A Distributed Approach to High-Rate Delay Tolerant Networking Within a Virtualized Environment

Rachel Dudukovich, Blake LaFuente
Cognitive Signal Processing
NASA Glenn Research Center
Cleveland, Ohio, USA
{rachel.m.dudukovich,blake.a.lafuente}@nasa.gov

Alan Hylton
Secure Networks, System Integration and Test
NASA Glenn Research Center
Cleveland, Ohio, USA
alan.g.hylton@nasa.gov

Brian Tomko
Information and Applications Office
NASA Glenn Research Center
Cleveland, Ohio, USA
brian.j.tomko@nasa.gov

Jeffrey Follo
Flight Software Branch
NASA Glenn Research Center
Cleveland, Ohio, USA
jeffrey.c.follo@nasa.gov

***Abstract*—The High-Rate Delay Tolerant Networking (HDTN) project has taken a distributed service-based approach to the development of a highly efficient delay tolerant networking (DTN) implementation. Through the analysis of many DTN implementations, system and mission requirements as well as the DTN protocol specifications, HDTN has worked to infuse modern computing technologies into the NASA approach to interplanetary networking.**

The initial use case of the HDTN software runs on a hypervisor representative of the International Space Station (ISS) DTN Gateway. In this scenario, multiple emulated payloads will send science data through HDTN to a mission operations center. HDTN will provide store and forward capability as well as network flow management.

This paper discusses the infusion path of cognitive networking technologies in the NASA Space Communications and Navigation (SCaN) networks using the DTN architecture and protocols as the basis for cognitive routing and network management capabilities. HDTN has been developing the Bundle Protocol encoding and decoding mechanisms and messaging framework that can be used as the basis for integrating DTN with various learning and decision-making processes. The concepts of distributed computing, network virtualization, software defined networking and delay tolerant networking are basic building blocks which will further the development of cognitive networking. In addition to discussion of the HDTN software development and testing, this paper examines the role that each of these technologies play in the evolution of the current state of space networking into an intelligent network of networks.

***Index Terms*—Delay Tolerant Networking, Distributed Computing, Network Function Virtualization**

I. Introduction

The High-Rate Delay Tolerant Networking (HDTN) [1], [2] project has been developing a state-of-the-art distributed software architecture aimed at optimizing network data processing tasks as much as possible in order to provide Delay Tolerant Networking (DTN) [3] services to high-rate space communication links, such as optical communication links. In particular, HDTN has been targeting the International Space Station (ISS) and LunaNet [4] as potential infusion paths.

High-Rate Delay Tolerant Networking

There are multiple use-cases for high-rate DTN services, such as hyperspectral imaging earth science missions [5], optical communication missions [6], and high rate trunk links envisioned as part of the next generation of NASA interplanetary networks. HDTN has promising infusion paths through several NASA missions including LunaNet and the ISS.

International Space Station: The International Space Station has been performing a series of communication upgrades, including the infusion of DTN in ISS science payloads as well as the implementation of a DTN gateway [7]. Optical communication will be demonstrated on ISS as part of the ILLUMA-T mission [8]. HDTN is addressing the need for a DTN gateway capable of supporting optical data rates (greater than 1 Gbps).

Testing of DTN aboard the ISS follows a flight heritage which first began in 2016 in which laptops represented DTN nodes and endpoints for on-orbit and ground [7]. The on-orbit system utilized ION, an open-source version of the bundle protocol developed by the Jet Propulsion Lab (JPL), and a Debian Virtual Machine. The ground gateway used a RedHat™ LINUX virtual machine running DTN 2.9 software and communicated to the on-orbit system using a Ku-Band communications system. This instance was able to achieve a nominal downlink of 20 Mbps and uplink of 4 Mbps over Licklider Transmission Protocol (LTP).

LunaNet: The NASA Space Communications and Navigation (SCaN) program has been developing the network

U.S. government work not protected under U.S. copyright law.

architecture for the next generation of missions to the lunar surface. LunaNet [4] has identified both DTN and high-rate links as essential parts of this architecture.

