Research Article

A Virtual Channels Scheduling Algorithm with Broad Applicability Based on Movable Boundary

Ye Tian, Qingfan Li, Yongxin Feng, and Xiaoling Gao

School of Information Science and Engineering, Shenyang Ligong University, ShenYang 110159, China

Correspondence should be addressed to Yongxin Feng; fengyongxin@263.net

Received 22 May 2013; Accepted 6 August 2013

Academic Editor: Baochang Zhang

Copyright © 2013 Ye Tian et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A virtual channels scheduling algorithm with broad applicability based on movable boundary is proposed. According to the types of data sources, transmission time slots are divided into synchronous ones and asynchronous ones with a movable boundary between them. During the synchronous time slots, the virtual channels are scheduled with a polling algorithm; during the asynchronous time slots, the virtual channels are scheduled with an algorithm based on virtual channel urgency and frame urgency. If there are no valid frames in the corresponding VC at a certain synchronous time slot, a frame of the other synchronous VCs or asynchronous VCs will be transmitted through the physical channel. Only when there are no valid frames in all VCs would an idle frame be generated and transmitted. Experiments show that the proposed algorithm yields much lower scheduling delay and higher channel utilization ratio than those based on unmovable boundary or virtual channel urgency in many kinds of sources. Therefore, broad applicability can be achieved by the proposed algorithm.

1. Introduction

To meet the requirements of new space systems and new space missions, Consultative Committee for Space Data Systems (CCSDS) expanded the contents of the conventional recommendations [1, 2] and put forward Advanced Orbiting Systems (AOS) recommendations [3, 4], which provide flexible and various data processing businesses. In AOS, a two-layer multiplexing mechanism, which includes packet multiplexing and virtual channel multiplexing mechanisms, is used to share one physical channel between multiusers. Packet channel multiplexing mechanism is a mechanism in which various data packets share one virtual channel (VC), and some achievements have been made in this respect [5–8]. Virtual channel multiplexing mechanism is a mechanism in which a number of virtual channels share one physical channel.

Virtual channels are a group of logic channels formed by dividing the physical channel into different time slots. Each VC transmits the user data with the same or similar properties. The specific algorithm used in this mechanism, namely, VC scheduling algorithm, decides the order of VCs

occupying the physical channel as well as the transmission efficiency of the physical channel. Therefore, the way to design the VC scheduling algorithm will be the focus of AOS systems.

Some research conclusions on virtual channel scheduling algorithms have been achieved [9–11], most of which are designed for a certain satellite data source, so their application scopes are limited. Even the dynamic scheduling algorithm (DSA) [11], which provides good performance for some types of data sources, could hardly meet the transmission requirements of all types of satellite data sources. In addition, it pays little attention to the difference between VC urgency and frame urgency, which will cause performance reduction.

To address the problems above, a virtual channels scheduling algorithm with broad applicability based on movable boundary is proposed, in which the transmission time slots are divided into synchronous ones and asynchronous ones; furthermore, the boundary between them is movable. During the synchronous time slots, the virtual channels are scheduled with a polling algorithm. If there are no valid frames in the corresponding synchronous VC at a certain synchronous time slot, a frame of the other synchronous VCs

FIGURE 1: The proposed scheduling algorithm.

or asynchronous VCs will be selected to occupy this time slot according to the scheduling mechanism; if there are no valid frames in all VCs, an idle frame is generated and transmitted. During the asynchronous time slots, the virtual channels are scheduled with an algorithm based on VC urgency and frame urgency. Simulation results show that the proposed algorithm produces much less scheduling delay and higher channel utilization ratio than the DSA algorithm.

