Motion-Prediction-based Wireless Scheduling for Multi-User Panoramic Video Streaming

Jiangong Chen[*], Xudong Qin[*], Guangyu Zhu[†], Bo Ji[‡], Bin Li[*]
[*]Department of ECBE, University of Rhode Island, Kingston, Rhode Island, USA
[†]Department of CSS, University of Rhode Island, Kingston, Rhode Island, USA
[‡]Department of CS, Virginia Tech, Blacksburg, VA, USA

Motivation

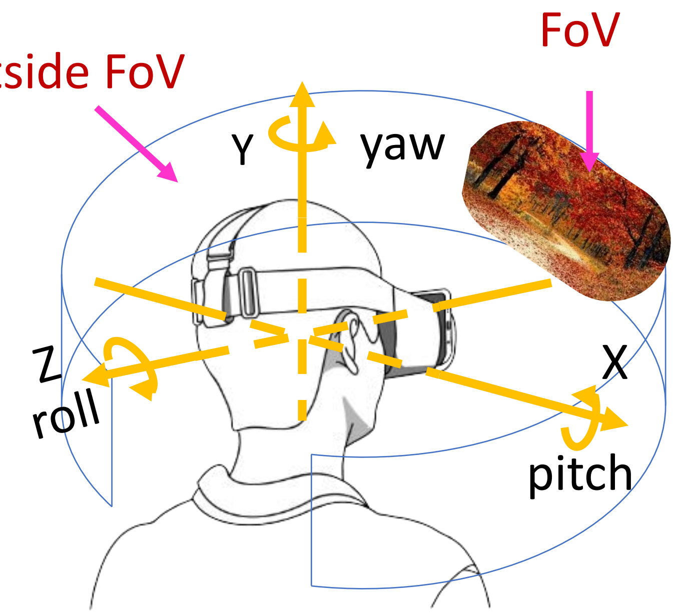
- Emerging Commercial Head-mounted Displays (HMDs)
- Panoramic video streaming provides an immersive experience for users as if they are in a virtual 3D world


Virtual Reality Classroom

- Main challenges:
  - Large network bandwidth requirement: 4~6x bandwidth consumption of a regular video with the same resolution
  - Seamless user experience: users would compete for limited bandwidth

Opportunity

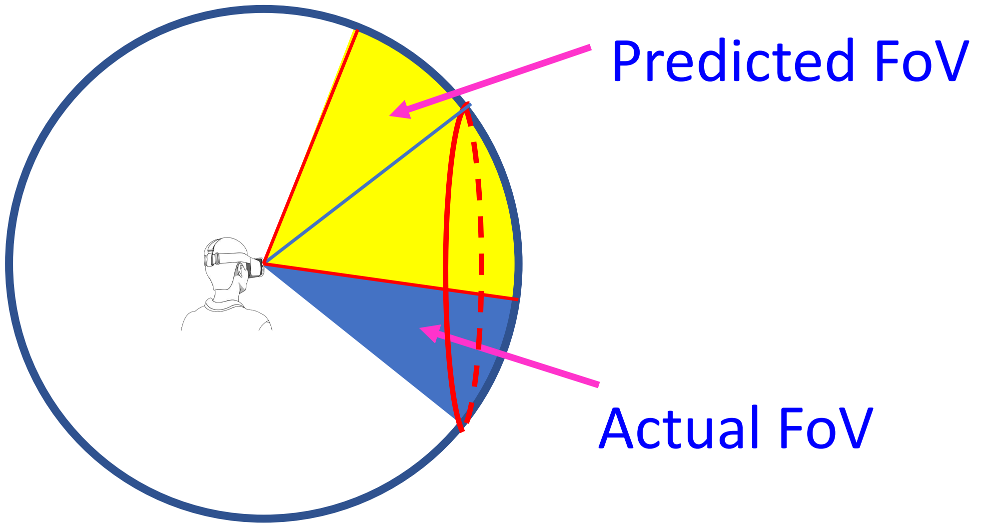
- A user may only see as little as 20% of 360° scenes, known as Field of View (FoV). It is sufficient to deliver 20% of 360° video scenes under perfect motion prediction.



