of 60s may be used to provide an upper bound, as introduced in Ref. [34].

6.0 Conclusion

The research carried out in this study focuses on the performance of transport protocol layers in the context of the migration from ATN/OSI protocol stack to ATN/IPS stack as impulsed by ICAO for future aeronautical network. The reference protocol for ATN/OSI stack is COTP4, in the case of IP protocols the transport layer uses one flavor or another of TCP. The applications are those already implemented in Europe for communications between pilots and controllers (FANS B: CPDLC and ADS-C) but also future applications requiring the transfer of significant volumes of information between the ground and the board.

We have developed a set of simulation models with the OMNET++ framework in order to assess end-to-end performances at application level. CPDLC traffic is modeled on the basis of statistic profiles derived from actual traffic traces recorded in a French regional control centre. For future applications, several generic profiles are used. As the network model aims to analyse the end-to-end performance, it encompasses of a behaviour model of the core network and delay/loss profiles of the VDL Mode 2 data link for the access part. The VDL Mode 2 model has been validated during previous studies carried out for French civil aviation organisation DGAC. The objective has been to stay as close as possible to ICAO standards.

Unsurprisingly, the protocol from the ISO/OSI COTP4 standard exhibits correct performance for the CPDLC application but is incapable of supporting large flow transfers. The assumptions on the underlying protocol layers of the OSI model, namely reliability of all successive links and collaboration of all intermediate nodes through the ECN explicit congestion notification, are not verified in an IP paradigm.

The receiver's control of the volume of data in flight quickly becomes ineffective. As expected, the behaviour of TCP turns out to be unsuitable for the very sporadic CPDLC application. In accordance with the studies presented in Refs [25, 27], TCP triggers mechanisms normally intended to detect link failures or routing problems, which quickly reduces end-to-end delay performance and makes it difficult to meet timing constraints. Note that the performance of the CPDLC transport protocols is evaluated in relation to RCP 130, which constitutes the most restrictive requirement but which ensures the use of data links in all airspaces (oceanic, continental). And certainly, such a way to adopt a relevant set of reliability mechanisms should be used for any new future ATN/IPS applications.

The migration from the ATN/OSI stack to the ATN/IPS stack cannot, therefore, be achieved solely by transcribing the TCP/IP protocols in the ATN context. Our proposal is to adopt a similar paradigm to that of the QUIC protocol currently under standardisation at IETF. The transport layer would then simply be UDP and would not provide any retransmission or flow control mechanism. These mechanisms can then be supported at the application level in order to be adapted to each type of application: time management by timers for CPDLC and more generally FANS B, flow management for future applications (weather, aircraft maintenance, etc.). Our future work focuses on ATN/IPS modeling with the final system protocol stack proposed in this paper. In addition to CPDLC and the file generator, a video streaming application will be investigated. Different air-ground subnetworks will also be considered in this model, including VDL Mode 2 and SATCOM. Multi-path transport will be explored to enable simultaneous use of air-ground subnetwork resources and improve overall end-to-end performance. A simulation-based performance analysis of the proposed protocol will be carried out by considering different scenarios. Furthermore, a benchmarking study will be conducted to compare the TCP new Reno and future results.

Acknowledgements. The authors would like to thank area control centre (ACC) of Bordeaux (France) for supplying the CPDLC log files required to perform the simulations.

Competing interests. The authors declare none.