2. Scheduling Algorithm with Broad Applicability Based on Movable Boundary

There are three kinds of virtual channels multiplexing modes [12]: the full synchronous multiplexing mode, the full asynchronous multiplexing mode, and the synchronous/asynchronous multiplexing mode. For the full synchronous multiplexing mode, the order of virtual channels occupying

FIGURE 2: Asynchronous VCs scheduling algorithm.

a physical channel is fixed and repetitive, so it is suitable for the case in which most business data are synchronous and data rates of all the VCs are equivalent. However, it is not suitable for handling burst business and cannot achieve high channel utilization ratio. For the full asynchronous multiplexing mode, when two or more virtual channels are waiting to be scheduled, they are scheduled according to the priority assigned or calculated; therefore, this mode is suitable for burst business and low data rate information. However, this mode usually results in the delay wobbling of synchronous data [12]. For the synchronous/asynchronous multiplexing mode, the virtual channels are divided into synchronous and asynchronous ones. Synchronous virtual channels are scheduled with the full synchronous multiplexing mode, while asynchronous virtual channels are scheduled with the full asynchronous multiplexing mode. Although the system complexity is increased in this mode, the shortcomings of the first two modes mentioned above can be avoided.

In order to reduce the scheduling delay and improve the channel utilization ratio, a virtual channels scheduling algorithm based on movable boundary is proposed. In the proposed algorithm, the synchronous/asynchronous

multiplexing mode is adopted. Time slots are divided into synchronous and asynchronous ones, which are used to schedule synchronous and asynchronous virtual channels, respectively. The system distributes the number of synchronous and asynchronous time slots according to the VC data rates and the number of VCs [12]. In terms of the scheduling strategy, the polling scheduling strategy is used for synchronous virtual channels, while the virtual channel scheduling strategy based on VC urgency and frame urgency is used for asynchronous virtual channels. The flowchart of the proposed scheduling algorithm is shown in Figure 1.

2.1. Synchronous Virtual Channel Scheduling Algorithm. For synchronous virtual channels, the polling scheduling algorithm is used. The so-called polling scheduling algorithm is that each virtual channel occupies the physical channel exactly according to the order assigned before. If there are no valid frames in the corresponding VC at a certain synchronous time point, the frame with the longest waiting time from the other synchronous VCs will be transmitted; if two or more frames from the other synchronous VCs have

FIGURE 3: The average time delay of each VC of source model 1.

the longest waiting time, the first frame in the synchronous VC with the smallest VC serial number among them is transmitted; if there are no valid frames in all synchronous VCs, a valid asynchronous frame will be selected and transmitted according to the asynchronous scheduling mechanism. Otherwise, an idle frame will be generated and transmitted through the physical channel.

Define the waiting time of the current first frame in the i th VC (VC_i), T_{wi} , as:

$$T_{wi} = k\Delta t - T_{ia} \quad k = 1, 2, 3, \dots, \quad (1)$$

where $k\Delta t$ is the current scheduling time point, and T_{ia} is the arrival time of the current first frame in VC_i .

2.2. Asynchronous Virtual Channel Scheduling Algorithm. For asynchronous virtual channels, the scheduling algorithm based on VC urgency and frame urgency is exploited. First, the VC urgency and frame urgency are calculated, respectively, and then they are balanced with a weighting coefficient

β to form a new urgent degree function of VCs. VCs are scheduled according to the values of the function.

2.2.1. VC Urgency. Define the transmission urgency of VC_i , $U_{li}(k)$, as

$$U_{li}(k) = a_i(k) \cdot Q_i(k) \cdot p_i, \quad (2)$$

where $a_i(k)$ is the Boolean function, and it decides whether VC_i can participate in the competition of the physical channel. The values of $a_i(k)$ are shown as

$$a_i(k) = \begin{cases} 1, & \text{there are some valid frames in } VC_i \\ & \text{at the current scheduling time point} \\ 0, & \text{there are no valid frames in } VC_i \\ & \text{at the current scheduling time point.} \end{cases} \quad (3)$$

FIGURE 4: The maximum time delay of each VC of source model 1.

