

Proposed Recommendation for  
Space Data System Standards

|  |
| --- |
| schedule-aware bundle routing |

proposed Standard

CCSDS 734.3-W-2

white Book

July 2017

AUTHORITY

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
|  | Issue: | White Book, Issue 2 |  |
|  | Date: | July 2017 |  |
|  | Location: | Not Applicable |  |
|  | | | |

**(WHEN THIS RECOMMENDED STANDARD IS FINALIZED, IT WILL CONTAIN THE FOLLOWING STATEMENT OF AUTHORITY:)**

This document has been approved for publication by the Management Council of the Consultative Committee for Space Data Systems (CCSDS) and represents the consensus technical agreement of the participating CCSDS Member Agencies. The procedure for review and authorization of CCSDS documents is detailed in the *Procedures Manual for the Consultative Committee for Space Data Systems*, and the record of Agency participation in the authorization of this document can be obtained from the CCSDS Secretariat at the address below.

This document is published and maintained by:

CCSDS Secretariat

Space Communications and Navigation Office, 7L70

Space Operations Mission Directorate

NASA Headquarters

Washington, DC 20546-0001, USA

FOREWORD

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CCSDS shall not be held responsible for identifying any or all such patent rights.

Through the process of normal evolution, it is expected that expansion, deletion, or modification of this document may occur. This Recommended Standard is therefore subject to CCSDS document management and change control procedures, which are defined in the *Procedures Manual for the Consultative Committee for Space Data Systems*. Current versions of CCSDS documents are maintained at the CCSDS Web site:

http://www.ccsds.org/

Questions relating to the contents or status of this document should be addressed to the CCSDS Secretariat at the address indicated on page i.

At time of publication, the active Member and Observer Agencies of the CCSDS were:

Member Agencies

* Agenzia Spaziale Italiana (ASI)/Italy.
* Canadian Space Agency (CSA)/Canada.
* Centre National d’Etudes Spatiales (CNES)/France.
* China National Space Administration (CNSA)/People’s Republic of China.
* Deutsches Zentrum für Luft- und Raumfahrt e.V. (DLR)/Germany.
* European Space Agency (ESA)/Europe.
* Federal Space Agency (FSA)/Russian Federation.
* Instituto Nacional de Pesquisas Espaciais (INPE)/Brazil.
* Japan Aerospace Exploration Agency (JAXA)/Japan.
* National Aeronautics and Space Administration (NASA)/USA.
* UK Space Agency/United Kingdom.

Observer Agencies

* Austrian Space Agency (ASA)/Austria.
* Belgian Federal Science Policy Office (BFSPO)/Belgium.
* Central Research Institute of Machine Building (TsNIIMash)/Russian Federation.
* China Satellite Launch and Tracking Control General, Beijing Institute of Tracking and Telecommunications Technology (CLTC/BITTT)/China.
* Chinese Academy of Sciences (CAS)/China.
* Chinese Academy of Space Technology (CAST)/China.
* Commonwealth Scientific and Industrial Research Organization (CSIRO)/Australia.
* CSIR Satellite Applications Centre (CSIR)/Republic of South Africa.
* Danish National Space Center (DNSC)/Denmark.
* Departamento de Ciência e Tecnologia Aeroespacial (DCTA)/Brazil.
* European Organization for the Exploitation of Meteorological Satellites (EUMETSAT)/Europe.
* European Telecommunications Satellite Organization (EUTELSAT)/Europe.
* Geo-Informatics and Space Technology Development Agency (GISTDA)/Thailand.
* Hellenic National Space Committee (HNSC)/Greece.
* Indian Space Research Organization (ISRO)/India.
* Institute of Space Research (IKI)/Russian Federation.
* KFKI Research Institute for Particle & Nuclear Physics (KFKI)/Hungary.
* Korea Aerospace Research Institute (KARI)/Korea.
* Ministry of Communications (MOC)/Israel.
* National Institute of Information and Communications Technology (NICT)/Japan.
* National Oceanic and Atmospheric Administration (NOAA)/USA.
* National Space Agency of the Republic of Kazakhstan (NSARK)/Kazakhstan.
* National Space Organization (NSPO)/Chinese Taipei.
* Naval Center for Space Technology (NCST)/USA.
* Scientific and Technological Research Council of Turkey (TUBITAK)/Turkey.
* Space and Upper Atmosphere Research Commission (SUPARCO)/Pakistan.
* Swedish Space Corporation (SSC)/Sweden.
* United States Geological Survey (USGS)/USA.

PREFACE

This document is a proposed CCSDS Recommended Standard. Its ‘White Book’ status indicates that the CCSDS does not yet believe the document to be technically mature and has not released it for formal review by appropriate technical organizations.

Implementers are cautioned **not** to fabricate any final equipment in accordance with this document’s technical content.

