

# TSDSIR: 机器人系统中多业务并发的消息数据流调度

丁男<sup>1,2</sup>, 冀承慧<sup>1</sup>, 胡创业<sup>1</sup>, 许力<sup>3</sup>

(1. 新疆师范大学计算机科学技术学院, 新疆 乌鲁木齐 830054;

2. 大连理工大学工业装备智能控制与优化教育部重点实验室, 辽宁 大连 116024; 3. 字节跳动, 北京 100086)

**摘要:** 针对机器人云-边-端协同场景中所面临的多业务并发、多模态消息数据混杂等特点, 基于机器人操作系统 2 (ROS2) 中消息分发机理分析, 提出了面向时间敏感属性的多消息动态调度方法。首先, 根据 ROS2 中多话题消息流并发属性和网络实时状态, 定义了消息数据优先级、最大传输时间和剩余传输时间等流调度参数, 进而对待调度的流队列状态进行实时表征。其次, 设计了包含时间优先和优先级优先的混杂切换调度模型, 并结合流调度参数给出调度切换规则及理论边界条件。最后, 提出了 ROS2 中面向时间敏感属性的多消息动态调度方法, 并在其数据分发服务层中集成应用。仿真实验与实际应用验证表明, 所提方法能提升 ROS2 中多消息流并发传输效率, 同时保障高优先级的、具有时间敏感属性的消息数据实时可靠传输。

**关键词:** 机器人操作系统 2; 数据分发服务; 时间敏感属性; 混杂切换; 多消息流调度

**中图分类号:** TP31

**文献标志码:** A

**DOI:** 10.11959/j.issn.1000-436x.2024133

## TSDSIR: multi-service concurrent message data flow scheduling in robotic systems

DING Nan<sup>1,2</sup>, JI Chenghui<sup>1</sup>, HU Chuangye<sup>1</sup>, XU Li<sup>3</sup>

1. School of Computer Science and Technology, Xinjiang Normal University, Urumqi 830054, China

2. Key Laboratory of Intelligent Control and Optimization for Industrial Equipment, Dalian University of Technology, Dalian 116024, China

3. ByteDance, Beijing 100086, China

**Abstract:** Aiming at the characteristics of multi-service concurrency and multi-modal message data mixing faced in the robot cloud-edge-end collaboration scenarios, a multi-message dynamic scheduling method oriented to time-sensitive attributes was proposed based on the analysis of the message distribution mechanism in the robot operating system 2 (ROS2). Firstly, scheduling parameters, including message priority and transmission times, were defined based on multi-topic concurrency and real-time network state in ROS2, enabling real-time queue state characterization. Then, a hybrid time/priority scheduling model was designed with specific switching rules and boundary conditions. Finally, the dynamic scheduling method, considering time sensitivity, was integrated into ROS2's data distribution service layer. Simulation experiments and practical applications verify that the proposed method can improve the efficiency of concurrent transmission of multiple message streams in ROS2, and at the same time guarantee the real-time reliable transmission of high-priority message data with time-sensitive attributes.

**Keywords:** ROS2, data distribution service, time-sensitive attribute, hybrid switching, multi-message flow scheduling

收稿日期: 2024-01-10; 修回日期: 2024-06-14

通信作者: 丁男, dingnan@dlut.edu.cn

基金项目: 国家自然科学基金资助项目(No.62072071, No.62262066);“十四五”国家重点研发计划基金资助项目(No.2022YFB4500800);新疆维吾尔自治区自然科学基金资助项目(No.2021D01E20)

**Foundation Items:** The National Natural Science Foundation of China (No.62072071, No.62262066), The National Key Research and Development Program of the “14th Five-Year” Plan (No.2022YFB4500800), The Natural Science Foundation of Xinjiang Uygur Autonomous Region (No.2021D01E20)

### 0 引言

ROS2 作为 ROS 的第二代，底层架构发生了较大变化，架构如图 1 所示。ROS2 的底层通信引入了数据分发服务 (DDS, data distribution service) 层，使得机器人应用中的实时性、可靠性和连续性有了一定程度的增强<sup>[1]</sup>。ROS2 中各节点之间基于 DDS 的消息传递有助于为其在多机器人协作系统应用场景中实现复杂应用和功能管理，并提供基础和保障。然而，随着人工智能和 5G 技术的发展，机器人进入 4.0 时代，结合云-边-端协同机制建立多机器人协同，已成为机器人应用领域的主要发展趋势<sup>[2]</sup>。机器人之间、机器人与云-边服务器之间大量复杂信息的实时交互呈现出多源并发、多模态混合等特点，这也为 ROS2 在多机器人协同领域的应用提出了新的挑战。

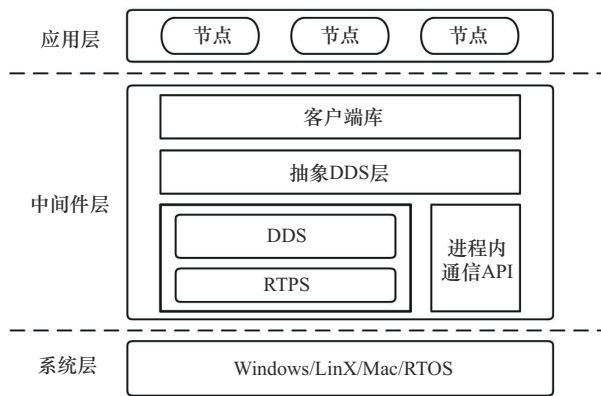


图 1 ROS2 架构

针对上述云-边-端融合的机器人应用规模和工作范围的复杂性，机器人操作系统中面临着多源数据并发、多模态数据共存等信息交互需求。目前，

ROS2 借助 DDS 通过一系列服务质量 (QoS, quality of service) 策略来优化不同数据的网络传输<sup>[3]</sup>，ROS2 中的 QoS 策略如表 1 所示。尽管多智能体云机器人操作系统可以通过增加容错机制、引入冗余机制、接入高性能网络等措施来应对这些挑战，但同时也带来了附加的资源消耗和系统复杂性增加的问题。此外，随着触感控制、时间敏感网络 (TSN, time-sensitive networking) 协议等在机器人领域的应用不断深入，在保障高优先级数据调度的同时，也增强了对数据单跳时间约束的要求。因此，面向 DDS 中对消息数据的多 QoS 策略联合调度问题亟待研究。

相对于传统网络协议数据报文调度，目前云-边-端融合的机器人在协同构成中，其信息主要包含连续动态和离散动态的混杂特性。将包含连续动态和离散动态的混杂切换理论应用到消息数据调度中可以有效解决离散关键数据优先和时间紧迫数据优先混杂传输的问题<sup>[4]</sup>，该方法广泛应用于分布式及工业物联网领域，但在机器人操作系统及其网络通信中应用较少。时间敏感网络支持对不同类型的流量进行分类和调度<sup>[5-6]</sup>，确保满足关键应用程序的带宽要求以及低时延的实时通信，但其实现和部署需要一定的技术支持和成本支持，特别是在机器人应用领域，需要考虑现有系统的兼容性和稳定性等因素。

Fast DDS 作为 ROS2 长期支持版本中默认通信中间件的实现，不仅遵循 OMG DDS 1.4<sup>[7]</sup> 和 OMG RTPS 2.2<sup>[8]</sup> 规范，而且在 Fast DDS 的实时发布订阅协议 (RTPS, real-time publish-subscribe protocol) 层上提供流控制器 (FlowController) 机制，用以限制和调度不同数据写入者发送消息数据的速率和顺序，该机制可以在特定的 QoS 策略中与创建发布者

表 1 ROS2 中的 QoS 策略

QoS 策略	值和含义
History	Keep all: 存储所有样本, 受底层中间件配置的资源限制; Keep last: 最多只能存储 N 个样本, 可通过队列“Depth”选项进行配置
Depth	Queue size: 仅在“History”策略设置为“Keep last”时有效
Reliability	Reliable: 保证样本送达, 可重试多次; Best effort: 尝试交付样本, 但如果网络不稳定, 可能会丢失样本
Durability	Volatile: 不尝试持久保存样本; Transient local: 发布者负责为后续加入订阅的节点持久保存样本
Deadline	Duration: 相邻话题消息发布之间的预期最长间隔时间
Lifespan	Duration: 从发布消息到接收消息之间不被视为超时或过期的最长时间(超时消息丢弃)
Lease Duration	Duration: 发布者在系统认为其失去活跃性之前必须指示其活动的最长时间(失去活跃性可能是故障的指示)
Liveliness	Automatic: 当任何一个发布者发布消息时, 系统将认为该节点的所有发布者仍然活动, 活跃状态将延长一个“Lease Duration”; Manual by topic: 如果发布者通过调用发布者 API 手动声明自己仍然活动, 系统将认为该发布者仍然活动, 活跃状态将延长一个“Lease Duration”

进行调优。文献[9]探究了在物联网多样化流量模式下, DDS 与其它订阅/发布通信中间件在时延与吞吐量方面的表现差异, 并着重考察了流控制器机制对 DDS 性能的优化潜力, 通过精细地调整流控制器参数和相关 QoS 策略, 能在特定数据流场景下显著提升 DDS 的通信效率。然而, 确定最适配的配置策略需要依赖大量实验数据分析。文献[10]在确认流控制器配置对 DDS 性能有显著影响的基础上, 依据系统配置与实际观测到的性能指标, 设计了一种算法来智能化调整各个发布者的发送速率, 以在维持系统通信高可靠性的同时, 一定程度上提升了单个话题 (Topic) 消息的吞吐量并改善其可靠性。因此, Fast DDS 在 RTPS 层提供的流控制器机制为在机器人网络通信中应用基于包含连续动态和离散动态的混杂切换的多消息动态调度方法提供了可行性, 但目前这方面相关的研究与解决方案较少。