$Q_i(k)$ is the urgency base of VC_i , and it calculates the delay from the last time point when VC_i is scheduled to the current time point. The values of $Q_i(k)$ are shown as

$$Q_i(k) = \begin{cases} 0, & \text{there are some valid frames in } VC_i \\ & \text{and } VC_i \text{ is scheduled at the } k-1 \\ & \text{time point} \\ Q_i(k-1), & \text{there are some valid frames in } VC_i \\ & \text{and } VC_i \text{ is not scheduled at the } k-1 \\ & \text{time point} \\ Q_i(k-1) + 1, & \text{there are no valid frames in } VC_i \\ & \text{at the } k-1 \text{ time point.} \end{cases} \quad (4)$$

p_i is the static priority of VC_i . If VC_i requests higher real-time performance, a larger value of p_i is assigned to it; otherwise, a smaller one is assigned.

2.2.2. Frame Urgency. Frame urgency can be calculated by the time that the current first frame has been waiting for. Define the frame urgency $U_{2i}(k)$ of VC_i as

$$U_{2i}(k) = a_i(k) \cdot \frac{(k\Delta t - T_{ia})}{\Delta t} \cdot p_i, \quad (5)$$

where T_{ia} is the arrival time of the current first frame in VC_i . Obviously, the smaller the value of T_{ia} is, the longer the waiting time of the current first frame is, which results in a larger value of $U_{2i}(k)$.

2.2.3. Urgent Degree Function of Virtual Channel. Define the urgent degree function of VC_i at the k th scheduling time point as

$$U_i(k) = \beta \cdot U_{1i}(k) + (1 - \beta) \cdot U_{2i}(k), \quad (6)$$

where β is the weighting coefficient, $0 < \beta < 1$.

FIGURE 5: The channel utilization ratio of source model 1.

FIGURE 6: The average time delay of system of source model 1.

Substituting (5) into (6), we can get

$$U_i(k) = a_i(k) \cdot \left[\beta \cdot Q_i(k) + \frac{(1 - \beta) \cdot (k\Delta t - T_{ia})}{\Delta t} \right] \cdot p_i. \quad (7)$$

As is shown in (7), both the VC urgency and the frame urgency are estimated when calculating the urgent degree functions of VCs. On one hand, this will prevent the VC with large amount of data from monopolizing the physical channel; on the other hand, it overcomes the problem in the literature [11–14] that the waiting time of the other data

FIGURE 7: The maximum time delay of system of source model 1.

frames in a VC are reset when the first frame in this VC is transmitted, which will result in the decline of the VC urgency.

At any asynchronous scheduling time point, the VC with the largest value of $U_i(k)$ is scheduled; if there are two or more VCs with the largest value of $U_i(k)$, the VC with the largest values of $U_i(k)$ and p_i is scheduled; if there are two or more VCs with the largest values of $U_i(k)$ and p_i , the VC with the smallest VC serial number among them is scheduled. Otherwise, an idle frame is generated and transmitted through the physical channel. The flowchart of asynchronous VCs scheduling algorithm is shown in Figure 2.

2.2.4. Strategy of Selecting Weighting Coefficient. For different types of satellite data sources, we can select an appropriate β by computer simulation to get good urgent degree functions of VCs so as to achieve good scheduling performance. Therefore, the proposed algorithm can be applied to diverse satellite data sources.

Consider the following three kinds of satellite data sources.

- Sources with large difference among VC loads. For this kind of sources, there is usually one or more VCs with large amount of data, whose number of data frames is much larger than the others. For VCs with large amount of data, the values of frame urgency are relatively large and the values of VC urgency are relatively small at most scheduling time points; for VCs with little amount of data, whose time of occupying physical channel is limited, the values of VC urgency are relatively large and the values of frame urgency are relatively small at most scheduling time points. In order to prevent VCs with large amount of

FIGURE 8: The average time delay of each VC of source model 2.

data from monopolizing the physical channel, we can select large values for β .

