

D2D网络中基于SVC视频流缓存的 跨层网络编码重传方案

姚玉坤,陈鑫垚,羊杰

(重庆邮电大学 通信与信息工程学院,重庆 400065)

摘要: 为了降低基站重传视频流的完成时间,针对视频流业务设计了一种D2D网络中基于可伸缩视频编码(Scalable Video Coding, SVC)的视频流缓存跨层网络编码重传方案。该方案首先采用SVC对视频流进行编码以应对终端处理能力的差异性,引入跨层网络编码搜寻最大独立集选取最佳的传输以及编码调度,减少完成时间。其次,针对多协作重传设备间的干扰问题,设计了最佳的资源调度算法,在给定发送设备集的前提下迭代优化设备的发送功率,在不增加完成时间的同时优化系统的吞吐量。仿真结果表明,所提出的方案能够有效地降低完成时延,减少重传次数,增加系统的弹性。

关键词: D2D网络;可伸缩视频流;跨层网络编码;完成时间;吞吐量

中图分类号: TN919 **文献标志码:** A **文章编号:** 1673-5439(2023)01-0011-009

A cross-layer network coding retransmission scheme based on SVC for video stream caching in D2D networks

YAO Yukun, CHEN Xinyao, YANG Jie

(School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

Abstract: In order to reduce the time of video stream retransmission of base stations, this paper designs a cross-layer network coding retransmission scheme based on scalable video coding (SVC) for video stream caching in D2D networks. First, this scheme uses SVC to encode the video stream to cope with the differences of terminal processing capabilities. It then introduces the cross-layer network coding to search for the largest independent set, and selects the best transmission and coding schedule to reduce the time span. Second, since the interference problem exists between multiple cooperative retransmission devices, an optimal resource scheduling algorithm is designed. Given a set of transmitting devices, it iteratively optimizes the transmit power of the devices, and optimizes the throughput of the system without increasing the time. The simulation results demonstrate that the proposed scheme can effectively reduce the time delay and the number of retransmissions, as well as increase the robustness.

Keywords: D2D network; scalable video streaming; cross-layer network coding; completion time; throughput

随着5G的到来,智能手机和移动服务的指数级增长导致了蜂窝设备流量的爆炸式增长,但是事实上,大量的蜂窝流量是由视频的应用程序产生的。同样地,随着移动通信的发展,出现了各种类型的移

动终端,终端呈现出差异性,这种差异性不仅表现在处理能力上还表现在对内容的需求上,因此对于这种实时的应用程序,不同用户的体验质量是不同的。为了适应这种多样性,文献[1]提出了H.264/SVC

收稿日期:2022-03-15;修回日期:2022-09-02 本刊网址: <http://nyzr.njupt.edu.cn>

基金项目:国家自然科学基金(61971080)资助项目

作者简介:姚玉坤,女,教授, yaoyk@cqupt.edu.cn

引用本文:姚玉坤,陈鑫垚,羊杰.D2D网络中基于SVC视频流缓存的跨层网络编码重传方案[J].南京邮电大学学报(自然科学版),2023,43(1):11-19.

(可扩展视频编解码器)作为 H.265/AVC(高级视频编解码器)的扩展。在 SVC^[2] 中,一个画面组(Group of Picture, GOP)被逐步编码成多个质量层,其中包含一个基本层(Base Layer, BL)和几个增强层(Enhancement Layer, EL)。BL 保证视频的基本质量,每个 EL 进一步提高视频的质量,用户可以通过获得除基本层之外更多的增强层来获取更好的视频质量,因此如何在网络中传输 SVC 视频流是一个值得研究的课题。

2000 年, Ahlswede 等^[3] 提出网络编码(Network Coding, NC), 打破了中间节点的存储转发的模式, 可以按照一定的规则对数据包进行编码转发, 有效地提高了传输效率。文献[4]提出了机会式网络编码, 中间节点对数据包进行简单的异或(eXclusive OR, XOR)操作, 终端接收到编码包后可以对其进行异或解码。立即可解网络编码(Instantly Decoding Network Coding, IDNC)作为异或编码的子类^[5], 一直被用于时延敏感的场景中, 在降低完成时间以及解码时延方面有着很大的优势。

近年来, 如何充分利用网络编码来减少重传次数、提高吞吐量并将其与可伸缩视频传输结合的问题引起了广大学者的注意。文献[6]提出了基于随机线性网络编码(Random Linear Network Coding, RLNC)的多播视频传输和分类调度机制, 通过降低高优先级的错误传输次数从而提高视频流的质量并降低完成时间。文献[7]提出了不等差错保护随机线性网络编码策略, 通过使用概率机制对下层数据包进行编码, 以实现下层可扩展视频流的高解码概率。文献[8]使用随机线性网络编码重传视频数据包, 在有损信道下保持基本层的高解码率, 提高视频质量的同时降低重传次数。文献[9]提出了一种在线算法, 通过优化分组传输和网络编码调度来降低视频数据包的重传次数以及完成时间。上述工作都在一定程度上提高了视频质量, 然而使用随机线性网络编码的方式, 接收器会接收许多冗余的包, 导致较长的解码延迟。文献[10]提出了一种无碰撞协同解决方案, 该方案使用 IDNC 来减少 D2D 网络中重传数据包的次数和完成时间, 并且通过仿真验证使用 IDNC 机制减少完成时间的方案明显高于使用 RLNC 的方案。因此, 机会编码方案在接收时对数据包进行立即解码的优点更适用于高效的 SVC 视频流的传输。

