Towards Flexibility and Accuracy in Space DTN Communications

Nikolaos Bezirgiannidis, Fani Tsapeli, Sotiris Diamantopoulos, Vassilis Tsaoussidis

Space Internetworking Center, Democritus University of Thrace
Panepistimioupoli Xanthis
+302541079473

{nbezirgi, ttsapeli, sdiaman, vtsaousi} @ ee.duth.gr

ABSTRACT

Although Interplanetary Telecommunications rely on preconfigured contact schedules to make routing decisions, there is a lack of appropriate mechanisms to notify the network about contact plan changes. In order to fill this gap, we propose and evaluate a framework for disseminating information about queueing delays and link disruptions. In this context, we present such a mechanism, focusing not only on its functional properties, but rather on its impact objectives: to improve accuracy and routing performance. Supportively, we couple this mechanism with a DTN-compatible protocol, namely Contact Plan Update Protocol (CPUP), which implements our dissemination policy. Through simulation of space scenarios we show that accuracy can be significantly improved in all cases while routing performance can achieve a wide range, from minor through to significant gains, conditionally.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design---*Store and forward networks*, C.2.2 [**Computer-Communication Networks**]: Network Protocols---*Routing Protocols*.

General Terms

Management, Performance.

Keywords

Delay Tolerant Networking, Interplanetary Communications, Contact Graph Routing, Dynamic Updates.

1. INTRODUCTION

Interplanetary communications are characterized by long propagation delays and intermittent but scheduled connectivity. In this context, Delay-Tolerant Networking (DTN) has been proposed [1] as a perfect candidate to support communications in Space in the future. Indeed, based on an end-to-end, store-and-forward architecture, DTN can naturally unify intra-planetary, deep space, near-Earth and terrestrial communications, forming a flexible and secure networking overlay.

Despite the deterministic nature of scheduled interplanetary communications there are still cases where unexpected disruptions or dynamic parameters of probabilistic nature such as queueing delay may drift the scheduled communication map. This calls for improved and up-to-date network information. This, in turn, requires enhancements on the routing algorithm to integrate the dynamic network parameters.

Although network dynamics can be observed and communicated centrally by space agencies, long propagation delays may render this information obsolete by the time it reaches remote space assets. A distributed, automated update mechanism, instead, based on information created and disseminated by all relevant space assets, could prove itself more beneficial in Space. This statement can only be justified if such knowledge can lead to more accurate data delivery estimations, improved routing decisions and, consequently, shorter delivery times.

In this context, we propose a dynamic framework to improve accuracy and flexibility in space communications. This framework consists of i) a mechanism that incorporates queueing delay and possible disruptions to calculate the *Earliest Transmission Opportunity* (*ETO*); this allows for determining the shortest route between any two assets, and ii) an update protocol, namely *Contact Plan Update Protocol* (*CPUP*), for the dissemination of knowledge that pertains to dynamic network features and parameter changes.

In this work we provide a first proof of our concept based on extensive simulation tests. The rest of the paper is organized as follows. Section 2 briefly describes the background and surveys the related work, while Section 3 presents the introduced mechanisms. In Section 4 we present the simulation results and discuss the expected impact and, finally, we conclude in Section 5.

CHANTS'13, September 30, 2013, Miami, FL, USA

ACM 978-1-4503-2363-5/13/09…$15.00.
http://dx.doi.org/10.1145/2505494.2505499

2. BACKGROUND AND RELATED WORK

Built around the Bundle Protocol (BP) [2], DTN provides services such as custody-based transmissions and the ability to exploit scheduled or opportunistic contacts in order to create a communication path between the sending and the receiving node even when there is no end-to-end connectivity. Both of these services are applicable to space communications, since space assets that produce scientific data are often constrained in terms of storage and power capacity.

