

# 面向时间敏感网络的多目标联合调度优化算法

徐艳丽, 宋凯, 孔洲

(上海海事大学信息工程学院, 上海 201306)

**摘要:** 针对时间敏感网络在联合路由与调度中存在的端到端时延、抖动和带宽利用率等性能指标优化不足、扩展性差等问题, 提出一种基于改进的非支配排序遗传算法的多目标联合调度优化算法。该算法建立了以 TSN 性能指标为优化目标的多目标网络架构, 全面分析流量的可行路径, 找到全局最优的多目标优化方案。实验结果表明, 所提算法在大规模复杂场景下显著改善了流量的端到端时延、抖动及带宽利用率综合性能。

**关键词:** 时间敏感网络; 非支配排序遗传算法; 多目标优化; 联合路由与调度

中图分类号: TP393

文献标志码: A

## The Multi-Objective Joint Scheduling Optimization Algorithm for Time-Sensitive Networks

Xu Yanli, Song Kai, Kong Zhou

College of Information Engineering, Shanghai Maritime University, Shanghai 201306, China

**Abstract:** To address the inadequate optimization of performance indicators, such as end-to-end delay, jitter, and bandwidth utilization, as well as the poor scalability in the joint routing and scheduling of Time-Sensitive Networking (TSN), a multi-objective joint scheduling optimization algorithm based on an improved Non-dominated Sorting Genetic Algorithm was proposed. Specifically, a multi-objective network architecture utilizing TSN performance indicators as optimization objectives was established, feasible traffic paths were comprehensively analyzed, and the global optimal multi-objective solution was identified. Furthermore, the effectiveness of the proposed algorithm was evaluated through extensive experiments. The proposed algorithm significantly improves the comprehensive traffic performance in terms of end-to-end delay, jitter, and bandwidth utilization in large-scale complex scenarios.

**Keywords:** Time-Sensitive Networking, Non-dominated Sorting Genetic Algorithm, multi-objective optimization, joint routing and scheduling

### 0 引言

随着工业互联网的快速发展, 越来越多的新兴应用对网络提出确定性传输需求。一些典型用例包括工业自动化、自动驾驶等。时间敏感网络

(time-sensitive networking, TSN) 被认为是应对这一挑战的关键技术<sup>[1]</sup>。TSN 提供了有界低时延、低抖动, 以及高可靠的传输保证<sup>[2]</sup>, 并通过多种调度机制保障关键业务的服务质量 (quality of service, QoS), 调度算法可依据不同应用场景选择适当策

收稿日期: 2026-02-02; 修回日期: 2026-04-29

通信作者: 徐艳丽, ylxu@shmtu.edu.cn

基金项目: 国家自然科学基金资助项目(No.62271303); 国家自然科学基金联合重点项目(U25A20399); 上海市教育委员会创新计划项目(No.2021-01-07-00-10-E00121); 上海市科学技术委员会关键技术研发项目(25DZ3102300)

**Foundation Items:** The National Natural Science Foundation of China (No.62271303), The Joint Key Project of the National Natural Science Foundation of China (No.U25A20399), The Innovation Program Project of Shanghai Municipal Education Commission (No.2021-01-07-00-10-E00121), The Key Technology Research and Development Project of Shanghai Municipal Science and Technology Commission (No.25DZ3102300)

略<sup>[3]</sup>。然而,研究多聚集于流量调度,忽视了路由选择对 TSN 传输性能的影响,从而导致 TSN 流量传输性能下降<sup>[4]</sup>。因此,流量调度与路由优化结合成为提升网络性能的关键<sup>[5]</sup>。

联合路由与调度优化已成为 TSN 网络领域的主要研究重点<sup>[6]</sup>,许多研究致力于通过不同算法提高网络性能。聂宏蕊等等<sup>[7]</sup>针对通用时间感知整形器(time-aware shaper, TAS)复杂门控规划问题,提出了基于自适应的队列与硬件调度时隙建立混合整数线性规划的路由与调度模型,实现调度成功率提高。Min 等<sup>[8]</sup>针对 TSN 中的容错路由和调度策略,提出了能够提高网络可靠性和实时性的联合优化方法。Li<sup>[9]</sup>以降低大规模网络架构下音视频桥接(audio video bridging, AVB)流量调度的端到端时延为目标,提出了 TSN 和 AVB 联合的路由调度算法,显著提升了整体网络的性能。Wang 等<sup>[10]</sup>提出了一种面向大规模应用的联合路由与群调度算法,显著提升了调度效率。Wang 等<sup>[11]</sup>提出了基于进化算法的 TSN 路由与调度算法,在大规模调度场景中实现更快的执行时间。Xu 等<sup>[12]</sup>基于可满足性模理论(satisfiability to dulo theories, SMT)设计了 TSN 的联合调度与路由方案,实现多跳和多队列网络中的确定性传输。Sun 等<sup>[13]</sup>提出基于改进灰狼优化算法的车载 TSN 联合路由与调度优化方法。

尽管现有研究在联合路由与调度算法方面取得了一定进展,但在复杂网络环境下,仍面临性能优化的挑战<sup>[14]</sup>。研究多采用传统算法对 TSN 的性能指标进行优化,传统算法虽然具有较高的准确度,但计算复杂度较高,且往往针对单一 TSN 性能指标如端到端时延进行优化,不适用于大规模复杂场景以及多目标优化。Zhang 等人以及何倩等人<sup>[15-16]</sup>提出了基于无等待调度的联合算法,以优化资源利用率和满足实时性需求。Li 等人及 Nayak 等人<sup>[17-18]</sup>通过优化模型和启发式方法减少非确定性排队时延,从而提高了实时性和资源利用率。Pahlevan 等<sup>[19]</sup>提出的多队列调度算法可灵活配置时间触发(time-triggered, TT)流的抖动模式,Falk 等<sup>[20]</sup>提出了一种基于冲突图的时间触发流量规划方法,用于解决分布式系统中的实时通信问题。Gu 等<sup>[21]</sup>提出了基于改进的非支配排序遗传算法(non-dominated sorting genetic algorithm II, NSGA-II)的多目标流量调度模型,降低了流量的端到端时延,显著

提升了调度性能。Reusch 等<sup>[22]</sup>采用约束规划和模拟退火(simulated annealing, SA)来提升网络的可靠性和安全性,SA 在处理大规模网络时表现出良好的性能。Sun 等人<sup>[23]</sup>所提算法融合动态基因表达调整机制(dynamic gene expression regulation, DGER),通过动态控制遗传算子来实现全局最优搜索和增加种群多样性。

此外,在实际应用中,TSN 面临复杂的网络拓扑和多样化的流量模式。尤其是面对混合流量时,混合调度需确保不同流量在共享网络中互不干扰,同时满足各自的端到端时延和抖动需求<sup>[24]</sup>。Deng 等人<sup>[25]</sup>提出的增强调度方法提升了 AVB 流性能并保障 TT 流实时性。王颖等人<sup>[26]</sup>提出了一种基于时间感知和循环排队转发的时间敏感网络联合路由和混合流量调度算法,有效解决 TSN 混合流量调度问题,提高可调度率和带宽利用率。

综上,现有研究在 TSN 联合路由与调度优化方面仍存在以下三点主要不足:多目标协同优化能力不足。现有研究大多以单一性能指标为优化目标,难以在端到端时延、抖动和带宽利用率等之间实现协同优化,无法满足复杂 TSN 场景下的多重性能需求;算法扩展性与全局搜索能力受限。传统 SMT 和启发式方法,在大规模网络拓扑下计算复杂度急剧上升,易陷入局部最优,难以适应大规模复杂场景;混合流量场景下的调度适应性不足。现有算法对混合流量(TT 流与 AVB 流混合场景)的联合调度支持有限,难以同时保障不同流量类型的可调度率。

