Design and Implementation of a High Efficiency Space Packet Routing Algorithm on a Spacecraft

Sheng Yu(✉), Bo Zhou, Jiaxiang Niu, Jianbing Zhu, Jian Guo, and Duo Wang

Beijing Institute of Spacecraft System Engineering, CAST, Beijing, China

Abstract. Space packet protocol is a widely used network layer protocol on a spacecraft. A modern spacecraft, such as the Tianhe core module of Chinese space station, has a very complex on-board network. Routing space packets in time is a challenge for such a spacecraft. To solve this problem, we design and implement a high efficiency space packet routing algorithm based on hash routing table. First, we introduce the typical structure of a space packet. Second, we introduce the concept of logical address and hardware address of a space packet. Third, we introduce the construction of the routing table based on the pairs of logical address and hardware address. At last, we implement a routing search algorithm based on the hash table and evaluate the performance the routing algorithm through experiments. We successfully applied the algorithm on the Tianhe core module and achieved good performance.

Keywords: Space packet protocol · Spacecraft · Routing algorithm · Hash table

1 Introduction

A spacecraft onboard network is referring to the local network that connects all devices on a spacecraft and exchanges telecommand (TC), telemetry (TM) and other data. State-of-the-art spacecraft onboard networks [1, 2] use the CCSDS (Consultative Committee for Space Data Systems) AOS (Advanced Orbiting System) protocol [3] as its data link layer protocol and SpaceWire [4], Ethernet, or MIL-STD-1553B bus as its physical layer implementation.

Space packet protocol is a widely used network layer protocol on spacecraft. Space packets are used in both ground-to-space networks and spacecraft onboard networks. A typical example using space packets is the TC application: A ground device encapsulates satellite commands in space packets, and uploads these packets to the satellite via ground-space data link. A router on the satellite receives these commands and send them to other devices via the spacecraft onboard network.

In a complex spacecraft, how to efficiently route these space packets is a challenging problem. The core module of Chinese space station, i.e., Tianhe, has a very complex on-board network, with more than 200 different devices connected together, and the on-board network uses the space packet protocol. A central unit is in charge of

Q. Guo et al. (Eds.): WiSATS 2021, LNICST 410, pp. 62–68, 2022.
https://doi.org/10.1007/978-3-030-93398-2_6

routing all space packets generated by both ground devices and on-board devices. Packet routing is a very time critical mission. In most case, the central unit needs to find a route for a space packet as soon as possible, thus, a high efficiency routing algorithm is desired.

In this paper, we proposed and implemented a high efficiency space packet routing algorithm based on hash table, and we used the algorithm on the Tianhe core module. First, we introduce the typical structure of a space packet. Second, we introduce the concept of logical address and hardware address of a space packet. Third, we introduce the construction of the routing table based on the pairs of logical address and hardware address. At last, we implement a routing search algorithm based on the hash table and evaluate the performance the routing algorithm through experiments.

2 Space Packet Protocol

The space packet protocol is defined in the CCSDS standard CCSDS 133.0-R-1 [5]. The space packet protocol is located in the network layer. A typical structure of a space packet used on the Tianhe core module is illustrated in the Fig. 1.

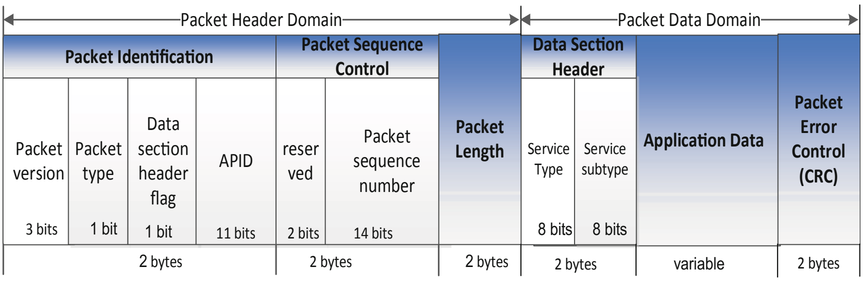

Fig. 1. A typical structure of a space packet used on the Tianhe core module

On the Tianhe core module, the meaning and possible value of each field of the space packet is listed below (Table 1).

