doi:10.3969/j.issn.1671-637X.2013.07.015

AFDX实时流量的时间确定性中间件接入模型研究

易 娟, 何 锋, 王 彤
(北京航空航天大学电子信息工程学院,北京 100191)

摘 要: 为保证COTS平台下的航空电子应用在航空电子全双工以太网(AFDX)上的时间确定性接入,设计了AFDX网络实时流量的中间件接入模型。通过基于中间层驱动过滤机制的协议转换,实现了双边网络的协议对接,并给出了帧顺序号(SN)的插入方法;通过对虚拟链路(VL)在时间上进行有效分配,避免了速率受限的数据流对物理链路的争用,在实现单条VL的BAG整形基础上提出了使实时流量抖动最小的VL复合时间调度算法。搭建典型实验环境对该模型进行仿真实验,结果表明,中间件模型能准确地实现数据帧TCP/IP协议与AFDX协议格式的相互转换,且该时间调度算法大大减小了VL复合所产生的抖动,保证了数据流传输的时间确定性。

关键词: 航空电子全双工以太网; 中间件; 时间确定性; 协议转换; BAG保障
中图分类号: V271.4; TP393 **文献标志码**: A **文章编号**: 1671-637X(2013)07-0062-05

Time Deterministic Middleware Model for AFDX Real-Time Flow

YI Juan, HE Feng, WANG Tong
(School of Electronics and Information Engineering, Beihang University, Beijing 100191, China)

Abstract: To ensure the time determinability of avionics applications based on Commercial-off-the-shelf (COTS) platform when accessing into AFDX (Avionics Full Duplex switched ethernet) network, an AFDX real-time flow middleware was designed. Using protocol conversion function based on filtering mechanism of intermediate driver, the middleware achieved communications between networks running with different protocol. Core code of frame sequence number (SN) insertion was described. A time scheduling algorithm which has the minimum jitter aimed at VL-Multiplexing was proposed, by distributing the sending-time of visual links(VL), it is able to avoid contention on a rate limited physical link. Finally, experiments were made to test the performance of AFDX real-time flow middleware and the results showed that: the middleware model achieved the format mutual conversion between TCP/ IP protocol and AFDX protocol, and the time scheduling algorithm can decrease jitter caused by VL multiplex apparently, which ensured the time determinability of AFDX communication.

Key words: AFDX; middleware; time determinability; protocol conversion; BAG guarantee

0 引言

相比普通计算机网络,航空电子机载总线网络在实时性和可靠性方面具有更高的要求。实时性要求航空电子总线网络提供端到端延时控制机制,满足消息的时间确定性传输;可靠性要求航空电子总线网络提供故障隔离、冗余检错等手段确保消息的有效传输。引入了虚拟链路技术的航空电子全双工交换式以太网(Avionics Full Duplex switched ethernet,AFDX),其具备流量整形、流量管制、优先级调度等确定性控制机制[1],这极大地增强了网络的实时性和可靠性,成为大中型飞机综合化互连的事实标准[2]。

在研究AFDX网络的协议行为和具体实施方案中,需要在已有COTS(Commercial-off-the-shelf)平台上,对COTS技术进行改造,使基于TCP/IP协议的航电应用能够透明地在AFDX网络上运行,完成实时流量在AFDX网络的准确接入。在这个过程中,模拟端系统基于虚拟链路通信的通信协议中间件将成为COTS平台与AFDX网络对接的关键部件。法国图卢兹大学航空航天实验室提出的Toolkit模型作为一种COTS平台与AFDX网络的对接部件,仅实现了不同协议网络的相互通信,并未保障数据流的时间确定性。

中间件处于操作系统与用户应用软件之间[3]。它在操作系统、网络之上,用户应用软件之下,为处于自

收稿日期:2012-06-23 **修回日期**:2012-08-14
基金项目:国家自然科学基金(60879024)
作者简介:易 娟(1989—),女,湖南岳阳人,硕士生,研究方向为航空电子总线与通信。

已上层的应用提供运行与开发环境，帮助系统开发者灵活、高效地开发和集成应用软件[4]。

本文通过分析与研究网络驱动接口规范中间层驱动程序框架结构，利用其在通信模型中的位置[4]，设计了适用于 AFDX 网络的实时流量中间件接入模型，并提出了相应的时间确定性保障算法。将该中间件安装在普通以太网端，普通以太网端上的航空电子应用即能在保证时间确定性的条件下与 AFDX 网络正常通信。这为航空电子应用通信过程无差别地在 AFDX 网络运行提供了解决方案。

