Drone Networks for Virtual Human Teleportation

Jacob Chakareski
Electrical and Computer Engineering Department, The University of Alabama

ABSTRACT

We consider a drone-based vision sensor network that captures collocated viewpoints of the scene underneath and sends them to a remote user for volumetric 360-degree navigable visual immersion on his virtual reality head-mounted display. The reconstruction quality of the immersive scene representation on the device and thus the quality of user experience will depend on the signal sampling rate and location of each drone. Moreover, there is a limit on the aggregate amount of data the network can sample and relay towards the user, stemming from transmission constraints. Finally, the user navigation actions will dynamically place different priorities on specific viewpoints of the captured scene. We make multiple contributions in this context. First, we formulate the viewpoint-priority-aware scene reconstruction error as a function of the assigned sampling rates and compute their optimal values that minimize the former, for given drone positions and system constraints. Second, we design an online view sampling policy that takes actions while exploring new drone locations to discover the best drone network configuration over the scene. We characterize its approximation versus convergence characteristics using novel spectral graph analysis and show considerable advances relative to the state-of-the-art. Finally, to enable the drone sensors to efficiently communicate their data back to the aggregation point, we formulate computationally efficient rate-distortion-power optimized transmission scheduling policies that meet the low-latency application requirements, while conserving the available energy. Our experimental results demonstrate the competitive advantages of our approach over multiple performance factors. This is a first-of-its-kind study of an emerging application of prospectively broad societal impact.

1. INTRODUCTION

Cyber-physical/human systems (CPS/CHS) are set to play an increasingly prominent and important role in our lives and society, advancing research and technology across different disciplines at the same time. Virtual/augmented reality (VR/AR) and drone swarms are two emerging CPS/CHS technologies of prospectively broad societal impact. VR/AR suspends our disbelief of being there (at a remote location), akin to *virtual human teleportation* [1]. Its ongoing explosion onto the market and into the mainstream[1] is a forerunner of the opportunities lying ahead. In brief, by equipping us with the capacity to travel virtually and apply super-human-like vision, VR/AR can help us achieve a broad range of technological and societal advances[2]. Drones can have a similar transformative impact on remote sensing applications, by lowering their cost and extending their scope [3].

We envision a future where drone-deployed VR/AR immersive communication will *break existing barriers* in remote sensing, monitoring, localization, navigation, etc. The thereby achieved advances will benefit diverse applications spanning the environmental/weather sciences, public/national safety, disaster relief, and transportation. By dispensing with the need for our physical presence (transportation) there, they will *make impact on climate change*[3], as well.

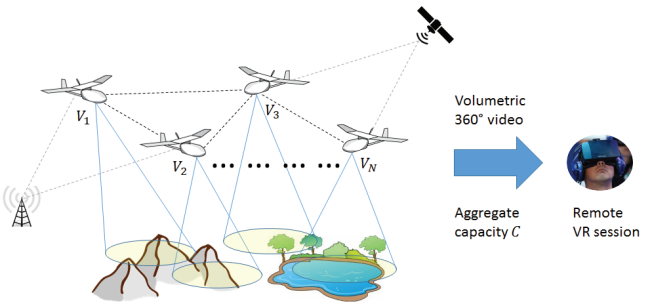

Figure 1: Virtual human teleportation via drone networks.

In this paper, we study a multi-view sensing drone network that delivers a VR immersive communication session to a remote user, as illustrated in Figure 1. The drones are spatially distributed over an area of interest, recording different viewpoints V_i of the scene underneath. The aggregate communication capacity of the sensing system is C and needs to support the delivery of the captured data to the remote user, for volumetric 360-degree-navigable visual immersion in the scene on his VR device. The reconstruction quality of the scene will depend on the location of each drone, the sampling rate R_i at which each V_i has been acquired, and the user navigation actions on the VR device.

[1]As evidenced by the flurry of related devices, services, and platforms appearing on the market, released broadly by electronics/device manufacturers and Internet companies, and stunning investment/acquisition deals/growth rates. It is expected that AR/VR revenue will hit $150B by 2020, disrupting the mobile market along the way [2].

[2]Seeing from multiple 3D perspectives, around and through obstacles, and in ways not possible before will lead to knowledge discovery that will considerably enhance human and machine perception and decision making in diverse tasks. These gains will stimulate societal-scale applications that will enhance energy conservation, quality of life, and the global economy.

[3]According to the U.S. Environmental Protection Agency, human transportation contributes to 1/3 of our carbon emission footprint [4]. Climate and weather disasters attributed to climate change alone cost the American economy more than $100B in 2012 [5].

