Abstract—The increasing popularity of Internet-capable mobile devices leads to the explosion of mobile data traffic. Although the emerging mobile broadband systems such as 4G LTE promise higher bandwidth and lower latency for video traffic, it is not power efficient to deliver video traffic over cellular networks. This paper studies how peer-to-peer communications via wireless local networks could complement the cellular networks so as to minimize the power consumption of mobile devices. Specifically, we consider the scenario in which a group of nearby mobile devices share the streaming contents, originated from the cellular networks, over the wireless local area networks in a peer-to-peer manner. We focus on investigating how the mobile devices should cooperate to minimize their own power consumption. The problem has been formulated into a linear programming (LP) model, and numerical results show that at least 31% of the power consumption can be saved on mobile devices through the cooperation among them.

I. INTRODUCTION

With the proliferation of Internet-capable mobile devices, streaming video to mobile users is becoming more and more popular. Besides the traditional Video on Demand services, content providers have developed applications to deliver real-time contents to mobile devices, e.g., WatchESPN [1].

In this paper, we consider delivering video streaming to a group of co-located mobile users (MUs). To motivate our discussion, we envision a group of classmates are reuniting at their Alma Mater’s Homecoming football game. While watching the football game, they are also viewing the same Major League Baseball (MLB) game using their smartphones at the same time. Fig. 1(a) depicts such a scenario where the MLB video content is streamed to a group of MUs. Originating from a content provider’s cloud, the streaming content is transported over the Internet and through the wireless operator’s private network to a Base Station (BS) which then transmits the data to MUs via the cellular links.

While emerging cellular broadband systems, such as 4G LTE, are capable of catering to the bandwidth and latency requirements [2], [3], the Radio Resource Control (RRC) of the cellular systems is not optimized for delivering streaming traffic [4], [5]. Specifically, during a streaming session, the communication component of an MU has almost no chance of entering the low-power idle state due to the short inter-arrival time of streaming data packets, which leads to high power consumption\(^1\) of the MU. For example, measurement studies such as in [3] have shown that, when using 4G networks to stream video contents to MUs, more than 50% of the system power is consumed by the communication unit.

To overcome the above problem, this paper investigates how the MUs may cooperate with each other to optimize MUs’ power consumption. In such a cooperative scenario, the co-located MUs form an ad-hoc local wireless network via the MUs’ local communication interfaces (e.g., WiFi or Bluetooth). Instead of downloading the entire video stream via cellular links, each MU may only request a sub-stream of the video content over the cellular link. The sub-streams are then transmitted and shared with other MUs via the local wireless networks as depicted in Fig. 1(b). Such a scenario is termed peer-to-peer (P2P) streaming to mobile users.

To decide how the cellular networks and the wireless local networks can be jointly utilized, we assume that the information about the networks, such as the topologies and the capacities of communication links, is collected by a BS. Based on the collected information, the BS formulates and solves an optimization problem that minimizes the maximum power consumption of the MUs. The solution is fed back to the BSs to simplify cellular core networks [6].

The remainder of this paper is organized as follows. Sec. II presents the background and the related work. Then, Sec. III gives a high-level description of the proposed P2P streaming scheme and discusses the relative technical issues to be addressed. The P2P streaming problem is formulated into a LP model with the objective of minimizing the maximum power consumption of MUs in Sec. IV, and numerical results are presented in Sec. V. Finally, Sec. VI concludes this paper and presents future research directions.

\(^1\)Throughout the paper, the term power is used to denote the rate at which energy is consumed.

---

This material is based on research sponsored by the Air Force Research Laboratory (FA9550-12-1-0086) and NSF (CNS-1016841). The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon.
II. BACKGROUND AND RELATED WORK

While 4G LTE network offers much higher data rates and much better services than its predecessors, its power consumption has always been a serious issue. To save power in the MU, the RRC function of 4G LTE networks employs Discontinuous Reception (DRX) as depicted in Fig. 2. Initially an MU is in the RRC_IDLE state. Upon receiving a packet, the MU is promoted to the RRC_CONNECTED state. The MU stays in the Continuous Reception mode (of the RRC_CONNECTED state) and starts a DRX inactivity timer \(T_i\), which is reset by every incoming packet. When \(T_i\) expires, the MU enters into the Short DRX mode and later switches into the Long DRX mode if the MU receives nothing within a certain time period. Finally, the MU may be demoted into the RRC_IDLE state after a long inactive period in the Long DRX mode.

