A NETWORK ARCHITECTURE FOR INTER-SATELLITE COMMUNICATION OF DISTRIBUTED NANO SPACECRAFT SYSTEMS

W. Frese, K. Brieß, and Z. Yoon

Department of Aeronautics and Astronautics, Technische Universität Berlin, 10587 Berlin, Germany

Abstract

Nanosatellites become more and more attractive for future space missions because of high cost effectiveness and flexibility. These high integrated satellites with a mass up to 10 kg gain more importance for space missions such as messaging services, disaster monitoring and earth observation. Distributed satellite systems with inter-satellite links provide enhanced system capabilities with respect to time-delay, system redundancy and availability. An interaction among the satellites requires an adapted network architecture. Therefore, to satisfy the required system constraints in diverse missions a concept for a satellite network architecture is proposed. The main focus is laid on nanosatellite network topologies with short and medium distances up to some 100 km applying one single (radio frequency) working channel. Potentially applicable multiple access techniques like CDMA, FDMA and TDMA are discussed and evaluated. The results show, that time division duplex scheme with session-oriented point-to-point protocols in data link layer is more suitable for selected missions and strong resource limitations in nano spacecraft platforms. Furthermore, an applicable layer model is defined and a possible protocol implementation is outlined.

Key words: wireless communications; nanosatellite; network; inter-satellite link, distributed spacecraft.

1. INTRODUCTION

Nanosatellites are frequently associated basically with the university-class small satellites for hands-on education of students. Nonetheless, they can also be used for on-orbit verification of novel space technologies by reason of their short design times, low development and launch costs. Today's trend towards miniaturisation of satellites and use of commercial off-the-shelf components (COTS) with reduced power consumption, has already led to remarkable progress in microsatellite technology. Advances in microelectronics and micro-mechanical systems make one confident, that the nanosatellites can easily match the capabilities of the microsatellites over the next decade. New dimensions in miniaturisation lead to a higher realiability of nano spacecraft through high amount of redundancy. The success of this approach has been already demonstrated in orbit [4], novel miniaturized satellite components are designed [12], e.g. a data transmitter in S-band for picosatellites [6]. Nonetheless, pico- and nanosatellites are still viewed by many as primitive, if not even as "space debris". Though, mostly having no capability for an active de-orbiting manoeuvre, they could be ejected into the self-decaying orbits according to the IADC's Space Debris Mitigation Guidelines [3].

Both, the increasing of autonomy on system level and the today's slide to multi-satellite systems lead to a necessity for an inter-satellite communication. The crosslinks become essential in a large number of applications, especially where robotic aspects are predominated. Apertures for earth observation or for astronomical purposes can be assembled or reconfigured on-orbit from a number of smaller satellites, each of them equipped with a terminal for inter-satellite communication. Satellite swarms are also a new application field, where a bi-directional *inter-satellite links* (ISL) is necessary. The techniques for control and self-organisation of a nanosatellite network are a essential in realising of a *distributed satellite system* (DSS) and one of the big technical challenges.

This paper proposes a network architecture for autonomous nano spacecrafts. The rest of this paper is organized as follows; section 2 gives an overview of the actual nanosatellite platforms, applications and the state of the art in terms of inter-satellite communications. Further, section 3 describes a formal model for analysing mobile small satellite networks and presents some major challenges which a designer of distributed control alorithm has to deal with. Section 4 points out the network topology and in section 5 the routing algorithms are introduced. Finally, we discuss the concept of proposed media access scheme data link and network layers.

2. NANOSATELLITE PLATTFORMS AND APPLICATIONS

In this section, some potential target nanosatellite applications and platforms are analysed. System level requirements on nanosatellite network are determined through the available mass, volume, onboard electrical energy ressources and geometrical interrelationship in DSS. A major impact on the functional organisation and operational principles of the network is given by the nominal operating mode forced by the application. In general, the robotic aspects of future satellite constellations such as demonstration of autonomous orbital manoeuvres and formation flying can be verified with nanosatellites. In addition, one should not forget the pure communication missions. Small- and medium-bandwidth "classic" store-and-forward messaging applications like Orbcomm or more sophisticated, also microsatellite-based, but equipped with crosslink terminals, COMMStellation can be mentioned here. Low-cost satellite control without ground segment, only via Internet and a store-and-forward satellite network in *low earth orbit* (LEO) was already demonstrated, even though not in real time [5]. The quality of service (QoS) for applications like this could be improved if crosslinks were available.

According to the generally applicable defintion for nanosatellites, their mass ranges between 1 and 10 kg. In this paper, we lay the focus on the nanosatellites with the average mass about 10 kg as most promising class in near future. A typical nanosatellite is SNAP-1, developed by Surrey Satellite Technology Ltd and launched in 2002 [14, 15]. The mass of the SNAP platform can vary in a wide range from 6 to 12 kg depending on the payload. The dimension is approx. 30×30×30 cm^3, and the unregulated bus voltage is 7. . . 9 V. Other advanced nanosatellites have comparable parameters, e.g. CanX, measuring 20×20×20 cm^3 and 7.5 kg [11]. The average power in a low earth orbit can be approximated with 5. . . 10 W if the state-of-the-art multi-junction GaAs solar cells are used. Today's nanosatellites have no capability for permanent attitude and orbit control, no large directional antennas find enough place on S/C's structure, hence omni-directional antenna patterns are usual.

