

DOI: 10.13382/j.jemi.B2306604

突发流量自适应的 6TiSCH 网络分布式 PID 调度*

张焱^{1,2} 陈峰¹ 谢泽堃¹ 蔡有鑫¹ 韩延^{1,2} 黄庆卿^{1,2}

(1. 重庆邮电大学工业物联网与网络化控制教育部重点实验室 重庆 400065;

2. 重庆邮电大学工业互联网研究院 重庆 401120)

摘要:工业物联网 6TiSCH 通过对时隙跳频模式和 IPv6 协议的引入,在支持海量节点接入、确定性可靠传输以及低功耗运行方面优势明显。虽然 6TiSCH 操作子层的定义为用户自定义调度策略提供了规范化的调度执行过程,但在实际应用中仍面临不同应用场景的上层调度函数(即决策实体)个性化设计的问题。本文将控制理论引入确定性工业无线网络的资源调度应用中,将分布式节点间通信资源调控抽象为闭环控制问题,结合经典比例积分微分(proportional-integral-derivative, PID)控制算法,阐述基于 PID 控制理念的 6TiSCH 网络资源调度原理,设计面向突发流量场景的 6TiSCH 网络分布式 PID 调度算法,构建突发流量友好的上层调度函数。基于 OpenMote-B 的 6TiSCH 网络平台实验表明,所提方法支持流量需求动态变化的节点间通信资源的自适应调度,尤其对多种类型突发流量可以快速响应。

关键词:工业物联网;6TiSCH;时隙跳频;分布式调度;PID

中图分类号: TP393;TH701 **文献标识码:** A **国家标准学科分类代码:** 510.99

Distributed PID scheduling in 6TiSCH network with burst traffic adaptation

Zhang Yan^{1,2} Chen Feng¹ Xie Zekun¹ Cai Youxin¹ Han Yan^{1,2} Huang Qingqing^{1,2}

(1. Key Laboratory of Industrial Internet of Things & Networked Control, Ministry of Education, Chongqing University of Posts and Telecommunications, Chongqing 400065, China; 2. Institute of Industrial Internet, Chongqing University of Posts and Telecommunications, Chongqing 401120, China)

Abstract: By introducing the time slotted channel hopping mode and IPv6 protocol, the industrial Internet of Things 6TiSCH exhibits significant advantages in supporting massive node connectivity, deterministic and reliable transmission, and low-power operation. While the defined 6TiSCH operation sublayer provides a standardized scheduling execution process for user-defined scheduling strategies, it still faces the challenge of personalized design for upper-layer scheduling functions (i. e., decision entities) in different application scenarios. In this paper, the control theory is introduced into the scheduling application of the deterministic industrial wireless network, the communication resource control between distributed nodes is generalized as a closed-loop control problem, the principles of resource scheduling in 6TiSCH network are elucidated by combing the classic PID control concept. A distributed PID scheduling algorithm for 6TiSCH network, which is specifically tailored for burst traffic scenarios, is designed, and an upper-layer scheduling function that is friendly to burst traffic is constructed. The experiments conducted on an OpenMote-B based 6TiSCH network platform demonstrate that the proposed method enables adaptive scheduling of communication resources between nodes with dynamically changing traffic demands, and presents rapid responses especially for various types of burst traffic.

Keywords: industrial internet of things; 6TiSCH; time slotted channel hopping; distributed scheduling; PID

收稿日期: 2023-06-06 Received Date: 2023-06-06

* 基金项目:重庆市教委科学技术研究项目(KJQN202000611)、重庆市自然科学基金面上项目(cstc2021jcyj-msxmX0556)资助、重庆市技术创新与应用发展专项重点项目(cstc2020jcsx-dxwtBX0050)资助

0 引言

工业无线技术为工业生产提供低成本、高灵活性的信息系统和环境^[1]。随着无线技术的研究不断深入,逐渐形成针对不同应用场景的无线通信技术规范,6TiSCH^[2](IPv6 over the TSCH mode of IEEE 802.15.4)网络作为其中一种将 IPv6 和时隙跳频模式(time slotted channel hopping, TSCH)融合的工业物联网(industrial Internet of Things, IIoT)网络架构,在支持海量节点接入、确定性可靠传输以及低功耗运行方面优势明显。