Routing in Delay-Tolerant Networks depends on the connectivity pattern of the networking environment. In this context, a lot of routing algorithms have been proposed by the scientific community to cope with opportunistic contacts. These protocols are usually based on replication [3] or coding schemes [4] while some of them apply utility-based schemes [5] in order to capture nodes' heterogeneity. DTLSR [6] disseminates routing information throughout an area and exploits the underlying stability of a network to select the optimal route between two communicating nodes.

In Space, on the contrary, contacts between assets are predetermined. In this context, Contact Graph Routing [7], the most prominent routing scheme, relies on the deterministic nature of space communications. In particular, all nodes utilize knowledge of both their current state and all scheduled communication contacts to compute plausible routes. Several metrics have been proposed to calculate the optimal path between two assets. Initially, earliest forfeit time was used as a means to maximize contact utilization. In [8], path information is encoded and transmitted as a bundle extension block, effectively reducing the complexity of the algorithm and in [9], Enhanced CGR (ECGR) replaces the earliest-forfeit-time with earliest-arrival-time metric.

Although the aforementioned algorithms make forwarding decisions dynamically, they rely on contact information that may be obsolete or otherwise fail to reflect current network dynamics. Although it is not immediately apparent, the diffusion of such information could contribute to routing significantly. Consider, for example, queuing delay on interplanetary networks. Although the contribution of queueing delay to total delay may be quantitatively negligible, especially when propagation delay is too long, this (minor) inaccuracy may lead to a scheduled deadline failure of a contact opportunity. Thus, even a minor queueing delay may cause severe end-to-end delays. Beyond that, queueing delay may be a significant part of total delay in several scenarios. Clearly, the significance of queueing delay, in any case, should not be underestimated.

Moreover, the prediction of the end-to-end delivery delay in space communication has emerged as a crucial and challenging research problem according to the Space Internetworking Strategy Group (SISG) [10]. Along these lines, authors in [11] propose BDTE, an administrative tool to estimate bundle delivery time, based on both deterministic and stochastic components. BDTE integrates well known schedules and unexpected channel errors that affect significantly end-to-end delivery latency; however, it does not account for queueing delay. In addition, several works propose end-to-end reliability mechanisms for space internetworks. In [12] and [13], the authors investigate the use of reliable end-to-end transport protocols in interplanetary networks whereas in [14] a bundle-layer reliability mechanism is proposed in order to provide end-to-end monitoring. A more accurate prediction of end-to-end latency will allow for the configuration of appropriate timers and, consequently, improve the performance of such mechanisms.

Another issue not addressed by the present routing mechanisms is their reaction on potential link disruptions. Space communications can be significantly affected by solar flairs and unfavorable weather conditions in the atmosphere. Although mission designers are able to mitigate many of these effects by using more robust encoding schemes, sometimes losses are inevitable. Furthermore, it is proved in [15] that in order to maximize data return in transmissions from Mars, the interplanetary link must be designed to operate under the best 80-90% of the weather conditions. A link with the ability to transmit data under worse conditions would necessitate heavier encoding, minimizing the available bandwidth. Although such a design of 90% link availability can often result in contact outages in the order of tens of minutes or even hours [16], recently proposed routing solutions for space communications do not tackle these problems.

In conclusion, lack of appropriate mechanisms that update network nodes with the latest contact information may lead to sub-optimal routes, wrong estimations or, in some cases, even loss of data.

3. CONTACT PLAN UPDATE FRAMEWORK

We enhance CGR framework with a pair of complementary elements that inform network nodes about recent events and perform routing decisions, accordingly. Practically, we introduce a single network parameter, *Earliest Transmission Opportunity* (*ETO*), which is determined by both the queueing delay and link disruptions. In support to our new algorithm called *CGR-ETO*, we propose the *Contact Plan Update Protocol* (*CPUP*) to disseminate the dynamically evolving network metrics through the network.

3.1 CGR-ETO

Contact Graph Routing uses predefined contact schedules that every network node is assumed to have. ECGR, in particular, examines all available paths to destination and concludes on an optimal path based on the earliest arrival time; it does not, however, include queueing delay information in the total delivery latency.

