

## SpaceWire 网络热点通信模式的缓存资源分配算法\*

龚泉铭<sup>1,2,3</sup>, 闫梦婷<sup>1,2</sup>, 姜秀杰<sup>1,2,3</sup>, 安军社<sup>1,2,3</sup>

(1. 中国科学院国家空间科学中心, 北京 100190;

2. 中国科学院复杂航天系统电子信息技术重点实验室, 北京 100190; 3. 中国科学院大学, 北京 100190)

**摘要:**基于 SpaceWire 网络在热点通信模式下的工作特点,研究缓存资源的分配算法。给出均匀通信模式和热点通信模式的释义;推导网络路由节点的满负荷概率和平均延时的解析方法,计算网络中的关键通信节点;再给出 SpaceWire 网络缓存资源分配算法。利用 Opnet 网络仿真平台建立 SpaceWire 通信模型,仿真了不同通信模式中采用缓存优化策略前后的关键性能指标。研究结果表明:在热点通信模式状态下,该缓存资源分配算法在保证总资源一定的前提下,网络系统的平均延时降低,优化了 SpaceWire 网络性能。

**关键词:**SpaceWire;分析建模;缓存优化;分配算法;性能优化

中图分类号:TN95 文献标志码:A 文章编号:1001-2486(2018)02-041-07

## Buffer allocation algorithm for SpaceWire network in hotspot communication mode

GONG Quanming<sup>1,2,3</sup>, YAN Mengting<sup>1,2</sup>, JIANG Xiujie<sup>1,2,3</sup>, AN Junshe<sup>1,2,3</sup>

(1. National Space Science Center, Chinese Academy of Sciences, Beijing 100190, China;

2. Key Laboratory of Electronics and Information Technology for Space Systems, Chinese Academy of Sciences, Beijing 100190, China;

3. University of Chinese Academy of Sciences, Beijing 100190, China)

**Abstract:** According to the working characteristics of SpaceWire network in hotspot communication mode, the buffer allocation algorithm was studied. The definition of uniform communication mode and hotspot communication mode were given; the calculation method of full-load probability and the average delay of nodes in the network were deduced, and the key communication nodes in the network were calculated; the buffer allocation algorithm for SpaceWire network was derived. Then, the SpaceWire communication model was established by using the Opnet network simulation platform, and the key performance indexes of the different communication modes were simulated before and after adopting the buffer optimization strategy. The study results show that, in the hotspot communication mode, the buffer allocation algorithm reduces the average delay of the network system under the same total resource. The performance of the SpaceWire network is optimized.

**Key words:** SpaceWire; analytical model; buffer optimizing; allocation algorithm; performance optimizing

SpaceWire 是由欧洲太空局(European Space Agency, ESA)首先提出,并由多个高校、科研机构等参与设计和优化的新型数据总线标准<sup>[1]</sup>,其能够为星载设备提供标准一致的通信协议,简化星上系统的设计集成,将星上的测控数据、载荷数据以及星内闭环控制等信息进行高速率、低误码率的传输。由于以上优点,SpaceWire 协议已经成功应用于多个国内外空间任务<sup>[2-3]</sup>,如:在风云四号气象卫星上,SpaceWire 总线完成了载荷成像设备大数据量、高速的数据传输任务<sup>[4]</sup>;Swift 卫星中,SpaceWire 作为高速数据传输协议,将  $\gamma$  射线探测器采集的科学数据及相关参数传输至冗余的

数据处理模块<sup>[5]</sup>。

目前对 SpaceWire 总线具有前瞻性的研究成果尚少。文献[6]以标准协议为基础,研究了 SpaceWire 网络在多数数据源、异步、多任务触发的复杂条件下保证传输时延稳定可控的传输机制;文献[7]简要介绍了 SpaceWire 网络构建方式,举例分析链型组网、环形组网、树形组网等几种传统拓扑结构,但缺乏在复杂网络条件下 SpaceWire 的路由设计方式和研究方法;文献[8]、文献[9]、文献[10]提出了一些面向具体功能模块的相关验证方法和技巧;文献[11]介绍了一种 SpaceWire 总线流量控制机制,其在一定程度上改

\* 收稿日期:2016-10-19

基金项目:中国科学院战略性先导科技专项基金资助项目(XDA04060300)

作者简介:龚泉铭(1990—),男,湖北荆州人,博士研究生,E-mail:gongquanming12@mails.ucas.ac.cn;