由于 6TiSCH 网络拓扑结构、数据流量等处存在动态变化中,6TiSCH 网络节点需要动态调度其通信资源,以实现灵活、高效的数据传输。调度函数(scheduling function, SF)作为 6TiSCH 网络中开展资源调度的上层决策实体,其决定了如何调控调度块和触发 6TiSCH 操作子层协议的策略等^[3],是 6TiSCH 网络架构的核心组成部分。然而,IETF 6TiSCH 工作组并没有对其进行强制的明确定义,在实际应用中由用户根据不同应用场景需求对调度函数进行个性化自定义设计,具有很高的灵活性。

Kharb 等^[4]对适用于 TSCH 网络的相关调度算法进行了梳理和综述,并将其分为集中式、分布式^[5]、自治式和混合式调度算法 4 种。在分布式调度算法方面,6TiSCH 标准化工作组先后定义了 OTF^[6]、SFO^[7]、SF1^[8]等调度函数。目前,MSF^[9](minimal scheduling function, MSF)是 IETF 6TiSCH 工作组发布的首个全功能标准化调度函数,涵盖包括适应动态网络、安全性和控制流量等功能。MSF 虽然在整体功能上较为全面,但是在突发流量较为频繁等特定场景下调度响应的快速性等性能仍需进一步提升。

在分布式网络中,节点间通信资源的调控任务可以看作是一个闭环控制的问题,即在相关约束下取得资源分配和资源需求的平衡,调度算法根据网络的实时反馈调整任务分配和资源分配,以使整个网络保持高效和稳定。PID 控制具有计算成本低、易实现,可扩展性好等特点,在众多领域中得到广泛应用^[10-11]。Domingo-Prieto 等^[11]于 2016 年提出一种基于分布式 PID 的 6TiSCH 网络调度方法,该方法以保持缓存队列中数据包数量为 0 为控制目标。Kori 等^[12]于 2022 年提出一种基于经典模糊控制理论的无线传感器网络资源调度方法,用以降低网络的延迟并提高带宽利用率。总体而言,目前关于控制算法本身的相关研究已经很多,但基于控制理念的工业无线网络资源调度相关研究较少。

本文借鉴 MSF 在安全性和适应动态网络方面的特性并基于 PID 控制理念,提出了一种突发流量自适应的 6TiSCH 网络分布式 PID 调度方法,该方法将控制理论引

入工业无线网络的资源调度应用,将分布式节点间通信资源调控抽象为闭环控制问题,构建突发流量友好的上层调度函数。本文工作主要特色为:1)调度目标方面:自适应的将突发流量下节点间通信资源的使用率控制在合理目标范围;2)技术手段方面:突出了经典控制理论与现代通信技术的创新性融合,借鉴工业控制领域经典 PID 算法,将节点间通信资源分配等价于控制问题,实时监控通信资源的使用率,通过计算 6TiSCH 通信资源的期望值与实际值之间的偏差,以闭环控制方式实现网络资源调度;3)实施效果方面,所设计算法继承 MSF 在安全和适应动态网络方面的优势,同时具备突发流量友好的特性。在 6TiSCH 网络试验平台上对所提方法进行了测试分析。4)方法泛化性方面:该方法可拓展至多种其他网络、控制算法以及应用领域,将网络通信资源调控抽象为控制问题,并基于控制理论开展网络资源调度,为网络调度提供一种解决方案参考。

1 6TiSCH 网络 TSCH 模式资源调度

工业环境下工业无线网络通常需要满足低时延和高可靠的应用需求^[13],IETF 6TiSCH 工业物联网所采用的时隙跳频技术可以使无线网络在复杂的工业环境下达到接近有线网络的可靠性和超低的能耗。

图 1 是 6TiSCH 网络 TSCH 模式资源调度示意图,图中表格横向表示时间,TSCH 网络中时间被分割成若干个时隙,一定数量的时隙组成时隙帧^[14]。时隙在一个时隙帧中的位置编号称为时隙偏移(SlotOffset)。图中表格纵向表示信道,以信道偏移(ChannelOffset)对不同信道进行编号。