1　实时流量中间件接入模型

1.1　网络驱动接口规范简介

中间驱动程序是网络驱动接口规范（Network Driver Interface Specification，NDIS）所支持的 3 种网络驱动类型之一，它处于协议驱动程序与网卡驱动程序之间，其在通信系统中的特殊位置使得从上层向网络发送的数据包和从外界接收的数据包都须经过它之后再继续传递，因而利用中间驱动程序可实现对网络数据包过滤和协议转换，这也为 AFDX 网络实时流量中间件的实现提供了可能。

1.2　AFDX 实时流量中间件接入模型

AFDX 中间件接入模型包括配置文件解析模块、协议转换模块与流量整形模块。配置文件解析模块通过对用户配置文件的解析，为中间件其他模块提供参数。

协议转换模块用于对数据帧实施 AFDX 协议与 TCP/IP 协议帧格式的相互转换。流量整形模块的功能是对普通以太网下无序的数据流进行时间调度，在单条链路上实现 BAG 整形，并在此基础上实施抖动最小的多链路复合算法，保证进入到 AFDX 网络数据流的时间确定性。AFDX 中间件接入模型系统结构以及数据流流向如图 1 所示。

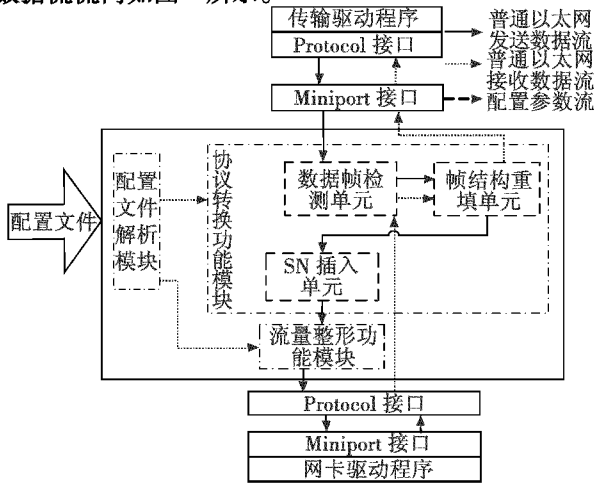


图 1　AFDX 通信中间件结构图

Fig. 1　Structure of AFDX communication middleware

2　协议帧格式转换

为实现基于 TCP/IP 协议的航电应用与 AFDX 网络的透明通信，中间件模型需要对数据帧进行帧格式转换。AFDX 网络的控制机制使其寻址方式与 TCP/IP 协议不同，同时其帧格式在数据域之后还存在顺序号（Sequence Number，SN）域。因此，协议转换模块须实现数据帧报头字段的重组和 SN 插入。AFDX 协议帧寻址方式如图 2 所示。

以太网 MAC 控制器标识符(48 bit)			
固定域 24 bit	User_Defined_ID 16 bit	Interface ID 3 bit	固定域 5 bit
0000 0010 0000 0000 0000 0000	nnnn nnnn nnnn nnnn	mmm	0 0000

a AFDX 协议源 MAC 格式

48 bit	
固定域 32 bit	虚拟链路 标识符 16 bit
xxxx xx11 xxxx xxxx xxxx xxxx xxxx xxxx	

b AFDX 协议目的 MAC 格式

IP 单播地址格式(源或单播目的)(32 bit)				
A 类 1 bit	私有地址 IP 7 bit	User_Defined_ID (用户定义标识)16 bit	Partition_ID (分区标识)	
0	000 0010	nnnn nnnn nnnn nnnn	空余域 (000)3 bit	5 bit

c AFDX 协议源 IP 格式

IP 寻址格式(32 bit)		
4 bit	28 bit	
D 类 1110	IP 多播标识符	
	固定域 12 bit = 0000 1110 0000	虚拟链路标识 符 16 bit

d AFDX 协议目的 IP 格式

图 2　AFDX 协议帧地址格式

Fig. 2　Address format of AFDX frames

在 IPv4 数据包结构中，总长度域的取值范围为 21 ~ 1500 字节，在 AFDX 中由于存在 SN，总长度域（不将 SN 考虑在内）的范围是 21 ~ 1499 字节。SN 按数据帧次序由 0 至 255 以步长 1 循环递增。同时，AFDX 网络为保护数据流，采用冗余的 A/B 网络来进行数据传递，这意味着一个数据帧及其备份会分别进入到 A/B 网络，而它们拥有相同的 SN。因此，在进行 SN 插入时，应区分进入不同网络的数据帧对 SN 分开计数，以记录数据帧的正确次序。

