SPICE Testbed: A DTN Testbed for Satellite and Space Communications

Ioannis Komnios, Ioannis Alexiadis, Nikolaos Bezirgiannidis, Sotiris Diamantopoulos, Sotirios-A. Lenas, Giorgos Papastergiou, and Vassilis Tsaoussidis

Space Internetworking Center, Office 1, Building A, Panepistimioupoli Kimmeria,
Department of Electrical and Computer Engineering, Democritus University of Thrace,
67100, Xanthi, Greece

{ikomnios, ialex, nbezirgi, sdiaman, slenas, gpapaste, vtsaousi}@ee.duth.gr

Abstract. This paper presents SPICE testbed, a state-of-the-art Delay Tolerant Networking testbed for satellite and space communications deployed at the Space Internetworking Center, Greece. The core of the testbed relies on the Bundle Protocol and its architecture has been designed to support multiple DTN implementations and a variety of underlying and overlying protocols. SPICE testbed is equipped with specialised hardware components for the accurate emulation of space links and ground stations, such as Portable Satellite Simulator (PSS) and CORTEX CRT system, as well as protocols and mechanisms specifically designed for space DTNs. Performance and functionality evaluations on SPICE testbed show that it is an ideal platform to evaluate newly developed mechanisms in a variety of space communication scenarios.

Keywords: Delay Tolerant Networking, DTN Testbed, Bundle Protocol, ION, DTPC, BSS, BDTE, Space Communications, Interplanetary Internet

1 Introduction

Delay Tolerant Networking (DTN [1]) has emerged as a promising solution to the upcoming explosion in the volume of data produced by space assets and delivered to Earth. Since 2007, the Consultative Committee for Space Data Systems (CCSDS [2]) has formed a work group for the standardisation of space delay-tolerant protocols. SPICE researchers have been actively involved in the standardisation procedures and have developed a prototype DTN testbed for space communications under a contract of the European Space Agency (ESA [3]).

Due to the nature of space communications and the restricted amount of space assets, the design of a space-oriented DTN testbed does not necessarily need hundreds of nodes, but requires accurate emulation of space components and links that support diverse protocol stacks, depending on the nature of each asset. It is crucial to offer to researchers a realistic testing environment in order to evaluate, benchmark and optimise new protocols. In this context, SPICE testbed has received funding from EC's FP7 SPICE project [4] in order to be enhanced with more nodes and specialised components that accurately emulate the functionality of typical ground stations, space links and satellites. Our aim is to build an experimental research environment for developing and evaluating a variety of new architectures and protocols for space communications. In particular, SPICE testbed presents the following key features:

i) *Realistic emulation of space communications*. Unlike the majority of existing DTN testbeds, which focus on terrestrial delay-tolerant communications, SPICE testbed provides a realistic experimental environment for satellite and space communications, including real and flight-ready components. Indeed, specialised hardware and software components have been incorporated into the testbed, enabling the testing, evaluation and validation of implemented mechanisms and protocols. Furthermore, a link with a geostationary satellite, namely HellasSat 2, is utilised on demand, to provide real satellite link characteristics for experimental purposes.

ii) *Compliance with typical equipment of major space agencies*. SPICE testbed incorporates typical components used by space agencies for the evaluation of protocols prior to mission launch. In particular, the Portable Satellite Simulator (PSS [5]) was built in compliance with ESA's requirements, while CORTEX CRT [6] is used by all major space agencies in their ground station facilities to support their missions. Finally, Satellite Tool Kit (STK [7]) is employed by mission designers as a tool to calculate not only exact satellite trajectories and contact durations, but also detailed communication characteristics, and perform link-budget analysis.

iii) *Interface provision for multiple underlying protocols*. SPICE testbed not only supports a variety of convergence layers for underlying protocols that comply with CCSDS standards and major space agencies, but also facilitates the development of novel routing, transport, and management schemes. Taking advantage of this functionality, SPICE researchers are able to validate such schemes against standardised protocols and perform interoperability testing.

iv) *Scalability*. SPICE testbed includes numerous nodes for the evaluation of complex communication scenarios that involve several space assets and can be further enhanced with virtual nodes installed on a high-performance server. Therefore, complex scenarios involving constellations of satellites (e.g., cubesats) and several end-users can be realistically modeled. It should also be mentioned that this scalability comes without adding any complexity, since the testbed is easily configured and controlled through dedicated workstations.