Table 1. The space packet structure used on the Tianhe core module

Section name	Meaning	Possible value	Length
Packet version	The version of the space packet protocol	[0, 3]	3 bits
Packet type	The type of the packet	0: a telemetry packet 1: a telecommand packet	1 bit

(*continued*)

Table 1. (*continued*)

Section name	Meaning	Possible value	Length
Data section header flag	Indicates whether the packet has a data section header, which concludes service type and service subtype	0: no 1: yes	1 bit
Application Process Identifier (APID)	ID of the destination application which accepts and processes the packet	[0x000, 0x7FF]	11 bits
Packet sequence number	The sequence number of the packet	[0x0000, 0x3FFF]	14 bits
Packet length	The length of the packet (in bytes)	[0, 836]	16 bits
Service type	An application usually has many service types and this value defines the service type of the packet	[0x00, 0xFF]	8 bits
Service subtype	Each service type may have many subtypes and this value defines the service subtype of the packet	[0x00, 0xFF]	8 bits
Application data	Payload of the packet	/	Variable
Packet error control field	CRC of the packet	[0x0000, 0xFFFF]	16 bits

3 Constructing a Hash Routing Table

The first step of our routing algorithm is to create a hash routing table based on all the routing information we have. Each entry of the hash routing table is a linked list node, which contains a piece of the routing information. Each node has three elements, which are a pointer to the next node, a logical address and a hardware address. The logical address consists of the type of the packet, the APID of the packet, the service type and the service subtype of the packet, and the length of logical address is 4 bytes.

The hardware address defines the destination address of a packet. The format of the hardware address is variable depends on the type of the network used on a spacecraft. In our case, we use three types of format of the hardware address. We use many 1553B buses and an Ethernet bus on the Tianhe core module. If the destination device of a packet is a Remote Terminal (RT) of one of the 1553B buses, the hardware address of the routing entry consists of a bus id, a RT address and a RT sub address of the destination device. If the destination device of a packet is on the Ethernet bus, the hardware address we use is the IP address of the destination device. If the destination of a packet is another process on the central unit itself, the hardware address is the process id of the destination. The length of hardware address is 16 bytes.

We use E_i to represent an entry of the routing table, so $E_i = \langle pNode_i, LA_i, HA_i \rangle$. Here $pNode_i$ is the pointer to the next node of the routing table. LA_i and HA_i are a logical address and a hardware address respectively. To evenly distribute routing entries within the routing table, we select a large hash key $K_D = 429677$. The size of the hash table is represented by S_T. The index of E_i in the hash table Idx_i is calculated as follows:

$$Idx_i = (LA_i \, mod \, K_D) \, mod \, S_T \tag{1}$$

Hash collisions happen when multiple entries have a same index. We use linked list to avoid hash collisions. If E_i and E_j have a same index, then $pNode_i$ equals the start address of E_j.

4 Forwarding a Space Packet

4.1 Finding a Route

A central unit, also known as the Kernel Process Unit (KPU) on the Tianhe core module, is in charge of collecting and forwarding space packets to other devices. These devices may connect to one of the MTD-STL-1553B buses or the Ethernet bus. Whenever KPU receives a space packet, it finds a route for the packet first.

KPU parses the packet and constructs the logical address of the packet, then it uses the Eq. (1) to calculate the index corresponding to the logical address. KPU uses the index to find the entry that contains the logical address in the routing table. This process is called hash search process. If the search succeeds, KPU get the hardware address for the packet, otherwise, KPU cannot find a route for the packet and it will drop the packet. Then, KPU forwards the packet to its destination based on the type of the hardware address (Fig. 2).

