Cross-Layer Designs Architecture for LEO Satellite Ad Hoc Network

Zhijiang Chang and Georgi Gaydadjiev

Computer Engineering Laboratory
Delft University of Technology
Mekelweg 4, 2628 CD Delft, the Netherlands
Telephone:. +31 15 278 6177
{`zhijiangchang,georgi`}`@ce.et.tudelft.nl`

Abstract. Future Low Earth Orbit (LEO) satellite networks are envisioned as distributed architectures of autonomous data processing nodes. Such ad hoc networks should deliver reliable communication channels for control commands and data among ground stations and satellites minimizing delay and power. The LEO satellite networks are different from the generic ad hoc scenario. In this paper, we first analyze the specifics of ad hoc LEO satellite networks. Next, we propose a cross-layer protocol architecture that includes three cross-layer optimizations: simple integrated MAC/PHY layer, novel Balanced Predictable Routing (BPR) and a dedicated QoS aware TCP sliding window control mechanism. They all contribute to the end-to-end delays improvement and successful delivery increase. It also fulfills the QoS requirements. According to our simulations, the coverage of ground stations is improved. The throughput percentage of all data types is improved by 5.8% on average and the QoS of high priority application is guaranteed.

Keywords: cross-layer design, LEO satellite network, ad hoc network.

1 Introduction

The future satellite applications require self-organized, dynamic network topology without predefined constellation [1,2]. Such applications include: 1) deep-space exploration that uses the Inter-Satellite Links (ISLs) of LEO network to communicate to the control center; 2) LEO satellites control that accesses satellites that do not have direct link to Ground Station (GS); 3) satellite telephone service in which the ground terminal is not directly connected to the satellite networks with a constellation. In order to support the above applications, the future satellite networks should be able to maintain the connectivity and efficiency when satellites join or leave the networks dynamically. Such networks are ad hoc networks. That will be characterized by their frequently changing topology and intermittent connectivity.

The current LEO micro satellites and their networking, however, do not meet the requirements to support the future applications. First, the satellites

J. Harju et al. (Eds.): WWIC 2008, LNCS 5031, pp. 164–176, 2008.

are launched by many different organizations for various purposes. Their efficiency is extremely low because most of them work alone. This is because: 1) the standalone LEO satellites are visible to GS control only for 10 to 20 minutes per rotation; 2) the LEO satellite's average lifetime is only around 6 years [3]. Secondly, the heterogeneous satellites have various computing and power capacities. This limits not only the efficiency of individual satellite, but also the potential cooperation among them. The heterogeneous architectures also lead to more complicated network organization, faster changing topology and less stable communication channels.

In order to develop a satellite networking system that fulfills the requirements of future applications and overcomes the above problems, we take the advantages of cross-layer designs. The cross-layer designs optimize the overall network performance by scarifying the layers' interdependencies [4]. A strict modularity and layer independence may lead to non-optimal performance in the future satellite networks. The heterogeneous network requires adaptability provided by the cross-layer designs. Furthermore, the LEO satellites can use cross-layer information such as signal strength variation to predict the motion of other satellites and the communication link quality.

The main contributions of this paper are:

- Analysis of QoS requirements and the related energy efficiency for ad hoc satellite networking context;
- Integrated MAC/PHY layer that provides network congestion and link quality information;
- Novel Balanced Predictable Routing (BPR) based on Dynamic Source Routing (DSR);
- A QoS aware TCP congestion control algorithm that especially ensures the delivery of the control commands;
- Careful simulation that validates our cross-layer architecture using ns-2.

This paper is organized as follows. We outline the special QoS requirements and problems of LEO satellite networks in section 2. Related work is discussed in section 5. Section 3 presents the proposed architecture and the three cross-layer optimizations. The simulation results are discussed in section 4. We finally conclude the paper in section 6.