The remainder of the paper is structured as follows: Section 2 details the architecture of SPICE testbed and its major components. In Section 3 we refer to novel protocols and mechanisms that have been developed and evaluated in the testbed, along with sample results, while in Section 4 we present related work. The paper is concluded in Section 5.

2. SPICE Testbed Architecture and Components

2.1 SPICE Testbed Architecture

Notionally, the testbed comprises two distinct parts, namely the data plane and the control plane, and its architecture is depicted in Fig. 1. Data is transferred between nodes to emulate communication among space and ground assets through the data plane, while configuration scripts, control messages, and reports related to the emulation are managed through the control plane. Each plane is described in detail below.

Control plane - The control plane is responsible for (a) configuring and controlling the testbed nodes in real time based on user input, (b) monitoring the correct node operation, (c) collecting any associated performance statistics, and (d) delivering the experimental results to the researchers. These operations are coordinated by a main controller accessible via the internal network or the Internet. A hardware firewall restricts remote access, allowing only encrypted VPN connections. Researchers configure the experiments to be conducted through a user interface (UI), available at the main controller. Link characteristics and emulation parameters are either imported directly by the users or provided by the STK workstation after conducting the relevant simulations. Upon the completion of an experiment, results are collected and stored in the main controller.

Data plane – SPICE testbed supports the emulation of a wide variety of space and satellite communication scenarios, including present and future missions. These scenarios may involve (a) a number of landed assets, such as landers and rovers, that generate scientific data and can possibly form a planetary network, (b) a set of space assets near Earth or in deep Space (e.g. LEO/MEO/GEO satellites, spacecraft, planetary relay satellites etc.) that can produce and/or relay data, (c) terrestrial facilities such as typical ground stations (GS), mission operation centers (MOC) and end-users. Researchers are able to emulate all these types of space communications taking advantage of the diverse protocol stack configurations supported by SPICE testbed (Fig. 2).

In particular, an implementation of Proximity-1 [8] is employed as a CCSDS data link protocol to interconnect planetary nodes with relay satellites. Within space DTN network each space asset is emulated by a distinct DTN node. Depending on the objective of the emulation, researchers may use one of the three available CCSDS data link protocol implementations to interconnect space and ground station DTN networks:

Type I: Software-based emulation of the basic functionality of TM/TC/AOS protocols [9-11]. Space assets and ground stations are emulated using only ION-DTN implementation [12].

Type II: Software-based emulation of the full functionality of TM/TC/AOS protocols including Space Link Extension (SLE). Space assets are emulated using ION-DTN, as well as SIMSAT Software. Ground stations do not support DTN and receive AOS/TM/TC packets using SIMSAT.

Type III: Hardware-based emulation of the full functionality of TM/TC protocols. Space assets are emulated utilizing ION-DTN and the combination of SIMSAT and PSS. In this case, ground stations do not support DTN and only receive TM/TC frames using CORTEX CRT system.

Communication between a ground station (GS) and a mission operations center (MOC) can be either DTN-based (Type I) or SLE-based (Type II and Type III). Space data are then transferred to the end-users using ION-DTN. HellasSat 2 satellite may also be employed to provide real satellite link characteristics between DTN nodes.