基于混杂切换的相关理论<sup>[11]</sup>, 针对机器人网络中混杂消息传输实时性和可靠性问题, 结合 TSN 和 ROS2 消息传输特点, 对 DDS 中不同话题消息流的数据调度进行相关研究, 并提出 ROS2-DDS 中面向时间敏感属性的多消息动态调度方法 (TSDSIR, dynamic scheduling method for multi-message communication in ROS2-DDS that considers time-sensitive attributes)。其中, 本文参考 TSN 的服务等级

(CoS, class of service) 机制和具体应用场景下不同类型消息的需求对不同话题消息流的数据划分不同的优先级。首先, 对在兼容 TSN 的机器人网络中, 通过设置相关 QoS 策略在 DDS 中将话题消息流和 TSN 流量建立映射, 使用 TSN 技术进行流量分类和调度。而在不兼容 TSN 的机器人网络中, 将 ROS2-DDS 多类型消息动态调度设计为包含优先权优先子系统和时间优先子系统的混杂切换调度模型, 根据子系统的调度切换规则和基于混杂切换调度模型的调度算法对通信中间件层中不同话题消息流的数据进行调度优化, 调度架构设计如图 2 所示。其次, 通过实验将 TSDSIR 和现有常见调度算法在数据的丢包率和平均时延等方面进行比较和分析, 验证了在同时考虑优先权优先和时间优先的情况下 TSDSIR 的有效性和合理性。最后, 将 TSDSIR 的实现集成到 ROS2 默认通信中间件中——“FastDDSWithTSDSIR”, 并将相关代码在 GitHub 上开源。另外, 结合远程视觉同时定位与地图构建 (SLAM, simultaneous localization and mapping) 方法应用场景, 将上述工作在搭载 Raspberry Pi 4B 主控的 ROS2 智能轮式机器人上进行应用集成与测试。

本文主要的研究工作如下。

1) 对机器人网络中不同话题消息流的数据划分不同优先级, 在兼容 TSN 的机器人网络中, 通

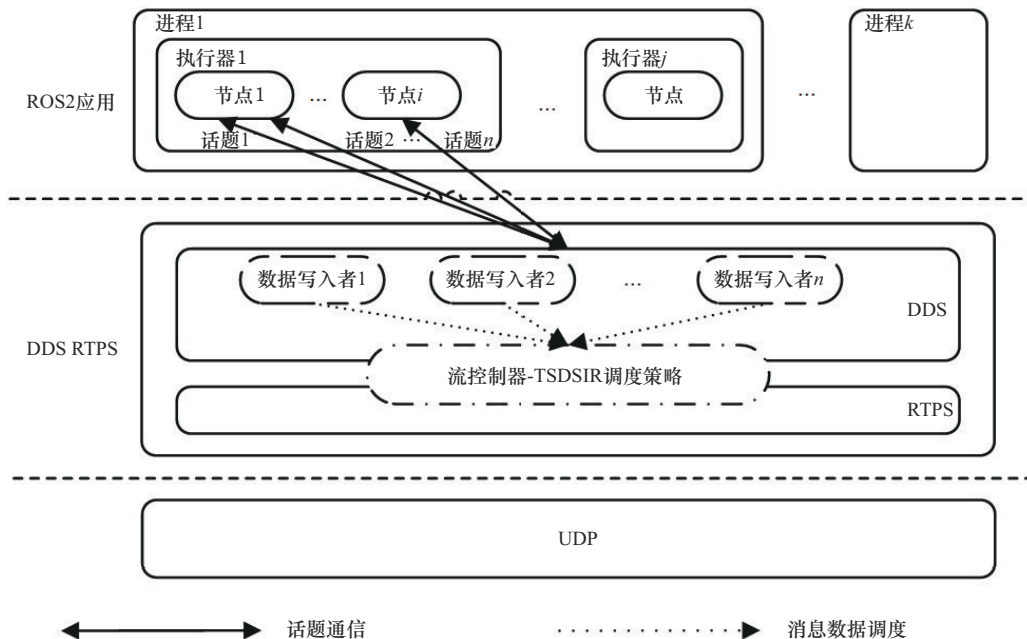


图2 消息数据调度架构

过设置相关 QoS 策略在 DDS 中将话题消息流和 TSN 流量建立映射, 从而使用 TSN 技术进行流量分类和调度。而在不兼容 TSN 的机器人网络中, 应用 TSDSIR, 并对单业务流时延进行特性分析, 给出其理论时延上界为调度提供依据。

2) 基于混杂切换理论和 ROS2 消息流特点, 将选取消息流中哪个 ROS2 话题的数据应优先传输的过程设计为包含 2 个子系统的混杂切换调度模型, 并定义了表征流队列数据即时状态的核心调度参数, 为该模型的有效调度提供参数配置与理论依据。

3) 在混杂切换调度模型的基础上, 结合流调度参数给出了调度切换规则及理论边界条件, 设计了 TSDSIR, 既考量各消息流数据的重要程度与优先排序, 也兼顾各消息流传输的紧迫程度, 并将该方法在 ROS2 默认通信中间件 Fast DDS 中实现与集成。

## 1 相关工作

云-边-端融合的机器人操作系统应用和多机器人协同离不开网络通信, 优化机器人网络通信是提高机器人应用效果的重要一环, 诸多针对机器人网络优化和 ROS2 通信网络优化的研究被提出, 同时一些应用于传统网络、工业物联网、车联网等网络通信优化的方法也可借鉴到机器人网络优化中。

在车联网领域, 面对传统 TCP/IP 协议在大规模网络扩展性、高密度环境效率低下、高移动性场景下寻址不可靠等局限, 新兴的命名数据网络技术是一个有效的解决方案, 通过其内置的内容存储机制, 可以有效缓存传输中的数据, 降低信息冗余, 增强数据多样性, 从而显著提升了机器人网络效能。文献[12]深化了对车载命名数据网络中缓存策略重要性的理解, 特别是在缓存选择与替换机制方面, 这些缓存策略对于利用网络内部的车载缓存资源克服数据传输效率低下和优化资源利用至关重要。本文所关注的待发送队列与上述的数据缓冲队列有相似之处, 但本文侧重于选择更紧迫的数据或更关键的数据优先发送, 提高数据的实时性和可靠性。文献[13]和文献[14]针对车联网中随时时延、局部极大值和数据拥塞问题以及物联网中众多传感器构成的大规模自由网络等问题分别提出了分布式路由协议和三维拓扑部

署方案, 以此优化更低时延的路由路径和网络拓扑, 其中提到的指导路径选择的道路权重评估算法是根据链接时的实时链路特性和断开时的历史交通信息构造时延模型来分配路段权重, 而本文的时延评估则同时考虑待发送队列中各消息数据的等待时间和优先权重。

针对工业物联网中工业应用对低时延和高可靠性的要求以及工业物联网异构网络融合问题, 文献[4]基于混合系统理论, 提出了一种新的 IP 协议时间敏感网络资源调度算法, 考虑单个包的优先级、传输时间和最大传输时延, 通过模拟实验验证, 该算法成功确保了高优先级数据包的及时传输, 同时有效调控了待发送队列的平均处理时间, 使之维持在既定的时间阈值内, 并显著降低了数据丢包的比例。从上述文献中得到启发, 本文将混杂切换理论应用到机器人网络中, 将选取 ROS2 中哪个话题消息流的数据优先发送的过程设计为包含 2 个子系统的混杂切换调度模型。文献[15]通过将 ROS2 及 DDS 中间件在应用层级设定的 QoS 策略需求转化至基于 Wi-Fi 的无线时间敏感网络链路层传输机制, 展示了即使在共享信道存在背景流量的情况下, 也可以通过配置 Wi-Fi 实现时间同步感知调度以满足机器人网络应用的具体 QoS 要求。因此, 本文首先参考 TSN 的服务等级机制和具体机器人应用对消息的特定需求对不同的话题消息流的数据划分不同的优先级, 若所处的机器人网络兼容 TSN 技术, 则可通过设置相关 QoS 策略在 DDS 中将话题消息流和 TSN 流量建立映射, 从而使用 TSN 技术进行流量分类和调度。

## 2 流控制器及调度参数描述

完全开源的 DDS 通信中间件实现——Fast DDS 具有高性能、实时性和可扩展性等特征, 并且在 RTPS 层提供了一种限制和调度消息数据发送速率和顺序的流控制器机制, 此机制通过一个独立的异步发送线程来运作, 统一管理所有采用相同流控制策略的数据发布者所发出的消息<sup>[16]</sup>, 如图 3 所示。流控制器机制定义了基于高优先级的消息流控制方法, 借助 `fastdds.sfc.priority` 属性, 系统允许为每一个数据发布者设定优先级, 这一数值范围被界定在 -10 到 10, -10 表示最高优先级顶级优先, 而 10 则表示最低优先级。因此, 参考 Fast DDS 中基于高