DroNet'17 June 23, 2017, Niagara Falls, NY, USA

ISBN 978-1-4503-4960-4/17/06…$15.00
DOI: http://dx.doi.org/10.1145/3086439.3086448

There are multiple objectives that we pursue in this context. First, we formulate the viewpoint-priority-aware scene reconstruction error as a function of the assigned sampling rates and compute their optimal values that minimize the former, for given drone positions and system constraints. Second, we design an online view sampling policy that takes actions while exploring new drone locations to discover the best drone network configuration over the scene. We characterize its approximation versus convergence characteristics using novel spectral graph analysis and show considerable advances relative to the state-of-the-art. Finally, to enable the drone sensors to efficiently communicate their data back to the aggregation point, we formulate computationally efficient rate-distortion-power optimized transmission scheduling policies that meet the low-latency application requirements, while conserving the available energy. Our experimental results demonstrate the competitive advantages of our approach over multiple performance factors. This is a first-of-its-kind study of an emerging application of prospectively broad societal impact.

Due to the complex interdependencies of the three system aspects above, in this preliminary study, we analyzed and optimized each individually, while still keeping them coupled in their design/operation through the selected sampling rates that appear in each. Motivated by our preliminary results and advances, we plan to investigate their tighter integration, as part of our ongoing/future work efforts, and analytically quantify the benefits of such an approach. Similarly, we plan to derive further theoretical insights and analysis of our optimization methods as part of a follow-up study, which could not be included here due to limited space.

The rest of the paper is organized as follow. First, we review related work in Section 2. Then, we describe our system modeling in Section 3, which is followed by a formulation of the space-time view sampling problem for a given drone network in Section 4. Analysis and optimization solution of this problem are provided in Section 5. Our online algorithm for cooperative drone sensing and control is described subsequently in Section 6. Finally, the design of the computationally efficient rate-distortion-power optimized transmission scheduling policies that the drones will facilitate to communicate their data is included in Section 7. Comprehensive experimental analysis and benchmarking are provided in Section 8, and we conclude in Section 9.

2. RELATED WORK

Drone-based vision sensor networks for remote VR immersion is a new topic that has not been studied before. Closely related areas include ground-based multi-camera wireless sensing for multi-view systems [6], immersive telecollaboration [7, 8], multi-view video coding/communication [9–12], and 360° video streaming [13, 14]. Another related area is graph-based signal processing for time-varying point clouds used in AR applications [15, 16]. Finally, machine learning applied to aerobatic helicopter control has been studied in [17]. Our objective of selecting the best scene viewpoints to capture for remote VR immersion is very different from state-of-the-art robotic exploration studies that aim to traverse a collection of (aerial or ground-based) waypoints such that the travel time is minimized or an information metric about the monitored phenomena is maximized [18–20]. Similarly, weighted random walks have been used used in robotics before, however, the present study is the first to formally characterize and control the trade-off between achieved expected reward and policy convergence time of reinforcement learning for UAV path planning, via spectral graph analysis and such models.

3. SYSTEM MODEL

Let $V = \{V_1, \ldots, V_N\}$ be the collection of aerial viewpoints captured by the drones. For every drone $\nu \in V$ (the association between captured viewpoints and UAVs is unique), let R_ν denote the temporal (data) sampling rate used in capturing viewpoint ν. The aggregate transmission capacity the swarm can use to send the captured data is C. The VR device constructs a 3D 360°-navigable video representation of the scene of interest from the captured viewpoints. Finally, let $R = (R_1, \ldots, R_N)$ denote the aggregate sampling rate vector.

4. SPACE-TIME VIEW SAMPLING

Let $Q_I = \int_{\nu \in \mathcal{V}} \gamma_\nu Q_\nu(R)$ denote the reconstruction quality of the scene on the VR device, where $\mathcal{V}$ denotes the aggregate view space for the scene, Q_ν is the reconstruction quality of an arbitrary viewpoint $v \in \mathcal{V}$, and γ_ν is the popularity (relevance) of viewpoint ν for the user[4]. In words, Q_I represents the expected reconstruction quality observed by the user across $\mathcal{V}$. Virtual (non-captured) views $\nu \in \mathcal{V}$ are synthesized for the user from the captured data via geometric signal processing[5] (DIBR [23]).

The problem of interest can then be formulated as

$$\max_R Q_I, \text{ subject to: } \sum_i R_i \leq C. \tag{1}$$

Note that R in essence samples the viewpoint locations V across time and space, simultaneously. In particular, $R_i = 0$ signifies that signal/location V_i is not sensed (at all). Note also that Q_I integrates the impact of the latency and interactivity requirements of the VR application.

5. ANALYSIS AND OPTIMIZATION

In our recent work [10], we have shown that when virtual view $\nu \in \mathcal{V}$ is interpolated via DIBR from the two nearest sampled viewpoints $V_i, V_j \in \mathcal{V}$, its reconstruction quality $Q_\nu(R)$ can be represented as a linear function of $Q_{V_i}(R_i)$ and $Q_{V_j}(R_j)$, each multiplied by polynomial powers of x, the relative position of ν with respect to them.