Recipients of this proposal are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

DOCUMENT CONTROL

|  |  |  |  |
| --- | --- | --- | --- |
| **Document** | **Title and Issue** | **Date** | **Status** |

CCSDS Schedule-Aware Routing, Feburary Current proposal.

xxx.0-W-0 Proposed Standard, Issue 1 2016

CONTENTS

Section Page

1 Introduction 1-8

1.1 proposed Recommendation 1-8

1.2 Purpose 1-8

1.3 Scope 1-8

1.4 Normative References 1-9

1.5 NOMENCLATURE 1-9

2 Overview 2-11

2.1 General 2-11

2.2 Architectural Considerations 2-12

3 data structures 3-14

3.1 contact plans 3-14

3.2 Routing Tables 3-15

4 Key concepts 4-16

4.1 Expiration time 4-16

4.2 OWLT margin 4-16

4.3 Estimated volume consumption 4-16

4.4 Excluded neighbors 4-16

4.5 Critical bundles 4-17

5 route DETERMINATION procedures 5-18

5.1 Contact Graph Routing 5-18

5.2 Alternative route determination procedures 5-25

5.3 Exception Handling 5-26

6 Acknowledgments 6-28

ANNEX A Informative References 6-29

ANNEX B Security, SANA, and Patent Considerations B-1

Figures

**No table of contents entries found.**

Tables

**No table of contents entries found.**

# Introduction

## proposed Recommendation

This document defines a is currently a PROPOSED Recommendation, meaning that it has not undergone CCSDS Agency review. The procedures defined in this document could change before adoption as a Draft Recommendation by CCSDS, or not be adopted by CCSDS at all.

## Purpose

This document defines a Proposed Standard for Schedule-Aware Bundle Routing (SABR) in the forwarding of Delay-Tolerant Networking (DTN) bundles in the space environment. SABR provides dynamic route computation in an environment of stable topology but time-varying connectivity when instances of connectivity are scheduled rather than opportunistic.

## Scope

SABR is intended for use in operation of the Bundle Protocol (BP) in the Solar System Internet.

### Organization of the Recommendation

This Recommendation is organized as follows:

TBS

### Definitions

#### Definitions from OSI Basic Reference Model

SABR is not an on-the-wire protocol but rather a procedure for computing paths over which to exercise on-the-wire protocols. As such, definitions from the OSI Basic Reference Model are not applicable.

#### Definitions from Open Systems Interconnection (OSI) Service Definition Conventions

SABR is not an on-the-wire protocol but rather a procedure for computing paths over which to exercise on-the-wire protocols. As such, definitions from the OSI service Definition Conventions are not applicable.

#### Terms Defined in This Recommendation

Within the context of this document the following definitions apply:

A *bundle* is a unit of data transmitted via the DTN bundle protocol from one DTN node (termed the bundle’s *source*) to another (termed the bundle’s *destination*).

Each bundle comprises a primary block, zero or more extension blocks, and a payload block. We here define the *header* of a bundle as the concatenation of the primary block, all extension blocks that precede the payload block, and the block header of the payload block itself. We defined the *payload* of a bundle as the content of the bundle’s payload block.

## Normative References

The following documents contain provisions which, through reference in this text, constitute provisions of this Proposed Standard. At the time of publication, the editions indicated were valid. All documents are subject to revision, and users of this Recommended Standard are encouraged to investigate the possibility of applying the most recent editions of the documents indicated below. The CCSDS Secretariat maintains a register of currently valid CCSDS Recommended Standards.

[1] *CCSDS Bundle Protocol Specification*. Recommendation for Space Data System Standards, CCSDS 734.2-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, September 2015.

[2] S. Burleigh. *Compressed Bundle Header Encoding*. RFC 6260. Reston, Virginia: ISOC, May 2011.

[3] T. Berners-Lee, R. Fielding, L. Masinter. *Uniform Resource Identifier (URI): Generic Syntax*. RFC 3986. Reston, Virginia: ISOC, January 2005.

## NOMENCLATURE

### NORMATIVE TEXT

The following conventions apply for the normative specifications in this Proposed Standard:

1. the words ‘shall’ and ‘must’ imply a binding and verifiable specification;
2. the word ‘should’ implies an optional, but desirable, specification;
3. the word ‘may’ implies an optional specification;
4. the words ‘is’, ‘are’, and ‘will’ imply statements of fact.

These conventions do not imply constraints on diction in text that is clearly informative in nature.

### INFORMATIVE TEXT

In the normative sections of this document, informative text is set off from the normative specifications either in notes or under one of the following subsection headings:

* Overview;
* Background;
* Rationale;
* Discussion.

# Overview

## General

The Bundle Protocol specification [1] defines procedures for forwarding data *bundles* through a delay-tolerant network such as the Solar System Internet. However, it intentionally omits definition of procedures for determining which nodes of the network to forward a given bundle through in order to ensure that it reaches its destination, leaving those specifications to other documents that are targeted for various network environments requiring a variety of specialized algorithms. One such environment is the infrastructure used to conduct space flight missions. Schedule-Aware Bundle Routing (SABR) is designed for that environment.

In the Internet, protocol operations can be largely driven by currently effective information that is discovered opportunistically and immediately, at the time it is needed, because the latency in communicating this information over the network is negligible: distances between communicating entities are small and connectivity is continuous. In a space flight mission environment, however, ad-hoc information discovery would in many cases take so much time that it could not be completed before the information lost currency and effectiveness. Instead, protocol operations must be largely driven by information that is pre-placed at the network nodes and tagged with the dates and times at which it becomes effective.

More specifically, the forwarding of bundles through a DTN-based flight mission network differs in several ways from the forwarding of packets through an IP-based network. In an IP-based network:

* Connectivity – the ability of topologically adjacent ("neighboring") network nodes to exchange packets – is typically continuous throughout the network. Lapses in connectivity are anomalous and may be interpreted as changes in topology.
* Signal propagation latencies are very small.
* Together, these characteristics ensure that the rate at which information on changes in connectivity may be propagated through the network far exceeds the rate at which those changes occur.