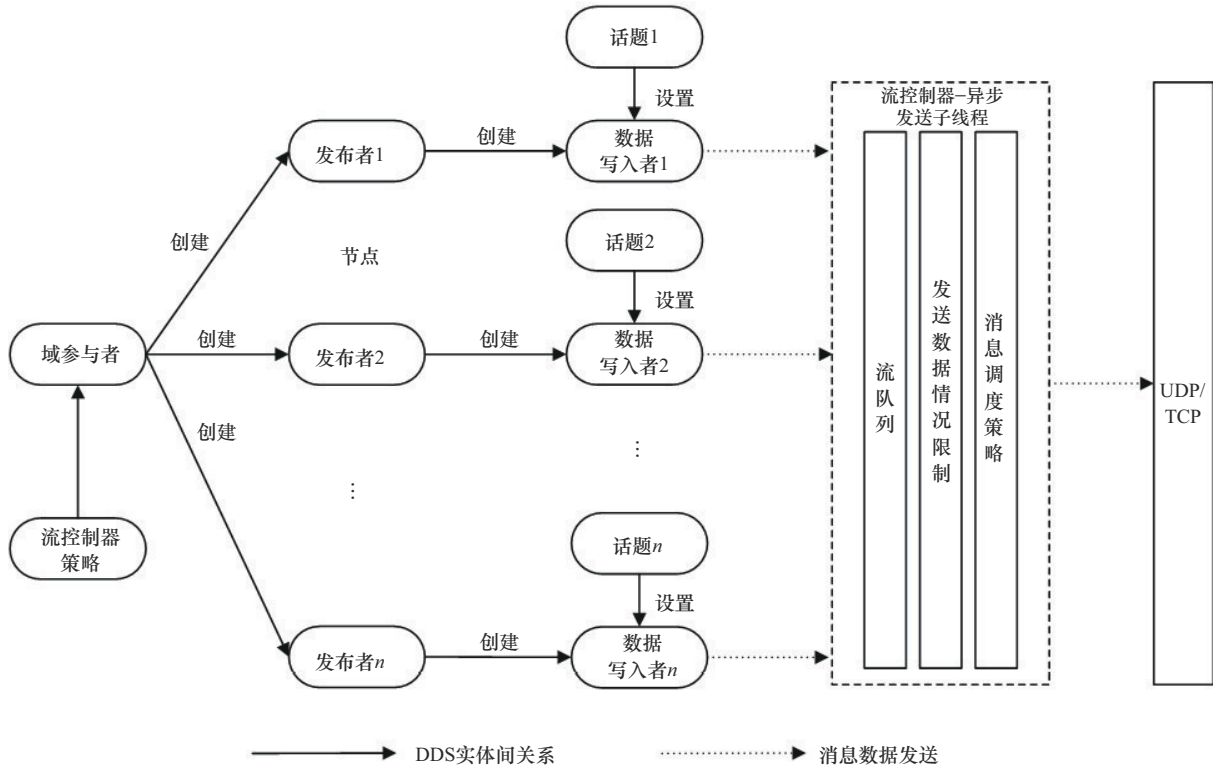


图3 Fast DDS的流控制器机制

优先级的消息流控制方法，将消息数据的优先级与负责发送消息数据的数据写入者的优先级相关联，定义消息数据优先权，来表示不同的话题消息数据在调度过程中的优先权重。

时间敏感网络技术标准中定义了服务等级，用3位二进制定义了0（最低）~7（最高）8个服务等级，传输类型分别对应基础、最大努力、卓越努力、严苛应用、时延和抖动小于100 ms的视频、时延和抖动小于10 ms的音频、内部网络控制和网络控制，并对网络场景进行不同的匹配是时间敏感网络中不同数据流调度的重要因素。同时，在不同应用场景下，待发送的传感器消息数据实时性和可靠性需求可能也是不同的，例如，机器人摄像头采集的图像帧信息在远程视觉SLAM应用中的实时性和可靠性需求比远程视频监控应用中的高。因此，综合时间敏感网络的服务等级机制和具体机器人网络应用的实时性和可靠性分析，将ROS2中不同话题通信传输的消息数据及其数据写入者划分优先级，并在负责封装传输消息的CacheChange\_t类中添加优先级属性，RTPS消息数据与流队列相关类如图4所示。

**定义1** 消息数据优先权，表示不同话题消息

在流队列调度过程中的优先权重。

$$p = \frac{10 - MP}{MP_{sum}} \quad (1)$$

其中， $MP \in \{x | -10 \leq x \leq 10, x \in \mathbb{Z}\}$ ，表示流队列中消息数据的优先级，可从封装消息数据的实体类CacheChange\_t的priority属性字段中获取，数值越小优先级越高。 $MP_{sum}$ 表示优先级的种类，根据上述优先级划分依据， $MP_{sum} = 21$ 。由定义1可知， $10 - MP$ 的范围是0~20， $MP_{sum}$ 值为21，则 $(10 - MP) < MP_{sum}$ ，所以消息数据的优先权重 $p \in [0, 1)$ ， $p$ 越大代表消息数据的优先权重越大。

针对消息数据在传输过程中的传输时间，结合理论传输约束和实际传输时延，本文给出了单消息数据的最大传输时间的定义与描述。

**定义2** 最大传输时间，表示在当前机器人网络环境下，优先级标记为MP的消息数据允许的最大时延。

$$d_{max}^{MP} = \min \{ d_T^{MP}, d_{lmax}^{MP} \} \quad (2)$$

其中， $d_T^{MP}$ 表示当前机器人网络中优先级为MP的周期性消息数据的最小发布频率或最短消息采样周期。如果机器人网络中存在多个机器人应用发布同

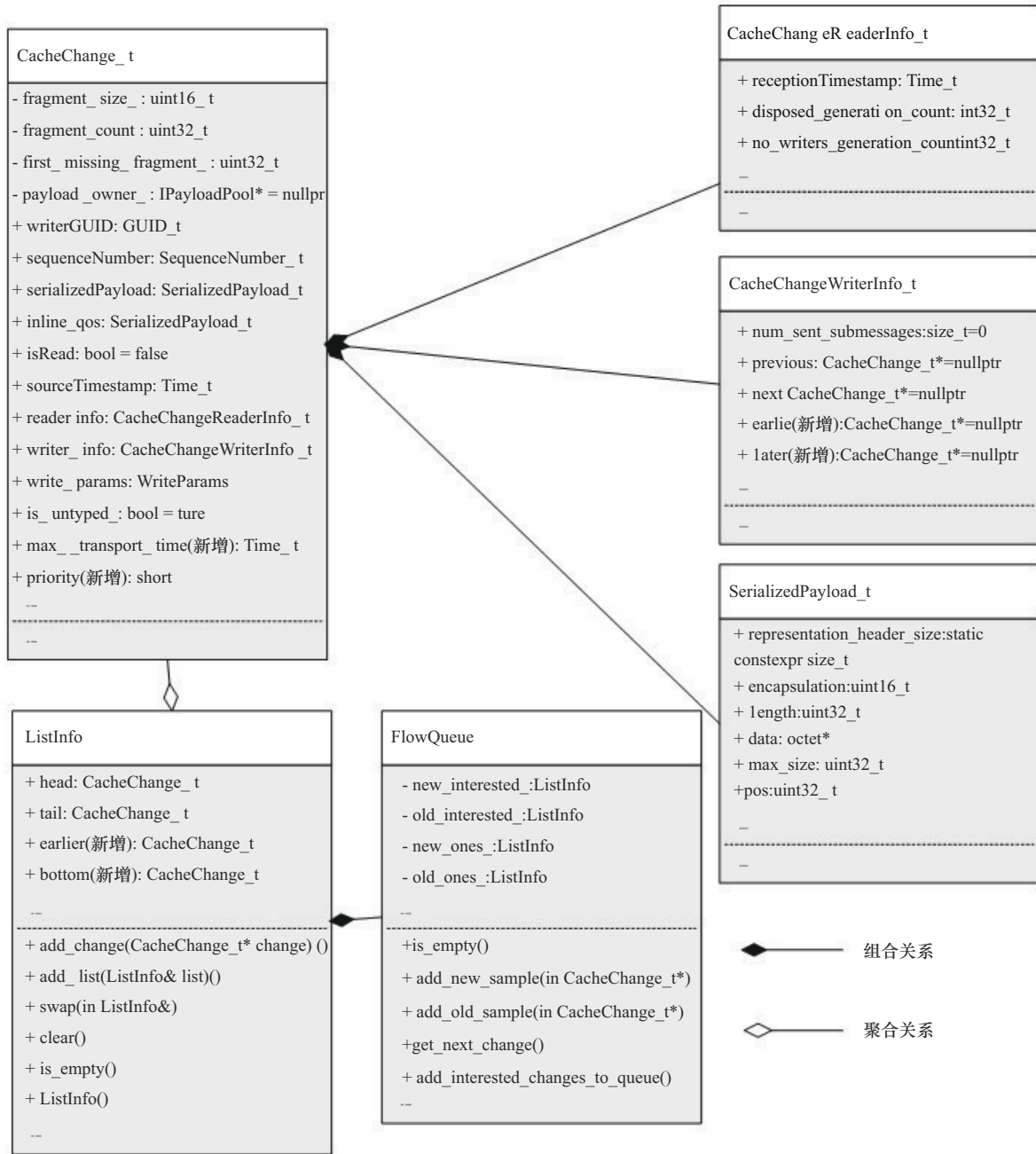


图4 RTPS消息数据与流队列相关类

优先级周期性消息且它们的发布频率不一致，则  $d_T^{MP}$  应选取这些频率中的最小值。 $d_{\max}^{MP}$  表示对于优先级为MP的随机非周期性消息数据所要求的最短传输时间或容忍的最大时延。

为了进一步分析各类型消息数据的传输状态，特别是考虑时间约束条件，针对各消息数据在调度时结合其等待状态，本文描述并明确了消息流队列的平均已等待时间参数指标，旨在深入解析各类型消息数据的调度时延特点。

**定义3** 平均已等待时间，表示当前待传输队列中消息数据的时延情况。

$$\bar{t} = \frac{\sum_{i=1}^n (1+p)t_w}{n} \quad (3)$$

其中，单个消息数据的已等待时间  $t_w$  为

$$t_w = t_{\text{now}} - t_{\text{in}} \quad (4)$$

其中， $t_{\text{now}}$  表示当前系统的时间， $t_{\text{in}}$  表示消息数据进入队列的时间或数据生成的时间。 $n$  作为队列中待发送消息的数量，其值始终不超过队列的总容量

$N$ 。由定义 3 可知, 在计算  $\bar{t}$  时, 优先级高的消息相较于优先级低的消息, 其等待时间的比重更大。此外,  $\bar{t}$  越大, 含有越多接近最大传输时间的消息, 表明当前队列中的消息急于被发送, 增强了消息发送的紧迫性。

