Min-Cost Multicast Streaming with Network Coding in Edge Computing Networks

Yeqiao Hou[1], Han Yang[3], Xianglong Li[1], Ling Deng[4], Liang Du[4], Zongpeng Li[12], Yining Li[23]
[1]School of Computer Science, Wuhan University, China
[2]Quan Cheng Laboratory, China
[3]Institute for Network Sciences and Cyberspace, Tsinghua University, China
[4]China Unicom Greater Bay Area Innovation Research Institute, China

***Abstract*—We study edge computing networks where selected routers are equipped with heterogeneous processors such as CPUs, GPUs and FPGAs, to enable in-network computing capabilities. Leveraging the IPv6-SRv6 paradigm, programmable networking integrates seamlessly with these hardware innovations, supporting application-defined compute-and-forward operations. Video streaming, a dominant application in edge networks, often involves edge routers performing replication, encoding/decoding, and transcoding tasks enroute to end users. These streams compete for limited computing and transmission resources, which is modeled as an online social welfare maximization problem. The proposed solutions employ key techniques that include: Firstly, a compact exponential algorithm framework that tackles the non-packing and non-covering structure of edge welfare maximization, by translating explicit algebraic constraints into implicit geometric constraints; Secondly, a primal-dual method that breaks down edge welfare optimization into simpler multi-round auctions; and Finally, a dual oracle leveraging network coding algorithms for efficient multicast optimization. Simulations studies demonstrate 61.2% social welfare improvement compared to benchmark algorithms, validating efficient coordination of resources while maintaining compatibility with edge infrastructure under dynamic workloads.**

***Index Terms*—Live Video Streaming, Edge Computing Networks, Online Auction, Multicast Algorithms, Network Coding**

I. Introduction

Live video streaming (LVS) applications are becoming increasingly common on the Internet. They are witnessed in sectors from online education and remote work to trade shows and social gatherings [1]. LVS is transforming the way we live and work, as new live video applications continue to emerge. These include 3D volumetric video interactions, robotic teleoperation, and metaverse, among others. The live streaming industry has experienced significant growth and is projected to reach a value of $224 billion by 2028, according to the Cisco Visual Networking Index, which predicts that mobile video traffic will account for 79% of global mobile data usage. This surge in live video traffic has driven by user demand, with audiences now expecting high-quality, low-latency video experiences.

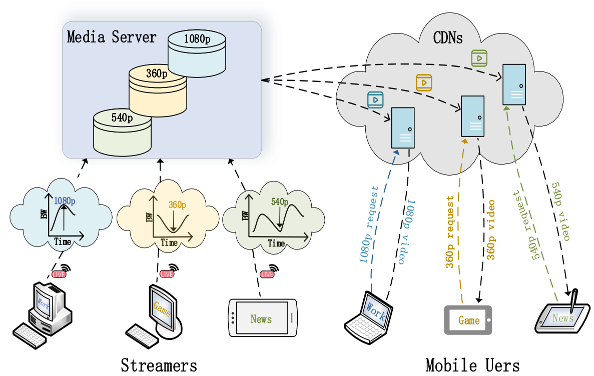

Figure 1: LVS system.

A typical LVS system consists of a streamer, a media server, and multiple Content Delivery Network (CDN) nodes, As shown in Fig. 1. The streamer captures the live video and streams it, at realtime, to a centralized media server, using a low-latency transport protocol such as RTP. The media server then receives the high-quality raw video for encoding and transcoding. Finally, the CDN nodes deliver the requested videos to numerous users on the network [2]. LVS introduces a new and substantial challenge to Internet service delivery, in contrast to traditional video-on-demand services and contemporary video telephony. LVS is both latency-sensitive and bandwidth-intensive. Meanwhile, the modern Internet is becoming increasingly heterogeneous, with network links like fiber, WiFi, LTE, satellite and 5G/6G mobile networks, each exhibiting diverse bandwidth and latency characteristics. Classical methods face challenges that include overly conservative bandwidth estimation [3], inability to track real-time network dynamics, and lack of agility

The corresponding authors are Zongpeng Li and Yining Li. This work has been supported in part by China Key RG Program (2022YFB2901300), Quancheng Lab (QCLZD202304), and China Unicom Guangdong.
ISBN 978-3-903176-72-0 © 2025 IFIP

in adapting to unforeseen network conditions.

Business growth leads to higher bandwidth costs from the CDN service. At the same time, live streaming service providers aim to minimize these costs while maximizing the streaming bitrate, which appears paradoxical. In fact, The concept of edge computing networks (ECN) has been introduced in response to this paradigm shift [4]. It assumes that multiple computing nodes are located within the edge network, each capable of providing service instances and bring service from remote servers to nearby edge devices. This approach helps enable higher network efficiency, lower power consumption, and lower latency.

In data networks, packet transmission can be modeled as the flow of bit streams, known as information flows. Unlike classic network flows, information flows can be buffered, forwarded, replicated, and coded. Previous research [5] has demonstrated that incorporating network coding (NC)—coded information flows at intermediate network nodes can enhance the capacity of a multicast network. NC enables the achievement of optimal multicast throughput through polynomial-time algorithms. Conversely, attaining optimal throughput with routing (Steiner tree packing) is an NP-complete problem [6]. NC offers notable advantages, particularly in terms of adaptability and robustness. What sets it apart is the opportunistic linear combination carried out throughout the network, rather than being confined to the source node. This approach is particularly effective when nodes have incomplete information of the overall network state. In practice, NC algorithms must address the limitations of simplistic network models commonly found in information theory while meeting the unique demands of multimedia streaming, particularly the strict delay constraints.