- (b) Sources with moderate difference among VC loads. For this kind of sources, there are no VCs with enough data to monopolize physical channel, so the value of β should be smaller than that of the first kind of sources. For the VCs with high data rates, the data frame urgency is dominant, so the smaller the value of β is, the better the delay performance is; for the VCs with low data rates, the VC urgency is dominant, so the larger the value of β is, the better the delay performance is. The right value of β can be determined by VC loads and real-time requirements.
- (c) Sources with little difference among VC loads. For this kind of sources, data rates of various VCs are relatively equivalent. Neither the VC urgency nor the frame urgency is dominant in the urgent degree function of VC, so the value of β has little effect on the performance of VCs scheduling delay, which makes it easy to be selected.

3. Simulation Results

Simulation is performed to compare the performance of the novel algorithm we proposed with those of the SAUB algorithm and the DSA algorithm. It is worth mentioning that the SAUB algorithm is the algorithm in which the same scheduling strategies as those used in the proposed algorithm are adopted, except that the boundary between the synchronous time slots and asynchronous ones is unmovable. Performance parameters include mainly the average scheduling delay of each VC, the maximum scheduling delay of each VC, the average scheduling delay of system, and the maximum scheduling delay of system and the channel utilization ratio.

3.1. Simulation on Sources with Large Difference among VC Loads. For this kind of sources, a source is selected whose VCs' division and parameter settings are shown in Table 1. VC1~VC6 are asynchronous VCs as that in the literature [15]. VC7 and VC8 are synchronous VCs. In addition, VC9 is used to generate and transmit idle frames. The highest data rate

FIGURE 9: The maximum time delay of each VC of source model 2.

TABLE 1: Source with large difference among VC loads.

	VC1	VC2	VC3	VC4	VC5	VC6	VC7	VC8	VC9
Type of source	High data rate source	High data rate source	Middle data rate source	High data rate source	Middle-low data rate source	Low data rate source	High data rate source	High data rate source	Idle frame
Type of service	Path service	Path service	Path service	Path service	Path service	Multiplexing service	Path service	Path service	—
Data rate (Mbps)	15	1.3	0.1	1.5	0.06	0.05	3.3	1.1	—

(VC1) is 300 times as large as the lowest data rate (VC6). The other simulation parameters are set as follows: (1) the length of data frame is 6000 bits; (2) downlink rate ranges from 2.4×10^7 bps to 2.5×10^7 bps; (3) the static priority values of VCs are $p_1 = 1.1$, $p_2 = 1.2$, $p_3 = 1.3$, $p_4 = 1.4$, $p_5 = 1.5$, and $p_6 = 1.6$, respectively.

Simulation results are shown in Figures 3, 4, 5, 6, and 7.

It is shown from the previous figures that for all the VCs, performance of the scheduling delay and the channel

utilization ratio achieved by the proposed algorithm are better than those achieved by the SAUB algorithm, no matter what the value of β is. For example, when the downlink rate is 2.46 Mbps and β is 0.98, the average scheduling delay of VC3, the maximum scheduling delay of VC3, and the channel utilization ratio achieved by the SAUB algorithm are 2.093×10^{-4} s, 8.671×10^{-4} s, and 0.9432, respectively, while those achieved by the proposed algorithm are 1.983×10^{-4} s, 8.485×10^{-4} s, 0.9447, respectively. Therefore, the average scheduling

FIGURE 10: The channel utilization ratio of source model 2.

FIGURE 12: The maximum time delay of system of source model 2.