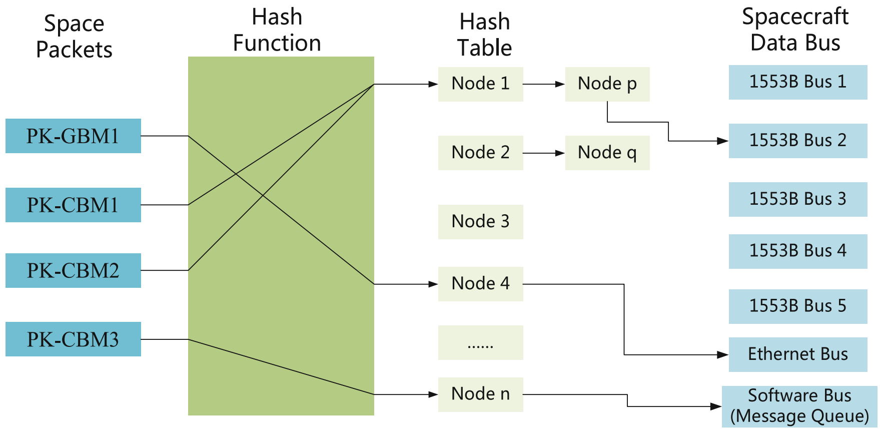

Fig. 2. The process of finding routes for space packets using hash search algorithm

4.2 Forwarding a Space Packet on 1553B Buses

If the hardware address of the packet is a 1553B address, KPU will forward the packet on one of the 1553B buses. First, KPU gets the bus_id from the hardware address and put the packet into a data buffer for the bus. Second, KPU generates a control word and a command word for the packet and put them into the buffer of a 1553B chip along with the packet. Third, the 1553B chip sends the packet to its destination based on the control and command word.

4.3 Forwarding a Space Packet on Ethernet Bus

If the hardware address of the packet is an IP address, KPU will forward the packet on the Ethernet bus using TCP/IP protocols. The TCP/IP protocol stack we use on the Tianhe core module is a version of the LWIP protocol [6]. In our case, the packet is always wrapped into an IPv4 data packet and is transmitted to the destination using UDP protocol. The packets which are send on Ethernet bus are telemetry data packets and are tolerant to packet loss.

4.4 Forwarding a Space Packet to Other Process

If the hardware address of the packet is a process id, KPU will forward the packet to another process on the KPU using message queue. In this case, the service type and service subtype in the packet are useful. The destination process receives the packet and deal with it based on the service type and service subtype of the packet. For example, the destination process can send telemetry packets to the ground via ground-space data link.

5 Performance Evaluation

The performance of the hash routing algorithm is determined by the size of the hash routing table S_T, and the choice of S_T is a tradeoff between temporal efficiency and spatial efficiency. If S_T is small, the routing table will occupy small memory space, but the chance of hash collision is high. In an extreme case, if S_T is 1, the routing table become a linked list and complexity of routing searching is O(n). If S_T is large, the chance of hash collision is low and the complexity of routing searching becomes O(1), but the routing table will occupy large memory space.

We have more than 500 different routing entries for space packets on the Tianhe core module, and we evaluated the relationship between the chance of hash collision and the value of S_T (see Fig. 3). We also evaluated the relationship between the maximum routing search time among all the packets and the value of S_T (see Fig. 4). The value of S_T we use in the experimental is increased exponentially.