应用 IDNC 的网络中, 网络交付问题被称为完成时间最小化问题, 从网络层的角度看, 采用 IDNC

调度解决视频流完成时间最小化问题可以转化为最少传输数据包数量的问题^[11-14], 这些工作通过数据包擦除概率模拟物理通道的状态, 并将这种擦除概率集成到编码决策中。例如文献[11]和文献[12], 将多个设备调度到同一个资源块, 从网络层的角度提高了系统的性能, 但是选择所有传输设备的最小速率, 从物理层的角度降低了性能的同时还增加了完成时间。与只依赖于数据包组合的网络层不同, 速率感知网络编码(Rate-Aware Instantly Decoding Network Coding, RA-IDNC)依赖于不同的设备容量和 IDNC 数据包组合来优化传输问题。文献[15]使用 RA-IDNC 来选取联合调度编码方案和传输速率从而减少整体的完成时间。文献[16]将完成时间最小化问题公式化, 并提出一种解决问题的 RA-IDNC 图方案, 选取最佳的传输以及编码调度来进行完成时间的优化。文献[17]将完成时间最小化问题转换为使用 RA-IDNC 图模型选取最大权重的独立集问题, 通过选取最大权重的传输调度来减少完成时间和重传次数。文献[18]使用 RA-IDNC 来优化云无线电接入网络的完成时间。文献[19]和文献[20]的作者分别开发了跨层 IDNC 算法, 优化集中式 C-RANs 的吞吐量、服务质量的同时降低了完成时间。文献[21]和文献[22]都是采用跨层网络编码来解决雾无线电接入网的服务质量和完成时间问题, 通过图论的方法将问题转换为联合跨层云基站和服务质量保证问题, 并通过迭代式方法搜索顶点的最大权重从而选取最佳的减少完成时间、提高服务质量的传输方案。以上文献的研究涉及到了 RA-IDNC 以及跨层网络编码, 从一定程度上降低了完成时间、重传次数。但是, 现有文献并没有在 D2D 网络中针对 SVC 视频数据进行跨层网络编码模型的设计, 同样也没有充分考虑物理层的传输速率并根据吞吐量指标重新进行调度集合的规划。

本文以最小化完成时间为目标, 在现有的研究基础上提出了一种基于 SVC 视频流缓存的跨层网络编码重传方案(Cross-Layer Network Coding Retransmission Scheme based on SVC Video Stream Caching in D2D Networks, NCVS)。具体贡献如下:

(1) 设计了基于 SVC 缓存的 D2D-RA-IDNC 图, 将完成时间最小化问题转换为最大权值独立集问题, 在此图中选取最佳的传输以及编码调度。

(2) 根据 SVC 视频流严格的解码顺序、潜在的解码机会、链路丢包率等因素设计顶点权重。

(3) 设计了资源调度算法, 迭代选取发送设备

使得拥有最大的潜在接收设备集的发送设备优先被选取,其次衡量特定发送设备的发射功率,并且在不增加完成时间的前提下优化系统的吞吐量。

(4) 将提出的方案与现有的方案进行比较,仿真结果表明,本文所提出的算法显著地减少了完成时间。

1 系统模型及相关定义

1.1 系统模型

如图 1 所示,D2D 网络模型中基站向 N 个终端设备 ($U = \{1, 2, \dots, n\}, n \in N$) 传输 SVC 编码的视频流。其中每个设备具有两个接口:一个无线接口和一个蜂窝接口。

一般地, SVC 视频流由若干个 GOP 组成,每个 GOP 有 L 个可伸缩视频层 $L_1, L_2, \dots, L_L$, 第一层为基本层,剩余层次为增强层。一个 SVC 数据包用 pkt_i^l 表示,其含义为第 i 层的第 l 个数据包,因此所有层的数据包组合表示为 $M_{\text{all}} = \{pkt_1^1, pkt_1^2, \dots, pkt_m^k, \dots, pkt_m^l, \dots, pkt_L^l\}$ 。假设一个 GOP 的所有 L 层 SVC 数据包的数量为 M ,表示为 $M = \sum_{i=1}^L m_i, 1 \leq i \leq L$ 。数据包传输时间用 t 表示, $t=0$ 表示传输阶段的开始。

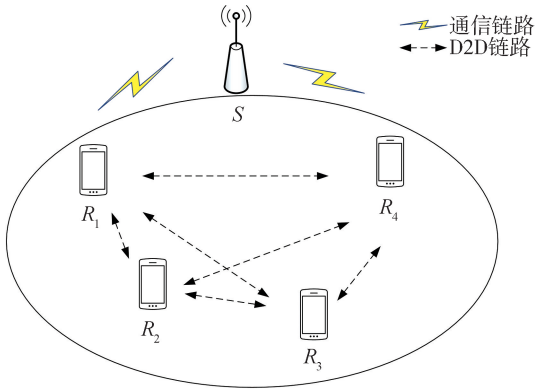


图 1 FC-D2D 网络模型图

传输模型中,第一个阶段基站广播 SVC 视频序列后,由于接收端的处理能力以及带宽限制等因素每个设备上的数据包集合可能有以下几种:

- (1) 拥有集 H_x : 接收端 u_x 成功接收到的数据包的集合。
- (2) 缺失集 DP_x : 接收端 u_x 丢失的并且不需要的数据包的集合。
- (3) 想要集 $W_x = M - H_x - DP_x$: 接收端 u_x 上仍然丢失并且需要的数据包的集合。

第二阶段是终端之间的协作重传,为了减少完成时间,设备之间通过网络编码辅助协作传输来获取它们需要的数据包。当设备成功接收到相邻设备传输的数据包时表示数据包被此设备成功恢复。

1.2 相关定义

定义 1 状态反馈矩阵 (State Feedback Matrix, SFM): 终端将数据包的接收情况反馈给基站,基站收集各终端接收情况后形成相关矩阵,此矩阵称为状态反馈矩阵,如式(1)所示。

$$\text{SFM} = f_{k,l} = \begin{cases} 0 & l \in H_k \\ 1 & l \in W_x, \quad \forall k \in U, \forall l \in M \\ -1 & l \in DP_x \end{cases} \quad (1)$$