To measure and further incorporate queueing delay into the network contact plan, we define *ETO* as the earliest plausible time within a transmission opportunity to forward a bundle with a certain priority. *ETO* has dual applicability: it measures the queue occupancy in specific outducts, while in parallel it can efficiently encode a link disruption. *ETO* values range from the contact start time, which is set as the default and minimum value, up to the contact end time, which is set as the maximum value. Different priority levels are reflected in corresponding *ETO* parameters for each transmission opportunity. In BP, for example, the three priority levels defined are bulk, normal, and expedited. In our simulations we have used a single priority level, since all produced bundles correspond to the scientific data extracted by a single space asset.

Unlike ECGR, CGR-ETO substitutes the contact start time with *ETO* to calculate the earliest arrival time, during contact graph traversals. This way it incorporates both the most recent available queueing information, for the given bundle priority, and potential notifications about link outages.

With regard to queueing information, *ETO* is updated after routing decisions are made in the local node and/or upon reception of CPUP messages from other nodes. In the former case, when a bundle is routed and inserted in an outbound queue towards a neighboring node, the local node increases the corresponding *ETO* for the specific transmission opportunity during which bundle transmission is expected to occur. The *ETO* parameter is updated for all priority levels equal to or lower than the priority level of the bundle to be transmitted. In the latter case, upon the reception of a CPUP message containing *ETO* information about specific contacts, the local node updates its contact plan accordingly.

In the same way, information about an outage can be encoded by an *ETO* increase. Predicting the level of such increase is challenging since it should correspond to the expected duration of disruption. However, in our initial study here we investigate routing performance assuming that estimation can be either accurate or within a margin of error level. Further studies about the outage distribution and optimization of *ETO* are within our future plans but beyond the scope of this paper.

3.2 Contact Plan Update Protocol

The CPUP Protocol Data Unit (PDU) allows for the efficient integration of multiple update commands within a single payload, as depicted in Table 1. Each command is encoded into a Command Block with the format displayed in Table 2.

Table 1. CPUP PDU Format

Byte 0	**Byte 1**	**Byte 2**	**Byte 3**
Version num.	Number of Command Blocks (SDNV)		
Byte 4	**Byte 5**	**Byte 6**	**Byte 7**
1^{st} Command Block			
...			
Byte 4×*n*	**Byte 4×*n*+1**	**Byte 4×*n*+2**	**Byte 4×*n*+3**
n^{th} Command Block			

Table 2. Command Block Format

Byte 0	**Byte 1**	**Byte 2**	**Byte 3**
Creation Timestamp		Command Expiry (SDNV)	
Byte 4	**Byte 5**	**Byte 6**	**Byte 7**
Command Originator (SDNV)			Command Type
Byte 8	**Byte 9**	**…**	**Byte *n***
Command Parameter 1 (SDNV)		…	Comm. Param. *k* (SDNV)

The CPUP header contains the protocol version followed by the "Number of Command Blocks" field. The latter is represented by a Self-Delimiting Numeric Value (SDNV) format [17] and, therefore, has a variable length. The PDU header is followed by the sequence of Command Blocks. For convenience in representation, "Number of Command Blocks" field is depicted as a three-byte SDNV and each Command Block is shown as a four-byte field.

The Command Block contains all necessary information pertaining to the update command. The "Creation Timestamp" field is used to detect and discard obsolete information: commands with creation time older than the most recent update time for a specific contact are ignored. Additionally, the "Command Expiry" field is used to identify the time after which the information contained in this Command Block is invalid or useless. The node that generated the command is stored in the "Command Originator" field, while "Command Type" is a one-byte field with different codes for adding, deleting, and editing contact and range registrations of the contact plan. The block ends with a sequence of "Command Parameter" fields that carry all the necessary information for the command execution. The number of Command Parameters is specific for each Command Type; for example, "Edit Contact bulk priority ETO" contains "Start Time", "Start Node", "End Node", which are the three necessary fields that uniquely identify the contact, followed by the field "New Value of bulk priority ETO". Given that SDNVs are of variable length, they are represented with different lengths in Table 2.

