

机载时间敏感网络流量调度算法性能研究

孔璇, 张双, 刘智武

(中国航空工业集团公司 西安航空计算技术研究所, 西安 710068)

摘要: 时间敏感网络是由 IEEE802.1 工作组开发的一套确定性实时通信的协议标准, 定义了以太网传输的时间敏感机制, 适用于机载安全关键系统网络通信; 流量调度机制可以有效保证机载时间敏感网络实时可靠通信, 时间感知整形器和帧抢占是典型的 TSN 流量调度算法, 目前还没有针对机载业务的传输需求进行性能评估; 通过分析 TSN 系统架构及调度算法的核心特性, 基于机载网络拓扑, 提出一个机载时间敏感网络的数据流传输时延模型, 并进行了数学推导, 使用离散仿真工具构建 TSN 网络仿真平台, 对 3 种典型网络流量调度算法进行验证, 得到了流量传输时延性能的上限值, 分析了流特性参数对调度算法选择的影响; 最后, 对机载应用流量调度算法的选择进行了总结。

关键词: 机载网络; 时间敏感网络; 调度; 时间感知整形; 帧抢占

Research on Performance of Airborne Time Sensitive Network Traffic Scheduling Algorithm

KONG Xuan, ZHANG Shuang, LIU Zhiwu

(Xi'an Aeronautics Computing Technique Research Institute, AVIC, Xi'an 710068, China)

Abstract: Time-sensitive network is a set of protocol standards for deterministic real-time communication developed by the IEEE802.1 working group, which defines the time-sensitive mechanism of the Ethernet transmission and is suitable for the network communication of airborne safety-critical systems; The traffic scheduling mechanism can effectively ensure the real-time and reliable communication of the airborne time-sensitive network, the time-aware shaper and frame preemption are typical traffic scheduling algorithms of the time sensitive network (TSN). Currently, there is no performance evaluation for the transmission requirements of airborne services on the TSN. By analyzing the system architecture of the TSN and key characteristics of the scheduling algorithm based on the airborne network topology, a transmission delay model of the airborne time-sensitive network is proposed, and the mathematical derivation is carried out. The discrete simulation tool is used to build a network simulation platform, and three typical network traffic scheduling algorithms are verified. The upper bound value of traffic transmission delay is calculated, and the influence of stream characteristic parameters on the selection of scheduling algorithm is analyzed. Finally, the selection of traffic scheduling algorithm in airborne scenarios is summarized.

Keywords: airborne networks; time-sensitive network; traffic scheduling; time-aware shaper; frame preemption

0 引言

在硬实时通信系统中, 消息传输的最大延迟必须得到保障且不能容忍任何的数据丢失^[1]。严格的时间确定性, 即确定的网络时延和抖动, 对于机载系统中的安全关键应用是至关重要的。某些带有关键控制信息的数据帧需要按照一定时间周期进行传输, 一旦时延和抖动超过了上限, 可能导致严重的机器、设备损坏及人身伤害。基于不同的设计思想, 目前应用在机载网络的确定性以太网主要有基于时间触发以太网 (TTE, time-triggered ethernet) 和时间敏感网络 (TSN, time sensitive network)。

在 TTE 网络中, 消息被划分为时间触发 (TT, time-triggered) 消息、速率受限 (RC, rate-constraint) 消息和尽力而为 (BE, best-effort) 消息^[2]。通过对链路资源预先规划和静态时间调度表的设计^[3], 实现 TT、RC 和 BE 消息

在同一网络中的数据转发, 保证了时间触发流量的延时是可控的, 并且固定在一定的范围内^[4]。TTE 能够精确描述网络中各种消息的调度行为, 但是实现较为复杂, 并且 TT 业务的分组调度是根据时间调度表对缓存空间的读写操作, 并没有采取基于排队理论的网桥实现, 和传统以太网的转发交换的具体实现和设计思想存在较大差异^[5]。

TSN 是由以太网音视频桥接 (AVB, audio video bridging) 网络演进而来的^[6], 其通过时钟同步、流预留、循环排队、门控列表和抢占等机制保证了数据流的有界低时延。该技术的核心是以时间感知整形 (TAS, time-aware shaper) 和帧抢占为代表的流量调度算法。

TAS 引入了时分多址 (TDMA, time division multiple access) 的思想, 通过门控时间表 (GCL, gate control list) 控制队列门控开关的打开闭合, 在链路上创建独享的保护

收稿日期: 2022-04-08; 修回日期: 2022-04-20。

作者简介: 孔璇 (1989-), 女, 江苏沛县人, 硕士, 工程师, 主要从事机载网络通信方向的研究。

引用格式: 孔璇, 张双, 刘智武. 机载时间敏感网络流量调度算法性能研究[J]. 计算机测量与控制, 2022, 30(6): 235-241.