当数据帧在协议转换模块入口被截获后，中间件模型将数据帧展开，将用户配置的检测信息与数据帧内容进行匹配，对检测到需要处理的数据帧，实施由 MAC 头至 UDP 头的重新填充，使其满足协议格式规范要求。

在中间驱动程序中传递的数据包以 NDIS_PACK-

ET类型的结构体存在。NDIS_PACKET为包描述符，一个数据包的包描述符指示了一系列以链表形式连接的缓存描述符NDIS_BUFFER。每片缓存的虚拟地址空间分别对应数据包各层实际占用的物理内存。

依照NDIS中数据包的存储方式，采取新建NDIS_BUFFER连接在原包描述符末尾的方法实现SN插入。实施SN插入的核心算法如下。

```
NewStatus NdisAllocateMemory(&bufferVA) Δ分配内存
If NewStatus == NDIS_STATUS_SUCCESS
then
NewStatusB NdisAllocateBufferPool(&NewStatusB,&PoolHandle)
    if NewStatusB == NDIS_STATUS_SUCCESS
    then
NewStatusC NdisAllocateBuffer(&NewStatusC,&pAddBuffer,&PoolH
-andle,&bufferVA) Δ分配缓存描述符,并与分配的内存映射
      if NewStatusC == NDIS_STATUS_SUCCESS
      then NdisChainBufferAtBack(pAddBuffer)
Δ将新增缓存描述符连接至缓存描述符链表末尾
SNDomain
MmGetSystemAddressForMdlSafe(Packet-> Private. Tail)
        If Partion_ID =0b000 then *(SNDomain)(SN_A ++)
        else then *(SNDomain)(SN_B ++)
Δ检验A/B网络,分别进行顺序号循环计数
```

3 时间确定性调度分析

流量整形是AFDX网络确定性机制实施的根本保障技术之一。通过对同一逻辑链路承载的前后相邻数据帧之间的时间间隔进行限制和约束，降低和平滑了VL上数据帧的突发流量，保证了VL逻辑带宽预分配机制，并增强了网络通信确定性。流量整形以单条VL为基本单元，在每个BAG间隔中发送帧的数目不会多于一个。对于多条VL上的数据流，在数据进入到接收端前，需对它们进行有效调度，以保证对于给定的VL数据帧能够在最大允许抖动内到达。

3.1 单条链路的BAG整形

在AFDX网络中，对于单条VL，数据帧没有抖动时BAG反映了两个相邻帧的起始二进制位之间的最小时间间隔。基于TCP/IP协议的航电应用通信所产生的数据流，帧间间隔并不受约束。因此当航电应用向AFDX网络发送数据时，数据流很可能无法满足AFDX网络对帧间间隔的要求。BAG整形即是完成数据流帧间间隔的调整。数据帧在经过中间件协议转换处理后包含了链路号信息。根据链路号，可将单条数据流上的数据帧划分至多条VL，然后对各VL实施BAG调整。数据流实际的帧间间隔 T_{RealBAG} 与用户配置的BAG值 $T_{\mathrm{BAG}u}$ 的大小关系是判别是否进行流量整形的条件。由AFDX协议规范可知，只有 $T_{\mathrm{RealBAG}} < T_{\mathrm{BAG}u}$ 的数据帧需要被处理。BAG整形核心算法通过内核延迟实现数据帧延迟发送，使数据帧的帧间间隔增大至 $T_{\mathrm{BAG}u}$，同时为保证流量整形的精确度，还对实际帧间间隔实时监控，动态调整延时长度。