Since CGR has not yet been standardized and the useful network information might vary depending on the different routing metrics, CPUP can be easily customized to include any parameter field, allowing the use of different routing objectives (*i.e.*, other than earliest delivery time). CPUP is designed to use the DTN protocol architecture and, hence, each PDU is inserted into a single bundle payload, utilizing BP mechanisms in order to forward the updated information.

3.3 CPUP Dissemination Mechanism

CPUP data units can be generated either in an administrative framework, *e.g.*, to initially setup a list of connection schedules, or automatically, triggered by outages or significant changes in queue occupancy. It is essential that the produced command messages are delivered to all network nodes timely, before their validity expires. For this purpose a flooding-based mechanism is utilized to disseminate CPUP data units.

Commands are automatically created when either an outage is detected or the queue occupancy in some contact increases beyond a predefined threshold. The per-contact queue occupancy corresponds to the outbound queue that consists of stored bundles destined to a corresponding neighbor and expected to be transmitted during some specific/given contact.

Commands are generated or updated separately per contact. The most recent information for a specific contact is disseminated to all other neighbors, provided that this information will be delivered timely (*i.e.*, before expiration). CPUP encodes all commands applicable to a specific neighbor into blocks and aggregates them into a CPUP PDU. It subsequently transmits this CPUP PDU to that neighbor at the next communication opportunity.

When a node receives a CPUP data unit, it updates the local contact plan with all applicable commands. Obsolete information, *i.e.*, commands with creation time older than the most recent contact modification, are discarded. The node, then, performs the same flooding procedure for every received command; it checks whether information can arrive before expiration and delivers it to the CPUP engine for aggregation. The initial node that originated the corresponding command, as well as the neighbor that propagated the CPUP data unit containing this command block, are excluded from the flooding process.

According to our design, the granularity of the generated update commands is determined by a threshold level which can be either an absolute time interval or a percentage of the contact duration. In our evaluation we have applied the percentage model; for example, a 1% update threshold level within a 5000s transmission opportunity triggers the generation of update commands each time *ETO* gets 50s greater than the previous *ETO*. The "Expiry Time" of the produced information is the new *ETO*; it will be delivered to the CPUP engine and conditionally forwarded to all neighboring nodes, if the CPUP expected delivery time precedes the new *ETO*. In this way, useless transmissions are restrained. The generation of *ETO* update commands can be also suppressed in nodes that are not responsible for relaying data. Finally, an update threshold level of 100% is associated with no dissemination of queue occupancy information and, hence, queueing delays are calculated based solely on the local forwarding decisions.

4. EVALUATION

We present an initial evaluation of the proposed framework in two reference space topologies, Deep-Space and Near-Earth (both depicted in Fig. 1), where a space asset (Node 1) extracts scientific data *in-situ* and transmits it to the Mission Operation Center (Node 6) via relay nodes 2 and 3, and Ground Stations 4 and 5. The investigated topologies could refer to typical scenarios of Mars and Lunar missions.

In the following two subsections we evaluate the impact of CGR-ETO and CPUP in two respective scenarios. At the first scenario we study how far network knowledge about queueing delays can improve routing performance, whereas at the second scenario we both quantitatively and qualitatively assess the importance of CPUP transmissions in disruptive network conditions. In both scenarios we examine the improvement of routing decisions, in terms of earliest delivery delay, and the accuracy gain in delivery latency estimations. We have performed simulations of 1-week duration each. Some specific scenarios have been enhanced by incorporating the connectivity maps provided by the Satellite Toolkit (STK) [18].