2 Special Issues of LEO Satellite Networking

Some special issues of the LEO satellite ad hoc network are considered while we design the cross-layer architecture. First, the satellite networks have special runtime QoS requirements for mission control and payload specific onboard applications. Second, different link quality of different types of ISLs is essential for the satellite networks. This section discusses these special issues that have significant impact on the proposed cross-layer architecture.

2.1 QoS Requirements for Applications

The satellite networks have strict bandwidth limitations. Different types of application in the LEO satellite networks have different bandwidth requirements for QoS. We classify the traffic flows to three classes: (i) the mission control flows that should be delivered at all costs are given the highest priority; (ii) the real-time services such as satellite telephone that have bandwidth and delay requirements have the second high priority; (iii) the non-realtime services such as FTP that can be delivered at best effort (as good as possible) have the lowest priority.

2.2 Inherit Problems of LEO Satellite Networks

Providing full ad hoc connectivity in the LEO satellite network is a challenging task. The following problems in the satellite networks should be considered: (i) High Bit Error Rates (BER) on satellite links are caused by signal interferences, such as atmospheric or ionosphere effects and artificial jamming. For instance, the quality of ISL changes rapidly when the link path goes through the atmosphere. (ii) Load balancing is a major problem in such networks because of the limited power budget of nodes. The normal routing protocols are more likely to use links with better quality such as shorter delay than links with longer delay. The traffic load should be fairly distributed in the network in order to avoid exhausting some satellites' energy when leaving others unused. (iv) Different types of ISLs have strong impact on the network topology. Intra-plane ISLs and Inter-plane ISLs of adjacent planes are stable links because that the relative position of the satellites is stable for certain periods. Cross-seam ISLs are fast changing unstable links that are only available to a short period.

3 Cross-Layer Architecture For LEO Satellite Networks

We propose an cross-layer architecture that involves three cross-layer designs. These three designs map QoS control at all layers, all being time-varying. The cross-layer optimizations provide not only QoS control to the applications, but also approaches to overcome the inherit problems of the satellite networks as stated in the previous section. The first optimization is an integrated MAC/PHY layer that provides more accurate and adequate information to other cross-layer optimizations. The second optimization controls the sliding window of TCP protocol in order to guarantee the delivery of application data with higher priority. The third optimization adapts the Dynamic Source Routing (DSR) protocol to LEO satellite network to use more stable links and balance the traffic in the network at the same time. Multiple cross-layer designs have potential risk of malfunctioning when interacting with each other. Such problems includes shared information access and adaptation loops [5]. In order to prevent such problems, we use the infrastructure for cross-layer design interaction proposed in [6] to ensure that the three optimizations are loop-free, and behave correctly according to their designs.

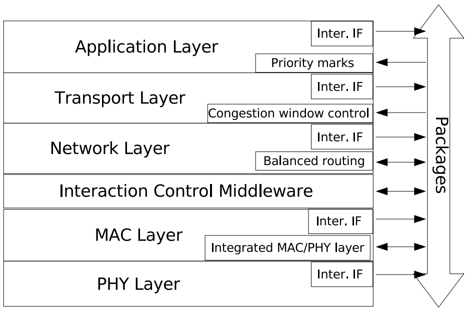

Fig. 1. The cross-layer satellite architecture

As shown in figure 1, the following cross-layer information is propagated in the protocol stack: the priority information provided by the applications; the wireless link quality information by the integrated MAC/PHY layer. The routing protocol uses links with better quality using the wireless link quality information. The TCP layer adjusts the congestion window size according to the MAC/PHY layer information and the application priority.