Mainly because of the above mensioned constraints the state of the art in nanosatellite communication technology is characterised by use of the UHF/VHF frequencies and a maximum data rate of 38.4 kbps for downlink. Some systems use S band for downlink (cp. SNAP-1 uses 38.4 kbps, CanX uses a link with nominal 32 kbps, up to maximal 2 Mbps). As for ISLs, the spectrum ranges from the UHF frequencies and distances up to some kilometers (SNAP) to ultra-wide-band (UWB) techniques and very short distances of some ten meters [9]. It is notable, that the SNAP-1/Tsingua-1 remains the only constellation, where an *point-to-point* (P-to-P) inter-satellite link was verifyed on-orbit, no multi-satellite networking with small satellites was demonstrated yet. As for data rates, the most advanced of the planned nano constellations like CanX4/CanX5 operate with the bitrates of 10 kbps [10] at max. 1000 m up to 5 km and two communication terminals only.

In this paper we propose a network architecture for nanosatellites, flexible enough to be used in a large number of applications, which can be easily implemented on state of the art communication hardware and integrated into a state of the art nanosatellite bus.

3. FORMAL NETWORK MODEL

We define a *satellite network* as a set of *nodes* that are able to communicate over wireless links. Nodes are characterised through their high *mobility*[1]. This means that, in general, the availability of the link between particular set of two nodes can change with time. However, this availability in satellite networks can be shown as highly predictable. In the graph theory the links are called *edges* and are represented by the *communication graph* $G(V, E)$, consisting of nodes V and edges E. Further, the terms *vertice* and *node* are used interchangeable.

We assume that all nanosatellites in a network are technically identical and have the same ressources onboard, most notably, the same electrical energy. Hence, it is not advantageous to distinguish between the routers and the user terminals, each node is equal and has the functionality of a router. In common case, a satellite network is a *meshed* network (see Section 4), that needs to be controlled by a *distributed algorithm*. This chapter gives a small and in some aspects incomplete introduction to the basic challenges facing a designer of distributed control algorithms for meshed satellite networks. As a big amount of related work is gathered through the ad hoc sensor network research, a comparison with the classic ad hoc concepts will be given in this chapter and can be used for further investigations.

Differences compared to the ad hoc networking have a reliable impact in several aspects. First of all, in ad hoc or sensor wireless networks one of the most determinant conditions is the unavailability of the position informations in a set of wireless nodes and a high and unpredictable mobility of the nodes. The above mentioned basic properties of an ad hoc wireless network remains the most challenging issues. E.g. it causes the necessity of highly dynamic collection of routing information. This decreases the data throughput inevitably. In contrast, a meshed nanosatellite network can be characterised as follows.

- Positions of all nodes at each moment are known; indeed, prediction into the future is possible.
- Significantly greather distances up to 100 km and more; consequently, propagation times need to be taken into account and make a big impact on the timing scheme and media access control.
- Clearly more ressources onboard the spacecraft compared to a node from a sensor ad hoc network.
- *Line-of-sight* (LOS) wave propagation is characterised through no shadowing and almost no reflection, scattering or diffraction, so the channel can be well approximated with the AWGN mathematical model.

[1]Not only fixed satellite constellations but also autonomous orbital maneuvres, reconfiguration or swarm behaviour must be taken into account.

- In general case, considering that the polar sun synchronous orbits are the first choice for small satellites, the satellite formation will be "stretched" over equatorial regions and will be more compact above the north and south pole; regular changes in edge number and weight are axpected.

The position of each node is determined by the orbital dynamics and is highly predictable. An exception occurs when orbital maneuvres are performed, which lead to significant and quick changes in the physical network topology. However, they are slow compared to the changes in signal quality and connection matrix of terrestrical wireless networks, where not only alone the movement of the nodes, but also the wave propagation effects such unpredictable shadowing and reflection. Moreover, the most concepts for sensor networks are limited by use of simple controllers (8 bit, few kilobytes of RAM) and only primary battery cells.

4. NETWORK TOPOLOGY

The network topology is determined on i) orbital configuration, ii) DSS operating mode and desired QoS and iii) possible ressource differences between the satellites. In Section 2 some multi-satellite applications were proposed. The orbital configuration is the most heavily weighted factor affecting the basical network topologies. The most easy consists of only very few S/Cs, placed into the same orbital plane with short inter-satellite distances, as in case of *precise formation flying* (PFF). The communicaion graph is fully-connected. This case can be approximated with a *fully connected mesh* or, if one satellite is used as a base station[2], with centralised *star topology* (see Figure 1a and 1b). Since each pair of nodes can be interconnected directly or via a base station, no advanced routing techniques are necessary. In case of the star topology, the channel syncronisation can be organised by the base. By fully connected scheme either a satellite with master function is choosen (arbitrary or via ground station) or a random access protocol, e.g. pure ALOHA, could be used. Another common case occurs when the satellites are placed into the same orbital plane with separation distances approximately equal to the maximal possible communication range (*line topology* see Figure 1c). The most common case is the *meshed topology* (see Figure 1d). The graph representation of the network can change with time. The control algorithm of this topology is relatively difficult, because a multi-hoping and no centralised controller is required. However, such algorithm has an important feature; it can be used to control each of the other above mentioned topologies, albeit not always with the desired performance.

The basic concept of the meshed topology can be extended as follows. In Ssection 3 was already mentioned that the density of the satellite swarms in LEOs with high inclinations show regular changes above the pole regions compared to equatorial areas. Figure 2 shows two subsets of nodes which are interconnected twice each orbital period. This is a significant special case in our analysis. If the distributed routing algorithm could perform well in interconnected subsets of nodes and fall back to a store and forward scheme in situation that no interconnection is possible, better QoS can be expected. In Section 5 we assume a suitable algorithm for this case.