Rotation coordinates and FoV

Practical Challenges

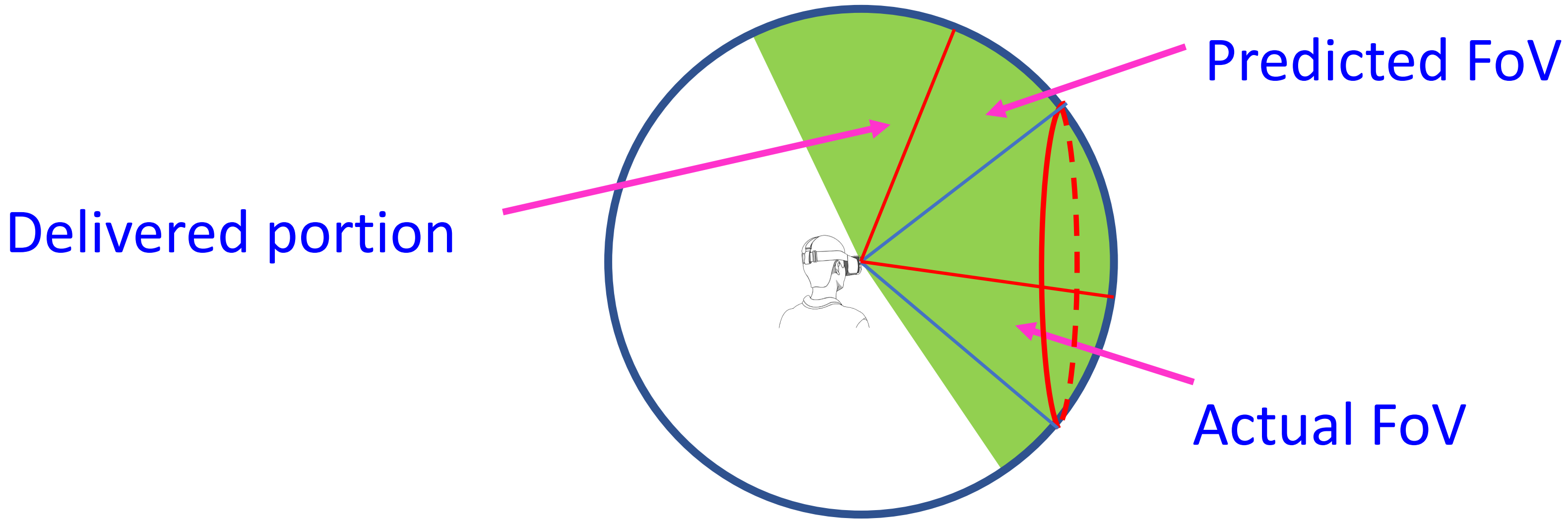
- Imperfect prediction: should deliver a portion larger than the FoV



With imperfect prediction, some of the actual FoV will be missed if we only send the predicted FoV

Practical Challenges

- Imperfect prediction: should deliver a portion larger than the FoV



The actual FoV can be completely covered if we send a larger portion based on the predicted FoV

Successful Viewing Probability

- Prediction errors of both pitch and yaw angles of user n follow the normal distribution with standard deviation σ_n^X and σ_n^Y, respectively
- We characterize the successful viewing probability $\delta_n(S_n[t])$ as a function of the delivered portion ratio $S_n[t]$ (normalized allocated rate)

$$\delta_n(S_n[t]) = \mathrm{erf}^2\left(\frac{\gamma_n(S_n[t])}{\sqrt{2}}\right)$$

where $\mathrm{erf}(x) \triangleq \frac{2}{\pi}\int_0^x e^{-y^2}dy$, $\gamma_n(S_n[t])$ is the number of standard deviations of the prediction error,

