Liva: Adaptive Bitrate Ranking with Dynamic Value Perception for Video Streaming

Zhiqiang Yang[1], Hang Xiao[2,*], Chongchong Yang[1], Chaoqian Liu[1], and Wenhao Zhu[1]
[1]Shanghai University, Shanghai, China
[2]Shanghai Minhang Polytechnic, Shanghai, China
{yangzq, 0124ycc, shengfang527, whzhu}@shu.edu.cn, xiaohang@shmp.edu.cn

***Abstract*—Adaptive Bitrate (ABR) algorithms are extensively deployed in video streaming to improve Quality of Experience (QoE). Common bitrate decision approaches typically emphasize physical factors such as network bandwidth and latency. We argue that designing truly effective mechanisms necessitates a more holistic perspective: beyond adapting to time-varying network conditions, they must also perform fine-grained resource allocation that explicitly integrates video content characteristics with user viewing preferences. In particular, users' quality sensitivity and retention willingness vary across video segments. By allocating bandwidth differentially according to segment-level user attention and content reuse potential, overall QoE can be improved with limited bandwidth overhead. To this end, we propose Liva, a lifecycle value-aware ABR algorithm. Liva introduces Value-Aware QoE (VA-QoE), a dynamic QoE metric that characterizes content value in real time by fusing collective interaction behaviors with content access characteristics. Based on this metric, Liva predicts the performance outcomes of candidate actions and formulates bitrate decisions to maximize long-term rewards, thereby generating high-quality playback sequences that enhance overall QoE. Extensive experiments demonstrate that, across diverse network conditions, Liva improves VA-QoE by approximately 11.3%–26.7% over the strongest baseline and effectively enhances the viewing experience on high-value segments.**

***Index Terms*—Video Streaming, Adaptive Bitrate, Quality of Experience, Deep Learning**

I. Introduction

In recent years, Internet video streaming services have witnessed an explosive proliferation. According to Ericsson's forecast [1], video traffic is expected to account for 76% of global mobile data traffic. Constrained by the inherent volatility and uncertainty of mobile networks, ensuring Quality of Experience (QoE) for users within bandwidth-limited and dynamic environments has emerged as a paramount challenge for content providers [2]. To address this, researchers have proposed a myriad of Adaptive Bitrate (ABR) optimization strategies, spanning from early parametric approaches [3], [4] to the nascent learning-based paradigms, such as Reinforcement Learning (RL) [5], [6]. By dynamically adjusting the bitrate of each video chunk based on real-time network conditions, these algorithms achieve an effective trade-off between rebuffering events and video clarity in fluctuating network environments.

Beyond network dynamics, video consumption scenarios exhibit another equally critical dynamic characteristic: users' quality sensitivity and retention willingness are not constant. Instead, they manifest segment-level dynamic variations influenced by content features and user viewing preferences. For instance, in action movies, high-motion scenes such as "car chases and explosions" serve not only as visual focal points where viewers are highly sensitive to quality, but also as frequent cache hotspots within CDN or P2P networks. In contrast, for plots involving "static dialogue," viewers demonstrate higher tolerance—exhibiting greater leniency toward quality degradation and relatively lower sensitivity to rebuffering. Furthermore, such segments are often accompanied by skipping or variable-speed playback behaviors, rendering the marginal benefit of investing additional bandwidth to enhance visual quality relatively limited. Therefore, we argue that in addition to adapting to network conditions, bitrate decisions should simultaneously consider segment-level content value and user viewing preferences. By purposefully biasing resources toward segments commanding high user attention, it is possible to enhance overall experience quality with limited bandwidth overhead.

To achieve a more precise alignment between bandwidth allocation and content value, this paper proposes Liva, a lifecycle value-aware ABR algorithm. The core of Liva is to extend existing QoE assessment paradigms by constructing a dynamic Value-Aware QoE (VA-QoE) metric capable of characterizing content value in real-time. Specifically, this metric integrates content popularity distribution with user interaction signals (e.g., retention and engagement), mapping content value onto time-varying weighting factors. This effectively quantifies both the importance of segment-level video content and the marginal utility of bandwidth allocation. Building upon this foundation, Liva employs offline supervised learning for action-conditional prediction to estimate the impact of candidate actions on fundamental metrics (e.g., video quality, rebuffering, and bitrate switching). It then performs look-ahead optimization within a rolling decision window, incorporating dynamic value weights to generate a sequence of bitrate decisions that maximizes long-term rewards. The main contributions of this paper are summarized as follows:

- We propose Value-Aware QoE (VA-QoE), a dynamic QoE metric that fuses content features with collective user interaction signals to construct time-varying value weights, providing an optimization objective for fine-grained resource allocation.