窗口^[7],使时间敏感业务在保护窗口内无干扰地传输,从而提供确定的时延^[8]。TAS的研究包括最坏情况分析和调度合成。通过最坏情况分析,可以得到通信系统的时延上限。文献[9]研究了基于IEEE 802.1Qbv的预定流量时间感知,结果表明,TAS可以支持非常低的延迟以及抖动保证,同时,利用帧抢占机制,可以增强整形器的性能;考虑到相同优先级帧间干扰的问题,文献[10]分析了TAS与基于信用值的整形器(CBS, credit-based shaper)的组合性能。然而,基于模拟和形式的方法并不能解决TSN的配置问题^[11],应通过调度合成提高网络通信的确定性。文献[12-13]指出,调度合成是通过计算传输路由和GCL共同实现的。

帧抢占允许高优先级流量中断低优先级流量的传输,减少高优先级的阻塞时间,解决了保护窗口过长引起的带宽浪费问题。文献[14-15]分析了帧抢占调度的优势并给出了时延及抖动性能的优化方法。

上述文献试图采用商用的TSN数据流传输场景进行网络仿真,评估流量调度机制对网络确定性的影响,且要求应用和网络完全同步,没有针对不同关键等级的机载业务给出量化的性能评估。针对上述问题,通过探索TAS和帧抢占算法的特征,推导出一个数据流传输时延的数学模型,使用开源TSN网络仿真平台,评估了TAS和帧抢占在机载应用场景下的性能结果,分析算法在可靠性、通信性能和适用场景的不同。

1 TSN 系统模型

1.1 交换架构

IEEE 802.1Q标准中规定了TSN交换机架构,如图1所示,TSN交换机的每个全双工端口都配置了8个优先级队列。到达交换机的数据帧在转发到出端口之前,要经过过滤、计时。出端口的8个队列对应不同的流量类别,交换机根据优先级将数据帧分配给出口队列,传输选择(TS, transmission selection)从优先级最高的队列头部选择符合条件的数据帧,发送到链路上,如果队列整形器的时隙与数据帧配置相符,则允许数据帧传输。TSN工作组规定了各种不同的整形器,如严格优先级和基于信用的整形器(IEEE 802.1Qav)。如图2所示,TAS是在其他整形器之后生效的,通过TS后,MAC负责传输数据帧,这里就涉及到帧抢占机制。

1) 时间感知整形器: TAS是出口队列的一个门控开关,分为基于流的TAS和基于类的TAS,通过打开或关闭对应的出口队列来启用或禁用它,只有启用的队列才能进行数据流的传输。驱动程序根据GCL来周期地改变门控开关的状态。GCL中规定了门控开关的开启时间和保持时间,GCL循环一次所需的时间与网络周期时间相同。

2) 帧抢占: 帧抢占采用了IEEE 802.1Qbu和IEEE 802.3br的抢占式MAC传输机制,将出端口分为2个MAC服务接口,即快速MAC(eMAC, express MAC)和可抢占MAC(pMAC, preemptable MAC)。如图3所示,pMAC

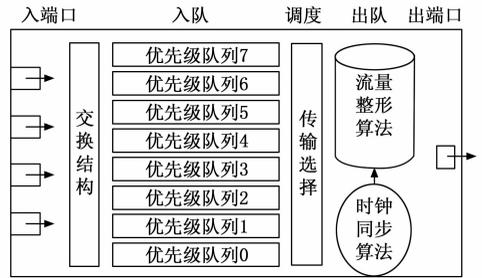


图1 TSN交换机架构

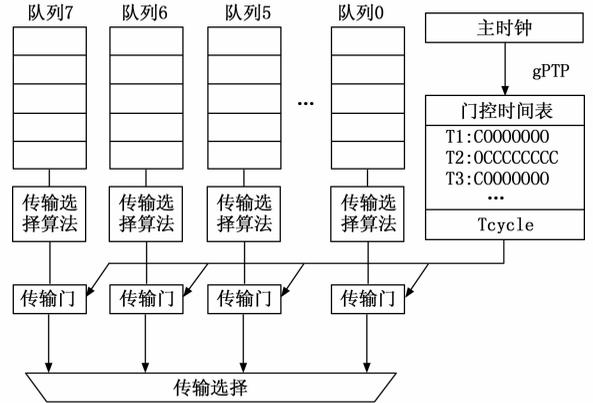


图2 出端口传输选择(TS)和门控开关

可被eMAC抢占,进入队列后,等待eMAC数据传输完成后,再传输pMAC数据。如果没有帧抢占,一个最大传输单元(MTU, maximum transmission unit)帧(1500字节)在1Gbps速率下,能够阻塞端口约12.5μs,而帧抢占将最大阻塞时间限制为1μs。如果发生了抢占,被抢占帧的剩余部分的数据负载将重新整合成新的帧,称为分片,其长度至少为64字节。端口的出口队列分为快速队列或可抢占队列,分别在eMAC和pMAC中处理快速帧和可抢占帧。

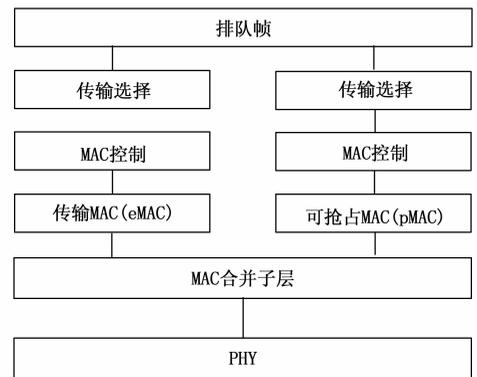


图3 帧抢占机制

1.2 流

流是指具有传输方向的数据流。TSN将数据流分为同步流、循环流和“尽力而为”流。对于同步流而言,发送周期需要高精度网络时钟同步技术的支持,数据流按照规

定的起始时间和偏移, 在指定的网络周期内完成传输。同步流具有最小的端到端时延和最小的抖动上限, 例如某些运动控制应用。循环流的发送周期无需与网络周期同步, 其传输起始时间可以是网络周期内的任何时刻, 其端到端延迟不超过一个网络周期, 抖动以周期时间来衡量。一个网络周期内, 同步流和循环流只传输一帧, 最大帧长为 300 字节。TSN 需要保证最大数据负载的同步流和循环流的时延上限。

