

AFDX 交换机的队列整形调度研究

郭利锋, 王 勇, 张 磊, 白 焱

(空军工程大学工程学院, 西安 710038)

摘要: AFDX 交换机的静态优先级非抢占式队列调度(SPUS)算法不能消除输出分组中的随机抖动。为此, 采用将虚拟时钟作为分组调度发放合法时间的方法对 SPUS 算法进行改进。在发送级联端口的分组输出之前, 先经过整形调度使数据流均匀地输出, 以确保交换机的服务质量。仿真结果表明, 改进后的算法可消除输出分组随机抖动, 适用于交换机级联端口。

关键词: 航空电子全双工交换式以太网; 交换机; 网络演算; 虚链路; 队列整形调度; 虚拟时钟

Research on Queue Shaping Schedule of AFDX Switch

GUO Li-feng, WANG Yong, ZHANG Lei, BAI Yan

(Engineering College, Air Force Engineering University, Xi'an 710038, China)

【Abstract】 Aiming at the shortage of AFDX switch's Static Priority Un-interrupt Scheduling(SPUS) algorithm in eliminating output random jitter, Virtual Clock(VC) is added to provide legitimate time in packet schedule for ameliorating the algorithm. Packets sending to cascading port enters shaping scheduling before output queue. Shaping scheduling maintains the switch's certainty and quality of service. Simulation results show that the algorithm after ameliorating can eliminate output random jitter, which beseems to switch's cascaded port.

【Key words】 Avionics Full Duplex Switched Ethernet(AFDX); switch; network calculus; Virtual Link(VL); queue shaping schedule; Virtual Clock(VC)

DOI: 10.3969/j.issn.1000-3428.2011.24.019

1 概述

随着航空电子技术的发展, 航电系统对数据总线的要求越来越高。以 ARINC 429 和 1553B 为代表的现役机载总线虽具有可靠性高的优点, 但其固有的传输速率低的缺点, 不能满足现代航空电子系统对通信速率和带宽的要求。航空电子全双工交换式以太网(Avionics Full Duplex Switched Ethernet, AFDX)是在商用交换式以太网的基础上, 通过对实时性和可靠性等方面进行改进建立起来的, 是一种全双工、高数据率、双冗余的新型总线, 充分利用 COTS 技术和开放式标准, 可有效缩短开发周期, 降低研发成本, 已成功应用于 A380 和 B787, 也是我国新研制大型客机的理想选择^[1]。

AFDX 交换机是构成网络的核心部分, 性能的好坏对网络服务的质量有重要影响。本文围绕 AFDX 交换机队列调度机制, 进行了分析和研究, 并针对其不足提出了改进。

2 AFDX 分析

AFDX 的拓扑结构为星型^[2]。为了保证系统的冗余特性, 每一个终端系统与一个交换机之间不仅有一条直接、双向的连接, 同时还与另一个交换机之间保持第 2 条双向连接。交换机与交换机之间级联。端系统与交换机之间或 2 个交换机之间都通过全双工链路连接。每台交换机连接端系统的数量由交换机的端口数和实际情况决定。

AFDX 交换机除了具有以太网交换机对数据帧的接收和转发功能外, 还应提供高质量的 QoS(Quality of Service)服务。ARINC 664 规范第 7 部分规定, AFDX 网络的确定性由端系统实现^[3]。AFDX 交换机只在输入端设置过滤和警管功能, 限制突发分组进入交换机。这样的设计只能保证进入交换机的虚链路(Virtual Link, VL)对带宽使用的确定性。

随着 AFDX 网络的复杂化, 端系统 VL 和交换机级联数

也将增多, 网络负载不断增大, 交换机级联端口输出队列的竞争会改变输出业务流的分组间隔, 产生新的抖动(jitter)。若交换机输出端连接的是端系统, 该抖动会被端系统的确定性机制消除; 若交换机输出端为级联端口, 带有新抖动的分组进入下一个交换机时, 造成下一级交换机产生更大的时延抖动, 并随着交换机的级联不断累加, 严重影响 AFDX 的确定性。目前的 AFDX 调度算法很少着眼于解决交换机级联输出端口的时延抖动问题, 因此, 研究可有效消除 AFDX 交换机级联端口的分组抖动的算法具有较好的应用价值。