在 TSCH 模式^[15]下,由一对时隙偏移和信道偏移所确定的一个小格称为调度块 cell,cell 是调度资源分配的基本单位。在同一时隙内通过信道偏移以支持不同频带内的同时通信,避免了频率冲突和碰撞。此外,节点只在其分配的时隙中进行通信,其余时间处于睡眠状态以实现低功耗运行^[16]。

虽然 6TiSCH 网络中 IEEE 802.15.4 TSCH 链路层标准中定义了时隙跳频通信机制,但是,网络节点具体使用哪个时隙和信道进行通信并没有被定义。为此,IETF 6TiSCH 工作组定义了一个支持分布式调度的 6TiSCH 操作子层(6TiSCH operation sublayer, 6top)^[17],其位于 IEEE 802.15.4 TSCH 链路层之上^[18]。如图 1 所示,6top 层包含了 6top 协议(6top protocol, 6P)和调度函数 SF。6P 和 SF 二者互相配合,共同支撑节点间通信资源分配和协调^[19],其中 SF 作为上层决策实体,决定何时添加或删除节点间的通信资源,并触发 6P 协议以发出节点间的调度请求,而 6P 定义了邻居节点之间 cell 的增加或者删

除的操作。

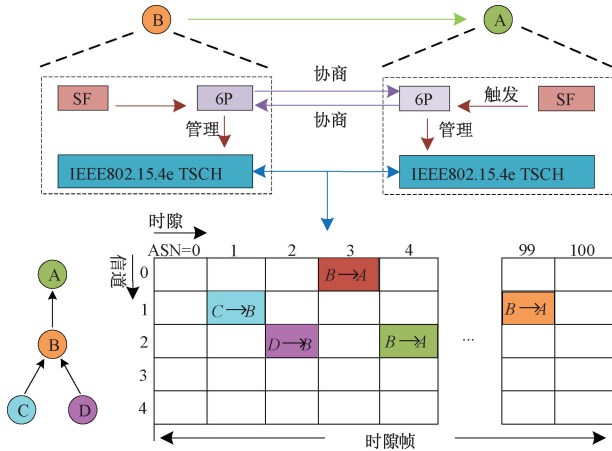


图 1 6TiSCH 网络 TSCH 模式资源调度示意图

Fig. 1 Diagram of TSCH scheduling in 6TiSCH network

2 突发流量自适应的 6TiSCH 网络 PID 调度

2.1 6TiSCH 网络分布式 PID 调度原理

PID 控制是一种应用广泛的控制算法,其通过测量控制系统实际输出 $r(t)$ 与期望目标 $y(t)$ 之间的偏差,计算并调整控制系统输入以使系统偏差最小化。实际输出与期望目标的误差 e_t 计算为:

$$e_t = y(t) - r(t) \quad (1)$$

整个控制过程的目标即实现误差为 0,将控制误差 e_t 作为输入,然后分别经过比例、积分和微分计算,匹配相应的系数 K_p, K_i, K_d 。PID 算法基础公式为:

$$p_t = K_p e_t + K_i \int_0^t e_t dt + K_d \frac{d}{dt} e_t \quad (2)$$

式中:比例增益 K_p 用于微调过程的常量,加快输出响应; K_i 为积分系数,控制器输出对误差信号的累积积分作用,用来消除稳态误差。 K_d 为微分系数,控制器输出对误差信号的微分作用,用来减小系统的超调量和响应时间。

6TiSCH 网络资源调度本质上是对 cell 的个数、cell 的使用率和 TSCH 模式网络资源等的合理调配。如图 2 所示,基于 PID 的 6TiSCH 网络调度核心思想是将节点间 cell 的分配任务看作是一个闭环控制问题,将 cell 数量的实际值与期望值之间的误差作为公式的输入,根据 PID 的输出来判断进行添加或者删除 cell。

2.2 突发流量自适应的分布式 PID 调度算法

基于上述理念,本文提出的突发流量自适应的 6TiSCH 网络分布式 PID 调度算法建立在 6TiSCH 网络的操作子层 6top 之上,其目的是让加入网络的节点拥有足够且合理数量的 cell 满足数据包的传输。在 MSF 的基础上结合 PID 算法的快速响应的特点,减小对 cell 使用率评估的时间间隔,使得 PID 算法在数据包数量激增时