针对上述三点不足,本文的主要工作如下:

1)提出了基于 NSGA-II 的多目标联合调度优化算法,实现了对 TSN 中流量的端到端时延、抖动和带宽利用率的综合优化,以满足大规模复杂网络环境中的多重性能优化需求。

2)算法确保在优化过程中覆盖不同流量的调度可能性,并引入了 SA 和 DGER,增强了算法的扩展性和全局优化能力。

3)在不同流量负载和网络拓扑下对所提算法进行了系统的实验验证,结果表明该算法在优化端到端时延、抖动和带宽利用率方面表现优异。

## 1 系统模型

表 1 总结了全文中使用的符号。

表1 符号说明表

符号	意义
$G(E, V)$	TSN网络的拓扑模型, 包含节点集合 $E$ 和顶点集合 $V$
$L$	链路总数, 表示网络中可用的链路数量
$f_i$	表示系统中第 $i$ 个流量
$r_i$	第 $i$ 条流量 $f_i$ 的传输路径
$R$	表示网络中所有可能的路径集合
$F$	系统中所有流量的集合
$T$	流量的周期
$D$	流量最大截止时间
$S$	流量大小
$v_s, v_t$	发送终端和接收终端系统
$l_i$	第 $i$ 条链路
$h_i$	链路周期
$\Phi_i[Va, Vb]$	表示流量 $i$ 在链路 $[Va, Vb]$ 上的传输起始时间
$T_i[Va, Vb]$	链路 $[Va, Vb]$ 上流量的周期
$l_i[Va, Vb]$	流量 $i$ 在链路 $[Va, Vb]$ 上的持续传输时间
$I_i[Va, Vb]$	表示流量 $i$ 在链路上的实际传输时间
$Q_i$	排队时延
$D_{f_i}$	总传输时延, 衡量所有流的端到端时延之和
$d_i$	第 $i$ 个流的端到端时延
$J$	总抖动, 用于衡量所有流量传输的稳定性
$U$	带宽利用率, 表示网络资源的使用效率
$\Gamma$	AHP综合调度目标函数, 用于评价联合优化的效果
$R_{sched}$	流量可调度性

### 1.1 网络拓扑

TSN网络拓扑如图1所示, 模型由终端系统 $ES$ 、交换机 $SW$ 和链路组成。将TSN网络的拓扑建模为有向图 $G = (E, V)$ , 顶点集合 $V$ 由网络中的所有设备集合(终端系统 $ES$ 和交换机 $SW$ 的集合)构成。 $E$ 表示设备之间的物理连接。从节点 $a$ 到节点 $b$ 的数据链路表示为 $[V_a, V_b]$ , 链路为全双工链路。每条链路 $[V_a, V_b]$ 包含带宽容量、传播时延以及链路出口端口的相关信息。

图1中, 网络拓扑由两个交换机和四个端系统组成。用 $r_i (i = 1, 2, \dots, n)$ 表示流量 $f_i$ 的传输路径。虚线箭头指向表示表示两条数据路径 $r_1$ 和 $r_2$ , 可以用链路 $[V_a, V_b]$ 表示为

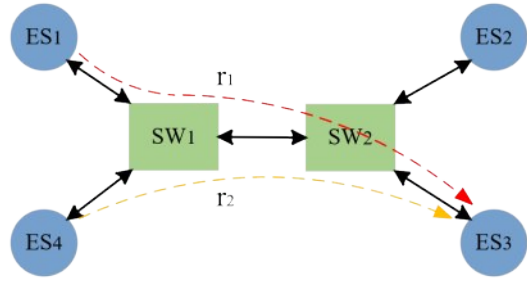


图1 TSN网络拓扑图

$$r_1 = \{ [ES_1, SW_1], [SW_1, SW_2], [SW_2, ES_3] \} r_2 = \{ [ES_4, SW_1], [SW_1, SW_2], [SW_2, ES_3] \}$$

为了显示传输路径, 这里使用路由 $R$ 表示网络中所有路由的路径集合, 即 $R = \{ r_1, r_2, \dots, r_n \}$ 。

### 1.2 流量模型

流量集合记为 $F = F^{TT} \cup F^{AVB}$ ,  $F^{TT}$ 表示TT流的流量集合,  $F^{AVB}$ 表示AVB流的集合。每类流 $f_i$ 具有以下属性 $(T, D, S, v_s, v_t)$ 。

当流量的数据大小超过一定大小的时候, 需要将流量数据划分为多帧处理后才能进行传输。为确保传输效率, 单个以太网帧最多传输1500字节(byte, B), 即所谓的一个最大传输单元(maximum transmission unit, MTU)大小。在调度时, 流量 $f_i$ 的第 $k$ 帧 $f_i^k$ 会根据其目标路由进行分发, 流量的源和目的地址根据网络拓扑和链路选择进行调整, 在选择的路径上对流量进行约束调度。

### 1.3 调度约束

本文考虑TSN网络中存在的两类流量: TT流, AVB流。为实现混合流量在共享链路上的无干扰传输, 本文分别采用IEEE 802.1Qbv定义的TAS和IEEE 802.1Qav定义的基于信用的整形器(credit-based shaper, CBS)对两类流量进行差异化调度管控。对于TT流, TAS在每个交换机出口端口维护一张门控列表, 以所有TT流周期的最小公倍数为超周期, 按时间槽精确控制各优先级队列的开关状态。TT流帧须在预先规划的时隙内完成发送, 且同一端口上任意两条TT流的发送时隙不得重叠, 从而在时间维度上消除帧碰撞与排队不确定性。对于AVB流, CBS通过信用累积与消耗机制对出口带宽进行动态限速。仅当信用值非负时允许帧进入发送。CBS将AVB流的最大可用带宽限制在链路容量的预留比例以内, 防止AVB流突发占用TT流

所需的确定性时隙资源，保障两类流量的隔离共存。

在上述双机制协同调度框架下，本文将调度问题建模为一组约束，以确保所有流量在满足无冲突时隙分配和带宽限速的前提下，实现端到端时延、抖动和带宽利用率的联合优化。具体约束如下所示。

#### (1) 基本约束

为了确保每一帧的传输时间都在预定的时间窗口内完成，保证传输的确定性，需要保证以下条件：

$$0 \leq \phi_i^{[V_a, V_b]} \leq T_i^{[V_a, V_b]} - l_i^{[V_a, V_b]}, (\forall f_i \in F) \quad (1)$$

**截止时间限制：**周期性流量的发送时间不能早于其规定的释放时间，并且必须在截止时间之前到达接收端。

**传输持续时间：**每条链路上的流量传输时间由链路带宽和流量大小决定。

#### (2) 时延约束

到达目的地的流量，其传输结束时间不能超过设定的最大时延  $d_i$ ：

$$\phi_i^{[V_{n-1}, V_n]} + I_i^{[V_{n-1}, V_n]} + Z^{[V_{n-1}, V_n]} - \phi_i^{[V_1, V_2]} \leq d_i, \quad (2)$$

$$r_i = \{ [V_1, V_2], \dots, [V_{n-1}, V_n] \}$$

公式用于确保流在网络中的传输时延满足时延约束，保证实时性要求。

#### (3) 帧隔离约束

为了确保帧的隔离，以避免多个帧在同一个队列中造成时延，公式为：

$$\phi_i^{[V_a, V_b]} + \alpha T_i^{[V_a, V_b]} + \delta \leq \phi_j^{[V_y, V_a]} + \beta T_j^{[V_a, V_b]} + Z^{[V_y, V_a]} + q_t \quad (3)$$

$$\phi_j^{[V_a, V_b]} + \alpha T_i^{[V_a, V_b]} + \delta \leq \phi_i^{[V_x, V_a]} + \beta T_i^{[V_a, V_b]} + Z^{[V_x, V_a]} + q_t \quad (4)$$