- We propose Liva, a lifecycle value-aware ABR algorithm. Liva employs a dual-stream spatiotemporal prediction network to predict the impact of candidate actions on fundamental experience metrics. By performing look-ahead decision optimization with VA-QoE as the objective, Liva enhances overall QoE and improves the viewing experience on high-value segments.

II. Related Work

Existing ABR research primarily focuses on network-level bitrate optimization [7]. Early control-theoretic methods, such as MPC [4] and BOLA [8], as well as auto-tuning methods like Oboe [9], provide basic playback stability, but remain sensitive to prediction errors under dynamic and heterogeneous networks. To improve robustness, Pensieve [5] introduced DRL into ABR. Later works further addressed DRL limitations: Comyco [10] and SABR [11] use imitation learning and behavior cloning to improve training efficiency and decision stability; MERINA [12] adopts meta-learning to enhance cross-environment generalization; and BETA [13] uses joint spatiotemporal learning to improve generalization in heterogeneous scenarios. SODA [14] instead emphasizes playback consistency by suppressing frequent bitrate switching while maintaining basic quality and smoothness. Overall, these methods optimize QoE mainly through throughput prediction, buffer occupancy, and bitrate switching, achieving stable bitrate decisions under complex network conditions. However, they mostly target general QoE optimization and make limited use of content value differences. As a result, under bandwidth scarcity or severe fluctuations, segment-level importance may not be fully reflected, leaving room to further improve overall experience.

To incorporate content features and user perception into bitrate decision-making, prior studies have explored video semantic cues and perceptual signals. On the content-aware side, Gao et al. [15] used CNN-based deep video analysis to extract spatiotemporal features and proposed a Content of Interest (CoI) metric for salient-region-aware bandwidth allocation. CAST [16] further exploits encoder-side complexity signals to capture segment-level difficulty variations and assign differentiated optimization weights. On the user-aware side, SENSEI [17] infers user sensitivity from real-time perceptual feedback such as facial expressions and gaze behaviors, while Ruyi [18] models heterogeneous user preferences through individualized QoE profiles and personalized reward weights in ABR. Overall, these studies highlight the benefit of incorporating content and user awareness into ABR. However, most existing methods rely on relatively single-dimensional cues for local adjustment, with limited emphasis on segment-level value representations that evolve over playback and support long-horizon decision-making. In addition, maintaining such signals at scale may incur extra system overhead. Therefore, it remains worthwhile to explore a unified way of integrating segment-level, multi-dimensional value signals into look-ahead planning to better prioritize critical segments through cross-temporal resource allocation.

III. Method

Content value dynamics are important for user QoE. We propose Liva, a lifecycle value-aware ABR algorithm (Fig. 1). This section presents the Value-Aware QoE objective (VA-QoE) and Liva's action-conditional prediction and look-ahead planning.

A. Value-Aware QoE Modeling

Existing ABR algorithms often assume "content homogeneity," overlooking dynamic variations in segment-level perceived value. High-value segments usually attract stronger attention and interaction, leading to higher QoE gains and stronger hotspot effects. More precise bitrate allocation would require estimating users' retention willingness and attention for each segment, which remains difficult in large-scale real-world systems due to content diversity, emotional implicitness, and user heterogeneity.

To address these challenges, we propose an efficient multi-dimensional content value representation based on the coupled "content-behavior-feedback" relationship. Instead of explicitly modeling individual micro-level psychological states, we use platform-available historical interaction data to derive standardized segment-level value proxy signals. Specifically:

1) **To address content diversity**, we introduce the hotspot prior sequence P_t from the structural popularity layer, as shown in Fig. 2(a). P_t represents the relative intensity of collective attention across segments along the playback timeline. Aggregated from large-scale user behaviors, it provides a structural hotspot prior over time and captures the global distribution of potentially high-value segments across diverse genres without explicit semantic parsing.

2) **To address emotional implicitness**, we introduce the interaction impulse sequence D_t from the instant feedback layer, as shown in Fig. 2(b), to measure viewers' immediate engagement intensity over time. Aggregated from platform-side interaction events within each segment window, D_t reflects highlight moments that often trigger collective resonance and emotional responses.