Armed with this representation of $Q_\nu(R)$, for virtual views ν, we can reformulate the objective in (1) by grouping terms associated with $Q_i(R_i) = Q_{V_i}(R_i), \forall i$, to have

$$\max_R \sum_i \alpha_i Q_i(R_i), \text{ subject to: } \sum_i R_i \leq C. \tag{2}$$

where the weights $\alpha_i \geq 0$ represent the aggregated terms.

(2) now represents a convex optimization problem [24]. In particular, the weights α_i can be normalized to add to one, and the functions $Q_i(R_i)$ can be accurately represented as concave functions $\exp(-\beta_i R_i)$ [25]. Thus, (2) can be efficiently solved using convex optimization methods [24].

[4]This quantity stems from the user view navigation pattern.
[5]This operation requires scene depth signals that can be captured via IR sensors [21] or estimated from the captured viewpoints (color signals) [22].

6. COOPERATIVE DRONE SENSING

We consider that the drones can sample viewpoints of the scene from a collection of locations (way-points) $\{l_i\}$ that we model as a graph. Due to the scene characteristics and topology, we assume there is a non-uniform cost c_{ij} of traversing the link between every two adjacent locations l_i and l_j. We discretize time and consider that at every time slot t, a drone can either stay at its current location (hovering) or move to one of the neighboring locations. Let S_t denote the current configuration (state) of the drone swarm across the graph. At the onset of the next time slot $t+1$, the swarm can collectively transition into another state denoted as S_{t+1}. The values that S_{t+1} can attain depend on S_t. Let $a = S_{t+1}$ denote the state transition action the swarm will take at time $t+1$. Furthermore, let $r(S_t, a)$ denote the reward the swarm will earn at time $t+1$ by carrying out a, given S_t. We formulate $r(S_t, a)$ as $u(S_t, a) - c(S_t, a)$, where $u(S_t, a)$ represents the utility the swarm will achieve by taking action a, given the current configuration S_t. Similarly, $c(S_t, a)$ represents the cost incurred by taking action a given S_t, accounted for as the sum of all cost factors c_{ij} activated by executing a. In our case, $c(S_t, a)$ captures the energy the swarm expends to move from state S_t to state S_{t+1} over the scene, while the utility $u(S_t, a)$ represents the scene reconstruction quality $Q_I = \int_{\nu \in \mathcal{V}} \gamma_\nu Q_\nu(S_{t+1}, R)$.

We are interested in finding the path planning policy $\pi : S_t \rightarrow a$ that will maximize the cumulative reward $\sum_t r(S_t, a)$. In addition to the necessarily online nature of π, which will have to explore different swarm configurations for new information (reward values) and exploit already gathered information (known reward values), another major challenge here is the discrete complex nature of this problem (*excessive search space*). Thus, we design π as a probabilistic policy, where every prospective action a is selected with probability p_a (having an expected reward objective $\sum_t \sum_a p_a\, r(S_t, a)$). This will allow for a richer policy/action space to be explored/harnessed, enabling connections to random walks, spectral graph theory, and convex optimization, as well.

Let $G = (\mathcal{S}, \mathcal{E})$ denote the state-space graph. We can show that π is equivalent to a non-negative symmetric weight matrix W that defines a weighted random walk on G [26]. Thus, finding the optimal π is equivalent to finding the corresponding W. To reduce the complexity of the latter, we parameterize W using a vector $\theta = (\theta_1, \ldots, \theta_m)$ and a dictionary of symmetric weight matrices W_i such that $W = \sum_i \theta_i W_i$, where $\sum_i \theta_i = 1$. The expected reward associated with the policy induced by θ is denoted as $\rho(\theta)$. Conventional MDP theory instructs an infinite execution time for an optimal policy to ensure expected reward maximization [27]. But, in our case, we want to compute θ that maximizes $\rho(\theta)$ quickly, for better scene dynamics adaptation. Moreover, we want to be able to characterize/control formally the trade-off between tolerable policy convergence time and approximate maximum expected reward earned thereby. Now, executing W on G is equivalent to running a Markov Chain on it with the transition probability matrix P_θ induced by W. Thus, how quickly our policy achieves the maximum expected reward is equivalent to how quickly the Markov Chain converges to its stationary distribution $\boldsymbol{\pi}_\theta$ over $\mathcal{S}$.