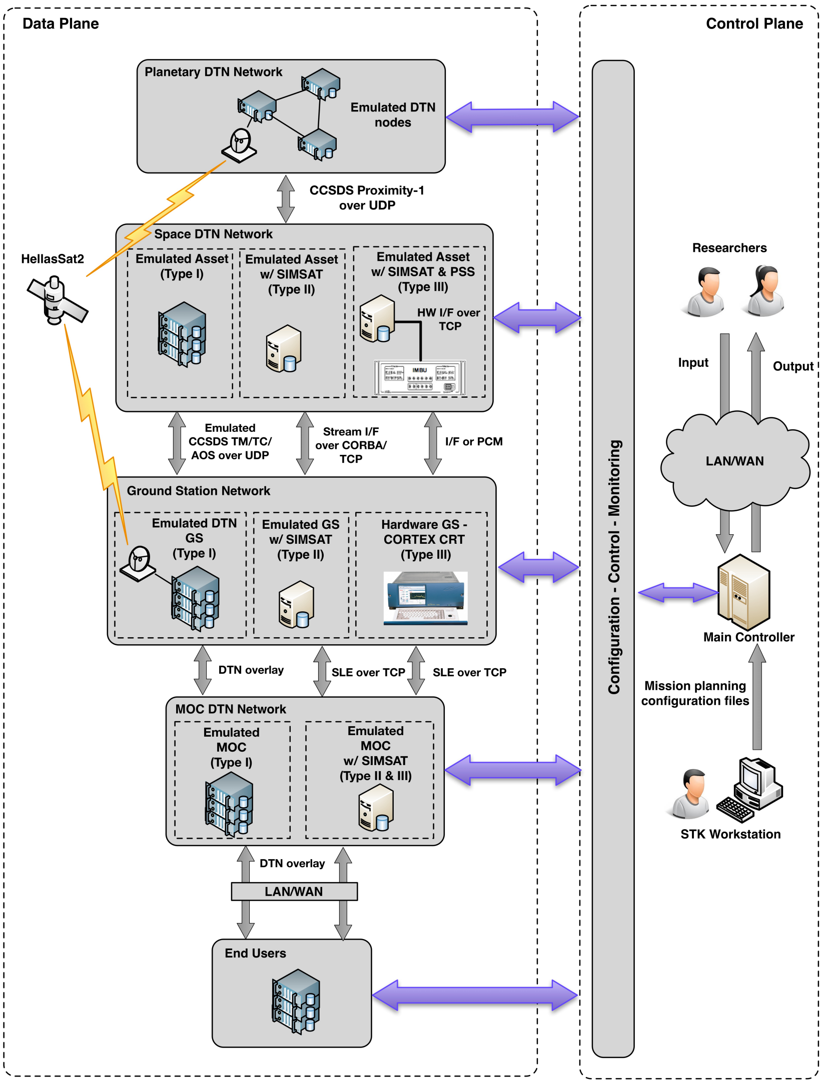

Fig. 1. SPICE Testbed Architecture

2.2 Hardware Components

DTN Nodes - The DTN nodes are fifteen rack-mounted servers used as distinct emulation nodes in experiments with network protocols of the DTN architecture. Each node is equipped with a quad-core Intel Xeon CPU operating at 2.4GHz with 4GB of RAM and 1TB of storage, running a Linux distribution. Private IP addresses are assigned to these servers so that they can communicate directly with each other locally. Additionally, they are divided into three groups of five, with a public IP address used by each group for inbound/outbound Internet traffic. Inbound traffic is strictly limited to a few ports needed for remote access and the DTN frameworks. Users have the ability to gain access by means of an IPSec VPN, which is configured on a hardware firewall. Each server constitutes a standalone DTN node implementing the full DTN stack. In certain scenarios, where more DTN nodes are needed, the testbed core can be extended by employing a number of virtual machines. For this purpose a high-performance computer is used, featuring two hexa-core Intel Xeon CPUs, 24GB of RAM and 12TB of redundant storage. The high-performance computer runs a bare-metal hypervisor, VMware vSphere, which sets up the virtualization layer. This makes for a scalable testbed core capable of accommodating more than 35 nodes, enough to emulate most space missions.

Portable Satellite Simulator (PSS) - Portable Satellite Simulator Mark III (PSS) [5] is a generic PC based system capable of injecting telemetry into the downlink chain of a ground station and receiving telecommands from the uplink chain, complying with CCSDS recommendations and European Cooperation for Space Standardisation (ECSS) and ESA standards. PSS offers several monitoring and control interfaces and a maintenance interface, which allows controlling and monitoring the PSS locally at the ground station or remotely from the control center. PSS is deployed in the testbed as a state-of-the-art hardware satellite model, incorporating the link layer protocol stack of a real satellite.