Assumption, that can make the theoretical much more easier, is that the graph, that represents a satellite network, is not directional.

[2] sometimes called master

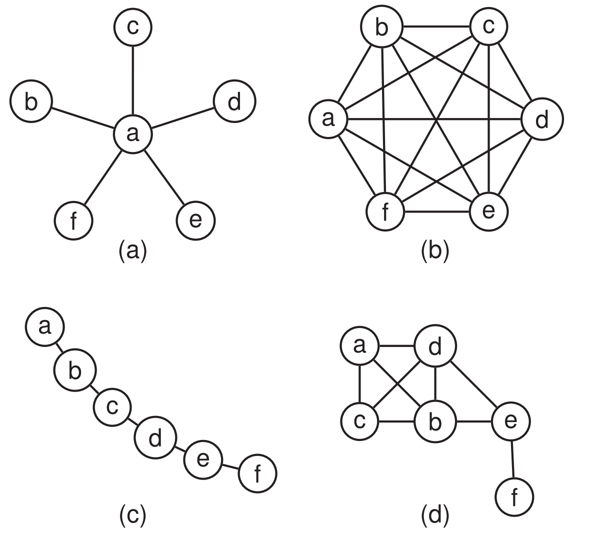

Figure 1. Probable network topologies for distributed nanosatellites. (a) Star topology with a single communication master as can be used for PFF. (b) Fully connected topology can also be used for PFF. (c) Line topology as can be used for communication missions in LEO. (d) Mesh topology, the most common case, frequently associated with a satellite swarm.

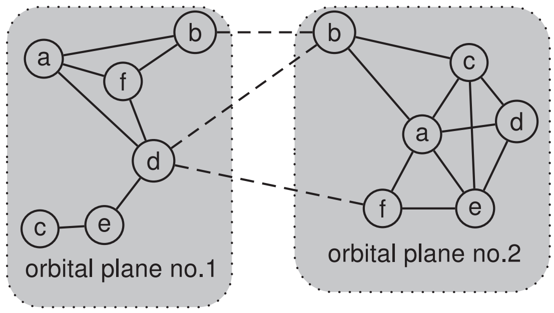

Figure 2. A possible graph representation of a meshed satellite network in two polar orbital planes. Dashed edges represent temporary available links above the geographical poles.

5. ROUTING

As already discussed in Section 4 in some nanosatellite applications no routing is required. This is the case if the maximal dimension of the constellation is smaller than the maximal possible communication range (see Figure 1b). In such cases, only the channel access techniques should be taken into consideration and require a major effort in optimizing the latencies. One should think about the PFF missions, where the communication links are a part of the algorithm for formation control.

In other cases, where a multi-hop communication is necessary to connect particular node to each other, flooding techniques could be used as for unicast connections, as well as for broadcasting. As flooding algorithms in most cases perform worse compared to routing ones, particularly at large distances, they are not analysed here.

The routing protocol is called *proactive*, when the routing information is stored in the memory and no necessity for building new routing table exists. Otherwise, a new routing information will be identified each time if a new message has to be transmitted (*reactive* protocol). The proactive routing is frequently associated with networks with fixed topologies. Reactive routing requires large overhead. As the mobility in satellite networks is predictable, on one hand, no overhead typical for reactive routing is required. On the other hand, the topology changes must be identified and the routing table has to be dynamically updated. In this paper we assume a memoryless distributed algorithm to control mobile meshed satellite networks. No control informations, e.g. no routing tables, must be exchanged between nodes and no central controller is required.

Definition (Unicast Geographic Routing in Meshed Satellite Networks) *Let $G = (V, E)$ be an Euclidean 3D graph consisting of a finite number of vertices n. The task of a geographic unicast routing algorithm $\mathcal{A}$ is to find an (optimal) way from any* source s *to any defined* destination t *while complying with the following conditions:*

- *All nodes $v \in V$ know their own geographic positions and geographic position of all other nodes $w \in V \backslash v$.*
- *There is no control information which needs to be stored in the node, memoryless algorithm.*
- *The node is not allowed to change any information addressed to an other node.*
- *The node is not allowed to maintain any information addressed to an other node except for the temporary storage before forwarding.*

The Dijkstra's algorithmus fulfils these requirenments [7]. Besides, Dijkstra delivers an optimal solution and has a finite running time and was therefore identified as a suitable algorithm for meshed satellite networks and performs well in all discussed network topology with connected graph representation (see Figure 1).

In the improbable case of a lost of position informations or in satellite swarms shortly after their deployment in orbit the initial position informations can be commanded by the ground station. If this is not possible, use of algorithms for 3D routing is unavoidable. After finding the neighbors through transmitting broadcast messages, an 3D routing algorithm can be used [8]. Such algorithms theoretically cannot provide optimal results. How such routing will perform in an orbital network is not known yet. In some cases a 3D geometry of a satellite constellation can be approximated by a plane Euclidean graph. Well known routing algorithms are *Face Routing*, *Adaptive Face Routing* or GOAFR+.

In a second special case, not interconnected subsets of nodes (see Figure 2) may arise as a result of a disadvantageous orbital dynamics. The communications performance and QoS can be badly affected by this constellation. A possible approach to avoid data loss and to increase QoS is to combine a routing with a store and forward technique by using an algorithm, that performs Dijkstra if the source vertice and the destination vertice can be connected directly or via hops. If the source and the destination are in different subnetworks, which will be connected later, the procedures described in Algorithm 1 and in Algorithm 2 could be used. The basic idea is to build a new graph, where the edge costs for a particular pair of vertices (s, d) are the times to the first possible contact. A shortest path tree for each node in the new graph $G'(V', E')$ can be searched with Dijkstra's algorithm, whereby one important condition has to be fulfiled: the costs of each path have to be in ascending order. For the cases where no contact between source and destination is possible, an ground station is used as a hop.