This work investigates min-cost multicast with NC and dynamic pricing in ECN to optimize real-time video streaming. Through auction mechanisms [7], we address supply-demand fluctuations by allocating resources to highest-value videos. We propose LVNC, an auction-driven online algorithm that jointly determines video bid acceptance (for encoding/transmission) and orchestrates efficient multicast flow/billing management. Our goal is to enhance overall ECN efficiency by maximizing social welfare, rather than solely focusing on increasing the revenue of video operators or the payoff of the ECN operator. The key contributions of this work are as follows:

- We tackle the ECN social welfare maximization problem by first formulating it as an integer linear program (ILP). The ILP includes linear constraints, such as resource capacities, and binary variables, such as whether a video is serviced. Even in the offline setting, the problem remains NP-hard. The challenge further escalates when videos arrive online, requiring real-time decision making without knowledge of future inputs. we design an efficient primal-dual algorithm that simultaneously handles multicast routing with NC and resource pricing decisions, involving an exponential number of variables.
- We introduce LVNC (**L**ive **V**ideo Streaming with **N**etwork **C**oding), an online algorithm for routing and pricing, which integrates three synergistic components: video admission control, the design of dynamic pricing functions, and minimum-cost multicast streaming with network coding. (i) By relaxing integer variables and introducing dual variables (interpreted as resource prices), LVNC evaluates the cost of multicast streaming with network coding via subroutine A_{lp}. A video request is accepted only if its estimated streaming cost is lower than its bid. (ii) To design the pricing function, exponential pricing functions dynamically adjust node/edge resource prices over time. Parameters of these functions are optimized using historical data to balance real-time supply-demand variations. (iii) For network coding and multicast routing, A_{lp} formulates optimal multicast flow with network coding as a linear program (LP), determining encoded node placement and content distribution based on user demands. The LP is solved via either the Lagrange duality method (with sub-gradient optimization) or standard LP solvers, where the latter offers precision at higher computational cost.
- We prove that LVNC guarantees an effective competitive ratio and runs in polynomial time, and extensive simulations indicate that LVNC outperforms the three benchmark algorithms in terms of social welfare of the video ECN ecosystem: Steiner_A (at least 61.2%), Steiner_NA (129.1%), Greedy (251.2%).

In the rest of the paper, Section II reviews related work; Section III introduces the system model; Section IV details LVNC's design and analysis; Section V evaluates performance; and Section VI concludes.

II. RELATED WORK

LVS in ECN. While existing bitrate adaptation algorithms [8, 9] address live streaming demands, their heuristic designs often compromise robustness. For instance, [10] dynamically optimizes bitrate and datacenter assignments to reduce provider costs, yet incurs excessive latency. Recent computing power network initiatives [4] emphasize network-edge-cloud collaboration to enhance resource utilization in edge computing. Despite economic incentives for adopting

ECN services [11], judicious allocation of limited edge transmission/computation resources remains critical to avoid performance bottlenecks.

Auction and Dynamic Pricing. This work integrates dynamic pricing and combinatorial auctions to optimize ECN resource allocation for multicast video streaming. Existing approaches include multi-round sealed combinatorial auctions [12] that maximize social welfare but suffer from bidder attrition and resource underutilization, as well as two-step revenue-focused mechanisms [13] vulnerable to strategic bid manipulation due to inter-round information leakage. Recent literature addresses these challenges through truthful bidding mechanisms [14] paired with dynamic pricing models, where historical data predicts marginal prices to align participant valuations with real-time supply-demand dynamics.

NC Meets Multimedia. NC ensures high-probability recovery of original packets at destination nodes through encoded transmissions. While foundational studies originate in information theory, NC has proven effective for multimedia streaming by enhancing throughput, resource utilization, and error resilience. Applications span peer-to-peer networks and wireless broadcast systems, where fountain/Reed-Solomon codes [15] enable distributed encoding at the cost of increased latency. Hybrid approaches like random NC with push mechanisms [16] and rateless re-encoding address latency and packet loss challenges. Multicast optimizations further exploit NC's path diversity to strengthen robustness against network dynamics [17], demonstrating its adaptability across heterogeneous environments.

III. Model and Problem Formulation

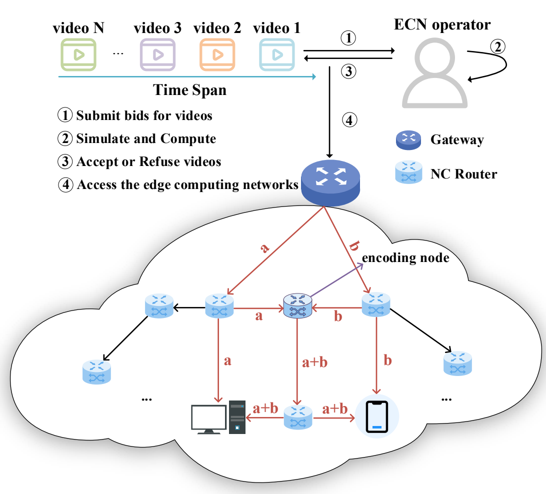

Figure 2: Live video streaming with network coding in ECN.

A. Auction Setup and Agent Interactions

ECN resource operator. We focus on an ECN system overseen by a central operator responsible for managing distributed routers equipped with network coding capabilities. These routers $u \in [U]$, along with the transmission links $e \in [E]$, form a network infrastructure offering limited computational and bandwidth resources, C_u and C_e). The ECN operator allocates these resources to video service providers through a competitive auction mechanism.