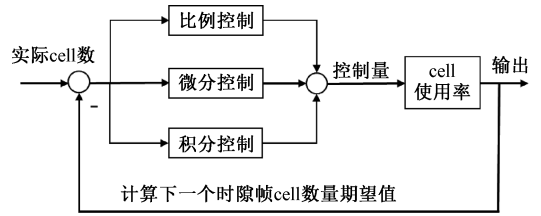


图 2 6TiSCH 网络 PID 调度原理示意

Fig. 2 Diagram of PID scheduling for 6TiSCH network

能够较快做出应对,快速调控合理数量 cell 进行数据传输。将 6TiSCH 网络中 cell 的分配看作一个闭环的控制问题,控制目标即为使分布式管理下的节点间 cell 数量满足数据流量的收发需求,使节点在应对数据流量激增的情况下,能够快速做出响应,调控合适数量的 cell 完成数据的传输。

1) cell 使用率评估

PID 算法在计算期望值时会根据监控的 cell 使用率来进行估算。计算公式为:

$$\text{cell 使用率} = \frac{\text{已使用的 cell 数量}}{\text{已分配的 cell 数量}} \quad (3)$$

式中:已使用 cell 数量 (NumCellsUsed) 和已分配 cell 数量 (NumCellsElapsed) 用于统计 cell 的个数。每经过 C 个时隙帧,通过统计的已分配和已使用的 cell 个数计算在当前时间段内 cell 的实际使用率,根据 cell 的实际使用率可以估计在下一时隙帧内需要分配 cell 的期望值。将期望值与实际值之间的误差代入 PID 公式并计算,根据输出值与阈值的大小来判断是否需要增加或者减少 cell。

2) cell 需求数估计

PID 算法利用当前时隙帧中实际的 cell 数量乘以使用率得到该节点通信时需要 cell 的实际需求,计算期望的 cell 数量时在实际需求的基础上加 n 是为了使节点在较低 cell 数量时能够保持一定的通信资源裕量。公式表示为:

$$r_t = C_n \frac{C_U}{C_E} + n \quad (4)$$

式中: r_t 为下一轮需要分配的 cell 数量期望; C_n 为当前一个时隙帧中 cell 的数量; C_U 为 NumCellsUsed 的值; C_E 为 NumCellsElapsed 的值。

3) 突发流量自适应的 PID 调度

为了更好地应对突发流量,本文以分布式节点间调度块 cell 使用率为 PID 调度控制变量,并对相邻两次调度间的时间间隔进行一致性控制,即每经过 C (文中 C 取值为 4) 个时隙帧,就会对 cell 使用率和 cell 需求数进行评估。

在计算出 cell 期望值后,用下一个周期所需要的 cell 期望值减去当前时隙帧中 cell 的实际值得到误差,作为

PID 控制的输入,公式表达为:

$$e_t = r_t - C_n \quad (5)$$

式中: e_t 为输入误差,也就是期望值和实际值之间的偏差, r_t 为 cell 的期望值, C_n 为当前时隙帧中 cell 的实际值。将误差 e_t 作为控制输入,代入式(6)计算下一轮需要增加或者删减的 cell 数量期望值 u_t 。

$$u_t = K_p e_t + K_i \int_0^t e_t dt + K_d \frac{de_t}{dt} \quad (6)$$

式中: K_p, K_i, K_d 为比例、积分、微分各项系数, e_t 为期望的 cell 与当前的 cell 数量误差。

鉴于 cell 作为通信资源的最小单位为 1,实际中 cell 的调控只能以整数个数进行增减,也即仅当式(6)计算得到的期望值 u_t 大于 1 或者小于 -1 时才会触发 cell 的增减操作,但这一限制就会使 PID 调控在较小 cell 数量需求时响应较慢。为此,设计基于阈值的 cell 增减触发规则,如式(7)所示。

$$\text{操作类型} = \begin{cases} \text{增加 cell, } u_t \geq A \\ \text{减少 cell, } u_t \leq D \\ \text{无触发, } D < u_t < A \end{cases} \quad (7)$$