3.2 多链路复合时间调度算法

经过BAG整形的各VL在进入到AFDX端系统之前，需要复合成为单条多路复用流。由于各VL数据抵达的时间不确定，因此在复合时，会出现对物理链路的争抢，导致复合后的数据流规整度降低，进入到AFDX网络的时间不确定性加大。为避免此种情况，设计了实时流量抖动最小的VL复合时间调度算法(Timing-Send,T-Send)，中间件通过该算法来限定不同VL数据流抵达物理链路的时间，分时利用物理链路，从而实现VL的无冲突复合。

AFDX协议规范中规定 $T_{\mathrm{BAG}i}=2^i$(单位为ms)，$i=1\sim7$。定义一组虚拟链路集合 $VL_{C0}\sim VL_{C7}$，每个虚拟链路集合 VL_{Ci} 中包含 J_i 条BAG相等的VL，不同虚拟链路集合BAG不等。假定将虚拟链路集合按BAG大小进行排列，将 VL_{Ci} 的BAG值记为 $T_{\mathrm{BAG}i}=2^i$。参数定义：设物理带宽为 C；集合 VL_{Ci} 中最大数据帧长为 $L_{\mathrm{max}i}$；所有链路上的最大帧长 $L_{\max}=\max\{L_{\mathrm{max}i}\}$，$i=0,\cdots,7$；令 $J_{\max}=\max\{J_j\}$，$i=0,\cdots,7$；VL_{Ci-s} 属于集合 VL_{Ci}，且该链路上首个消息的抵达时刻为 VL_{Ci} 上所有VL的首个消息抵达时刻中排在 s 位的时刻。m 为最小BAG间隔内所能容纳的虚拟链路集合数且 $m=\left|\dfrac{T_{\mathrm{BAG}_0}}{J_{\max}L_{\max}/C}\right|$。

命题：当 J_i 满足 $0\leqslant J_i\leqslant\left|\dfrac{T_{\mathrm{BAG}i}}{L_{\mathrm{max}i}/C}\right|$ 且 $m\geqslant2$ 时，若设定集合 VL_{Ci} 上经过BAG整形的消息中，链路 VL_{Ci-1} 首个数据包的抵达时刻 $t_{i1}=(2^i-1)\dfrac{J_{\max}L_{\max}}{C}$，且 VL_{Ci-s} 上首个数据包抵达时刻 $t_{is}=t_{i1}+s*\dfrac{L_{\mathrm{max}i}}{C}$，$s=2,\cdots,J_i$，则这 $\sum_{i=0}^{7}J_i$ 条VL复合后的数据流依然能保证其各自帧间间隔进行传递。

证明：取最不利情况，即 $m=\left|\dfrac{T_{\mathrm{BAG}_0}}{J_{\max}L_{\max}/C}\right|=2$，由于 VL_{Ci} 上被规整后的消息中，首批数据包(一批数据包指属于同一链路集合，但不同VL的一组数据包)抵达时刻 $t_{i1}=(2^i-1)\dfrac{J_{\max}L_{\max}}{C}$，又 $T_{\mathrm{BAG}i}=2^i$，那么该集合上第 l 批数据包抵达的时间为

$$t_{i1\_l}=(2^i-1)\frac{J_{\max}L_{\max}}{C}+2^i*(l-1)\qquad(1)$$

同理，可得到 VL_{Cj} 上规整后第 l 批数据包抵达时间为 t_{j1_l}。假设存在数据包冲突，则有

$$t_{i1\_l}=t_{j1\_l'},0\leqslant i,j\leqslant 7\text{ 且 }i\neq j \tag{2}$$

将式(1)代入式(2)可得

$$(2^i-1)\frac{J_{\max}L_{\max}}{C}+2^i*(l-1)=(2^j-1)\frac{J_{\max}L_{\max}}{C}+2^j*(l'-1) \tag{3}$$