姜秀杰(通信作者),女,研究员,博士,博士生导师,E-mail:jiangxj@nssc.ac.cn

善了总线的性能,但未充分考虑实际应用过程中有限网络资源下的缓存资源分配问题。然而实际应用中,有限资源条件下总线网络系统性能的优劣程度是星载总线系统主要的技术难题之一,因此如何对 SpaceWire 网络通信性能进行分析以及对网络中缓存资源进行合理分配,是该技术实现过程中需重点考虑的问题。

本文依据 SpaceWire 网络的特点,创建符合实际应用的网络模型,认真分析网络数据传输的问题点,在此基础上,推导出一套缓存资源分配的算法。该算法能够在已知网络总的缓存资源量的条件下,结合分析模型中得出的通信约束条件,给出适合 SpaceWire 网络实现的缓存分配方案,实现改善网络性能的目的。

### 1 SpaceWire 网络性能分析模型

#### 1.1 SpaceWire 协议概述

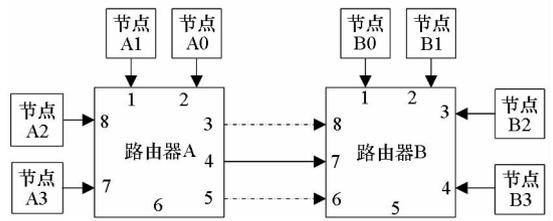
SpaceWire 网络的基本结构单元为节点、路由以及链路。节点和路由、路由和路由之间均以链路串接,构成整个 SpaceWire 网络系统。SpaceWire 协议推荐一种自适应网络机制,其关键技术点在于在单一链路的传输效率有限的条件下,可以采用调整总线中路由通路数量的方式实现提升整个网络的数据传输效率的目的。因此将路由作为基本单元,搭建具有很高自由度的复杂网络,可以形成结构各异的 SpaceWire 路由网络拓扑方式。

图 1(a)以两个基本路由单元构建了一种简单的网络结构,即双路由结构<sup>[12-14]</sup>。在数据源包的整个传输路径上(如节点 A1 到节点 B2),在路径(3→8)被占用的情况下,路由 A 可以自适应地选择另一条通路(4→7)进行数据包传输。只要三条等效链路中某一条存在空闲,数据源包就可以被即时传输。该双路由结构实现了三组数据源包的并行传输,极大提高了传输效率。

细化的基本网络路由单元如图 1(b)所示,整个网络单元包含 8 个接口节点,各节点既可以作为数据的直接输入输出端口,也可以连接其他相似的网络单元模块。路由单元的核心功能包括 Codec 节点数据处理、数据传输路径仲裁选择、数据交换开关矩阵等,并且数据源包的传输是虫孔路由。其中,Codec 节点数据处理部分完成对数据源包的缓存、包头地址信息的解析;仲裁选择模块根据地址信息进行路径选择和数据冲突时进行传输顺序的处理。

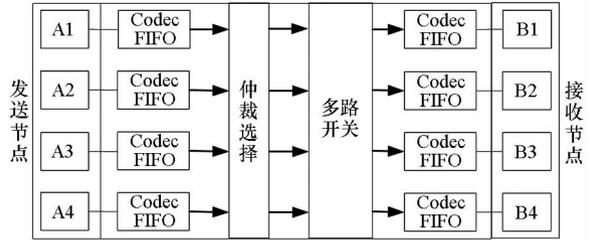
#### 1.2 SpaceWire 网络特点分析

不同数量的路由核和节点可以实现多种网络



(a) 双路由结构

(a) Structure of dual-route



(b) SpaceWire 基本网络路由单元

(b) Basic route unit of SpaceWire

图 1 SpaceWire 拓扑结构

Fig. 1 Topology of SpaceWire

拓扑结构,从而保证 SpaceWire 网络能够搭建适应不同需求类型的网络形式。然而,所有复杂网络在细化分解后,均可以视为若干基本双路由结构,而双路由结构又是由两个多节点星形连接的单路由网络构成。因此,这类基本网络单元的两个重要指标——传输性能和缓存资源量将是影响整个 SpaceWire 网络的性能和资源分配的基本因素。给出定义:

**定义 1** 均匀通信网络,是指各个输入路由节点的数据输入速率相同且数据到达路由节点的概率也相同的网络,即网络的通信量是均匀的。

**定义 2** 热点通信网络,是指网络中存在某个路由节点的数据输入速率或数据到达概率高于其他路由节点的网络,其网络的通信存在某一个热点。

SpaceWire 网络作为一种空间应用的网络,各路由节点数据传输特征值(如传输路径、传输速率、传输概率等)是不同的。缓存资源平均分配并不能保证有限的网络缓存资源得到充分利用。一旦某一路节点出现异常,如缓存满载,将导致同样使用该节点发送的数据源包阻塞,无法传输;同时由于 SpaceWire 网络通常采用虫孔路由机制,会导致其他节点也阻塞。由此可得:缓存满负荷概率的大小是缓存容量是否满足使用需求的重要标志。而网络中各个端口的满负荷概率主要受两个因素的制约,一是数据输出仲裁阻塞概率,二是下级路由节点的满负荷导致的反馈阻塞。实际应用中,由于缓存资源紧张导致反馈阻塞的概率远远小于数据输出时的仲裁阻塞概率。根据上述讨

论得出:

**定义3** 在一个网络中,如若存在某一个最容易阻塞的路由节点,即满负荷的可能性最大,则该路由节点可视为数据在网络中通信的瓶颈,即为关键通信节点。

对关键通信节点的处理,为后文热点通信网络的缓存分配提供了依据。

## 2 SpaceWire 网络缓存资源分配算法

### 2.1 网络条件约束及参数定义

以 SpaceWire 网络及上述分析为基础,搭建合理的 SpaceWire 网络模型,来进一步讨论相关性能和资源分配。网络模型的相应规则约束如下。

**约束1:**SpaceWire 网络中的路由方式为静态路由。

**约束2:**只考虑传输路径上的缓存空间,数据传输路径起点和终点的缓存空间无限大。

**约束3:**整个数据传输过程中带宽不变,排除带宽变化引入干扰。

**约束4:**路由器互联是点对点单链路互联,不包含多对一、一对多等情况。

在上述规则约束的条件下,进一步给出如下定义:

**定义4** 给定图  $G(R, DIR, L)$ ,其中  $R$  为网络中路由节点的集合,  $R_{x,y}$  为网络中坐标为  $(x, y)$  点的路由节点,  $DIR$  为网络中数据包传输方向的集合 ( $dir$  为其中元素),  $L$  为数据源包大小的集合。

**定义5** 对于网络链路上节点的特征有如下定义:  $C$  为网络中路由节点缓存容量的集合,  $D$  为网络中链路节点延时的集合,  $V$  为链路速率的集合。

根据上述约束和定义内容,为方便分析介绍和过程运算,参数说明如表1所示。

表1 参数表  
Tab.1 Parameter list

参数名称	参数定义
$C_{x,y,dir}$	$R_{x,y}$ 在 $dir$ 方向的缓存容量
$\lambda_{x,y,dir}$	$C_{x,y,dir}$ 的数据输入概率
$\mu_{x,y,dir}$	$C_{x,y,dir}$ 的传输效率
$\rho_{x,y,dir}$	$C_{x,y,dir}$ 缓存容量的利用率
$P_{x,y,x',y',dir'}$	$C_{x,y,dir}$ 的数据通过 $C_{x',y',dir'}$ 的概率
$P_{x,y,dir,dir'}$	$C_{x,y,dir}$ 的数据向 $dir'$ 方向转发的概率
$\mu_{x,y,dir,dir'}$	$C_{x,y,dir}$ 中的数据向 $dir'$ 方向的平均传输效率
$P_{x,y,full,dir}$	$C_{x,y,dir}$ 缓存满的概率
$D_{x,y,dir}$	$C_{x,y,dir}$ 中数据包的传输时间
$C_{total}$	网络总的缓存大小

表1中,  $D_{x,y,dir}$  由两部分组成:数据包准备时间  $D_L$ ,即数据达到路由节点到准备输出到下一级为止的时间段;数据源包从开始向下一级传输到完整传输到下一级的时间段  $D_{x,y,dir,dir'}$ 。

### 2.2 关键通信节点的确定方法

为进一步分析节点缓存的满负荷概率,需计算数据包通过节点的总传输时长。其由两个时间段决定,即数据达到路由节点到准备输出到下一级为止的时间和数据源包从开始向下一级传输到完整传输到下一级的时间。其表达式为:

$$D_{x,y,dir} = D_L + D_{x,y,dir,dir'} \quad (1)$$

上式中,  $D_L$  可分解为多个小时间单元,即 SpaceWire 链路转换时间  $D_{SpW}$ 、地址解析及路径选择时间  $D_{Head}$ 、路由仲裁时间  $D_{Arb}$ 、数据源包穿过开关矩阵时间  $D_{Cross}$ ,从而有:

$$D_L = D_{SpW} + D_{Head} + D_{Arb} + D_{Cross} \quad (2)$$

在网络搭建完成后上述时间单元便已给定。需要说明的是,在以往的网络分析中,特别是计算机网络<sup>[15]</sup>,一种观点认为无须考虑数据在路由包头解析前的相关操作,从而忽略了这些操作带来的时间延迟。但是,SpaceWire 网络的特点在于,SpaceWire 链路为高速串行链路,数据到达路由端后需要进行必要的串并转换。因此,SpaceWire 网络的建模过程需要考虑  $D_{SpW}$  对模型准确度的影响,并且其与链路速率有很强的相关性,对网络性能有十分重要的影响。