We quantify the latter precisely via the mixing time t_{mix} of the chain, which represents the smallest number of steps n to run the chain over G to achieve an ϵ-approximation of $\boldsymbol{\pi}_\theta$ [26]. In turn, t_{mix} is upper bounded by the second largest eigenvalue modulus $\mu(P_\theta) = \mu(\theta)$ of G (i.e., the matrix P_θ) as $t_{\text{mix}}(\epsilon) \leq \frac{A(\epsilon)}{1-\mu(P_\theta)}$ [26], where $A(\epsilon)$ is a constant. Thus, $\mu(\theta)$ governs for how long our policy needs to run to achieve the maximum reward. Hence, we can effectively control the latter via the former. Given these advances, we can formulate our policy optimization problem as

$$\textbf{P1}\text{: Maximize:} \quad \rho(\theta), \tag{3}$$
$$\text{subject to:} \quad \mu(\theta) \leq \delta, \tag{4}$$
$$\theta_i \geq 0,\ \textstyle\sum_{i=1}^m \theta_i = 1. \tag{5}$$

$$\textbf{P2}\text{: Maximize:} \quad \rho(\theta) - \lambda\mu(\theta) \tag{6}$$
$$\text{subject to:} \quad \theta_i \geq 0,\ \textstyle\sum_{i=1}^m \theta_i = 1. \tag{7}$$

where in **P1**, (4) ensures that the computed policy is sufficiently fast. Our objective in **P2** is to use λ to control the trade-off between the achieved expected reward and policy convergence rate, for effective adaptation to scene dynamics.

Next, we derive expressions for $\rho(\theta)$ and $\mu(\theta)$. Let $p_\theta(s, i)$ be the probability that we select policy W_i while in state s, given θ. Then $p_\theta(s, i)$ can be computed as

$$p_\theta(s, i) = \frac{\theta_i H_i(s)}{H(s)}, \forall s, i, \tag{8}$$
$$\text{where } H_i(s) = \sum_j \mathbf{W}_i(s, j), H(s) = \sum_j \mathbf{W}(s, j).$$

Then, since action a may appear in multiple policies W_i, given state s, the probability that it is selected can be computed as $d_\theta(s, a) = \sum_{i:s\rightarrow a} p(s, i)$. Consequently, the expected reward can be computed as:

$$\rho(\theta) = \sum_s \boldsymbol{\pi}_\theta(s) \sum_a d_\theta(s, a) r(s, a). \tag{9}$$

Furthermore, we formulate $\mu(\theta)$ as:

$$\mu(\theta) = ||D_\theta^{-1/2} W D_\theta^{-1/2} - \sqrt{\boldsymbol{\pi}_\theta}\sqrt{\boldsymbol{\pi}_\theta}^T||_2, \tag{10}$$

where D_θ is a diagonal matrix with diagonal elements taken from the stationary distribution $\boldsymbol{\pi}_\theta$. We can show that $\rho(\theta)$ and $\mu(\theta)$ are convex functions[6]. Thus, we can address **P1** and **P2** using tools from convex optimization [24].

Algorithm 1 Fast Explore and Exploit Learning (FEEL)

Require: $\theta^0, \alpha, \lambda, k = 0, \epsilon$, initial state s_0
1: **repeat**
2: Execute the parameterized policy W for N epochs.
3: Observe and update the earned reward $r(s, a)$.
4: Compute $\nabla\rho(\theta)|_{\theta^k}$ and $\nabla\mu(\theta)|_{\theta^k}$
5: Update $\theta^{k+1} = \theta^k - \alpha\left(\nabla\rho(\theta)|_{\theta^k} - \lambda^k \nabla\mu(\theta)|_{\theta^k}\right)$
6: Increment iteration counter $k = k + 1$
7: Project θ^k back to feasible set (if necessary).
8: **until** *convergence* (Change in objective function $< \epsilon$)

To verify our idea, we designed a gradient descent with projection onto convex sets algorithm (FEEL) to solve **P2**, shown in Algorithm 1. As the gradients $\nabla\rho(\theta)$ and $\nabla\mu(\theta)$ can be computed in closed form, FEEL is implemented effectively, adapting λ at every iteration to ensure fast policy convergence by initially emphasizing reward structure exploration and then gradually transitioning to expected reward

[6]To conserve space, we omit this derivation here.

exploitation/maximization [28]. We can analytically derive the necessary conditions on λ to ensure convergence[6].

7. DRONE TRANSMISSION SCHEDULING

We model the transmission system of a drone as illustrated in Figure 2. The data captured by the vision sensor is queued for transmission. Power control and transmission scheduling actions are carried out to decide the packets to be transmitted next and the amount of transmit power to be used to carry that out. Via receiver feedback, the transmitter is aware of the current forward channel quality.

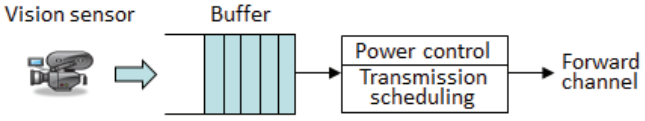

Figure 2: Drone vision sensor transmission system.