A flight mission network based on DTN is different:

* There is no expectation of continuous connectivity throughout the network. Lapses in connectivity may be routine, lengthy, and recurring; they should not be interpreted as changes in topology.
* Signal propagation latencies may be large.
* Together, these characteristics imply that the rate at which information on changes in connectivity may be propagated through the network may be far lower than the rate at which those changes occur.

Because of these differences, the constraints within which forwarding routes are computed in a DTN-based network are different from those within which IP routes are computed, so route computation procedures must be different.

In particular, IP routing at each router can be based on a local understanding of current connectivity in the network that may be assumed to be generally accurate and generally stable over time. The route to a given destination host, once computed, may be stored in a routing table for future reference and will only need to be changed upon the arrival of new connectivity information – conveyed by routing protocol messages generated immediately in response to detected changes in connectivity – that invalidates that route.

DTN routing enjoys no such advantages. The potential delay in the arrival of information regarding connectivity changes makes all such information potentially obsolete; a Bundle Protocol Agent (BPA) that relied solely on this flow of information might never have a fully accurate understanding of current connectivity in the network.

Yet BPAs that must compute routes in a DTN-based network have no alternative but to rely on that understanding, imperfect as it may be. Each BPA must therefore augment its model of connectivity in the network by other means. Some elements of the model may simply be asserted by network management, i.e., as static routes. Some changes in proximate DTN network connectivity may be discovered in real time. Other connectivity changes may be predicted on a probabilistic basis.

Schedule-Aware Bundle Routing is designed for use in networks where changes in connectivity are planned and scheduled, rather than predicted, discovered, or contemporaneously asserted.

Scheduled changes in connectivity characterize a number of potential DTN application environments:

* Episodes of communication between robotic spacecraft in interplanetary space and ground tracking stations on Earth are typically scheduled weeks or months before they occur.
* The beginning and end of each communication opportunity between an orbiting spacecraft and a communication asset on a planetary surface – either Earth or another planet – can readily be computed from known orbital elements.
* Power-conserving motes of sensor webs may communicate on infrequent, fixed intervals established by network configuration.

Where changes in connectivity are scheduled, a global "contact plan" of all such events may be distributed in advance to all BPAs, enabling each BPA to have a theoretically accurate understanding of connectivity in the network at any specified moment. The Schedule-Aware Bundle Routing procedures compute bundle forwarding decisions from this time-varying model of network connectivity.

## Architectural Considerations

As discussed in the Bundle Protocol (BP) specification [1], the source and destination of each bundle are BP endpoints, identified by BP endpoint ID (EID) strings that are Uniform Record Identifiers (URIs) [3].

However, the actual agents of bundle origination, forwarding, and delivery are instances of Bundle Protocol procedures implementation (bundle protocol agents) that are installed at physical computational entities termed "nodes".

For bundle forwarding purposes, a BP endpoint only exists so long as at least one node is "registered" in that endpoint: only the operation of the BPA at a node can cause a bundle to be delivered to an endpoint, and a BPA can only deliver a bundle to an endpoint within which that BPA's host node is registered.

An endpoint in which only a single node may be registered at any time is termed a “singleton” endpoint. The forwarding of a bundle to a singleton endpoint is functionally equivalent to “unicast” transmission in the Internet and is the most familiar and widely implemented mode of network communications.

No specifications yet exist that would govern the forwarding of a bundle to a non-singleton endpoint, e.g., “multicast” transmission, so for the purposes of this document only bundle “unicast” transmission is considered; SABR is not applicable to the forwarding of bundles to non-singleton endpoints. So for the purposes of this document, the existence of a node may be regarded as a precondition for the existence of an endpoint, and arrival of a bundle at some node's BPA is a precondition for the delivery of that bundle to an endpoint.

Moreover, it is not unusual for a single node to be registered in multiple endpoints, each serving the needs of a different DTN application operating at that node. When this is the case, the arrival of a bundle at some single BPA is a precondition for the delivery of that bundle to any of a potentially large number of (singleton) endpoints.

For these reasons, the design of SABR is based on the concept of forwarding each bundle to the sole node that is registered in the bundle's destination endpoint (rather than directly to the destination endpoint), leaving to the node's BPA the task of final delivery.

Execution of this concept requires that nodes be recognized as first-class BP architectural elements, which must be uniquely identified in order to ensure accurate bundle delivery to the correct destination endpoints. SABR assumes that all nodes in the network are identified by unique "node numbers" as discussed in the specification for Compressed Bundle Header Encoding (CBHE) [2]. When SABR is used to forward a bundle to an endpoint identified by a CBHE-conformant EID, the destination node number can simply be extracted from the EID. SABR can also be used to forward bundles to an endpoint identified by a non-CBHE-conformant EID, but only if that EID can somehow be mapped to the appropriate destination node number; mechanisms for accomplishing this mapping are beyond the scope of this specification.

NOTE – Note again that the design of SABR precludes its use for routing a bundle to a "multicast" endpoint: by definition, a multicast endpoint ID cannot be mapped to the number identifying a single node. SABR can only be used for forwarding bundles to "singleton" endpoints.

# data structures

## contact plans

The basic strategy of SABR is to take advantage of the fact that, since communication operations are planned in detail, the communication routes between any pair of bundle protocol agents in a population of DTN nodes that have all been informed of one another’s plans can be inferred from those plans rather than discovered via dialogue (which is impractical over long-one-way-light-time space links). This information takes the form of *contact plans*.

A contact plan comprises two types of information items: *contacts* and *range intervals*.