为方便分析传输时间的相关特性<sup>[16]</sup>,将该节点传输路径相同数据包传输过程抽象为一个并联的虚拟通道。数据源包长度相同的数据将被划入同一个虚拟通道中,构成并联的数据包传输队列,如图2所示。

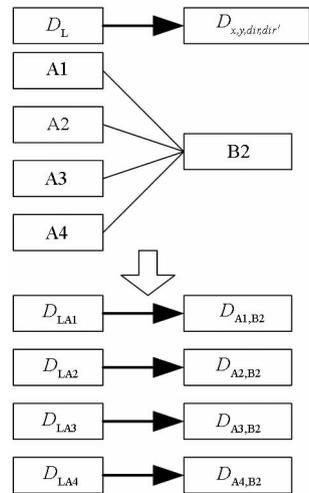


图2 路由接收数据排队示意图

Fig.2 Data receive queue for SpaceWire router

其中,每一个  $D_{L_{Ai}}$  和  $D_{A_i, B_2}$  ( $i = 1, 2, 3, 4$ ) 对应于该接受节点在式(1)中的两个时间段的值。每个虚拟队列传输数据源包的概率取决于该数据包在对应路径上传输的概率,即  $P_{x,y,dir,dir'}$ 。总的传输时间则可以由每个虚拟队列的传输时间求和得出,即

$$D_{x,y,dir,B_2} = \frac{1}{\sum \mu_{x,y,dir,B_2} \times P_{x,y,dir,B_2}} \quad (3)$$

$P_{x,y,dir,dir'}$  在路由方式已经确定的情况下为常量,因此数据包向某条路径传输的效率  $\mu_{x,y,dir,dir'}$  成为影响  $D_{x,y,dir,dir'}$  的唯一条件。进一步讨论分析过程如下:

首先将图 2 中各个虚拟队列看作一个整体,这个整体的数据输入概率为:

$$\lambda_{B_2} = \lambda_{x,y,A_1} \times P_{x,y,A_1,B_2} + \lambda_{x,y,A_2} \times P_{x,y,A_2,B_2} + \lambda_{x,y,A_3} \times P_{x,y,A_3,B_2} + \lambda_{x,y,A_4} \times P_{x,y,A_4,B_2} \quad (4)$$

将图 2 所需的传输时间表示为:

$$D_L = \lambda_{x,y,A_1} \times \frac{L_{A_1}}{V_{A_1}} + \lambda_{x,y,A_2} \times \frac{L_{A_2}}{V_{A_2}} + \lambda_{x,y,A_3} \times \frac{L_{A_3}}{V_{A_3}} + \lambda_{x,y,A_4} \times \frac{L_{A_4}}{V_{A_4}} \quad (5)$$

从而得到总的传输时间为:

$$D = (1 - P_{x,y,full,dir'}) \times D_L + P_{x,y,full,dir'} \times (D_{x,y,dir} + D_L) \quad (6)$$

所以,其传输效率为:

$$\mu = \frac{1}{(1 - P_{x,y,full,dir'}) \times D_L + P_{x,y,full,dir'} \times (D_{x,y,dir} + D_L)} \quad (7)$$

根据 M/M/1 排队模型公式,数据在图 2 部分的平均等待时间:

$$t_{B_2} = \frac{1}{\mu - \lambda_{B_2}} \quad (8)$$

对于图 2 中某一虚拟队列应用 M/M/1 排队模型可知(以 A1 为例):

$$t'_{B_2} = \frac{1}{\mu_{x,y,A_1,B_2} - \lambda_{x,y,A_1} \times P_{x,y,A_1,B_2}} \quad (9)$$

假定路由器的路由机制为均等路由,保证不同发送节点通向同一接收节点的等待时间相同,从而有  $t_{B_2} = t'_{B_2}$ 。联立式(4)~(9)可得出在 B2 路径上的传输效率  $\mu_{x,y,A_1,B_2}$ 。同理,可以得出在其他几个路径上的传输效率  $\mu_{x,y,A_2,B_2}$ 、 $\mu_{x,y,A_3,B_2}$ 、 $\mu_{x,y,A_4,B_2}$ 。再由式(3)可推导出传输时间  $D_{x,y,dir,B_2}$ ,将其代入式(1)中即可计算得出单个路由发送节点总的传输时间。