3) **To address user heterogeneity**, we adopt the behavioral retention curve ρ_t as a collective proxy for users' willingness to continue watching, as shown in Fig. 2(c). Aggregated from large-scale playback logs, ρ_t denotes the proportion of sessions that remain active at segment t and reflects a time-varying drop-off risk profile shaped by viewing behaviors. To capture local transitions of collective interest, we consider both the absolute level of ρ_t and its local variation trend. A faster decline indicates weaker interest and higher drop-off risk, while a slower decline suggests more stable attention and stronger willingness to continue watching. Accordingly, we define the retention-trend metric as the relative first-order difference of the retention rate, where ϵ is a numerical stability term:

$$g_t \triangleq \frac{\rho_t - \rho_{t-1}}{\rho_{t-1} + \epsilon}. \tag{1}$$

Because interaction scale and hotspot magnitude vary across videos, directly using raw platform signals may let absolute

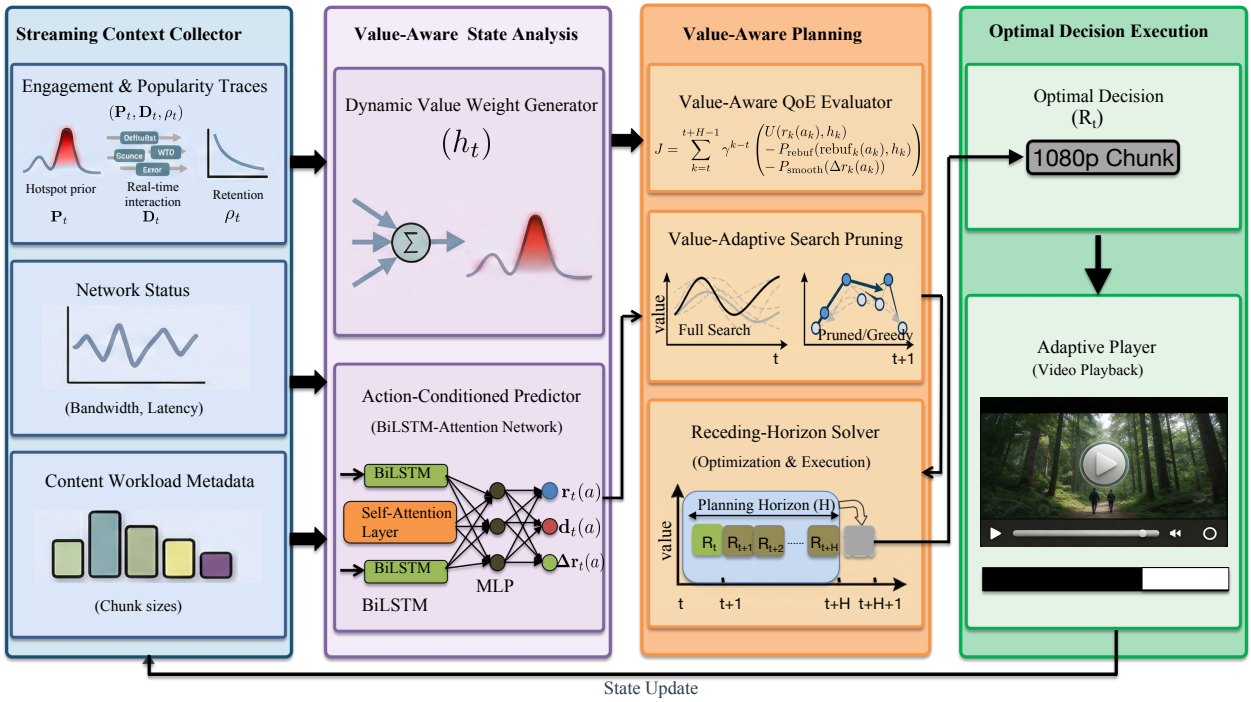

Fig. 1. Liva Architecture

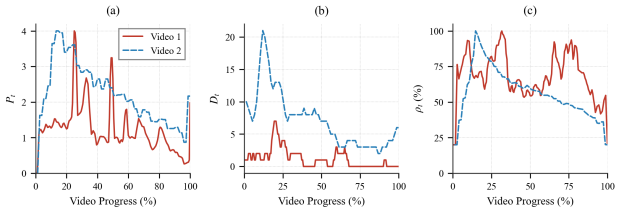

Fig. 2. Dynamics of content value features over time

traffic volume dominate value modeling and obscure within-video relative fluctuations. We therefore apply per-video adaptive normalization to map value signals onto a unified relative scale, so that the model focuses on temporal structure rather than cross-video magnitude differences. For newly published or long-tail content with sparse interactions, some value dimensions may be missing or noisy. In such cases, we impute missing dimensions with default values to avoid systematic bias, allowing the model to fall back to a baseline ABR policy that treats segment values as indistinguishable. As interaction data accumulates, these signals become more stable and more useful for optimization. Based on the normalized complementary value features, we aggregate the multi-dimensional signals into a standardized time-varying value weight, denoted as h_t:

$$h_t = (P_t)^{\delta} \cdot \left[1 + \alpha \tanh\left(\frac{g_t}{\tau}\right)\right] \left[1 + \eta \cdot \mathrm{sigmoid}(D_t)\right]. \quad (2)$$

This formulation uses the structural hotspot prior P_t as the value baseline and applies dynamic gains through a multiplicative term derived from the retention trend g_t and the instantaneous interaction impulse D_t, jointly characterizing segment value at time t. Based on this value model, we formulate ABR as a sequential decision-making problem. Under time-varying bandwidth constraints, the goal is to determine a bitrate sequence $\vec{r} = \{r_1, r_2, \ldots, r_n\}$ that maximizes the value-aware quality of experience (VA-QoE) over the entire playback. The objective is computed by accumulating value-weighted rewards and penalties for each chunk:

$$\text{VA-QoE} = \sum_{t=1}^{N} \left(U_t - P_{\text{rebuf},t} - P_{\text{smooth},t}\right). \quad (3)$$

High-value segments typically attract stronger visual attention and exhibit higher subjective sensitivity; consequently, the experiential return from an identical bitrate increase is not uniform across segments. To align bitrate allocation with this perceptual non-uniformity, we model the video quality utility U_t as a value-modulated nonlinear function:

$$U_t = \beta \left[\ln\left(1 + \frac{r_t}{r_{\min}}\right)\right]^{\kappa h_t}. \quad (4)$$

Here, the logarithmic term captures the diminishing marginal returns of bitrate, while h_t in the exponent modulates the sensitivity of quality gains for high-value segments. κ serves as the modulation intensity coefficient.

Under time-varying network conditions, persistently selecting high bitrates can drain the playback buffer and trigger rebuffering events, which severely degrade user experience. This is especially critical for high-value highlight segments, where any interruption may incur substantially larger perceptual loss than in ordinary segments. Therefore, to explicitly

discourage rebuffering risk during decision-making, we define the rebuffering penalty $P_{\text{rebuf},t}$ as:

$$P_{\text{rebuf},t} = \lambda h_t \left(e^{s\cdot \text{rebuf}_t} - 1\right). \tag{5}$$

The smoothness term maintains the standard switching penalty in the logarithmic domain:

$$P_{\text{smooth},t} = \mu \left|\ln(r_t) - \ln(r_{t-1})\right|. \tag{6}$$

B. Liva ABR Algorithm Design

Existing ABR algorithms do not explicitly model time-varying content value, which limits QoE optimization. To address this limitation, we present Liva, which explicitly decouples the objective network/playback state from the content value state in both representation and decision-making. Liva uses a prediction model to estimate the network- and buffer-level effects of candidate actions (e.g., download latency and bitrate switching magnitude), while incorporating dynamic value weights into planning to modulate rewards and enable look-ahead optimization tailored to the value distribution of each video. We model video delivery as a finite-horizon optimal control problem. The observation at time t is (s_t, h_t), where s_t denotes the objective network and playback state, including historical throughput, download latency history, current buffer occupancy b_t, and the previously selected bitrate r_{t-1}, while h_t is the content value weight defined in Eq. (2). For a candidate action a, i.e., selecting bitrate level $r_t(a)$, the prediction model estimates the state transition $s_t \to s_{t+1}$ and, through analytical mapping, derives the QoE meta-metric vector $\mathbf{m}_t(a)$:

$$\mathbf{m}_t(a) = \left[r_t(a),\, d_t(a),\, \Delta r_t(a)\right]. \tag{7}$$

Specifically, $r_t(a)$ denotes the selected bitrate (determined directly by the action), $d_t(a)$ denotes the predicted chunk download duration (output by the predictor), and the bitrate-switching magnitude is computed from the current action and the previously selected bitrate:

$$\Delta r_t(a) = \left|\ln r_t(a) - \ln r_{t-1}\right|. \tag{8}$$

Given the buffer dynamics, the rebuffering time under action a can be derived from $d_t(a)$ and the current buffer occupancy b_t as:

$$\text{rebuf}_t(a) = \max\left(0,\, d_t(a) - b_t\right). \tag{9}$$

Liva aims to plan an optimal action sequence over a look-ahead horizon H, denoted by $\mathbf{r}_{t:t+H-1}$, to maximize the cumulative VA-QoE:

$$\max_{\mathbf{r}_{t:t+H-1}} J = \sum_{k=t}^{t+H-1} \gamma^{k-t} \cdot Q_{\text{VA}}\big(\mathbf{m}_k(a_k), h_k\big). \tag{10}$$