1.3 帧延迟

为了确定网络传输时延和抖动上限, 需要分析端到端时延的组成因素。

考虑一个路径长度为 3 的数据流: 发送方——交换机——接收方。从发送方开始传输数据到交换机接收到第一个比特的时间间隔为传播延迟, 取决于电缆的材料和长度 ($\approx 5 \text{ ns/m}$); 从接收到数据帧的第一个比特到接收到全部帧的时间间隔为帧相关延迟, 取决于帧长和数据速率:

$\frac{l_{\text{frame}} \text{ bit}}{r_{\text{link}} \text{ bit/s}}$; 交换机收到数据帧后, 对该帧进行处理并分配出

端口, 这个时间为帧无关延迟, 与帧大小无关; 数据帧在出端口排队等待, 直到它被 TS 选中, 该帧在出口队列中的等待时间为排队延迟, 取决于所在队列前方的帧数, 以及出口队列的 TAS 门控调度。排队延迟分为门控延迟和干扰延迟, 门控延迟表示由于传输门关闭而使当前帧排队的时间, 有确定上限; 干扰延迟表示由于其他帧的传输而使当前帧排队的时间, 依赖于出端口的当前负载。

发送方发出一帧数据, 从发送方开始传输数据到接收方收到全部数据的时间间隔称为端到端延迟, 由传播延迟、帧相关延迟、帧无关延迟及排队延迟共同决定, 其中三者统称为路径延迟, 每个流的路径延迟是恒定的, 它只取决于路径长度和帧长。因此, 保证端到端的传输延迟需要确定排队延迟的上限。

1.4 网络拓扑

在时间敏感网络中, 主要的网络构件包括端系统、网络交换机和物理链路。

端系统作为数据流的发送方或接收方, 需要同时支持同步流、循环流和“尽力而为”流的收发处理。与传统以太网不同, TSN 端系统集成了时钟同步模块和流量整形模块, 基于统一的全局时钟, 能够满足数据流无冲突的发送和接收。

网络交换机主要功能是同时支持 TSN 三种数据流的转发, 同时, 交换机集成了时钟同步模块、流量过滤管模块和数据容错控制机制。时钟同步模块记录数据流通过交换机入端口和出端口的高精度时间戳, 以支持网络各端口队列的流量整形算法; 通过流量过滤管制模块, 可以解决数据流在交换机内部的带宽占用问题, 保证数据流能够在流量整形器计划的时延窗口转发出去; 交换机转发数据流的整个过程, 采用数据容错控制机制对错误进行恢复。

TSN 端系统与交换机之间以及交换机与交换机之间,

通过物理链路两两互连的, 构成的网络拓扑按照交换机的互连方式可以分为总线型和交换型, 如图 4 所示。

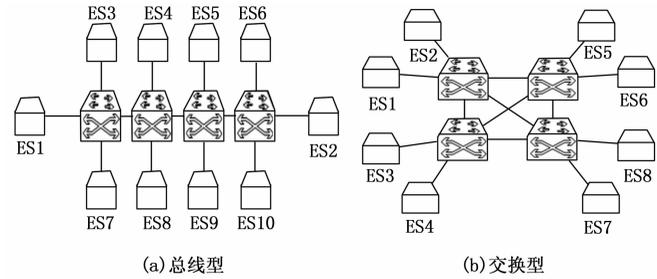


图 4 时间敏感网络拓扑结构

总线型拓扑是时间敏感网络的典型拓扑结构, 在工业界已得到广泛应用。在总线型拓扑中, 每个交换机都需要连接一个或多个端系统, 且每个交换机都与其相邻交换机相连。尽管拓扑结构简单, 但不同调度算法在总线型网络上的表现出的特征和性能仍存在差异。如果网络系统需要链路冗余, 总线型拓扑的通信路径可以扩展成一个环, 即将收发两端连接成环形, 以防止网络分区。

为了提高网络的传输带宽和冗余容错能力, 机载网络通常采用交换型拓扑结构。在交换型拓扑中, 端系统连接到交换机上, 交换机采用级联的方式构成交换网络, 交换机与交换机之间至少存在两条相互隔离的通信链路。

2 TSN 调度算法

2.1 帧抢占

帧抢占原理如图 5 所示, 当链路中有可抢占帧时, 快速帧提出传输请求, 发送方首先判断当前状态下是否允许切片操作, 如果允许, 则在适当位置中断可抢占帧的传输。待快速帧传输完成后, 被中断的可抢占帧后半部分会被加上适当的前导码继续传输。接收方根据前导码来判断帧的类型、进行帧的重组。