AFDX 的拓扑结构如图 1 所示。

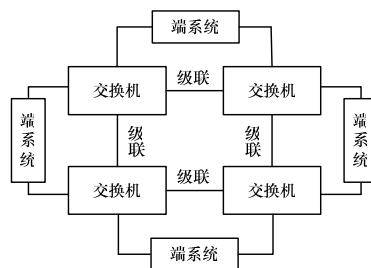


图 1 AFDX 拓扑结构

3 静态优先级非抢占式队列调度算法

AFDX 网络中的分组采用静态优先级方式, 分为高低 2 个优先级, 高优先级为实时消息, 低优先级为非实时消息。ARINC 664 规范第 7 部分规定: 只要高优先级队列非空, 高优先级分组永远优先发送; 低优先级分组正在发送时, 后到的高优先级分组不能打断发送, 必须等到优先级低的分组发

作者简介: 郭利锋(1985—), 男, 硕士研究生, 主研方向: 嵌入式系统, 机载计算机; 王 勇, 副教授; 张 磊、白 焱, 硕士研究生
收稿日期: 2011-06-09 E-mail: guolifeng1985@163.com

送结束^[3]。这种方式符合基于静态优先级的非抢占式队列调度(Static Priority Un-interrupt Scheduling, SPUS)算法。根据文献[4]对 SPUS 算法基于网络演算的分析, 可得出 AFDX 交换机调度器高低 2 个优先级分组的最大时延为^[4]:

$$\begin{cases} D_H = \frac{\sum_{j \in E_H} l_{max}^j + l_{max}^L}{C} + D_{tech} \\ D_L = \frac{\sum_{j \in (E_H \cup E_L)} l_{max}^j}{C - \sum_{j \in E_H} BAG_j} + D_{tech} \end{cases} \quad (1)$$

其中, C 表示交换机的物理链路输出速率; D_{tech} 为固有时延; E_H 和 E_L 分别表示高低 2 个优先队列的聚合数据流; BAG 表示带宽分配间隔。如果交换机内高优先级和低优先级的分组都是严格同步进入的, 式(1)中的 l_{max}^L 为 0, $D_H = \sum_{j \in E_H} l_{max}^j / C$ 。对于同一分组而言, l_{max}^j 和 C 固定, 所以 D_H 固定, 交换时延固定, 抖动为零。此时, 低优先级分组不会对高优先级分组产生影响; 高优先级分组始终可以及时发送出去。若高低 2 个优先级的分组进入的时刻是随机的, 低优先级分组出现而高优先级分组尚未到达时, 调度器会先服务优先级低的分组, 后到的高优先级分组不能打断发送, l_{max}^L 不再为 0, 而是由随机到达的低优先级分组的长度决定。此时高优先级分组的交换延时不再固定, 会产生随机抖动。

由上可知, 在 SPUS 算法下, 高优先级分组的发送延迟明显小于优先级低的分组的延迟。但该算法只考虑了优先级, 没有考虑进入曲线, 会产生随机的时延抖动, 难以保证交换机级联端口的确定性服务。

4 基于虚拟时钟的 SPUS 算法

4.1 算法思想

为消除 SPUS 算法的不足, 实现交换机之间的可靠级联, 为 SPUS 算法加入虚拟时钟(Virtual Clock, VC), 提出基于虚拟时钟的 SPUS 算法。通过在交换机内设置一个虚拟时钟来计算分组的预计发送时间, 并按大小进行排序, 以此顺序作为队列调度转发的依据。同时, 为到达分组发放合法发送时间, 同一优先级内优先发送结束时间最短的分组。虚拟时钟的主要功能是控制随机数据流的传输, 用一个根据虚拟时间排队的队列来模拟公平队列调度中的多个队列。