若 PID 输出的期望值 u_t 超过增加 cell 阈值 A 或低于删减 cell 阈值 D ,调度函数就会触发 6P 请求,以进行增加或删减 cell 的实际操作,以此来稳定节点间 cell 使用率保持在一个合理的范围。

4) 调度算法参数分析

PID 公式的比例部分将误差信号与比例增益相乘,产生一个控制输出,其大小与误差成比例。积分部分将误差信号积分,以抵消系统的稳态误差。微分部分对误差信号进行微分,以预测误差信号的未来变化趋势,从而使控制器能够更快地响应系统的变化。

假设当使 $K_p = 1$,设置其他两项系数为 0,即 PID 控制器公式的输出为与当前输入误差成比例的结果。由于对系统引入了惯性环节,需要通过降低比例误差的权重同时增加积分项对误差的作用来限制突变数据流量造成的误差过大。 K_p 用于确定在某个误差情况下需要增减的 cell 数量。算法中期望分配的 cell 数量和实际需求之间存在一定空闲余量,需要分配额外的 cell,因此在设置 K_p 的值时应该为积分项和微分项留出一定的权重空间,从而避免系统的不稳定性。 K_i 用于稳定系统,用于累计过去的误差。综上,在设置 PID 控制器各系数 K_p, K_i, K_d 时,总体应遵循:

$$K_p \leq 1, K_p \rightarrow 1^- \quad (8)$$

$$K_i \rightarrow 0^+, K_d \rightarrow 0^+, K_i \gg K_d \quad (9)$$

式中: $K_p \rightarrow 1^-$ 表示设置 K_p 为小于且较接近于 1 的值, $K_i \rightarrow 0^+, K_d \rightarrow 0^+$ 表示设置 K_i, K_d 的值均接近于 0,且在大小关系上满足 K_i 远大于 K_d 。

经验试凑法、极点配置原理、临界比例度法、零极点

相消原理等传统方法、以及基于进化算法、神经网络等的新型方法均可用于 PID 参数整定^[20]。本文在上述参数选取范围内,使用经典的经验试凑法按“先比例,再积分,后微分”步骤对参数进行整定。

cell 增减触发阈值 A 和 D 的设置同样对所提分布式 PID 调度算法性能产生影响。在设置阈值 A 和 D 时,总体应遵循:

$$A \rightarrow 1 \quad (10)$$

$$D \rightarrow -1^+ \quad (11)$$

式中: $A \rightarrow 1$ 表示在设置增加 cell 阈值时,将增加 cell 阈值设置在 1 左右,根据式(6)以及实验经验总结,增加 cell 阈值 A 大于 1 会影响系统响应的快速性,若增加 cell 阈值 A 小于 1 则会导致系统对流量波动较为敏感; $D \rightarrow -1^+$ 表示在设置删减 cell 阈值 D 时,阈值设置应略大于 -1,若删减 cell 阈值 D 小于 -1 则会影响系统在流量波峰经过后收回 cell 的快速性。

3 测试验证

3.1 实验平台与参数设置

搭建基于 OpenMote-B 硬件的 6TiSCH 网络试验平台对所提调度算法进行测试。实验基于开源项目 OpenWSN^[21]进行,该项目支持 6TiSCH 在内的完整网络协议栈,包括物理层、链路层和网络层协议。不仅适用于 Windows、Linux、Mac OS 操作系统部署仿真,还支持 OpenMote-B 等硬件设备。

为了测试 PID 算法在不同流量类型下的性能,本次实验利用工程中自带的程序来根据实验需求产生不同速率的数据包,从而模拟不同的场景,例如具有恒定流量的场景和具有突发流量的场景。

此外,鉴于 MSF 是 6TiSCH 工作组提出的首个全功能标准化调度算法^[22],本文也对 PID 算法和 MSF 算法在分布式网络下对阶跃型和脉冲型突发流量所表现出的性能进行对比分析。

根据第 2 章中所提的参数整定方法以及多次实验结果对比,在本次的对比实验中将 PID 控制器的各项系数 K_p, K_i, K_d 分别设置为 $K_p = 0.9, K_i = 0.072, K_d = 0.01$ 。在输出阈值方面,将增加一个 cell 的条件设置为:当 u_t 增加到 1 以上才能对 cell 进行增加操作;减少一个 cell 的条件设置为:当 u_t 减少到 -0.7 以下才能对 cell 进行减少操作。