3.1 Integrated MAC and PHY Layer

We propose an integrated MAC/PHY layer to simplify the information provided to the upper layers. We propose a normalized variable called Degree of Collision (DOC) to represent a node's wireless link quality. The two main contributions of the paper are: 1) the Balanced Predictable Routing (BPR) mechanism presented in section 3.2 and 2) the QoS aware TCP sliding window control introduced in section 3.3. We only use DOC as an attribute representing the ranking of link quality in the BPR and QoS aware TCP control. The performance of BPR and QoS aware TCP sliding window control are invariant to how the DOC is calculated. We do not focus on optimal DOC calculation, but we rather provide a simple straight-forward expression as a proof of concept. For instance, if we calculate P_e using both the distance and the satellite's angular position to ground station as parameters, the DOC can represent the link quality more accurately. But DOC accuracy does not influence the performance of the proposed TCP sliding window algorithm and BPR, therefore DOC optimizations are outside the scope of this paper.

DOC is calculated using two probability functions: 1) *error probability* P_e as a function of BER and SNR, and 2) *collision possibility* P_c of outgoing packets from this node. A higher DOC value indicates more network congestion or package loss (1).

$$DOC = f(P_e, P_c), DOC \in (0, 1] \tag{1}$$

The errors related to noise and collision are equally important indications of link quality. Therefore the DOC is calculated as the weighted sum of probability of error P_e and collision P_c (2). For simplification, we use $W_e = W_c = 0.5$ in our simulation. A more careful selection of the two parameters may improve the accuracy of DOC. But as stated earlier, this accuracy does not influence the performance of the proposed mechanisms.

$$DOC = W_e \times P_e + W_c \times P_c \tag{2}$$

$P_c = N_c/N$ is the ratio of collided packets N_c and total packets in an observation period N. In Space, where we can safely assume that no obstacle stands on the path between the satellites, the distance is the dominating element of the error probability. Therefore, P_e is calculated using exponential distribution probability density function with the distance as a parameter:

$$P_e = f(x;\lambda) = \lambda \times e^{-\lambda x},\ \lambda = 1 \text{ and } x = Range_{max} - distance \tag{3}$$

In (3), $Range_{max}$ is the satellite's maximum communication range (distance to GS) in the order of 10^4 meters. This simulates that the error probability sharply increases when the distance approaches the maximum range (in the last 30 kilometers). P_e equals to 1 when the distance is larger than $Range_{max}$. The range of ISL can hardly exceed the distance between the GS and the satellite because the ground stations have much higher receiver gain than the satellites.

3.2 Balanced Predictable Routing

We propose a Balanced Predictable Routing (BPR) that emphasizes cross-node cooperation and cross-layer optimization within an individual node. First, the BPR uses predicted stability of a route to guarantee successful delivery. Second, the BPR provides load balancing for the entire network. Without a load balancing mechanism, the more stable routes are expected to be overloaded because the stability mechanism always selects them.

There are many ad hoc routing protocols such as Dynamic Source routing (DSR) [7], Ad-hoc On-demand Distance Vector (AODV) [8] and Temporally-Ordered Routing Algorithm (TORA) [9]. We develop the BPR based on DSR for the following reasons: (i)DSR is on-demand routing that does not use periodic messages to update the routing information. Consequently, it consumes less bandwidth and energy than table-driven (proactive) routing protocols. According to [10], DSR has smaller routing overhead than other protocols when the nodes never pause like the satellites. (ii) DSR records the complete route from source to destination. Therefore, the source node can optimize the route using all the intermediate links' information. (iii) The intermediate nodes also utilize the route cache information efficiently to reduce the control overhead. (iv) DSR does not maintain a routing table and consequently needs less memory space. (v) The LEO satellite network has limited hop-count (from one to three in our simulation). A simple node identifier instead of full IP address can be used in satellite networks. Both the limited hop-count and the simple identifier reduce the overhead in packet headers, which is the main disadvantage of the DSR.

