Adaptive Rate Control Algorithm for SHVC: Application to HD/UHD

T.Biatek[1,2], W. Hamidouche[3], J.-F. Travers[2], O. Deforges[3]
[1]b<>com, [2]TDF, [3]IETR / INSA de Rennes

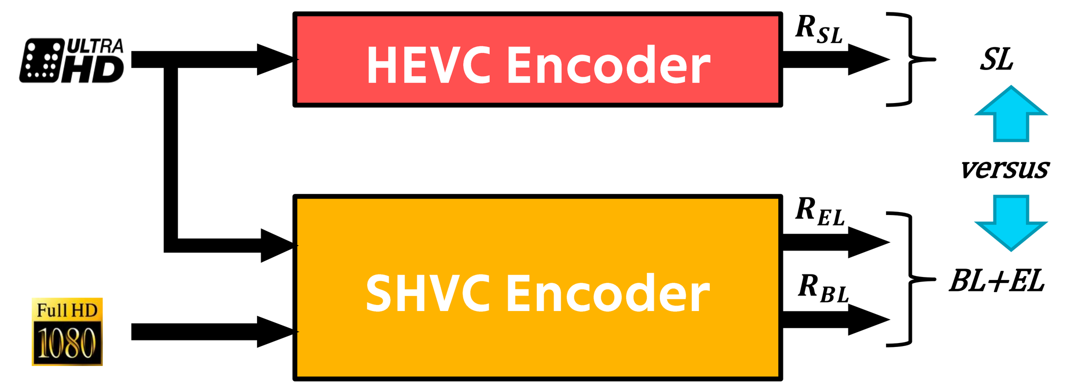
1. Context

- **SHVC: a promising technology**
  - Scalable video coding
  - Substantial compression gains
  - Backward compatibility
- **Adaptive Rate Control (ARC)**
  - Adjusts encoding parameters to reach a targeted bitrate
  - Used to deploys services on networks
- **New services introduction**
  - Provide UHD through EL
  - While keeping HD service in BL

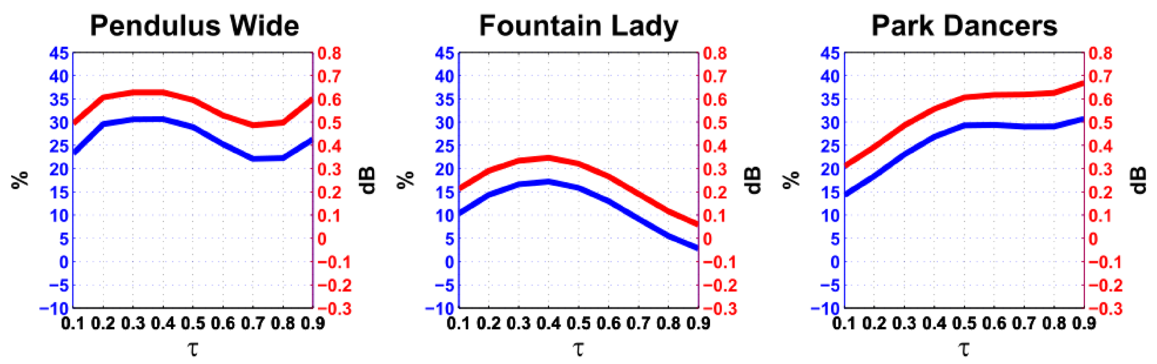
Need → ARC in SHVC to enable deployment of backward compatible UHD services!

2. Related work and motivations

- **Impact of the bitrate ratio on performance ?**
  - Defined as $\tau = R_{BL}/(R_{BL}+R_{EL})$
  - Explored in our previous work
  - **→ τ has a strong impact !**