A *contact* is here defined as an interval during which it is expected that data will be transmitted by DTN node A (the contact’s sending node) and most or all of the transmitted data will be received by node B (the contact’s receiving node). Implicitly, the sending node will utilize some “convergence-layer” protocol underneath the Bundle Protocol to effect this direct transmission of data to the receiving node. Each contact is characterized by its start time, its end time, the identities of the sending and receiving nodes, and the rate at which data are expected to be transmitted by the sending node throughout the indicated time period.

The *duration* of a contact is given by subtracting the contact’s start time from its end time.

The *volume* of a contact is product of its duration and its data transmission rate.

NOTE – A contact is specifically not an episode of activity on a link. Episodes of activity on different links – e.g., different radio transponders operating on the same spacecraft – may well overlap, but contacts by definition cannot; they are bounded time intervals and as such are innately “tiled”. For example, suppose transmission on link X from node A to node B, at data rate RX, begins at time T1 and ends at time T2; also, transmission on link Y from node A to node B, at data rate RY begins at time T3 and ends at time T4. If T1 = T3 and T2 = T4, then there is a single contact from time T1 to time T2 at data rate RX + RY. If T1 < T3 and T2 = T4, then there are two contiguous contacts: one from T1 to T3 at data rate RX, then one from T3 to T2 at data rate RX + RY. If T1 < T3 and T3<T2 < T4, then there are three contiguous contacts: one from T1 to T3 at data rate RX, then one from T3 to T2 at data rate RX + RY, then one from T2 to T4 at data rate RY. And so on.)

A *range interval* is a period of time during which the displacement between two nodes A and B is expected to vary by less than 1 light second from a stated anticipated distance. (We expect this information to be readily computable from the known orbital elements of all nodes.) Each range interval is characterized by its start time, its end time, the identities of the two nodes to which it pertains, and the anticipated approximate distance between those nodes throughout the indicated time period, to the nearest light second.

Protocols for distributing contact plan information to bundle protocol agents are beyond the scope of this specification.

## Routing Tables

Each node uses the Range Intervals and Contacts in the contact plan to build a "routing table" data structure.

The routing table constructed locally by each node in the network is a list of *entry node lists*, one entry node list for every other node D in the network that is cited in any Contact or Range Interval in the contact plan.

A *route* for a bundle whose current location is node X and whose destination is node D is defined as a sequence of contacts such that (a) the sending node for the first contact is X, (b) the receiving node for the last contact is D, (c) the receiving node for contact i is the sending node for contact i+1, and (d) the time at which contact i+1 ends is no earlier than the time at which contact i begins. An implementation’s representation of a route is here termed a *route object*.

The receiving node for the first contact of a route is termed the route’s *entry node*. For any given route, the contact from the local node to the entry node constitutes the initial transmission segment of the end-to-end path to the destination node. Additionally noted in each route object are all of the other contacts that constitute the remaining segments of the route’s end-to-end path.

The *termination time* of a route is the earliest end time among all contacts in the route.

The *volume limit* of a route is the minimum value of volume among all contacts in the route.

Any node that may be the entry node of a route for a bundle whose current location is X is termed a *neighbor* of node X. All, and only, the neighbors of node X are adjacent to X in the time-varying network topology described by the contact plan.

Every bundle has an assigned numerical *priority*. For the purposes of SABR, it is assumed that a bundle may be transmitted to a neighbor only after every bundle with higher priority that is queued for transmission to that same neighbor has been transmitted.

Each route to destination node D, from the local node X, whose entry node is G is referred to as a route to D “through” G.

Each entry in the entry node list for node D is a list of the neighbors of local node X; included with each entry of the entry node list is a list one or more routes to D through the indicated neighbor, termed a *route list*.

# Key concepts

## Expiration time

The *expiration time* of a bundle is computed as its creation time plus its TTL.

NOTE – Every bundle transmitted via DTN has a *time-to-live* (TTL), the length of time after which the bundle is subject to destruction if it has not yet been delivered to its destination. When computing the next-hop destination for a bundle that the local bundle agent is required to forward, there is no point in selecting a route that can't get the bundle to its final destination prior to the bundle’s expiration time.

## OWLT margin

The One-way light time margin (*OWLT margin*) is defined as the maximum delta by which the OWLT between any pair of nodes can change during the time a bundle is in transit between them.

One-way light time (OWLT) – that is, distance – is obviously a factor in delivering a bundle to a node prior to a given time. OWLT can actually change during the time a bundle is en route, but route computation becomes intractably complex if we can't assume an OWLT "safety margin" – a maximum delta by which OWLT between any pair of nodes can change during the time a bundle is in transit between them.

The OWLT delta is necessarily mission-specific, but in practice it may be simplest to assume a worst-case constant. For example, as of the date of publication of this document we might posit that the maximum rate of change in distance between any two nodes in the network is 450,000 miles (720,000 km) per hour, which is 125 miles (200 km) per second. (This is the projected maximum speed of the Solar Probe Plus spacecraft, planned for launch in 2018.)

At this speed, the distance between any two nodes that are initially separated by a distance of N light seconds will increase by a maximum of 125 miles (200 km) per second of transit. This will result in data arrival no later than roughly (N + Q) seconds after transmission – where the OWLT margin value Q is (125 \* N) divided by 186,000 – rather than just N seconds after transmission as would be the case if the two nodes were stationary relative to each other. When computing the expected time of arrival of a transmitted bundle we simply use N + Q, the most pessimistic case, as the anticipated total in-transit time.