In order to rank the routes to the same destination according to the link stability, we add a variable $S \in (0, 1]$ in the routing cache to indicate the stability of the route as shown in the algorithm below. The value of S equals to 1 when the route is most stable. The value of S decreases when the route becomes less stable. S is periodically updated during the time when the satellite travels between the two polar regions. More frequent update makes the link information more up-to-date, but leads to more computation overhead. We update S 50 times in our simulation. The value of S is calculated according to DOC and the availability of the route. If the satellite passes the polar region and starts moving in another direction, S is reset to 1 and the calculation starts over. The following pseudocode presents the algorithm to calculate the stability variable.

```
WHILE {traveling from one polar to the other}
    S = 1
    FOR {each observation time}
        IF {if route is available}
            S = S / (1 + DOC)
        ELSE {route is broken}
            S = S/2
        ENDIF
    ENDFOR
ENDWHILE
```

The stability variable is being constantly calculated when the satellite travels between the polar regions. The variable is reset to 1 in two polar region. The above actions are taken because: (i) some Micro LEO satellites go into standby mode, or even power off inside the polar region; (ii) the distance between satellites rapidly changes in the polar region and many entries in the routing table need to be recalculated; (iii) even without route entry reconstruction, the satellites may still be overloaded due to massive possible handoffs.

We use the link quality variable DOC instead of orbit information to predict future condition of routes because (i) the LEO satellites with different power and antenna capacity do not necessarily share a good communication channel even when they are close; (ii) the Global Positioning System (GPS) is not available on most micro LEO satellites.

With the stability variable, the BPR behaves differently from the original DSR in the following aspects: (i) The stability variable S is broadcasted along with the routing information in the route discovery package during the route discovery phase. (ii) Unlike the DSR that only stores one route to destination in the route cache, BPR stores multiple routes to the same destination. The route with highest stability S is selected in the route discovery phase. (iii) If a node considers itself overloaded, it drops the Route Request message in the route discovery phase. (iv)The reverse route is used to return the Route Reply message. The stability variable S is not attached to the Route Reply message. (v)All nodes overhear the broadcasted Route Request messages to update the stability variable of each route. (vi) In the Route Maintenance Phase, the erroneous hop is not removed from the node's route cache. Instead, the stability variable S of all routes containing the hop is updated. (vii) When the satellites pass the polar region, the route cache is cleaned. Therefore, all the routes are recalculated on-demand.

Considering the antenna and computing capacity of LEO satellites, we propose to use the stable intra-plane and inter-plane ISLs, when avoiding the fast changing cross-seam links. The proposed algorithm distinguishes well the more stable ISLs from the cross-seam ISLs and other fast changing ISLs, because the fast changing links always have smaller S. If the link quality of an ISL varies in a short range during observation period, it is going to maintain the quality level in the future. This is because the satellites move on orbits so that the satellite's relative position varies in a small range even when the satellites' absolute speed is very high. Eventually, such satellites can develop a constellation without orbit information of each other.

In order to achieve load balancing, we use the local and global view[1] [11] that is already provided by the cross-layer interaction architecture [6]. The node is overloaded if the ratio of local and global views is greater than 1 [11]. Remaining energy is also an important indications of the overload. Therefore, the satellites running out of power are considered overloaded.

3.3 QoS Aware TCP Congestion Control

The general approach to guarantee the delivery of high priority packets is the prioritized queuing at the IP or MAC layers. This approach, however, rearranges the order of the outgoing packets only. In a wireless network, the interference increases when the number of packets in media increases. This means the number of packets sent to the wireless media should also be controlled in order to reduce interference. Consequently, we propose a QoS aware TCP congestion control mechanism to reduce the number of outing packets when the high priority packets needs to be sent.

Our proposal dynamically controls the TCP sliding window size to reduce the wireless media interference and delay of high priority application such as the control command. According to the standard sliding window control algorithm, the window size is half of the original value when congestion happens in the stable phase. We borrow the idea to temporary reduce the network traffic for a very short time. The QoS aware sliding window control mechanism reduces the window size during the time when the satellite node is sending or relaying the command flow. The window size should be quickly reduced when link quality is bad or collision probability is high. On the other hand, if the link quality is high and collision probability is low, the window should be slowly reduced. Therefore, we use formula 4 to adjust the congestion window size.

$$Size_{window} = \frac{Size_{window}}{(1 + DOC^{e})}, e = NapierConstant \tag{4}$$

The sliding window size is reverse proportional to DOC. According to formula 4, the sliding window is divided by a value in the range between 1 and 2.