6. FREQUENCY ALLOCATION

The Space Frequency Coordination Group (SFCG) has adopted several resolutions and recommendations related to the operation of inter-satellite links. According to these in [1] all frequencies suitable for implementing in multiple spacecraft systems were analysed and a selection of five preferred frequency bands was proposed (S, Ku, Ka bands).

- 2025 - 2110 MHz, SPACE OPERATION, EARTH EXPLORATION SAT., SPACE RESEARCH
- 2200 - 2290 MHz, SPACE OPERATION,EARTH EXPLORATION SAT.,SPACE RESEARCH
- 14.5 - 15.35 GHz, Space Research (The 14.5-15.35 GHz band is on the agenda of WRC-03 for possible upgrade to primary status)
- 22.55 - 23.55 GHz, INTER-SATELLITE
- 25.25 - 27.5 GHz, INTER-SATELLITE

The nanosatellites are considered to be ultra low-cost spacecraft for space research and technology verification. Hence, the design-to-cost approach is applied even

Algorithm 1 Store and Forward with Interconnected Satellites

procedure StoreAndForward($G(V,E), source, E \rightarrow \mathbb{R}^{+}$) ▷ $c(e)$ are e.g. distances, $e \in E$
  if G is connected **then**
    perform Dijkstra's algorithm
    return
  end if
  $G'(V', E'), k(V') = n, k(E') = n - 1$, complete graph, $G = G' \backslash g$, whereby g - additional vertice, such that all edges $e(g, v), v \in G$ have at least one cost factor $0 \leq c(e,t) < \infty$ for $t = 0...finiteinterval$
  $c'(e) \leftarrow \infty$ for each $e \in E'$
  $t \leftarrow 0$ ▷ time
  while $t < interval$ **do**
    increase time t
    for each edge $e(v, w) = \infty$ **do**
      if $c(e,t) < \infty$ **then** ▷ link possible
        $c'(e) \leftarrow t$
      end if
    end for
  end while
  perform Dijkstra for $G'(V, E'), source$ so that the edge costs are ascending.
  store messages according to $c'(e)$
  forward messages, **if** edge costs are ∞ **then** forward to g
end procedure

Algorithm 2 Dijkstra's Algorithm for Ascending Costs

procedure Dijkstra+($G(V,E), sources,$)
  $ready[s] = true$
  $ready[v] = false$ for each $v \in V \backslash s$
  $dist[s] = 0$
  $dist[v] = \infty$ for each $v \in V \backslash s$
  p_queue Q
  for each edge outcoming $e = (s, v)$ from s **do**
    $previous[v] = s$
    $dist[v] = c(e)$
    Q.Insert($v, dist[v]$)
  end for
  while Q is not empty **do**
    $v = Q$.DeleteMin()
    $ready[v] = true$
    for each outcoming edge $e = (v; w)$ from v **do**
      if $w \in Q$ **and** $dist[v] + c(e) < dist[w]$ **and** $c(e) \geq c(previous[v]; v)$ **then**
        $previous[w] = v$
        $dist[w] = dist[v] + c(e)$
        Q.DecreasePriority($w, dist[w]$);
      else$w \notin Q$ **and** $ready[w] = false$
        $previous[w] = v$
        $dist[w] = dist[v] + c(e)$
        Q.Insert($w, dist[w]$);
      end if
    end for
  end while
end procedure

for the communication system of nanosatellites and determines the frequency selection. S band is preferred for use in nanosatellite networks. In particular, the good availability of miniaturised commercial components providing high power efficiency and the easy antenna design are the decision criteria. However, it must be mentioned that currently the S band frequency allocation is strongly occupied due to telemetry and telecommand services. Therefore only relatively small spectral bandwidth can be claimed.

7. MEDIUM ACCESS AND CHANNEL CODING

A suitable *channel access* method for satellite networks depends on several criteria. Technical feasibility of a hardware realisation, application requirements, network topology and the number of terminals are some of the most important of them. In this section we give a short analysis of the time, frequency, code und space division techniques in terms of their suitability for nanosatellite networking. Additionally, for packet switched channels a short analysis of random access and token passing procedures is provided. Space division is not analysed due to large and complex antennas needed to be integrated into a nano spacecraft and high requirements to the attitude determination and control.

FDMA and CDMA have an advantage that multiple crosslinks can occure simultaneously. In case of FDMA the most problematic issue is that the more terminals are participated in communication the greather is the frequency band allocation required for the mission. In Section 6 has been already mensioned that today S band frequencies are used extensively for telemetry and telecommand services in a great number of missions. Furthermore, use of FDMA leads to major complexity in design through a necessity of an extensive band-pass filtering and adds to communication system costs.

CDMA requires quick and precise RF power control and a complex signal processing. FDMA and ecpecially CDMA would be a good choice for precise formation flying missions with a little number of spacecraft and very hard time latency requirements as it proposed in [13]. In this paper we propose a network architecture which must be suitable for most nanosatellite missions today and in near future, although with different restrictions. In a network where short links as well as large ones can occur simultaneously and only simple antennas can be used (both assumptions are valid for nanosatellite networks) the signal strength in a decoder can vary within a large range. A safe signal separation is difficult.