Earth Science: DTN has also been tested to show its potential benefit in Earth Science missions. One example is the NASA Tracking Data Relay Satellite System (TDRSS) where DTN was used as a potential solution for issues surrounding asymmetric and simplex link data rates [5]. The Internet Router in Space (IRIS) demonstrated that DTN is able to reduce latency and make use of the nodes to store and forward data to prevent data loss when links are disrupted [5].

DTN in the Cognitive Networking Roadmap

Delay Tolerant Networking plays a significant role in the development of future cognitive networking technologies [9], [10]. Missions such as LunaNet have specifically identified DTN as part of the future lunar architecture, therefore it is reasonable to envision DTN as the basis of future network technologies since LunaNet will be a major milestone in the development of an interplanetary network. DTN provides a basic architecture which defines a common network layer among dissimilar network elements, allowing interoperability between different classes of nodes. The store-carry-forward technique implemented by DTN allows science data to remain in non-volatile storage when network connectivity is disrupted. Custody transfer ensures that data will not be deleted until it has been accepted by a participating DTN node, guaranteeing that data will eventually be delivered to its final destination. In addition to these elementary DTN functions, there are several additional concepts that can further enable cognitive networking. These include:

- Bundle Priority - Data can be intelligently prioritized according to critically.
- Fragmentation - Units of data can be sized according to predicted data rates and contact duration. The bundle size could be optimized using a variety of techniques.
- Anycast - The cognitive network system can select one or multiple links to transmit data to. Decision-making can be applied to determine if all available paths or a subset of paths should be used.
- Link Selection - A variety of techniques can be used to predict an optimal link or path from a set of possible options. Multiple objectives can be considered such as data rate, predicted delivery time, dollar cost, and reliability.
- Routing - Beyond link selection which may be applied to a single or multi-hop network, routing can determine optimal end-to-end paths in a multi-hop network. Several existing algorithms are tailored specifically to routing in DTNs, which typically occurs at the bundle layer. These include but are not limited to: Contact Graph Routing (CGR) [11], the Cognitive Space Gateway (CSG) [12], Delay Tolerant Link State Routing (DTLSR) [13], Probabilistic Routing Protocol for Intermittently Connected Networks (PRoPHET) [14], and many others.
- Neighbor Discovery - In order to enable opportunistic networking, nodes must become aware of when new neighbors have joined the network. They must be able to trade connectivity information with each other without the need for pre-configured connections. Several approaches exist including Link Layer Discovery Protocol (LLDP) and DTN IP Neighbor Discovery [15], [16].

In addition to these basic building blocks, the HDTN project overlaps with cognitive networking by attempting to apply high performance computing elements to a potentially size, weight, and power (SWaP) constrained platform. Machine learning and artificial intelligence have made strides in the utilization of Graphics Processing Units (GPUs), high performance solid state drives, and memory/processing intensive applications. Many of these same technologies can be applied to high performance networking and network function virtualization. Fig. 1 shows several of these interrelated technologies. The elements of delay tolerant networking, machine learning and artificial intelligence, network architectures and modeling, and high performance/low SWaP computing comprise the research areas that may help to enable cognitive networking for the space environment.

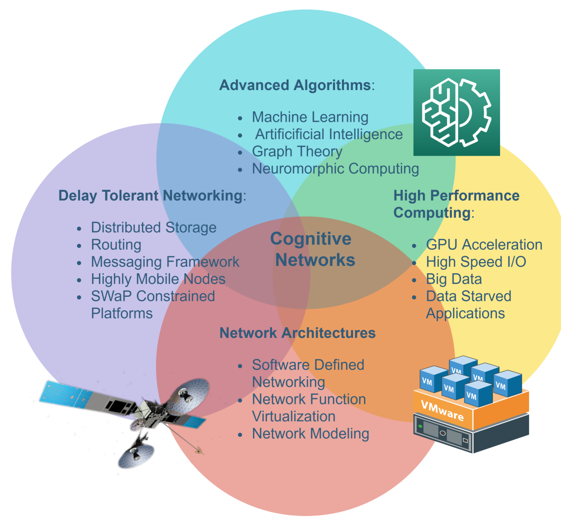

Fig. 1. Cognitive Networking Technologies

II. Software Implementation and Testing

The HDTN software architecture was developed to simplify the complex networking tasks required by a DTN bundle agent (bundle ingress/decoding, bundle storage, storage maintenance, event scheduling, and bundle egress/encoding) by decomposing them into independent processes. These processes may execute as a distributed system on multiple different hosts (both physical and virtual) or on the same system. This allows I/O intensive tasks such as bundle storage and retrieval to potentially be parallelized if needed.