## Estimated volume consumption

The size of a bundle is the sum of the sizes of its payload and its header, but bundle size is not the only lien on the volume of a contact. The total *estimated volume consumption* (or “EVC”) for a bundle is the sum of the sizes of the bundle’s payload and header and the estimated convergence-layer overhead. For a bundle whose header is of size M and whose payload is of size N, the estimated convergence-layer overhead is defined as 3% of (M+N), or 100 bytes, whichever is larger.

## Excluded neighbors

A neighboring node C that refuses “custody” (as described in [1]) of a bundle destined for some remote node D is termed an *excluded neighbor* for (that is, with respect to computing routes to) D. So long as C remains an excluded neighbor for D, no bundles destined for D will be forwarded to C – except that occasionally (once per lapse of the round-trip time (RTT) between the local node and C) a custodial bundle destined for D may be forwarded to C to “probe” the link, i.e., to test whether or not the neighbor must remain excluded. (Bundles that are forwarded under such circumstances are termed *probe bundles*.) C ceases to be an excluded neighbor for D as soon as it accepts custody of a bundle destined for D.

## Critical bundles

A *Critical bundle* is one that has the Critical quality-of-service flag set, notionally because it absolutely has got to reach its destination and, moreover, has got to reach that destination as soon as is physically possible.

For an ordinary non-Critical bundle, the SABR dynamic route computation algorithm uses the routing table to select a single neighboring node to forward the bundle through. It is possible, though, that due to some unforeseen delay the selected neighbor may prove to be a sub-optimal forwarder: the bundle might arrive later than it would have if another neighbor had been selected, or it might not even arrive at all.

For Critical bundles, the SABR dynamic route computation algorithm causes the bundle to be inserted into the outbound transmission queues for transmission to all neighboring nodes that can plausibly forward the bundle to its final destination. The bundle is therefore guaranteed to travel over the most successful route, as well as over all other plausible routes. Note that this may result in multiple copies of a Critical bundle arriving at the final destination.

# route DETERMINATION procedures

The neighboring node(s) to which a bundle whose payload’s size is S and whose priority is P shall be forwarded from node X in order to arrive at node D shall be determined as follows.

## Contact Graph Routing

### Contact graph

The contact graph for node D at node X is a conceptual directed acyclic graph comprising:

* At the root vertex, a notional contact from node X (the local node) to itself.
* A terminal vertex, a notional contact from node D to itself.
* One additional vertex for each contact in the contact plan that signifies transmission either directly “to” node D or indirectly to node D (i.e., to the “from” node of some other contact that signifies transmission directly or indirectly to node D) **and** either directly “from” node X or indirectly from node X (i.e., from the “to” node of some other contact that signifies transmission directly or indirectly from node X).
* An edge between two vertices wherever one vertex corresponds to a contact signifying transmission “to” some node (the origin of the edge) and the other vertex corresponds to a contact signifying transmission “from” that same node (the termination of the edge).

NOTE – The structure of the contact graph may seem somewhat counterintuitive; it bears almost no relation to the topology of the network, a more familiar graph. The vertices of the graph correspond to **contacts**, not to nodes. Contacts are episodes of data transmission from one node to another. The root vertex may be thought of as corresponding to bundle creation, the conveyance of bundle content “from” the application “to” the BPA, both at the source node. The terminal vertex may be thought of as corresponding to bundle delivery, the conveyance of bundle content “from” the BPA “to” the application, both at the destination node. The edges are episodes of data retention at a node. That is, the vertex for contact 1 (transmission of data from node A to node B) is connected to the vertex for contact 2 (transmission of data from node B to node C) by an edge indicating a period of data retention at node B while node B waits for contact 2 to start. See Figure 5‑1, Figure 5‑2, and Figure 5‑3 for an illustration.

![](data:None;base64,)

Figure 5‑1 Network Topology Example

![](data:None;base64,)

Figure 5‑2 Contact Plan Example: contacts

![](data:None;base64,)

Figure 5‑3 Contact Plan Example: range intervals

![](data:None;base64,)

Figure ‑4 Node A’s Contact Graph for Node D, given this contact plan.

### Contact Plan check

If no contacts in the contact plan identify transmission to node D, then the contact graph routing (CGR) procedures shall not be used to select the neighboring node(s) to which this bundle shall be forwarded. The alternative route determination procedures defined in section 5.2 shall be used instead.

### Route Pruning

If the contact plan has been modified in any way since the route computation procedures were most recently performed, all route lists for all nodes through all neighbors shall be discarded.

NOTE – Contact plan changes may invalidate any or all earlier route computations.

Every contact whose end time is in the past shall be deleted from the contact plan (and therefore, implicitly, from the contact graphs for all nodes). Every route (in every route list) that includes one or more of these deleted contacts shall be deleted.

Each route whose volume limit is less than the sum of the estimated volume consumptions of all bundles for which the route’s entry node has been selected as the *most preferred neighbor* for forwarding (as discussed later), since instantiation of the route, shall be deleted.

NOTE – As soon as the maximum amount of data that can be conveyed on the route has been allocated, the route can no longer be considered in routing decisions. It may be reasonable to queue additional bundles for transmission to the same neighbor, but only if they can be forwarded on a different route.

### Route computation

For each of the local node’s neighbors that has an empty list of routes to node D, a new route list for node D shall be computed.