调度过程中, 数据流的排序和转发都是根据交换机内的 VC 完成的。VC 是一个 48 位的计数器, 上电复位后, 由 0 开始递增计数, 时间精度是 250 ns, 即每 250 ns 计数器自增 1^[5]。为了便于 AFDX 网络统一计时, VC 由 CPU 配置校准。当分组的合法发送时间没有到来之前, 分组会被阻塞, 即使此时系统空闲, 也不进行转发。合法时间的设置应符合 AFDX 协议要求, 即分组的合法发送时间为前一分组的发送时间加上 BAG 。这样设置的优点是可保证同一虚链路下 2 个连续分组的间隔保持 BAG , 从而保证分组的确定性, 减小分组的随机抖动。

4.2 SPUS 算法与基于 VC 的 SPUS 算法比较

SPUS 算法属于工作保持调度算法^[5], 只要系统中有等待分组, 调度器就一定工作, 即 IP 网络中使用的尽力传递服务。基于 VC 的 SPUS 算法属于非工作保持算法, 即使系统中有等待分组, 合法时间未到来, 也可能暂时不对其进行调度。

SPUS 算法平均交换延时小, 实现简单, 对于不需要完成级联功能端口优势明显。基于 VC 的 SPUS 算法在保证分

组确定性, 消除抖动方面比 SPUS 算法要好, 但加入 VC 后, 用 FPGA 实现时占用的逻辑资源较多, 复杂度高。如果每个队列都采用基于 VC 的 SPUS 算法, 交换机调度性能将会明显下降。

由上可知, 基于 VC 的 SPUS 算法与其他队列调度算法相比, 可最大程度地减少低优先级分组对高优先级分组的时延影响, 消除交换机级联端口的输出分组的抖动, 主要应用于交换机级联端口, 但若应用于非级联端口调度效率不高。

5 AFDX 交换机整形调度

5.1 整形调度算法

根据基于 VC 的 SPUS 算法思想, 本文提出适用于级联端口的基于时间索引表的整形调度算法。算法思想为: 在发送分组时记录分组的实际发送时间 T_{tx} , 根据分组间隔估算该 VL 下一分组的预计发送时间 T_f 并存入时间索引表; 当新分组到达时输出调度从时间索引表中获得 T_f , 根据 VC 指示的当前时刻综合考虑各个 VL 分组, 按照 T_f 值由小到大的顺序对分组进行排队; 当 T_f 值最小的分组达到 VC 时, 分组输出完成整形调度。时间索引表记录分组的 T_f , 允许发送抖动 J_{allow} 等时间信息。

预计发送时间 T_f 是交换机由 VL 前一个分组的实际发送时间 T_{tx} 加上该 VL 的 BAG_j 得来的, 即 $T_f = T_{tx} + BAG_j$ 。每次分组发送完毕会根据 T_{tx} 校正更新时间索引表中的 T_f , 通过校正可以避免分组抖动的累加, 使分组的时延性得到改善。

分组的交换时延 D_s 包括 2 个部分: 固有时延 D_{tech} 和等待时延 D_{wait} ^[5-6]。 D_{tech} 是交换机存储转发的固有时延, 大小由帧长决定; D_{wait} 是分组未得到合法时间时在队列中的等待时间, 大小与上一分组的交换时延和 Max_jitter 有关。ARINC 664 第 7 部分规定 D_s 的最大值为 100 μs ^[3]。若设 T_f 的提前量不超过 10 μs , 当 $(T_f - VC_i) > Y$ (VC_i 是虚拟时钟的当前时刻, Y 是时延阈值), 则按 $T_f = \max(T_f - 10 \mu s, T_f - J_{allow})$ 修正分组的 T_f , 使 D_s 始终小于 100 μs 。

整形调度的关键是调整 D_{wait} 的大小。当抖动导致分组间隔大于 BAG 时, 整形调度将优先发送这些分组, 使 D_{wait} 接近 0; 当分组间隔小于 BAG 时, 整形调度可以通过改变 D_{wait} 的大小来吸收 $jitter$, 使分组均匀输出。算法流程如图 2 所示。