假设 $j<i$，令 $l''=l'-1-2^{i-j}*(l-1)$。则上式可化简为

$$2^j*(2^{i-j}-1)*\frac{J_{\max}L_{\max}}{C}=2^j*l'' \tag{4}$$

因为 $T_{\text{BAG0}}=2^0\approx 2\frac{J_{\max}L_{\max}}{C}$，则有 $\frac{J_{\max}L_{\max}}{C}\approx\frac{1}{2}$，代入上式有

$$2^{i-j}-1=2*l'' \tag{5}$$

l''为整数，故式(5)无解，因此通过上述调度能够使 BAG 不等的 VL 复合后的数据流之间无冲突。

在集合 VL_{Ci}的一个 BAG 中，链路 VL_{Ci-s}上首个数据包抵达时刻 $t_{is}=t_{i1}+s*\frac{L_{maxi}}{C}$，因而该链路上的第 k 个数据包到达的时间为

$$t_{is\_k}=t_{i1}+s*\frac{L_{maxi}}{C}+k*T_{\text{BAG}i} \tag{6}$$

同样可得到链路 $VL_{Ci-s'}$上的第 k'个数据包到达的时间 $t_{is'\_k'}$。假设存在数据冲突，则有

$$t_{is\_k}=t_{is'\_k'} \tag{7}$$

将式(6)代入式(7)可得

$$t_{i1}+s*\frac{L_{maxi}}{C}+k*T_{\text{BAG}i}=t_{i1}+s'*\frac{L_{maxi}}{C}+k'*T_{\text{BAG}i} \tag{8}$$

令 $s''=s-s'$，$k''=k'-k$，化简得

$$s''*\frac{L_{maxi}}{C}=k''*T_{\text{BAG}i} \tag{9}$$

又 $0\leqslant J_i\leqslant\left|\frac{T_{\text{BAG}i}}{L_{maxi}/C}\right|$，且 $s\leqslant J_i$，故 $s''*\frac{L_{maxi}}{C}<k''*T_{\text{BAG}i}$，与式(9)矛盾，所有式(9)无解，即对于 BAG 相等的 VL，复合后的数据流之间也无冲突，因此对任意虚拟链路有 $J=0$。

在 $m=\left|\frac{T_{\text{BAG}_0}}{J_{\max}L_{\max}/C}\right|>2$ 的情况下，同样可利用上述分时抵达的方法对各 VL 进行调度。同时为了更充分利用物理链路资源，可在最小的 BAG 内安排多于两组虚拟链路集合。此时，各 VL 抵达物理链路的时刻计算复杂度加大。

综上所述，如果限定不同 VL 上消息从中间件输出的时刻点，那么这 $\sum_{i=0}^{7}J_i$ 条 VL 就能有效地利用物理链路资源，实现无冲突复合，且复合后的消息流能按照各自帧间间隔进行传输。

标准的 AFDX 虚拟链路调度器以 FIFO(First Input First Output) 方式将数据包发送到物理链路上。利用网络演算的方法可以计算在多条 VL 复合时利用 FIFO 方式调度所产生的抖动延迟[5-9]。对于一个有 n 条 VL 的端系统，如果第 i 条虚拟链路的 BAG 记为 $T_{\text{BAG}i}$，最大数据帧长为 L_{maxi}，那么由其调度算法引起的该条 VL 上的最大抖动为 $J_i=\frac{1}{C}\sum_{i=1}^{n}L_{maxi}$[6]。而本文提出的时间调度模型，采取限定数据流抵达物理链路时刻的方式实现了各条 VL 无冲突复合，使得在理论上有 $J_i=0$，从而保证了进入到 AFDX 的网络流量比采用传统的 FIFO 调度方式具有更高的时间确定性。

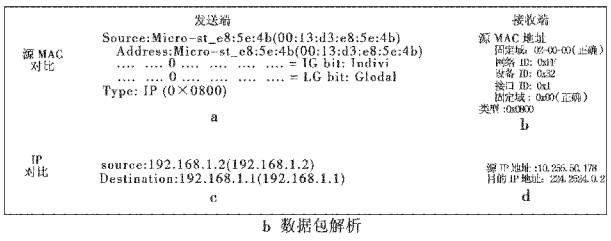
4　实验和结果分析

为评价 AFDX 中间件接入模型的性能，验证其协议转换以及时间确定性保障功能，将 AFDX 中间件模型安装在普通以太网卡终端进行了实验。实验系统由一台普通以太网卡终端与一台航空电子交换式全双工以太网卡终端组成，两个网络通过网线互联进行消息发送。

为验证 AFDX 通信中间件协议转换功能的正确性，在配置文档中设定待处理源端口号为 5，并填写其他配置参数，例如，设定用户自定义标识为 0xFF32、分区标识为 0b001 等。在普通以太网卡终端分别以源端口 137 以及 5 向 AFDX 网卡终端发送数据包，在 AFDX 网卡终端抓包显示结果见图 3。