考虑到消息数据的传输时间约束, 本文给出了剩余传输时间的定义与描述。

**定义 4** 剩余传输时间, 表示在确保单条消息不会因超时而丢失, 同时保证接收端应用程序的即时性不受影响的前提下, 该消息还能在机器人网络中继续传输的最长时间。

$$t_r = d_{\max}^{\text{MP}} - t_w \quad (5)$$

其中,  $d_{\max}^{\text{MP}}$  表示优先级为 MP 的消息数据的最大传输时间,  $t_w$  表示该消息目前已在队列中等待的时间。

### 3 切换模型和调度算法

#### 3.1 混杂切换模型

混杂系统是一类既包含连续动态又包含离散动态的系统, 其特点是系统的行为包含了连续的时间演化和离散的事件触发。其中切换系统是一种特殊的混杂切换模型, 它由多个子系统组成, 每个子系统都包含一个连续动态模型和一个离散动态模型。切换系统的行为是由这些子系统之间的切换规则控制的, 当切换系统的状态满足某些条件时, 会自动切换到另一个子系统<sup>[17]</sup>。混杂切换模型简单明了且能力和适用范围非常广泛, 可以应用于控制系统、机器人系统、通信系统等领域, 具有很好的实用价值。

针对机器人网络中多样化消息调度需求, 综合考虑优先级与时延约束, 本文构建了一个融合了优先级与时间敏感属性的混杂切换调度模型, 如图 5 所示, 该模型包含时间优先子系统和优先权优先子系统。时间优先子系统着重考虑时间紧迫性, 队列中的消息依据剩余传输时间  $t_r$  升序排列, 旨在快速传输即将达到容忍极限的消息。当多条消息的剩余传输时间  $t_r$  相同时, 系统则转而将参考消息数据优先权  $p$  进行降序排列, 确保即使在时间窗口紧张的情况下, 也能维持消息的重要程度排序。而优先权优先子系统侧重于消息数据优先权, 首先按消息数据优先权  $p$  进行降序排列, 确保高优先权消息优先处理。在消息数据优先权  $p$  相同时, 则依据剩余传输时间  $t_r$  进行升序排列, 防止高优先权消息因即将超时而丢失。

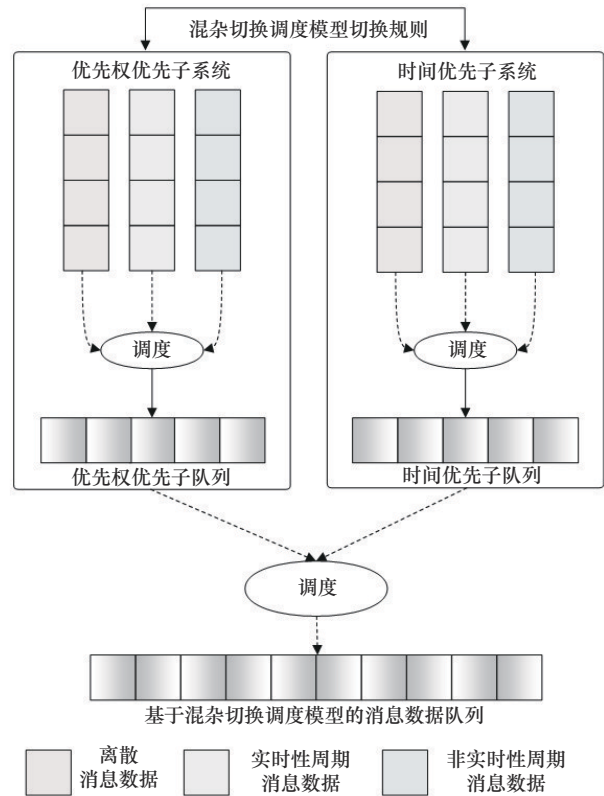


图 5 混杂切换调度模型

#### 3.2 切换规则及边界条件

整合混杂切换调度模型及关键调度参数后, 设计以下子系统的切换规则。① 从时间优先子系统到优先权优先子系统的转换契机在于队列平均已等待时间  $\bar{t}$  跌落到低于剩余等待时间下限  $t_{\min}$ , 或是当平均已等待时间  $\bar{t}$  未超过剩余等待时间上限  $t_{\max}$ , 且当队列中包含亟须处理的关键信息片段时。既保障了高优先级消息的快速传输, 也顾及紧急且重要的数据流通。② 从优先权优先子系统到时间优先子系统的转换触发条件为队列中监测到接近超时阈值的消息, 或当前队列的平均已等待时间  $\bar{t}$  超越了剩余等待时间上限  $t_{\max}$ 。为确保时间敏感型消息得以及时处理, 避免数据丢失。关于如何判断消息数据即将超时以及  $t_{\max}$  和  $t_{\min}$  的取值如下。

$$t_w \geq R_0 \left( d_{\max}^{\text{MP}} - \frac{\text{RTT}}{2} \right) \quad (6)$$

$$t_{\max} = R_{\max} \sum_{\text{MP}=-10}^{10} \left( d_{\max}^{\text{MP}} - \frac{\text{RTT}}{2} \right) \frac{n_{\text{MP}}}{n} \quad (7)$$

$$t_{\min} = R_{\min} \sum_{\text{MP}=-10}^{10} \left( d_{\max}^{\text{MP}} - \frac{\text{RTT}}{2} \right) \frac{n_{\text{MP}}}{n} \quad (8)$$

当消息数据的剩余等待时间满足式(6)时,认为该消息数据即将超时。其中,  $d_{\max}^{\text{MP}}$  表示优先级为MP的消息数据的最大传输时间, RTT表示往返时延,  $R_0$ 表示实数系数。  $t_{\max}$ 和  $t_{\min}$ 分别由式(7)和式(8)给出, 其中  $n_{\text{MP}}$ 表示当前待发送队列中优先级为MP的消息数据的个数,  $n$ 表示待发送队列中消息数据的总数,  $R_{\max}$ 和  $R_{\min}$ 表示实数系数。

### 3.3 调度算法与分析

在上述有关消息数据的参数和混杂切换调度模型的基础上, 设计了ROS2-DDS中面向时间敏感属性的多消息动态调度方法, 如算法1所示。

**算法1** ROS2-DDS中面向时间敏感属性的多消息动态调度方法

输入 待发送消息数据队列

输出 待发送队列的队首消息数据

```

1) while(true) do
2)   根据式(3)计算待发送队列的平均已等待时间  $\bar{t}$ 
3)   获取当前所处的子系统
4)   if 当前所处的子系统为优先权优先子系统 then
5)     根据式(6)计算并统计即将超时的消息数据
6)     if  $\bar{t} > t_{\max}$  或者存在即将超时的消息数据 then
7)       切换为时间优先子系统
8)     end if
9)   else
10)    统计离散消息数据
11)    if 存在离散消息数据并且  $\bar{t} < t_{\max}$ , 或者  $\bar{t} < t_{\min}$  then
12)      切换为优先权优先子系统
13)    end if
14)  end if
15)  发送队首消息数据
16) end while

```

TSDSIR算法基于消息数据优先权、待发送队列的平均已等待时间和最大传输时间, 判断当前系统所处的子系统, 如果处于优先权优先子系统, 则判断待发送队列是否存在即将超时的消息数据; 如果处于时间优先子系统, 则判断待发送队列是否存在离散的高优先权消息数据。2个子系统均采用稳

定的插入排序算法, 新产生的消息数据根据当前系统所处的子系统的排序规则插入待发送队列。根据切换规则判断当前系统状态量是否满足切换条件, 如果满足, 进行子系统切换, 然后发送待发送队列的队首消息数据; 如果不满足, 则直接发送待发送队列的队首消息数据。对TSDSIR算法的相关属性进行分析, 可得出以下定理。

**定理1** 对于发送缓冲队列长度为  $n$  的发送节点, 运行TSDSIR算法的最坏时间复杂度和空间复杂度分别为  $O(n^2)$  和  $O(n)$ 。

**证明** 如算法1所示, TSDSIR算法是一个循环发送队首消息数据的执行过程, 直至发送任务终止。对于长度为  $n$  的发送队列, 计算队列中消息数据的平均已等待时间和查找是否存在即将超时的消息数据的时间复杂度均为  $O(n)$ 。时间优先队列和优先权优先队列的构建采用插入排序算法以保持相对稳定的消息数据发送次序, 在最坏的情况下, 插入排序算法的时间复杂度为  $O(n^2)$ , 因此, 对于数据规模为  $n$  的TSDSIR算法的时间复杂度为  $T(n) = O(\max(n, n^2)) = O(n^2)$ 。TSDSIR算法的空间复杂度与待处理消息数据的数量  $n$  有关, 即时间优先队列和优先权优先队列通常需要存储  $n$  个待发送消息数据队列, 其消耗的存储空间与待处理消息数据的数量呈线性关系, 因此TSDSIR算法的空间复杂度为  $S(n) = O(n)$ 。

证毕。

## 4 实验分析

针对本文所提调度算法从2个角度进行测试分析。①基于仿真模式进行性能测试, 在Linux平台上使用C++和CMake编写模拟程序, 对ROS2中网络资源的调度过程进行实验仿真, 针对机器人网络数据分发服务中不同优先级话题数据和离散消息数据占比不同的情况, 对TSDSIR算法进行仿真实验。通过与非时间敏感的严格优先级调度 (SP, strict priority scheduling) 算法、经典加权循环调度 (CWRR, classical weighted round robin scheduling) 算法以及其改进之后的交替加权循环 (IWRR, weighted round robin with interleaved arbitration) 算法的实验对比<sup>[18]</sup>, 从消息数据传输时延、丢包情况等方面分析了TSDSIR算法的性能。②Fast DDS集成与实物机器人应用, 将TSDSIR算法移植并添加到Fast

DDS 的流控制器调度策略中,并在实际的局域网多终端发布订阅通信中,与 Fast DDS 内置的流控制器调度策略对比分析消息数据的传输时延和丢包情况,然后结合远程视觉 SLAM 应用场景,利用以 Raspberry Pi 4B 为主控的 ROS2 智能轮式机器人对上述成果进行应用测试。