4.1 Impact of Queueing Delay

We compare CGR-ETO and ECGR using end-to-end *Bundle Delivery Delay* (*BDD*) measurements for two different cases, where data production is 50% and 100% of the maximum amount of data that can be forwarded to the network, respectively (*i.e.*, from Node 1 to both relay nodes). We also compare ECGR against CGR-ETO (with ETO thresholds 1%, 5%, and 100%), in terms of both the CPUP overhead they inflict, and the *BDD Prediction Accuracy* (*BDDPredAcc*), using the following metrics:

$$Relative\,Overhead = \frac{Total\ CPUP\ Bytes}{Total\ Data\ Payload}$$

$$BDDPredAcc = \frac{BDD\text{ - }Estimated\ BDD}{BDD}$$

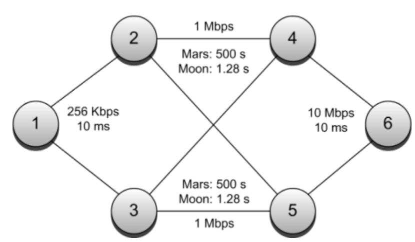

Figure 1. Simulation Topology

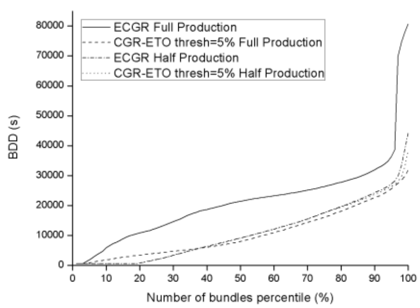

Figure 2. CDF of *BDD*

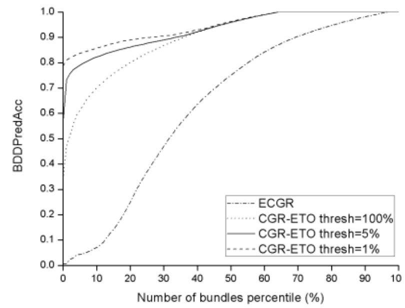

Figure 3. CDF of *BDDPredAcc*

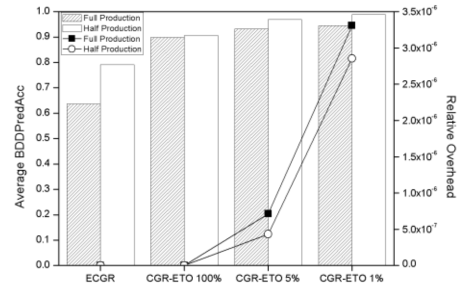

Figure 4. Average *BDDPredAcc* and *Relative Overhead*

Intuitively, we expect that CGR-ETO contribution will be more significant in scenarios with heavy traffic, where queueing delay greatly affects network performance.

The Cumulative Distribution Function (CDF) of *BDD* (Fig. 2) shows that the contribution of CGR-ETO at the light-traffic scenario is relatively small, since queueing delay is a minor portion of the total delivery delay. At the heavy traffic scenario, however, it achieves significant improvement compared to ECGR; indeed, queueing information assists Node 1 in performing more prudent routing decisions, balancing the traffic load between the two relay nodes. Consequently, CGR-ETO efficiently mitigates the effects of intense network load.

In Figure 3 we present the CDF of *BDDPredAcc* for ECGR and CGR-ETO with the three distinct threshold levels, at the scenario with heavy data production rate. CGR-ETO performs 100% accurate predictions of the delivery delay for about 40% of the transmitted data. CGR-ETO with 1% *ETO* threshold, in particular, achieves at least 80% *BDDPredAcc* for *all* bundles. As indicated in Figure 3, incorporating *ETO* in the *BDD* prediction has a substantial effect on the prediction error reduction even without disseminating the *ETO* information (*i.e.*, no CPUP packets are transmitted). In Figure 4 the average *BDDPredAcc* is depicted in conjunction with the overhead imposed by CPUP for both low- and heavy- data production rates. Smaller *ETO* thresholds improve *BDDPredAcc*, with minor increase in the CPUP overhead.