源端口	目的...	源 IP	目的 IP	源 MAC	目的 MAC	源分区	SN
137	137	192.168.1.2	192.168.1.255	00-13-D3-E8-5E-4B	FF-FF-FF-FF-FF-FF	2	1
137	137	192.168.1.2	192.168.1.255	00-13-D3-E8-5E-4B	FF-FF-FF-FF-FF-FF	2	1
5	40	10.255.50.178	224.224.0.2	02-00-00-FF-32-20	03-00-00-05-00-02	178	0
5	40	10.255.50.178	224.224.0.2	02-00-00-FF-32-20	03-00-00-05-00-02	178	1
5	40	10.255.50.178	224.224.0.2	02-00-00-FF-32-20	03-00-00-05-00-02	178	2
5	40	10.255.50.178	224.224.0.2	02-00-00-FF-32-20	03-00-00-05-00-02	178	3

a 接收端抓包结果



b 数据包解析

图 3　接收端抓包与发送端抓包结果数据帧解析对比

Fig. 3　Frame-parsing comparison between the sender and receiver

由抓包结果来看，来自于用户配置源端口(端口 5)的 UDP 数据包被中间件处理，数据帧报头字段进行了重填，并且含有正确的 SN，帧格式符合 AFDX 协议规范。对数据帧进行协议解析可知，经协议转换后，数据帧地址格式中所含参数与用户配置参数一致。由此可见中间件能够正确解析配置文件解析并进行协议转换。

为验证 AFDX 中间件的流量整形功能，设置参数 $T_{\text{BAG}u}=128$ ms，普通以太网卡终端数据包发送工具选择自动发送 UDP 包，时间间隔 T_{SendGap}分别设置为 50 ms 与 150 ms。在收发两端同时进行抓包，对两种 T_{SendGap}

下选取4000 ms内收发两端抓获的数据包时间间隔进行统计,统计结果如图4所示。

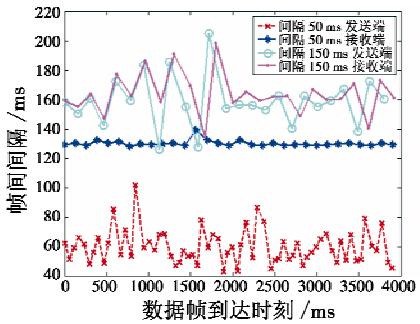


图4 单链路整形调度结果统计图

Fig. 4 Statistic of BAG shaping results

图4中,首个数据帧对应时刻设为0。由图4可知,当 $T_{SendGap}=50$ ms时,发送端数据流的帧间间隔在50 ms左右波动较大,接收端数据流规整有序,帧间间隔基本保持为128 ms。这是由于 $T_{SendGap}=50$ ms < $T_{BAGu}=128$ ms,中间件模型对数据流进行了流量整形。当 $T_{SendGap}=150$ ms时,发送端的数据流帧间间隔均处于128 ms以上,但在150 ms左右波动较大。波动是由于发送端通信源的时间不确实性造成的。由于帧间间隔已符合BAG的要求,中间件模型识别出该数据流不需BAG处理,所以在接收端,接收到的数据流帧间间隔情况与发送端数据流基本无异。由此可知,中间件模型不会增加数据流的不稳定性。在时间轴上观察统计结果不难发现,以50 ms为间隔发送数据包时,发送端较为密集的数据流经过中间件的整形后变得缓和。

针对FIFO算法和T-Send算法引起各链路上的抖动延时,采用Matlab进行了仿真实验。网络仿真共使用了24条VL,总带宽设置为10 Mb/s。并根据实际需要将 L_{maxi} 均设置为1518 Bytes,包含的 T_{BAGu} 有1 ms、2 ms、4 ms、8 ms、16 ms、32 ms、64 ms、128 ms,且每个 T_{BAGu} 值对应有3条虚拟链路。对于FIFO算法,无优先级。对于T-Send算法,在计算由其引起的抖动延时时,将中间件模型的处理时间考虑在内,中间件模型处理时间在100 ~ 200 μs范围内波动。仿真结果统计如图5所示。