对于每个独立的路由发送节点满足 M/M/1/K

模型,其满概率计算公式为:

$$P_{x,y,full,dir} = \frac{1 - (\rho_{x,y,dir})^{C_{x,y,dir}}}{(1 - \rho_{x,y,dir})^{C_{x,y,dir} + 1}} \quad (10)$$

式中,  $\rho_{x,y,dir} = \frac{\lambda_{x,y,dir}}{\mu_{x,y,dir}}$ ,  $\mu_{x,y,dir} = \frac{1}{D_{x,y,dir}}$ 。

根据式(10)得出的单个路由发送节点满负荷概率,每个路由节点的阻塞概率都可以由其对应的总的传输时间  $D_{x,y,dir}$  和缓存容量确定出来。从而,通过计算各个路由发送节点的阻塞概率,找出概率最高的节点,并以此为基础对 SpaceWire 网络性能进行优化。

### 2.3 SpaceWire 网络缓存分配算法描述

根据上节的分析描述,计算网络中阻塞概率最高的路由节点,并将其视作突破口,利用如图 3 所示算法进行优化处理。

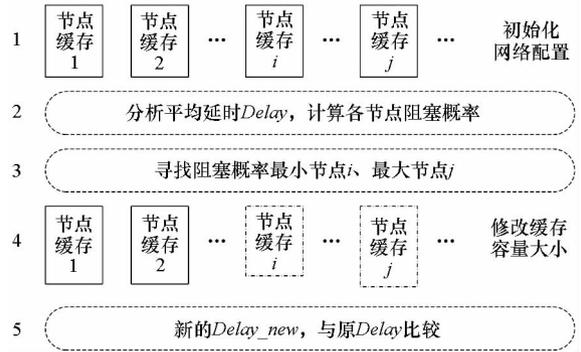


图 3 缓存分配算法描述示意图

Fig. 3 Graph of buffer allocation algorithm

具体算法过程如下:

**步骤 1:**初始化配置网络系统参数,并将所有缓存资源均匀分配到每个路由节点的缓存中,通过仿真得出当前参数下,数据传输在网络中的平均延时 Delay。

**步骤 2:**根据输入的配置参数对网络系统进行分析,得出每个路由节点缓存的阻塞概率表达式。

**步骤 3:**寻找阻塞概率最大的路由节点缓存  $C_{max}$  以及阻塞概率最小的路由节点缓存  $C_{min}$ 。

**步骤 4:**修改系统配置参数,修改  $C_{max}$  的容量  $Volume_{Max}$  使其加 1,同时将  $C_{min}$  的容量  $Volume_{Min}$  减 1。将修改后的参数重新输入仿真模型中计算新的网络平均延时  $Delay_{new}$ 。

**步骤 5:**如果  $Delay_{new}$  小于  $Delay$ ,则认为新的参数设置更有效,保留配置结果并跳回步骤 2。如果  $Delay_{new}$  大于等于  $Delay$ ,则保留原配置参数,并将  $Volume_{Max}$  和  $Volume_{Min}$  改回原来

的值。

### 3 仿真结果与分析

为了更深入探讨在上述模型下的分析结果和算法的可行性,利用 Opnet 软件搭建网络模型并仿真,利用仿真结果验证算法对网络性能的影响<sup>[17]</sup>。

#### 3.1 仿真平台的搭建

由于 Opnet 平台没有适用于 SpaceWire 标准的仿真模块,无法直接进行 SpaceWire 网络仿真。根据图 1(b) 给出的 SpaceWire 拓扑结构,在 Opnet 平台环境下自行搭建的网络模型环境如图 4 所示。SpaceWire 为全双工网络,即每个节点的接收与发送的传输及缓存相互独立,因此在模型中只描述了左端发送右端接收的方式,整个 SpaceWire 的全双工网络情况的缓存资源量为其两倍。