The captured viewpoint data is encoded by the vision sensor. The encoded data units exhibit prediction dependencies, as illustrated in Figure 3, which are induced on them by the sensor encoder, to increase its compression efficiency. This, however, makes the decision logic of acting upon them more challenging, as illustrated next. In particular, let there be L data units currently enqueued in the transmission buffer. Each data unit l is characterized with its delivery deadline $t_{l,d}$, size B_l in bytes, reduction in reconstruction error ΔD_l of the respective viewpoint that l will contribute to, if received and decoded on time, and two sets of ancestor and descendant data units $\{j \preceq l\}$ and $\{j \succeq l\}$, induced by the encoding prediction dependency graph. $t_{l,d}$ represents the latest time by which l needs to be received, in order to be usefully decoded.

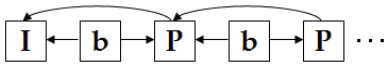

Figure 3: The data unit dependencies of an encoded viewpoint.

The forward channel is modeled as a packet erasure channel, with loss probability $\epsilon \triangleq \epsilon(h,c)$, where h denotes the current channel state/quality, as informed by receiver feedback, and c the selected transmit power, and transmission delay τ, which may be non-deterministic, with density $p_\tau \triangleq p_\tau(h,c)$. We consider that c can be selected from a finite set of possible choices, i.e., power levels, denoted as C.

Now, let $\pi = (\pi_1, \ldots, \pi_L)$ denote the packet transmission policy of the drone, where $\pi_l \in \{0,1\}$ indicate the two possible choices of not sending or sending data unit l at present (current time t). Similarly, let $c = (c_1, \ldots, c_L)$ denote its transmit power control policy, where $c_l = 0$, for $l : \pi_l = 0$. Furthermore, let $\epsilon(\pi_l) \triangleq P\{\tau_l > t_{l,d} + \alpha - t\}$ be the expected error of transmitting data unit l under policy π_l, where α captures the latency/interactivity requirements of the VR application. It can be computed as

$$\epsilon(\pi_l) = \begin{cases} 1, & \text{if } \pi_l = 0, \\ \epsilon + (1-\epsilon)\int_{t_{l,d}+\alpha-t}^{\infty} p_\tau, & \text{if } \pi_l = 1. \end{cases} \quad (11)$$

The expected reduction in aggregate reconstruction error under policy π can then be formulated as

$$D(\pi, c) = \sum_l \Delta D_l \prod_{j \preceq l} (1 - \epsilon(\pi_j)). \quad (12)$$

The corresponding transmission rate can be formulated as $R_T(\pi) = \sum_{l:\pi_l=1} B_l = \sum_l \pi_l B_l$. Finally, let $E(\pi, c) = \sum_l c_l \pi_l B_l$ denote the consumed transmit power. The sender is interested in solving the following optimization problem

$$\max_{\pi,c} D(\pi, c), \quad (13)$$
$$\text{subject to: } R_T(\pi) \leq R_i, E(\pi, c) \leq E_B,$$

where E_B is the available transmit power budget. Recall that R_i represents the sampling rate allocated to drone i.

Note that (13) represents a discrete optimization problem that is therefore challenging to solve. An exact solution requires a brute-force search to enumerate the $2^L \times |C|^L$ choices that (π, c) can take on as a pair. Instead, we design a much faster iterative algorithm that computes an approximate optimal solution at much lower complexity, as follows.

Using the Generalized Lagrange multiplier method [29], we reformulate (13) as the unconstrained optimization

$$\min_{\pi,c} -D(\pi, c) + \lambda_1 R_T(\pi) + \lambda_2 E(\pi, c), \quad (14)$$

where for mathematical convenience, we have replaced the max operator from (13) with a min operator. $\lambda_i > 0$ denote the corresponding Lagrange multipliers. Starting from an initial solution, we then iteratively solve (14) one variable pair (π_l, c_l) at a time, while keeping the others $((\pi_j, c_j)$, for $j \neq l)$ fixed, until convergence.

Precisely, let $(\pi^{(0)}, c^{(0)})$ denote an initial choice for the joint transmission scheduling/power control policy. Further, let $n = 1, 2, \ldots$ denote the iteration count. Select one policy pair $(\pi_l^{(n)}, c_l^{(n)})$ to optimize at iteration n, e.g., in a round-robin fashion, for $l = 1, \ldots, L$. Set (fix) the remaining policy pairs $(\pi_j^{(n)}, c_j^{(n)}) = (\pi_j^{(n-1)}, c_j^{(n-1)})$, for $j \neq l$, and compute the values of $(\pi_l^{(n)}, c_l^{(n)})$ that solve (14). Increment n and move on to the next l, until $J(\pi^{(n)}, c^{(n)}) = J(\pi^{(n-1)}, c^{(n-1)})$, where $J(\pi, c)$ represents the objective function of (14).