FIGURE 11: The average time delay of system of source model 2.

delay and the maximum scheduling delay of VC3 achieved by the proposed algorithm are 5.26% and 2.15% lower than those achieved by the SAUB algorithm, respectively, and the channel utilization ratio achieved by the proposed algorithm is 0.15% higher than that achieved by the SAUB algorithm. It is also shown that for most of the VCs, when the value of β is large, performance achieved by the proposed algorithm is better than that achieved by the DSA algorithm. For example, when the downlink rate is 2.46 Mbps and β is 0.98, the average scheduling delay of VC3, the maximum scheduling delay of VC3, and the channel utilization ratio achieved by the DSA algorithm are 2.664×10^{-4} s, 1.001×10^{-3} s, and 0.9442, respectively. Therefore, the average scheduling delay and the maximum scheduling delay of VC3 achieved by

the proposed algorithm are 25.56% and 15.25% lower than those achieved by the DSA algorithm, respectively, and the channel utilization ratio achieved by the proposed algorithm is 0.05% higher than that achieved by the DSA algorithm.

When the value of β is small, the performance of the proposed algorithm declines. This is consistent with our theoretical analysis. For all the VCs, scheduling delay varies with the change of β . In practice, we can select an appropriate value of β to meet the delay requirements of all VCs without increasing the downlink rate. Obviously, such an advantage cannot be achieved by the DSA algorithm.

3.2. Simulation on Sources with Moderate Difference among VC Loads. For this kind of sources, manned spacecraft source model [16] is used, whose VCs' division and parameter settings are shown in Table 2. VC1~VC6 are asynchronous VCs. VC7 and VC8 are synchronous VCs. In addition, VC9 is used to generate and transmit idle frames. The highest data rate (VC6) is 47.5 times as large as the lowest data rate (VC4). The other simulation parameters are set as follows: (1) the length of data frame is 8000 bits; (2) downlink rate ranges from 2.85×10^8 bps to 3.0×10^8 bps; (3) the static priority values of VCs are $p_1 = 1.6$, $p_2 = 1.5$, $p_3 = 1.4$, $p_4 = 1.3$, $p_5 = 1.2$, and $p_6 = 1.1$, respectively.

Simulation results are shown in Figures 8, 9, 10, 11, and 12.

Similar to model 1, for most of the VCs, the performance of the proposed algorithm is better than that of the SAUB algorithm and that of the DSA algorithm, no matter what the value of β is. For example, when the downlink rate is 2.91 Mbps and β is 0.5, the average scheduling delay of VC3, the maximum scheduling delay of VC3, and the channel utilization ratio achieved by the SAUB algorithm are 1.123×10^{-4} s, 4.358×10^{-4} s, and 0.9691, respectively, those achieved by the DSA algorithm are 1.339×10^{-4} s, 6.734×10^{-4} s, and 0.9680, respectively, and those achieved by the proposed

FIGURE 13: The average time delay of each VC of source model 3.

TABLE 2: Manned spacecraft source model.

	VC1	VC2	VC3	VC4	VC5	VC6	VC7	VC8	VC9
Type of source	Platform network data	Physiological telemetry data	Delayed telemetry data	Internet data	Test data	CCD image data	Video surveillance data	Video conference data	Idle frame
Type of service	Path service	Path service	Path service	Internet service	Path service	Bit-stream service	Bit-stream service	Bit-stream service	—
Data rate (Mbps)	10	10	35	2	75	95	44	11	—

algorithm are 9.287×10^{-5} s, 3.459×10^{-4} s, and 0.9692, respectively. Therefore, compared with those achieved by the SAUB algorithm, the average scheduling delay and the maximum scheduling delay of VC3 achieved by the proposed algorithm are 17.30% and 20.63% lower, respectively, and the channel utilization ratio achieved by the proposed algorithm

is 0.01% higher; compared with those achieved by the DSA algorithm, the average scheduling delay and the maximum scheduling delay of VC3 achieved by the proposed algorithm are 30.64% and 48.63% lower, respectively, and the channel utilization ratio achieved by the proposed algorithm is 0.12% higher.

FIGURE 14: The maximum time delay of each VC of source model 3.

For the VCs with high data rates (e.g., VC6), the smaller the value of β is, the shorter the scheduling delay is; for the VCs with lower data rates (e.g., VC3), the larger the value of β is, the shorter the scheduling delay is, which is consistent with theoretical analysis.