CORTEX Command Ranging and Telemetry System (CORTEX CRT) - CORTEX CRT [6] is a state-of-the-art Telemetry and Telecommand base-band COTS. CORTEX CRT system allows a continuous improvement of the signal processing and supports future standards through telemetry processing, CCSDS telecommanding processing, ranging measurements etc. In essence, CORTEX CRT is able to decode and process telemetry data received from a satellite through an antenna and encode telecommand data transmitted to a satellite. CORTEX CRT has field-proven compatibility with most of satellites, high level of reliability with no preventive maintenance, and has been extensively used by many space agencies, including NASA, ESA and JAXA. Within SPICE testbed, CORTEX CRT emulates the functionality of a real ground station collecting and transmitting data from/to satellites.

HellasSat 2 - A satellite link over HellasSat 2 has been set up at the premises of SPICE for the evaluation of Bundle Protocol (BP [13]) over a real satellite link, subject to errors and disruptions due to weather conditions.

2.3 Software Components

DTN Implementations - Interplanetary Overlay Network (ION) [12] is an implementation of the DTN architecture developed by Jet Propulsion Laboratory (JPL) and released as open source software. It includes implementations of the DTN Bundle Protocol, the Licklider Transmission Protocol (LTP [14]), Bundle Security Protocol (BSP [15]), and two CCSDS application protocols that have been adapted to run over the BP/LTP stack: class-1 (unacknowledged) CCSDS File Delivery Protocol (CFDP [16]) and Asynchronous Message Service (AMS [17]). ION is the key DTN implementation of SPICE testbed, since it has been specifically designed for delay-tolerant space communications. Several protocols that have been developed by researchers of the Space Internetworking Center have been already incorporated in the latest ION release, and other are planned to be released in the following versions. **DTN2** [18]**,** which is the reference implementation of the DTN architecture, and **IBR-DTN** [19], an implementation of the Bundle Protocol designed for embedded systems and smartphones are also included in the SPICE Testbed, mainly for interoperability testing purposes.

CCSDS File Delivery Protocol (CFDP) [16] **-** The ESA CFDP ground segment implementation provides a full implementation of the CCSDS File Delivery Protocol. ESA's CFDP provides a Java library and a daemon implementation for reliable and unreliable file transfer in space and on the ground. This implementation will be used by the European Space Agency on ground for upcoming space missions.

SIMSAT - SIMSAT [20] is a general-purpose real-time simulation infrastructure developed for ESA. SIMSAT supports standard simulation services such as cyclic and event-based real-time scheduling of models, logging of simulation events etc. The SIMSAT User Interface is used to coordinate experiments that utilise the Portable Satellite Simulator.

Satellite Tool Kit (STK) - STK [7] is an off-the-shelf mission modeling and analysis software for space, defense and intelligence systems and is used as an external component to the DTN testbed. STK Professional Edition is used to create and manage high-level objects (satellites, aircraft, facilities, etc.), propagate and orient vehicles, analyse relationships between objects, visualise objects in 2D and 3D and animate in real or simulated time. With STK Communications detailed transmitter and receiver elements with antenna pointing are defined, direct or bent pipe communication links over time are analysed and link budget analysis of each communication link is performed, contact periods among communicating elements are calculated, accidental/intentional jamming effects are analysed etc. Finally, Integration Module integrates with other applications in order to develop custom applications to automate repetitive tasks from outside of the application. Information like bandwidth, error rates, propagation delay, disruption periods and connectivity schedule constitute network parameters are imported to SPICE testbed prior to each experiment. Several scenarios have been implemented so far both for satellite and deep-space communication experiments.

Netem - The netem tool [21], which is included in recent Linux kernel versions (2.6+), is used to alter networking properties and emulate variable delay, loss, duplication and re-ordering.

3 Protocols Designed and Evaluated

SPICE testbed is an ideal platform to evaluate different DTN implementations, protocols, applications and services. Its architecture and components have already contributed to the design, implementation, and optimization of novel algorithms and protocols, with respect to the challenging conditions of satellite and space communications. [22-25]. Moreover, SPICE testbed has been used as the key testing platform in several European and ESA funded projects including FP7 SPICE, FP7 Space Data Routers [26], ESA's Extending Internet Into Space, ESA's BitTorrent study and more. In the following subsections, we briefly present sample works, along with experimental results obtained with SPICE testbed.