### 4.1 仿真实验设计

实验场景设置为发布不同话题消息的机器人端和根据相应话题接收消息的服务端之间的发布订阅消息场景。仿真实验场景和仿真参数设置如图 6 和表 2 所示,机器人采集传感器数据并通过不同的发布节点发布感知数据,充当发送节点。服务器通过各种订阅节点收集感知数据并进行复杂的推理决策,充当接收节点。发送节点生成消息数据并发送,接收节点负责接收消息数据,然后对接收到的消息数据进行统计分析。

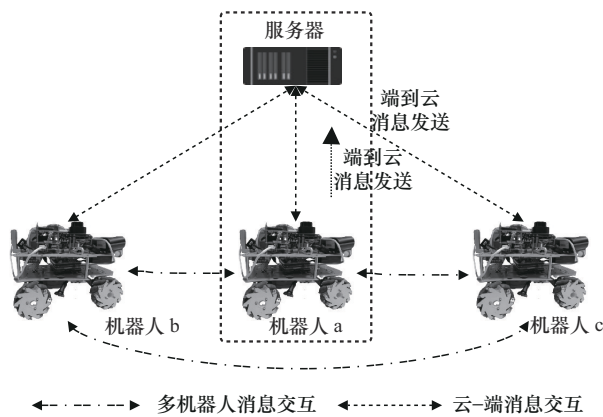


图 6 仿真实验场景

表 2 仿真参数设置

参数	值
机器人节点中的发送缓冲队列长度	10
机器人节点和服务器节点之间的平均传播时延/ms	1

仿真实验参考时间敏感网络中的典型流量,选取不同优先级的消息数据并设置不同优先级消息数据的参数,同时根据机器人网络通信的需求进行调整和优化。仿真实验选取优先级为 0、1、4 和 5 的消息数据分别代表机器人网络中的控制消息数据(命令、指令等)、网络管理消息数据(网络发现、配置等)、低时延消息数据(实时语音等)和普通消息数据(图像、视频等)进行实验,其中,优先级的值越小表示消息数据的优先级越高,即消息数据优先权重  $p$  越大,优先级的值越大表示消息数据

的优先级越低,即消息数据优先权重  $p$  越小,各优先级的消息数据参数设置如表 3 所示。

表 3 消息数据参数设置

类型	周期性	最大传输时间/ms	优先级
控制消息数据	离散	20	0(高)
网络管理消息数据	连续	50	1
低时延消息数据	连续	100	4
普通消息数据	连续	200	5(低)

仿真实验选取 SP 算法、CWRR 算法以及其改进之后的 IWRR 算法作为对比算法,进行仿真模拟实验分析。① SP 算法是一种按照流量优先级进行调度的算法,将消息数据按照其优先级从高到低依次加入队列,并按照优先级从高到低依次进行服务。② CWRR 算法是一种按照流量权重进行调度的算法,所有流量按照其权重值从大到小依次分配资源,每个流量按照所分配的资源进行服务,直到所有流量被服务完毕。③ IWRR 算法是一种改进的 WRR 算法,它将所有流量分成多个组,每个组内的流量按照其权重值从大到小依次分配资源。当所有组内的流量被服务完毕后,下一轮会重新按照组的顺序进行资源分配,确保每个流量都能得到公平的服务机会。在该仿真实验中,CWRR 和 IWRR 算法的不同优先级消息数据的发送权重参数设置如表 4 所示。

表 4 WRR 算法各优先级消息数据的发送权重参数设置

优先级	权重
0	0.4
1	0.3
4	0.2
5	0.1

在仿真实验中,随机生成多组包含不同优先级的若干消息数据序列时,各比例离散性消息数据分布均匀且随机设置,分别测试消息数据在不同算法调度下的传输情况,减小误差和随机性。在不同离散性消息数据占比的场景中,测量多组数据并取实验结果平均值,保证实验结果的普遍性和准确性。

### 4.2 仿真实验结果与分析

随机生成包含 10 000 个消息数据的消息序列,其中包括优先级为 0、1、4 和 5 的 4 种类型,并将

离散性消息数据的比例设置为30%，按顺序将这些随机生成的消息数据序列输入仿真实验，并采用TSDSIR进行调度优化，得到待发送队列的平均已等待时间仿真结果，如图7所示。图7截取了第200个到第600个发送消息数据的平均已等待时间的变换情况，随着机器人节点与服务器节点之间消息数据的持续收发，待发送队列的平均已等待时间 $\bar{t}$ 不断变化，但基于混杂切换调度模型的TSDSIR将 $\bar{t}$ 限制在 $(t_{min}, t_{max})$ ，可以在一定程度上保证消息数据的传输时间不超过传输时延的上界，从而保证了消息数据能够及时到达服务器接收端。

随机生成10组共计10 000条消息的数据序列，每组消息数据内部包含0、1、4和5的4种优先级，且离散性比例逐步变化，设定值分别为5%~50%的等差序列进行仿真，实施对发送队列中消息调度的测试，涉及TSDSIR、SP、CWRR和IWRR4种算法。实验结果聚焦于离散性消息数据比例5%~50%变化区间内各算法处理不同优先级消息时的平均时延对比，如图8所示。在离散性消息数据占比不同的场景下，4种算法在优先级0、1和4的消息数据平均时延都在最大传输时间以内，由图8(a)分析可知，在高优先级的消息数据仿真结果中，TSDSIR在不同离散性消息数据占比下均维持较低的平均时延，与以高优先级实时性著称的SP算法差距甚小，而CWRR和IWRR算法的平均时延随离散性消息数据占比的增加而上升，相较于TSDSIR和SP算法，其平均时延较高。由图8(b)和图8(c)分析可知，在

中等优先级的消息数据仿真结果中，随着离散性消息数据占比的增加，TSDSIR的平均时延上升且略高于其他算法，但仍远低于最大传输时间。由图8(d)分析可知，在低优先级消息数据仿真结果中，随着离散性消息数据占比的增加，4种调度算法的平均时延均呈上升趋势。其中，SP、CWRR和IWRR算法在离散性消息数据占比约35%、40%和45%时，平均时延超过最大传输时间。然而，TSDSIR在不同离散性消息数据占比下均能保持平均时延低于最大传输时间，特别是在离散性消息数据占比增大时，其平均时延远低于SP算法的平均时延。其次，TSDSIR在不同离散消息的负载情况下，相比对比算法在平均时延指标方面体现了较好的性能，反映了该方法在不同离散消息的负载情况下的适应性。综上所述，TSDSIR在高优先级消息数据传输中有着与SP算法相媲美的实时性，同时在中等优先级消息数据中也同其他3种算法一样有着较低的传输时延，并且在低优先级消息数据中有着比另外3种算法更低的平均时延，即TSDSIR在平衡各优先级消息数据的实时性方面优于其他3种调度算法。因此，TSDSIR在不同离散性消息数据占比和优先级消息数据的传输中具有较好的实时性和可靠性，同时还具备对不同数据传输场景的广泛扩展性和高度适应性。

为11组各含10 000条消息数据的序列赋予不同优先级，并设定离散性消息数据占比从0递增至50%（以5%为步长），进行仿真实验。运用TSDSIR、SP、CWRR和IWRR4种算法来调度这些序