4.2 Impact of Disruptions

We examine the performance of CPUP at the Near-Earth scenario under the presence of unplanned contact disruptions. In this scenario, we assume that the links between Ground Station 4 and relay nodes 2 and 3 suffer from frequent outages, due to severe weather. We assume that CGR-ETO is used to notify nodes about outages and we compare its performance against ECGR. We consider two different cases for CGR-ETO evaluation: the first one corresponds to an ideal scenario, where Node 2 is able to provide accurate knowledge about outage duration, while the second case corresponds to an outage duration predicted with an error of up to 20% of its duration. Although the assumption of accurate knowledge of outage duration may sound unrealistic, it serves here as a reference scenario for the evaluation of CGR-ETO.

In Figure 5 we depict the average *BDD* for the ECGR and CGR-ETO with accurate and inaccurate outage duration prediction, respectively. We consider three distinct cases, where disruptions occur at different times: at the start, the middle, or the end of each contact. Outage occurrence at the start or at the middle point of contacts does not affect CGR-ETO routing decisions and consequently *BDD*, since the selected route is optimal despite its temporal disconnection (see the first six columns in Fig. 5). On the other hand, average *BDD* is affected when disconnections occur at the end of the link, since data waiting for transmission at the specific outduct will not be forwarded until the next contact opportunity. This justifies our claim that diffusion of disruption knowledge may result in improved routing decisions.

In Figure 6 we present the CDF of *BDDPredAcc* for outages that occur at the middle of the contact, with intervals equal to 20% of the transmission window. Figure 7 depicts the average *BDDPredAcc* for outages with varying duration. It is obvious from both figures that CGR-ETO outperforms ECGR when information on disruptions is accurate. What is more important, though, is that there is significant improvement on *BDDPredAcc* even when the estimation of disruption duration is not accurate.

5. CONCLUSIONS

In this work we have proposed a new routing framework that consists of two components, the CGR-ETO algorithm at its core and CPUP as a supporting protocol. Our framework has the potential to improve routing decisions; we demonstrated significant delivery delay reduction in heavy-traffic scenarios and we observed similar behavior in disruptive environments. The improvement is more significant, naturally, when there is an alternative link and occasionally when disruptions appear later in the transmission window. It is worth noticing that delay prediction accuracy undoubtedly improves, irrespectively of

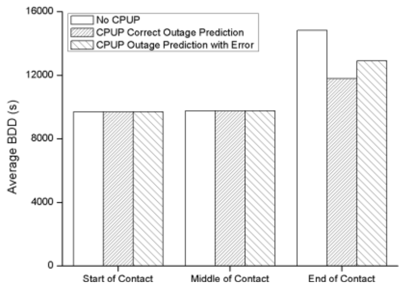

Figure 5. Average *BDD* for 20% outages

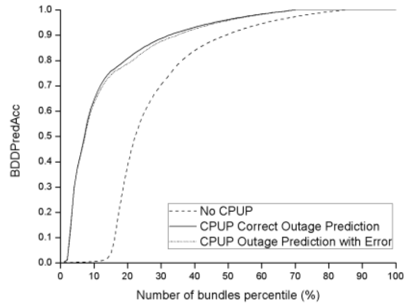

Figure 6. CDF of *BDDPredAcc* for 20% outages

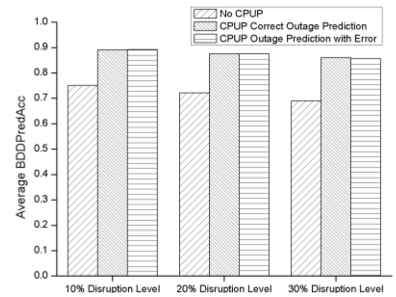

Figure 7. Average *BDDPredAcc* for outage at the middle of contacts

the particular conditions of each scenario. This, alone, justifies the necessity of CGR-ETO and the accompanying protocol, which was the initial claim of this paper.

That said and given the dominant role of DTN in the design of future space missions, we consider this initial work as an important first step towards a robust, unified architecture, which will provide efficient routing and accurate mission planning.