Video operators. We examine an application scenario in which each video operator $i \in [I]$ submits a bid with a bidding price b_i and a desired throughput λ_i for streaming of video i. The streaming starts at time t_{ia} and finishes at t_{ib}. Furthermore, each video streaming i serves a set of users R_{ij} who require the same video, with each streaming corresponding to j users. We assume that each video submits only one bid at a time. We use B_i to denote the bidding language of the video operator:

$$B_i = \{b_i, \lambda_i, \{t_{ia}, t_{ib}\}, R_{ij}\}.$$

System Overview. As shown in Fig. 2, we assume that video i enters the ECN system through a common gateway S, distributing the video to target users via efficient multicast. Our model features two interacting entities in an auction: the ECN resource operator as the auctioneer, and video providers as bidders. The ECN-Video interaction consists of three steps, *First*, incoming video streams follow a Poisson arrival process, where the arrival rate determines the system's traffic load. *Second*, The ECN resource operator invokes an algorithm to compute the multicast transmission flow with network coding for the videos, and obtain the coding cost and transmission cost. If the video revenue remains non-negative, both parties formalize their agreement and proceed to the third stage of interaction; otherwise, the process is terminated. *Third*, The ECN operator allocates transmission and coding resources to an accepted video i according to the strategy computed in Step 2, the video i is sent to the group of users R_{ij} with same needs, and finally collect payments from video operators.

Auction Preliminaries. We define binary variables $x_i \in \{0, 1\}$: when i's bid is accepted, x_i equals 1, and 0 otherwise. Rejected bids can be queued and resubmitted later. When a bid from a video operator arrives, let w_i represent the true valuation of operator i and p_i denote its payment. The payoff for video operator i is $y_i(b_i) = \sum_{i\in[I]} w_i x_i - \sum_{i\in[I]} p_i$. Meanwhile, the utility for ECN operator is the total payment of all winner videos, $\sum_{i\in[I]} p_i$. Together, these two utilities contribute to the overall ECN social welfare. In reality, users tend to maximize their individual payoffs, and may attempt to manipulate/misreport their true valuations to achieve higher payoffs. Our focus is not on maximizing either the videos' payoffs or the ECN resource operator's revenue; instead, we aim to maximize social welfare of the ECN ecosystem, for which eliciting truthful bids is important.

Definition 1 (Truthful Auction)**.** *An auction is deemed* truthful *if each video achieves the highest payoff by honestly reporting their true valuation,* i.e., $y_i(w_i) \geq y_i(b_i), \forall w_i \neq b_i$ *[14].*

Definition 2 (Individual Rationality)**.** *An auction is* individually rational *if every video's payoff is guaranteed to be non-negative. [14].*

Definition 3 (Social Welfare)**.** *The* social welfare *in the ECN market is the aggregated video payoff* $\sum_{i\in[I]} w_i x_i - \sum_{i\in[I]} p_i$ *plus* $\sum_{i\in[I]} p_i$*, which is equivalent to* $\sum_{i\in[I]} w_i x_i$*.*

Table I: Notation

Inputs	Description
I, T	# of video streaming, time slots
i, t	indexes of video streaming and time slots
R_{ij}	set of users j for same video i
S	gateway, the source node of the edge computing networks
b_i	bidding price of video i
λ_i	the throughput for video streaming i.
t_{ia}, t_{ib}	the streaming i start time and job completion deadline
C_u	capacity in node u from U
C_{uv}	capacity of edge uv from E
Decisions	**Description**
x_i	video i' bid wins (1) or not (0)
f_i	each valid multicast flow from i to j in the edge computing networks
$f_{it}(uv)$	# of bandwidth consumed when video i is transmitted on edge (u, v)
$f_{it}(u)$	# of coding resources consumed when video i is encoded on router u.
NC_i	the polytope to support the multicast transmission of chosen video i.
$\alpha_t(l)$	the unit price for l at time t
y_i	utility of video i if admitted

B. Problem Formulation

Social Welfare Maximization. When video i arrives at the ECN system from the Internet through the gateway, the largest social welfare routers are selected to perform network coding multicast transmission. In the ECN, the gateway is the multicast source S, and the user is the receiver R_{ij} of video i. Intuitively, C denotes the capacities, and f_i represents the network flow for video stream i. $f_{it}(uv)$ indicates the bandwidth used to transmit video i over edge (u, v), while $f_{it}(u)$ represents the computing resources consumed to encode video i at router u. Under the assumption of truthful bidding, social welfare maximization is subject to: (i) edge capacity constraints and (ii) node capacity constraints, as follows:

$$\text{Maximize} \sum_{i\in[I]} b_i x_i \tag{1}$$

$$\text{subject to: } \sum_{i\in[I]} f_{it}(u) \leqslant C_u, \forall u \in [U], \forall t \in [T] \tag{1a}$$

$$\sum_{i\in[I]} f_{it}(uv) \leqslant C_{uv}, \forall (u,v) \in [E], \forall t \in [T] \tag{1b}$$

$$x_i, f_{it} \in NC_i, \forall i \in [I] \tag{1c}$$

$$x_i \in \{0,1\}; f_{it}(u), f_{it}(uv) \geq 0, \forall i, \forall u, \forall (u,v), \forall t \tag{1d}$$

Constraints (1a) and (1b) are *capacity constraints:* (1a) indicates that the cumulative network coding resource consumption at a node $u \in [U]$ should not surpass the nodal capacity. (1b) states the aggregated video transmission resource consumption cannot exceed the capacity of each edge $(u, v) \in [E]$.
Constraint (1c) is the *multicast with NC constraint*, which ensures that each flow adheres to the network coding requirements, enabling the multicast transmission of the selected video i. The polytope NC_i defines a feasible NC-encoded multicast flow, from the source S to each receiver R_{ij}.