By grouping terms in (14) associated with (π_l, c_l), we can formulate the key step of the iterative optimization as

$$\pi_l^{(n)}, c_l^{(n)} = \arg\min_{\pi_l, c_l} S_l^{(n)} \epsilon(\pi_l) + \lambda \pi_l B_l, \quad (15)$$

where $\lambda = \lambda_1 + \lambda_2 c$ and $S_l^{(n)} = \sum_{j \succeq l} \Delta D_j \prod_{m \preceq j, m \neq l} (1 - \epsilon(\pi_m^{(n)}))$. $S_l^{(n)}$ captures the impact of data unit l on the reconstruction quality of viewpoint i. The optimization is guaranteed to converge, as the objective function $J(\pi, c)$ is bounded from below and is monotonically non-increasing at every iteration. It is formally described in Algorithm 2.

Algorithm 2 Optimal transmission scheduling policy

Require: $\pi_l^{(0)}, c_l^{(0)}, \lambda_1, \lambda_2, n = 0, \theta$
1: **repeat**
2: $n = n + 1$; $l = (n \mod L)$
3: Solve (15) $\Rightarrow \pi_l^{(n)}, c_l^{(n)}$
4: **until** *convergence* $(J^{(n-1)} - J^{(n)} < \theta)$

8. EXPERIMENTS

We carry out experiments to evaluate the effectiveness of our framework. They involve aerial viewpoint data captured in both, a controlled laboratory setting and an outdoor (in-the-wild) environment [30]. For reproducibility and scientific analysis, our experimentation methodology was carried

as follows. Viewpoint data was acquired using DJI Phantom 4 UAVs equipped with 4K cameras and 4G LTE dongles [31]. Since this is a proof-of-concept study of a novel application and given the limited space here, we only considered a one-hop aerial network in our experiments. That is, each UAV is directly connected to a ground-based base station to which it sends its data. We collected measurements of the wireless link to the base station for different UAV locations. These traces and the UAV captured data were then used to apply our optimization framework components via numerical simulations. The viewpoint popularity distribution γ_ν was compiled based on traces of user head movement motion that we collected. We designed a reference system to compare to, as follows. The drones sample the captured viewpoints at equal rate $R_i = C/N$ (uniform allocation) and employ conventional Automatic Repeat Query (ARQ) scheduling [32] to transmit the captured data towards the destination. The reference system is denoted as *Baseline* and our optimization framework as *Optimal*, henceforth. H.264 video encoding is used to encode the captured viewpoint data at each drone.

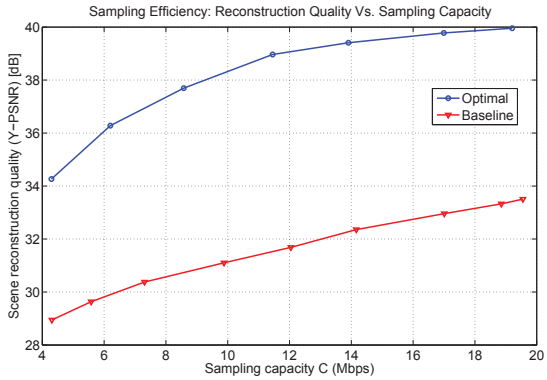

Figure 4: Recon. quality vs. sampling capacity (outdoors).

Sampling efficiency. In Figure 4, we examine the sampling efficiency of our framework relative to the baseline reference. In particular, we graph the achieved expected scene reconstruction quality (the inverse of the objective in (1)) versus the available sensing capacity, for the two methods. It can be seen that the optimization is able to leverage the available resources much more effectively, enabling considerable reconstruction quality gains over the baseline system, for the same C. Furthermore, we can see that *Optimal* is able to upscale its performance at much higher rate as C is increased, relative to *Baseline*. The equivalent results for the indoor data look similar. To conserve space, we do not include them here.

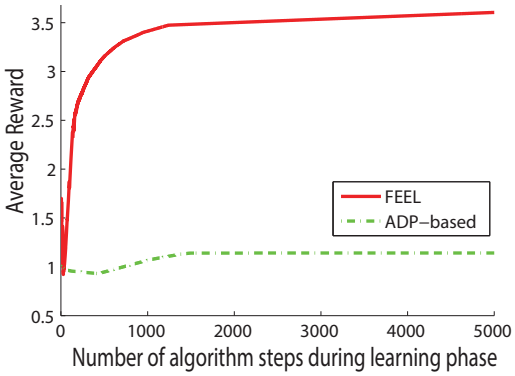

Figure 5: Expected reward vs. iteration count.