Here, $\gamma \in [0,1]$ is the discount factor, and $Q_{\text{VA}}(\mathbf{m}_k(r_k), h_k)$ denotes the value-aware QoE at the granularity of a single chunk, as formulated in Eq. (3). It combines the chunk-level rewards and penalty terms, modulated by the value weight h_k, and is accumulated over the look-ahead window for optimization. To capture highly nonlinear network dynamics, we design a dual-stream spatiotemporal prediction network that predicts chunk download latency $d_t(a)$ conditioned on the current state s_t and candidate action a, as shown in Fig. 1. The network contains a network-temporal branch and an action-workload branch. In the network-temporal branch, BiLSTMs encode the past k-step throughput sequence $\vec{x}_t$ and download-latency history $\vec{\tau}_t$, and a self-attention module aggregates the throughput hidden states into a compact network representation z_{net}, emphasizing informative historical segments. In the action-workload branch, an MLP takes the current buffer occupancy b_t, previously selected bitrate r_{t-1}, chunk size $C(a)$, and summary statistics of throughput and buffer variations to produce the workload representation z_{act}. We then fuse z_{net} and z_{act} to predict $d_t(a)$, which is used to update the buffer state and derive $\text{rebuf}_t(a)$ during look-ahead planning. Since download latency follows a long-tailed distribution, we train the predictor end-to-end on the offline trajectory dataset $\mathcal{D}$ with a log-domain Huber loss to improve robustness to extreme latency outliers:

$$\mathcal{L}(\theta) = \frac{1}{|\mathcal{D}|} \sum_{j\in\mathcal{D}} L_\delta\Big(f_\theta(s_j, a_j) - \log(1 + d_j)\Big), \tag{11}$$

where $f_\theta(s_j, a_j)$ denotes the network output in the log domain, d_j represents the ground-truth download latency of the j-th sample, and $L_\delta(\cdot)$ is the Huber loss function.

Given the predicted state transitions and action meta-metrics $\mathbf{m}_t(a)$, the Liva planner solves the optimal bitrate sequence in Eq. (10) over a look-ahead window of $H = 3$. To reduce planning overhead, we propose a Value-Adaptive Pruning strategy that adjusts the search granularity according to the value weight h_k within the planning horizon. When h_k is high, the planner performs more exhaustive action enumeration; when h_k is low, it adopts a greedy strategy to select the "Maximum Safe Path," i.e., the highest feasible bitrate under the predicted stall-free constraint. The system further employs Receding Horizon Control (RHC), executing only the first action in each cycle and re-optimizing at the next time step using the latest state, thereby correcting prediction errors and adapting to network fluctuations.

IV. Experiments

A. Datasets and Experimental Environment

To evaluate Liva, we conduct trace-driven experiments in an ABR simulator [5] using network traces from ABRBench [11]. We use four public trace sets: FCC-16, Puffer-21, Puffer-22, and SolisWi-Fi. Videos are segmented into 4-second chunks. The bitrate ladders are $\{300, 750, 1200, 1850, 2850, 4300\}$ kbps for FCC-16 and Puffer, and $\{1000, 2500, 5000, 8000, 16000, 40000\}$ kbps for SolisWi-Fi. To construct h_t, we randomly sample 100 public Bilibili videos and use the "High-Energy Progress Bar," real-time *danmaku* timestamps, and retention statistics curves to obtain P_t, D_t, and ρ_t, respectively. All signals are aligned to 4-second chunk boundaries, aggregated at the chunk level, and normalized. All data are split into training and test sets with a ratio of 7:3.