Definition 4 (The NC_i polytope)**.** *For flow* f_{ij}*, each node has flow conservation requirements, and the multicast flow* f_i *takes link-wise max over conceptual flows* f_{ij}*. The fact that problem (1) models social welfare maximizing multicast follows from Theorem 1 of [18].* NC_i *is the polytope:*

$$\text{NC}_i : \begin{cases} \sum_{\{v|(u,v)\in E\}} (f_{ijt}(uv) - f_{ijt}(vu)) = 0, \\ \qquad\qquad \forall i, \forall j, \forall (u,v), \forall t \\ f_{it}(uv) \geq f_{ijt}(uv), \forall i, \forall j, \forall (u,v), \forall t \\ x_i \lambda_i \leqslant f_{ijt}(R_{ij} S), \forall i, \forall j, \forall t. \end{cases}$$

i flow conservation for conceptual network flow from gateway to specified users $j \in R_{ij}$. $\sum_{\{v|(u,v)\in E\}} (f_{ijt}(uv) - f_{ijt}(vu)) = 0$ means total incoming flow rate at a node equals its total outgoing flow rate at time $t \in [t_{ia}, t_{ib}]$.

ii The multicast flow f_i on each edge is defined as the maximum flow among all users at any time $t \in [t_{ia}, t_{ib}]$, given by $f_{it}(uv) = \max_j f_{ijt}(uv)$. This follows from the feasibility condition of NC-encoded multicast.

iii We introduce virtual directed links with infinite capacity from each destination node $j \in R_{ij}$ to the source S. These virtual links ensure flow conservation at the source S and destination nodes R_{ij}. Therefore, throughput of video streaming λ_i not to exceed any of these network flow rates to ensure that the users R_{ij} can receive the video streaming i smoothly.

Challenges. Making a joint decision efficiently and effectively is difficult, since external inputs to the problem arrive at realtime, and the objective values of the problem cannot be decoupled over time due to time-correlated adjustment costs. Furthermore, even in offline setting, problem (1) is an ILP that can be proven NP-hard [19].

IV. Algorithm Design

A. Algorithm Overview

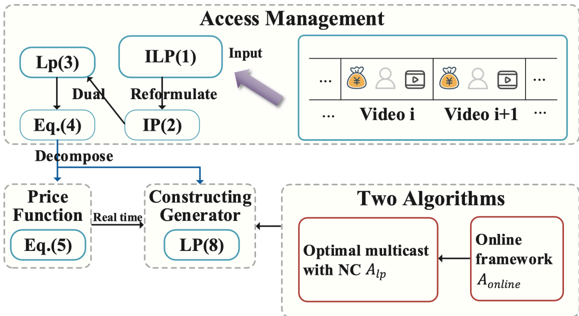

Figure 3: The LVNC framework and process.

In this section, we introduce the framework and ideas of LVNC in Fig. 3, which can be divided into three parts:

- Each incoming video i is formulated as ILP (1), converted via compact exponential optimization into ILP (2) with auxiliary variables. The primal-dual method and complementary slackness theorem derive dual problem (3) and equation (4).
- Upon video arrival, the ECN operator computes transmission/NC costs via A_{lp}, then accepts bids only if social welfare is positive. Real-time pricing adapts dynamically using historical allocation data through price function (5).
- Problem (8)'s optimal multicast flow is solved as an LP via solver or Lagrangian duality with subgradient optimization. While both yield optimal solutions, computational complexity differs. A_{lp} determines flow/routing using dual variable $\alpha_t(l)$ for NC-oriented multicast.

B. Access Management

Problem Reformulation. The problem becomes more complex when we assume each flow to be a valid network flow and enable multicast transmission with network coding for the selected video i. To tackle these issues, we adopt the primal-dual algorithm approach. we transform the original problem (1) into a compact-exponential ILP with a packing structure. Let NC_i represent the set of feasible multicast flows for video i, with each flow satisfying constraints (1a) and (1b). Furthermore, we redefine variable x_i as a binary variable x_{if_i}, where $f_i \in NC_i$, indicating whether video i selects a valid multicast flow with network coding. In addition, we made a bold attempt to introduce $l \in [U+E]$ to represent the capacity limits at both the nodes u and edges (u,v) of the ECN system.

$$\text{Maximize} \sum_{i\in[I]} \sum_{f_i \in NC_i} b_{if_i} x_{if_i} \tag{2}$$

$$\text{s.t.:} \sum_{i\in[I]} \sum_{f_i: l, t\in f_i} f_i(l) x_{if_i} \leqslant C_l, \forall l \in [U+E], \forall t \in [T] \tag{2a}$$

$$\sum_{f_i \in NC_i} x_{if_i} \leqslant 1, \forall i \in [I] \tag{2b}$$

$$x_{if_i} \in \{0,1\}, \forall i \in [I], f_i \in NC_i \tag{2c}$$

Each flow is an index corresponding to a specific element in polytope NC_i. Constraints (2a) are equivalent to (1a) and (1b), Constraint (2c) includes all other constraints from problem (1). In summary, solving problem (1) is equivalent to solving problem (2), which may involve an exponentially large number of variables.

Dual Problem. To solve (2) efficiently, we apply a method based on the dual approach, which leads to the formulation of the dual problem (3). This involves relaxing the integer variables $x_{if_i} \in \{0,1\}$ to $x_{if_i} \geq 0$. We define dual variables $\alpha_t(l)$ and y_i for constraints (2a) and (2b) respectively. The dual LP of the relaxed (2) is:

$$\text{minimize} \sum_{i\in[I]} y_i + \sum_{t\in[T]} \sum_{l\in[U+E]} C_l \alpha_t(l) \tag{3}$$

$$\text{s.t.:} y_i \geq b_{if_i} - \sum_{t\in f_i} \sum_{l\in f_i} f_i(l)\alpha_t(l), \forall i \in [I], \forall f_i \in NC_i \tag{3a}$$

$$y_i, \alpha_t(l) \geq 0, \forall i \in [I], \forall t \in [T], \forall l \in [U+E] \tag{3b}$$

Rationale Behind. We define the dual variable $\alpha_t(l)$ as the unit price of bandwidth resources on $l \in [U+E]$ at time t. As a result, the expression $\sum_{t\in f_i} \sum_{l\in f_i} f_i(l)\alpha_t(l)$ reflects the video i's total multicast cost. The right-hand side (RHS) of (3a) represents the utility of video i minus the total cost. The corresponding constraint (3a) becomes tight due to complementary slackness when video i is accepted. Given that $y_i \geq 0$, we have:

$$y_i = \max\{0, \max_{f_i \in NC_i} \text{RHS of } (3a)\}. \tag{4}$$

The payoff of video i is y_i when selecting an optimal NC-coded multicast flow $f^* = \max_{f_i \in NC_i}$ RHS of $(3a)$.