定义 2 通道容量 (Channel Capacity, CC): 设备 u_k 到设备 u_i 的可实现速率 R_{u_k, u_i} 的集合小于或者等于通道容量 C_{u_k, u_i} , 是 R_i 的一个子集,其中可实现速率表示为 $R_{u_i, u_k} = \{r \in R_{u_k, u_i} \mid r \leq C_{u_i, u_k} \& \& u_i \in T_i(p_i^*)\}$ 。

定义 3 解码时延 (Decoding Delay, DD): 如果发送速率为可行速率,表示为 $R_{u_i, u_k} = \{r \in R_{u_k, u_i} \mid r \leq C_{u_i, u_k} \& \& u_i \in T_i(p_i^*)\}$, 接收端接收到编码包时有以下 3 种情况:

- (1) 不可解编码包: $|p_i^* \cap W_i| \neq 1$, 发送端发送的编码包中不包含接收端需求的一个数据包。
- (2) 立即可解编码包: $|p_i^* \cap W_i| = 1$, 发送端发送的编码包中只包含接收端需求的一个数据包。
- (3) 非立即可解编码包: $|p_i^* \cap W_i| \geq 2$, 接收端接收到的编码包中包含需求的两个或者两个以上数据包。

不可解编码包、非立即可解网络编码包被终端接收或者终端未接收到编码包时解码时延会增加一个单位,表示为

$$T_{u_i}(t) = \begin{cases} 0 & |W_i| \neq \emptyset \& \& |W_i \cap p_i^*| = 1 \\ 1 & \text{其他} \end{cases} \quad (2)$$

定义 4 单独完成时间 (Individual Completion Delay, ICD): 在任意时间 m 内,发送端 u_i 的发送速率为可行速率时,非空需求集的终端 $u_m \in N_w$ 传输以及解码所需求数据包的总时间。表述为

$$\text{ICD} = |\bar{W}_i| + \sum_{m=1}^t T_i(m) \& \& R_{u_i, u_k} = \{r \in R_{u_k, u_i} \mid r \leq C_{u_i, u_k} \& \& u_i \in T_i(p_i^*)\} \quad (3)$$

定义 5 完成时延 (Completion Delay, CD): 发

送端 u_k 的发送速率为可行速率时所有终端获取需求数据包的传输总时间。表述为

$$CD = \sum_{i=1}^n I_i CD \quad (4)$$

1.3 D2D-RAIDNC 模型

为了估计每个设备的最佳传输以及编码调度策略,本文采用 D2D-RAIDNC 模型,假设传输调度系数 $tr_{r,x} = 1$ 表示设备 u_x 在时隙 r 时被选作传输设备, $tr_{r,x} = 0$ 表示设备 u_x 在时隙 r 时被选作接收设备。因此,可以将传输调度策略表示为 $n \times 1$ 的向量: $tr_r = \{tr_{r,1}, tr_{r,2}, \dots, tr_{r,n}\}$, 将 FC-D2D 网络中的一系列传输调度策略表示为: $TRQ = \{tr_1, tr_2, \dots, tr_Q\}$ 。

D2D-RAIDNC 图模型表示为 $G_c(N_c, E_c)$, 为发送设备 u_x 的邻居 u_y 生成相关顶点 $v_{i,x,y}^{l,r}$, 其中

表 1 本文主要符号及含义

符号	含义	符号	含义
u_i	终端设备	$CP_r(u_x)$	传输设备 u_x 在第 r 轮采用的编码模式
pkt_i^l	第 i 层的第 l 个数据包	$G_c(N_c, E_c)$	D2D-RAIDNC 模型
M_w	非空需求集	A	传输设备集
M	SVC 数据包的数量	$tr_{r,x}$	设备 u_x 在时隙 r 被选做传输的标志
$O_x(CP_r(u_x))$	传输设备 u_x 的接收设备	E_c	冲突边即存在 RA-IDNC 图的边
H_x^r	时隙 r 中传输设备 u_x 的拥有集	Vw_x	D2D-RAIDNC 模型顶点
$\mathbb{R}_{u_k, u_i}$	设备 u_x 到设备 u_i 的信道增益	$T_i(p_i^*)$	目标设备
q_y	从基站正确接收到需求数据包的概率	$p_{x,y}$	设备 u_x 传输到设备 u_y 的丢包率
$g_{x,y}$	设备 u_y 接收来自发送设备 u_x 发送数据包的系数	$ u_k $	第 k 层仍然缺少的数据包的数量
$ n_k $	在第 k 层拥有的数据包的数量	$NBR(u_x)$	设备 u_x 的邻居

2 基于 SVC 缓存的 D2D-RAIDNC 调度方案

2.1 问题描述

2.1.1 完成时间最小化

在 D2D 传输调度中,为了减小完成时间以及重传次数,需要在每个时隙 t 中选择一组传输设备和数据包组合、以及相应的传输速率和最佳编码决策。本文将减少完成时间调度问题表述为子问题 P_1 , 为了选取最佳的调度决策,在每个传输时隙中计算拥有非空需求集设备的完成时间下界。

此处引入给定时隙 t 时, $u_i \in N_w$ 的完成时间 $T_i(t)$ 的下界^[23]为

下标 x 表示发送设备, y 表示接收设备, i 表示 SVC 的第 i 层;上标 r 表示发送速率, l 表示 SVC 的第 i 层的第 l 个数据包。若满足以下一个条件则将顶点相连。