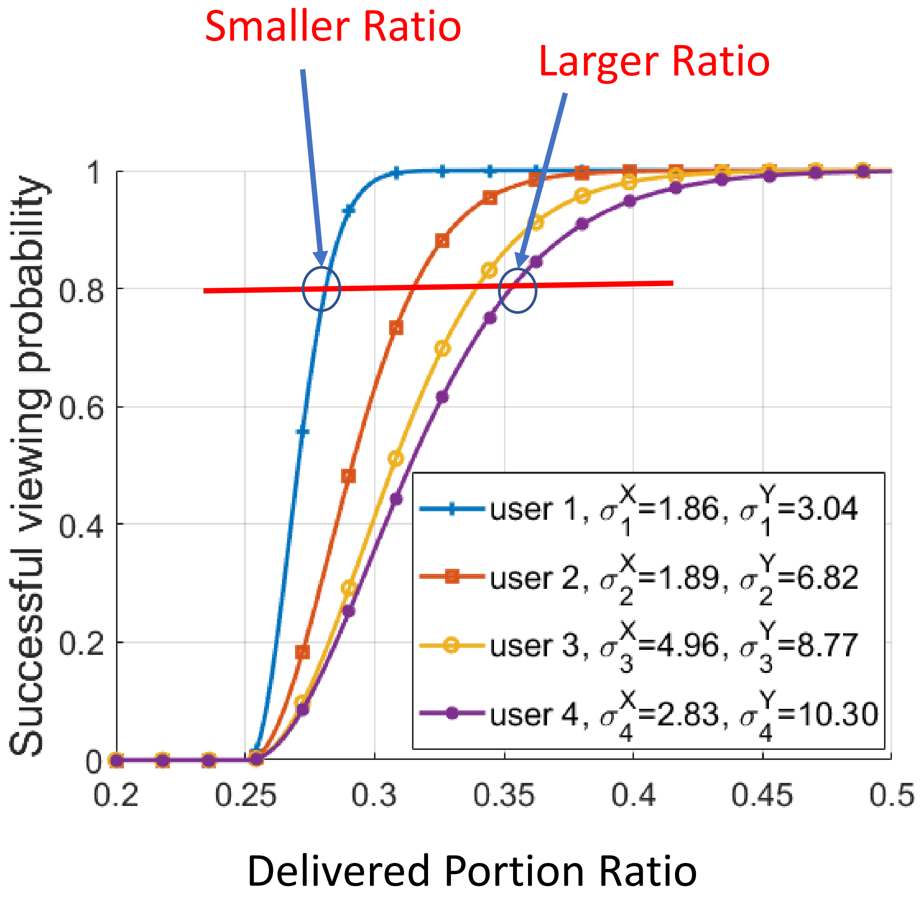
i.e.,

$$\hat{X}_n[t] - \gamma_n(S_n[t])\sigma_n^X < X_n[t] < \hat{X}_n[t] + \gamma_n(S_n[t])\sigma_n^X$$

$$\hat{Y}_n[t] - \gamma_n(S_n[t])\sigma_n^Y < Y_n[t] < \hat{Y}_n[t] + \gamma_n(S_n[t])\sigma_n^Y$$

Predicted angles

Real angles



First Goal: Maximizing Throughput

- Maximize the application-level throughput (defined as the weighted sum of the expected successful viewing probability)

$$\max_{(S_n[t])_{n=1}^{N}} \lim_{L\to\infty} \frac{1}{L} \sum_{t=0}^{L-1} \sum_{n=1}^{N} w_n E[\delta_n(S_n[t])]$$