3.3. Simulation on Sources with Little Difference among VC Loads. For this kind of sources, a source is selected whose VCs' division and parameter settings are shown in Table 3. VC1~VC6 are asynchronous VCs as shown in the literature [17]. VC7 and VC8 are synchronous VCs. In addition, VC9 is used to generate and transmit idle frames. In order to make the experimental results reflect balanced loads in general sense, the data rates of VC1~VC8 are set around 20 Mbps, but they are slightly different. The other simulation parameters are set as follows: (1) the data frame length is 6000 bits; (2) downlink rate ranges from 1.6×10^8 bps to 1.75×10^8 bps;

(3) the static priority values of VCs are $p_1 = 1.1$, $p_2 = 1.2$, $p_3 = 1.3$, $p_4 = 1.4$, $p_5 = 1.5$, and $p_6 = 1.6$, respectively.

Simulation results are shown in Figures 13, 14, 15, 16, and 17.

Similar to source model 1 and model 2, for most of the VCs, the performance of the proposed algorithm is better than that of the SAUB algorithm and the DSA algorithm, no matter what the value of β is. For example, when the downlink rate is 1.63 Mbps and β is 0.1, the average scheduling delay of VC3, the maximum scheduling delay of VC3, and the channel utilization ratio achieved by the SAUB algorithm are 4.882×10^{-4} s, 1.978×10^{-3} s, and 0.9833, respectively; those achieved by the DSA algorithm are 8.443×10^{-4} s, 5.943×10^{-3} s, and 0.9827, respectively, and those achieved by the proposed algorithm are 4.204×10^{-4} s, 1.889×10^{-3} s, and 0.9833, respectively. Therefore, compared with those achieved by the SAUB algorithm, the average scheduling delay and the maximum scheduling delay of VC3 achieved by the

TABLE 3: Source with little difference among VC loads.

	VC1	VC2	VC3	VC4	VC5	VC6	VC7	VC8	VC9
Type of source	High data rate source	High data rate source	High data rate source	High data rate source	High data rate source	High data rate source	High data rate source	High data rate source	Idle frame
Type of service	Path service	Path service	Path service	Path service	Path service	Path service	Path service	Path service	—
Data rate (Mbps)	20	21	19.5	19	22	18.5	19	21	—

FIGURE 15: The channel utilization ratio of source model 3.

FIGURE 17: The maximum time delay of system of source model 3.

FIGURE 16: The average time delay of system of source model 3.

proposed algorithm are 13.89% and 4.50% lower, respectively; compared with those achieved by the DSA algorithm, the average scheduling delay and the maximum scheduling delay of VC3 achieved by the proposed algorithm are 50.21% and 68.21% lower, respectively, and the channel utilization ratio achieved by the proposed algorithm is 0.06% higher. Because data rates of all the VCs are equivalent, the value of β has relatively little effect on the performance of scheduling delay.

4. Conclusions

We proposed a novel algorithm for virtual channel scheduling based on movable boundary, in which time slots are divided into synchronous ones and asynchronous ones. During the synchronous time slots, the polling scheduling algorithm is used, while during the asynchronous ones, the VC scheduling algorithm based on VC urgency and frame urgency is used. When there are no valid synchronous data frames in the corresponding VC at a certain synchronous time slot, a frame of the other synchronous VC or asynchronous VCs will be transmitted. If there are no valid frames in all VCs, an idle frame will be generated and transmitted. Compared with

the SAUB algorithm, the proposed algorithm can reduce the scheduling delay and improve the channel utilization ratio, while compared with the DSA algorithm, the proposed algorithm is much better as far as the time delay and applicability scope are concerned. In addition, the proposed algorithm is suitable for the scheduling of diverse data sources.

Acknowledgments

This work was supported by the Natural Science Foundation of China (61101116), the Outstanding Young Scholar Plan of Liaoning Province (LJQ2012018), the Innovative Team Project of Liaoning (LT2011T005), and the Open Foundation of Key Laboratory of Shenyang Ligong University (4771004kfs06).