条件 1: $u_x = u_{x'} \&\& ((l_1 \neq l_2) || (i_1 \neq i_2)) \&\& (pkt_{i_1}^{l_1}, pkt_{i_2}^{l_2}) \notin H_{u_x} \times H_{u_{x'}}$ 表示相同的发送设备发送不同的数据包,但是这两个发送设备至少有一个没有拥有另一个设备需求的数据包。

条件 2: $r \neq r'$ 表示发送设备的发送速率是不同的。

条件 3: $u_{x_1} \neq u_{x_2} \&\& u_{y_1} = u_{y_2}$ 表示不同的发送设备传输数据包到相同的接收设备。

本文所用的符号如表 1 所示。

$$T_{u_i}(t) \approx \frac{B |W_{u_i}(0)|}{\bar{R}_{u_i}} + \mathbb{N}_{u_i}(t) \quad (5)$$

式中, $\mathbb{N}_{u_i}(t)$ 表示为设备 u_i 经历的时间延迟, $\bar{R}_{u_i}$ 表示来自所有设备的信道容量的平均值。

因此本文将 FC-D2D 系统中 SVC 视频流完成时间最小化问题表述为

$$\min_{\substack{T_i(p_i^*), r_{u_k}, T_{u_k} \\ A \in U}} \{ \max_{u_i \in N_w} T_{u_i}(t) \} \quad (6)$$

同时约束于下列条件:

$$CP_r(u_x) \in G_c(N_c, E_c), \forall tr_{r,x} = 1, \quad (7)$$

$$tr_{r,x} = 1 \&\& tr_{r,y} = 0$$

$$\forall u_x, u_y \in N(G), u_n \in (O(CP_r(u_x)) \cap$$

$$O(CP_r(u_y)) \&\& (CP_r(u_x), CP_r(u_y)) \in E_c \quad (8)$$

$$H_x^r = H_x^{r-1}, \forall tr_{r,x} = 1 \quad (9)$$

$$H_y^r = H_y^{r-1} \cup \{pkt_i^l\}, \forall w_x = Vw_x - \{v_{i,y}^l\}, \\ \forall pkt_i^l \in CP_r(u_x), v_{i,y}^l \in CP_r(u_x), \\ u_y \in O_x(CP_r(u_x)), \forall tr_{r,x} = 1 \quad (10)$$

$$H_y^r = H_y^{r-1}, \forall n_y \notin \bigcup_{tr_{r,x}=1} O_x(CP_r(u_x)) \quad (11)$$

$$H_x \cup DP_x \cup W_x = M, \forall u_y \in N(G) \quad (12)$$

$$Q \in Z \quad (13)$$

约束(7)中保证了每个编码模式的可行性。

约束(8)保证了同一时隙发送到相同接收设备的不同发送设备都不会被选为传输设备。约束(9)说明了设备在充当传输设备时无法获取新的数据包。约束(10)表示在第 r 轮中成功获取的数据包将进行更新,并且将相关顶点从 D2D-RAIDNC 模型图中删除。约束(11)表示在每一轮传输中,对于每一个设备,如果它不是接收设备,则无法接收数据包,也无法更新数据包。约束(12)确保所有数据包都将被成功收到。约束(13)保证了数据包的完整性。

2.1.2 吞吐量优化

传输设备数量的增加会导致多个发送设备之间传输信道的干扰,从而降低信道容量,如式(14)所示。因此,需要采用一种资源分配机制,优化设备的发送功率,有效地选择接收设备集,使得在不增加完成时间的前提下优化和速率。

$$\text{SINR}_{u_k, u_i}(Q) = \frac{Q_{u_k} |\mathbb{R}_{u_k, u_i}|^2}{N_0 + \sum_{u_m \neq u_k} Q_{u_m} |\mathbb{R}_{u_m, u_i}|^2} \quad \forall u_k, u_m \in A \quad (14)$$

式(14)表示了多个协作发送设备发送数据包的信噪比,表明采用多个协作发送设备之后会增大噪声的占比从而降低信噪比。

在本文中,将系统的吞吐量优化问题表述为子问题 P_2 :

$$\max_{\{Q_k\}} \sum_{k=1}^A N_k = \max_{\{Q_k\}} \sum_{k=1}^A \sum_{u_i \in T_i(p_i^*)} \log_2(1 + \text{SINR}_{u_k, u_i}) \\ 0 \leq Q_k \leq Q_{\max}, \forall k \quad (15)$$

此时的目的是实现从设备 u_k 到设备 u_i 的最优

功率分配,因此根据目标函数(15)得到 $\frac{\partial N_k}{\partial Q_k} = 0$ 的局部最优解,将第 t 个时隙第 k 个发射设备的给定发射功率表示为

$$\frac{\partial N_k}{\partial Q_k} = 0 \rightarrow Q_k = \frac{\sum_{u_i \in u(f_{u_k})} \frac{\text{SINR}_{u_k, u_i}}{1 + \text{SINR}_{u_k, u_i}}}{\sum_{m=1, m \neq k} \sum_{u_j \in u(f_{u_m})} \left(\frac{\text{SINR}_{u_m, u_j}}{1 + \text{SINR}_{u_m, u_j}} \right)^2 \frac{\mathbb{R}_{u_k, u_i}}{Q_m \mathbb{R}_{u_m, u_j}}} \quad (16)$$

由于上述两个子问题是 NP-hard 问题,因此在求解最佳传输调度以及吞吐量时采用贪心算法将其进行分解,求出每个发送设备的局部最优解从而满足全局最优。

2.2 传输设备集的功率分配算法

为了减小设备之间的干扰,功率分配机制有效地对传输设备进行功率分配。因此在本节提出了一种贪婪算法,该算法计算给定传输设备集的最优功率分配,以优化传输设备集的吞吐量(见算法1)。