---

[1] The local view presents the queuing length of the node; The global view presents the average queuing length of the neighboring nodes.

We use the Napier constant e in DOC^e as shown in (4) so that the overall throughput of the network is not sharply reduced if the network has little congestion or other kinds of packet loss. In other words, a small DOC does not affect the behavior of the TCP sliding window.

4 Validation and Results

In order to validate our cross-layer optimizations, we implement the cross-layer optimizations in the Network Simulator 2 (ns-2) version 2.28 [12]. Our simulation is based on the ns-2 satellite package provided by [13]. We made the following improvements to the satellite package: (i) The energy model is introduced to simulate the satellite's behavior without the energy source (in the shadow of the earth); (ii) 802.11 MAC like collision model is introduced to calculate the collision probability in satellite network; (iii) The centralized routing of satellite package is replaced by BPR.

4.1 Simulation Scenario

We use the following configuration in our simulation: 1 to 17 Satellite nodes on random polar orbits (altitude 500-800km) within 5 degrees deviation with random start elevation degree (based on longitude 4.0 E); GSL from 500kbps to 2Mbps for each satellite; ISL from 1Mbps to 2Mbps; both symmetric links; Two Ground stations: A (in Delft 51.9792 N, 4.375 E): B (New York 40.30N, 73.24W); Data Sources: 10 CBR on UDP simulates the realtime data from GS A to B, 20 FTP on TCP simulates non-realtime data from satellites to GS, FTP to simulate control commands sent from ground stations to satellites when there is a connection; duration: 1 day (86400s).

4.2 Performance Analysis

Assuming T_i is the time when satellite i is connected to GS and T_{total} is the total fly time of all satellites, the coverage is defined as $\frac{\sum T_i}{T_{total}}$. As shown in figure 2, the coverage improvement mainly happens when there are 5 to 11 satellites in the formation. In this case, the margin effect of the elevation mask has a strong impact on the links' availability. The margin effect is caused by interference from atmosphere and the relative position between the satellite and satellite/GS. When the satellites start moving into or leaving the elevation mask, the link quality changes very rapidly. The cross-layer design using MAC and PHY layer feedbacks can predict the satellites that are falling out of line of sight and consequently switch to other satellites for communication in advance. The gain of cross-layer optimization decreases when there are more satellites in the network. This is because multiple routes are available when multiple ISLs are above the elevation mask. Consequently the margin effect is avoided by switching to other links.

We compare the successful delivery in term of throughput percentage w/o cross-layer designs. Both the individual cross-layer designs and their combination

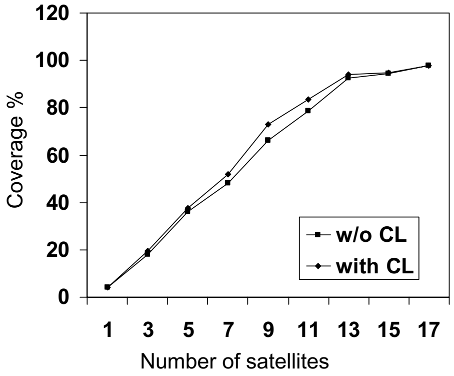

Fig. 2. coverage comparison with and without CL design

are simulated. In figure 3, “CL 1” is the integrated MAC/PHY layer; “CL 2” is the BPR; and “CL 3” is QoS aware TCP sliding window control. The throughput percentage is improved when we use the integration of the three optimizations because (i) the BPR distributes the traffic more evenly in the network, which leads to less congestion on route with better link quality; (ii) the stability variable algorithm ensures that fast changing ISLs and satellites in energy conservation mode are avoided in the route.