由图5可知,使用FIFO算法复合后,数据流的抖动值比较大而且分布不均匀,均值为884 μs,均方值为357.1 μs。在同样条件下,使用T-Send算法复合后产生的抖动的均值为162 μs,均方值为18.1 μs。因此T-Send算法通过对各链路在时间上进行有效分配使数据流复合引起的抖动时延远远小于FIFO算法,并且其抖动值更加平稳。图中所显示的T-Send算法下抖动时延为非0值,这是由于计算时包含了中间件模型处理数据流的时间。综上所述,通过单链路上的BAG整形以及多链路复合的T-Send算法,AFDX中间件接入模型能够保证进入到AFDX网络数据包流的时间确定性。

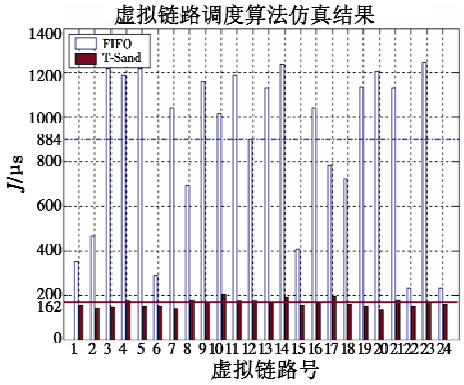


图5 多链路复合调度引起抖动时延结果图

Fig. 5 Statistics of jitter caused by VL multiplex algorithm

5 结束语

本文在NDIS基础上,提出了适用于AFDX网络的实时流量中间件接入模型,并针对多链路复合提出了T-Send时间调度方法,成功实施了基于TCP/IP协议的航空电子应用在AFDX网络上的确定性接入过程,已有航空电子应用(第三方应用)不会感知在接入航空电子全双工和普通以太网的不同,在网络的升级过程中,最大化保护了已有开发资源,降低了网络接入难度。而且,本文中提出的T-Send时间调度算法,结合单条链路上的BAG整形,引起的抖动时延要远小于标准的AFDX虚拟链路调度FIFO算法,使进入到AFDX网络的数据流具有更好的时间确定性。

参 考 文 献

[1] ARINC 664. Aircraft date network, part 7: Deterministic networks [M]. USA: ARINC Compration, 2003.

[2] 熊华钢,周贵荣,李峭. 机载总线网络及其发展 [J]. 航空学报,2006,27(6):1135-1144.

[3] 张云勇,张智江. 中间件技术原理与应用 [M]. 北京:清华大学出版社,2004.

[4] TILEVICH E, SMARAGDAKIS Y. NRMI: Natural and efficient middleware [J]. IEEE Transaction on Parallel and Distributed Systems, 2008,19(2):174-187.

[5] FLOROIU J W, LONCACU T C, RUPPELT R, et al. Using NDIS intermediate drivers for extending the protocol stack a case study [J]. Computer communications, 2001,24:703-715.

[6] 张勇涛,黄臻,熊华钢. 保证速率的AFDX交换机实时调度算法[J]. 北京航空航天大学学报,2010,36(12):1412-1416.

[7] 陈昕,周拥军,蒋文保,等. AFDX协议性能分析及调度算法研究[J]. 电子学报,2009,37(5):1001-1005.

[8] SCHARBARG J L, RIDOUARD F, FRABOUL C. A probabilistic analysis of end-to-end delays on an AFDX avionic

(下转第76页)

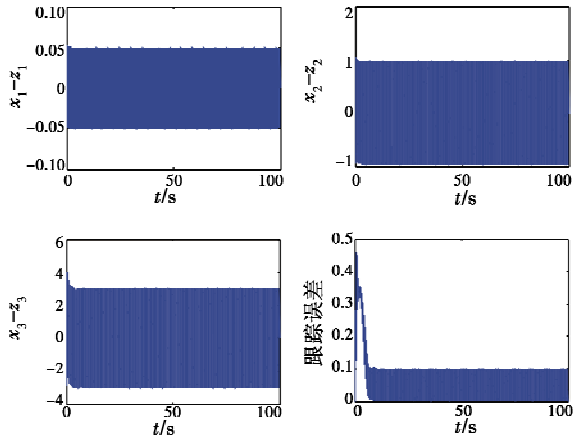


图3 高频噪声下的系统误差

Fig. 3 System error with high frequency noise

考虑在使用 ESO 之前对输出信号 y 进行滤波处理，即

$$y_F=\frac{\omega_n}{s+\omega_n}y \tag{19}$$

其中，$\omega_n=20$。于是扩张状态观测器方程改为

$$\begin{cases}\dot{z}_1=z_2+\beta_1(y_F-z_1)\\ \dot{z}_2=z_3+\beta_2(y_F-z_1)\\ \dot{z}_3=\beta_3(y_F-z_1)\end{cases} \tag{20}$$