References

- [1] Apaza, R.D. and Popescu, L. The way to the future has already started: ICAO Aeronautical Telecommunication Network (ATN) using Internet Protocol Suite (IPS) Standards and Protocol evolution update, IEEE/AIAA 37th Digital Avionics Systems Conference (DASC), 2018, pp 1–6.
- [2] Mahmoud, M.S.B., Pirovano, A. and Larriue, N. Aeronautical communication transition from analog to digital data: A network security survey, *Comput. Sci. Rev.*, 2014, **11**, pp 1–29.
- [3] ICAO. Technical Specifications for ATN using ISO/OSI Standards and Protocols, Part III – Upper Layer Communications Service (ULCS) and Internet Communications Service (ICS), ICAO Doc 9880, 2016, second ed.
- [4] ICAO. Manual for the ATN Using IPS Standards And Protocols, ICAO Doc 9896, 2010, first ed.
- [5] ICAO. Manual for the ATN Using IPS Standards And Protocols, ICAO Doc 9896, 2015, second ed.
- [6] RTCA Special Committee 214. RTCA DO-350A/EUROCAE ED-228A Safety and Performance Requirements Standard for Baseline 2 ATS Data Communications (Baseline 2 SPR Standard), RTCA/EUROCAE, March 2016.
- [7] Eurocontrol European Aviation Challenges of Growth, Brussels, Belgium, *Technical Report 2, Autumn/Winter Skyway*, 2018. Available Online, Accessed October 15, 2024, <https://www.eurocontrol.int/sites/default/files/publication/files/2018-autumn-winter-skyway-issue-69.pdfurl>.
- [8] Federal Aviation Administration Commitments, Implementation, and Plan, NextGen Annual Report, 2020. Available Online, Accessed October 15, 2024, <https://www.faa.gov/sites/aa.gov/files/2022-06/NextGenAnnualReport-FiscalYear2020.pdfurl>.
- [9] ICAO Manual of Technical Provisions for the Aeronautical Telecommunication Network (ATN), ICAO Doc 9705, third ed.
- [10] ICAO. Manual On Detailed Technical Specifications For The Aeronautical Telecommunication Network (ATN) Using ISO/OSI Standards And Protocols - Part I - Air-Ground Applications, ICAO Doc 9880, 2016, second ed.
- [11] ICAO. Manual On Detailed Technical Specifications For The Aeronautical Telecommunication Network (ATN) Using ISO/OSI Standards And Protocols, Part IIB : Ground-Ground Application – ATS Message Handling Service (ATSMHS), ICAO Doc 9880, 2016, second ed.
- [12] ICAO. Manual On Detailed Technical Specifications For The Aeronautical Telecommunication Network (ATN) Using ISO/OSI Standards And Protocols, Part IIA : Ground-Ground Application – ATS Interfacility Data Communications (AIDC), ICAO Doc 9880, second ed.
- [13] ICAO. Global Operational Data Link Document (GOLD), ICAO, 2013, second ed.