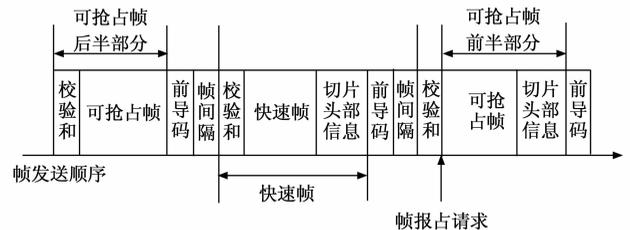


图 5 帧抢占原理

帧抢占通过抢占 BE 帧的传输来减少快速帧的排队时间, 可以为单帧的端到端延迟和抖动提供最坏情况下的上限。然而, 随着网络中快速帧数量的增加, 快速帧不能互相抢占, 帧抢占会降级为严格优先级调度。在最坏的情况下, 网络中所有的快速帧都使用相同的出端口, 按照严格的优先级顺序传输。帧抢占的端到端延迟和抖动上限主要依赖于发送方, 假设所有的快速帧都是已知的、周期的, 通过最坏情况分析, 可以计算出每一跳的排队延迟上限。

因此,采用帧抢占调度算法,时延、抖动的确定性强烈依赖于对网络中所有潜在的快速帧的了解。

2.2 基于流的调度

基于流的调度算法采用了 TTE 网络规划的思想,关注每条数据流的传输,它适用于机载网络同步流的传输。为了实现每个流在时间和空间上的隔离,基于流的调度要求发送周期与网络周期对齐,并且需要精确配置每条数据流传输的起始时刻(即 GCL)和路由,以保证最小的排队延迟。由于排队延迟很小,甚至不存在,流的路径延迟通常优于端到端延迟,从而实现了传输的高确定性。

基于流的调度只有在转发高优先级流量时,BE 的传输门才关闭,因此,网络的带宽利用率很高。在机载网络中,基于流的调度规划是一个复杂 HP-Hard 问题,规划难度较大。为了便于应用程序及时地提供数据以及网络在预留的时隙内发送数据,网络设备需要非常高精度的时间同步。时间上的微小偏差可能导致后续所有通信错过时间窗口的级联反应,从而产生网络重新同步的额外时延。为了防止这种情况的发生,需要对所有的流进行严格警管,因此,基于流的调度要求每个流在其路径上的每个交换机上有两个门操作。当发送方发送了过多流量或是在错误的时间发送流量时,占用了其他优先级流量的带宽,交换机通过其入端口门操作和出端口门操作实施基于流的调度策略,保证了数据流的时延。

在交换机的入端口处,设置流过滤器,内建的门操作根据数据帧的特征,如虚链路(VLAN)、MAC 地址、IP 地址等,对转发前的每条数据流进行约束、监管,以防止在交换机出端口处的队列被非法帧淹没。交换机在对流量的警管策略上,按照控制方式可以分为以下两种:

- 1) 对单条流的限流:限制单条流的发送,降低到过滤器的预设门限值;
- 2) 对单条流的阻断:完全阻断错误的或流量异常的某条流。

此外,基于流的调度在交换机的出端口处还设置了基于信用的整形器(CBS),通过门控操作确保交换机在特定的时机将被筛选出的数据转发出去。

综上,基于流的调度非常适合于机载同步流的调度,其缺点是规划复杂度高,且存在实际的可伸缩性问题(GCL 长度)。

2.3 基于类的调度

与基于流的调度相比,基于类的调度是一种保证周期消息的端到端延迟和抖动的 TAS 方法。如图 6 所示,假设发送方按照如下规则发送周期消息:

- 1) 发送周期与网络周期相同;
- 2) 发送周期和网络周期不严格对齐;
- 3) 发送时刻可以偏离发送周期的 ±50%,即一个完整的周期。

在基于类的调度中,所有具有相同时延和抖动要求的数据流被分配到同一个专用的通信类中。由于 TAS 只区分

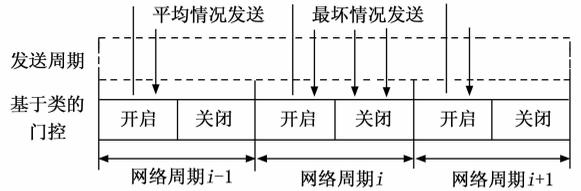


图 6 基于类的调度

出口队列,这个通信类需要映射到一个专用的出口队列上。每个网络周期开放一次传输门,门打开的时间间隔称为类窗口,其长度可以任意配置,但它需要与网络中的交换机匹配。由于传输门只打开一次,类窗口必须足够大,以保证所有流都在一个周期内发送出去。为此,提出下列公式来计算类窗口的大小。

如前所述,所有循环流,端到端延迟不能大于一个网络周期时间:

$$\forall s \in S: d_{e2e}^s \leq t_{cycle} \quad (1)$$

也可以表示为:

$$\max_{s \in S} (d_{e2e}^s) \leq t_{cycle} \quad (2)$$

基于这个约束条件,给出一个基于公式(2)的类窗口大小 t_w 的推导。

该推导适用于基于有向图 $G(v, \epsilon)$ 建模的任意网络。顶点 v 代表交换机和主机,交换机的全双工链路用一个出站边和一个入站边表示,交换机在出端口配置了队列和传输门,参数符号见表 1 所示。

表 1 参考符号

符号	含义
$v_H \subset v$	主机
$v_S \subset v$	交换机
S	流
$P_S(s) \subseteq v_S, s \in S$	在流 S 路径上的交换机
$P_E(s) \subseteq \epsilon, s \in S$	在流 S 路径上的边
$S_e(e) \subseteq S, s \in \epsilon$	出端口的数据流
t_w	类窗口的大小
t_{cycle}	循环时间
d_{f-ind}	帧无关延迟
d_{f-dep}	帧有关延迟