其中， $q_t$  表示排队时延，公式(3)和(4)确保在同一个队列中的不同流不会产生过多的时延，避免链路中断或队列溢出。

#### (4) 链路无冲突传输约束

为了确保多条数据流同时经过同一链路时不产生冲突，链路需要保证以下条件：

$$\phi_i^{[V_a, V_b]} + \alpha T_i^{[V_a, V_b]} \geq \phi_j^{[V_a, V_b]} + \beta T_j^{[V_a, V_b]} + l_j^{[V_a, V_b]} \quad (5)$$

$$\phi_j^{[V_a, V_b]} + \beta T_j^{[V_a, V_b]} \geq \phi_i^{[V_a, V_b]} + \alpha T_i^{[V_a, V_b]} + l_i^{[V_a, V_b]} \quad (6)$$

$$\forall \alpha \in \{0, 1, \dots, (h_i/T_i^{[V_a, V_b]} - 1)\}, \quad (7)$$

$$(\forall f_i, f_j \in F | i \neq j)$$

$$\forall \alpha \in \{0, 1, \dots, (h_i/T_i^{[V_a, V_b]} - 1)\}, \quad (8)$$

$$\forall \beta \in \{0, 1, \dots, (h_j/T_j^{[V_a, V_b]} - 1)\}, \quad (8)$$

$$(\forall f_i, f_j \in F | i \neq j)$$

公式(5)和(6)确保两条流量在同一链路的传输时间必须前后衔接，公式(7)和(8)确保同一链路同一时刻不能有两非零持续时间的流量传输。

#### (5) 流量传输约束

流在经过多个节点时的传输时间，确保流量在每条链路上的传输时间合理分配。在传输时，各流的时间必须满足以下要求：

$$\phi_i^{[V_x, V_b]} \geq \phi_j^{[V_a, V_x]} + l_j^{[V_a, V_b]} + Z^{[V_a, V_x]} + \delta, \quad (9)$$

$$([V_a, V_x], [V_x, V_b] \in r_i)$$

其中， $\delta$  表示节点处理时延。 $Z^{[V_a, V_x]}$  表示链路  $[V_a, V_x]$  传播时延。公式确保跨节点传输的时间逻辑连贯，避免调度混乱。

### 1.4 调度目标

#### (1) 最小化端到端时延

端到端时延衡量流量从源节点到目的节点的传输时间，本文目标是 minimized 所有流量的总端到端传输时延，以提高流量的传输效率和实时性。目标公式为：

$$D_f = \sum_{i=1}^N d_i \quad (10)$$

其中， $d_i$  表示第  $i$  个流的端到端时延， $N$  为总流量数。通过最小化  $D_f$  来优化网络整体的时延性能。

#### (2) 最小化抖动

抖动用于衡量流量传输的稳定性，抖动过大会导致关键业务质量下降以及实时控制应用的误动作。为了更具体地量化抖动，引入了抖动的标准差作为衡量指标。抖动的公式为：

$$J = \sum_{i=1}^N \sqrt{\frac{1}{K_i} \sum_{k=1}^{K_i} (d_k^i - \bar{d}^i)^2} \quad (11)$$

其中,  $d_k^i$  是第  $i$  个流的第  $k$  个数据包的到达时间,  $\bar{d}^i$  是该流中所有数据包的平均到达时间,  $K_i$  为数据包的数量。

### (3) 最大化带宽利用率

带宽利用率目标是在满足 TT 流严格的时延和抖动要求的前提下, 尽量提高带宽利用率, 使网络资源得到充分利用。其公式为:

$$U = \frac{1}{L} \sum_{l=1}^L \frac{B_l^u}{B_l^t} \quad (12)$$

其中,  $B_l^u$  表示第  $l$  条链路的已用带宽,  $B_l^t$  表示第  $l$  条链路的总带宽。

对三个性能指标进行综合优化分析, 使用层次分析法 (analytic hierarchy process, AHP) 计算三个目标的权重, 加权融和来构建综合目标函数。

$$\Gamma = \omega_1 D_f + \omega_2 J + \omega_3 U \quad (13)$$

$\omega_1, \omega_2, \omega_3$  分别为端到端时延、抖动和带宽利用率的权重, 反映了各目标在整体优化中的重要性。在车载 TSN 场景中, 确定性低时延是核心传输保障, 端到端时延对实时控制系统的影响最为直接; 抖动过大将导致周期性控制指令的时序错乱, 其重要性次于时延; 带宽利用率则在满足前两项约束的前提下进行最大化优化。依据上述优先级关系, 代码构造三阶成对比较矩阵, 经归一化计算得到权重向量:

$$(w_1, w_2, w_3) \approx (0.648, 0.230, 0.122)$$

一致性比  $CR < 0.1$ , 满足一致性要求。

上述权重反映了工业自动化这一典型 TSN 部署场景的业务特征。在以 AVB 流为主的多媒体传输场景中, 抖动对用户体验的影响更为显著, 此时可适当提升抖动的权重; 而在以大流量数据交换为主的场景下, 带宽利用率的权重则应相应增大。因此, 本文所建立的 AHP 权重框架具备良好的场景适应性, 实际部署时可依据业务需求重新构造成对比较矩阵, 以获得符合特定场景的最优权重配置。

本文在实验评估中采用 5:3:2 的简化权重比例, 与 AHP 理论权重在优先级排序上保持一致 (端到端时延 > 抖动 > 带宽), 简化权重便于不同算法间的横向对比, 同时降低权重精度对实验结论的干扰。

## 2 基于 NSGA-II 的多目标联合调度优化算法

本节中, 针对大规模复杂 TSN 网络场景中的端到端时延、抖动、带宽利用率等性能指标优化不足、扩展性差等问题, 将其近似为多目标联合路由与调度优化问题, 本文提出一种基于 NSGA-II 的多目标联合调度优化算法来解决这一问题。为了能够找到全局最优解, 算法将流量的路径分配方案作为个体进行调度优化, 在每一轮迭代中, 对选中的个体进行调度优化和适应度评估, 确保不断筛选出优质解, 最大程度地提高了找到全局最优解的可能性。算法设计包含以下主要步骤: 初始化与个体组成、适应度函数设计与评估、交叉与变异操作, 以及 Pareto 最优解的寻找。