In general, time division duplex (TDD) seems to be a good choise for nanosatellites. Fore small formations synchronous time-division multiplexing can be used (classic TDMA). All S/Cs need to be synchronous in time which is not easy to guarantee. A central controller has to be used in this case for time synchronisation and control of the network. This asymmetrical functional organisation leads to a higher ressource consumption in the communication master. Additionally, data exchange over large or variable distances lead to a necessaty of bigger guard distances. An other disadvantage is a fixed bandwidth allocation for each S/C, a fixed time slot is reserved for each S/C whether or not the data need to be send.

In this paper we propose two channel access methods for nanosatellite networks, both based on asynchronous time-division rules for packet switching: *token passing* and *random access* (RA). In both cases P-to-P protocols are used, only one P-to-P connection is possible at each time. A bi-directional P-to-P data transfer is time-duplexed and organised in relatively short *sessions* with variable lengths. Each session consists of a number of TDD frames (see Section 8). In this context, a suitable *channel coding* is very important.

There are two basical principles of channel coding. If the errors are corrected at the decoder it is called *forward error correction* (FEC). This approach is better if the channel is bad, but not too bad. The reliability depends on channel quality while the data throughput remains constant. FEC seems to be a better choice for simplex channels, e.g. for payload data downlinks. Error correction is especially recommended for pico- and nanosatellites satellites, where not much power onboard the S/C is available [6]. Some bit errors in payload data can be tolerated if the decoder can detect error blocks in the data stream. The second principle is based on *automatic repeat request* (ARQ). In contrast to FEC, here the reliability is constant and the data throughput can vary depending on the channel quality.

A hybrid scheme is defined as the best approach for nanosatellite networks because of two reasons. First of all, the communication distances in a nanosatellite network, as these defined herein, ranges from some hundred meters to hundred kilometers or even more. Huge distances and low ressources make a reliable communication with nanosatellites very challening. In addition, no directional antennas can be used in nanosatellites and an omni-directional antenna with a truly isotropic power pattern is not possible. An additional coding gain is a good choise to meet the requirenments, but cannot ensure a loss-free data exchange in an autonomous distributed satellite system. On the other hand, when a bi-directional data transfer is required, ARQs can make the transmision reliable at very low *signal-to-noise ratios* (SNR).

A token passing scheme (see Figure 3) is collision-free, easy to implement and proposed to be a good choice for formations with a small number of nodes. In this case no routing is needed. In case of a line network topology (Figure 1c) the *token* is passed from one end of the formation to the other and back again. For fully-connected topologies (Figure 1b) a token ring organisation is preferred. The data throughput is disadvantageous as in this case multi-hop connections are used although each particular pair of nodes can be interconnected directly.

More advanced concept is the random access (see Figure 4). RA is preferred fo meshed topologies where multi-hoping is unavoidable and there is a great number of nodes. Through a natural space separation even simultaneous links on the same channel are possible. Bacause each session consists of several TDD frames (see Section 8), in case of an collision not the whole data is lost but collided frames only. This scheme can be extended through a collision avoidance rule and/or a power control technique.

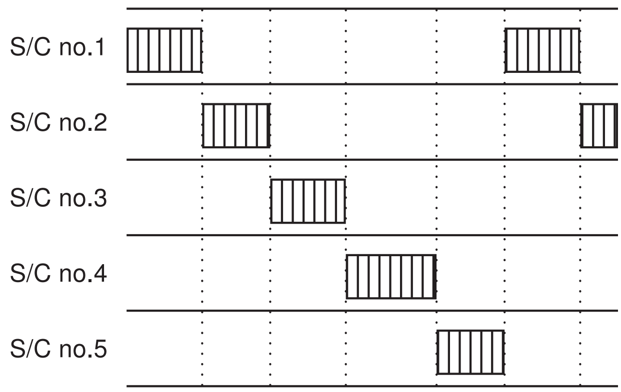

Figure 3. Token passing (ring). Boxes indicate short bi-directional sessions with flexible or fixed lengths. Dotted lines stand for token passing.

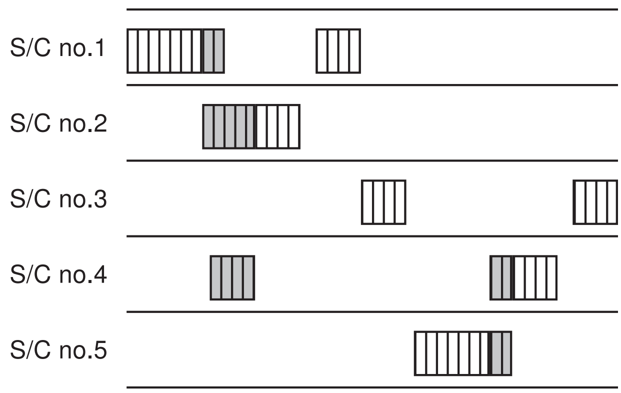

Figure 4. Simplified diagramm for random channel access. Boxes indicate short bi-directional sessions with flexible or fixed lengths. Shadowed boxes indicate session frames which have collided.

8. DATA LINK LAYER

A data link layer must incorporate rules or procedures for link *establishment*, for lossless *data transfer* and for *terminating* communication. A P-to-P protocol and time-division duplexing seems to be a best solution for nanosatellite networks and performs well especially in a meshed topology where multi-hoping is unavoidable and no central controller is beneficial.