3.1 Delay-Tolerant Payload Conditioning

Delay-Tolerant Payload Conditioning (DTPC) protocol [27] is an end-to-end transport protocol that was designed in collaboration with NASA's Jet Propulsion Laboratory and is used on top of the Delay-Tolerant Networking (DTN) architecture, offering services such as controlled aggregation of application data units (ADUs) into DTPC *data items* with application-specific elision, end-to-end reliability, in-order delivery and duplicate suppression. A thorough evaluation procedure was developed to assure the correct functionality of the services offered by the DTPC protocol using up to 4 DTN nodes of SPICE testbed, as well as Netem to emulate various link properties (i.e. propagation delay, data rate and error rate). DTPC protocol was implemented into the ION-DTN software platform, along with two test applications for sending and receiving data, respectively. The evaluation scenarios are space-oriented in that propagation delays are in the order of seconds and contact times are scheduled. Therefore, LTP protocol was used as a convergence layer protocol in all bundle nodes. The time-sequence graph depicted in Fig. 3 tracks the delivery of ADUs at the receiver and shows that DTPC successfully retransmits items that were lost or considered lost due to loss of ACKs. In the latter case, the duplicate items are recognized as such and are discarded. DTPC protocol has been included in the official ION-DTN distribution since version 3.2.0.

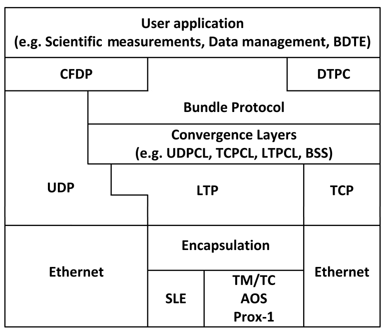

Fig. 2. SPICE testbed protocol stack

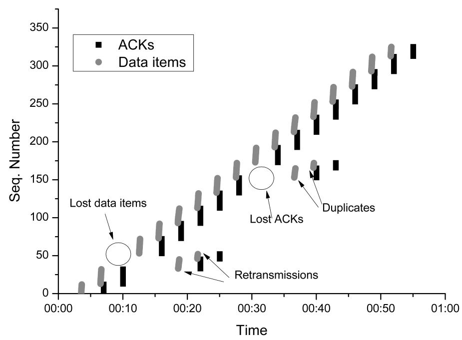

Fig. 3. DTPC - Evaluation of end-to-end reliability. [27]

3.2 Bundle Streaming Service

Bundle Streaming Service (BSS) [28] is a framework that supports the delivery of streaming media in DTN bundles. It exploits the characteristics of such networks to allow for reliable delay-tolerant streaming while improving the reception and storage of data streams. BSS is a joint work between Space Internetworking Center and Jet

Propulsion Laboratory, NASA. It is incorporated into ION-DTN, along with two test applications for sending and receiving data, respectively, using the BSS framework. The extensive evaluation process of BSS mechanism was solely based on SPICE DTN testbed. In particular, several Space network configurations were emulated under different sets of network protocol stacks under variable propagation delays, high error rates, both symmetric and asymmetric network topologies in terms of communication link properties, as well as different number of communicating nodes and bundle sizes. The acquired results, presented in Fig. 4, show that BSS framework clearly outperforms ION's default forwarding mechanism (ipnfw) over the entire set of experiments.

3.3 Delivery Time Estimation for Space bundles

Bundle Delivery Time Estimation (BDTE) tool [29] is an application designed to work in an administrative framework and provide accurate predictions for end-to-end data delivery delays. It exploits a novel algorithmic method, based on the Contact Graph Routing algorithm [30], which predicts bundle route and calculates plausible arrival times along with the corresponding probabilities, based on measurements taken in network nodes and disseminated using DTN management protocol. The application development, as well as the design and implementation of the experimental database for supporting the application were developed in SPICE testbed and were included into the ION-DTN platform, with a plan to be released under the open source license. Fig. 5 depicts a sample BDTE application output, with the possible end-to-end arrival times of a bundle and the cumulative probability distribution.