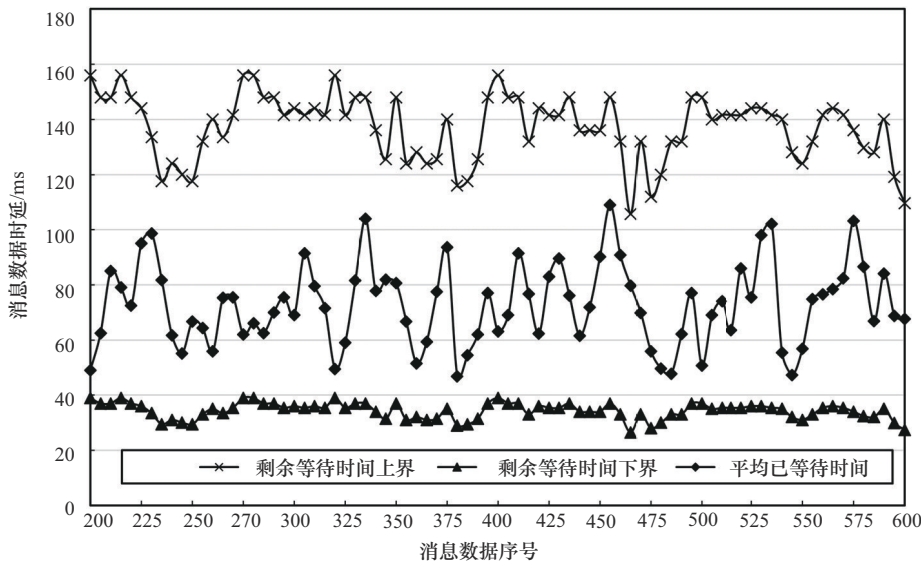


图7 待发送队列的平均已等待时间

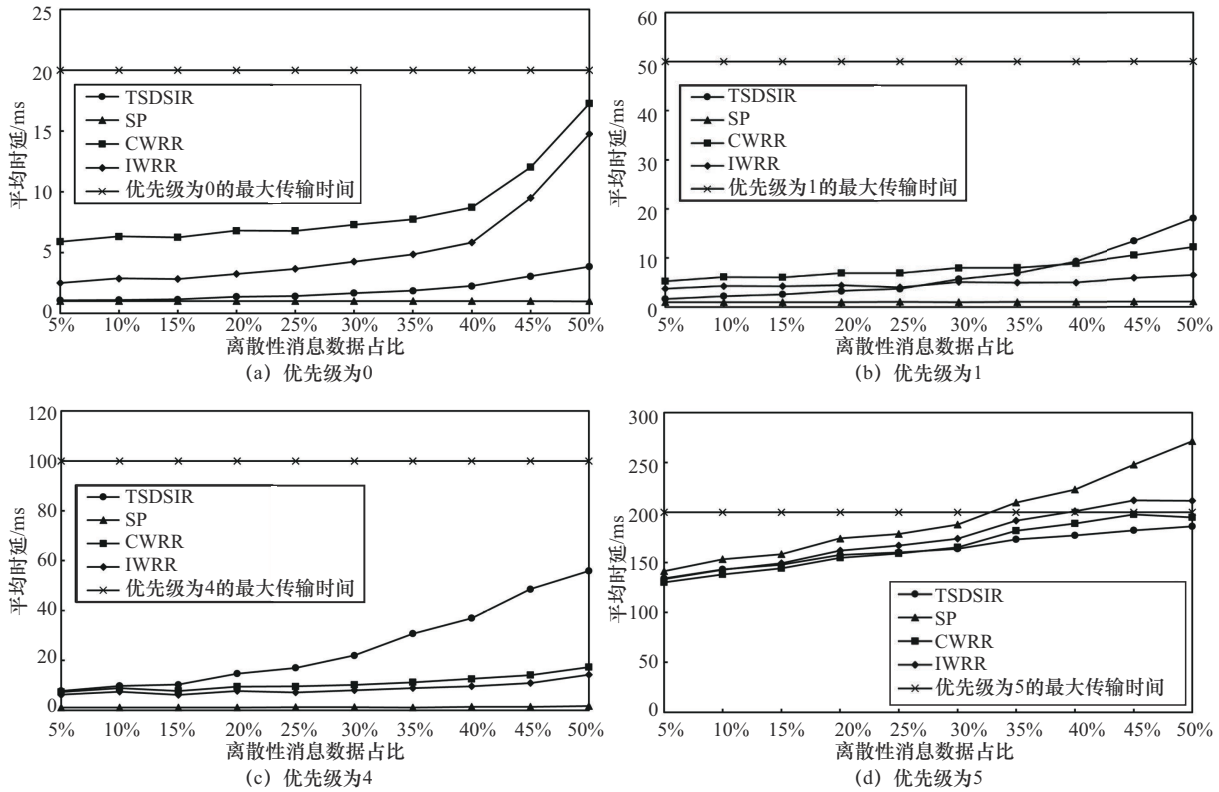


图8 不同优先级下各调度算法在不同离散比下的平均时延

列中的消息发送队列，并记录在连续传输 10 000 条消息数据过程中，不同离散性消息数据占比下的总丢包情况。图 9 展示了这一系列实验模拟的结果，TSDSIR 在所有测试条件下丢包率最低，与其他 3 种算法形成明显对比。尤其是在离散性消息数据占比 0 至 25% 的范畴内，TSDSIR 几乎无超时丢包现象，而 SP 算法的丢包率最高，CWRR 与 IWRR 算法表现居中。随着离散性消息数据占比攀升，尽管所有算法的丢包率均有所增长，TSDSIR 依旧维持了远低于其他 3 种算法的丢包水平。综上所述，TSDSIR 在低离散度环境下能高度确保各优先级消息的准时传递，即使面对高离散度挑战，依然能有效促进更多消息数据的可靠传输。

进一步，聚焦于离散性消息数据占比达 50% 的情况，如图 10 所示，TSDSIR 对低优先级消息数据展现出极低的丢包率，远优于其他算法，尽管在中优先级消息上的丢包率略高于其余算法但差距微小，而在高优先级消息上，TSDSIR 与 SP 算法的性能接近，且两者皆显著优于 CWRR 与 IWRR 算法。总体而言，SP 算法虽保障了高优先级消息的时效性，却牺牲了低优先级消息的传输效率，导致该类消息的丢包率最高。CWRR 与 IWRR 算法在中等优

优先级消息处理上较为及时，但未能平衡高、低优先级消息的丢包问题。反观 TSDSIR 算法在确保高优先级消息的高效与实时传输的同时，显著提升了低优先级消息的传输可靠性，优于其他算法的表现。

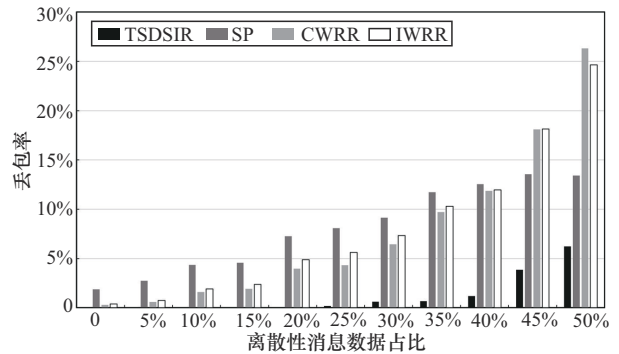


图9 离散数据占比 0~50% 场景下总丢包率

### 4.3 TSDSIR 集成实现与实物实验分析

在测试并验证了 TSDSIR 性能的仿真实验基础上，首先，将 TSDSIR 集成实现到 Fast DDS 的流控制器调度策略中，并参考目前典型机器人云平台服务场景<sup>[19]</sup>，利用搭载 Ubuntu22.04 操作系统的 Raspberry Pi 4B 和 PC 机构建本地云-端网络通信实验场景，将集成后的 Fast DDS 部署到本地云-端系统并

编译。其次，通过局域网中本地云-端订阅发布通信，进行多组不同优先级混杂消息数据的发布订阅实验，并与未使用 TSDSIR 的调度策略进行对比与分析。最后，将集成后的 Fast DDS 部署在以 Raspberry Pi 4B 为主控的 ROS2 智能轮式机器人中，有利于本地云-端网络通信构建远程视觉 SLAM 实验场景，在包含不同优先级的混杂消息数据传输下对 TSDSIR 流控制器调度策略进行应用与测试。

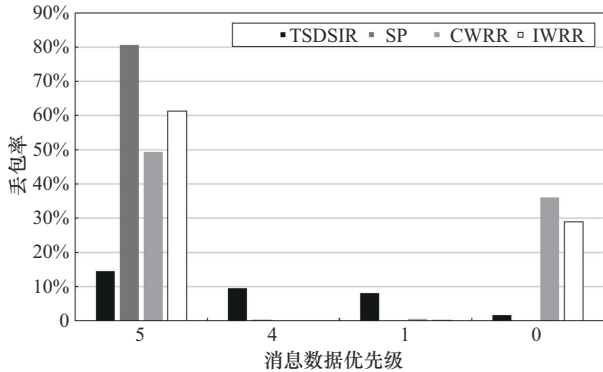


图 10 离散性消息数据占比 50% 场景下各消息数据优先级丢包率

### 4.3.1 TSDSIR 实现与集成

将本文提出的 TSDSIR 集成实现到 Fast DDS 的流控制器机制中，作为其流控制器策略的一个新选项。Fast DDS 现有的流控制器机制内嵌有 4 种策略：FIFO、HIGH\_PRIORITY、ROUND\_ROBIN 和 PRIORITY\_WITH\_RESERVATION。FIFO 采取简单直接的先到先服务原则处理消息；HIGH\_PRIORITY 则侧重于优先传输来自高优先级消息数据源的信息；ROUND\_ROBIN 策略通过循环遍历每个数据源以确保公平的发送机会；而 PRIORITY\_WITH\_RESERVATION 与 HIGH\_PRIORITY 相似，额外要求数据写入者预留带宽资源。依据所选的调度逻辑，流控制器机制会配置相应的流队列接纳来自不同源头的消息数据队列，并依据选定的策略在这些队列中作出发送优先级的决策，流控制器调度策略如图 11 所示。

TSDSIR 调度策略主要包含优先权优先和时间优先 2 个子系统以及根据相应调度参数判断子系统的切换规则，即按优先权降序排列的优先权优先队列、按剩余传输时间升序排列的时间优先队列以及何时从哪个优先队列优先发送消息数据。因此，结合 Fast DDS 中流队列双链表的存储结构，在 Fast DDS 的 TSDSIR 流控制器调度策略实现中，设计出如图 12 所示的流队列存储结构（图中省略节点的前驱链指

针），head 和 tail 节点表示优先权优先队列的头节点和尾节点，进入流控制器的消息数据根据其优先级属性分别进入相应的优先权队列，由高优先级到低优先级的优先权队列组成优先权优先队列。earlier 和 later 节点表示时间优先队列的头节点和尾节点，进入流控制器的消息数据同时将按照剩余传输时间在该时间优先队列中升序排列。该方法使用类十字链表的方式同时建立优先权优先队列和时间优先队列，每个消息数据的左右链指针连接优先级相邻的消息数据，上下链指针连接剩余传输时间相近的消息数据，在一定程度上降低了 Fast DDS 流控制器策略中 TSDSIR 算法的时间复杂度和空间复杂度。