- [14] Fantappiè, P., et al. Future Communications Infrastructure (FCI) Functional Requirements Document (FRD), Program H2020-SESAR-2015-2, Deliverable D5.2.010, Ref. Ares(2018)3966781, June 2018. Available Online, Accessed June 14, 2024, <https://ec.europa.eu/research/participants/documents/downloadPublic?documentIds=080166e5bc833861appId=PPGMSurl>.
- [15] Postel, J. Transmission Control Protocol, RFC 793, September 1981, doi: [10.17487/RFC0793](https://doi.org/10.17487/RFC0793).
- [16] Jacobson, V., Braden, R. and Borman, D. TCP Extensions for High Performance, RFC 1323, doi: [10.17487/RFC1323](https://doi.org/10.17487/RFC1323).
- [17] Postel, J. User Datagram Protocol, RFC Editor, RFC 768, doi: [10.17487/RFC0768](https://doi.org/10.17487/RFC0768).
- [18] Deering, S. and Hinden, R. Internet Protocol, Version 6 (IPv6) Specification, RFC Editor, RFC 2460, doi: [10.17487/RFC2460](https://doi.org/10.17487/RFC2460).
- [19] Polese, M., Chiariotti, F., Bonetto, E., Rigotto, F., Zanella, A. and Zorzi, M. A survey on recent advances in transport layer protocols, *IEEE Commun. Surv. Tutorials*, 2019, **21**, (4), pp 3584–3608.
- [20] Rhee, I., Xu, L., Ha, S., Zimmermann, A., Eggert, L. and Scheffenegger, R. CUBIC for Fast Long-Distance Networks, RFC 8312, February 2018, doi: [10.17487/RFC8312](https://doi.org/10.17487/RFC8312).
- [21] Iyengar, J. and Swett, I. (Eds) QUIC Loss Detection and Congestion Control, RFC 9002, May 2021, doi: [10.17487/RFC9002](https://doi.org/10.17487/RFC9002).
- [22] Mascolo, S. and Grieco, L. Additive increase early adaptive decrease mechanism for TCP congestion control, 10th International Conference on Telecommunications, ICT 2003, vol. 1, pp 818–825.
- [23] Allman, M., Paxson, V. and Blanton, E. TCP Congestion Control, RFC 5681, September 2009, doi: [10.17487/RFC5681](https://doi.org/10.17487/RFC5681).
- [24] Özmen, S., Hamzaoui, R. and Chen, F. Survey of IP-based air-to-ground data link communication technologies, *J. Transp. Manag.*, 2024, **116**, pp 102579.
- [25] Ehammer, M., Gräupl, T. and Rokitansky, C-H. TCP/IP over aeronautical communication systems—Effects on bandwidth consumption, IEEE/AIAA 28th Digital Avionics Systems Conference, IEEE, 2009, pp 4–A.
- [26] EUROCONTROL/FAA. Communications Operating Concept and Requirements for the Future Radio System, COCR Version 2.0, May 2007, Available Online, Accessed October 15, 2024, https://www.eurocontrol.int/sites/default/files/field_tabs/content/documents/communications/cocr-future-radio-system-v.2.pdfurl.
- [27] Muhammad, M., de Cola, T., Kissling, C. and Berioli, M. Reliable aeronautical services protocol: motivation, design, and performance, IEEE Global Communications Conference (GLOBECOM), IEEE, 2013, pp 3003–3008.
- [28] Lala, V., Pirovano, A. and Radzik, J. Rationale for a new transport protocol in the aeronautical telecommunication network, IEEE/AIAA 42nd Digital Avionics Systems Conference (DASC), October 2023, pp 1–8, doi: [10.1109/DASC58513.2023.10311286](https://doi.org/10.1109/DASC58513.2023.10311286).
- [29] Langley, A., et al. The QUIC transport protocol: design and internet-scale deployment, In SIGCOMM, ACM, 2017, pp 183–196.
- [30] Henderson, T., Floyd, S., Gurtov, A. and Nishida, Y. The NewReno Modification to TCP's Fast Recovery Algorithm, RFC 6582, April 2012, doi: [10.17487/RFC6582](https://doi.org/10.17487/RFC6582).
- [31] Honda, M., Huici, F., Raiciu, C., Araújo, J. and Rizzo, L. Rekindling network protocol innovation with user-level stacks, *ACM Comput. Commun. Rev.*, 2014, **44**, (2), pp 52–58.
- [32] Karn, P. and Partridge, C. Improving round-trip time estimates in reliable transport protocols, SIGCOMM 87 Proceedings of the ACM Workshop on Frontiers in Computer Communications Technology, August 1987.
- [33] Jarvinen, I., Kojo, M., Raitahila, I. and Cao, Z. Fast-Slow Retransmission Timeout and Congestion Control Algorithm for CoAP, RFC, <https://tools.ietf.org/id/draft-jarvinen-core-fasor-02.html>.
- [34] Paxson, V., Allman, M., Chu, J. and Sargent, M. Computing TCP's Retransmission Timer, June 2011, RFC 6298, doi: [10.17487/RFC6298](https://doi.org/10.17487/RFC6298).
- [35] Jacobson, V. Congestion avoidance and control, *Comput. Commun. Rev.*, 1988, **18**, (4), pp 314–329.