公式(2)的端到端延迟 d_{e2e}^s 由路径延迟 d_{path}^s 和时变排队延迟 d_{queue}^s 组成,即:

$$d_{e2e}^s = d_{path}^s + d_{queue}^s \quad (3)$$

时变排队延迟 d_{queue}^s 由两个部分组成: $d_{queue}^s = d_{gate}^s + d_{interference}^s$ 。 d_{gate}^s 表示传输门关闭后一个帧的阻塞时间, $d_{interference}^s$ 表示传输门打开时,由于等待帧 s 之前的其他帧传输完成所产生的排队延迟。

传输门开启一次的持续时间为 t_w ,如果数据帧刚好在类窗口关闭时刻到达,则必须等待 $d_{gate}^s \leq t_{cycle} - t_w$,据此,并结合公式(3),公式(2)可以改写为:

$$\max_{s \in S} (d_{\text{path}}^s + d_{\text{interference}}^s + t_{\text{cycle}} - t_w) \leq t_{\text{cycle}} \quad (4)$$

t_{cycle} 和 t_w 与循环流 s 无关, 故:

$$t_w \geq \max_{s \in S} (d_{\text{path}}^s + d_{\text{interference}}^s) \quad (5)$$

根据 d_{path}^s 和 $d_{\text{interference}}^s$ 的上限, 可以推导出 t_w 的上限:

$$t_w \geq \max_{s \in S} (d_{\text{path}}^s + d_{\text{interference}}^s) \geq \max_{s \in S} (d_{\text{path}}^s) + \max_{s \in S} (d_{\text{interference}}^s) \quad (6)$$

根据最长路径和最大帧长, 可以确定路径延迟:

$$\max_{s \in S} (d_{\text{path}}^s) = \max_{s \in S} (|P_S(s)| \cdot$$

$$(d_{f-\text{ind}} + d_{\text{prop}} + d_{f-\text{dep}}^s)) + (d_{\text{prop}} + d_{f-\text{dep}}^s) \leq$$

$$\max_{s \in S} (|P_S(s)| + 1) \cdot (d_{f-\text{ind}} + d_{\text{prop}} + d_{f-\text{dep}}^s) \leq$$

$$\max_{s \in S} (|P_S(s)| + 1) \cdot (d_{f-\text{ind}} + d_{\text{prop}} + \max_{s \in S} (d_{f-\text{dep}}^s)) \quad (7)$$

同样, 假设在最坏情况下, 流 s 的某一帧被通信路径上某个出端口传输的其他流阻塞, 得到干扰延迟 $d_{\text{interference}}^s$ 的上限。如图 5 所示, 节点在周期传输门关闭后、周期 $i+1$ 传输门开启前发送数据帧, 最坏情况下, 流 s 的数据帧需要等待两个网络周期, 即:

$$\max_{s \in S} (d_{\text{interference}}^s) \leq 2 \cdot \max_{s \in S} \left(\sum_{e \in P_e(s)} \sum_{s' \in S_e(s)} d_{f-\text{dep}}^{s'} \right) \leq 2 \cdot \max_{s \in S} (|P_E(s)| \cdot \max_{e \in P_e(s)} \sum_{s' \in S_e(s)} d_{f-\text{dep}}^{s'}) \quad (8)$$

将式 (7) 和式 (8) 代入式 (6), 可得:

$$t_w \geq 2 \cdot \max_{s \in S} (|P_E(s)| \cdot \max_{e \in P_e(s)} \sum_{s' \in S_e(s)} d_{f-\text{dep}}^{s'}) +$$

$$\max_{s \in S} (|P_S(s)| + 1) \cdot (d_{f-\text{ind}} + d_{\text{prop}} + \max_{s \in S} (d_{f-\text{dep}}^s)) \quad (9)$$

综上, t_w 的值足够大, 当考虑路径长度和流 s 其他帧的潜在干扰时, 能够保证每个流都可以到达目的节点。式 (9) 提供了一个 t_w 的上限, 如果要 t_w 变得更小, 有以下 3 个方法:

- 1) 计算每跳的最大 $d_{\text{interference}}^s$;
- 2) 限制发送模型所允许的传输抖动;
- 3) 使用相应的流量整形和警管策略来执行更可预测的行为。

基于类的调度保证了一个周期内的端到端延迟和抖动, 通过 t_w 的数学模型, 证明了基于类的调度需要很少的 GCL 条目, 发送方只需要粗时间同步, 由于粗同步精度需要更少的同步时间, 基于类的调度允许在运行时动态地启用和禁用发送端口。然而, 机载安全关键业务不允许发生数据丢弃, 因此, 基于类的调度需要在交换机上有大量高优先级的缓冲区。由于计算 t_w 时考虑了最坏情况, 类窗口很大, 而 t_w 的空闲时间不能被 BE 使用, 导致带宽效率很低。

综上, 基于类的调度提供了端到端延迟和抖动保证、同步开销和网络配置之间的权衡, 但代价是更宽松的边界和更大的缓冲区。

3 仿真分析

为了测试帧抢占和 TAS 算法在机载网络业务流量下的最佳和最坏边界, 使用离散仿真工具 NeSTiNg 开展调度算

法的性能评估, 该工具扩展了 OMNeT++/INET 协议栈的 TSN 功能, 包括帧抢占、TAS 和基于信用的整形^[16]。

3.1 仿真模型

OMNeT++ 可以用于模拟通信网的业务流, 通信协议的模型, 排队网络以及多处理器和分布式系统^[17], NeSTiNg 作为融合 IEEE TSN 网络的模拟器, 可以塑造各种流量整形器的模型, 还可以模拟交换机入端口、出端口的虚链路 (VLAN) 标记和多个分支、循环的控制功能。