Online policy learning. To examine the efficiency of our algorithm for cooperative drone sensing and control from Section 6, denoted as FEEL (Algorithm 1), we compared it to a state-of-the-art learning method based on adaptive dynamic programming (ADP) [33] and observed that FEEL demonstrates considerable performance gains. In particular, we applied both methods to data we collected in our lab associated with a small state-space graph of prospective drone network states over the scene comprising 32 states. The achieved normalized expected reward versus the number of executed algorithmic steps is shown in Figure 5. We can see that FEEL notably outperforms ADP by leveraging its ability to initially explore the policy space quickly.

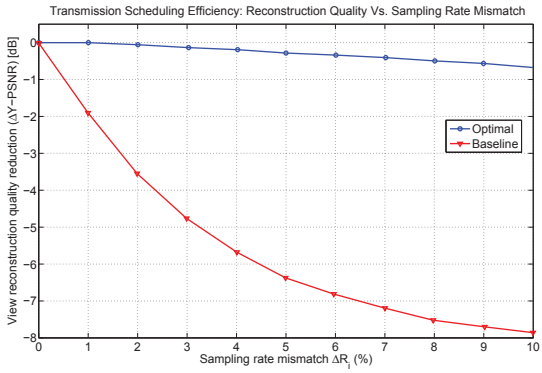

Figure 6: Transmission efficiency versus rate mismatch ΔR_i (%).

Transmission/power scheduling efficiency. We carry out two experiments here to examine the adaptivity of our transmission scheduling to sampling rate mismatch and its robustness to unreliable transmission, respectively. In particular, in the first experiment, we consider that the captured viewpoint data $\{V_i\}$ has been sampled at assigned target sampling rate R_i (12 MBps here), however, the actual available transmission rate at which it can be sent to the ground-based back-end station is smaller, due to temporal channel fluctuations. We measure the expected reconstruction quality of the viewpoint for different values of the mismatch ΔR_i, expressed in percent of R_i. These results are shown in Figure 6. We can see the our approach offers a graceful degradation as ΔR_i is increased. In fact, due to its rate-distortion optimized design, our transmission scheduling is able to select the most important data units to transmit such that its transmission efficiency is consistently maximized across the whole range of sampling rate mismatch values examined, which in turn minimizes the reduction in reconstruction quality of the viewpoint, due to the latter, as evident from Figure 6. On the other hand, even a small mismatch can degrade the end-of-the-end performance of the baseline scheduling method considerably, as Figure 6 shows, due to its signal-agnostic operation.

The second experiment measures the achieved reconstruction quality as the average packet loss rate ϵ on the aerial network link is increased. These results are shown in Figure 7. We can see that our transmission scheduling offers robustness and graceful degradation to increasing packet loss, without dramatically penalizing the end-to-performance. On the other hand, the baseline system exhibits a considerable degradation in performance, even for small packet loss rate values, which then progressively increases further, as the packet loss increases, as shown in Figure 7.

Finally, we observed in our experiments that in normal

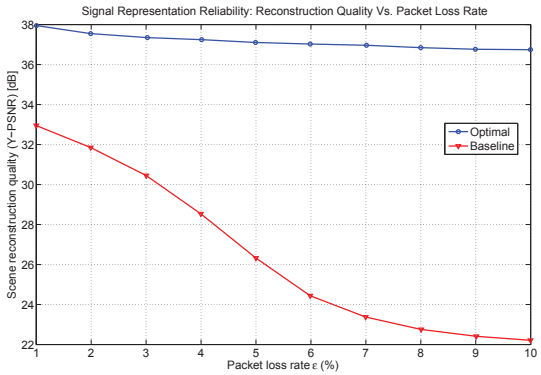

Figure 7: Transmission robustness versus packet loss rate ϵ (%).

operation conditions our power control scheduling consumes approximately 25-30% less transmit power relative to the baseline reference due to its optimized design. This implies that our design can operate on a considerably tighter energy budget, while still delivering the same performance, which is encouraging for practical deployment.

9. CONCLUSION

We studied drone-based vision sensor networks for virtual human teleportation, where the drones capture collocated viewpoints of the scene underneath for remote volumetric 360-degree navigable visual immersion on a VR device. Our contributions in this context are multiple. First, we formulated the viewpoint-priority-aware scene reconstruction error as a function of the assigned viewpoint sampling rates to each drone and compute their optimal values that minimize the former, for given drone positions and system constraints, where the viewpoint priorities are induced by the user navigation actions. Second, we design an online view sampling policy that takes actions while exploring new drone locations to discover the best drone network configuration over the scene, as the drone network does not have an a priori scene viewpoint knowledge. We characterize the approximation versus convergence characteristics of our policy using novel spectral graph analysis and show considerable advances relative to the state-of-the-art. Finally, to enable the drone sensors to efficiently communicate their data back to the aggregation point, we formulate computationally efficient rate-distortion-power optimized transmission scheduling policies that meet the low-latency application requirements, while conserving the available energy. Our experimental results demonstrate competitive advantages over conventional methods employed in practice and motivate further investigation.