- Constraints:
  - Wireless interference constraints

$$\text{s.t. } (S_n[t])_{n=1}^{N} \in \mathbf{S}^{(C[t])}, \forall t \geq 0$$

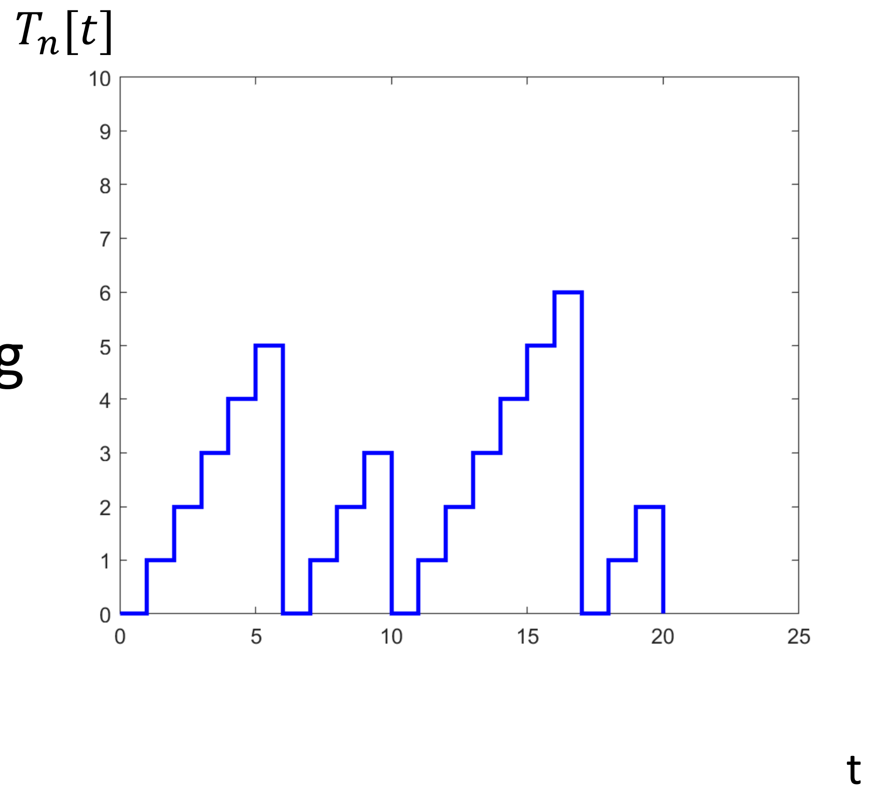
  - The average allocated transmission rate should not be less than some minimum rate

$$\lim_{L\to\infty} \frac{1}{L} \sum_{t=0}^{L-1} E\big[S_n[t]\big] \geq r_n, \forall n$$

Second Goal: Providing Seamless Experience

- Seamless user experience, keep service regularity (defined as the variance of the time between two consecutive successful views for each user)
- Time-Since-Last-Service (TSLS) counter:

$$T_n[t+1] \triangleq \begin{cases} 0, & \text{if } I_n(S_n[t]) = 1; \\ T_n[t]+1, & \text{otherwise.} \end{cases}$$



- [Li, Li, Eryilmaz 2014] showed that minimizing the expected TSLS counter is equivalent to minimizing the normalized variance of the time duration between successful services
- Our second goal is equivalent to minimizing

$$\lim_{L\to\infty} \frac{1}{L} \sum_{t=0}^{L-1} E[T_n[t]]$$

Motivating Example

time / rate / user	0		1		2		3		…
User 1	1,		0,		0,		0,		…
User 2	0,		1,		0,		0,		…
User 3	0,		0,		1,		0,		…
User 4	0,		0,		0,		1,		…

(a) Service rate of each user in each time slot.

time / result / user	0		1		2		3		…
User 1	1,		0,		0,		0,		…
User 2	0,		1,		0,		0,		…
User 3	0,		0,		1,		0,		…
User 4	0,		0,		0,		1,		…

(b) Successful content delivery rate of each user in each time slot.