在种群初始化中, 首先定义了网络拓扑, 为每个流量生成可行路径集合假设网络拓扑包含多个 NS 和 SW, 网络中有多个流量  $f_1, f_2, \dots, f_n$  等。对于每个流量  $f_i$ , 通过路径搜索算法为其生成若干可行路径, 形成初始种群的个体。每个个体包含一组路径, 代表不同流量的传输方案, 即  $I = \{P_{1k_1}, P_{2k_2}, \dots, P_{nk_n}\}$ 。其中,  $P_{ik_i}$  表示流量  $f_i$  的第  $k_i$  条路径。如图 2 所示, 三条流量  $f_1, f_2, f_3$  的可行路径集合。其中, 流量  $f_1$  从 NS1 到 NS5 有两条可行路径  $P_{11}, P_{12}$ , 流量  $f_2$  从 NS2 到 NS2 有三条可行路径  $P_{21}, P_{22}, P_{23}$ , 流量  $f_3$  从 NS3 到 NS4 有三条可行路径  $P_{31}, P_{32}, P_{33}$ 。通过从每个流量的可行路径集合中随机选择路径组合生成了初始种群, 例如,  $I_A = \{P_{11}, P_{21}, P_{31}\}$ ,  $I_B = \{P_{11}, P_{22}, P_{33}\}$ ,  $I_C = \{P_{12}, P_{23}, P_{32}\}$ ,  $I_D = \{P_{12}, P_{21}, P_{32}\}$ 。随着算法的迭代, 种群通过选择、交叉和变异操作生成新个体, 并结合非支配排序选择出下一代种群。

每一代中, 个体会根据适应度进行选择。在适应度评估前, 需要对每个选中的个体进行调度优化。通过对个体内的路径分配方案进行进一步调整, 确保满足网络中的各种约束条件。接着, 从种群中随机选择若干个体进行“锦标赛”选择和交叉操作生成新个体, 选择操作旨在保留适应度较高的个体, 交叉操作通过在两个父个体间交换部分路径信息来产生新个体。变异操作随机改变个体中的部分路径, 以引入新的探索空间, 从而增加种群的多样性。例如, 图 2 选中的两个父代个体  $I_A =$

$\{P_{11}, P_{21}, P_{31}\}$ ,  $I_B = \{P_{11}, P_{22}, P_{33}\}$ , 随机选择一个交叉点  $k$ , 交换  $k$  点后的路径信息, 生成两个子代个体  $I'_A = \{P_{11}, P_{21}, P_{33}\}$ ,

$I'_B = \{P_{11}, P_{22}, P_{31}\}$ 。通过交叉操作可以保持遗传多样性, 同时在搜索空间内探索新的解。交叉变异操作的概率在每一代中根据网络状态进行动态调整, 初期设置较高的概率以增加探索空间, 后期则降低概率以提高收敛性。精英保留策略则确保每代保留最优个体, 提高解的整体质量。

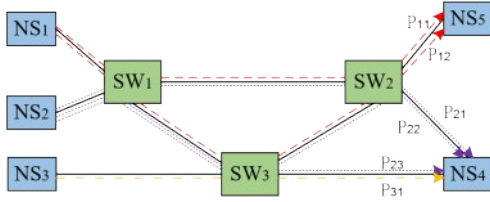


图2 网络与流量示意图

为适应复杂网络环境, 本文引入了 DGER 和 SA, 使调度方案能更好地适应不同流量类型的需求。算法 1 展示了本文算法的详细实现过程。

**算法 1** 基于 NSGA-II 的多目标联合调度优化算法

**输入:** 网络拓扑  $G(E, V)$ , 流集合  $F$ , 种群规模  $N$ , 当前迭代次数  $gen$ , 最大迭代次数  $N_{gen}$ , 初始温度  $T_0$ , 退火系数  $\alpha$ , 拥塞阈值  $U_{th}$ , 初始变异概率  $P_{mut}^{init}$

**输出:** Pareto 最优的联合路由与调度方案集

- 1) 初始化网络拓扑  $G(E, V)$  及流集合  $F$  中每条流的可行路径集合, 随机初始化种群  $P_0$
- 2) 对  $P_0$  中每个个体执行 **算法 2**, 计算  $D_{f_i}$ 、 $J$ 、 $U$ , 并进行非支配排序
- 3)  $T \leftarrow T_0$
- 4) **for**  $gen = 1$  **to**  $N_{gen}$  **do**
- 5) 更新当前代动态变异概率  $P_{mut}$
- 6) 使用锦标赛选择和多点交叉算子, 从  $P_{gen-1}$  生成子代种群  $Q_{gen}$
- 7) **DGER**
- 8) **for**  $Q_{gen}$  中的每个个体  $X$  **do**
- 9) 统计  $X$  在当前路由下的全网链路利用率  $\{U_{ab} | \forall l_{ab} \in E\}$
- 10) **for**  $X$  中的每条流量路径  $p_i$  **do**:

11) **if**  $rand() < P_{mut}$  且满足  $\exists l_{ab} \in p_i$  满足  $U_{ab} > U_{th}$  **then**,  $rand()$  表示  $(0, 1)$  区间均匀分布的随机生成函数

12) 流集合  $F$  中筛选出未拥塞路径集, 随机选择一条替换  $p_i$

13) **end if**

14) **end for**

15) **SA**

16) 随机选取一条并突变其路径, 生成邻域新个体  $X'$

17) 调用 **算法 2**, 计算适应度差值  $\Delta\Phi = \Phi(X') - \Phi(X)$

18) **if**  $\Delta\Phi < 0$  或  $rand() < \exp(-\Delta\Phi/T)$  **then**

19)  $X \leftarrow X'$ , 根据 Metropolis 准则接受新解

20) **end if**

21) **end for**

22) 调用 **算法 2** 更新  $Q_{gen}$  的适应度

23) 合并种群  $R_{gen} = P_{gen-1} \cup Q_{gen}$

24) 通过快速非支配排序与拥挤度计算, 从  $R_{gen}$  中选出  $N$  个个体作为下一代  $P_{gen}$

25)  $T \leftarrow T \times \alpha$ , 温度衰减

26) **end for**

27) **return** 最终种群  $P_{N_{gen}}$  的最优 Pareto 前沿集

最终算法生成了一组 Pareto 最优解集, 通过 AHP 方法进一步综合分析出全局最优解, 输入权重向量 (实验采用 5:3:2 权重) 确保多目标间达到合理平衡。

本文采用路由寻优与调度校验相解耦的联合优化架构, 遗传算法的外层负责探索全局最优的流量路径组合。而在内层适应度评估阶段, 算法将具体的路由方案代入调度约束公式中。通过计算各条链路上的流量重叠情况与带宽分配  $U$ , 推导出  $D_{f_i}$  和  $J$ 。某路由方案导致时隙冲突无法调度, 则对其施加极大的适应度惩罚。具体调度评估逻辑如 **算法 2**:

**算法 2** 调度评估算法

**输入:** 个体路由方案

$X = \{p_1, p_2, \dots, p_{|F|}\}$ , 网络拓扑  $G(E, V)$ , 流集合  $F$

**输出:**  $D_{f_i}$ 、 $J$ 、 $U$

1) 初始化各链路的已占用时隙集合  $S_{ab} = \emptyset$ ,

$D_{f_i} = 0, J = 0$

2) 根据公式(12)计算各链路的利用率, 之后求出  $U$

3) for  $\forall f_i \in F$  及其传输路径  $p_i$  do

4) for 路径  $p_i$  上的每一跳链路  $l_{ab}$  do

5) 基于最早可用时隙策略, 查找满足  $[\tau_i^{ab}, \tau_i^{ab} + c_i^{ab}] \cap \mathcal{S}_{ab} = \emptyset$  的无冲突传输起始时间  $\tau_i^{ab}$

6) 更新该链路的占用状态:  $\mathcal{S}_{ab} \leftarrow \mathcal{S}_{ab} \cup \{[\tau_i^{ab}, \tau_i^{ab} + c_i^{ab}]\}$ ,  $c_i^{ab}$  指持续传输时间