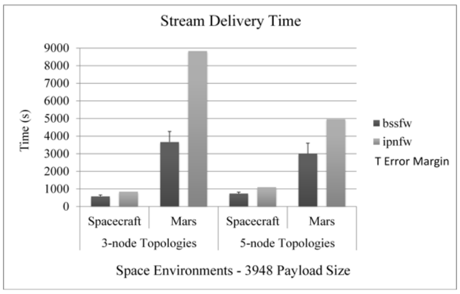

Fig. 4. Comparison between BSS and IPN forwarder based on stream delivery time of a representative sample of cases from Space scenarios. [28]

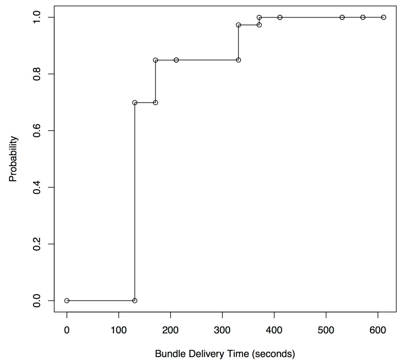

Fig. 5. BDTE output: Bundle delivery times and corresponding probabilities. [29]

4 Related Work

Given the specialised nature of space DTN communications, only a few testbeds have been developed on the field. The TATPA testbed [31] was one of the first proposals to integrate DTN in a satellite emulator. The authors of [32] have built a basic DTN testbed based on a space link simulator (SLS) and performed relay operations. This testbed consists of three nodes only and, thus, can only emulate small-scale scenarios. Similarly, Muri and McNair [33] have emulated a single optical flight terminal that relays data between two ground stations, focusing however on providing a high-capacity optical channel. As an extended approach, DTNbone [34] has been initiated as an interoperability-testing platform and consists of a collection of nodes worldwide running DTN bundle agents and applications. Given its distributed nature, each node of DTNBone is managed by the owner organisation and does not facilitate extensive testing. One of the most extended testbeds, namely the DTN Engineering Network (DEN [35]), has been developed by NASA and comprises physical and virtual machines and flight-like hardware located at different NASA centers and supporting universities. The DEN is configured with reference implementations of advanced communication protocols, including the Bundle Protocol and Licklider Transmission Protocol, and has been used by NASA to validate both software implementations and decentralized operational concepts. SPICE testbed constitutes a European alternative to DEN that provides an extended DTN experimental platform. In contrast to the closed DEN, SPICE testbed is widely available to the research community for validation and evaluation of existing protocols, as well as deployment and experimentation of newly developed mechanisms in a reliable, efficient DTN testbed.

5 Conclusions

Building a DTN testbed for satellite and space communications is a challenging task. Careful design is required to identify and fulfill the distinct requirements of present and future space missions. In this paper, we have presented a state-of-the-art DTN testbed that is scalable and includes several DTN nodes and specialised components that accurately emulate the whole protocol stack of satellite and space communications, as well as the operation of typical ground stations. SPICE testbed has leveraged the development and evaluation of novel protocols and mechanisms for space DTN. Promising results so far showcase the competence of SPICE testbed as an experimental platform with potential to be the first European validation and evaluation tool for future space internetworking protocols. Our aim is to encourage researchers to develop and test new space DTN concepts utilising SPICE testbed.

References

1. Fall, K.: A Delay-Tolerant Networking Architecture for Challenged Internets. In SIGCOMM 2003, pp. 27—34.
2. The Consultative Committee for Space Data Systems (CCSDS), http://public.ccsds.org/default.aspx
3. Koutsogiannis, E., Diamantopoulos, S., Papastergiou, G., Komnios, I., Aggelis, A., Peccia, N.: Experiences from Architecting a DTN Testbed. In: J. Internet Engineering, v3, n1, 219-229. Kleidarithmos Press (2010)