C. Price Function Design

Resource prices are designed to reflect demand-supply dynamics, with transmission and computing price functions designed using variables: $\delta_t(l)$, representing the capacity allocated at edge (u,v) and node u at t. We design the marginal prices $\alpha_t(l)$ as functions of $\delta_t(l)$. A suitable price function must satisfy the following conditions: (i) The prices are constrained within a range, with a lower bound representing no resource usage and an upper bound indicating complete resource depletion. (ii) $\alpha_t(l)$ increases as $\delta_t(l)$ increases. To achieve this, we define the price function to exhibit exponential growth:

$$\alpha(\delta_t(l)) = (\xi_l)^{\frac{\delta_t(l)}{C_l}} - 1 \tag{5}$$

where $\xi_l = max_i\{\frac{b_{if_i}}{\sum_l f_i(l)}\} + 1$.

When resources are not allocated, we have $\alpha(\delta_t(l))$ = 0, whereas when the capacity on edges (u, v) and nodes u is exhausted at time t, denoted as $\delta_t(l) = C_l$, we have $\alpha_t(l) = \xi_l - 1$, leading to $b_{if_i} - \sum_{t\in f_i}\sum_{l\in f_i} f_i(l)(\xi_l - 1) < 0$. Without assumptions on edge or node capacities, scenarios may arise where transmission resources on edges (u, v) and computing resources at nodes u remain abundant. In such cases, $\sum_l f_i(l)(\xi_l - 1) \to 0$ as $\xi_l - 1 \to 0$. Consequently, with the objective of ensuring $b_{if_i}/\sum_l f_i(l)<\xi_l-1$, we define $\xi_l = \max_i\left\{\frac{b_{if_i}}{\sum_l f_i(l)}\right\} + 1$.

D. Optimal Multicast Transmission

After investigating resource pricing methods, our focus shifts to finding the optimal multicast transmission with network coding f^* for accepted video i. Our objective function for problem is further simplified as follows:

$$\min_l cost(t_{ia}, t_{ib}, l) = \sum_{t\in f_i}\sum_{l\in f_i} f_i(l)\alpha_t(l) \tag{6}$$

At this time, the network $G = (U, E)$ contains both point capacity C_u,$\forall u$ and edge capacity C_{uv},$\forall(u, v)$. We introduce some virtual points to convert all the capacity on the point to the edge. For example, The virtual edge (u, u') can replace the point u in Fig. 4 and the capacity $f_i(u)$ at point u disappears in the new network G' and is replaced by the capacity $f_i(uu')$ at edge (u, u'). Now we have a new network topology $G' = (U', E')$. Essentially, the objective function (6) is further optimized into:

$$minimize \sum_{t\in f_i}\sum_{(u,v)\in f_i} f_i(uv)\alpha_t(uv) \tag{7}$$

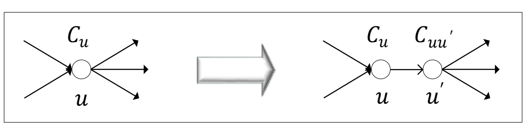

Figure 4: Capacity Conversion at Point.

We assume that the streaming rate is constant across the temporal domain when a video i is multicast. Therefore, we drop the notion of time in the dual subproblem. let the marginal price for link (u, v) at t be $\sum_{t\in f_i}\alpha_t(uv) = \beta_{uv}$. The dual subproblem is:

$$\text{minimize} \sum_{(u,v)} f_i(uv)\beta_{uv} \tag{8}$$

subject to: $$\sum_{\{v|(u,v)\in E\}} (f_{ij}(uv) - f_{ij}(vu)) = 0, \forall i, \forall j, \forall(u, v) \tag{8a}$$

$$f_i(uv) \geq f_{ij}(uv), \forall i, \forall j, \forall(u, v) \tag{8b}$$

$$\lambda_i \leqslant f_{ij}(R_{ij}S), \forall i, \forall j \tag{8c}$$

$$f_i(uv) \geq 0, f_{ij}(uv) \geq 0, \forall i, \forall j, \forall(u, v) \tag{8d}$$

To minimize resource cost, the crux lies in determining the optimal position for network coding about problem (8). Thus, we come up with the algorithm discussed in Alg. 2 to resolve the above problem. Problem (8) is a standard linear program that can be solved using an LP solver, or through Lagrangian duality and sub-gradient optimization. Both methods yield the optimal solution, but their time complexities differ.

E. Algorithm Design

Our LVNC algorithm determine the optimal NC-coded multicast flow f^* for video i by invoking the subalgorithm A_{lp} (Alg. 1 , line 4). If the video's return is positive, the ECN operator configures the flow and updates the prices and usage of l based on f^*. The bid for video i' is accepted once the corresponding price p_i is paid, and x_i is set to 1 (lines 5-8). Conversely, the video's bid is rejected (line 10).