Each process replicates its state using ZeroMQ [17] as its message bus. Fig. 2 shows a high-level diagram of the HDTN bundle generation tool (BPGen), HDTN ingress, storage, and egress processes. In this instance, the system is using an asynchronous TCP convergence layer (TCPCL).

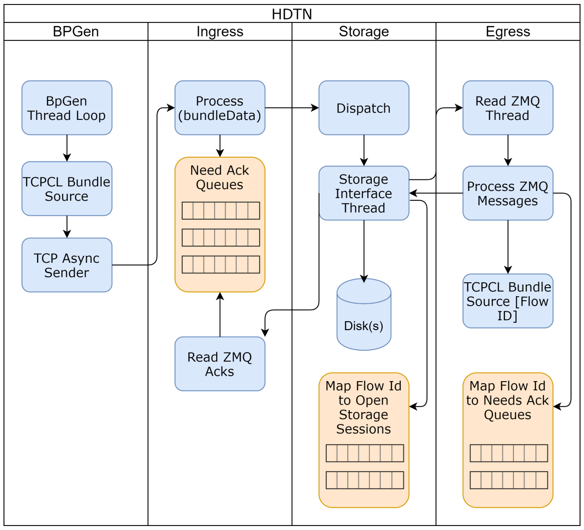

Fig. 2. High-Level HDTN Processes

The HDTN software architecture intentionally avoids the use of shared memory for several reasons. In order to attain the high data rates (1 Gbps or greater) that HDTN is targeting, processes should not lock in order to synchronize the use of shared resources. Instead, HDTN uses a message bus approach to share data between its processes. This will allow each process to operate independently of one another. This approach will also allow processes to be distributed among multiple physical or virtual machines, which could potentially provide a performance increase through parallelization.

Convergence Layers

HDTN currently supports UDP, TCP, STCP and LTP (in development) convergence layers. These are among the most common transport protocols used for DTN. TCP and STCP can support links with small delays but will begin to perform poorly as round trip times increase. It is for this reason that UDP and LTP [18] over UDP are frequently used. In addition to protocols supported within the common terrestrial network stack, custom convergence layers can be developed to interface to software defined radios and other hardware used within the space environment. Fig. 3 shows the initial configuration HDTN will use for performance and interoperability testing. This configuration is similar to what is used on-board ISS for the DTN science payloads, DTN gateway, and ground nodes.

Scheduler and Routing Interface

The next main component that is being developed for HDTN is a scheduler and routing interface. The most basic version of this will use a table-based contact schedule to determine which neighbors are available for the egress to forward data to. If no appropriate neighbor is available, data will be stored in non-volatile storage. The scheduler must also determine when stored data can be released to a newly available neighbor. The input data used for the scheduler will roughly follow the contact plan developed as part of the CGR algorithm,