B. Implementation Details

Liva's prediction module uses a BiLSTM layer combined with a Self-Attention mechanism. The BiLSTM hidden dimension is set to 64 (32 per direction), and the number of attention heads is set to 4. The historical input window lengths are configured as 8 steps for the throughput sequence and 5 steps for the download duration sequence. Regarding the value model parameters, we set the popularity power exponent to $\delta = 1.05$, the retention adjustment weight to $\alpha = 0.20$, the smoothing factor to $\tau = 5.0$, and the danmaku weight to $\eta = 0.10$. For the planner, the window length is set to $H = 3$, and the discount factor is $\gamma = 0.9$. The rebuffering penalty weight λ and the smoothness penalty weight μ are set to 4.3 and 1.0, respectively. Unless otherwise specified, default values are adopted for all other settings.

C. Comparison with Baseline Algorithms

We compare Liva with six representative baselines: **BB**, a heuristic buffer-based method; **BOLA** [8], a Lyapunov-based method; **RobustMPC** [4], an MPC-based QoE optimization method; **QUETRA** [19], a queueing-theoretic method; **Pensieve** [5], a DRL-based method; and **Comyco** [10], an imitation learning-based method.

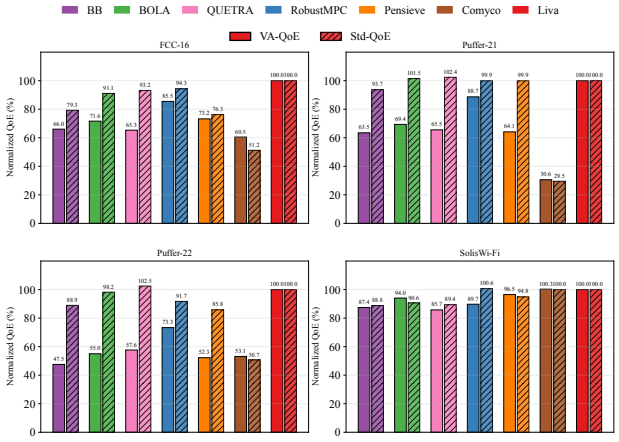

Fig. 3. Average QoE Performance

To verify whether Liva can consistently enhance VA-QoE under varying network dynamics while maintaining the fundamental viewing experience (Std-QoE), we evaluate these methods on four network trace datasets: FCC-16, Puffer-21, Puffer-22, and SolisWi-Fi. For fair value assessment, we tailored Pensieve, Comyco, and RobustMPC by replacing Pensieve's training reward with VA-QoE, adjusting Comyco's expert policy to maximize cumulative VA-QoE, and using VA-QoE as the utility function in RobustMPC's planning objective. Fig. 3 shows the average performance of all methods (solid bars: VA-QoE; hatched bars: Std-QoE), normalized with respect to Liva, and Fig. 4 shows the CDF of VA-QoE. Liva achieves the highest VA-QoE on all four datasets, outperforming the second-best method by 14.5%, 11.3%, and 26.7% on FCC-16, Puffer-21, and Puffer-22, respectively, while remaining comparable

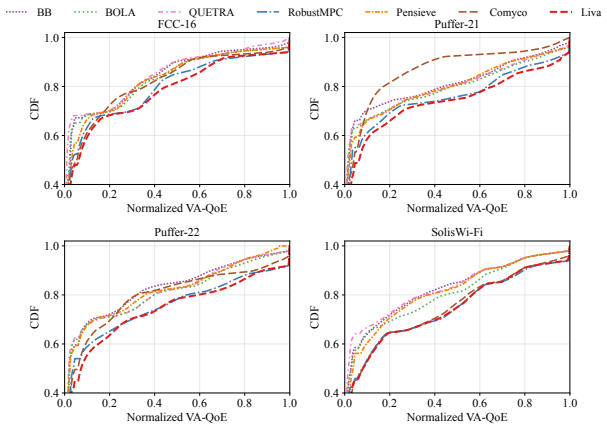

Fig. 4. CDF of normalized VA-QoE

to the best-performing baseline on SolisWi-Fi, where VA-QoE is close to saturation. Meanwhile, Fig. 3 shows that Liva improves VA-QoE without compromising Std-QoE: its Std-QoE is close to the strongest baseline on FCC-16, Puffer-22, and SolisWi-Fi, while on Puffer-21 QUETRA achieves slightly higher Std-QoE but substantially lower VA-QoE. Fig. 4 further shows that Liva shifts the VA-QoE distribution to the right across all datasets, indicating fewer low-VA-QoE cases and a better overall experience distribution.