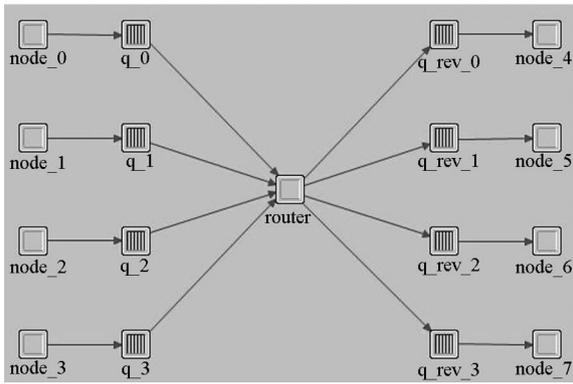


图 4 Opnet 网络仿真模型

Fig. 4 Network simulation model of Opnet

在图 4 所示的网络仿真模型中, node\_0 ~ node\_3 为数据源生成单元,对应于图 1(b) 中的发送节点,用以调节数据源包输出形式以及源包的插入率; q\_0 ~ q\_3 为源包缓存单元,用于模拟网络节点的缓存; q\_rev\_0 ~ q\_rev\_3 为源包接收单元;中间 router 为路由控制单元,以实现要求的路由传输方式,即图 1(b) 中仲裁选择和开关矩阵功能。模型中各单元的仿真参数说明如下:

- 1) 总缓存空间大小  $C_{total} = 4096$ 。
- 2) 数据源包发送单元 (node\_0 ~ node\_3) 以泊松分布将数据注入网络模型中,源包容量为 900,插入率可调,初始值为 0.5。
- 3) 源包缓存单元 (q\_0 ~ q\_3) 容量初始值  $C_{q_i} = 1024$ ,约束条件如下:

$$\begin{cases} 1 \leq C_{q_1}, C_{q_2}, C_{q_3}, C_{q_4} < 4096 \\ C_{q_1} + C_{q_2} + C_{q_3} + C_{q_4} = 4096 \end{cases}$$

通过对该网络模型中各节点的缓存容量、源包插入率、源包容量等参数的调整完成仿真。每次仿真注入相应目标参数提取 1000 个数据采样点,得出不同参数组合下节点的平均延时变化情况,并由 MATLAB 对结果加以解析。

#### 3.2 关于总缓存资源均匀分配下的分析

总缓存资源均匀分配,即均匀流量模式,保证各类配置参数一致。该仿真模型中有 4 个发送路由单元,分别设置相同的缓存空间 1024,以及相同的数据插入率和数据源包容量。不难发现,在该模式下网络中各个路由单元的流量一致,对缓存空间的需求也一致。如果调整任意单元的缓存容量必然会影响到整个网络的性能。改变后缓存空间最小的路由节点便成为该网络中数据通信的关键节点,从而降低整个网络的性能。因此不做深入分析。

当各路由节点的缓存空间一致,但数据流量不同,即各个节点的数据插入率不一致时,必然存在某一个路由节点的数据插入率要高于其他节点,该模式即为热点通信模式,该节点即为热点流量节点。其他路由节点即为非热点流量节点。对该模式的仿真分析如图 5 所示,热点流量节点的平均延时要明显高于非热点流量节点,说明网络性能需要优化。

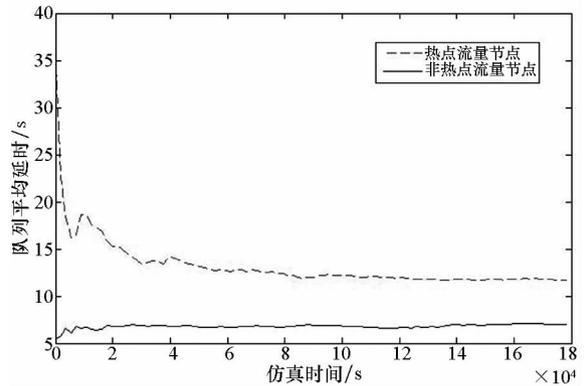


图 5 缓存均匀分配下平均延时曲线

Fig. 5 Curves for average delay under the uniform cache

#### 3.3 关于缓存资源经算法优化后的分析

结合第 2.3 节提出的算法,通过调整各节点的缓存空间的方式来优化网络性能。具体步骤如下:①降低非热点流量节点的缓存容量,单次递减量为 1;②增加热点流量节点的缓存容量,增加量与①中减少量相同;③调整完毕后,完成一次仿真分析过程;④记录 20 次缓存容量调整后的热点和非热点流量的仿真结果,共 40 组数据。

在此指出,由图 5 可知平均延时是条变化曲

线,曲线趋于稳定的值可以作为该曲线的典型值。本次结果比对分析中,提取每次仿真 1000 个采样点的后 500 个采样点作为该次仿真结果的典型值。对全部仿真结果统计分析后,得出如图 6 所示平均延时随缓存空间变化量的变化曲线图。