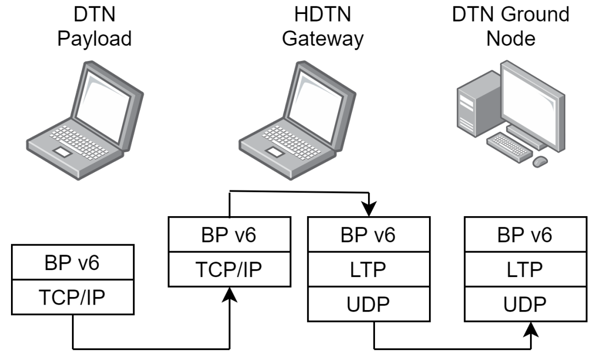

Fig. 3. HDTN Test Protocol Stack

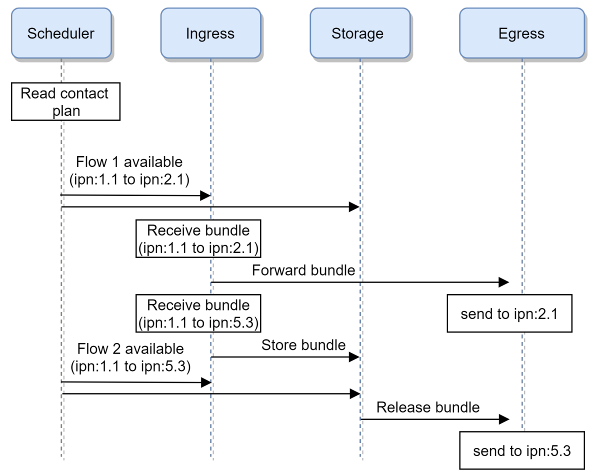

Fig. 4. Basic Scheduler Concept

consisting of the source and destination nodes, contact start and stop time, data rate, and distance between nodes. In addition, we assign a flow id to each source and destination pair. The schedule will be an event driven process that will read contacts from the contact plan and send event notifications to the ingress, storage and egress as needed. Fig. 4 shows a high-level diagram of this concept.

Performance Evaluation

The HDTN software is currently being tested in several ways. The first is with a custom set of unit tests that were developed to ensure that messages are reliably completing the entire message pipeline, meaning all messages received from the BPGen test tool are exiting the egress and are being counted by a bundle sink tool (BPSink). These tools have been released along with the basic bundle encoding and decoding functions as part of the BPCodec library [19]. The main HDTN bundle agent components are planned for open source release once testing and development are completed.

In addition to HDTN specific unit testing, the UDP, TCP, STCP, and LTP convergence layers are being tested with the Bundle Protocol version 6 [20] implementations from two well known NASA-based DTN implementations. The first of these is the Interplanetary Overlay Network (ION) which has been developed out of NASA's Jet Propulsion Laboratory [21]. The next implementation is the newly released DTN Marshall Enterprise Implementation (DTNME) [22]. DTNME is based on DTN2 [23] and is being developed out of Marshall Space Flight Center by the Huntsville Operations Support Center (HOSC) DTN team. Both ION and DTNME are planned for use on the ISS to provide DTN capabilities to science payloads. For this reason, HDTN must maintain compatibility and interoperability with both implementations.

Initial benchmark testing is in progress for the LTP convergence layer. Preliminary testing was done using two HDTN lab computers connected over a 1 GbE link. An 800 MB file was transferred using LTP over UDP. The Linux tool NetEm[24] was used to create a 5 percent packet drop in the final test case. These results are quite promising and show that HDTN should be able to support 1 Gbps downlinks from the ISS to ground. Table 1 summarizes these preliminary tests.

TABLE I
INITIAL LTP OVER UDP PERFORMANCE

Data Rate (Mbps)	MTU	TX Packet Drop (%)
950	40000	0
914	1000	0
428	1000	5

ISS Emulation Environment

The HDTN software will be tested on a virtual machine image that is used for the ISS DTN payload gateway. Kernel-based Virtual Machine (KVM) is used as the hypervisor with a Debian guest operating system. This virtual machine is hosted within the space networking lab at NASA Glenn Research Center and will be connected to the JSC Software Development and Integration Laboratory (SDIL). The SDIL can emulate the ISS Joint Station LAN (JSL) payloads, DTN gateways, as well as the DTN ground nodes located at the Huntsville Operations Support Center (HOSC). This will create a realistic software and networking environment to test the basic HDTN functionality. Once the software has been tested successfully, the virtual machine image with the HDTN software installed will be delivered to the ISS team. The virtual machine can then be installed on a laptop on-board ISS as an experiment to demonstrate HDTN. Fig. 5 shows a high-level diagram of this network between Sonny Carter SDIL and Glenn Research Center.

Initial testing will be focused on establishing a connection to the SDIL DTN network, setting up a simplified configuration representative of the ISS DTN network, and sourcing bundles from an emulated DTN payload node running ION and forwarding the bundles through HDTN to an emulated ground node. The payload and ground nodes are located within the SDIL and HDTN runs locally at GRC. The protocols to be used are as shown in Fig. 3. HDTN's goal is to reach data rates of at least 1 Gbps, however this may be dependent on payload and ground node configuration, as well as the HDTN virtual environment.