The DRX scheme, however, is not optimized for streaming traffic with high QoS requirement and low wake-up latency [4], [5]. Although allowed to “micro-sleep” in both the Short and the Long DRX modes, an MU has to frequently wake up to monitor the Physical Downlink Control Channel (PDCCH), and the power consumption of the MU in these two modes is much higher than that in the RRC_IDLE state. Given that the inter-arrival time of streaming data packets is usually short, the MU has little chance of returning to the RRC_IDLE state during the entire streaming session, which leads to high device power consumption. Measurement studies have reported that receiving one unit of data over cellular networks consumes more power than transmitting the same unit of data over WiFi (or for the same amount of data, transmission over WiFi consumes less power than reception over 4G LTE) [7]–[9].

In this paper, we consider enabling cooperation among colocated MUs using the P2P local wireless networks in order to deliver video stream to these MUs in a power-efficient manner. Using local wireless networks to complement the cellular networks have been studied by several literatures. For example, Cellular-WiFi systems use wirelessely inter-connected WiFi access points to extend the coverage of the cellular BSs [10], [11]. Other work aims to optimize the resources to efficiently deliver P2P video traffic over wireless networks. In [12], the authors show that P2P cooperation can be used to provide near-live TV stream in a resource-constrained mobile environment. In [13], a cross-layer P2P solution is proposed to distribute live video stream over a mobile ad hoc network by multicasting the streaming content over a spanning tree in an overlay network.

While most existing work uses the P2P networks to reduce MUs’ Internet access requirement [14] and to improve the streaming quality [14], [15], less attention has been paid to the power consumption of the MUs. In this paper, we take into account the different power efficiencies of the cellular links and the local P2P links, and investigate how streaming content should be shared among MUs so as to reduce their power consumption. Unlike existing work such as [16]–[18] where network operators need to supply external rewards (e.g., money) as extrinsic incentives for MUs to assist other MUs’ communications, we will show that, in our scenario, each MU has intrinsic incentives to cooperate as such cooperation can lower the power consumption and prolong the battery lifetime of the MU.

To the best of our knowledge, [19] is the most related work to this paper. In [19], a distributed algorithm named LocalTree is proposed to find tree-like routes for static MUs to disseminate video streaming. Meanwhile, LocalTree builds a streaming mesh for the MUs with high dynamics. However, LocalTree only allows one MU (termed puller) in the network to pull data packets from the BS, which leads to high power consumption of the puller. In contrast, this paper allows multiple MUs to retrieve streaming content from the BS and achieves fair power consumption among cooperating MUs.

III. SYSTEM OVERVIEW

Given that each MU is equipped with multiple radio access technologies, to receive streaming content, an MU can either...
download the streaming content from a base station or use a wireless local area network (WLAN) such as WiFi to receive the streaming from other neighboring MUs. In such a situation, each MU may only request a portion of the streaming content from the base station and shares the received content with other MUs via WLANs. In the meantime, an MU receives the remainder of the streaming content from its neighboring MUs so that the MU can reconstruct the original content.

Our goal is to investigate P2P streaming with respect to the power consumption of MUs. Particularly, we aim to determine the P2P cooperative strategy for delivering streaming to MUs that achieves the minimum MUs’s power consumption. To facilitate such a usage scenario, the following issues must be jointly addressed:

1) Streaming data routing. We use the fluid flow model [20] to model video streaming with an encoding rate of $S$ (bits/second). With this model, the original streaming can be divided into “sub-streams” with arbitrary data rates and each sub-stream may be independently routed over the network following different paths. An MU receives the streaming from multiple sub-streams and combines them together. When receiving streaming content from multiple sub-streams at an aggregated rate of $S$, an MU can reconstruct the original streaming content. We aim to achieve the global streaming of rate $S$ in the network such that all MUs can playback the streaming content at the original encoding rate. For example, in Fig. 3(a), a stream is divided into two sub-streams and each sub-stream traverses the network over a different path. Global streaming is achieved when $s_1 + s_2 = S$, with $s_1$ and $s_2$ denoting the data rates of the two sub-streams, respectively.