10. REFERENCES

[1] J. G. Apostolopoulos, et al., "The road to immersive communication," *Proceedings of the IEEE*, vol. 100, no. 4, pp. 974–990, Apr. 2012.

[2] T. Merel, "Augmented and virtual reality to hit $150 billion, disrupting mobile by 2020," Tech Chrunch Online Magazine, Apr. 2015.

[3] Agricultural Drones: MIT TR: 10 Breakthrough Technologies 2014.

[4] US Environ. Protection Agency, "U.S. Greenhouse Gas Inventory Report: 1990-2014," Apr. 2016.

[5] The White House. Climate Change Action Plan. http://www.whitehouse.gov/climate-change/

[6] J. Chakareski, "Uplink scheduling of visual sensors: When view popularity matters," *IEEE Trans. Communications*, Feb. 2015.

[7] R. Vasudevan, et al., "Real-time stereo-vision system for 3D teleimmersive collaboration," in *Proc. IEEE ICME*, Jul. 2010.

[8] M. Hosseini and G. Kurillo, "Coordinated bandwidth adaptations for distributed 3D tele-immersive systems," in *Proc. ACM MMVE Workshop*, Mar. 2015.

[9] G. Cheung, et al., "Interactive streaming of stored multiview video using redundant frame structures," *IEEE TIP*, Mar. 2011.

[10] J. Chakareski, et al., "User-action-driven view-rate scalable multiview coding," *IEEE TIP*, Sep. 2013.

[11] J. Chakareski, "Wireless streaming of interactive multi-view video via network compression and path diversity," *IEEE TCOM*, Apr. 2014.

[12] J. Chakareski, et al., "View-popularity-driven joint source and channel coding of view and rate scalable multi-view video," *IEEE JSTSP*, Apr. 2015.

[13] F. Qian, et al., "Optimizing 360 video delivery over cellular networks," in *Proc. ACM Workshop All Things Cellular*, Oct. 2016.

[14] M. Hosseini and V. Swaminathan, "Adaptive 360 VR video streaming: Divide and conquer!" in *Proc. IEEE ISM*, Dec. 2016.

[15] H. Q. Nguyen, et al., "Compression of human body sequences using graph wavelet filter banks," in *Proc. IEEE ICASSP*, May 2014.

[16] D. Thanou, et al., "Graph-based compression of dynamic 3D point cloud sequences," *IEEE Trans. Image Processing*, Apr. 2016.

[17] A. Ng, et al., "Inverted autonomous helicopter flight via reinforcement learning," in *Proc. Int'l Symp. Experimental Robotics*, Jun. 2004.

[18] W. Burgard, et al., "Coordinated multi-robot exploration," *IEEE Trans. Robotics*, May 2005.

[19] J. Delmerico, et al., "Active autonomous aerial exploration for ground robot path planning," *IEEE Robotics and Automation Letters*, Apr. 2017.

[20] G. Hollinger and G. Sukhatme, "Sampling-based robotic information gathering algorithms," *Int'l J. Robotics Research*, Aug. 2014.

[21] S. Gokturk, et al., "A time-of-flight depth sensor – system description, issues and solutions," in *Proc. IEEE CVPR Workshop*, Jun. 2004.

[22] M. Tanimoto, T. Fuji, and K. Suzuki, "Multi-view depth map of Rena and Akko & Kayo," ISO/IEC MPEG Document M14888, Oct. 2007.

[23] H.-Y. Shum, et al., *Image-Based Rendering*, Springer, Sep. 2006.

[24] S. Boyd and L Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

[25] M. Chen, et al., "Utility maximization in peer-to-peer systems," in *Proc. ACM SIGMETRICS*, Jun. 2008.

[26] D. Aldous and J. A. Fill, *Reversible Markov Chains and Random Walks on Graphs*. http://www.stat.berkeley.edu/~aldous/RWG/

[27] M. Puterman, *Markov Decis. Processes*, Wiley, 1994.

[28] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, MIT Press, 1998.

[29] H. Everett, "Generalized Lagrange multiplier method for solving problems of optimum allocation of resources," *Operations Research*, May-June 1963.

[30] "VIRAT Video Dataset." http://www.viratdata.org/

[31] DJI Camera Drones for Aerial Surveillance. http://www.dji.com/

[32] W. Stevens, *TCP/IP Illustr., Vol. 1: Protocols*, 1994.

[33] F. L. Lewis and D. Liu, *Reinforcement learning and approximate dynamic programming for feedback control*. Wiley, 2013.