Protocols for Highly-Dynamic Airborne Networks

Mohammed J.F. Alenazi, Dan S. Broyles, Santosh A. Gogi, Hemanth Narra, Kamakshi Sirisha Pathapati, Kevin Peters, Dongsheng Zhang, Egemen K. Çetinkaya, Abdul Jabbar, Justin P. Rohrer, and James P.G. Sterbenz

EECS and ITTC, The University of Kansas – http://www.ittc.ku.edu/resilinet

I. MOTIVATION

• Motivation
Highly dynamic airborne tactical networks pose unique challenges to end-to-end data transmission. Mobility introduces a significant challenge since airborne nodes can travel at relative speeds as high as Mach 7. In addition, the network is severely bandwidth-constrained due to the limited spectrum allocated to tactical networks. The energy available for data transmission among airborne nodes is limited. Intermittent connectivity is also a challenge, which is caused by the extremely short contact duration between any two nodes. The current TCP/IP-based Internet architecture is not designed to function in this environment. We present the design, modelling, and implementation of the ANTP protocol suite that is optimised for the tactical environment, while maintaining edge-to-edge compatibility with the legacy Internet architecture.

• Dynamic Airborne Tactical Environment
A typical airborne tactical network as depicted in Figure 1 consists of three types of nodes: airborne nodes (AN), ground stations (GS), and relay nodes (RN). The airborne nodes contain a variety of data collection devices. The GSs are located on the ground (stationary or portable) and typically have a much higher transmission range than that of an AN. The GS also houses a gateway (GW) that connects the airborne network to several terminals that may run control applications for various devices on an AN.

II. DESIGN and MODELLING of the ANTP PROTOCOL SUITE

• AeroTP: TCP-Friendly End-to-End Transport
We have designed a new domain-specific transport protocol AeroTP, which is targeted for analysis of the aeroenvironmental environment while being TCP-friendly to allow seamless splicing with conventional TCP at the network edge in the GS and on the AN. AeroTP has several operational modes that support different service classes: reliable, nearly-reliable, quasi-reliable, best-effort connections, and best-effort datagrams. The first three of these are fully TCP compatible, the last fully UDP compatible, and the others TCP-friendly with reliability semantics matching the needs of the mission as shown in Figure 3. The AeroTP header is designed to permit efficient translation between TCP/UDP and AeroTP using a nsi-3 network simulator. Over the course of the simulation, both TCP and AeroTP are able to deliver the full 1 MB of data transmitted for low error rates <0.000035, but above that the TCP performance drops rapidly while AeroTP is still able to deliver nearly all the data at the highest error rates as shown in Figure 4. In the same plot we see that UDP loses a portion of the data due to corruption as the BER increases, and that the AeroTP quasi-reliable mode loses a much smaller percentage. Our simulation results show that AeroTP has a significant advantage in lossy environments.

• AeroNP: IP-Compatible Network Protocol
The AeroNP is an IP-compatible network protocol, which provides services to the AeroTP transport protocol as well as the AeroRP routing protocol. AeroNP encapsulates the segments and packets coming from AeroTP and AeroRP into the protocol header shown in Figure 5. In addition, AeroNP provides QoS, congestion-control, and error detection services to the transport layer protocol. QoS is provided by maintaining priority queues for the different levels of application data. Depending on the mission requirements, geolocation information can be included in the AeroNP header. We designate the AeroNP header with the geolocation information as the extended header, whereas the AeroNP header without the geolocation information is referred to as the basic header.

• AeroRP: Geolocation-Assisted Routing
Both reactive and proactive routing protocols fail to operate in partially connected networks since a complete path may not exist at all times. We designed and modelled a geographic routing protocol that leverages location information combined with limited updates to build the forwarding table. Determining the next-hop is based on a metric called time to intercept (TTI) that is calculated based on inter-node distance (Δd), transmission range (R), and speed component (v):

TTI = \begin{cases} 0 & \text{for } \Delta d \leq R \\ \Delta d - R & \text{otherwise} \end{cases}

Our ns-3 simulation analysis indicates that multimodal AeroRP outperforms traditional MANET routing protocols as shown in Figure 6.

III. IMPLEMENTATION

• Architecture
The implementation architecture is designed to provide several features: maintainability, reliability, and data analysis accessibility. To achieve maintainability, the system is fully designed based on the object oriented programming (OOP) approach. This approach attempts to eliminate the dependency between data structures, which allows us to upgrade the components while minimizing the number of error generated by interdependencies. For reliability, the system employs try-catch error handling to avoid any run-time errors. For performance analysis, the system provides a shared logging system that can aggregate the logs in a single web server. Based on these considerations, the system is divided into several components as shown in Figure 7.

AeroNP provides the interface between the higher layers of the ANTP protocol suite and airborne network. The system architecture is implemented in Python in two phases. First, we implement each protocol separately and test their functionality on PlanetLab testbed. After that, we integrate all the components and verify the correctness of the system integration. Finally, the implementation is deployed in embedded processors on radio-controlled aircraft and ground vehicles.

IV. REFERENCES


Acknowledgements: This project is supported in part by the US DoD and IFT Programs