D. Evaluation of Value-Aware Resource Allocation

To evaluate the ability of Liva to bias resource allocation toward high-value content, we define the Hotspot Preference Ratio (E) as the ratio of the average bitrate in hotspot regions to that in non-hotspot regions:

$$E = \frac{\bar{R}_{\text{hot}}}{\bar{R}_{\text{normal}}}. \tag{12}$$

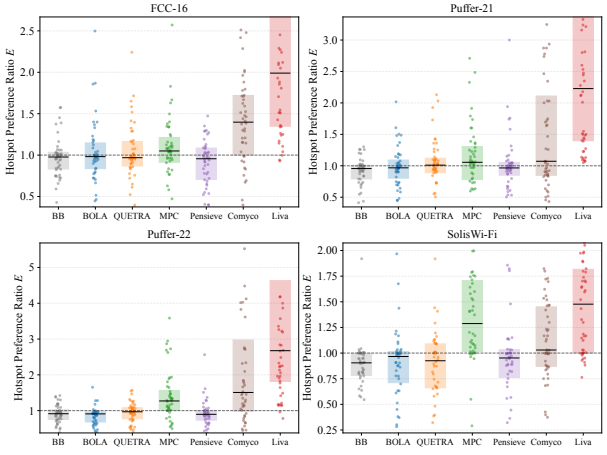

Fig. 5. Hotspot preference ratio (E) under different ABR algorithms

Fig. 5 illustrates the distribution of normalized hotspot preference ratios (E) across the four datasets, with the gray

dashed line ($y = 1.0$) denoting the average baseline level. Each scatter point represents a single test session, while colored boxplots summarize the overall distribution; the thick black line indicates the median, and the box spans the interquartile range (IQR). Traditional heuristic methods (BB, BOLA, and QUETRA) cluster around $y = 1.0$, indicating limited sensitivity to content value and no clear preference in resource allocation. After adapting Comyco and RobustMPC with the VA-QoE objective, their distributions improve but remain below that of Liva. In contrast, Liva exhibits consistently higher medians and overall distribution levels across all datasets.

E. Ablation Study on Value Components

To verify the necessity of value awareness and the deployability of Liva under missing value signals, we conduct component ablation and partial observability experiments on the value weight h_t using the FCC-16 and Puffer-22 datasets, as shown in Fig. 6. Assuming equal value for all segments ("w/o value", where $h_t \equiv 1$) leads to a significant drop in VA-QoE, indicating that optimizing only traditional QoE is insufficient under the true value objective. Under partial observability, h_t is generated during planning using only the available signals, while VA-QoE is evaluated using the complete h_t and normalized against the full Liva model. The results show that Liva retains about 81.5%–93.5% of its performance with partial signals, whereas removing value information reduces performance to about 74%.

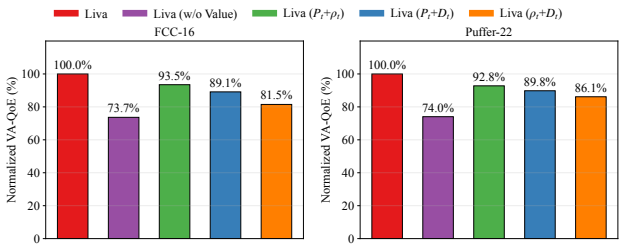

Fig. 6. Value Component Ablation and Partial Observability Results

V. Conclusion

Liva is a lifecycle value-aware ABR algorithm that aligns bandwidth allocation with the dynamic perceptual value of video content. Unlike methods that assume content homogeneity, Liva builds a real-time, value-weighted QoE metric from collective interaction behaviors and content access characteristics. With a deep learning predictor and global resource planning, Liva couples limited bandwidth with fluctuating content value. Experiments show that Liva outperforms existing ABR algorithms in Value-Aware QoE (VA-QoE) while improving high-value hotspots without degrading the viewing experience in non-hotspot regions.

Acknowledgment

This work is supported by National Natural Science Foundation of China (No. 12471500), and Shanghai Technical Service Computing Center of Science and Engineering, Shanghai University.

References

[1] Ericsson, "Mobile network traffic q3 2025: Ericsson mobility report (mobile traffic update)," 2025, accessed: Jan. 30, 2026. [Online]. Available: https://www.ericsson.com/en/reports-and-papers/mobility-report/dataforecasts/mobile-traffic-update

[2] M. Seufert, S. Egger, M. Slanina, T. Zinner, T. Hoßfeld, and P. Tran-Gia, "A survey on quality of experience of http adaptive streaming," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 1, pp. 469–492, 2014.