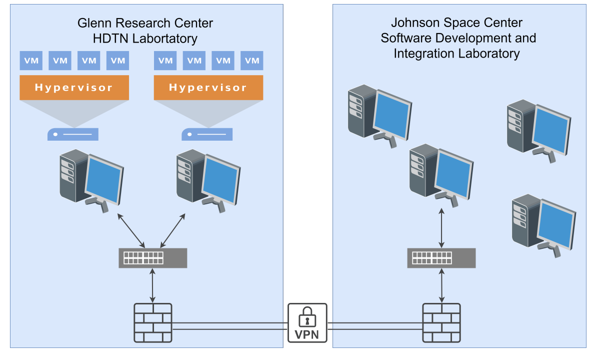

Fig. 5. ISS Emulation Network

III. FUTURE WORK

It is envisioned that upon completion of the HDTN SDIL testing, the software can be used on a variety of complementary missions requiring DTN capabilities such as potential CubeSat missions and cognitive networking demonstrations. There are several areas related to cognitive networking that the HDTN team intends to explore. These include:

- Software Defined Networking (SDN) - SDN can be used for a variety of purposes such as intelligent traffic control, load balancing and implementation of Quality of Service (QoS) requirements [25].
- High Performance Virtualization Environments - High performance server hardware is continually evolving. HDTN seeks to utilize these advances to enable high-rate networking, network function virtualization and to improve the virtual test environment as discussed in [26].
- Cognitive Routing - The basic bundle agent developed by HDTN can serve as the basis of a modular architecture in which a variety of algorithms can be implemented. Many cognitive and AI/ML based concepts exist. The Cognitive Space Gateway is at the forefront among many of these since its central decision-making element, the Cognitive Network Controller, has been previously used in a realistic demonstration on-board ISS [12]. The independent processes of the HDTN architecture could also be enhanced through the development of a Multi-Agent Reinforcement Learning system [27] to optimize decision-making.
- Delay Tolerant Networking Dataset - While there are many types of datasets readily available to begin studying and developing machine learning and artificial intelligence models, there are few resources available for space networks and DTN. Logging features to record network statics are being developed as part of HDTN. It is planned that statistics such as bundles per second, bundle delivery ratio, congestion information, buffer capacity, and node contact information can be saved in a time series format to produce datasets that can be used for a variety of machine learning purposes.
- Custom Convergence Layers - In order to support a variety of missions, lower-level interfaces to radio hardware

must be developed. In addition, support for multicast and anycast are also of interest.

These areas will further the development of cognitive networking technologies and are considered key areas to focus on for future endeavors.

Acknowledgment

The authors wish to acknowledge the contributions of Dr. Daniel Raible, John Nowakowski, Brenda Lyons and Philip Baldwin for their continuing support of the High Rate Delay/Disruption Tolerant Networking project at NASA Glenn Research Center. The authors also wish to thank Gilbert Clark for all of his work on the development of HDTN and cognitive networking concepts.

References

[1] A. Hylton and D. Raible. "High Data Rate Architecture (HiDRA)". In: *Ka and Broadband Communications Conference 2016*. 2016.

[2] A. Hylton et al. "Rising above the cloud: Toward high-rate delay-tolerant networking in low earth orbit". In: *Advances in Communications Satellite Systems. Proceedings of the 37th International Communications Satellite Systems Conference (ICSSC-2019)*. 2019, pp. 1–17. DOI: 10.1049/cp.2019.1216.

[3] V. Cerf, S. Burleigh, and K. Fall. *Delay-Tolerant Networking Architecture*. https://tools.ietf.org/html/rfc4838. Apr. 2007.

[4] D. J. Israel et al. "LunaNet: a Flexible and Extensible Lunar Exploration Communications and Navigation Infrastructure". In: *2020 IEEE Aerospace Conference*. 2020, pp. 1–14. DOI: 10.1109/AERO47225.2020.9172509.