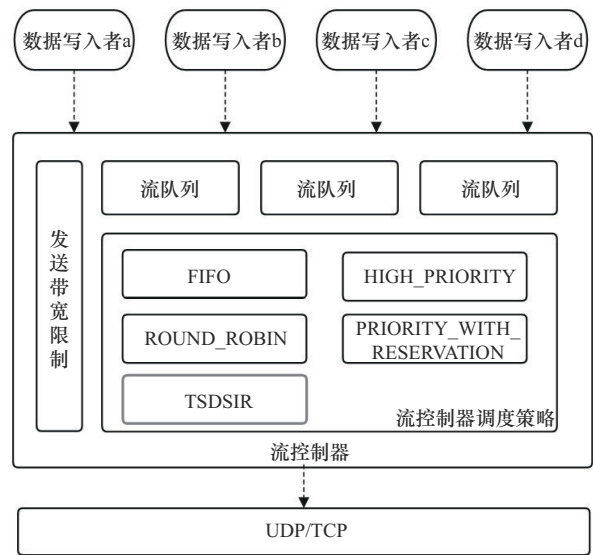


图 11 流控制器调度策略

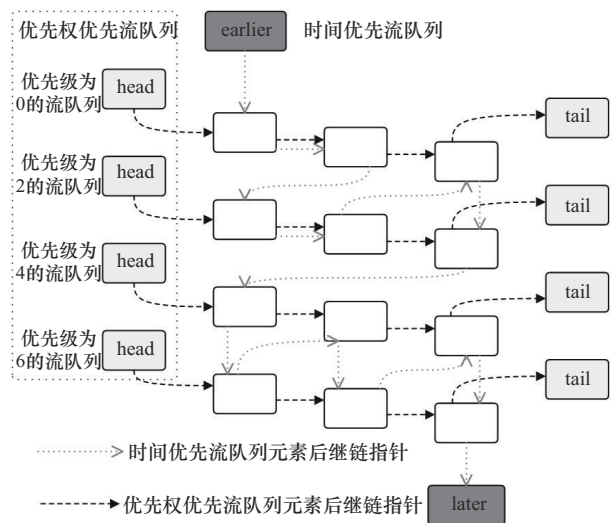


图 12 TSDSIR 流队列存储结构

### 4.3.2 实物实验与测试分析

实物实验设置为: PC 主机和 Raspberry Pi 4B 通过 RJ45 网口连接同一本地路由器, 保证本地局域网稳定的同时设置较为准确的时间同步, 并利用 DDS 的自动发现机制和分布式通信特点构建发布订阅通信。Raspberry Pi 4B 端发送不同优先级的消息数据, PC 端接收消息数据并统计丢包率和平均传输时间等信息。实物实验选取优先级为 0、2、4 和 6 的消息数据, 分别表示机器人系统中不同即时性和关键性需求的话题通信数据, 不同优先级消息数据参数设置如表 5 所示, 每类优先级消息数据设置 1 000 组, 每个消息数据的大小设置为 489 680 B (带有消息数据的 RGB 图像帧), 设置较大的消息数据来增大实验中消息数据的传输时间, 从而减小不稳定网络和多设备时间同步对实验结果带来的误差。

表 5 不同优先级消息数据参数设置

优先级	最大传输时间/ms	发布频率/Hz
0(高)	30	30
2	32	30
4	38	30
6(低)	40	30

在实物实验中, 首先, 将集成 TSDSIR 的 Fast DDS 中间件部署在 PC 主机和 Raspberry Pi 4B 中, 并分别在本地云侧和端侧实现 DDS 的消息订阅应用程序和消息发布应用程序。其次, 在消息发布应用程序中, 开辟 4 个发送子线程, 每个子线程创建数据写入者并负责发送某一话题下的指定优先级消息数据。然后, 配置流控制器发送带宽和调度策略, 并在发布不同优先级消息数据的数据写入者中引用该流控制器, 流控制器的发送带宽设置为 400 Mbit/s, 将流控制器的调度策略分别切换至 FIFO、HIGH\_PRIORITY、ROUND\_ROBIN 和 TSDSIR 进行实物实验对比分析。最后, 分别针对 4 种不同的调度策略进行多次实验, 统计实验结果并取平均值 (实际网络中进行多端发布订阅通信存在一定的误差), 4 种调度策略的消息数据丢包率和平均传输时间结果分别如图 13 和图 14 所示。

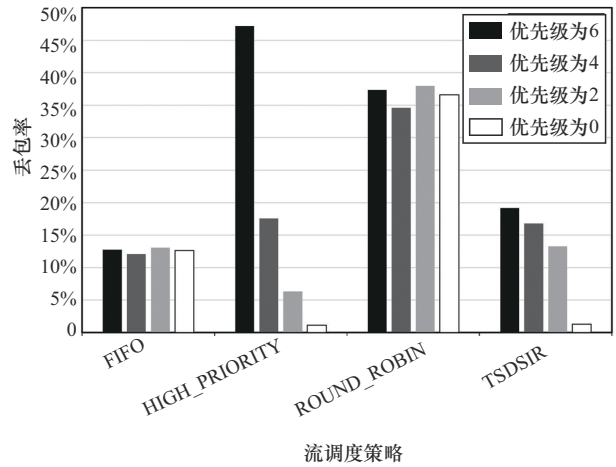


图 13 混杂消息数据传输下各流调度策略丢包率

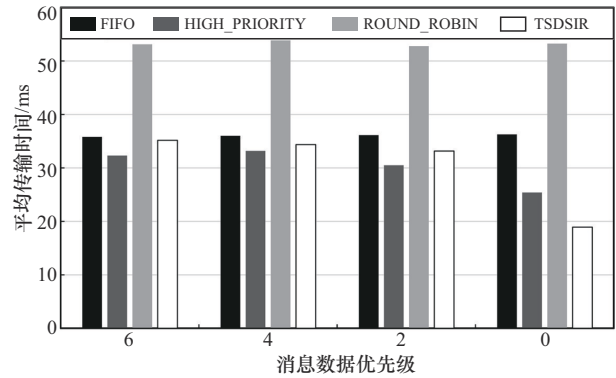


图 14 混杂消息数据传输下各流调度策略平均传输时间

从图 13 中可以看出, FIFO 流调度策略整体丢包率偏低, 但关键高优先级消息的丢包仍较为显著; HIGH\_PRIORITY 流调度策略显著降低了中高优先级, 尤其是最高优先级消息数据的丢包率, 不过牺牲了低优先级消息的传输可靠性, 其丢包率在四者中居首; ROUND\_ROBIN 策略则展现出全优先级较高的丢包趋势; 相比之下, TSDSIR 流调度策略在所有优先级上的丢包率均保持低位, 尤其在保护高优先级消息免于丢失方面, 表现与 HIGH\_PRIORITY 流调度策略接近。从图 14 中可以看出, 在传输的各优先级消息数据中, 使用 TSDSIR 流调度策略传输的消息数据平均传输时间相对于其他 3 种流调度策略较少, 尤其是高优先级的消息数据平均传输时间相对于使用其他流调度策略最少。综上所述, 针对不同优先级的混杂消息数据同时传输时, 集成 TSDSIR 流调度策略的 Fast DDS 发布订阅通信具有较低丢包率和较低平均传输时间, 尤其是在高优先级消息数据传输中表现更佳。

### 4.3.3 实物应用与测试

在 4.3.2 节所述的实验环境基础上，进一步搭建基于本地云-端的远程视觉 SLAM 应用场景。

在实物应用与测试中，采用以 Raspberry Pi 4B 为主控的麦克纳姆轮机器人和远程通信的 PC 端，两者通过 Wi-Fi 连接同一无线路由器，应用整体架构设计如图 15 所示，机器人通过摄像头传感器连续采集图像信息，并实时发送至远程 PC 端进行视觉 SLAM 处理，同时，机器人也将自身状态信息发送告知远程通信的 PC 端，如欠压警告信息、故障诊断信息等。其中，机器人相关警告信息设置为高优先级的离散性信息，连续发送的图像帧信息设置为较低优先级的连续性信息。

将集成 TSDSIR 流调度策略的 Fast DDS 部署到机器人端和 PC 端，机器人端将实时发送采集的图

像帧信息和不定时发送机器人相关警告信息等（为了更好地设置实验场景，警告信息通过编写模拟节点随机发送）统一通过 Fast DDS 通信中间件发布，远程通信的 PC 端负责订阅接收机器人端发送的图像帧信息，并利用基于 ORB-SLAM2 的远程视觉 SLAM 功能节点进行轨迹估计，同时订阅接收机器人相关警告信息并及时响应处理。由于麦克纳姆轮机器人的主控 Raspberry Pi 4B 性能和带宽有限，实时采集发送的图像均需要进行压缩处理。

如图 16 和图 17 所示，在该实验场景中，麦克纳姆轮式机器人按图中虚线移动并实时采集数据发送，PC 端能实时订阅到该机器人发送的警告信息，且在保证离散数据的实时性的前提下，远程单目视觉 SLAM 订阅图像帧并进行的估计轨迹误差在正常范围内。

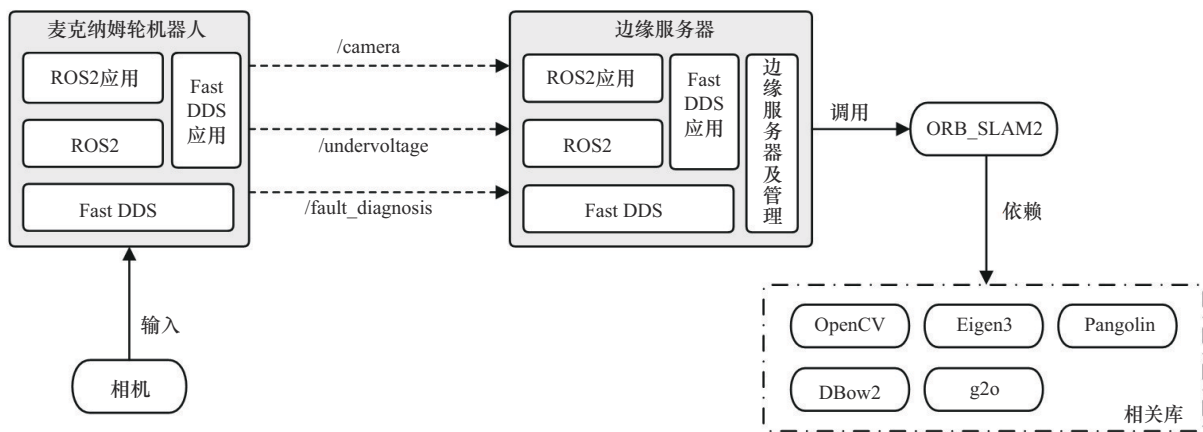
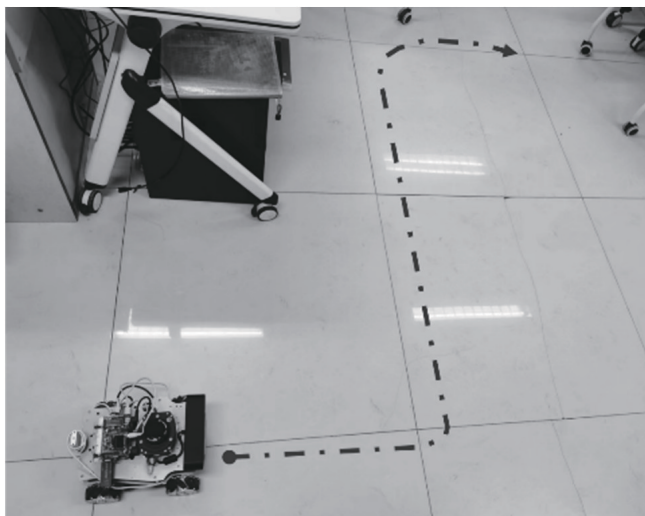
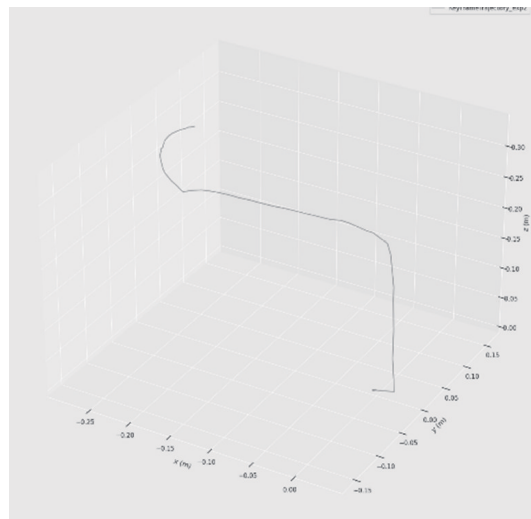