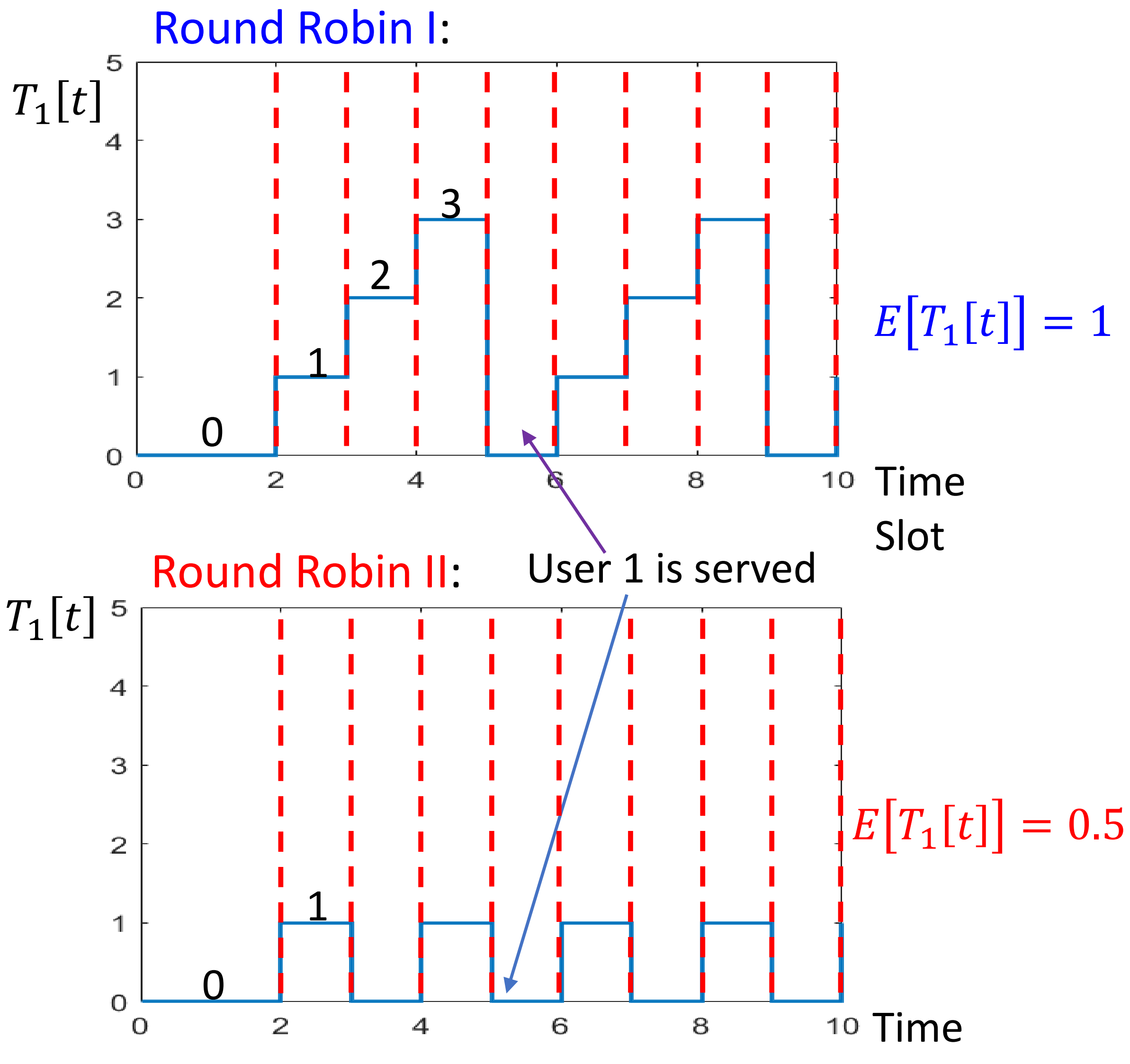
Motivating Example

time / rate / user	0		1		2		3		…
User 1	1,	0.5	0,	0	0,	0.5	0,	0	…
User 2	0,	0.5	1,	0	0,	0.5	0,	0	…
User 3	0,	0	0,	0.5	1,	0	0,	0.5	…
User 4	0,	0	0,	0.5	0,	0	1,	0.5	…

(a) Service rate of each user in each time slot.

time / result / user	0		1		2		3		…
User 1	1,	1	0,	0	0,	1	0,	0	…
User 2	0,	1	1,	0	0,	1	0,	0	…
User 3	0,	0	0,	1	1,	0	0,	1	…
User 4	0,	0	0,	1	0,	0	1,	1	…

(b) Successful content delivery rate of each user in each time slot.



Example of TSLS dynamics of user 1

Motion Prediction

- Perform independently for each user in each axis since the correlation between $X_n[t]$ and $Y_n[t]$ is much smaller than their autocorrelations

- Autoregressive Model:

$$\hat{X}_n[t] = -\sum_{k=1}^{W} a_n[k] X_n[t-k] \quad \text{and} \quad \hat{Y}_n[t] = -\sum_{k=1}^{W} b_n[k] Y_n[t-k]$$

- [Fuller 2009] showed that the prediction error converges to the Gaussian distribution as the number of data samples goes to infinity

Scheduling Algorithm Design

- A virtual queue for each user that measures the degree of violation of the average service rate constraint

$$Q_n[t+1] \triangleq (Q_n[t] + r_n - S_n[t])^+, \forall n, \forall t$$

- Non-standard Lyapunov function that combines the virtual queue and TSLS counter

$$V[t] = \frac{1}{2}\sum_{n=1}^{N} Q_n^2[t] + \eta \sum_{n=1}^{N} T_n[t]$$

- Wireless scheduling:
  - Select the schedule $\boldsymbol{S}^*[t]$ following:

Virtual Queue — TSLS counter — Successful Viewing Prob.

$$S^*[t] \in \underset{\boldsymbol{S} \in \mathbf{S}^{(c)}}{\operatorname{argmax}} \sum_{n=1}^{N} (S_n[t]Q_n[t] + (\eta T_n[t] + K w_n)\delta_n(S_n[t]))$$

where η and K are tunable parameters — Weight of User n

- Computation complexity: like the Max-Weight algorithm

Operation Example

$$S^*[t] \in \underset{\boldsymbol{S}\in\mathbf{S}^{(c)}}{\operatorname{argmax}} \sum_{n=1}^{N} \overbrace{\left(S_n[t]Q_n[t] + (\eta T_n[t] + K w_n)\delta_n(S_n[t])\right)}^{W_n[t]}$$

Tunable parameters: $\eta = 0.2, K = 1$
Schedule one user at each time slot
Available rate: $S_n[t] = 0.5$
Successful viewing prob.: $\delta_n(S_n[t]) = 1$

$Q_1[1] = 0.5, T_1[1] = 3, w_1 = 0.3$

User 1

$W_1[1] = 0.5 * 0.5 + (0.2 * 3 + 1 * 0.3) * 1 = 1.15$

$Q_2[1] = 0.5, T_2[1] = 2, w_2 = 0.6$

User 2

$W_2[1] = 0.5 * 0.5 + (0.2 * 2 + 1 * 0.6) * 1 = 1.25$

$Q_2[1] = 0.5, T_2[1] = 1, w_2 = 1$

User 3

$W_3[1] = 0.5 * 0.5 + (0.2 * 1 + 1 * 1) * 1 = 1.45$

Scheduled!

Theoretical Bounds

Algo: $S^*[t] \in \underset{\boldsymbol{S}\in\mathbf{S}^{(c)}}{\operatorname{argmax}} \sum_{n=1}^{N}(S_n[t]Q_n[t] + (\eta T_n[t] + Kw_n)\delta_n(S_n[t]))$

- Our proposed algorithm asymptotically optimizes the application-level throughput and provides seamless user experience guarantees while meeting the minimum service rate requirement, i.e.,

$$\lim_{L\to\infty}\frac{1}{L}\sum_{t=0}^{L-1}\sum_{n=1}^{N}E[w_n\delta_n(S_n[t])] \geq U^* - \frac{B(\eta)}{K} \quad ; \quad \lim_{L\to\infty}\frac{1}{L}\sum_{t=0}^{L-1}\sum_{n=1}^{N}U_n^*E[T_n[t]] \leq \frac{B(\eta) + KNw_{\max}}{\eta}$$

where $B(\eta) \triangleq \sum_{n=1}^{N}\frac{r_n^2+R_M^2}{2} + \eta N$, U^* is the optimal value of the optimization problem

- $K \nearrow$, application-level throughput $\nearrow$, mean TSLS $\nearrow$ (seamless user experience $\searrow$)
- $\eta \nearrow$, application-level throughput $\searrow$, mean TSLS $\searrow$ (seamless user experience $\nearrow$)

Simulation

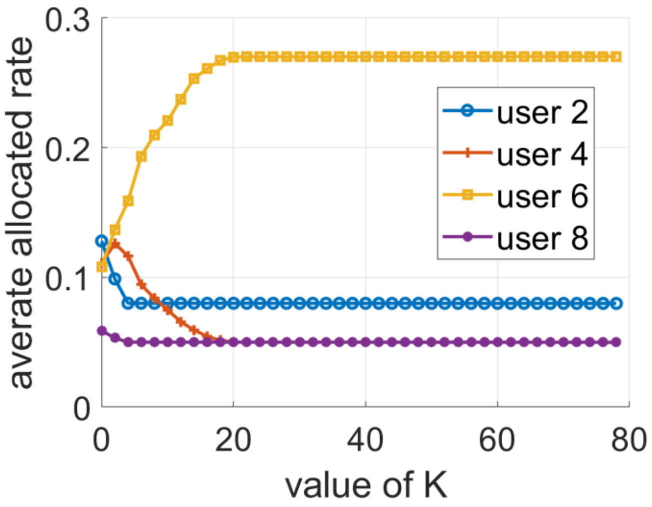
- 8 users
- Synthetic head motion data generated from the dataset [Bao, Wu, Zhang, Ramli, Liu, 2016]
- ON-OFF channel fading
- At most two users can be scheduled
- Total available rate: 1
- Rate set: $\{0,0.3,0.4,0.5,0.7,1\}$