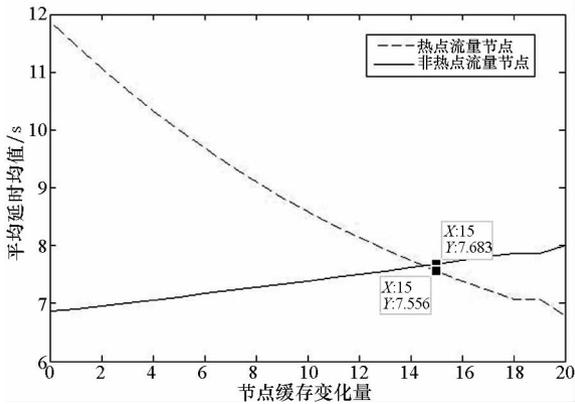


图 6 节点缓存空间对平均延时的影响

Fig.6 Curves for average delay under the changing cache

由前文可知,平均延时最大的路由节点单元影响着整个网络的性能。在图 6 中,当节点缓存变化量为 15 时,热点流量节点和非热点流量节点的平均延时分别为 7.556 s 和 7.683 s。整个网络的平均延时也到达最低点 7.683 s。此时,网络中各节点平均延时曲线如图 7 所示。

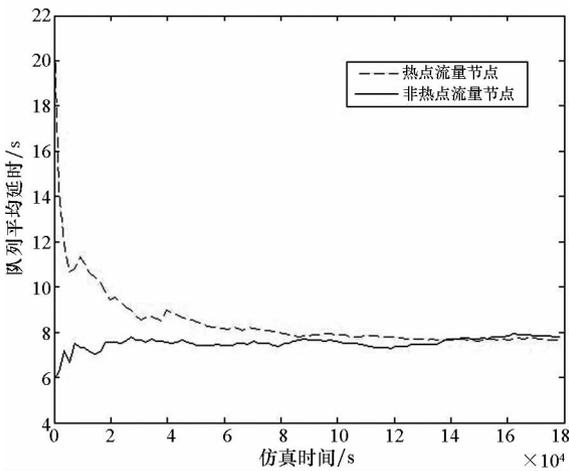


图 7 网络性能最优时节点平均延时曲线

Fig.7 Curves for average delay under the best network performance

由上述分析可知,网络模型经算法调整后性能得到优化。热点流量节点的平均延时典型值由 11.880 s 下调到 7.556 s,下降率为 36.4%,下降幅度明显。而非热点流量节点的平均延时典型值只提升了 12.0% (6.862 s→7.683 s),影响不大。同时,网络性能优劣取决于阻塞概率最大的路由节点,即平均延时最大的,可以认定经算法调整后

的网络性能有了明显的提升。

## 4 结论

对 SpaceWire 网络的均匀通信模式和热点通信模式的工作特点进行了分析和介绍,重点讨论了热点通信模式在不同缓存结构下对数据通信的性能影响。根据 SpaceWire 网络通信特性搭建数学模型,分析总结出一套关键通信节点的计算方法和缓存分配方法。在缓存资源总量一定的情况下,通过测算 SpaceWire 网络系统中的关键通信节点调节网络中路由单元的缓存空间的分配方式,以期改善网络性能。利用 Opnet 软件搭建一套 SpaceWire 网络系统测试平台,分别针对均匀通信模式和热点通信模式两种情况进行仿真。在  $C_{total} = 4096$  的测试环境下,分析了缓存资源经算法优化前后网络性能变化情况。结果表明:在不改变网络系统总缓存容量的条件下,该算法能有效改善 SpaceWire 网络性能,可以为今后 SpaceWire 的设计实现中缓存资源的分配提供理论依据。