[3] T.-Y. Huang, R. Johari, N. McKeown, M. Trunnell, and M. Watson, "A buffer-based approach to rate adaptation: Evidence from a large video streaming service," in *Proceedings of the 2014 ACM conference on SIGCOMM*, 2014, pp. 187–198.

[4] X. Yin, A. Jindal, V. Sekar, and B. Sinopoli, "A control-theoretic approach for dynamic adaptive video streaming over http," in *Proceedings of the 2015 ACM conference on special interest group on data communication*, 2015, pp. 325–338.

[5] H. Mao, R. Netravali, and M. Alizadeh, "Neural adaptive video streaming with pensieve," in *Proceedings of the conference of the ACM special interest group on data communication*, 2017, pp. 197–210.

[6] A. Bentaleb, M. Lim, M. N. Akcay, A. C. Begen, and R. Zimmermann, "Bitrate adaptation and guidance with meta reinforcement learning," *IEEE Transactions on Mobile Computing*, vol. 23, no. 11, pp. 10 378–10 392, 2024.

[7] A.-H. Lee, H. Bae, and C.-K. Kim, "Cocktail: Video streaming qoe optimization with chunk replacement and guided learning," *Computer Communications*, vol. 219, pp. 204–215, 2024.

[8] K. Spiteri, R. Urgaonkar, and R. K. Sitaraman, "Bola: Near-optimal bitrate adaptation for online videos," *IEEE/ACM transactions on networking*, vol. 28, no. 4, pp. 1698–1711, 2020.

[9] Z. Akhtar, Y. S. Nam, R. Govindan, S. Rao, J. Chen, E. Katz-Bassett, B. Ribeiro, J. Zhan, and H. Zhang, "Oboe: Auto-tuning video abr algorithms to network conditions," in *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*, 2018, pp. 44–58.

[10] T. Huang, C. Zhou, R.-X. Zhang, C. Wu, X. Yao, and L. Sun, "Comyco: Quality-aware adaptive video streaming via imitation learning," in *Proceedings of the 27th ACM international conference on multimedia*, 2019, pp. 429–437.

[11] P. Luo, Y. Zhao, B. Zhang, G. Yang, B.-H. Soong, and C. Yuen, "Sabr: A stable adaptive bitrate framework using behavior cloning pretraining and reinforcement learning fine-tuning," *arXiv preprint arXiv:2509.10486*, 2025.

[12] N. Kan, Y. Jiang, C. Li, W. Dai, J. Zou, and H. Xiong, "Improving generalization for neural adaptive video streaming via meta reinforcement learning," in *Proceedings of the 30th ACM international conference on multimedia*, 2022, pp. 3006–3016.

[13] G. Zhang, Z. Wang, H. Wei, M. Xiao, H. Yuan, D. Yu, and X. Cheng, "A novel spatial-temporal learning method for enhancing generalization in adaptive video streaming," *IEEE Transactions on Mobile Computing*, 2025.

[14] T. Chen, Y. Lin, N. Christianson, Z. Akhtar, S. Dharmaji, M. Hajiesmaili, A. Wierman, and R. K. Sitaraman, "Soda: An adaptive bitrate controller for consistent high-quality video streaming," in *Proceedings of the ACM SIGCOMM 2024 Conference*, 2024, pp. 613–644.

[15] G. Gao, L. Dong, H. Zhang, Y. Wen, and W. Zeng, "Content-aware personalised rate adaptation for adaptive streaming via deep video analysis," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE, 2019, pp. 1–8.

[16] W. Li, J. Huang, Y. Liang, J. Liu, W. Zhang, W. Lyu, and J. Wang, "Cast: An intricate-scene aware adaptive bitrate approach for video streaming via parallel training," in *International Conference on Algorithms and Architectures for Parallel Processing*. Springer, 2023, pp. 131–147.

[17] X. Zhang, Y. Ou, S. Sen, and J. Jiang, "{SENSEI}: Aligning video streaming quality with dynamic user sensitivity," in *18th USENIX Symposium on Networked Systems Design and Implementation (NSDI 21)*, 2021, pp. 303–320.

[18] X. Zuo, J. Yang, M. Wang, and Y. Cui, "Adaptive bitrate with user-level qoe preference for video streaming," in *IEEE INFOCOM 2022-IEEE Conference on Computer Communications*. IEEE, 2022, pp. 1279–1288.

[19] P. K. Yadav, A. Shafiei, and W. T. Ooi, "Quetra: A queuing theory approach to dash rate adaptation," in *Proceedings of the 25th ACM international conference on Multimedia*, 2017, pp. 1130–1138.