The cross-layer design interaction architecture insures that the system benefits from all cross-layer designs. In figure 3, the “CL 3” is important to improve the delivery of command data then the network has higher traffic load (13 to 17 nodes). This optimization has little impact on the realtime and non-realtime traffics when the network load is low. This validates our design that it should not affect low priority services when the network is not congested. The result in figure 3 also shows that the delivery percentage is lowest when the constellation consists of 7 satellites. This is because the single path, however, is unstable due to the margin effect of the elevation mask and the fast changing distance. When the number of satellites increases, multiple routes are available at the same time. Consequently, the delivery failure is reduced.

The delivery percentage in figure 3 is high ($\geq 88\%$) because the satellite-GS link is in optimized status. And due to bandwidth constraints, the TCP sliding window is always small, which also leads to reduced network congestion. This, however, is not true in real environment. In the real environment, the ISLs can be asymmetric links with various bandwidth, which leads to significant packet lost. This cannot be simulated in the ns-2 simulator.

Table 1 compares the average number of forwarded packets per satellite and its standard deviation. The results show that the cross-layer architecture not only improves the throughput, but also distributes the traffic more evenly in the network. Considering that most of satellite’s energy is consumed by the telecommunication system, we reduce the chance of exhausting some satellites while leaving others full of energy.

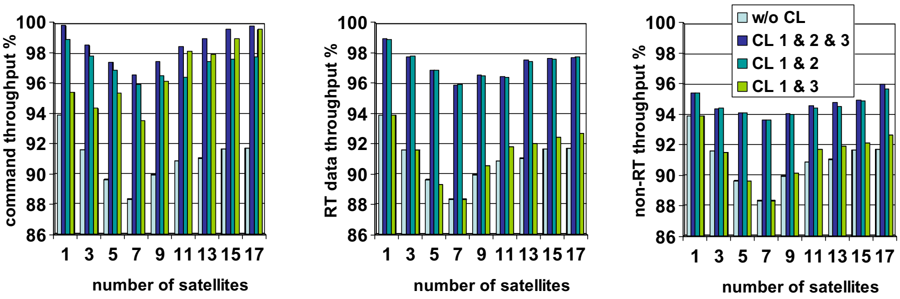

Fig. 3. The throughput % of prioritized traffics

Table 1. Average and standard deviation of number of forwarded packets per satellite

sat.	5	7	9	11	13	15	17
w/o CL average	13577.4	17497.3	19257.6	20431.9	20954.8	21314.5	21853.6
w/o CL deviation	1185.1	953.7	2249.7	1835.3	1973.8	1745.3	2034.6
with CL average	14671.2	18616.7	20266.2	21557.2	21574.6	22054.8	22612.9
with CL deviation	232.0	516.1	1024.0	966.5	1054.3	1234.7	1095.3

Our proposal also improves the end-to-end delay of packets as shown in figure 4 because the MAC layer retransmission is reduced by using more stable links. Our proposal, however, may have negative effect on the delay for the following reasons. First, the load balancing algorithm pushes some packets to the edge of the network in order to reduce traffic in its center. This action increases the hop count of those packets. Second, the BPR always prefers to use the more stable links, which increases the queuing length of nodes with good link quality.

4.3 Overhead Analysis

The proposed cross-layer designs introduce internal overhead within a node as well as external overhead on the network. The internal and external overhead is calculated as the number of bytes added to normal packets to carry cross-layer information. The internal overhead consists of two parts: the overhead of the individual cross-layer design and the overhead of the architecture to enable correct interaction among multiple designs. The three cross-layer designs and the interaction architecture together introduce an internal overhead less than 0.25%. Therefore, the internal overhead is neglectable. In our proposal, the external overhead is the one-hop neighbors' information used to calculate the global view in order to achieve load balancing. Because the satellites have very limited one-hop neighbors (between one and three in our simulation) the external overhead is low (below 2%). Figure 5 depicts the internal and external overheads of our proposal.