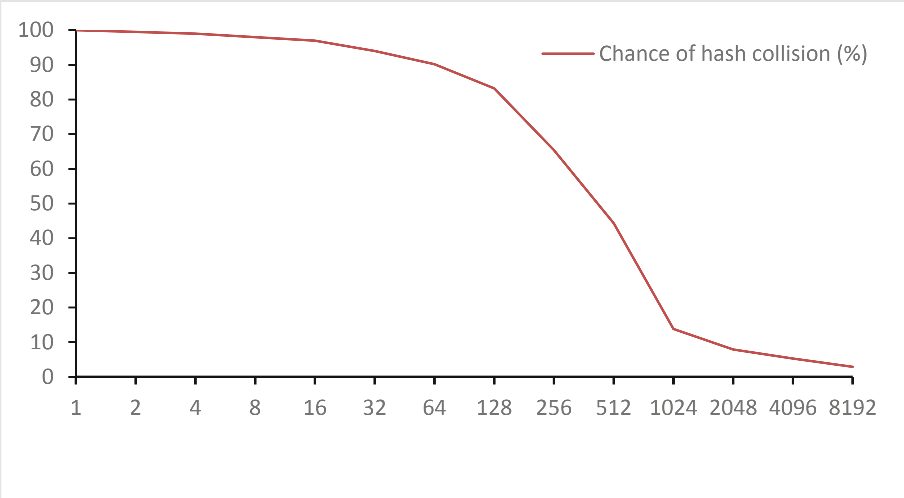

Fig. 3. Chance of hash collision versus the size of the routing table

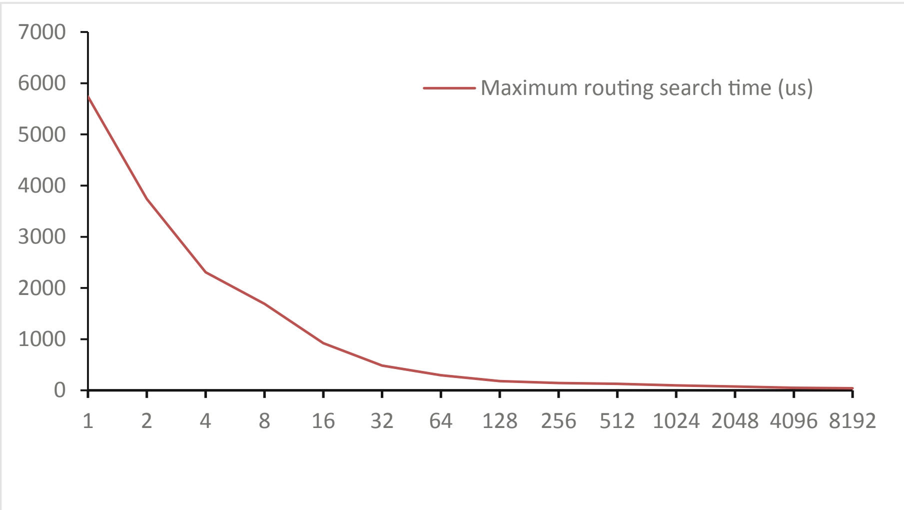

Fig. 4. Maximum routing search time versus the size of the routing table

Based on the experiment results, we choose $S_T = 1024$. The maximum routing search time is below 100 µs and the memory usage is acceptable.

6 Summary

In this paper, we design and implement a high efficiency space packet routing algorithm based on hash routing table. The algorithm is used on the Tianhe core module of the Chinese space station. It solves the difficult problem of routing different space packets in the complex ground-to-space network and spacecraft onboard network on the Tianhe core module. We introduce the method of creating the hash routing table and forwarding a space packet. Finally, we present the experimental result of how the size of the hash routing table effects the performance of the routing algorithm.

References

1. Niu, Y., Zhao, W.: Design and analysis of a strong fault-tolerant on-board spacewire bus network. J. Comput. Appl. **34**(9), 2497–2500 (2014)
2. Mao, C., Guan, Y., David, J.: Research and design of on-board dynamic SpaceWire router. J. Electron. Inf. Technol. **32**(8), 1904–1909 (2010)
3. CCSDS 732.0-B-2: AOS Space Data Link Protocol. CCSDS blue book, July 2006
4. ECSS-E-ST-50-12C: SpaceWare - Links, Nodes, Routers and Networks. ESA-ESTEC (2012)
5. CCSDS 133.0-B-1: Space Packet Protocol. CCSDS blue book, September 2012
6. Yu, S., Qiu, Q., Zhou, B., et al.: Design and implementation of spacecraft onboard network based on LWIP. In: 2018 Eighth International Conference on Instrumentation & Measurement, Computer, Communication and Control (IMCCC) (2018)