References

- [1] Space Packet Protocol, *Recommendation For Space Data System Standards, CCSDS 133.0-B-1*, Blue Book, Issue 1, CCSDS, Washington, DC, USA, 2003.
- [2] Overview of Space Link Protocols, *Consultative Committee For Space Data System. CCSDS 130. 0-G-1*, Draft Green Book, CCSDS, 2004.
- [3] AOS space data link protocol, *Recommendation For Space Data System Standards, CCSDS 732.0-B-1*, Blue Book, Issue 1, CCSDS, Washington, DC, USA, 2003.
- [4] AOS space data link protocol, *Recommendation For Space Data System Standards, CCSDS 732.0-B-2*, Blue Book, Issue 2, CCSDS, Washington, DC, USA, 2006.
- [5] Y. Tian, X. Na, Y. Xia, X.-L. Gao, and L.-S. Liu, "Research on the performance and application of adaptive frame generation algorithm in AOS protocol," *Journal of Astronautics*, vol. 33, no. 2, pp. 242–248, 2012.
- [6] Y. Tian, C.-S. Pan, Z.-J. Zhang, and Y.-Q. Zhang, "Research on adaptive frame generation algorithm in AOS protocol," *Journal of Astronautics*, vol. 32, no. 5, pp. 1171–1178, 2011.
- [7] Y. Tian, D. Zhang, Z. Tan, C. Pan, and X. Gao, "Research on the packet time delay of frame generation algorithms in advanced orbiting systems," *Chinese High Technology Letters*, vol. 21, no. 11, pp. 1121–1128, 2011.
- [8] Y.-T. Zhao, C.-S. Pan, Y. Tian, Z.-J. Zhang, and M.-X. Bi, "Research on multiplexing efficiency of MPDU based on CCSDS advanced orbiting systems," *Journal of Astronautics*, vol. 31, no. 4, pp. 1195–1199, 2010.
- [9] Y. Wen, *Study of Advanced Orbiting System Multiplexer*, Inner Mongolia University of Technology, Hohhot, China, 2007.
- [10] Y. Q. Gu and W. Z. Tan, "CCSDS downlink virtual channel schedule and performance analysis," *Chinese Space Science and Technology*, vol. 21, pp. 29–35, 2001.
- [11] A. P. Riha and C. Okino, "An advanced orbiting systems approach to quality of service in space-based intelligent communication networks," in *Proceedings of the 27th IEEE Aerospace Conference*, Big Sky, March 2006.
- [12] Y.-X. Bie, C.-S. Pan, and R.-Y. Cai, "Research and simulation on AOS virtual channel multiplexing technique," *Journal of Astronautics*, vol. 32, no. 1, pp. 193–198, 2011.
- [13] G. Z. Shao, Z. B. Hua, J. J. Sun et al., "Design and Implementation of On-board Packard Telemetry Equipment," *Journal of Telemetry, Tracking, and Command*, vol. 22, pp. 9–14, 2001.
- [14] Q. Liu, C. Pan, G. Wang, and W. Liu, "CCSDS advanced orbiting systems, data links protocol: Study on virtual channels scheduling algorithm," in *Proceedings of the 8th International Conference on Intelligent Systems Design and Applications (ISDA '08)*, pp. 351–355, November 2008.
- [15] Y. K. Ma, Z. Z. Zhang, and N. T. Zhang, "Simulation study on data processing system based on CCSDS AOS," *Journal of Telemetry, Tracking, and Command*, vol. 23, pp. 26–30, 2002.
- [16] Z. Tian and Q. J. Zhang, "Research on advanced orbiting systems virtual channels scheduling strategy of manned spacecraft," *Spacecraft Engineering*, vol. 15, pp. 20–26, 2006.
- [17] Y. Zhao, *High Data Rate Multi-Connect Encoder*, Xidian University, Xi'an, China, 2007.