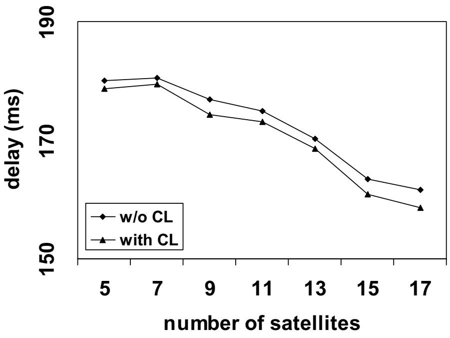

Fig. 4. The average packet delay from GS A to GS B

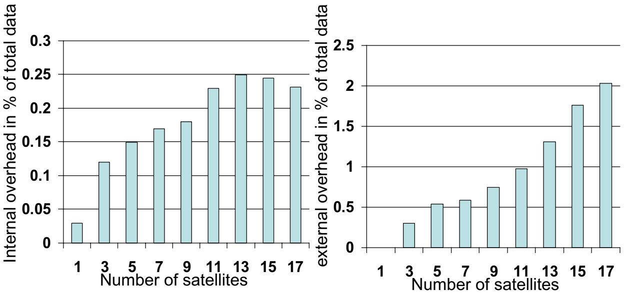

Fig. 5. The internal and external overheads of cross-layer designs

5 Related Works

Previous research shows cross-layer designs are promising for QoS control in the MANET. In [14], a cross-layer framework for WLAN QoS support is proposed. The authors show that QoS at MAC layer can be optimized by taking advantage from the IP, TCP and application layers. In [15], a combined cross-layer design for QoS content delivery is proposed. The authors introduced a QoS-aware scheduler and power adaptation scheme at the MAC layer for an efficient resource utilization in the upper layers. Their results show that the cross-layer design provides a good scheme for wireless QoS content delivery. These works can not be directly compared to our proposal because our proposal considers the characteristics and special requirements of the satellite networks instead of MANET. The above works provide the general approach of QoS optimization that involves all layers in the protocol stack.

Previous works also focus on MANET higher layer optimization using information from MAC layer and below. In the traditional wired communication world, the bit error rate (BER) of the link can be neglected. The TCP layer assumes that the package loss is an indication of congestion. In the wireless world, however, the package loss is mainly caused by loss on the wireless link instead of congestion. Many papers have analyzed this problem and proposed solutions such as using link connectivity to notify the TCP protocol if congestion really happens [16] and [17]. Vania Conan et al. proposed in [18] the WIDENS architecture. This architecture emphasizes the low-level protocol integration by virtually providing a super low layer that combines the DLC, MAC and PHY layers. Special communication channel between the network layer and integrated low layer is also established in order to support hard QoS routing in Mobile Ad hoc NETwork (MANET). In [11], Rolf Winter et al. proposed the CrossTalk architecture. Unlike the above designs, the CrossTalk architecture emphasizes cross-node cooperation as well as cross-layer design within individual node. The above works provide guidance for the proposed BPR and QoS aware TCP sliding window control mechanism.

6 Conclusion and Future Work

Future LEO satellite networks are expected to have a dynamic topology and become ad hoc networks. The current architecture of the LEO satellite networks, however, cannot fulfill the requirements for such an fast changing network environment. In this paper, we first discussed the special QoS requirements and the inherit problems of the LEO satellite networks. Then, we proposed two cross-layer designs, namely BPR and QoS aware TCP sliding window control, both using information from an integrated MAC/PHY layer. The BPR improved the total throughput while considering the load balancing at the same time. The QoS aware TCP mechanism guaranteed the delivery of high priority services while avoiding unnecessary decrease of the total throughput. The end-to-end delay was also reduced because BPR reduced the MAC layer retransmission by selecting the links with better quality. In the future, we will continue designing and simulating the cross-layer optimizations such as the energy consumption control in the LEO satellite network environment. This will provide more systematic solutions for the future satellite networks.