To reduce each MU’s power consumption, data duplication must be avoided and each sub-stream needs to be routed over some tree structure imposed on the network so that no MU receives the same content more than once. Consequently, routing the sub-streams is equivalent to assigning a sub-stream data rate to each possible tree in the network. Additionally, we need to ensure that the sum of the data rates of the trees each MU joins is equal to $S$, so that every MU can reconstruct the original streaming. However, because the number of possible trees is exponential in the size of the network, simply listing all the trees is not efficient. Thus, we also make use of the notion of conceptual flow [21] to break a sub-stream tree into multiple unicast conceptual flows from the base station to the MUs, and each MU receives streaming data from multiple conceptual flows. For example, Fig. 3(b) shows that the sub-stream $s_1$ is broken into three unicast conceptual flows and Fig. 3(c) shows that two conceptual flows are delivering streaming data to MU 1. Several conceptual flows can share the same physical link without contending for the link capacity, while the actual data rate carried over the physical link is determined by the maximum rate of the conceptual flows over the link. As indicated in [22], it is possible to allocate the streaming content to be delivered over each physical link with the knowledge of the rate of the conceptual flows.

2) Local P2P link capacity. When sharing the streaming content with neighboring MUs over WLANs, we need to consider the limited capacity of the P2P links, i.e., each MU has limited capacity in transmitting data over its outgoing P2P links and receiving data over its incoming P2P links. We do not consider the ‘broadcast advantage’ of wireless transmissions (where one source can send data to multiple destinations using one transmission) since most current P2P applications use application-level multicast, i.e., multicast is realized by multiple concurrent IP unicast sessions managed by the applications. Also, to manage transmission conflicts, we assume there is a perfect transmission scheduling algorithm to avoid transmission contentions in P2P links.

3) MU power consumption model. We take into account the MUs’ communication power consumption for both the cellular network and the local P2P network. We model the power consumption of an MU by using the power consumption in each RRC state and the percentage of time the MU stays in the state. While the power consumption in each state is usually fixed and determined by the hardware, an MU’s sojourn time in each state is determined by the RRC timeout mechanism as well as by the traffic pattern. Thus, accurately modeling an MU’s power consumption is difficult and requires knowledge about the average packet size and the inter-packet delay. In our study, we adopt the simplified model proposed in [9] and assume that the average power consumption $P$ is linearly proportional to the throughputs, i.e.,

$$P = \alpha^C + \alpha^W = \alpha^C + \alpha^W t_d + \beta, \tag{1}$$

where $t_d^c$, $t_d^w$ and $t_d^d$ are the throughput in the cellular downlink, the P2P outgoing links, and the P2P incoming links, respectively, and $\alpha^C$, $\alpha^W$ and $\alpha^d$ are the power efficiencies on the corresponding links. $\beta$ is the base power consumption of an MU. Notice that, in the following discussion, we do not consider the traffic in the cellular uplink, since, in most existing streaming applications, the data traffic mainly flows from the base station to the MUs.

IV. PROBLEM FORMULATION

We model the network as a directed graph $G = (\mathcal{N} + \{0\}, \mathcal{E})$ with $\mathcal{N} = \{1, \cdots, N\}$. Vertex 0 denotes the base station and the vertices in $\mathcal{N}$ denote MUs. Each MU is a receiver of the streaming. The edge set $\mathcal{E}$ models the communication links. A directional edge that originates from vertex 0 is a cellular
Lastly, Eq. (7) ensures the rates of the conceptual flows over incoming flow rate is equal to the total outgoing flow rate.

\[ \min_{\{t_{ij}, f_{ij}\}} P_{\text{max}} \]

subject to

(a) MU power consumption:
\[ p_i = \alpha_d c_{t_{0i}} + \alpha_u c_{t_{0i}} \sum_{j \in N} t_{ij} + \alpha_d c_{t_{0i}} \sum_{j \in N} t_{ji} + \beta \leq P_{\text{max}}, \quad \forall i \in N \]

(b) P2P link capacity constraints:
\[ \sum_{j \in N} t_{ij} \leq W^O, \quad \sum_{j \in N} t_{ji} \leq W^I, \quad \forall i \in N \]

(c) Conceptual flow rate constraints:
\[ t_{ij} \geq f_{ij}^k, \quad \forall i, j \in E, \forall k \in N \]

(d) Conceptual flow conservation constraints:
\[ \sum_{j \in N + \{0\}} f_{ij}^k - \sum_{j \in N + \{0\}} f_{ji}^k = \begin{cases} S & \text{if } i = 0 \\ -S & \text{if } i = k \\ 0 & \text{otherwise} \end{cases}, \quad \forall i \in N + \{0\}, \forall k \in N \]

(e) Variable range constraints:
\[ t_{ij} \geq 0, f_{ij}^k \geq 0, \quad \forall i, j \in E, \forall k \in N \]

downlink while the other edges are the local wireless links between MUs. For each communication link \( <i,j> \), variable \( t_{ij} \) is introduced to denote the rate of the traffic carried by the link. Also, \( f_{ij}^k \) is used to denote the rate of the conceptual flow from the base station to the \( k \)th MU over link \( <i,j> \).