相关实验数据在每次进行 cell 调度时获得,在完成一次调度后,系统会将当前时隙帧的 cell 信息通过串口打印函数输出,包括增加或减少 cell 的数量,当前已经发包数、绝对时隙值和当前时隙帧中分配的 cell 数量和已使用的 cell 数量。

3.2 阶跃型突发流量模式下测试验证

阶跃型突发流量模式下,本次实验控制发包速率为每个时隙帧一个包,并且在阶跃突变后一直维持在该速度,通过对比试验比较两种算法对于该流量的适应情况。时隙帧个数和时隙持续时间分别设置为 101 和 20 ms。将 MSF 算法每次评估使用率的预定值 N 设置为 32。

如图 3 所示,两种算法在应对每个时隙帧 1 个包的发包速率下的响应以及 cell 的使用率。可以明显看出面向突发流量的 PID 算法在短时间内迅速增加 cell 以此降低 cell 的使用率至稳定的范围。而 MSF 算法则是在经历了大约 30 多个时隙帧之后进行使用率评估才进行 cell 的增加。使得在此期间 cell 的使用率一直维持在非常高的水平,理论上 MSF 算法在未增加到第 2 个 cell 之前使用率一直维持在 100%,根据结果分析原因应该是因为第 1 个使用的 cell 在时隙帧内未被统计,从第 2 个时隙帧才开始统计,这也是 MSF 算法中 cell 的使用率从 0 开始增长的原因。这也反映出 MSF 算法的另一弊端,在设定预定值 N 时,设定值太小会导致 cell 的使用率误差较大,设定值过大就会导致系统响应过慢,面对突发数据流量无法快速响应。

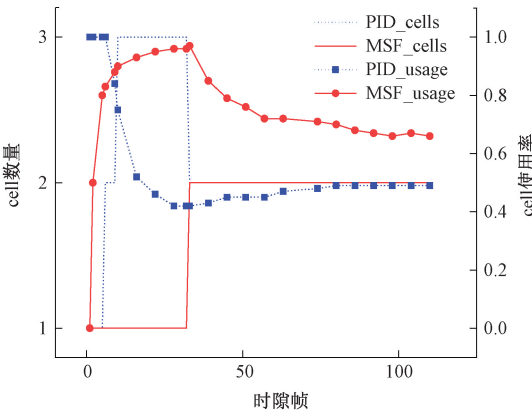


图 3 不同算法对阶跃型突发流量的响应

Fig. 3 Responses to step burst traffic by different algorithms

3.3 脉冲型突发流量模式下测试验证

为了对比所提调度与 MSF 算法在脉冲型突发流量模式下的性能。本次对比试验通过控制发包程序实现不同的发包速率,在第 30 个时隙帧时,发包速率由开始的每 3 个时隙帧 1 个包增加到每个时隙帧 1 个包,并保持该速率持续 20 个时隙帧,在第 50 个时隙帧时流量重新降为每 3 个时隙帧 1 个包。

图 4 展现了面向突发流量的 PID 算法与 MSF 算法在突发流量下的响应效果。在经历了 4 个时隙帧之后,也就是 PID 算法的一次评估周期的时间,PID 输出结果判定此时需要增加 cell,在第 35 个时隙帧时做出了进行

增加 cell 的操作,在第 36 个时隙帧时已有两个 cell。而原有的 MSF 算法由于其较长的评估周期导致它无法短时间内识别出这一流量波动,在流量波峰过后的第 34 个时隙帧才检测到此次突发流量。在突发流量过去后,面向突发流量的 PID 算法也能在 4 个时隙帧内迅速调控 cell,减少至 1 个,防止通信资源的浪费。MSF 算法依旧是需要等到经历固定的预设值 N 个 cell,才会进行下一次评估。