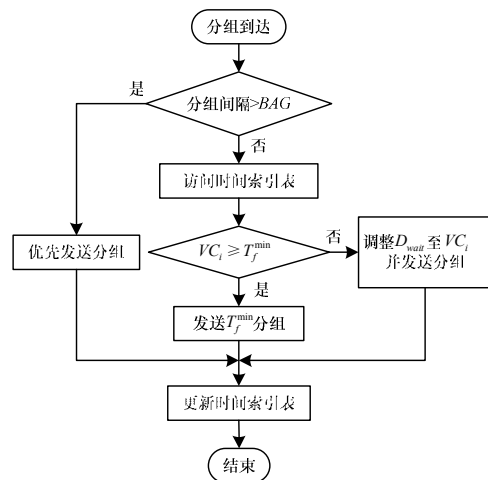


图 2 整形调度算法流程

在图2中, T_f^{min} 为同一优先级分组的最短预计发送时间。若分组被提前发送, 提前的时间会被限制在 AFDX 协议允许的范围内, 并且提前量会在每次发送结束后进行修正, 保证提前被发送的分组不会被下一级交换机的警管模块丢弃。

5.2 整形调度器

按照上述基于时间索引的整形调度算法, 为 AFDX 交换机的级联端口增加整形调度器, 如图3所示。

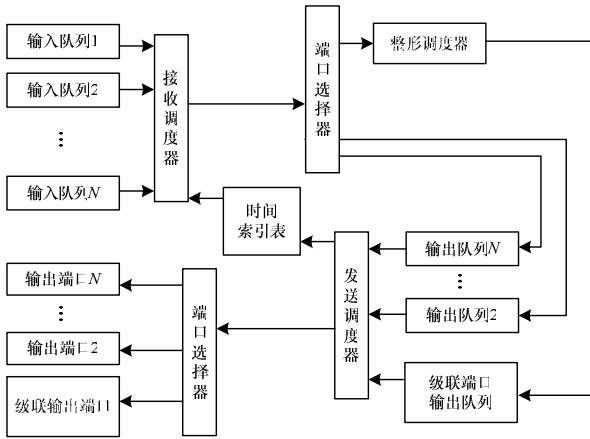


图3 AFDX 交换机整形调度

在 AFDX 交换机级联端口加入整形调度器。发往级联端口的分组在进入输出队列之前先进入整形调度器, 经过整形且满足要求的分组再进入级联端口的输出队列。发送调度器将级联端口输出队列的分组发往端口选择器, 经过端口选择发送到级联输出端口, 完成分组调度。整形调度减小了分组突发产生的抖动, 使级联输出端口数据流均匀。发往非级联端口的分组按照交换机的正常调度完成分组的交换功能。

6 仿真与分析

根据 AFDX 的拓扑结构和 ARINC 664 规范, 在 OPNET 仿真环境下建立包括 16 个端系统和 1 个交换机的 AFDX 网络仿真模型, 对 2 种算法进行仿真测试, 验证不同算法下输出分组的确定性。

端系统模型包括: VL 分组产生, VL 分组成形和 VL 分组汇聚模型 3 个部分, 按照每个 VL 所配置的 BAG 值调整业务发送速率。AFDX 交换机根据 VL 表中的 VL_ID 信息为索引查配置表, 进而将分组搬移到指定的目的输出端口。在交换机的 0 端口加入整形器, 并设置计时器记录同一 VL 的所有分组的发送时间。在交换机的 1 端口~15 端口对 SPUS 算法进行仿真测试, 在交换机的 0 端口对基于 VC 的 SPUS 算法进行仿真测试。使用 Input_scheduler 完成分组的入队操作, 对于 0 端口, Input_scheduler 完成分组信息存储、查找时间注册表和分组的筛选排序。Output_scheduler 完成分组的出队操作, 对于端口 0, Output_scheduler 完成时间标签的修改。Output_buffer 完成时间注册表的更新。交换机模型在 Policing 和 Output_buffer 模块中分别记录分组的到达时间和发送时间, 交换延时等于发送时间与接收时间的差值。