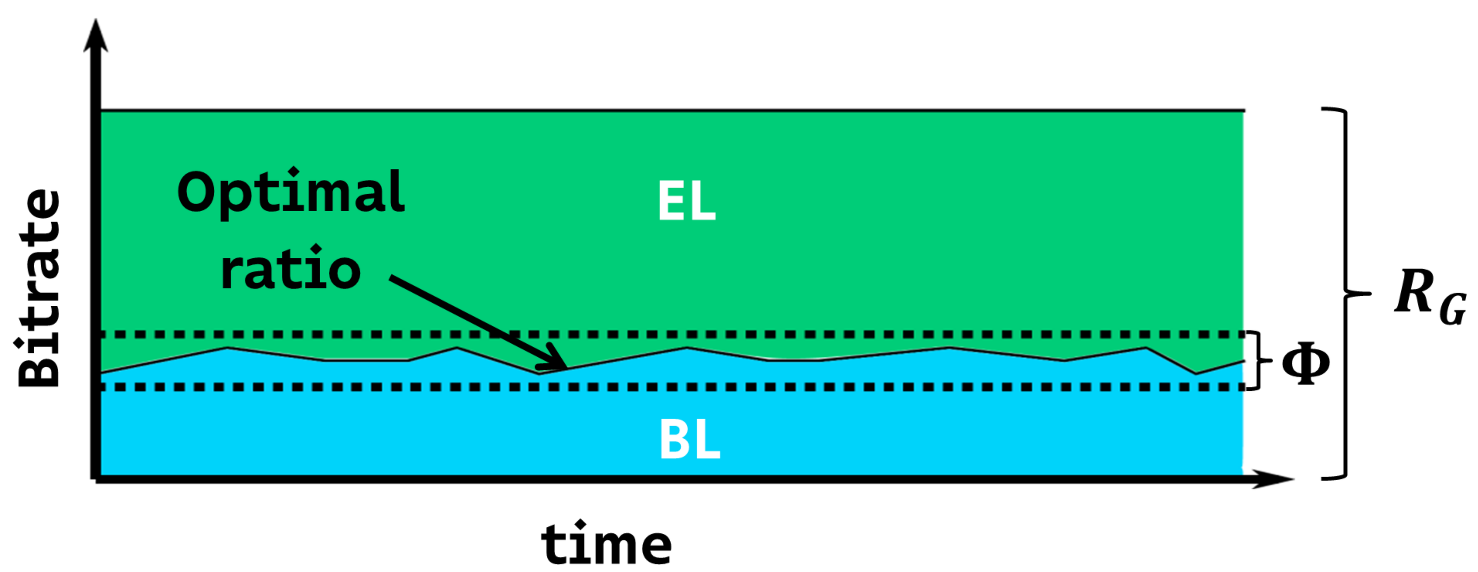
- **Existing ARC approaches**
  - Fine Granular Scalability in SVC
  - Enable fast transcoding
  - Separated bitrate per layer
  - **→ Do not exploit τ !**



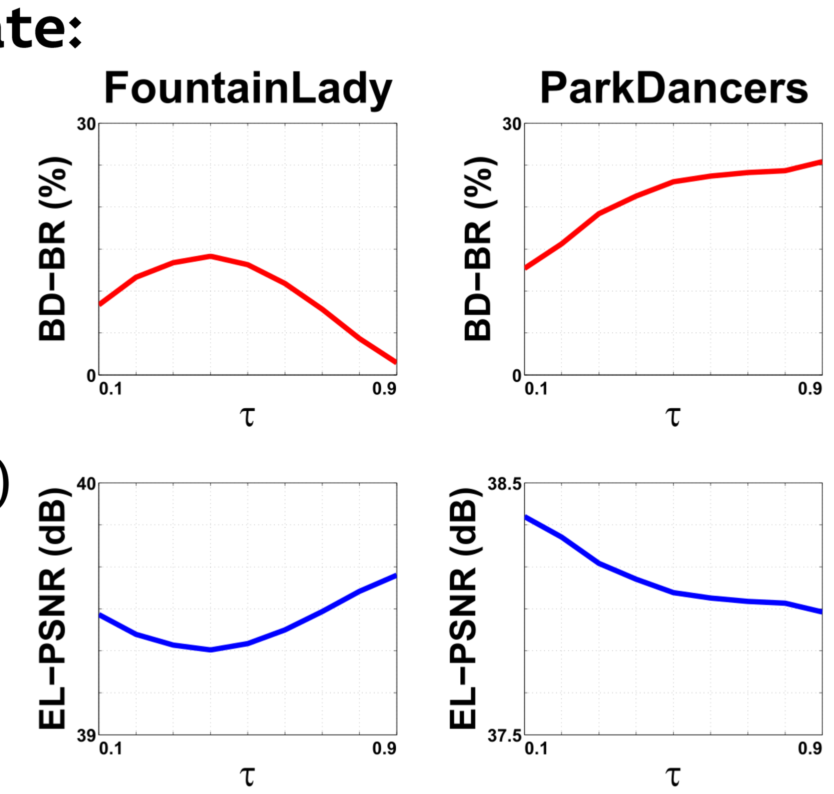
Objective → An ARC scheme based on variable ratio adjustment under global bitrate constraint (BL+EL) could strongly improve performance !

3. Proposed method

- **Encoding parameters**
  - Global bitrate: $R_G = R_{BL} + R_{EL}$
  - Authorized ratio interval: $\Phi = [\tau_{min}, \tau_{max}]$
- **Goal**
  - Adjust the bitrate ratio τ in Φ
  - To optimize the objective coding performance
- **Illustration:**