7) end for

8) 计算单条流量  $f_i$  的  $d_i$  和  $J_i$

9) if  $d_i > D$  (违背最大截止时间) then

10) 对该个体的适应度施加极大的惩罚项, 淘汰不可调度方案

11) end if

12) 根据公式(10)和公式(11)计算出  $D_{f_i}$  和  $J$

13) end for

14) return  $D_{f_i}, J, U$

### 3 仿真设计与结果分析

在本节中, 基于 NSGA-II 的多目标联合调度优化算法在 TSN 场景中的有效性进行了全面验证。仿真评估分为四个部分, 从不同角度展示算法的优化效果和适应性。在实验一中, 使用经典的多目标测试集验证本文提出的多目标联合调度优化算法在标准数据集上的表现, 证明该算法在多目标优化问题上的有效性和适应性。在实验二中, 将算法与几种现有解决方案进行对比, 评估本文算法在多种性能指标上的表现, 特别是端到端时延、抖动和带宽利用率的优化能力。在实验三中, 设计了针对 SA 和 DGER 机制的消融实验, 验证其对性能提升的有效性。为进一步验证设计的有效性, 在实验四中扩展了应用场景, 对混合流量场景下的调度能力进行了测试。在扩展实验中, 网络拓扑包含了多种类型的流量, 并考虑了不同流量负载情况, 以评估算法在复杂场景中的适应性。

本文基于 Python 语言搭建 TSN 仿真环境, 并运行在配备 Intel Core i9-12900H 处理器 (20 核)、32GB 内存的计算机上。网络拓扑结构包含若干交换机和终端系统, 代表了典型的 TSN 应用场景, 如图 3 所示。该拓扑由 17 个终端节点和多个交换节点构成, 链路带宽为 100 Mbps, 链路周期为 20ms。

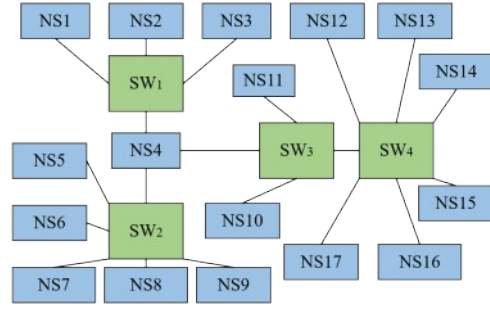


图3 模拟车载网络拓扑图

#### 实验一: 多目标优化有效性评估

在实验一中, 利用经典的多目标测试集对本文算法进行测试, 评估算法在处理复杂目标空间时的适应性和有效性。本节实验采用三个主要性能指标来衡量算法的表现: 超体积 (hypervolume, HV), 反向世代距离 (inverted generational distance, IGD) 以及世代距离 (generational distance, GD)。通过这些指标, 验证算法在优化多目标时的收敛性、逼近性和解的多样性。这些解集展示在优化端到端时延、抖动和带宽利用率三个目标之间的最优权衡。算法实验参数设置如下: 种群大小设置为 1200, 迭代次数 400。交叉概率初始值 0.7, 最小值 0.4, 变异概率初始值 0.3, 最小值 0.1。实验结果如图 4 所示。

如图 4 所示, HV 随着迭代次数的增加而逐渐上升并趋于稳定, 表明解集的覆盖性良好; IGD 和 GD 则随迭代显著下降, 证明解集逐步逼近 Pareto 前沿, 收敛效果优异。综上, 这些指标表明算法在解集质量、收敛性和多样性方面的表现均非常好。将算法在 DTLZ2 和 ZDT2 数据集上运行, 得到了一组 Pareto 前沿解集。

如图 5 所示二维 Pareto 前沿图展示了算法在标准数据集上的 Pareto 前沿分布情况。图中黑色圆形表示真实的 Pareto 前沿, 而灰色三角形表示算法获得的 Pareto 前沿解。可以看出, 蓝色点与红色点的分布非常接近, 表明算法生成的解集能够很好地逼近真实的 Pareto 前沿, 展示了算法在多目标优化问题上的解多样性和逼近性, 进一步验证了算法在标准测试集上的有效性。

图 6 三维 Pareto 前沿图进一步展示了算法生成的 Pareto 前沿解集与真实 Pareto 前沿的接近程度。黑色圆形表示真实的 Pareto 前沿, 灰色三角形为算

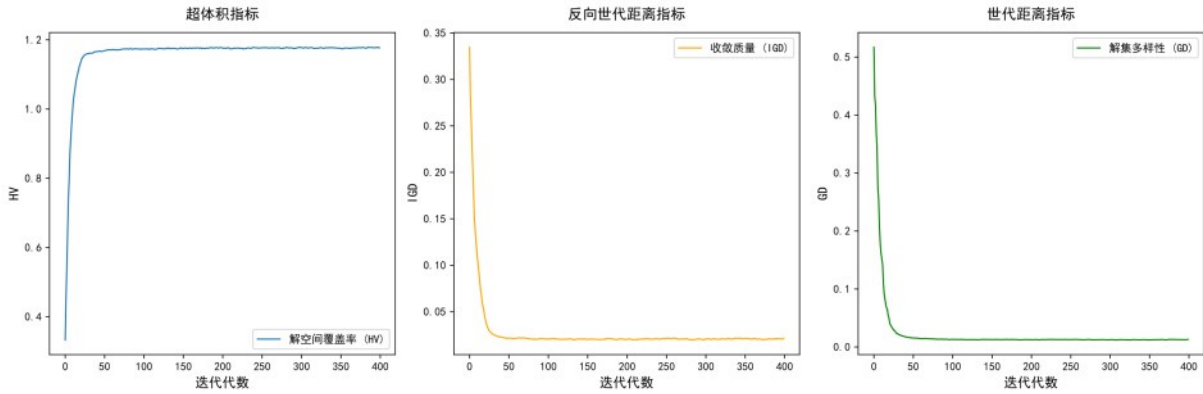


图4 性能指标变化图

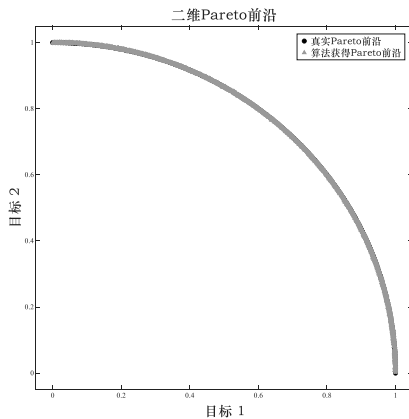


图5 二维Pareto前沿图

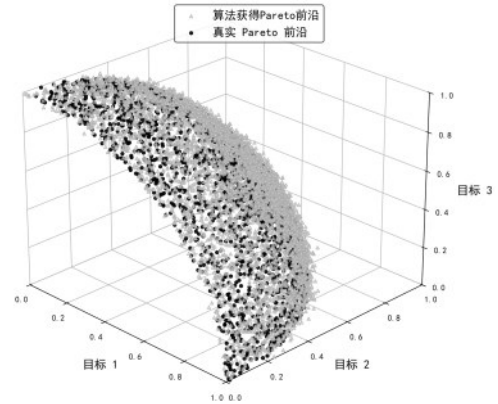


图6 三维 Pareto 前沿图

法获得的 Pareto 前沿。从图中可以看出，大部分三角形分布在圆形附近，表明算法生成的解集在各目标维度上能够较好地逼近真实的 Pareto 前沿，验证了算法的有效性和解集的逼近质量。同时，解集分布较为均匀，说明算法在解多样性上的表现也很出色。