[5] F. Davis, J. Marquart, and G. Menke. "Benefits of Delay Tolerant Networking for Earth Science Missions". In: *IEEE Aerospace Conference 2012*. 2012.

[6] H. Hemmati, A. Biswas, and I. Djordjevic. "Deep-Space Optical Communications: Future Perspectives and Applications". In: *Proceedings of the IEEE* 99.11 (2011), pp. 2020–2039. DOI: 10.1109/JPROC.2011.2160609.

[7] A. Schlesinger et al. "Delay/Disruption Tolerant Networking for the International Space Station (ISS)". In: *2017 IEEE Aerospace Conference*. 2017, pp. 1–14. DOI: 10.1109/AERO.2017.7943857.

[8] A. Seas et al. "Optical Communications Systems for NASA's Human Space Flight Missions". In: *International Conference on Space Optics (ICSO) 2018*. 2018.

[9] G. Clark et al. "Architecture for Cognitive Networking within NASA's Future Space Communications Infrastructure". In: *34th AIAA International Communications Satellite Systems Conference*. 2016.

[10] W. Ivancic et al. *Cognitive Networking With Regards to NASA's Space Communication and Navigation Program*. Tech. rep. NASA Glenn Research Center, 2013.

[11] J. Fraire et al. "Assessing Contact Graph Routing Performance and Reliability in Distributed Satellite Constellations". In: *Journal of Computer Networks and Communications* (2017).

[12] R. Lent. "An Anycast Service with Cognitive DTN Routing". In: *2020 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*. 2020, pp. 1–6. DOI: 10.1109/ANTS50601.2020.9342841.

[13] M. Demmer and K. Fall. "DTLSR: Delay Tolerant Routing for Developing Regions". In: *Proceedings of the 2007 Workshop on Networked Systems for Developing Regions*.

[14] A. Lindgren and A. Doria. *Probabilistic Routing Protocol for Intermittently Connected Networks*. https://tools.ietf.org/html/draft-lindgren-dtnrg-prophet-02. 2006.

[15] M. Rodolfi. "DTN Discovery and Routing: From Space Applications to Terrestrial Networks". MA thesis. University of Bologna, 2014.

[16] D. Ellard, R. Altmann, and A. Gladd. *DTN IP Neighbor Discovery (IPND)*. https://tools.ietf.org/html/draft-irtf-dtnrg-ipnd-02. Nov. 2012.

[17] P. Hintjens. *ZeroMQ*. https://zeromq.org/. 2020.

[18] M. Ramadas, S. Burleigh, and S. Farrell. *Licklider Transmission Protocol-Specification, RFC 5326*. https://tools.ietf.org/html/rfc5326. 2008.

[19] G. Clark, R. Dudukovich, A. Hylton. *HDTN-BPCodec*. https://github.com/nasa/HDTN-BPCodec. 2020.

[20] K. Scott and S. Burleigh. *Bundle Protocol Specification*. https://tools.ietf.org/html/rfc5050. 2007.

[21] S. Burleigh. *Interplanetary Overlay Network (ION)*. https://sourceforge.net/projects/ion-dtn/.

[22] J. Deaton. *DTN Marshal Enterprise Implementation (DTNME)*. https://github.com/nasa/DTNME.

[23] M. Demmer. *DTN2 Manual*. http://dtn.sourceforge.net/DTN2/doc/manual/.

[24] S. Hemminger. "Network Emulation with NetEm". In:

[25] M. Li, P. Si, and Y. Zhang. "Delay-Tolerant Data Traffic to Software-Defined Vehicular Networks With Mobile Edge Computing in Smart City". In: *IEEE Transactions on Vehicular Technology* 67.10 (2018), pp. 9073–9086. DOI: 10.1109/TVT.2018.2865211.

[26] T. Li et al. "GPU Resource Sharing and Virtualization on High Performance Computing Systems". In: *2011 International Conference on Parallel Processing*. 2011, pp. 733–742. DOI: 10.1109/ICPP.2011.88.

[27] L. Busoniu, R. Babuska, and B. De Scutter. "Innovations in Multi-Agent Systems and Applications – 1". In: Springer, 2010. Chap. Multi-agent reinforcement learning: An overview, pp. 183–221.