Algorithm 1: A Primal-Dual Online Framework A_{online}

Input: I, T, B_i, R_{ij}, C_{uv}, C_u, $G = (U, E)$, $\forall i, \forall j, \forall(u, v), \forall u, \forall t$
Output: x_i
1 Define prices function $\alpha(\delta_t(l))$ according to (5)
2 Initialize $x_i = 0, \delta_t(l) = 0, \alpha_t(l) = \alpha_0(l) = 0, \forall i, \forall l, \forall t,$; Let $x_{if_i} = 0, \forall f_i \in NC_i$, by default
3 **for** *every arrival of new video* i **do**
4 (y_i, p_i, f^*)=$A_{lp}(B_i, C_l, \{\delta_t(l)\})$
5 **if** $y_i > 0$ **then**
6 Update $\delta_t(l) = \delta_t(l) + \sum_i f_i(l), \alpha_t(l) = \alpha(\delta_t(l))\forall i, \forall l, \forall t$
7 Charge p_i for video i
8 Accept video i's bid, set $x_i = 1$ and launch video i according to the optimal multicast transmission flow f^* with NC in ECN
9 **else**
10 Reject video i and $x_i = 0$

We employ the Gurobi LP solver (Alg. 2, line 3) to determine the optimal multicast flow with NC. The computation considers the necessary bandwidth capacity for both nodes and edges to process the video (line 4). After filtering out the nodes and edges that meet these conditions, we compute the total cost based on marginal prices (line 7). The payoff for video i is the difference between the bid and the cost (line 8).

Algorithm 2: Subframework for Optimal Flow with NC of Video i: A_{lp}

Input: B_i, $C_l, G = (U, E)$, $\delta_t(l)$
Output: f^*, y_i
Initialize $f^* = 0, y_i = 0, \forall i$
while *a tuple* $\{S, f_i(l), f_{ij}(l), C_l, R_{ij}, \lambda_i\}$ **do**
  Calculate multicast flow f^* with NC use LP Solver
  if $f_i(l) + \delta_t(l) \leqslant C_l, \forall l$ **then**
    Preserve edges and nodes on the flow with NC
  for *each* $t \in [t_{ia}, t_{ib}]$ **do**
    $Cost_i = \sum_l \sum_t f_i(l)\alpha_t(l)$
  Compute $y_i = b_{il} - Cost_i$
return f^*, y_i

F. Theoretical Analysis

Theorem 1. *(Truthfulness). The online auction in LVNC is truthful.*

Proof. See Appendix A in technical report [20]. □

Theorem 2. *(Individual Rationality). The online auction in LVNC is individually rational.*

Proof. See Appendix B in technical report [20]. □

Theorem 3. *(Budget-Balanced). The online auction in LVNC is Budget-Balanced.*

Proof. See Appendix C in technical report [20]. □

Theorem 4. *(Polynomial Time). The time complexity of LVNC is* $\mathcal{O}(IT \cdot (|E| \cdot K \cdot J)^3)$.

Proof. See Appendix D in technical report [20]. □

Theorem 5. *(Competitive Ratio). The competitive ratio is a metric used to assess the performance of online algorithms relative to the offline optimum. It is an upper bound on the ratio between the achieved social welfare and the optimal social welfare. For problem (1), we achieve a competitive ratio of* $2 \log \xi$, *Where* $\xi = max_l\{\xi_l\}$.

Proof. See Appendix E in technical report [20]. □

V. Performance Evaluation

A. Simulation

Simulator Setup. We evaluate LVNC by simulating it on both synthetic and real networks, comparing with three benchmark algorithms. We utilize `igraph` to generate synthetic networks, with the Forest-fire algorithm producing 500 nodes, including 150 users. We further assess the performance on a real-world network, FilmTrust, a small dataset with a density of 1.14%, scraped from the entire FilmTrust website. For each time slot, the number of videos arriving online follows a Poisson process, with an average rate of 2 videos per second. The number of time slots is set to last $60s$.

Baselines. We evaluate the performance of our LVNC algorithm with three benchmark algorithms:

- Steiner_A: A Steiner tree algorithm [21] can find a connected subtree containing all end nodes such that the sum of the edge weights of this subtree is minimal. We introduce auction and admission control in Steiner, accepting only bids with a positive profit.
- Steiner_NA: We extend Steiner_A to Steiner_NA to account for the impact of lacking auction and admission control on algorithm performance.
- Greedy: The algorithm [22] relies on the principle of a greedy approach to find edges at the lowest possible cost.

B. Simulation Results

Impact of Capacity. Fig. 5 shows the impact of the edge's computing capacity on profit. The max number of videos processed by a single edge is set to 2, 4, 6, and 8, respectively. As processing capability increases, social welfare improves for all algorithms. However, the gap in normalized social welfare between LVNC and Steiner_A narrows, from 0.32 to 0.17, because network coding's advantage is more pronounced with limited edge resources, allowing video transmission without fully occupying bandwidth. Similarly, Fig. 6 considers the effect of node capacity. In the best case, LVNC's social welfare increases by 61.2%, 129.1%, and 251.2% compared to Steiner_A, Steiner_NA, and Greedy, respectively. However, this gap narrows over time, and network coding shows a greater advantage in resource-constrained environments.

Multicast Transmission. Fig. 7 illustrates the impact of user count on profits. We categorize the 150 users into three groups based on the proportion of viewers: 0-30%, 30-60%, and 60-100%. With fewer users, the scenario resembles unicast, and as all users participate, it becomes more like broadcast. We observe that social welfare increases with the number of users for all four algorithms. However, in the multicast range (30-60%), LVNC yields the best performance, with a difference of 0.25 compared to Steiner_A.

Impact of Bidding Price. The bid price is a predictive function based on video size and transmission time. To account for the randomness in user bids, we introduce a random number to simulate bid fluctuations. We consider the impact of 10%, 30%, and

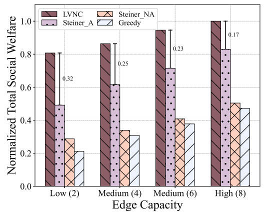

Figure 5: Social welfare under varying edge capacity.