Two ends of a link are called *caller* and *responder*. A caller satellite is the initiator of a session establishment process. A responder satellite receives session establishment parameters from a caller. A session is a continuous dialog between caller and responder. It consists of three distinct phases: session establishment, data services and session termination. The point-to-point bidirectional TDD data stream is divided into frames with constant length t_W (see Figure 5). Each frame contains two subframes: one caller frame (Xmt) and one responder frame (Rcv). The data transfer is asymmetrical, Xmts are longer as Rcvs which typical consist only of an ARQ or control informations. Each Xmt consists of a preambel (Pr) for one-shot packet frame synchronisation, one or two headers and one data block. Rcvs have a similar structure. It can be easily shown that the time t_1 (Figure 5) is equivalent to the two-way propagation delay $2t_p$. To keep the frame length constant, $t_1 + t_2$ is selected to be slightly more than the time needed for the maximal distance between two satellites.

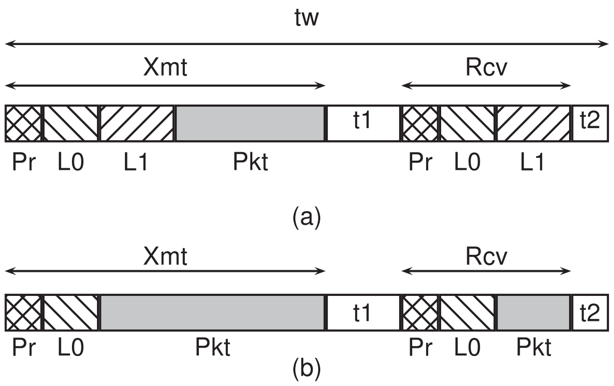

Figure 5. TDD frame structure. (a) Link establishment and adjustment phase. (b) Data transfer phase (with ARQ possible in one direction only).

The so called zero layer header (L0) contains 12 BPSK modulated and convolutional coded data bits with the code rate $r = 1/2$. A frame counter for the ARQ procedure and data direction informations are provided by L0. The modulation and coding scheme can be changed quickly during the session depending on the channel condition by L1. L1 is used for i) session establishment and ii) adaption of the modulation and coding scheme during data transfer. Parameters like S/C caller and responder IDs, modulation and coding options are also provided by L1. L1 includes 31 bits, the modulation and coding are the same as in case of L0 to guaranty a low bit-error probability in bad channels.

Once a session is established and the channel is stable, no L1 header is necessary anymore, so more bandwidth can be provided for data packets. The responder answers during data transfer phase with L0, wich is detected by caller as an acknowledge (ack). If no ack can be correctly decoded by the caller within the time $t_1 + t_2$ (negative acknowledge, nak), Pkt will be retransmitted in a new Xmt frame. The ARQ protocol follows the classic *stop-and-wait* procedure. A terminating rule is implemented for the case that only naks are detected and the SNR is not good enough to guarantee an orderly data transfer on lowest modulation/FEC combination.

As for data packets (Pkt), an *adaptive coding and modulation* (ACM) scheme is used, the parameters are as follows.

- BPSK, convolutional coding with $r = 1/2$
- BPSK, convolutional coding with $r = 3/4$
- QPSK, convolutional coding with $r = 1/2$
- QPSK, convolutional coding with $r = 3/4$
- 16-QAM, convolutional coding with $r = 1/2$
- 16-QAM, convolutional coding with $r = 3/4$

The modulation and coding scheme can be quickly changed within few TDD frames depending on the current SNR. A SNR detected by the responder is transmitted in each Rcv frame. After receiving of few Rcv frames and building an average SNR value (wich can take some ten milliseconds), the caller makes a decision about the necessity of a changing the modulation and coding scheme based on responders and his own SNR. An adoption occurs then by caller that sends a L1 header within the next TDD frame.

The optimal Pkt size n (netto) depends on error probabilities, which are different for Pr, L0 and Pkt, p_{Pr}, p_{Hdr} and p_{Pkt} respectively (we consider the data transfer phase from Figure 5b). p_{Hdr} and p_{Pkt} are given through BERs p_b for Pkt and header ($p_{b_{Hdr}}$ and $p_{b_{Pkt}}$). The average time t_V for a correct transmission to be received in an AWGN channel is given by

$$t_V = t_W/(1-p) \tag{1}$$

$$1-p = (1-p_{Pr})(1-p_{b_{Hdr}})^{n'}(1-p_{b_{Pkt}})^{n} \tag{2}$$

where t_W is the length of the whole frame, $n' = 12$ is the bit number for L0 and n is the bit number of the packet to be defined. n' and n are given without FEC bits. The maximum throughput λ_{max}, in frames/second, is then just reciprocal of t_V. The average data rate D, in bits/sec delivered in responder, is then, $D = \lambda_{max} n$. The frame time t_W can be derived from the symbol rate C in baud and from the number of all symbols in a frame n_W inclusive a symbol equivalent n_{TDD} for the time $t_1 + t_2$: $n_{TDD} = C(t_1 + t_2)$. Assuming that the symbol rate C is 80 kbaud, for normalised data rate D/C,

$$t_W = n_W/C \tag{3}$$

$$t_W = \frac{1}{C}(2n_{Pr} + 2n_{L0}/r_{Hdr} + 0.5n/r_{Pkt} + n_{TDD} + n_{L1}/r_{Hdr}) \tag{4}$$

$$t_W = (0.5n/r_{Pkt} + n_{TDD} + 174)/C \tag{5}$$

$$D/C = \frac{n(1-p_{Pr})(1-p_{b_{Hdr}})^{n'}(1-p_{b_{Pkt}})^{n}}{(0.5n/r_{Pkt} + n_{TDD} + 174)} \tag{6}$$