6. ACKNOWLEDGMENTS

The research leading to these results has received funding from the European Community's Seventh Framework Programme [FP7/2007-2013] under grant agreement n° 263330 (project title: SPACE-DATA ROUTERS for Exploiting Space DATA) for authors Tsapeli and Diamantopoulos and under grant agreement n° 264226 (project title: Space Internetworking Center-SPICE) for authors Bezirgiannidis and Tsaoussidis. This paper reflects only the authors' views and the Community is not liable for any use that may be made of the information contained therein.

7. REFERENCES

[1] Fall, K. 2003. A delay-tolerant network architecture for challenged internets. In *ACM SIGCOMM*, pp. 27-34.

[2] Scott, K. and Burleigh, S.. 2007. *Bundle Protocol Specification*. IETF RFC 5050, Internet Engineering Task Force.

[3] Spyropoulos, T., Psounis, K., Raghavendra, C.S. Spray and wait: an efficient routing scheme for intermittently connected mobile networks. In *WDTN* 2005*,* Philadelphia, Pennsylvania, USA, August 26–26, 2005.

[4] F Tsapeli, V Tsaoussidis. Routing for opportunistic networks based on probabilistic erasure coding. In *WWIC* 2012, Santorini, Greece, June 5-7, 2012.

[5] Lindgren, A., Doria, A., Schelén, O. Probabilistic routing in intermittently connected networks. ACM SIGMOBILE Mobile Computing and Communications Review, v.7 n., 3, July 2003.

[6] Demmer, M., and Fall, K. DTLSR: delay tolerant routing for developing regions. In *NSDR* 2007, Kyoto, Japan, August 27–27, 2007.

[7] Burleigh, S. 2010. Contact Graph Routing: Internet Draft, July 2010, http://tools.ietf.org/html/draft-burleigh-dtnrg-cgr-01S.

[8] Birrane, E., Burleigh, S., Kasch, N. Analysis of the contact graph routing algorithm: Bounding interplanetary paths. *Acta Astronautica*, Volume 75, June–July 2012.

[9] *SISG Operations Concept for SSI - final version*, Interagency Operations Advisory Group, Space Internetworking Strategy Group, January 2011.

[10] Seguí, J., Jennings, E., Burleigh, S. Enhancing Contact Graph Routing for Delay Tolerant Space Networking. In *GLOBECOM,* pages 1–6, 2011.

[11] Bezirgiannidis, N., Burleigh, S., and Tsaoussidis, V. 2012. *Delivery Time Estimation for Space Bundles*. Technical Report: TR-DUTH-EE-2012-21, Democritus University of Thrace.

[12] Papastergiou, G., Alexiadis, I, Burleigh, S. 2012. *Delay Tolerant Payload Conditioning*. Technical Report: TR-DUTH-EE-2012-24, Democritus University of Thrace.

[13] Papastergiou, G., Bezirgiannidis, N., Tsaoussidis, V. On the Performance of Erasure Coding over Space DTNs. In *WWIC 2012*, Santorini, Greece, June 5-7, 2012.

[14] E. Koutsogiannis, F. Tsapeli and V. Tsaoussidis. Bundle Layer End-to-end Retransmission Mechanism. In *BCFIC 2011*, Riga, Latvia, February 16-18, 2011.

[15] Noreen, G., Komarek, T., Diehl, R., Shambayati, S., Breidenthal, J., Lopez, S., Jordan, F. 2003. *Mars Telecommunications Orbiter Ka-band system design and operations*. Technical Report. NASA.

[16] Harcke, L. J., Yi, P. F., Sue, M. K., and Tan, H. H. *Recent Ka-Band Weather Statistics for Goldstone and*

Madrid. The Telecommunications and Data Acquisition Progress Report 42-125, May 15, 1996.

[17] Eddy, W., Davies, E. 2011. Using Self-Delimiting Numeric Values in Protocols, draft-irtf-dtnrg-sdnv-09. Internet Draft.

[18] Satellite Toolkit. http://www.agi.com/