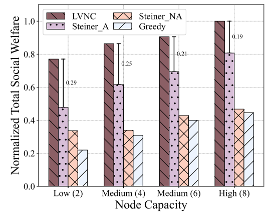

Figure 6: Social welfare under varying node capacity.

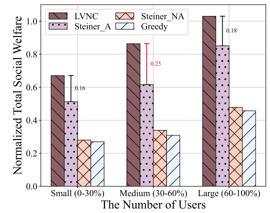

Figure 7: Impact of the number of users.

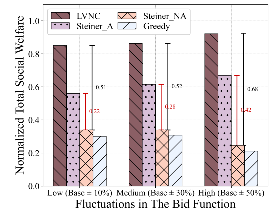

Figure 8: Impact of fluctuations in bids.

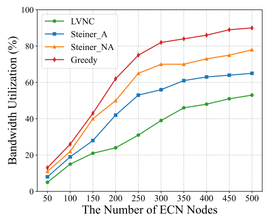

Figure 9: Bandwidth utilization across different ECN sizes.

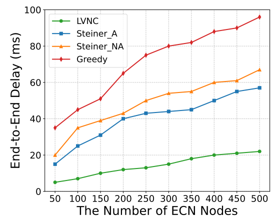

Figure 10: End-to-end delay across different ECN sizes.

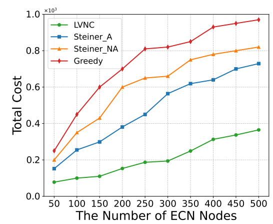

Figure 11: Total cost across different ECN sizes.

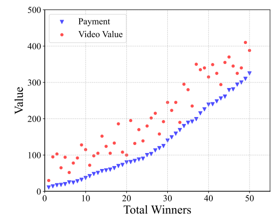

Figure 12: Video value versus payment of winning bid.

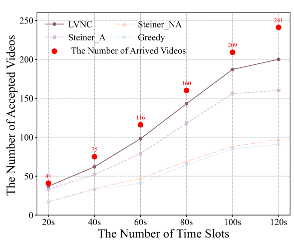

Figure 13: # of videos received in different time slots.

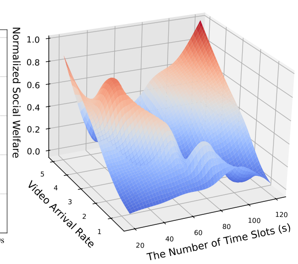

Figure 14: Social welfare achieved.

50% price fluctuations on profit from Fig. 8. The LVNC and Steiner_A algorithms employ a bidding mechanism, while Steiner_NA and Greedy do not. As bid fluctuations increase, the profits of LVNC and Steine_A significantly outperform those of Steiner_NA and Greedy. Specifically, the difference for LVNC rises from 0.51 to 0.68, and for Steiner_A, it increases from 0.22 to 0.42.

Impact of ECN Sizes. Fig. 9 compares bandwidth utilization of LVNC, Steiner_A, Steiner_NA, and Greedy as ECN nodes scale from 50 to 500. While all methods show increased bandwidth usage with network expansion, LVNC consistently achieves the lowest consumption. At 500 nodes, LVNC reduces bandwidth by 18.5% (vs. Steiner_A), 32.1% (vs. Steiner_NA), and 41.1% (vs. Greedy), attributed to its elimination of redundant transmissions. LVNC achieves efficiency through optimized resource allocation and reduced redundancies, while Greedy incurs resource overuse via bandwidth over-provisioning.

Fig. 10 presents the variation in end-to-end delay as the ECN network size expands. As ECN nodes scale to 500, LVNC reduces latency by 61.4% (vs. Greedy), 42.8% (vs. Steiner_NA), and 36.1% (vs. Steiner_A). While all methods exhibit rising delays due to increased relay nodes, LVNC's optimized path selection minimizes redundant hops. In contrast, Steiner-based methods suffer from suboptimal routing, and Greedy incurs overhead from short-sighted cost prioritization.

Fig. 11 illustrates the total cost incurred under different ECN sizes. As the network expands, all methods experience an increase in cost due to the growing number of transfer nodes involved. At 500 nodes, LVNC lowers costs by 36.4% (vs. Greedy), 22.6% (vs. Steiner_NA), and 17.9% (vs. Steiner_A). Cost escalation across methods correlates with network expansion, but LVNC mitigates this through balanced resource allocation. Greedy's over-provisioning and Steiner's inefficient node utilization drive their higher costs.

Individual Rationality. As shown in Fig. 12, a comparison between payments and the worth of videos accepted by LVNC reveals that payment amounts never exceed the corresponding video valuations. This ensures non-negative payoffs for all accepted submissions, as the allocated compensation consistently falls below the perceived value of the content. By maintaining this value-to-payment gap across all selected cases, the mechanism guarantees that creators retain a net benefit, thereby satisfying the individual rationality criterion.

Impact of Time Slots. We consider the number of videos received in different time periods in Fig. 13. LVNC achieves 22.6% higher video acceptance than Steiner_A during stable periods, with gaps widening to 106.5% (vs. Steiner_NA) and 110.7% (vs. Greedy) during peak congestion. While no algorithm achieves full acceptance, LVNC maintains stable submission rates across fluctuating traffic intensities via dynamic optimization, whereas baseline methods that exhibit significant performance degradation during high-volume phases.

Social Welfare. The social welfare optimization shown in Fig. 14 reflects LVNC's adaptability to both temporal constraints and workload variations. Under crowded system conditions, the integration of network coding and ECN frameworks not only enhances LVNC's capability to sustain stable acceptance rates amid fluctuating submission intensities but also improves resource utilization efficiency, leading to a measurable increase in social welfare.