算法1 发射功率分配算法

输入:传输设备、数据包组合以及每个传输设备相关的接收设备

输出:所有传输设备的最终传输功率

- 1.初始化: $Q = Q_0, \forall k = 1, 2, \dots, A, t = 1$
- 2.重复应用式(16)更新第 k 个传输设备的功率分配
3. $Q = Q_{u_k}, \forall k = 1, 2, \dots, A, t = t + 1$
- 4.直到满足式(15)

2.3 设备的传输以及编码调度算法

该算法采用启发式算法的思想从权重最高的顶点开始,然后以迭代的方式向极大独立集中加入顶点,前提是保留极大独立集的属性,并且保证加入极大独立集的顶点是其余顶点中权重最高的。接收设备将收到的数据包的相关顶点删除并且更新状态反馈矩阵,直到满足用户的全部需求(见算法2)。

算法2 设备的传输以及编码调度算法

输入: $G = (N, E), H_x^r, W_x^r$ //输入构建的 RA-IDNC 图;设备在第 r 轮的拥有集、需求集

输出: tr_r, I_w //第 r 轮的传输调度

- 1.初始化: $tr_r, I_w, CP_r(u_x), O_x(CP_r(u_x))$ // $CP_r(u_x)$ 表示传输设备 u_x 在第 r 轮的编码模式集合; $O_x(CP_r(u_x))$ 表示设备 u_x 传输数据包的接收设备
- 2.for $x=1:n$ //对于每个发送设备 $u_x \in N(G)$
- 3.产生 $G_C = \{N_C, E_C\}$
- 4.对于每个顶点 $u \in Vw_x$, 用计算式(20)计算它的顶点权重
- 5.对 Vw_x 中的顶点按照权重进行递减顺序的排序
- 6.while $Vw_x \neq \emptyset$ do
- 7.从 Vw_x 队列删除第一个顶点 $v_{i,y}^l$
- 8.if $CP_r(u_x) \cup \{v_{i,x,y}^l\}$ 形成一个最大独立集
9. $CP_r(u_x) = CP_r(u_x) \cup \{v_{i,x,y}^l\}$ //将最大权重的顶点加入独立集

```

10.  $O_x(CP_r(u_x)) = O_x(CP_r(u_x)) \cup \{v_{i,x,y}^{l,r}\}$ 
11. end if
12. end while
13. end for
14. for  $i = 1:n$ 
15.  $w(O_x(CP_r(u_x))) = w(CP_r(u_x)), \forall O_x(CP_r(u_x)) \in N_C //$ 
对可行的独立集进行权重计算
16. end for
17. while  $N_C \neq \emptyset$  do
18. 从  $N_C$  删除第一个顶点
19. if  $I_w \cup O_x(CP_r(u_x))$  形成一个极大独立集
20.  $I_w = I_w \cup \{O_x(CP_r(u_x))\}$ 
21.  $tr_{r,x} = 1$ 
22. 从  $N_C$  中删除  $\{O_v(CP_r(u_v)) \mid \forall u_v \in NBR(u_x)\}$ 
23. 重复步骤 3~12
24. end if
25. end while

```

2.4 跨层网络编码资源调度算法

所提出的算法在给定发送设备集的前提下,首先选取具有最大潜在接收设备的发射设备,并且将此设备包含在 A 中。随后在每次迭代中增加一个传输设备,这一设备与其他传输设备相比具有最大的潜在接收设备集,依次迭代,使得传输的和速率最大化(见算法 3)。

算法 3 跨层网络编码资源调度算法

输入:传输设备、数据包组合以及每个传输设备相关的接收设备
输出:总完成时间 T

```

1. 初始化:  $A = \emptyset, M_w = N_w, X = \emptyset, Q = Q_0 //$  发送设备集为
空,非空需求集为可能的发送或者接收设备,目标设备集合为空,此时
不考虑设备间的干扰且传输功率水平对于每个潜在的传输设备
 $Q_0 \in Q_{feasible}$  是可行的。
2. 将  $M_w$  作为潜在的发射或者目标设备构建 D2D-RAIDNC 图
3. 利用算法 2 选择最大权重的独立集。其中传输设备为  $u_k$ , 最大
独立集对应的数据包组合为  $f, u_k$  的目标设备集合为  $u(f_{u_k})$ 
4. 使用式(5)计算  $X$  集合中目标设备的单独完成时间的下界,
并通过  $B/r(t)$  计算非目标设备的时间
5.  $A = \{A, u_k\} //$  首先将最大接收集的发送设备加入
6.  $M_w = M_w \setminus (\{u_k\} \cup X(f_{u_k})) //$  将作为未来的传输或者接收
设备的集合进行更新
7. 将传输功率设置为  $Q_0$  计算 SINRs,并考虑来自集合  $A$  中设备的
干扰
8. 利用算法 1 优化  $A$  中的传输设备的传输功率,优化吞吐量
9. if 和速率增加,  $A = A \cup u_i, M_w = M_w \setminus u_i$ 
10. end if
11. 重复步骤 5~10
12. 重新计算  $T$ , 并保存使得完成时间最小化的解决方案
13. 直到没有设备可以添加到集合  $A$  中

```

2.5 顶点权重设计

本节在设计顶点权重的时候主要考虑以下几个因素:

(1) 由于 SVC 严格的解码顺序,拥有下层顶点的包有利于更好地解码更高层的数据包,因此为基本层的数据包组成的顶点赋予更高的权重。

(2) 拥有更多顶点以及更高权重的数据包的设备优先传输,使得更多的终端获取需求的数据包。

(3) 为数据包提供潜在解码增益的顶点赋予更高的权重,下面分别根据数据包成功接收率、终端解码率以及顶点效用值等方面综合考虑设计权重。

(a) 数据包成功接收率

$$P_{j,x,y}^{k,r} = \{q_y + (1 - q_y)(1 - (1 - g_{x,y}(1 - p_{x,y})))\} * \left(\frac{B}{r}\right) \quad (17)$$