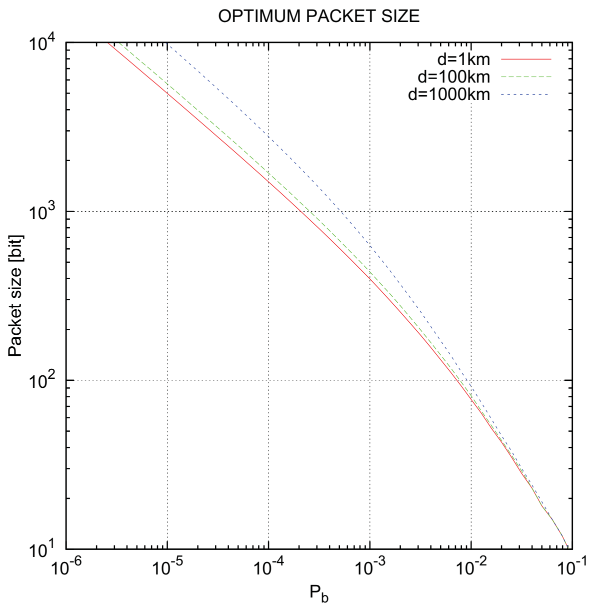

Figure 6. Optimum packet size for QPSK with symbol rate $f_s = 80\,k$ and convolutional coding with coding rate $r = 3/4$ and $k = 9$.

Computer simulations have shown that n_{opt} depends relatively little on p_{Hdr} and p_{Prm} for given frame structure. For ACM there is an additional difficulty in connection with n_{opt} since it depends on the modulation and coding scheme. The example which is given here is for the nominal case of QPSK with convolutional code rate $r = 3/4$ and $k = 9$. If the bit-error probability is choosen to be $p_{b_{Pkt}} = 10^{-4}$, one should be able to calculate the optimum packet length n_{opt} from Equation 5. In Figure 6 optimum packet sizes for different bit-error probabilities are shown. Three different n_{TDD} values for 1, 10 and 1000 km are processed. n_{TDD} should be changed for each mission independently based on the maximum theoretical communication distance. 100 km are proposed to be a good approximation for nanosatellite networks. n_{opt} is then approximately 1120 symbols. The maximal theoretical channel bitrate with QPSK and code rate $r = 3/4$ is then 99.7 kbps and the average bitrate at BER $p_{b_{Pkt}} = 10^{-4}$ is 84.2 kbps respectively.

9. FUNCTIONAL ORGANISATION

The network layers above the data link layer are proposed to be impelented on a S/C controller. As nanosatellites have very limited ressources onboard, it is mostly probable that no separate network controller can be used in practice, and therefore the S/C onboard computer (OBC) must be used for network control. In this section a brief overview of the functional relationships between the ISL communication terminal and the OBC is given.

In Section 5 a routing algorithm was proposed. A routing function is provided by the network layer. According to the proposed memoryless routing rule the network layer must be able to calculate the current geometrical relations inside the whole nanosatellite network. It can take place via an orbit propagator. As the orbital elements changes relatively slowly compared to the speed of the distributed network algorithm, there is no need to handle these changes by the network layer itself. This function can be provided by higher layers and the realization can be different for each mission.

The functional corellations between the date link layer and the network layer is related to the *Proximity-1* protocol proposed by CCSDS for planetary missions where autonomous communication terminals are used [2]. Here we give only a short overview of the protocol. Variable length frames are used for data exchange between the S/C controller and the ISL terminal. There are two kinds of data units. One data unit is for unspecified data provided by a data service and the other unit for user and control informations, typically generated by the network layer (see Figure 7).

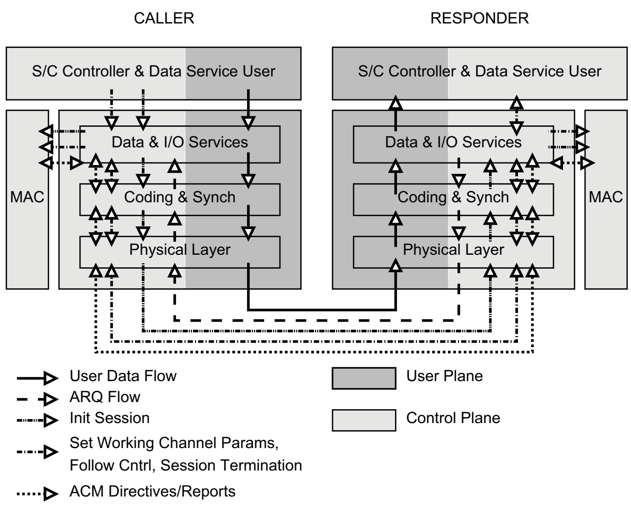

Figure 7. Flow of data and messages.

10. CONCLUSION AND OUTLOOK

A network architecture for distributed nano satellites was proposed within this paper. A concrete routing algorithm for dynamical satellite networks was identified and described. Also a time-division based data protocol for the data link layer was specified with an analysis of the optimal packet size which depends on error probabilities. The functional seperation of network layer and data link layer enables easy adaption of the ISL terminal into other missions.

The proposed routing algorithm and data link protocol will be implemented and tested in a dedicated suitcase test with five communication nodes. The results will be presented in following publications.

Overall, the small satellites can benefit from changing the focus from one-of-a-kind large spacecraft towards a group of smaller satellites, and a large number of applications require a crosslink capability between autonomous spacecraft.

ACKNOWLEDGMENTS

The R&D project SLink, on which this paper is based, is a co-operation between the Technische Universität Berlin (Berlin Institute of Technology) and the company IQ wireless GmbH. We are grateful to all project participants for their contributions and to Dr. Jäckel, to H. Podolski and to F. Hartmann for great technical support.