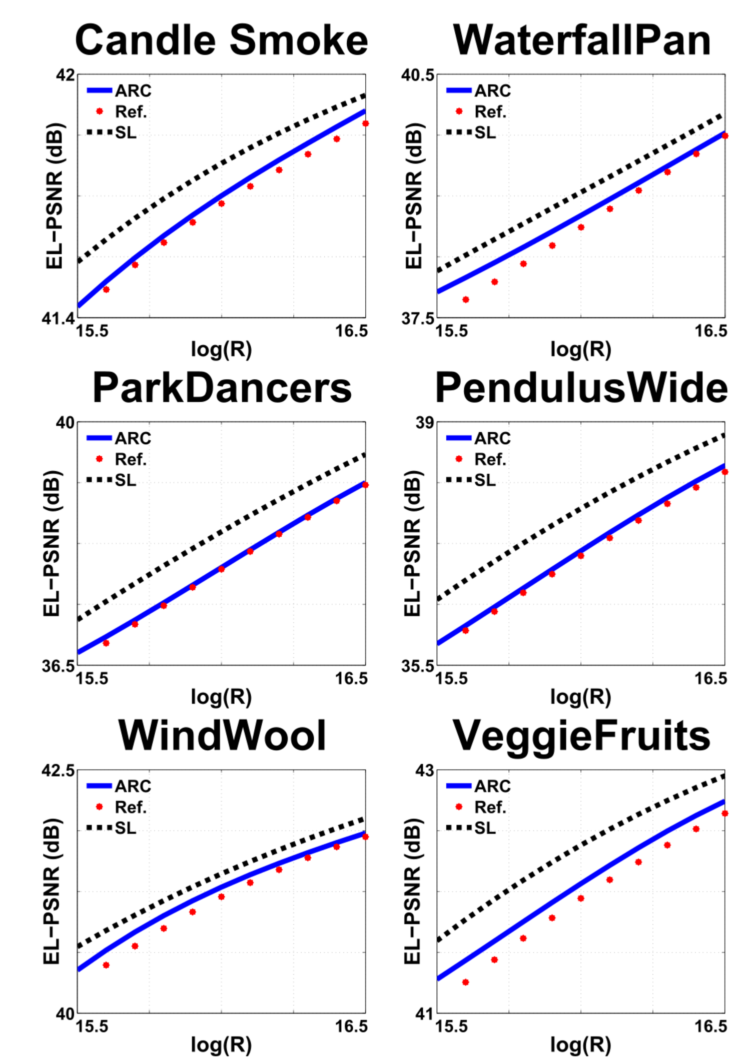
- **Step 1: GOP-Level global targeted bitrate**
  - $T_{AvgPic} = \frac{R_{PicAvg} \times (N_{Coded}+SW) - (R_{BL}+R_{EL})}{SW}$ with $R_{PicAvg} = \frac{R_G}{f}$
  - $T_{GOP} = T_{AvgPic} \times N_{GOP}$
- **Step 2: Optimization problem**
  - $\tau_{opt} = \max_{\tau \in \Phi} G(\tau)$
  - with $\max(\tau_{last} \times 0.8, \tau_{min}) \leq \tau_{opt} \leq \min(\tau_{last} \times 1.2, \tau_{min})$
- **Step 3: Layer-Level targeted bitrate:**
  - $T_{BL} = \tau_{opt} \times T_{GOP}$
  - $T_{EL} = (1-\tau_{opt}) \times T_{GOP}$
- **On-the-fly $G(\tau)$ estimation:**
  - $G(\tau) = Q_{EL}(\tau) \triangleq \alpha \times \tau + \beta$
  - Update buffer of N pairs (τ_i, q_i)
  - $\begin{pmatrix} \sum\tau_i^2 & \sum\tau_i \\ \sum\tau_i & N \end{pmatrix} \begin{pmatrix} \hat{\alpha} \\ \hat{\beta} \end{pmatrix} = \begin{pmatrix} \sum\tau_i q_i \\ \sum q_i \end{pmatrix}$



4. Experiments and analysis

- **Data set**
  - EBU UHD-1 dataset
  - Ten 3840x2160p40 8-bits 10-sec sequences
  - HD versions built with SHM-9.0 down-sampler
- **Encoding parameters**
  - $\Phi = [\tau_0 - 25\%, \tau_0 + 25\%]$ for ratio interval
  - $\tau_0 = \frac{1}{2\times\sqrt{2}}$ which is the ratio achieved by using the CTC
  - $R_G \in \{5, 10, 15, 20\}$ Mbps
- **Two approaches are compared to the single-layer (BD-BR)**
  - Our method integrated in the SHM-9.0 → G_{ARC}
  - Native SHM-9.0 working at fixed ratio τ_0 → G_{Ref}
  - Comparison between fixed and ARC → G_X
- **Observations:**
  - Bitrate overhead reduced from 20% to 16%.
  - With a crossed BD-BR improvement of 4.25%
  - Best method for 9 in 10 sequence.

Sequence	G_{Ref}	G_{ARC}	G_X
Candle Smoke	18.97%	12.89%	6.08%
Fountain Lady	15.68%	16.16%	+0.48%
Lupo Boa	11.31%	10.46%	0.85%
Lupo Confetti	13.35%	11.60%	1.75%
Park Dancers	23.80%	22.22%	1.58%
Pendulus Wide	30.56%	26.49%	4.07%
Studio Dancer	18.92%	14.57%	4.35%
Waterfall Pan	20.72%	14.61%	6.11%
Wind Wool	19.64%	9.83%	9.81%
Veggie Fruits	31.33%	22.91%	8.42%
Average	**20.43%**	**16.17%**	4.25%



Next Step → Quality and bitrate requirements per layer