式(17)综合考虑设备协作传输以及基站传输数据包时链路丢包率的情况,设计了数据包 pkt_j^k 被终端 u_y 成功接收的概率。其中, $g_{x,y} = 1$ 表示设备 u_x 发送给设备 u_y 的数据包被成功接收,反之则接收失败。

(b) 终端解码率

$$P_D(v_{i,x,y}^{l,r}) = \prod_{pkt_j^k \in CP_r(u_x), pkt_i^l \neq pkt_j^k} P_{j,x,y}^{k,r} \quad (18)$$

式(18)考虑了数据包在终端的解码率,此种情况发生在传输设备 u_x 拥有接收设备需要的数据包 pkt_j^k 并且此数据包不是接收设备已经获得的数据包 pkt_i^l 的情况下,明显地终端解码率等于设备 u_y 已经获得的 $CP_r(u_x) - \{pkt_j^k\}$ 的数据包。

(c) 顶点效用值

$$u(v_{i,x,y}^{l,r}) = \frac{\sum_{1 \leq k \leq L} (|n_k| - |u_k|)}{\sum_{1 \leq m \leq k} |u_m|} \quad (19)$$

式(19)统计了已经获得的数据包以及接收 l 层仍然需要的低层数据包,为了接收 k 层的数据包需要接收的底层数据包的总数表示为 $\sum_{1 \leq m \leq k} |u_m|$; 因此当需求的层数越低则顶点的效用值越高,使得最低层的顶点优先被选取。

综合上述的因素进行 D2D-RAIDNC 图顶点权重的设计,如式(20)所示。

$$w(v_{i,x,y}^{l,r}) = (1 - p_{x,y}) P_D(v_{i,x,y}^{l,r}) u(v_{i,x,y}^{l,r}), \quad \forall (pkt_i^l, u_y) \mid (pkt_i^l \in CP_r(u_x), u_y \in O_x(CP_r(u_x))) \quad (20)$$

因此,本文设计的编码权重为将顶点的权重进行相加求得的总和权重,如式(21)所示。

$$w(CP_r(u_x)) = \sum_{pkt_l^i \in CP_r(u_x), u_y \in O_x(CP_r(u_x))} w(v_{i,x,y}^{l,r})$$

(21)

2.6 复杂度分析

在所提出的 NCVS 算法中,每个设备缓存一组数据包,构建此设备的 D2D-RAIDNC 图的顶点数为 $V = |C| \times N$, 因此需要 VN 个顶点构成所有设备的 D2D-RAIDNC 图,建立邻接矩阵需要的计算复杂度为: $O(V^2N^2)$ 。由于在独立集中具有相同传输速率的数据包才可以被传输设备传输,因此算法中有 $|R|$ 个独立集。在每个时隙中对于每个传输设备只能传输每个目标设备需求的一个编码包,因此在 MIS 中最多有 V 个顶点,每次具有给定速率的迭代需要 $O(VN)$ 的计算复杂度。搜索其余顶点,最多为 $N-1$,因此搜寻所有速率以及最大权重的 IS 的计算复杂度为 $O(|R|(VN+N-1))$,总的计算复杂度为 $O(|R|(VN+N-1)) + O(V^2N^2)$ 。由于计算复杂度取决于最大部分且 $O(V^2N^2) \gg O(|R|(VN+N-1))$, 因此本文提出算法的计算复杂度为 $O(V^2N^2)$ 。

2.7 通信开销分析

本文采用集中式的传输方式,基站采用 NCVS 算法得到相应的传输设备、数据包组合和接收设备集合并通知传输设备其传输策略。在算法的整个过程中,设备首先反馈数据包的接收状态到基站,基站需要知道所有设备的相关消息,这需要几个字节。而未编码的方案同样需要将数据包的接收情况反馈给基站,因此与未编码的解决方案相比,已经编码的设备不需要额外的通信开销。然而,编码方案需要每个发送编码包设备的头部开销,但是这种开销与整体的数据包大小相比,是可以忽略不计的。

3 仿真与分析

本文使用 OPNET Modeler 14.5 网络仿真软件进行仿真,将提出的方案与未编码的方案、协同 IDNC 方案^[14]、EW-IDNC^[24]方案以及 MTPC^[25]方案进行对比分析。实验网络模型为 D2D 网络模型,仿真实验以平均完成时延、平均解码时延作为衡量指标,针对 SOCCER 视频序列进行多次实验,且视频序列均经过 SVC 编码。视频序列 SOCCER 有 300 帧,帧的速率为 30 f/s,经过计算该序列具有 300/8≈38 个 GOP,经过 SVC 编码后每个 GOP 被分为 4 个视频层,具有 8 帧。

仿真参数设置如表 2 所示。

表 2 仿真参数设置表

参数	数值
基站的覆盖范围/m	500
基站到终端的丢包率	0.2~0.3
设备之间的丢包率	0.10~0.15
帧数	[8,16]
帧速率/(f·s ⁻¹)	30
终端设备数	[10,15,20,25,30]
路径损耗/dB	148 + 37.6lg(<i>d</i> _{<i>u_k</i>} , <i>u_i</i>)
瑞利衰落	(0,1)
噪声功率/(dBm·Hz ⁻¹)	-174
设备的最大功率 <i>Q</i> /(dBm·Hz ⁻¹)	-42.60