如 1.4 节所述, 为了更加真实地反映调度算法在机载背景流量下的性能, 采用交换型拓扑开展仿真验证。端系统和交换机的仿真模型设计具体如下:

1) 端系统模型: 主要由时钟同步模块、带宽预留模块、发送模块和接收模块构成。

(1) 时钟同步模块: 负责为网络模型提供统一的全局时钟, OMNeT++ 提供了 `sim_Time()` 函数, 可以获取当前仿真系统时间值, 其精度符合高精度时间同步协议 (PTP, precise time protocol) 的要求^[18], 主节点的时钟同步模块通过调用该 API 将全局时钟同步至从节点的时钟模块。当端系统模型发送或接收数据流时, 其内部的数据帧处理函数调用 INET 框架提供的 IEEE802 时钟模块接口函数获取当前时间戳, 并将其嵌入至发送数据帧或接收数据帧中。时钟同步模块提供时钟同步周期处理功能, 即当时钟同步周期开始时, 时钟同步模块向全网广播时钟同步数据帧; 当主节点接收到从节点的延迟响应请求时, 检查时间戳信息和数据帧类型, 调用 OMNeT++ 的 `getHardwareDelay()` 函数, 模拟硬件延时, 随后将时间戳嵌入应答数据帧中发送至从节点。此外, 从节点的时间同步模块通过调用 `updateClock()` 函数, 修正本地的时钟偏差。

(2) 带宽预留模块: 负责为发送方和接收方预留充足的通信带宽, 主要包括通信节点注册函数、网络配置更新函数、通信节点故障管理函数等。对网络中发送方模型和接收方模型内的每一条数据流创建唯一标识符 StreamID, 通过定义数据流的发送方标识、接收方标识、数据帧负载长度、数据帧间隔、保护窗口长度等参数, 配置仿真环境的背景流量。例如, 带宽预留模块通过调用 OMNeT++ 内建的 `getCurrentPeriod()`、`getCurrentTicks()` 获取当前时刻并判断该时刻是否在保护窗口内, 通过调用 INET 框架的各类 EVENT 状态和 EtherFrame 类型, 创建或注销特定的数据流。

(3) 发送模块: 负责数据帧的发送控制, 在模块出口队列处构建调度模块, 将不同数据流输出到交换机的入端口上。根据 IEEE802.1 协议规定, 在 NeSTiNg 调度模块中, 每条数据流通过 VLAN 标识映射到深度为 8 的发送队列上, 基于类的调度算法通过 VLAN 标识中的一个 3 位的优先级码点 (PCP, priority code point) 为每一类数据流配置不同的 GCL 长度。此外, 发送模块还具备发送处理接口函数, 负责传输门发送数据帧的逻辑判断。

(4) 接收模块：负责从交换机的各个出端口接收数据流，通过调用各个数据流的数处理模块完成。

2) 交换机模型：主要由交换转发模块和端口调度模块构成。

(1) 交换转发模块：负责数据流在交换机内部的数据处理，主要由 INET 框架的链路组件实现。通过构建 MAC 模块模拟传输和转发延迟，构建转发模块跟踪数据帧从交换机入端口队列到出端口队列的流动，通过中继单元的延迟模块，模拟交换机的处理延迟，最后根据 VLAN 标识将数据帧移交至不同的出端口队列。

(2) 端口调度模块：负责交换机入端口及出端口的调度算法实现。交换机除了具备端系统发送模块和接收模块的功能外，还需构建基于流的检索模块，通过 PCP 值，传输门选择适合传输的下一条数据流。控制路径分为从入端口向下到出端口，以及从出端口向上转发。在向下控制路径空闲之后，传输门才能触发对新到达的数据流的处理；在向上控制路径空闲之后，传输门根据 GCL 依次轮询，排队等待下一条符合条件的数据流。

3.2 仿真过程

使用网络规模 $n_i \in (10, 20, 30, 50)$ 为的网络拓扑，链路带宽均为 1 Gbps。遍历 n_i 值，随机为网络分配 n_i 个同步流、周期流和“尽力而为”流传输业务，以模拟不同的机载业务。具体如下：

1) 同步流参考了文献[19]中 TT 业务帧的数据负载、发送周期、门控开关的保护窗口等参数，即配置同步流的发送周期分别为 $2\ 000\ \mu\text{s}$ 、 $8\ 000\ \mu\text{s}$ 、 $16\ 000\ \mu\text{s}$ 、 $32\ 000\ \mu\text{s}$ ，配置同步流的帧间隔分别为 $2\ 000\ \mu\text{s}$ 、 $4\ 000\ \mu\text{s}$ 、 $8\ 000\ \mu\text{s}$ 、 $16\ 000\ \mu\text{s}$ ，消息负载可变且无分组转发，优先级最高，不同任务流量的优先级可变，具有一定随机性；

2) 周期流参考了文献[20]中机电系统采集、状态判断、控制、告警发送、数据处理、异常处理、故障记录等任务的特征，即配置周期流的周期分别为 $100\ \text{ms}$ 、 $200\ \text{ms}$ 、 $400\ \text{ms}$ ，消息优先级中等，且不同任务流量的优先级不同，任务周期越小，优先级越高；