通过实验一的测试，验证了本文提出的基于 NSGA-II 的多目标联合调度优化算法在经典多目标测试集上的有效性，展示了算法在解集质量、收敛性和多样性上的优势。

### 实验二：综合性能评估实验

实验二比较本文提出的算法与多种现有优化算法在 TSN 中的性能表现。选取了一个包含 25 条 TT 流的场景，针对端到端时延、抖动和带宽利用率这三个关键性能指标进行全面对比分析，流量信息具体如下表 2 所示。

通过比较不同算法在相同场景下的表现，验证本文算法在 TSN 网络性能上的优化能力。参选对比算法主要是 TT 流和混合流量场景下表现优异的

算法，具体包括 CG<sup>[20]</sup>、LS\_TB<sup>[27]</sup>、LS\_PL<sup>[28]</sup>、SMT<sup>[29]</sup>、CP<sup>[30]</sup>、DT<sup>[15]</sup>、GA<sup>[31]</sup> 算法。为确保实验的可靠性，对比郑等人实验<sup>[32]</sup>，所有算法的初始条件、网络拓扑、流量模型都保持一致，各个算法的核心参数设置如表 3 所示：

每种算法对流量进行路由规划和调度后，记录每条流量的传输时延、抖动以及整个网络的带宽利用率。每次实验进行了 10 次独立仿真，避免偶然因素影响结果，取各项指标的平均值作为最终结果，以评估算法的稳健性和优化效果。为全面和公平地评估各算法性能，使用 AHP 并具体分析，将权重设置为 5:3:2。实验结果如表 4 所示。

从表 4 可以看出，本文提出的算法在端到端时延方面表现出明显的优势，相较于其他算法，总端到端时延最小，为 3278.61 $\mu$ s。表明该联合路径规划与调度方法在减少传输时间方面具有更高的有效

表2 流量信息

Flow	vs	vt	S(Byte)	T(ms)	D( $\mu$ s)	Priority
1	ES1	ES4	125	20	20	4
2	ES14	ES3	125	50	20	3
3	ES15	ES5	125	200	60	3
4	ES16	ES4	750	200	60	3
5	ES17	ES6	750	200	60	3
6	ES5	ES7	750	500	60	2
7	ES6	ES8	750	500	50	2
8	ES7	ES4	750	500	50	3
9	ES8	ES9	365	500	50	2
10	ES9	ES4	365	50	50	3
11	ES1	ES2	365	500	50	5
12	ES1	ES4	365	200	20	2
13	ES1	ES11	625	200	20	5
14	ES13	ES4	250	200	20	1
15	ES2	ES4	250	200	20	1
16	ES2	ES3	250	100	100	1
17	ES2	ES5	250	100	10	1
18	ES2	ES6	250	100	10	1
19	ES2	ES7	125	100	10	1
20	ES4	ES8	500	100	10	3
21	ES4	ES9	125	100	10	3
22	ES12	ES2	125	100	20	3
23	ES12	ES2	600	200	20	3
24	ES11	ES13	250	200	20	3
25	ES12	ES14	625	150	100	3

表3 算法核心参数设置

算法	核心参数	参数设置
SMT	求解器引擎	Z3 Solver
	隔离级别	帧级隔离
CP	求解器引擎	IBM CP Optimizer
	并行工作数	4
DT	存在性变量	设置禁用
	搜索策略	Incremental
LS_TB	冲突检测	First-instance
	回溯机制	设置禁用
LS_PL	初始释放点	0
	回溯范围	局部
CG	全局冲突集	设置禁用
	搜索顺序	按链路依赖关系
GA	阶段划分	逐阶段
	求解器引擎	Gurobi ILP
Our	MIPGap	0.01
	模式切换	自动
SA	种群规模	1200
	最大迭代次数	400
DGER	交叉概率	0.8
	变异概率	0.1
SA	种群规模	1200
	最大迭代次数	400
SA	交叉概率	0.7~0.4
	变异概率	0.3~0.1
SA	初始温度	1000
	降温率	0.75
SA	终止温度	0.001
	指数衰减系数	3

性。在抖动方面, LS\_PL 算法表现稍优, 抖动值约为  $4.01\mu\text{s}$ , 而本文算法抖动值为  $4.15\mu\text{s}$ , 接近最优, 且优于大多数对比算法。在带宽利用率方面, 本文算法也表现出较好的均衡性, 带宽利用率为 71.36%, 相比其他算法具有更高的资源利用率。这表明本文算法在保证端到端时延和抖动性能的同时, 能够更好地利用网络资源。本文算法在三个性能指标上都展现出了较为突出的表现, 尤其在端到端时延和带宽利用率上具有明显优势, 本文提出的基于 NSGA-II 的联合路径规划与调度算法在 TSN 中具有显著的优化效果。

#### 实验三: 消融对比验证实验

为验证本文所提算法中 DGER 和模拟 SA 各自

对性能提升的贡献度, 本节设计了消融实验, 通过逐步移除上述改进模块, 对比分析各模块的有效性。实验场景在与实验二相同的网络拓扑和流量配置下进行, 共设置以下四种情况:

1) Case 1: NSGA-II, 交叉和变异概率参数设置与实验一保持一致, 不启用 SA, 变异操作采用随机路径替换。

2) Case 2: NSGA-II+SA, SA 的初始温度  $T_0 = 1000$ , 冷却系数  $\alpha = 0.75$ , 停止温度  $T_{stop} = 10^{-3}$ , 交叉和变异概率设置保持一致。

表4 不同权重下的综合性能比较(5:3:2)

	Our	CG	LS_TB	LS_PL	Smt	CP	DT	GA
端到端时延(μs)	3278.61	3989.53	3778.61	3460.75	3375.92	3341.42	4415.33	3937.54
抖动(μs)	4.15	4.65	4.23	4.01	4.12	5.42	5.95	5.08
带宽利用率(%)	71.36	62.21	64.36	56.61	54.33	58.23	65.12	61.82

3) Case 3: NSGA-II+DGER, 交叉概率和变异概率按指数退火方式动态调整, 变异操作采用拥塞感知路径选择策略, 不启用SA。

4) Case 4: Our, 本文设计的算法。

每次实验进行10次独立仿真, 避免偶然因素影响结果, 取各项指标的平均值作为最终结果, 以评估算法的稳健性和优化效果。对三项性能指标赋予了不同的权重, 权重设置为5:3:2。实验结果如表5所示。

表5 消融对比实验

	Case 1	Case 2	Case 3	Case 4
端到端时延(μs)	3652.47	3481.35	3418.92	3278.61
抖动(μs)	4.87	4.53	4.39	4.15
带宽利用率(%)	64.23	67.81	69.14	71.36

表5表明两种机制在协同优化时产生了正向叠加效应: DGER通过拥塞感知变异为SA提供了更高质量的初始扰动候选解, 而SA的全局搜索能力又进一步弥补了DGER在局部最优规避方面的不足, 二者相辅相成, 共同推动本文算法在三项性能指标上达到最优水平。

综上,实验三验证了本文所提两项改进机制各自的有效性及其协同优化的必要性, 为完整算法的性能优越性提供更充分的实验支撑。

实验四: 负载情况性能评估实验

为进一步验证设计的有效性, 本节中扩展了应用场景, 对混合流量下的调度能力进行了测试。在该实验中, 网络拓扑包含多种类型的流量, 包括TT流和AVB流, 并考虑不同流量负载情况, 以评估算法在复杂场景中的适应性和稳定性。实验四旨在分析本文所提算法在TT流场景下的调度能力, 以及在混合流量共存情况下的调度性能。