图 2 表示在 D2D 模型中设备传输 16 个大小为 1 Mb 的数据包帧的平均完成时间随着设备数量变化的情况。仿真结果表明,随着设备数量的增加,平均完成时间增加。与其他方案相比,由于本文提出的算法优先选择最大潜在接收设备的发送设备,选取最佳的传输调度的同时优化每个传输设备的传输功率,在不增加平均完成时间的同时提高了系统的吞吐量,而未编码的方案未选择最佳的传输调度,至少需要传输 M 次来满足所有设备的需求。一方面,RA-IDNC 算法在每个时隙中只选择一个最佳传输设备,没有充分考虑传输调度以及每个传输设备的传输速率,另一方面,EW-IDNC 方案以及 MTPC 方案都没有充分考虑协作重传设备之间的干扰以及吞吐量的优化导致设备之间重传的完成时间的增长,因此本文提出的方案在降低平均完成时间方面有着很大的优势。

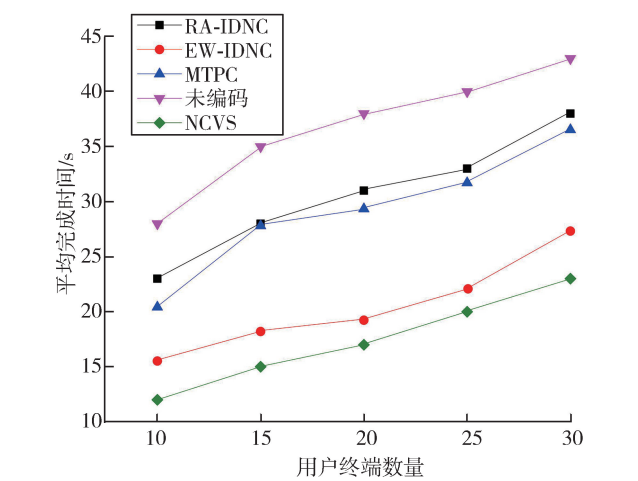


图 2 平均完成时间随着数据包数变化的性能对比图

如图 3 所示,模拟了 20 个传输设备与 16 个数据包组成的 D2D 系统平均完成时间随着数据包大小变化的情况。仿真结果表明,随着数据包帧的逐渐增大,平均完成时间呈线性增加。从物理层看,随着数据包帧大小的增加,需要更多的比特来传输数据包帧,从而造成了传输时间的延长。而本文提出的算法对吞吐量优化的同时减小完成时间,选取最佳的方案进行重传,因此本文提出的算法具有明显的优势。

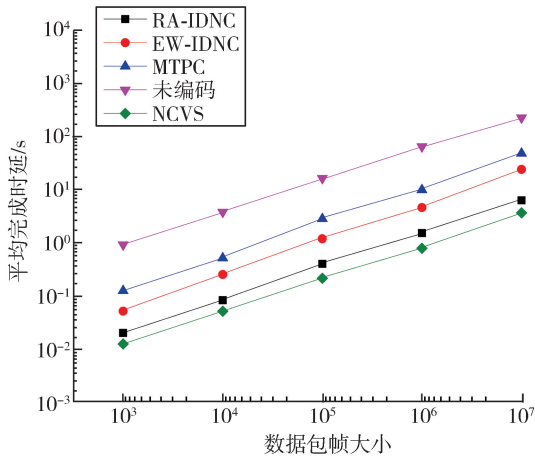


图 3 平均完成时延随着数据包帧大小变化的性能对比图

图 4 表示了在 D2D 模型中设备传输 16 个大小为 1 Mb 的数据包帧平均解码时延随着设备数量的变化情况。仿真结果表明,随着设备数量的增加,平均解码时延增加,这是因为随着设备数量的增加,需要满足用户需求的数据包个数增加,因此造成了解码时延的增长。与其他方案相比,本文提出的算法综合考虑了数据包的解码顺序、未来的解码机会以及数据包的效用等因素设计权重,使得底层中更多的数据包可以被解码,大大缩短了解码时延,相较于

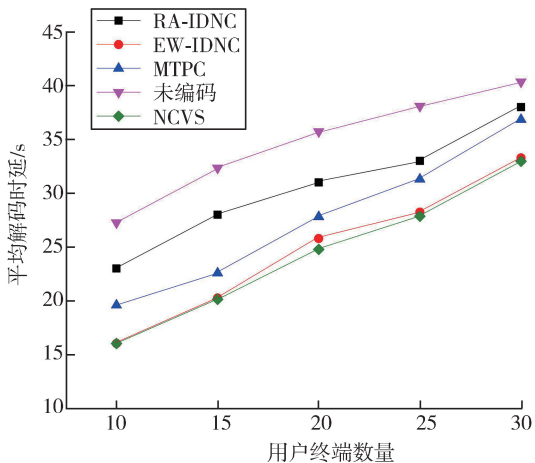


图 4 平均解码时延随着用户终端数量变化的性能对比

RA-IDNC 算法以及未编码方案来说缩短解码时延的效果较明显,因为多设备协作传输更多终端可以解码的编码包使得解码时延明显缩短。本文提出的算法首先传输 SVC 视频流的底层数据包使得终端可以解码更多的数据包,与 EW-IDNC 算法相比降低解码时延的效果差别不大,因而在减小解码时延方面有着很大的优势。

4 结束语

本文针对 FC-D2D 网络提出了基于 SVC 缓存的跨层网络编码重传方案。该方案针对 SVC 视频流提出了基于 SVC 缓存的 D2D-RAIDNC 模型,使得多个协作发送设备可以同时发送编码包,减少完成时间以及重传次数。同时,综合考虑视频流的终端解码率、链路丢包率以及顶点效用值等因素设计顶点的权重,从而选取最优的重传方案。考虑物理层的功率分配问题设计基于跨层网络编码的资源调度算法,在不增加完成时间的同时实现吞吐量的优化。仿真实验表明,在 FC-D2D 网络中,NCVS 方案在一定程度上降低了完成时间、解码时延以及重传次数。

参考文献:

[1] SCHWARZ H, MARPE D, WIEGAND T. Overview of the scalable video coding extension of the H. 264/AVC standard[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2007, 17(9): 1103-1120.