VI. Conclusion

This work is among the first work to study multicast streaming in the emerging ECN paradigm. It aims to optimize LVS multicast transmission by combining network coding with dynamic pricing mechanisms. First, we formulate social welfare maximization as an ILP, refined via compact exponential optimization techniques. A novel exponential pricing function dynamically adjusts resource costs to ensure efficient allocation. Our LVNC framework achieves polynomial-time competitiveness and superior social welfare compared to benchmarks. Future extensions may address ECN challenges like adaptive load balancing and cross-layer resource optimization.

References

[1] H. Zhang, A. Zhou, G. Wang, C. Li, and H. Ma, "Toward Optimal Live Video Streaming QoE: A Deep Feature-Fusion Approach," *IEEE/ACM Transactions on Networking*, vol. 32, no. 3, pp. 2360 – 2375, 2024.

[2] N. Chen, S. Zhang, Z. Ma, Y. Chen, Y. Jin, J. Wu, Z. Qian, Y. Liang, and S. Lu, "ViChaser: Chase Your Viewpoint for Live Video Streaming With Block-Oriented Super-Resolution," *IEEE/ACM Transactions on Networking*, vol. 32, no. 1, pp. 445–459, 2023.

[3] Y. Xu, X. Li, Y. Yang, Z. Lin, L. Wang, and W. Li, "Fedabr: A Personalized Federated Reinforcement Learning Approach for Adaptive Video Streaming," in *Proc. of 22nd IFIP Networking*, 2023, pp. 1–9.

[4] X. Tang, C. Cao, Y. Wang, S. Zhang, Y. Liu, M. Li, and T. He, "Computing Power Network: The Architecture of Convergence of Computing and Networking Towards 6G Requirement," *China Communications*, vol. 18, no. 2, pp. 175–185, 2021.

[5] Z. Li, B. Li, and L. C. Lau, "On Achieving Maximum Multicast Throughput in Undirected Networks," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2467–2485, 2006.

[6] C. Fragouli, J.-Y. Le Boudec, and J. Widmer, "Network Coding: An Instant Primer," *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 1, pp. 63–68, 2006.

[7] L. Lu, J. Yu, Y. Zhu, and M. Li, "A Double Auction Mechanism to Bridge Users' Task Requirements and Providers' Resources in Two-Sided Cloud Markets," *IEEE Transactions on Parallel and Distributed Systems*, vol. 29, no. 4, pp. 720–733, 2017.

[8] B. Wang, M. Xu, F. Ren, C. Zhou, and J. Wu, "Cratus: A Lightweight and Robust Approach for Mobile Live Streaming," *IEEE Transactions on Mobile Computing*, vol. 21, no. 8, pp. 2761–2775, 2021.

[9] Y. Li, X. Zhang, C. Cui, S. Wang, and S. Ma, "Fleet: Improving Quality of Experience for Low-Latency Live Video Streaming," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 9, pp. 5242–5256, 2023.

[10] Y. Zheng, D. Wu, Y. Ke, C. Yang, M. Chen, and G. Zhang, "Online Cloud Transcoding and Distribution for Crowdsourced Live Game Video Streaming," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 8, pp. 1777–1789, 2016.

[11] Y.-H. Hung, C.-Y. Wang, and R.-H. Hwang, "Optimizing Social Welfare of Live Video Streaming Services in Mobile Edge Computing," *IEEE Transactions on Mobile Computing*, vol. 19, no. 4, pp. 922–934, 2019.

[12] H. Zhang, F. Guo, H. Ji, and C. Zhu, "Combinational Auction-Based Service Provider Selection in Mobile Edge Computing Networks," *IEEE Access*, vol. 5, pp. 13 455–13 464, 2017.

[13] M. B. Safianowska, R. Gdowski, and C. Huang, "Combinatorial Recurrent Multi-Unit Auctions for Fog Services," in *Proc. of ICS*, 2016.

[14] T. Roughgarden, "Algorithmic Game Theory," *Communications of the ACM*, vol. 53, no. 7, pp. 78–86, 2010.

[15] C. Wu and B. Li, "Optimal Peer Selection for Minimum-Delay Peer-to-Peer Streaming with Rateless Codes," in *Proc. of P2PMMS*, 2005.

[16] M. Wang and B. Li, "R2: Random push with random network coding in live peer-to-peer streaming," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 9, pp. 1655–1666, 2007.

[17] J. Zhao, F. Yang, Q. Zhang, Z. Zhang, and F. Zhang, "LION: Layered Overlay Multicast With Network Coding," *IEEE Transactions on Multimedia*, vol. 8, no. 5, pp. 1021–1032, 2006.

[18] R. Ahlswede, N. Cai, S.-Y. Li, and R. W. Yeung, "Network Information Flow," *IEEE Transactions on information theory*, vol. 46, no. 4, pp. 1204–1216, 2000.

[19] A. Schrijver, *Theory of Linear and Integer Programming*. John Wiley & Sons, 1998.

[20] "Technical report," [Online]. Available: https://www.jianguoyun.com/p/DVphT2YQ3pHqCxjd5PkFIAA, 2025.

[21] Z. Zhang, C. Wu, and Z. Li, "Near-Optimal Topology-Adaptive Parameter Synchronization in Distributed DNN Training," in *Proc. of 40th INFOCOM*, 2021.

[22] X. Gao, A. Majidi, Y. Gao, G. Wu, N. Jahanbakhsh, L. Kong, and G. Chen, "Nous: Drop-Freeness and Duplicate-Freeness for Consistent Updating in SDN Multicast Routing," *IEEE/ACM Transactions on Networking*, vol. 32, no. 5, pp. 3685 – 3698, 2024.