实验中对不同算法在TT流场景下的最优输出, 以端到端时延为主要评价标准, 进一步验证算法在复杂网络环境中的调度性能。实验中使用TT

流作为测试对象, 并逐步增加流量数量, 以评估各算法在高负载下的调度效果。为此, 在表1基准场景基础上加入一定数量的TT流, 其大小范围在125~750B之间。为保持实验公平性, 所有算法在相同流量信息下进行相同设置, 实验结果如图7所示。

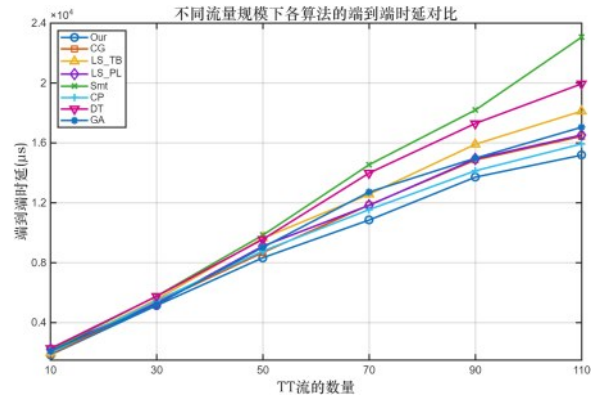


图7 不同流量规模下各算法的端到端时延对比图

图7展示了不同算法在逐渐增加TT流时总的端到端时延的性能表现。可以观察到, 随着TT流数量增加, 所有算法的端到端时延均呈上升趋势。本文提出的算法在所有流量数量下均表现出最低的端到端时延, 特别是在TT流数量增加到50以上时更为显著。这表明采用以流量的路径分配方案为个体的种群设计, 确保算法在优化过程中全面覆盖所有可行路径, 从而增强算法的多样性和探索全局最优解的能力, 这也是其高负载条件下表现优异的关键原因之一。

在第二个实验场景中, 本文在TSN网络中加入AVB流, 以进一步评估算法的调度性能。该实验重点在于分析AVB流在TT流背景下的可调度率, 即有多少AVB流在满足所有时延和带宽约束的情况下成功调度。

在TSN网络传输中, 当流数据超过MTU时, 需要分片为多个以太网帧进行传输。为排除数据分片对调度机制产生的额外影响, 本实验将AVB流

设置在 200~1500 B 之间, 因此, 本文的可调度率统一按流的数量进行计算。可调度率用于衡量 TSN 网络中能够按时完成调度的流量比例, 具体分析公式如下:

$$R_{\text{sched}} = \frac{\sum_{i \in F} \mathbb{I}(d_i \leq D)}{|F|} \quad (14)$$

其中,  $\mathbb{I}(\cdot)$  函数表示端到端时延小于其截止时间的流, 当满足条件  $d_i \leq D$  时, 其值为 1, 否则为 0。

在上述分析框架中, AVB 流的传输不受帧到达时间的限制, 即假设其具有足够的缓冲资源, 使其可在调度允许的时间窗口内完成发送。因此为更好反映 AVB 流的调度需求, 将优化目标中的最大带宽利用率调整为 AVB 流的可调度率。实验在表 1 的基准场景中加入若干 AVB 流, 通过公式(14)评估不同算法在混合流量场景中的可调度率。实验结果如图 8 所示。

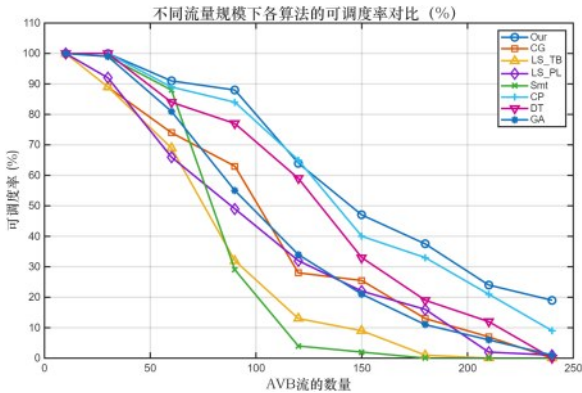


图 8 不同流量规模下各算法的可调度率对比图

图 8 显示各算法的可调度率随 AVB 流增加的变化趋势。随着 AVB 流的增加, 所有算法的可调度率均有所下降, 因为更多 AVB 流占用了网络资源, 导致部分流量未能满足调度要求。在混合流量场景下, 本文算法在确保 TT 流可调度的同时, 更好地调度 AVB 流, 表现出较高的 AVB 流可调度率。相比之下, 其他算法在负载增加时可调度率显著下降, 特别是在带宽资源紧张的情况下。该结果进一步验证了本文所提算法在应对复杂混合场景时的有效性。实验表明, 本文算法能够在满足 TT 流严格实时性要求的前提下, 提升 AVB 流的整体调度性能, 特别是在高负载情况下仍能保持较好的调度质量。

本文算法的优越性主要是以下的机制的协同优

化, 如表 6 所示:

表 6 算法创新机制	
机制	优化
SA	以概率接受劣解, 避免局部最优
DGER	维持种群多样性, 防止过早收敛
三目标协同优化	驱动找寻负载均衡的路径, 避免链路拥塞积累
大种群	全路径枚举, 覆盖更广解空间, 提升全局最优概率

基于表 6 机制, 本文提出的联合调度算法在复杂混合流量场景下具有明显的优势, 能够有效优化网络性能。针对单独 TT 流场景和混合流量场景, 均表现出较好的调度效果。

实验四验证了本文算法在多目标优化环境下的扩展性, 随着流量负载的增加, 本文算法在流量规模扩展时性能退化更为平缓且优于经典对比算法, 且引入的 SA 和 DGER 有效缓解了大规模场景下的早收敛问题, 使算法在网络规模扩展时仍能维持较好的优化性能, 提供更高的调度效率和较好的网络服务质量。

## 4 结论

本文提出了一种基于 NSGA-II 的多目标联合调度优化算法, 用于解决 TSN 网络在联合路由与调度中存在的多目标优化不足、扩展性差等问题。算法同时优化端到端时延、抖动和带宽利用率, 提供了一种能够适应复杂网络环境的有效调度方案。为增强算法的全局搜索和精细优化能力, 本文引入了 DGER 和 SA 机制。实验结果验证了算法在优化多目标性能上的显著优势。

在未来的工作中, 计划引入强化学习等更优算法, 以进一步提升算法在动态和不确定网络环境下的适应性和性能。此外, 还考虑在算法中引入帧抢占机制, 以进一步减少关键任务流量的端到端时延和抖动, 提高网络的实时传输性能, 也将继续探索提升算法计算效率的方法, 使其能在更大规模的网络环境中应用。