[2] 岳雯雁, 赵凤, 朱琦. D2D 网络中 SVC 视频文件的缓存策略[J]. 南京邮电大学学报(自然科学版), 2022, 42(2): 54-65.

YUE Wenyan, ZHAO Su, ZHU Qi. SVC video caching strategy in D2D network[J]. Journal of Nanjing University of Posts and Telecommunications (Natural Science Edition), 2022, 42(2): 54-65. (in Chinese)

[3] AHLSEWEDE R, CAI N, LI S Y R, et al. Network information flow[J]. IEEE Transactions on Information Theory, 2000, 46(4): 1204-1216.

[4] WANG C C, SHROFF N B. Beyond the butterfly — a graph-theoretic characterization of the feasibility of network coding with two simple unicast sessions[C] // IEEE International Symposium on Information Theory. 2008: 121-125.

[5] DOUIK A, SOROUR S, TEMBINE H, et al. A game theoretic approach to minimize the completion time of network coded cooperative data exchange[C] // IEEE Global Communications Conference. 2014: 1583-1589.

[6] BEN RHAÏEM O, CHAARI FOURATI L, AJIB W.

- Network coding-based approach for efficient video streaming over MANET[J]. *Computer Networks*, 2016, 103: 84–100.
- [7] VUKOBRATOVIC D, STANKOVIC V. Unequal error protection random linear coding strategies for erasure channels[J]. *IEEE Transactions on Communications*, 2012, 60(5): 1243–1252.
- [8] WANG S J, GONG C, WANG X D, et al. An efficient retransmission strategy for wireless scalable video multicast[J]. *IEEE Wireless Communications Letters*, 2012, 1(6): 581–584.
- [9] TRASKOV D, MEDARD M, SADEGHI P, et al. Joint scheduling and instantaneously decodable network coding[C]//GLOBECOM. 2009: 1–6.
- [10] DOUIK A, SOROUR S, AL-NAFFOURI T Y, et al. Delay reduction in multi-hop device-to-device communication using network coding[J]. *IEEE Transactions on Wireless Communications*, 2018, 17(10): 7040–7053.
- [11] 王磊, 林煌达, 康彬, 等. 时延约束下采用 SVC 编码的 D2D 协作视频多播传输方案[J]. *信号处理*, 2019, 35(3): 369–377.
WANG Lei, LIN Huangda, KANG Bin, et al. D2D cooperative video multicast transmission based on delay constraint and SVC streaming[J]. *Journal of Signal Processing*, 2019, 35(3): 369–377. (in Chinese)
- [12] AL-ABIAD M S, DOUIK A, HOSSAIN M J. Coalition formation game for cooperative content delivery in network coding assisted D2D communications[J]. *IEEE Access*, 2020, 8: 158152–158168.
- [13] ZOU W X, CHENG T F, LAN S. An improved real-time video stream differentiated transmission algorithm based on IDNC[C]//International Conference on Computing, Networking and Communications (ICNC). 2020: 193–197.
- [14] DOUIK A, SOROUR S, AL-NAFFOURI T Y, et al. Delay reduction in multi-hop device-to-device communication using network coding[C]//International Symposium on Network Coding (NetCod). 2015: 6–10.
- [15] DOUIK A, SOROUR S, AL-NAFFOURI T Y, et al. Instantly decodable network coding: from centralized to device-to-device communications[J]. *IEEE Communications Surveys & Tutorials*, 2017, 19(2): 1201–1224.
- [16] SAIF M, DOUIK A, SOROUR S. Rate aware network codes for coordinated multi base-station networks[C]//IEEE International Conference on Communications. 2016: 1–7.
- [17] DOUIK A, SOROUR S, AL-NAFFOURI T Y, et al. Rate aware instantly decodable network codes[J]. *IEEE Transactions on Wireless Communications*, 2017, 16(2): 998–1011.
- [18] AL-ABIAD M S, DOUIK A, SOROUR S. Rate aware network codes for cloud radio access networks[J]. *IEEE Transactions on Mobile Computing*, 2019, 18(8): 1898–1910.
- [19] WANG X M, YUEN C, XU Y L. Coding-based data broadcasting for time-critical applications with rate adaptation[J]. *IEEE Transactions on Vehicular Technology*, 2014, 63(5): 2429–2442.
- [20] AL-ABIAD M S, HASSAN M Z, DOUIK A, et al. Low-complexity power allocation for network-coded user scheduling in fog-RANs[J]. *IEEE Communications Letters*, 2021, 25(4): 1318–1322.
- [21] AL-ABIAD M S, HOSSAIN M J, SOROUR S. Cross-layer cloud offloading with quality of service guarantees in fog-RANs[J]. *IEEE Transactions on Communications*, 2019, 67(12): 8435–8449.
- [22] AL-ABIAD M S, SOROUR S, HOSSAIN M J. Cloud offloading with QoS provisioning using cross-layer network coding[C]//IEEE Global Communications Conference. 2018: 1–7.
- [23] AL-ABIAD M S, DOUIK A, SOROUR S, et al. Throughput maximization in cloud-radio access networks using cross-layer network coding[J]. *IEEE Transactions on Mobile Computing*, 2022, 21(2): 696–711.
- [24] KARIM M S, SADEGHI P, SOROUR S, et al. Instantly decodable network coding for real-time scalable video broadcast over wireless networks[J]. *EURASIP Journal on Advances in Signal Processing*, 2016, 2016: 3.
- [25] PARK J S, GERLA M, LUN D S, et al. Codecast: a network-coding-based ad hoc multicast protocol[J]. *IEEE Wireless Communications*, 2006, 13(5): 76–81.

(责任编辑:李小溪)