The first route in this list shall be computed by using Dijkstra’s algorithm to find the lowest-cost path through the relevant projection of the contact graph for node D, beginning at the root of the graph and ending at the terminal vertex.

The relevant projection of the contact graph for node D shall be the entire contact graph for node D except for all contacts from the local node to any node other than the neighbor for which the route list is being computed.

The following definitions constrain SABR route computation:

1. The *earliest transmission time* for a contact from the local node to one of its neighbors is defined as the start time of that contact or the current time, whichever is later. The earliest transmission time for any other contact is defined as the start time of that contact or the earliest arrival time (defined below) for the immediately preceding contact in the route, whichever is later. No contact whose end time is before its earliest transmission time (i.e., before the earliest arrival time for the preceding contact in the route under consideration) shall be included in a route.
2. The *earliest arrival time* for a contact is pessimistically defined as the sum of the earliest transmission time for that contact plus the range in light seconds from the contact’s sending node to its receiving node, plus the applicable one-way light time margin.

For the purposes of the Dijkstra search, the cost of edge N in the graph shall be computed as the 3-tuple {earliest arrival time of K, volume of K, start time of K} where K is the contact that is the vertex in which edge N terminates. The cost of edge N shall be judged to be less than the cost of edge M if:

* the earliest arrival time of contact K is less than the earliest arrival time of the contact that is the vertex in which M terminates, or
* the two earliest arrival times are the same but the volume of contact K is greater than that of the contact in which M terminates, or
* the two earliest arrival times and volumes are the same but the start time of contact K is less than that of the contact in which M terminates.

The *best-case delivery time* characterizing a route is defined as the earliest arrival time for the contact that immediately precedes the terminal vertex contact in this route.

NOTE – At the conclusion of route computation a route list for node D through each neighbor of the local node exists, but any such list may contain no routes.

Additional routes may by computed for the route list of a given neighbor by applying Yen’s K Shortest Path algorithm. The details of this procedure are beyond the scope of this specification. When a route list contains multiple routes, the routes shall be ordered by ascending best-case delivery time and, where best-case delivery times are equal, by descending volume limit.

### Route list check

If no neighbor of the local node has a non-empty route list for node D, the CGR procedures shall not be used to select the neighboring node(s) to which this bundle shall be forwarded. The alternative route determination procedures defined in section 5.2 shall be used instead.

### CGR preparation

The conceptual list of nodes to which the bundle may be forwarded, here termed the *proximate nodes list*, shall initially be empty. This list shall contain one entry for each node that is the entry node of a candidate route (selected as discussed below) that could result in arrival of the bundle at node D. Each entry in this list shall indicate the following properties of the first (i.e., lowest-cost) route in the route list of the indicated node: the ID for the proximate node, the projected bundle arrival time, the number of included contacts, and the termination time.

The conceptual list of nodes to which the bundle must **not** be forwarded, here termed the *excluded nodes list*, shall be populated as follows:

* If the bundle is a non-Critical bundle which was previously forwarded to a node that refused custody (and is now being re-forwarded due to that custody refusal), then *backward propagation* of the bundle (that is, transmission of the bundle back to the node from which it was directly received) is authorized; otherwise, backward propagation of the bundle is not authorized. If backward propagation of the bundle is **not** authorized, then the node from which the bundle was directly received shall be added to this list.
* Every excluded neighbor for node D, for which this bundle would **not** serve as a probe bundle if forwarded to that neighbor, shall be added to this list.

### Populating the proximate nodes list

Consideration of routes on which the bundle might be forwarded shall be subject to the following definitions.

The *earliest transmission opportunity* for a route shall be computed as follows:

* The *adjusted start time* for a contact is defined as the contact’s start time or the current time, whichever is later.
* The *applicable backlog* for the route is the sum of the EVCs of all bundles currently queued for transmission to the route’s entry node whose priority is greater than or equal to that of the bundle that is to be forwarded.
* An *applicable prior contact* for the route is any contact with end time later than the current time that has the same sending and receiving nodes as the route’s initial contact but an earlier start time.
* The *applicable duration* of an applicable prior contact is given by the contact’s end time minus its adjusted start time.
* An *applicable prior contact volume* is defined as the product of data transmission rate and applicable duration for some applicable prior contact.
* The *applicable backlog relief* for the route is the sum of all of its applicable prior contact volumes.
* The *residual backlog* for the route is the applicable backlog minus the applicable backlog relief, or zero, whichever is greater. (This is the projected backlog at the adjusted start time of the route’s initial contact.)
* The *backlog lien* on the route’s initial contact is given by the residual backlog divided by the contact’s data transmission rate.
* The *earliest transmission opportunity* for the route is given by the adjusted start time of the route’s initial contact plus the backlog lien on that contact.

The *first byte transmission time* for the initial contact of a route is defined as the bundle’s earliest transmission opportunity on this route. The first byte transmission time for each subsequent contact on that route is defined as the start time of the contact or the last byte arrival time (as defined below) for the immediately preceding contact in that route, whichever is later.

The *last byte transmission time* for a contact is the contact’s first byte transmission time plus the *applicable radiation latency*, which is given by the EVC of the bundle divided by the contact’s data transmission rate, or the contact’s end time, whichever is earlier.

The *first byte arrival time* for a contact is defined as the first byte transmission time for that contact plus the range in light seconds from the contact’s sending node to its receiving node, plus the applicable one-way light time margin.

The *last byte arrival time* for a contact is defined as the last byte transmission time for that contact plus the range in light seconds from the contact’s sending node to its receiving node, plus the applicable one-way light time margin.