## 参考文献 (References)

- [1] Parkes S M. SpaceWire-links, nodes, routers and networks: ECSS - E - ST - 50 - 12C [ R ]. European Cooperation for Space Standardization ( ECSS ), 2008.
- [2] Parkes S, Sheynin Y, Nomachi M. SpaceWire missions and architectures [ C ] // Proceedings of the 60th International Astronautical Congress, 2009.
- [3] Roberts D, Parkes S. SpaceWire missions and applications [ C ] // Proceedings of International SpaceWire Conference, 2010.
- [4] 董瑶海. 风云四号气象卫星及其应用展望 [ J ]. 上海航天, 2016, 33 ( 2 ): 1 - 8.  
DONG Yaohai. FY - 4 meteorological satellite and its application prospect [ J ]. Aerospace Shanghai, 2016, 33 ( 2 ): 1 - 8. ( in Chinese )
- [5] NASA. The Neil Gehrels Swift Observatory [ EB/OL ]. ( 2016 - 09 - 28 ) [ 2016 - 10 - 18 ]. <http://swift.gsfc.nasa.gov/docs/swift/swiftsc.html>.
- [6] 杨志, 李国军, 杨芳, 等. SpaceWire 星载网络通信协议设计 [ J ]. 宇航学报, 2012, 33 ( 2 ): 200 - 209.  
YANG Zhi, LI Guojun, YANG Fang, et al. Design of communication protocol for SpaceWire on-board networks [ J ]. Journal of Astronautics, 2012, 33 ( 2 ): 200 - 209. ( in Chinese )
- [7] 凤雷, 鲁文帅, 付平. SpaceWire 网络层协议分析与应用研究 [ J ]. 测试技术学报, 2010, 24 ( 6 ): 522 - 526.  
FENG Lei, LU Wenshuai, FU Ping. Analysis of SpaceWire network layer protocol and application research [ J ]. Journal of Test and Measurement Technology, 2010, 24 ( 6 ): 522 - 526. ( in Chinese )
- [8] 赵俊艺, 何兵哲. 基于 SpaceWire 总线的星载数据系统 [ J ]. 飞行器测控学报, 2009, 28 ( 2 ): 13 - 17.  
ZHAO Junyi, HE Bingzhe. Onboard data system based on

- SpaceWire bus [J]. Journal of Spacecraft TT&C Technology, 2009, 28(2): 13-17. (in Chinese)
- [9] 董玲玲, 关永, 李晓娟, 等. 用 LTL 模型检验的方法验证 SpaceWire 检错机制 [J]. 计算机工程与应用, 2012, 48(22): 88-94.  
DONG Lingling, GUAN Yong, LI Xiaojuan, et al. Verification for SpaceWire error detection mechanism by LTL model checking [J]. Computer Engineering and Applications, 2012, 48(22): 88-94. (in Chinese)
- [10] 李黎明, 关永, 吴敏华, 等. 运用定理证明的形式化方法验证 SpaceWire 编码电路 [J]. 小型微型计算机系统, 2012, 33(6): 1372-1376.  
LI Liming, GUAN Yong, WU Minhua, et al. Formal method for verifying SpaceWire encoding circuit by applying theorem proving [J]. Journal of Chinese Computer Systems, 2012, 33(6): 1372-1376. (in Chinese)
- [11] 赵云富, 吴一帆, 孙强, 等. SpaceWire 总线的流量控制机制研究 [J]. 微电子学与计算机, 2016, 33(1): 1-5.  
ZHAO Yunfu, WU Yifan, SUN Qiang, et al. Research on the flow control mechanism of SpaceWire [J]. Microelectronics & Computer, 2016, 33(1): 1-5. (in Chinese)
- [12] 闫梦婷. SpaceWire 总线 RMAP 协议的实现与网络性能研究 [D]. 北京: 中国科学院大学, 2016.  
YAN Mengting. SpaceWire bus and RMAP protocol network performance research and implement [D]. Beijing: University of Chinese Academy of Sciences, 2016. (in Chinese)
- [13] 闫梦婷, 安军社, 龚泉铭. 基于双路由单元的 SpaceWire 总线性能评价方法研究 [J]. 国防科技大学学报, 2017, 39(1): 86-91.  
YAN Mengting, AN Junshe, GONG Quanming. Algorithm of performance evaluation for SpaceWire group adaptive routing [J]. Journal of National University of Defense Technology, 2017, 39(1): 86-91. (in Chinese)
- [14] Shakhmatov A V, Chekmarev S A, Vergasov M Y, et al. SpaceWire router IP-core with priority adaptive routing [C]// Proceedings of IOP Conference Series: Materials Science and Engineering, 2015, 94(1): 012002.
- [15] Sanchez L, Pati G, Murray V, et al. Reduced power consumption in the FPGA-based universal link for LVDS communications [C]// Proceedings of 7th Latin American Symposium on Circuits & Systems (LASCAS), 2016: 283-286.
- [16] Ryu S M. Analysis of transmission delay of timecode over SpaceWire network using OMNeT++ [J]. Journal of the Korea Institute of Information and Communication Engineering, 2015, 19(9): 2022-2028.
- [17] 代真, 何锋, 熊华钢. 混合机制下的 SpaceWire 传输延时仿真分析 [J]. 计算机工程与设计, 2015, 36(1): 1-5.  
DAI Zhen, HE Feng, XIONG Huagang. Transmission delay simulation and analysis in SpaceWire under combined mechanism [J]. Computer Engineering and Design, 2015, 36(1): 1-5. (in Chinese)