FUNDING

The project SLink is impelemted on behalf of the Federal Ministry of Education and Research (ger. Bundesministerium für Bildung und Forschung, BMBF) under Ref.Nr. 50 YB 1009.

SYMBOLS AND ABBREVIATIONS

$G(V,E)$	Communication graph
V	Set of vertices (nodes)
E	Set of edges
r	Coderate
D	Data rate
C	Symbol rate
p_b	Probability of a bit-error
k	Constraint length
ack	Acknowledge
ACM	Adaptive Coding and Modulation
ALOHA	ALOHA net, ALOHA protocol
ARQ	Automatic Repeat Request (Query)
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BPSK	Binary Phase Shift Keying
CCSDS	The Consultative Committee for Space Data Systems
CDMA	Code Division Multiple Access
COTS	Commercial On-The-Shelf
DSS	Distributed Satellite System
FDMA	Frequency Division Multiple Access
FEC	Forward Error Correction
IADC	Inter-Agency Space Debris Coordination Comittee
ID	Identifier
ISL	Inter Satellite Link
LEO	Low Earth Orbit; niedrige Erdumlaufbahn
LOS	Line-of-sight
nak	Negative Acknowledge
OBC	Onboard Computer
P-to-P	Point-to-point
QAM	Quadrature Amplitude Modulation
QoS	Qualities of Service
QPSK	Quadrature Phase Shift Keying
PFF	Precise Formation Flying
RA	Random Access
RAM	Random Access Memory
RF	Radio Frequency
S/C	Spacecraft
SFCG	Space Frequency Coordination Group
SNR	Signal-to-Noise Ratio
TDD	Time Division Duplex
TDMA	Time Division Multiple Access
UHF	Ultra High Frequency
UWB	Ultra Wideband
VHF	Very High Frequency
Pr	Preambel, synchronisation field
L0	L0 frame header
L1	L1 frame header
Pkt	Data packet, information field from network layer
Xmt	Caller frame
Rcv	Responder frame

REFERENCES

[1] Distributed spacecraft crosslink study. part 1, spectrum requirements and allocation survey report and recommendations. Research note, Goddard Space Flight Center, prepared by ITT Industries, Advanced Engineering & Sciences Division, june 2002.

[2] Proximity-1 space link protocol - rationale, architecture, and scenarios. Informational report CCSDS 210.0-G-1, The Consultative Committee for Space Data Systems, Reston, VA, USA, august 2007.

[3] Space debris mitigation guidelines. Technical Report IADC-02-01, Inter-Agency Space Debris Coordination Committee, sept 2007.

[4] F. Baumann, K. Brieß , M. Herfort, and S. Trowitzsch. Flight experience with the picosatellite beesat. In *59$^{\mathrm{th}}$ German Aerospace Congress, DGLRK 2010, Deutsche Gesellschaft für Luft- und Raumfahrt*, aug 31 - sept 2 2010.

[5] F. Bernelli-Zazzera, A. Ercoli-Finci, M. Molina, M. Cattaneo, M. Dioli, L. De Peron, I. Bertolini, R. Bianchi, P. Sabatini, L. Crocco, F. Schiavi, and A. Zucconi. On orbit technology validation for a university microsatellite. In *4$^{\mathrm{th}}$ IAA Symposium On Small Satellites for Earth Observation, Berlin*, apr 7-11 2003.

[6] K. Briess, R. Alavi, K. Jaeckel, and H. Podolski. S-band communication for nano- and pico satellites for cross platform compatibility. In *59$^{\mathrm{th}}$ International Astronautical Congress, Small Satellites Missions Symposium, Glasgow*, 2008.

[7] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959. 10.1007/BF01386390.

[8] R. Flury and R.R. Wattenhofer. Randomized 3d geographic routing. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pages 834 –842, april 2008.

[9] R.K. Lee, R. Watson, C. Kitts, P. Stang, and B. Palmintier. Anomaly detection using the emerald nanosatellite on board expert system. In *Aerospace Conference, 2004. Proceedings. 2004 IEEE*, volume 1, pages 6 vol. (xvi+4192), march 2004.

[10] N. G. Orr, J. K. Eyer, B. P. Larouche, and R. E. Zee. Precision formation flight: The canx-4 and canx-5 dual nanosatellite mission. In *Small Satellites Systems and Services, Rhodes, 4S Symposium on*, may 2008.

[11] F.M. Pranajaya and R.E. Zee. The generic nanosatellite bus: From space astronomy to formation flying demo to responsive space. In *Advances in Satellite and Space Communications, 2009. SPACOMM 2009. First International Conference on*, pages 134 –140, july 2009.

[12] S. Roemer, S. Stoltz, F. Baumann, and H. Kayal. Rw-1 the worldwide smallest commercial, flight proven reaction wheel for pico and nano satellites - fields of operation and flight results. In *Small Satellites Systems and Services, Funchal, 4S Symposium on*, may 31 - june 4 2010.

[13] R. Sun, J. Guo, E. Gill, and D. Maessen. Characterizing network architecture for inter-satellite communication and relative navigation in precise formation flying. In *Advances in Satellite and Space Communications (SPACOMM), 2011 Third International Conference on*, pages 52 –57, april 17-22 2011.

[14] C. I. Underwood, M. J. Crawford, and J. W. Ward. A low-cost modular nanosatellite based on commercial technology. In *AIAA/USU Small Satellite Conference, Logan, UT, USA*, 1998.

[15] C. I. Underwood, G. Richardson, and J. Savignol. In-orbit results from the snap-1 nanosatellite and its future potential. *Philosophical transactions - Royal Society. Mathematical, physical and engineering sciences*, 361:199–203, 2003.