在仿真过程中, 端系统模型共发送 5 120 个分组, 交换机模型记录这 5 120 个分组的到达间隔和发送间隔。限于篇幅, 选取部分数据罗列。横轴表示到达分组序号, 纵轴为分组抖动情况, 范围为(-150 μs, 150 μs)。纵轴上半部分表示抖动为正, 下半部分抖动为负。SPUS 算法下输出分组间隔如图4所示。

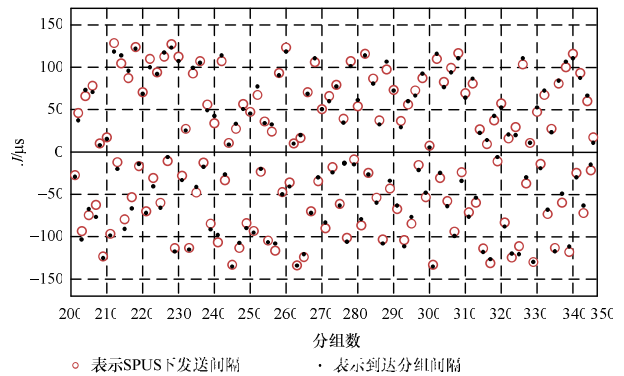


图4 SPUS 算法输出分组间隔

由图4可知, SPUS 算法下到达分组间隔与发送分组间隔非常接近, 输出整形效果不明显。因此, SPUS 算法不能消除输出分组抖动。基于 VC 的 SPUS 算法下输出分组间隔如图5所示。

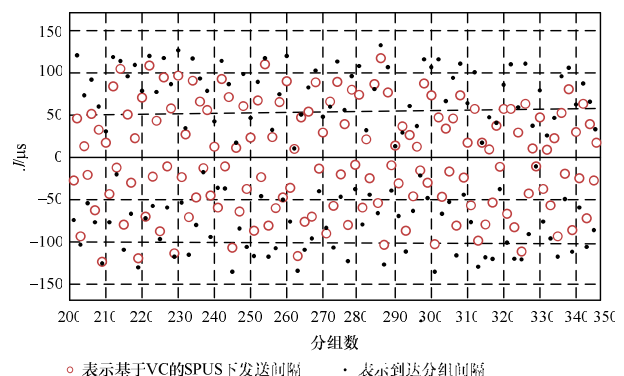


图5 基于VC的SPUS算法输出分组间隔

由图5可知, 基于 VC 的 SPUS 算法下输出分组的抖动变小, 输出分组被整形, 限制了突发分组对输出队列的影响。

通过仿真可知, SPUS 算法虽不能消除输出分组抖动, 但平均交换时延小; 基于 VC 的 SPUS 算法能消除输出分组抖动, 但平均交换时延大。

7 结束语

本文使用网络演算理论分析和研究了 AFDX 交换机的 SPUS 算法。针对该算法的不足进行改进, 比较了改进前后 2 种算法的优劣, 并根据基于 VC 的 SPUS 算法思想设计的整形调度器, 使交换机具备较好的级联能力。仿真结果与理论分析一致, 证明了本文观点的正确性和可行性。

参考文献

- [1] 叶佳宇, 陈晓刚, 张新家. 基于 AFDX 的航空电子通信网络的设计[J]. 测控技术, 2008, 27(6): 56-58.
- [2] 杜亚娟. 航空全双工交换式以太网探究[J]. 计算机工程, 2009, 35(11): 77-79.
- [3] AEEC. 664P7-1-2004 Aircraft Data Network, Part 7-Avionics Full Duplex Switched Ethernet Network[S]. 2004.
- [4] 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): 1000-1005.
- [5] 杨 磊. AFDX 交换芯片的整形调度算法研究[D]. 西安: 西安电子科技大学, 2007.
- [6] 杨曼丽, 李运华, 袁海斌. 网络控制系统的时延分析及数据传输技术研究[J]. 控制与决策, 2004, 19(4): 362-363.

编辑 任吉慧