4. EC's FP7 Space Internetworking Center (SPICE) project, http://www.spice-center.org
5. Portable Satellite Simulator (PSS), http://www.spacelinkngt.com/PSSIMBU.html
6. CORTEX CRT system, http://www.zds-fr.com/en/products/10/satellite-ttc.html
7. Satellite Tool Kit (STK), http://www.agi.com/products/
8. Proximity-1 Space Link Protocol, CCSDS 211.0-R-3.1, (2002)
9. CCSDS Telemetry (TM) Space Data Link protocol Recommendation for Space Data Systems Standards. CCSDS 132.0-B (2003)
10. CCSDS Telecommand (TC) Space Data Link protocol Recommendation for Space Data Systems Standards. CCSDS 232.0-B-2 (2010)
11. AOS Space Data Link Protocol, CCSDS 732.0-B-1, Blue Book, Issue 1 (2003).
12. Interplanetary Overlay Network (ION-DTN), Jet Propulsion Laboratory Ohio University, http://sourceforge.net/projects/ion-dtn/
13. Scott, K., Burleigh, S.: Bundle Protocol Specification. IETF RFC 5050 (2007)
14. Ramadas, M., Burleigh, S., Farrell, S.: Licklider Transmission Protocol. IETF RFC 5326.
15. Symington, S., Farrell, S., Weiss, H., Lovell, P.: Bundle Security Protocol Spec. RFC 6257
16. CCSDS File Delivery Protocol (CFDP) Recommendation for Space Data System Standards. CCSDS 727.0FBF4 (2007)
17. Asynchronous Message Service (AMS). http://trs-new.jpl.nasa.gov/dspace/bitstream/2014/37837/1/05-0923.pdf
18. DTN2 Delay Tolerant Networking Reference Implementation, http://sourceforge.net/project/showfiles.php?group_id=101657
19. Doering, M., Lahde, S., Morgenroth, J., Wolf, L.: IBR-DTN: an efficient implementation for embedded systems. In ACM CHANTS '08. pp. 117–120. New York, ACM, NY.
20. SIMSAT simulator, http://www.terma.com/space/ground-segment/satellite-simulators/
21. Network Emulator, http://www.linuxfoundation.org/collaborate/workgroups/networking/netem
22. Bezirgiannidis N. and Tsaoussidis V.: Packet size and DTN transport service: Evaluation on a DTN Testbed. In ICUMT 2010.
23. Papastergiou G., Bezirgiannidis N., and Tsaoussidis V.: On the Performance of Erasure Coding over Space DTNs. In WWIC 2012, Santorini, Greece, June 6-8, 2012.
24. Lenas S.-A., Burleigh S., and Tsaoussidis V.: Reliable Data Streaming over Delay Tolerant Networks. In WWIC 2012, Santorini, Greece, June 6-8, 2012.
25. Clarke N. L., Katos V., Menesidou S.-A., Ghita B., and Furnell S.: A novel security architecture for a space-data DTN. In WWIC 2012, Santorini, Greece, June 6-8, 2012.
26. FP7 Space-Data Routers for Exploiting Space Data project, http://www.spacedatarouters.eu
27. Papastergiou, G., Alexiadis, I., Burleigh, S., Tsaoussidis, V.: Delay Tolerant Payload Conditioning protocol. J. Computer Networks. In press (2013)
28. Lenas, S.A., Burleigh, S., Tsaoussidis, V.: Bundle Streaming Service: Design, Implementation and Performance Evaluation. Transactions on Emerging Telecommunications Technologies.
29. Bezirgiannidis, N., Burleigh, S., Tsaoussidis, V.: Delivery Time Estimation for Space Bundles. Aerospace and Electronic Systems, IEEE Transactions, Vol.49, No.3, pp. 1897-1910 (2013)
30. Burleigh, S.: Dynamic Routing for Delay-Tolerant Networking in Space. In SpaceOps 2008
31. Caini, C., Firincielli, R., Lacamera, D., Tamagnini, S., Tiraferri, D.: The TATPA Testbed, a Testbed for Advanced Transport Protocols and Architecture performance evaluation on wireless channels. In: Proc. TridentCom, pp. 1-7, Orlando, Florida (2007)
32. Sun, X., Yu, Q., Wang, R., Zhang, Q., Wei, Z., Hu, J., Vasilakos, A.: Performance of DTN protocols in space communications. Wireless Networks, pp. 1–19 (2013)
33. Muri, P., McNair, J.: A performance comparison of DTN protocols for high delay optical channels. In WCNCW 2013, IEEE, pp. 183-188.
34. Delay-Tolerant Network Research Group, DtnBone, http://www.dtnrg.org/wiki/DtnBone
35. Birrane, E., Collins, K., Scott, K.: The Delay Tolerant Networking Engineering Network – Constructing a Cross-Agency Supported Internetworking Testbed. In SpaceOps (2012)