宋凯 (2002-), 男, 河南安阳人, 主要研究方向为时间敏感网络。

孔洲 (1999-), 男, 山东济宁人, 主要研究方向是时间敏感网络。



### 参考文献:

- [1] 郭梦杰, 寿国础, 刘雅琼, 等. 面向混合多域组网的软件定义时间敏感网络[J]. 通信学报, 2025, 46(6): 1-17.  
Guo M J, Shou G C, Liu Y Q, et al. Software-Defined Time-Sensitive Networking for Hybrid Multi-Domain Networking[J]. Journal on Communications, 2025, 46(6): 1-17.
- [2] Yang D, Cheng Z R, Zhang W T, et al. Burst-aware time-triggered flow scheduling with enhanced multi-CQF in time-sensitive networks[J]. IEEE/ACM Transactions on Networking, 2023, 31(6):2809-2824.
- [3] Zhao L X, Pop P, Steinhorst S. Quantitative performance comparison of various traffic shapers in time-sensitive networking[J]. IEEE Transactions on Network and Service Management, 2022, 19(3): 2899-2928.
- [4] Li Q, Li D, Jin X, et al. A simple and efficient time-sensitive networking traffic scheduling method for industrial scenarios[J]. Electronics, 2020, 9(12): 2131.
- [5] Stüber T, Osswald L, Lindner S, et al. A survey of scheduling algorithms for the time-aware shaper in time-sensitive networking (TSN)[J]. IEEE Access, 2023, 11: 61192-61233.
- [6] Xie Y B, Zhang L. Resource aware joint routing and scheduling strategy in time sensitive networks [C]. Chongqing: The 44th Chinese Control Conference, 2025.
- [7] 聂宏蕊, 李绍胜, 刘勇. 时间敏感网络中基于 IEEE 802.1Qch 标准的优化调度机制[J]. 通信学报, 2022, 43(9): 12-26.  
Nie H R, Li S S, Liu Y. An optimized scheduling mechanism based on the IEEE 802.1Qch standard for time-sensitive networking. Journal on Communications, 2022, 43(9), 12 - 26.
- [8] Min J H, Kim W, Paek J, et al. Effective routing and scheduling strategies for fault-tolerant time-sensitive networking [J]. IEEE Internet of Things Journal, 2024, 11(6): 11008.
- [9] Li J W, Wu W F, Li Z Y, et al. Optimization design of joint routing and scheduling algorithm for time sensitive networking-audio video bridging [J]. Journal of Beijing Normal University (Natural Science), 2025, 61(5): 659-668.
- [10] Wang W D, Zhang L. A Joint Routing and Group Scheduling Algorithm for Time-Sensitive Networks [C]. Chongqing: The 44th Chinese Control Conference, 2025.
- [11] Wang Z K, Liao W Z, Xia X Y, et al. Routing and Scheduling in Time-Sensitive Networking by Evolutionary Algorithms [J]. Biomimetics, 2025, 10(5): 333-333.
- [12] Xu J, Liu C L, Huo J H, et al. Traffic scheduling mechanism for time-sensitive networks based on satisfiability modulo theories[J]. Computer Science, 2025, 52(11A): 1-6.
- [13] Sun W, Zou Y, Guan N, et al. Joint Routing and Scheduling Optimization of In-Vehicle Time-Sensitive Networks Based on Improved Grey Wolf Optimizer [J]. IEEE Internet of Things Journal, 2024, 11(4): 7093-7106.
- [14] Ashjaei M, Lo Bello L, Daneshmand M, et al. Time-sensitive networking in automotive embedded systems: state of the art and research opportunities[J]. Journal of Systems Architecture, 2021, 117: 102137.
- [15] Zhang Y Z, Xu Q M, Wang S L, et al. Scalable no-wait scheduling with flow-aware model conversion in time-sensitive networking [C]. Rio de Janeiro: IEEE Global Communications Conference (GLOBECOM), 2022: 413-418.
- [16] 何倩, 郭雅楠, 赵宝康, 等. 无等待与时隙映射复用结合的时间触发流调度方法[J]. 通信学报, 2024, 45(8): 193-204.  
He Q, Guo Y N, Zhao B K, et al. A time-triggered flow scheduling method combining no-wait and timeslot mapping multiplexing. Journal on Communications, 45(8), 193 - 204.
- [17] Li Z Q, Xie H, et al. Joint traffic routing and scheduling algorithm eliminating the effect of queuing delay jitter in time-sensitive networking [C]. Macau: 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC), 2022: 1877-1882.
- [18] Nayak N G, Dürr F, Rothenmel K. Routing algorithms for IEEE802.1Qbv networks[J]. ACM SIGBED Review, 2018, 15(3): 13-18.
- [19] Pahlevan M, Tabassam N, Obermaisser R. Heuristic list scheduler for time-triggered traffic in time-sensitive networks[J]. ACM SIGBED Review, 2019, 16(1): 15-20.
- [20] Falk J, Dürr F, Rothenmel K. Time-triggered traffic planning for data networks with conflict graphs [C]. Sydney: 2020 IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS), 2020: 124-136.
- [21] Gu Z R, Ding J Y, Chai Z L. TC-NSGA-II: A multi-objective optimization algorithm for TSN-CAN gateway congestion scheduling [J]. Journal of Beijing University of Aeronautics and Astronautics, 2025.
- [22] Reusch N, Zhao L X, Craciunas S S, et al. Dependability-aware routing and scheduling for time-sensitive networking[J]. Real-Time Systems, 2022, 58(4): 535-578.
- [23] Sun Q S, Zhang Y J, Lu B L, et al. A study of gene expression programming algorithm for dynamically adjusting the parameters of genetic operators [J]. PLOS ONE, 2025, 20(1): e0321711.
- [24] Zhou X, He F, Zhao L X, et al. Hybrid scheduling of tasks and messages for TSN-based avionics systems [J]. IEEE Transactions on Industrial Informatics, 2024, 20(2): 1081-1092.
- [25] Deng L B, Zeng G, Kurachi R, et al. Enhanced real-time scheduling of AVB flows in time-sensitive networking[J]. ACM Transactions on De-

- sign Automation of Electronic Systems, 2024, 29(2): 1-25.
- [26] 王颖, 庄智涵, 喻鹏, 等. 基于时间感知和循环排队转发的 TSN 联合路由和混合流量调度算法[J]. 通信学报, 2025, 46 (4): 109-120.  
Wang Y, Zhuang Z H., Yu P, et al. A TSN joint routing and hybrid traffic scheduling algorithm based on time-aware shaping and cyclic queuing and forwarding. Journal on Communications, 2025, 46(4), 109 - 120.
- [27] Vlk M, Brejchova K, Hanzalek Z, et al. Large-scale periodic scheduling in time-sensitive networks [J]. Computers & Operations Research, 2022, 137: 105512.
- [28] Bujosa D, Ashjaei M, Papadopoulos A V, et al. HERMES: Heuristic multi-queue scheduler for TSN time-triggered traffic with zero reception jitter capabilities [C]. Paris: The 30th International Conference on Real-Time Networks and Systems, 2022: 70-80.
- [29] Zhou Y, Samii S, Eles P, et al. Time-triggered scheduling for time-sensitive networking with preemption [C]. Taipei: 2022 27th Asia and South Pacific Design Automation Conference (ASP-DAC), 2022: 262-267.
- [30] Vlk M, Hanzalek Z, TANG S. Constraint programming approaches to joint routing and scheduling in time-sensitive networks [J]. Computers & Industrial Engineering, 2021, 157: 107317.
- [31] Lu Y Q, Huang C H, Chen J R, et al. Time sensitive network scheduling method based on genetic algorithm [J]. Journal of South China University of Technology (Natural Science Edition), 2024, 52 (2): 1-12.
- [32] Zheng Z, Chen Y, Liu Y, et al. Real-time scheduling for 802.1Qbv time-sensitive networks: a survey and experimental study [J]. IEEE Communications Surveys & Tutorials, 2024.

#### [作者简介]



徐艳丽 (1984-), 女, 山东聊城人, 博士, 上海海事大学信息工程学院教授, 主要研究方向为时间敏感网络、自动驾驶、卫星通信、空天地海一体化通信技术等。