仿真结果如图4所示。

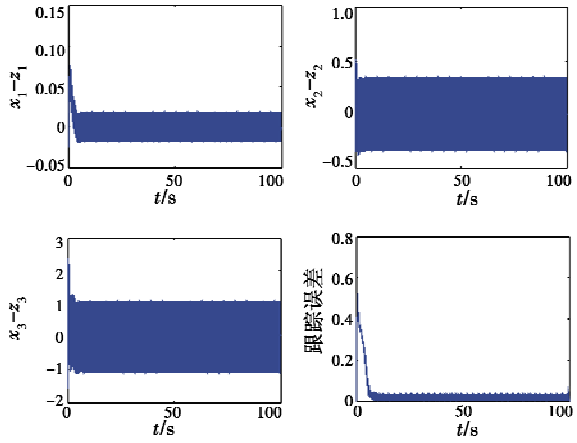


图4 滤波后高频噪声下的系统误差

Fig. 4 System error with high frequency noise after filtering

由图3与图4可知，高频正弦噪声情况下，估计误差 $x_j-z_j(j=1,2,3)$ 随着阶次的升高而增加。设计低通滤波器可以有效地减少噪声对系统的影响，从而验证了结论3。

4 结论

本文讨论了含有量测噪声条件下自抗扰控制器 ADRC 对付非线性不确定性动态的能力，分析了在常值噪声、低频噪声以及高频噪声条件下 ADRC 的估计精度，并通过仿真实例进行了验证。ADRC 尽管在很多领域有了大量的应用，然而量测噪声的存在不可避免。如何与各种滤波方法相结合，如何分析与处理随机噪声等大量问题都值得继续研究。

参 考 文 献

[1] 韩京清. 自抗扰控制技术[M]. 北京：国防工业出版社，1998.

[2] HOU Y, GAO Z, JIANG F, et al. Active disturbance rejection control for web tension regulation [C]//Proceedings of the 40th IEEE Conference on Decision and Control, 2001:4974-4979.

[3] HUANG Y, LUO Z W, SVININ M, et al. Extended state observer based technique for control of robot systems [C]// Proceedings of the 4th World Congress on Intelligent Control and Automation, 2002:807-2811.

[4] FENG G, HUANG L. A new robust algorithm to improve the Y. Liu dynamic performance on the speed control of induction motor drive [J]. IEEE Transactions on Power Electronics, 2004, 19(6), 1614-1627.

[5] 黄朝东. 几类非线性不确定性系统的控制与反馈能力研究 [D]. 北京：中国科学院数学与系统科学研究院，2012.

[6] BALL A A, KHALIL H K. High-gain observers in the presence of measurement noise: A nonlinear gain approach [J]. Automatica, 2009(45):936-943.

[7] AHRENS J H, KHALIL H K. High-gain observers in the presence of measurement noise: A switched-gain approach [C]//Proceedings of the 47th IEEE Conference on Decision and Control. Mexico, Dec. 9-11, 2008:2288-2293.

[8] KHALIL H K. Nonlinear Systems [M]. Prentice Hall: Upper Saddle River, New Jersey, 2002.

[9] XUE W, HUANG Y. Comparison of the DOB based control, a special kind of PID control and ADRC control [C]// Proceedings of 2011 American Control Conference, 2011: 4373-4379.

[10] YANG X, HUANG Y. Capability of extended state observer for estimating uncertainties [C]//Proceedings of 2009 American Control Conference, the 2009:3700-3705.

(上接第66页)

network [J]. IEEE Trans Industrail Informatics, 2009, 5 (1):38-49.

[9] BOYER M, FRABOUL C. Tightening end to end delay upper bound for AFDX network calculus with rate latency FIFO servers using network calculus [C]//IEEE International Workshop on Factory Communication Systems. Dresden: IEEE, 2008:11-20.