	User 1	User 2	User 3	User 4
Required rate r_n	0.1	0.08	0.11	0.05
Weight w_n	0.2	0.1	1.0	0.8
Fading prob. p_n	0.8	0.9	0.7	0.9
	User 5	**User 6**	**User 7**	**User 8**
Required rate r_n	0.18	0.06	0.16	0.05
Weight w_n	0.9	1.2	0.3	0.2
Fading prob. p_n	0.8	0.9	0.7	0.8

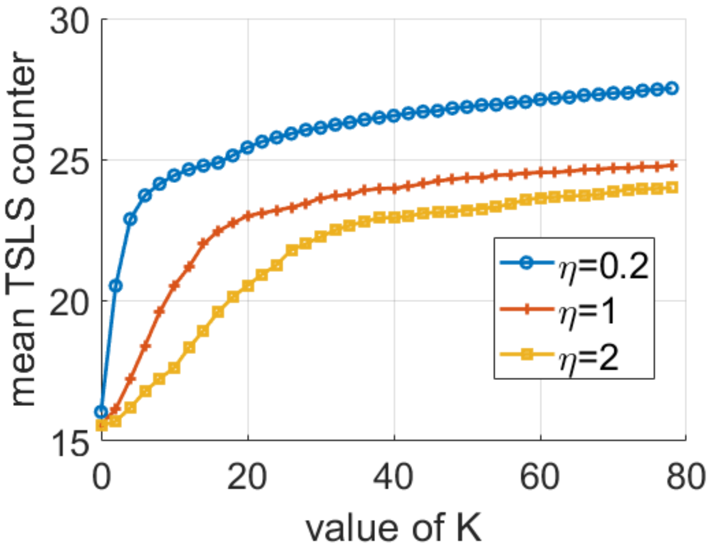
Simulation (Cont’)

Required rate:

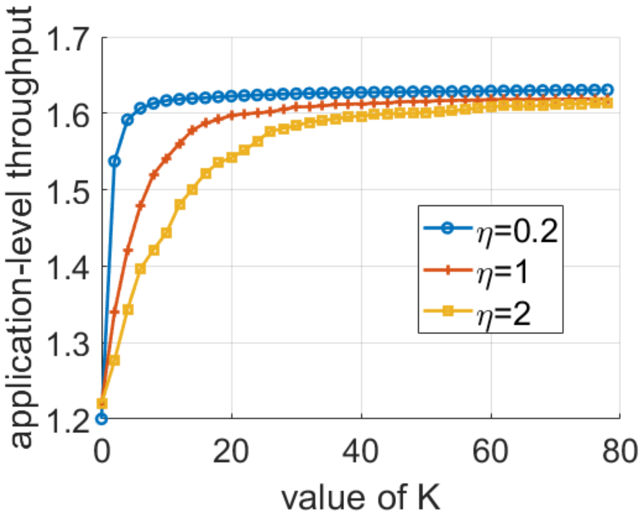
User	r_n
User 2	0.08
User 4	0.05
User 6	0.06
User 8	0.05



Average rate for each user: $\eta = 1$



Average TSLS



Application-level throughput

Related Work

- Panoramic Video Streaming:
  - [Bao, Wu, Zhang, Ramli, Liu, 2016]
  - [Qian, Han, Xiao, Gopalakrishnan, 2018]
  - [Perfecto, Elbamby, Del Ser, Bennis, 2020]
  - [Chen, Li, Srikant, 2020]
  - …
- Wireless Scheduling Design:
  - [Li, Li, Eryilmaz, 2014]
  - [Neely, 2010]
  - [Hou, Kumar, 2013]
  - …

Conclusions

- The successful viewing probability as the function of the delivered portion
- A motion-prediction-based scheduling algorithm by integrating it into the stochastic network optimization framework
- The proposed algorithm can provide desired application-level throughput and service regularity guarantees
- Simulation results with real datasets demonstrated the efficiency of our proposed algorithm

Thank you!