The *projected bundle arrival time* for the route, then, is defined as the computed last byte arrival time for the contact immediately preceding the terminal vertex contact.

Recalling that the bundle protocol agent’s computed routing table contains one entry node list for each destination node (such as D), and that each such entry node list contains one item for each neighboring node, and that each such entry node list item contains the identity of the neighboring node and a list of routes through that neighboring node to the destination node, *candidate* routes on which the bundle might be forwarded shall be selected from the route lists for node D of the local node’s neighbors in the following manner:

* Each route that is not the first (lowest-cost) route in its list shall be ignored.
* Each route whose best-case delivery time is after the bundle’s expiration time shall be ignored.
* Each route whose entry node is a member of the excluded nodes list shall be ignored.
* Each route that includes any contact indicating transmission to node X shall be ignored unless node D and node X are identical (“loopback” transmission).
* Each route for which earliest transmission opportunity is after the end time of the initial contact shall be ignored.
* Each route for which projected bundle arrival time is after the bundle’s expiration time shall be ignored.
* All other routes shall be deemed *candidate routes*.

NOTE – A candidate route may include a contact whose volume is less than the bundle’s estimated volume consumption. In general, inability to transmit a given bundle during any contact with any node other than the entry node cannot ever be accurately anticipated, due to the possibility of transmission backlog (of unknown and unknowable size) at any node. Anticipatory fragmentation, described later, can mitigate this risk somewhat, but in practice we rely on ad-hoc recovery mechanisms to address such routing failures.

The entry node of each candidate route shall be added to the proximate nodes list.

### Proximate nodes list check

If the proximate nodes list contains no nodes, the CGR procedures shall not be used to select the neighboring node(s) to which this bundle shall be forwarded. The alternative route determination procedures defined in section 5.2 shall be used instead.

### CGR forwarding

If the bundle is flagged as a critical bundle, then a copy of this bundle shall be enqueued for transmission to every node in the proximate nodes list.

Otherwise:

* The most preferred neighbor in the proximate nodes list shall be selected as follows:
  + If one of the nodes in this list is associated with a projected bundle arrival time that is earlier than that of all other nodes in the list, then it shall be the most preferred neighbor.
  + Otherwise, if one of the nodes with the earliest projected bundle arrival time is associated with a smaller number of contacts than every other node with the same projected bundle arrival time, then it shall be the most preferred neighbor.
  + Otherwise, if one of the nodes with the earliest projected bundle arrival time and smallest number of contacts is associated with a later termination time than every other node with the same projected bundle arrival time and number of contacts, then it shall be the most preferred neighbor.
  + Otherwise, the node with the smallest node number among all nodes with the earliest projected bundle arrival time and smallest number of contacts and latest termination time shall arbitrarily be chosen as the most preferred neighbor.
* Let Z be the bundle’s estimated volume consumption, and let S be the smallest contact volume among all contacts included in the best candidate route for the most preferred neighbor. If S is greater than or equal to Z, or if the bundle cannot be fragmented, or if anticipatory fragmentation (an implementation option, described below) is not desired, the bundle shall simply be enqueued for transmission to the most preferred neighbor. Otherwise the anticipatory fragmentation procedures shall be performed.

NOTE – An implementation may find it advantageous to note each successfully forwarded bundle in a history list, removing an item from the history list whenever the TTL of the corresponding bundle expires. This will enable a newly received bundle to be discarded immediately if it was previously forwarded, minimizing the incidence of routing loops. Such a mechanism should be used only with great care, however. A bundle that reaches a downstream node from which it cannot be forwarded toward the destination might be forwarded back to an earlier forwarding point from which an alternate route might be taken; upon receiving such a bundle, if the TTL of this bundle is not expired, the earlier forwarder will find that bundle in the history list but even so must not discard it.

#### Anticipatory Fragmentation

NOTE – Transmission opportunity utilization may in some cases be improved by fragmenting a bundle at the time the CGR procedures have computed the bundle’s route. If the best candidate route associated with the most preferred neighbor includes a contact whose volume is less than the size of the bundle, then it can be assumed that the bundle would need to be forwarded in multiple episodes from that contact’s sending node. Since contacts enabling those episodes might or might not be available, while contacts on less theoretically optimal routes from the current forwarding node might be undersubscribed, it may instead be desirable to fragment the bundle immediately and forward the fragments on different routes.

Anticipatory fragmentation shall be performed as follows:

* The bundle shall be fragmented into two fragmentary bundles, bundle A containing the first S octets of the original bundle’s payload and bundle B containing the last (Z – S) octets of that payload.
* Bundle A shall be enqueued for transmission to the most preferred neighbor.
* Route determination shall be performed for bundle B as detailed in this specification, except that the best candidate route for the neighbor for which bundle A was enqueued shall **not** be deemed a candidate route for bundle B or for any fragment of bundle B.

## Alternative route determination procedures

### Routing to neighbor

If it is known (e.g., by management) that bundles can be transmitted directly to node D, then the bundle shall be enqueued for transmission to that neighboring node and no other alternative route determination procedure shall be performed.

NOTE – We attempt to perform contact graph routing first, before checking for a direct transmission option, because the direct next contact with node D might be far in the future; it is possible that an indirect route through some other neighbor might enable the bundle to be delivered earlier.