3) “尽力而为”流参考了文献[21]中机载视频影像的业务特点，配置“尽力而为”流的数据负载为需要分组发送的长消息，消息优先级最低，不同任务流量的优先级相同。

随后，随机选择一个发送方及一个接收方，在基于流的调度仿真中，每个节点只模拟单个流的源节点以及多个流的目的节点，流的负载在间隔 $[64\ \text{B}, 300\ \text{B}]$ 中随机抽取。帧抢占仿真时，发送方以线路速率向接收方发送可抢占的 BE 帧。

3.3 仿真结果分析

图 7 列出了 3 种调度算法的累计排队延迟，即单个帧在从发送方到接收方的路径上等待队列所花费的总时间。

在图 7 (a) 中，可以看到，在机载网络业务流量下，帧抢占产生的累计排队延迟的平均值是微秒级的。然而，帧抢占最坏情况下的延迟超过 $300\ \mu\text{s}$ ，这个数量级比平均

累积排队延迟大很多，也就是说，即使只有一个业务类别使用快速帧，由于排队延迟的变化，端到端延迟也会有差异。此外，由于流量分配的随机性， n_i 值的增大并不一定会使排队延迟变小。统计结果表明，对于关键等级高的实时业务，采用帧抢占并不能保证确定性，需要提前对整个机载网络业务进行充分的规划和编排。

在图 7 (b) 中，可以看到，在机载网络业务流量下，基于流的调度可以可靠地实现接近零排队，最大累计排队时延为 $5.1\ \mu\text{s}$ 。然而，高确定性带来的问题是，需要使用强大的网络配置工具，对每条机载业务数据流进行规划，解决 NP-Hard 问题。

基于类的调度的仿真结果如图 7 (c) 所示。 n_i 的值越大，累计排队延迟就越小。主要有以下两个原因：

首先，由于 t_{cycle} 是恒定的 $5\ \text{ms}$ ，随着机载网络规模和业务数量的增加， n_i 越大，类窗口 t_w 就越大，即当传输门关闭时，数据到达接收方所造成的排队延迟就越小；

其次， n_i 增大， d_{path}^s 增大，即 t_w 的值变大，使得在传输门关闭期间通信的概率成比例地降低。

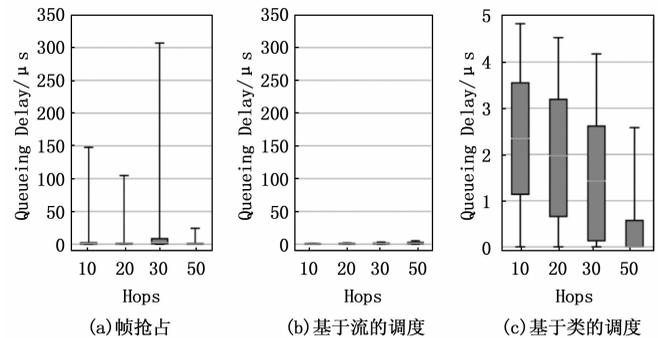


图 7 仿真结果分析

4 结束语

文章研究了 TSN 的 3 种调度算法，即帧抢占、基于流的调度和基于类的调度，分析了这些算法的工作原理，建立了数学模型，开展了机载应用场景的仿真。结果表明，帧抢占不需要网络同步，但其确定性强依赖于对机载网络潜在有冲突的快速帧的规避；基于流的调度适用于机载强实时业务，但规划较为复杂，且存在 GCL 长度伸缩的问题；基于类的调度适用于周期业务，它很容易进行配置规划，但由于类窗口较大，带宽效率较低。总之，每种方法都可以服务于特定的、对网络存在不同要求的机载应用，单一的调度算法不能服务于所有的机载业务，因此需要将这些算法组合起来。综上所述，后续需评估网络调度算法之间的相互作用，为进一步的研究提供充分的机会。

参考文献：

- [1] MALCOLM N, ZHAO W. Hard real-time communication in multiple access networks [J]. Journal of Real-Time Systems, 1995, 8 (1): 35 - 77.
- [2] KOPETZ H, ADEMAJ A, GRILLINGER P, et al. The time-