图 15 应用整体架构设计

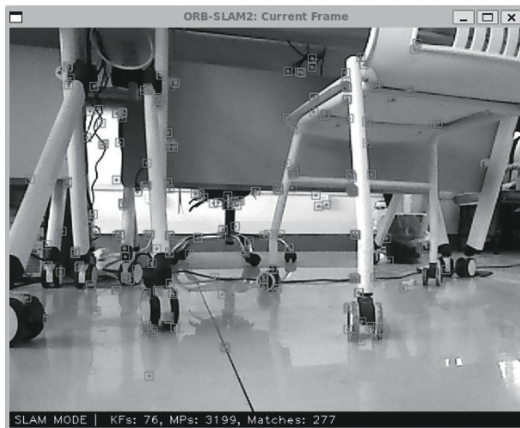


(a) 机器人实际运动轨迹

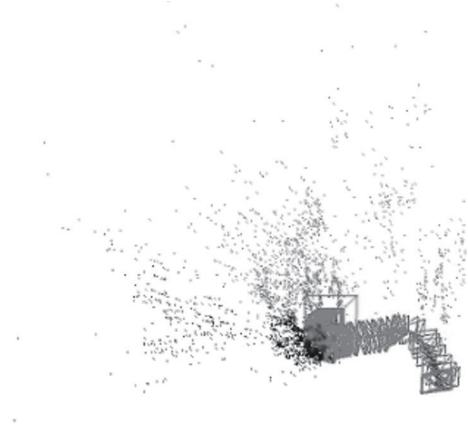


(b) 机器人远程视觉SLAM估计轨迹

图 16 机器人实际运动轨迹和远程视觉 SLAM 估计轨迹



(a) 机器人远程视觉SLAM中特征点提取



(b) 机器人远程视觉SLAM位置估计

图17 机器人远程视觉SLAM中特征点提取和位置估计

## 5 结束语

本文根据机器人网络中对多样性、多模态混杂消息传输的实时性和可靠性需求, 针对以 DDS 为通信中间件的 ROS2, 将选取哪个话题消息流数据优先发送的过程设计为包含 2 个子系统的混杂切换调度模型并定义了有关待发送队列中消息数据的相关参数, 为消息数据调度提供依据, 然后, 基于混杂切换的调度模型, 提出了 ROS2-DDS 中面向时间敏感属性的多消息动态调度方法, 既考虑消息数据的关键性和优先级也考虑各消息数据发送的紧迫程度。

通过仿真实验和 TSDSIR 集成实验与实物测试分别证明和实践了本文方法相对于现有方法, 满足了高优先级消息数据的传输需求, 降低了丢包率, 在面向离散数据、连续数据和不同优先级数据传输的场景中提高了消息传输的实时性和可靠性。其次, 集成 TSDSIR 调度策略的 DDS 在混杂切换场景下展现出较好的传输性能, 对采用 DDS 作为通信中间件的系统具有良好的扩展性。TSDSIR 方法在不同离散数据负载下均能保持出色的传输性能, 有效平衡各优先级消息数据的传输需求, 展现出在多样化数据传输场景中的高度适应性。

## 参考文献:

- [1] 芦倩, 李晓娟, 关永, 等. 面向数据流的 ROS2 数据分发服务形式建模与分析[J]. 软件学报, 2021, 32(6): 1818-1829.  
LU Q, LI X J, GUAN Y, et al. Modeling and analysis of ROS2 data distribution service for data flow[J]. Journal of Software, 2021, 32(6): 1818-1829.
- [2] 王大伟, 王卓, 王鹏, 等. 基于边缘计算的云原生机器人系统[J]. 智能科学与技术学报, 2020, 2(3): 275-283.  
WANG D W, WANG Z, WANG P, et al. Cloud native robot system based on edge computing[J]. Chinese Journal of Intelligent Science and Technology, 2020, 2(3): 275-283.
- [3] 鲁敬敬, 秦云川, 刘志中, 等. 机器人操作系统 ROS 安全性研究综述[J]. 软件学报, 2024, 35(2): 1010-1027.  
LU J J, QIN Y C, LIU Z Z, et al. Survey on security of robot operating system ROS[J]. Journal of Software, 2024, 35(2): 1010-1027.
- [4] DING N, WANG Y H, ZHAO Y W, et al. Packet scheduling algorithm based on hybrid system theory in the industrial Internet of things[C]// Proceedings of the 2021 IEEE International Conference on Smart Internet of Things (SmartIoT). Piscataway: IEEE Press, 2021: 33-39.
- [5] 黄韬, 汪硕, 黄玉栋, 等. 确定性网络研究综述[J]. 通信学报, 2019, 40(6): 160-176.  
HUANG T, WANG S, HUANG Y D, et al. Survey of the deterministic network[J]. Journal on Communications, 2019, 40(6): 160-176.
- [6] MESSENGER J L. Time-sensitive networking: an introduction[J]. IEEE Communications Standards Magazine, 2018, 2(2): 29-33.
- [7] Object Management Group. Object management group. data distribution service specification: Version 1.4[S]. 2015.
- [8] Object Management Group. Object management group. DDS interoperability wire protocol specification: Version 2.2[S]. 2014.
- [9] KANG Z W, CANADY R, DUBEY A, et al. A study of publish/subscribe middleware under different IoT traffic conditions[C]//Proceedings of the International Workshop on Middleware and Applications for the Internet of Things. New York: ACM Press, 2020: 7-12.
- [10] AULIYA R S, CHEN C C, LIN P R, et al. Optimization of message delivery reliability and throughput in a DDS-based system with per-publisher sending rate adjustment[J]. Telecommunication Systems, 2023, 84(2): 235-250.
- [11] MAHEMUTI P, YANG L, LUO L L. Modeling, analysis and synthesis of hybrid system: a review[J]. Applied Mechanics and Materials, 2014,

615: 36-43.

- [12] CHEN C, WANG C, QIU T, et al. Caching in vehicular named data networking: architecture, schemes and future directions[J]. IEEE Communications Surveys & Tutorials, 2020, 22(4): 2378-2407.
- [13] CHEN N, QIU T, LU Z L, et al. An adaptive robustness evolution algorithm with self-competition and its 3D deployment for Internet of things[J]. IEEE/ACM Transactions on Networking, 2022, 30(1): 368-381.
- [14] CHEN C, LIU L, QIU T, et al. Delay-aware grid-based geographic routing in urban VANETs: a backbone approach[J]. IEEE/ACM Transactions on Networking, 2019, 27(6): 2324-2337.
- [15] SUDHAKARAN S, MAGESHKUMAR V, BAXI A, et al. Enabling QoS for collaborative robotics applications with wireless TSN[C]//Proceedings of the 2021 IEEE International Conference on Communications Workshops (ICC Workshops). Piscataway: IEEE Press, 2021: 1-6.
- [16] IOANA A, KORODI A. DDS and OPC UA protocol coexistence solution in real-time and industry 4.0 context using non-ideal infrastructure[J]. Sensors, 2021, 21(22): 7760.
- [17] LIN H, ANTSAKLIS P J. Hybrid dynamical systems: an introduction to control and verification[J]. Foundations and Trends® in Systems and Control, 2014, 1(1): 1-172.
- [18] TABATABAEE S M, LE BOUDEC J Y, BOYER M. Interleaved weighted round-robin: a network calculus analysis[C]//Proceedings of the 2020 32nd International Teletraffic Congress (ITC 32). Piscataway: IEEE Press, 2020: 64-72.
- [19] 周凤余, 尹磊, 宋锐, 等. 一种机器人云平台服务构建与调度新方法[J]. 机器人, 2017, 39(1): 89-98.
- ZHOU F Y, YIN L, SONG R, et al. A novel building and scheduling method of cloud platform services for robot[J]. Robot, 2017, 39(1): 89-98.

## [作者简介]



丁男 (1978-), 男, 辽宁大连人, 博士, 大连理工大学教授、博士生导师, 主要研究方向为物联网与工业互联网技术、网络系统安全与数据分析、多智能体协同、嵌入式人工智能等。



冀承慧 (1998-), 男, 山东泰安人, 新疆师范大学硕士生, 主要研究方向为物联网技术及其应用、云-边-端机器人系统通信优化等。



胡创业 (1994-), 男, 河南周口人, 新疆师范大学讲师, 主要研究方向为物联网安全、多智能体协同、自然语言处理等。



许力 (1982-), 男, 辽宁大连人, 博士, 字节跳动高级工程师, 主要研究方向为网络虚拟化、云计算、概率图模型等。