进一步,从 MSF 面对突发流量的响应可以看出,只有当 NumCellsElapsed 的值达到固定预设值 N 时才会进行 cell 使用率的评估,评估完成后才会执行 cell 的增加或者删除指令,这就会导致在遇到较大数据流量时,MSF 无法快速地调控合适数量的 cell 进行通信;同时,在流量趋于平稳后,无法及时回收 cell,造成通信资源浪费的问题。

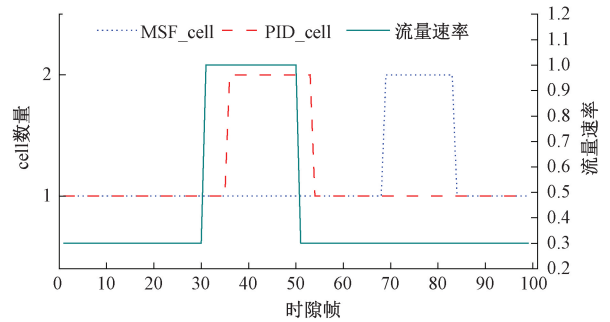


图 4 不同算法对脉冲型突发流量的响应

Fig. 4 Responses to pulse burst traffic by different algorithms

4 结 论

本文提出一种突发流量自适应的 6TiSCH 网络分布式 PID 调度算法,并进行试验验证,得到如下结论:

1) 将分布式无线网络节点间通信资源调控抽象为控制问题,将控制理论用于工业无线网络的资源调度应用,为分布式无线网络资源调度提供新思路,也为其他领域应用中需要进行网络调度以及控制资源使用率的场景提供一种新的解决方案参考;

2) 提出基于 PID 的 6TiSCH 网络分布式资源调度算法,以通信资源使用率为被控对象,通过周期性对 cell 使用率和 cell 需求数进行评估、以及对任意相邻两次调度间时间间隔的一致性控制,在 6TiSCH 网络中的 6top 层与 6P 协议协同实现分布式节点间 cell 数量的自适应调控;