- triggered ethernet design [C] // ISORC 2005. Eighth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing, 2005. IEEE, 2005; 22-33.
- [3] 高会生, 赵书建. 时间触发通信流控技术研究综述 [J]. 电力信息与通信技术, 2021, 19 (12): 16-26.
- [4] 芦 伟, 朱纪洪, 杨佳利, 等. 时间触发以太网确定性调度设计与研究 [J]. 计算机测量与控制, 2019, 27 (3): 264-270.
- [5] 曾 磊. TSN 帧抢占及循环队列调度研究 [D]. 西安: 西安电子科技大学, 2020.
- [6] IMTIAZ J, JASPERNEITE J, HAN L. A performance study of Ethernet audio video bridging (AVB) for industrial real-time communication [C] // IEEE Conference on Emerging Technologies & Factory Automation, IEEE, 2009; 1-8.
- [7] 李 珂. 时间敏感网络交换调度机制研究与实现 [D]. 西安: 西安电子科技大学, 2019.
- [8] 詹双平, 冯景斌, 崔春来, 等. 大规模确定性网络调度技术综述 [J]. 鹏城实验室通讯, 2020, 2: 7-13.
- [9] THANGAMUTHU S, CONCERN, CUIJPERS P J L, et al. Analysis of ethernet-switch traffic shapers for in-vehicle networking applications [C] // Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition. EDA Consortium, 2015; 55-60.
- [10] ZHAO L, POP P, CRACIUNAS S S. Worst-case latency analysis for IEEE 802.1qbv time sensitive networks using network calculus [J]. IEEE Access, 2018, 6: 1.
- [11] 万 浩. 边缘网关节点中 OPC UA-TSN 融合技术研究 [D]. 武汉: 华中科技大学, 2019.
- [12] NAYAK N G, DURR F, ROTHERMEL K. Routing Algorithms for IEEE802.1Qbv Networks [J]. SIGBED Rev., 2018, 15 (3): 13-18.
- [13] FALK J, DURR F, ROTHERMEL K. Exploring practical limitations of joint routing and scheduling for TSN with ILP [C] // 2018 IEEE 24th Int. Conf. on Embedded and Real-Time Computing Systems and Applications, Hakodate, Japan, 2018.
- [14] 王朴野. 时间敏感网络中调度算法的研究与仿真 [D]. 北京: 北京邮电大学, 2018.
- [15] 丛培壮, 田野, 龚向阳, 等. 时间敏感网络的关键协议及应用场景综述 [J]. 电信科学, 2019, 10: 31-42.
- [16] FALK J, HELLMANN S D, CARABELLI B, et al. NeST-ing: simulating IEEE time-sensitive networking (TSN) in OMNeT++ [C] // 2019 International Conf. on Networked Systems, Munich, Germany, March, 2019.
- [17] 唐 敏. 基于 OMNeT++ 的 INET 框架消息传递研究 [J]. 电脑与信息技术, 2011, 19 (1): 31-33.
- [18] PRYTZ G. A performance analysis of EtherCAT and PROFIBUS IRT [C] // Emerging Technologies and Factory Automation, 2008. ETFA 2008. IEEE International Conference on IEEE, 2008; 408-415.
- [19] 屈 静. 航电机载网络的业务调度与性能评价技术研究 [D]. 西安: 西安电子科技大学, 2020.
- [20] 陈 福, 张晓红, 赵 刚, 等. 飞机机电综合管理平台应用任务调度方法研究 [J]. 计算机测量与控制, 2016, 24 (3): 255-262.
- [21] 郝 朝, 王 灏. 基于 RTP 的机载网络视频实施监控系统设计 [J]. 计算机技术与发展, 2019, 29 (5): 131-134.
- [5] MEHMOOD A, LIAQUAT M, BHATTIAI, et al. Trajectory planning and control for lane-change of autonomous vehicle [C] // International Conference on Control, Beijing: Automation and Robotics, 2019; 331-335.
- [6] JULIA N, ERIK C, MATTIAS B, et al. Lane change maneuvers for automated vehicles [J]. IEEE Transactions on Intelligent Transportation Systems, 2017, 18 (5): 1087-1096.
- [7] 杨 也, 倪建军, 陈一楠, 等. 改进 RRT* 的室内机器人路径规划算法 [J]. 计算机测量与控制, 2020, 28 (1): 241-245.
- [8] YAO P, WANG H L, SU Z K. Cooperative path planning with applications to target tracking and obstacle avoidance for multi-UAVs [J]. Aerospace Science and Technology, 2016, 54: 10-22.
- [9] 张家旭, 王 晨, 赵 健. 基于改进人工势场法的汽车弯道超车路径规划与跟踪控制 [J]. 汽车工程, 2021, 43 (4): 546-552.
- [10] 冀 杰, 姬 鹏, 彭 和, 等. 面向车辆主动避障的三维虚拟危险势能场设计 [J]. 汽车工程, 2016, 38 (9): 1065-1071, 1079.
- [11] YUAN C C, WENG S F, SHEN J, et al. Research on active collision avoidance algorithm for intelligent vehicle based on improved artificial potential field model [J]. International Journal of Advanced Robotic Systems, 2020, 17 (3): 1-10.
- [12] LIU Z X, YUAN X F, HUANG G M, et al. Two potential fields fused adaptive path planning system for autonomous vehicle under different velocities [J]. ISA Transactions, 2021, 112 (6): 176-185.
- [13] 安林芳, 陈 涛, 成艾国, 等. 基于人工势场算法的智能车辆路径规划仿真 [J]. 汽车工程, 2017, 39 (12): 1451-1456.
- [14] 杨 瑶, 付克昌, 蒋 涛, 等. 改进 A* 算法的智能车路径规划研究 [J]. 计算机测量与控制, 2020, 28 (10): 170-176.
- [15] 龚建伟, 姜 岩, 徐 威. 无人驾驶车辆模型预测控制 [M]. 北京: 北京理工大学出版社, 2014.
- [16] 李亚勇, 蔡英凤, 陈 龙. 考虑前后方车辆行驶状态的 ACC 系统控制方法 [J]. 汽车工程, 2019, 41 (8): 865-871.
- [17] 汽车百科全书编纂委员会. 汽车百科全书 [M]. 北京: 中国大百科全书出版社, 2010.
- [18] 李 卓, 姚立泽, 王丽霞, 等. 适用于复杂动态环境的智能车运动规划方法 [J]. 机械设计与制造, 2020 (12): 259-264.