Acknowledgments. This work is supported by the Dutch Government as part of the decree on subsidies for investments in the knowledge infrastructure (Bsik) program. This work is done within the Micro Satellite (MISAT) project.

References

1. Prescott, G., Smith, S., Moe, K.: Real-time information system technology challenges for nasa's earth science enterprise. In: 20th IEEE Real-Time Systems Symposium, Phoenix, AZ., U.S.A. (December 1999)
2. Shen, C.C., Rajagopalan, S., Borkar, G., Jaikaeo, C.: A flexible routing architecture for ad hoc space networks. Computer Networks 46(3), 389–410 (2004)
3. Giambene, G., Chini, P.: Introduction to satelliste communications and resource management. In: chapter of Resource Management in Satellite Networks: Optimization and Cross-Layer Design (March 2007)

4. Kawadia, V., Kumar, P.R.: A cautionary perspective on cross layer design. In: IEEE Wireless Commun., February 2005, vol. 12(1), pp. 3–11. IEEE Computer Society Press, Los Alamitos (2005)
5. Srivastava, V., Motani, M.: The road ahead for cross-layer design. In: Proceedings of 2005 2nd International Conference on Broadband Networks, pp. 551–556. IEEE, Los Alamitos (2005)
6. Chang, Z., Gaydadjiev, G.N., Vassiliadis, S.: Infrastructure for cross-layer designs interaction. In: the 16th IEEE International Conference on Computer Communications and Networks (IC3N), August 2007, pp. 19–25 (2007)
7. Johnson, Maltz, Hu.: The dynamic source routing protocol for mobile ad hoc networks (dsr). In: IETF Draft (April 2003)
8. Perkins, C., Belding-Royer, E., Das, S.: Ad hoc on-demand distance vector (aodv) routing. In: Internet Engineering Task Force (IETF) draft (July 2003)
9. Park, V.D., Corson, M.S.: Temporally-ordered routing algorithm (tora) version 1: functional specification. In: Internet Engineering Task Force (IETF) draft (November 1997)
10. Broch, J., Maltz, D.A., Johnson, D.B., Hu, Y.C., Jetcheva, J.: A performance comparison of multi-hop wireless ad hoc network routing protocols. In: Mobile Computing and Networking (MobiCom), pp. 85–97 (1998)
11. Winter, R., Schiller, J.H., Nikaein, N., Bonnet, C.: Crosstalk: cross-layer decision support based on global knowledge. In: Communications Magazine, vol. 44(1), pp. 93–99. IEEE, Los Alamitos (2006)
12. ns 2: The network simulator version 2, `http://www.isi.edu/nsnam/ns/`
13. Henderson, T.R., Katz, R.H.: Network simulation for leo satellite networks. In: American Institute of Aeronautics and Astronautics (2000)
14. Pau, G., Maniezzo, D., Das, S., Lim, Y., Pyon, J., Yu, H., Gerla, M.: Cross-layer framework for wireless lan qos support. In: the IEEE International Conference on Information Technology Research and Education (ITRE) (2003)
15. Chen, J., Lv, T., Zheng, H.: Joint cross-layer design for wireless qos content delivery. In: IEEE International Conference on Communication (2004)
16. Raisinghani, V.T., Singh, A.K., Iyer, S.: Improving tcp performance over mobile wireless environments using cross layer feedback. In: IEEE International Conference on Personal Wireless Communications, New Delhi, India (2002)
17. Shakkottai, S., Rappaport, T.S., Karlsson, P.C.: Cross layer design for wireless networks. In: IEEE Commun. Mag., October 2003, pp. 74–80. IEEE, Los Alamitos (2003)
18. Aiache, H., Conan, V., Barcelo, J.M., etc., L.C.: Widens: Advanced wireless ad-hoc networks for public safety. IEEE Computer Society Press, Los Alamitos (2005)