3) 搭建了基于 OpenMote-B 硬件的 6TiSCH 网络试验平台,在阶跃型突发流量和脉冲型突发流量两种流量模

式下对所提算法进行了测试分析,结果表明所提分布式PID资源调度算法对于多种类型突发流量具有良好的自适应性。

参考文献

- [1] ALI J, ZAFAR M H, HEWAGE C, et al. The advents of ubiquitous computing in the development of smart cities—A review on the Internet of Things (IoT) [J]. *Electronics*, 2023, 12(4): 1032.
- [2] KALITA A, KHATUA M. 6TiSCH-IPv6 enabled open stack IoT network formation: A review [J]. *ACM Transactions on Internet of Things*, 2022, 3(3): 1-36.
- [3] HA Y, CHUNG S H. Traffic-aware 6TiSCH routing method for IIoT wireless networks [J]. *IEEE Internet of Things Journal*, 2022, 9(22): 22709-22722.
- [4] KHARB S, SINGHROVA A. A survey on network formation and scheduling algorithms for time slotted channel hopping in industrial networks-ScienceDirect [J]. *Journal of Network and Computer Applications*, 2019, 126:59-87.
- [5] JUNIOR I F V, GRANJAL J, CURADO M. A distributed network-aware TSCH scheduling [C]. 2023 19th International Conference on the Design of Reliable Communication Networks (DRCN). *IEEE*, 2023: 1-8.
- [6] PALATTELLA M R, WATTEYNE T, WANG Q, et al. On-the-fly band-width reservation for 6TiSCH wireless industrial networks [J]. *IEEE Sensors Journal*, 2015, 16(2): 550-560.
- [7] DUJOVNE D, GRIECO L A, PALATTELLA M R, et al. 6TiSCH 6top scheduling function zero/experimental (SFX) [R]. New York: Internet Engineering Task Force, 2017.
- [8] ANAMALAMUDI S, ZHANG M, SANGI A, et al. Scheduling function one (SF1) for hop-by-hop scheduling in 6TiSCH networks [J]. *Internet Engineering Task Force (IETF)*, 2017.
- [9] CHANG T, VUČINIĆ M, VILAJOSANA X, et al. RFC 9033. 6TiSCH Minimal Scheduling Function (MSF) [S]. CA, USA: RFC Editor, 2021.
- [10] 刘洋,行鸿彦,侯天浩. 四旋翼无人机串级变论域模糊PID姿态控制研究 [J]. *电子测量与仪器学报*, 2019, 33(10):46-52.
- LIU Y, XING H Y, HOU T H. Research on attitude control of cascade variable universe fuzzy PID for quadrotor UAV [J]. *Journal of Electronic Measurement and Instrumentation*, 2019, 33(10):46-52.
- [11] DOMINGO-PRIETO M, CHANG T, VILAJOSANA X, et al. Distributed PID-based scheduling for 6TiSCH networks [J]. *IEEE Communications Letters*, 2016, 20(5):1006-1009.
- [12] KORI G S, KAKKASAGERI M S. Agent driven resource scheduling in wireless sensor networks: Fuzzy approach [J]. *International Journal of Information Technology*, 2022, 14(1): 345-358.
- [13] 王頊,段斯静,黄庆卿,等. 工业物联网确定性调度中TDMA紧时隙时间精度边界可靠性分析 [J]. *仪器仪表学报*, 2018, 39(6):120-131.
- WANG T, DUAN S J, HUANG Q Q, et al. Reliability analysis of time precision boundary for tight slots of TDMA in deterministic scheduling of industrial internet of things [J]. *Chinese Journal of Scientific Instrument*, 2018, 39(6):120-131.
- [14] VILAJOSANA X, WATTEYNE T, CHANG T, et al. IETF 6TiSCH: A tutorial [J]. *IEEE Communications Surveys & Tutorials*, 2020, 22(1):595-615.
- [15] ELSTS A. TSCH-Sim: Scaling up simulations of TSCH and 6TiSCH networks [J]. *Sensors*, 2020, 20(19): 5663.
- [16] 杨伟,汪浩,万亚东,等. IETF 6TiSCH 工业物联网研究综述:标准、关键技术与平台 [J]. *计算机科学与探索*, 2020, 14(3):361-376.
- YANG W, WANG H, WAN Y D, et al. Survey on IETF 6TiSCH industrial Internet of things: Standards, key technologies and platforms [J]. *Journal of Frontiers of Computer Science and Technology*, 2020, 14(3): 361-376.
- [17] KALITA A, KHATUA M. 6TiSCH-IPv6 enabled open stack IoT network formation: A review [J]. *ACM Transactions on Internet of Things*, 2022, 3(3): 1-36.
- [18] FARAG H, GRIMALDI S, GIDLUND M, et al. REA-6TiSCH: Reliable emergency-aware communication scheme for 6TiSCH networks [J]. *IEEE Internet of Things Journal*, 2020, 8(3): 1871-1882.
- [19] RIGHETTI F, VALLATI C, RASLA D, et al. Investigating the CoAP congestion control strategies for 6TiSCH-based IoT networks [J]. *IEEE Access*, 2023, 11: 11054-11065.
- [20] 陈超波,胡海涛,高嵩. 人工蜂群的分阶PID控制器参数自适应研究 [J]. *控制工程*, 2020, 27(6): 956-961.

CHEN CH B, HU H T, GAO S. Parameters adaptive design of fractional order PID controller based on artificial bee colony algorithm[J]. Control Engineering of China, 2020, 27(6): 956-961.

- [21] GALLEGOS RAMONET A, NOGUCHI T. Performance analysis of IEEE 802.15.4 bootstrap process [J]. Electronics, 2022, 11(24): 4090.
- [22] TANAKA Y, MINET P, VUČINIĆ M, et al. YSF: A 6TiSCH scheduling function minimizing latency of data gathering in IIoT[J]. IEEE Internet of Things Journal, 2021, 9(11): 8607-8615.

作者简介



张焱(通信作者), 2011年和2016年分别于重庆大学获得学士学位和博士学位, 现为重庆邮电大学副教授, 主要研究方向为机电装备智能诊断与安全服役、工业物联网。

E-mail: yanzhang@cqupt.edu.cn

Zhang Yan (Corresponding author)

received his B. Sc. and Ph. D. degrees both from Chongqing University in 2011 and 2016, respectively. Now he is an associate professor at Chongqing University of Posts and Telecommunications. His main research interests include equipment intelligent diagnosis and safety management, and industrial Internet of Things.