### static routing

If at least one static route has been defined, by management, that indicates that all bundles destined for node D may be directed to some gateway node, then the bundle shall be forwarded to the gateway node associated with the most narrowly defined of all such static routes and no other alternative route determination procedure shall be performed.

NOTE – The details of static route declaration syntax are an implementation matter. As an example of what is intended, suppose each static route declaration associates a gateway node’s ID (expressed as a BP endpoint ID) with a range of destination node numbers. Suppose the range of nodes numbered 10 through 30 are associated with the gateway node identified by the EID “ipn:901.0” while the range of nodes numbered 16 through 19 are associated with the gateway node identified by the EID “ipn:816.0”. In this case, a bundle destined for node 27 would be forwarded to ipn:901.0 but a bundle destined for node 17 would be forwarded to ipn:816.0. That is because, while 17 is between 10 and 30, it is also between 16 and 19, the more narrowly defined static route. In effect, the route to ipn:816.0 overrides the more general static route to ipn:901.0, but only for bundles whose destination nodes are in the range 16 to 19.

For this purpose, the route to the gateway node shall be determined by exercise of the route determination procedures defined in this Proposed Standard.

### Routing failure

If none of the route determination procedures defined in this Proposed Standard result in the enqueuing of the bundle for transmission on some convergence-layer transmission channel, then route determination shall be deemed failed. Procedures for responding to failure in route determination shall be an implementation matter.

## Exception Handling

### Overbooking management

Enqueuing a bundle for transmission to some node may result in a postponement of transmission of one or more lower-priority bundles: the premises upon which a lower-priority bundle’s best candidate route was selected may no longer hold. When such an anomaly can be detected, new routes for the affected bundles should be determined by exercise of the route determination procedures defined in this Proposed Standard. Procedures for detecting and remediating such anomalies shall be an implementation matter.

### Contact failure

Route determination as defined in this Proposed Standard assumes that the information in the contact plan is reliable. However, actual periods of contact may not always conform precisely to the plan: when a planned contact begins later than planned, ends earlier than planned, or does not occur at all, or when the actual volume of a contact is less than the planned volume (e.g., because some contact time was consumed by unanticipated bundle retransmission), one or more bundles enqueued for transmission on the corresponding convergence-layer transmission may not be transmitted. In this event, new routes for the affected bundles should be determined by exercise of the route determination procedures defined in this Proposed Standard. Procedures for detecting and remediating such anomalies shall be an implementation matter.

NOTE – The efficiency of SABR varies directly with the degree of agreement among the contact plans exposed to the nodes. Deficiencies in this agreement will introduce errors, resulting in sub-optimal transmission of bundles to neighboring nodes.

NOTE – In the general case it is impossible to determine whether forwarding along a given route will in fact result in bundle delivery at the destination node: storage resource availability and transmission opportunity commitments at “downstream” nodes cannot reliably be predicted in a delay-afflicted network. Contingent forwarding and reforwarding procedures must therefore be in place at all nodes to minimize the likelihood of bundle expiration prior to delivery. Such procedures are beyond the scope of this specification. However, the developer is advised that extremely large bundles are in general more vulnerable to forwarding failures due to resource exhaustion than small bundles are. Proactive fragmentation into relatively small fragmentary bundles can improve bundles’ forwarding prospects at all points along the end-to-end path, by enabling brief contacts to be allocated to these smaller bundles.

### Custody refusal

When a BP custody refusal signal is received, citing some previously forwarded bundle, a new route for the affected bundle may be determined by exercise of the route determination procedures defined in this Proposed Standard. In this event, backward propagation of the bundle shall be authorized as described in 5.1.7 above. Moreover, if the node that sent the BP custody refusal signal is the neighboring node to which the bundle was previously forwarded then that node shall deemed an excluded neighbor as defined in 4.4 above.

Alternatively, a node’s failure to forward a timely custody acceptance signal citing some previously forwarded bundle may similarly initiate determination of a new route for the affected bundle by exercise of the route determination procedures defined in this Proposed Standard.

Procedures for detecting these conditions and initiating remedies are an implementation matter.

# Acknowledgments

CCSDS gratefully acknowledges the important contributions to Contact Graph Routing practice made by researchers from outside the CCSDS community. In particular, we thank:

* Juan Fraire, for in-depth analysis of the route computation algorithm.
* Sara El-Aloui, for proposing anticipatory fragmentation.
* Carlo Caini, for investigation of overbooking management.
* Nikos Bezirgiannidis, for thoughts on the computation of earliest transmission opportunity.

1. Informative References

(This annex is **not** part of the Recommendation.)

[A1] *CCSDS Streamlined Bundle Security Protocol*, Recommendation for Space Data System Standards (White Book), CCSDS 734.5-W-2, in preparation.

1. Security, SANA, and Patent Considerations
   1. Security Considerations

This document specifies how to determine the routes over which bundles should be forwarded, given information such as a set of static routes and a contact plan. How the contact plan(s) should be distributed to the various nodes in the network, and whether all nodes need exactly the same information or not is beyond the scope of this document. It should be noted, however, that the information used to generate / update the contact plan(s) should be secured. There are a number of instances of routing attacks in the terrestrial Internet (intentional and unintentional) to motivate this. One possible means of securing contact plan distribution is by using the CCSDS Streamlined Bundle Security Protocol ([A1])

* 1. SANA Considerations

This recommendation does not require any new registries to be instantiated, and does not use any registry information from SANA.

* 1. Patent Considerations

At the time of publication, CCSDS was not aware of any patents pertaining to the technology described in this document.