The optimization problem is formulated as a linear programming problem, shown in Eqs. (2)-(7). The constraints in Eq. (3) specify the power consumption of each MU in accordance with the streaming data rates over the cellular link and the P2P links, and \( P_{\text{max}} \) denotes the maximum power consumption of all MUs. The constraints limiting the data rates to be carried by the local P2P networks are specified in Eq. (4). Specifically, for each MU, the aggregated incoming data rate of the P2P links must be less than \( W^I \), and the total outgoing data rates must be less than \( W^O \). Eq. (5) ensures that the actual data rate of link \( <i,j> \) is the maximum rate of the conceptual flows that use link \( <i,j> \). Eq. (6) is the flow conservation constraint, \( i.e. \), for the \( k \)th conceptual flow, the total outgoing flow rate of vertex 0 (the base station) is \( S \) and the total incoming flow rate of vertex \( k \) is \( S \), while, for the other vertices, the total incoming flow rate is equal to the total outgoing flow rate. Lastly, Eq. (7) ensures the rates of the conceptual flows over link \( <i,j> \) and the physical data rate of link \( <i,j> \) are always non-negative.

Our objective is to minimize the maximum power consumption of MUs (\( P_{\text{max}} \)) subject to the constraints (3)-(7). We choose to minimize \( P_{\text{max}} \) instead of the total power consumption of all MUs because minimizing the total power consumption does not ensure fairness over the network, \( i.e. \), some MUs may have low power consumption while other MUs may consume power at a very high level.

V. NUMERICAL RESULTS

In this section, we present numerical results to demonstrate that P2P cooperation significantly improves the delivery of video streaming to MUs. We generate different network topologies by randomly placing a different number of MUs onto a terrain of size 200m \( \times \) 200m. We assume the local P2P network uses a fixed WiFi communication range of 100 m so that, if two MUs are within the range of each other, they can directly communicate. We only evaluate the network topologies that are connected by WiFi, since each connected component of a partitioned network is considered as a smaller network and evaluated separately. In addition to a WiFi interface, each MU also has a cellular interface to communicate with the BS. In this study, we assume that the quality of a cellular connection does not vary with the location of the MUs and the capacity of each cellular connection is higher than the encoding rate of the streaming traffic.

For each generated network topology, we evaluate the scheme in which MUs cooperate to minimize the maximum per-MU power consumption (P2P:MIN-MAX) and the scheme where there is no cooperation between MUs (NO-P2P). For the first scheme, we set up a linear programming problem as presented in Eqs. (2)-(7) and then use the GNU Linear Programming Kit (GLPK) [23] to compute the optimal solution. We also compare our schemes with the LocalTree algorithm [19]. Table I summarizes the important parameters used in the numerical study. The values of the MU’s power efficiencies in different communication links are taken from the measurement study in [7]. Results shown in Figs. 4-7 are the average of 30 random trials.

Fig. 4 shows that, cooperation among the co-located MUs can greatly reduce the maximum MU power consumption
Power efficiency in cellular downlink \( \alpha_C \) 40 mW/Mbps
Power efficiency of WiFi transmission \( \alpha_W \) 11 mW/Mbps
Power efficiency of WiFi reception \( \alpha_W \) 5 mW/Mbps
Base power consumption \( \beta \) 100 mW
WiFi outgoing link capacity \( W_O \) variable
WiFi incoming link capacity \( W_I \) 10 Mbps
Streaming rate \( S \) 10 Mbps

| TABLE I | SIMULATION PARAMETERS. |

can be greatly reduced in comparison to both NO-P2P and LocalTree. Even when there are only two MUs cooperating, a MU’s power consumption is saved by about 31% compared to NO-P2P. This is due to the fact that, with cooperation, MUs share streaming content using the more power efficient WiFi networks to reduce the traffic transferred over the cellular connections. Also, notice that the maximum MU power consumption decreases when there are more MUs in the network, which implies that each MU benefits from more cooperation with more neighbors. Although LocalTree also allows cooperation among the MUs, the LocalTree algorithm causes the highest maximum MU power consumption. The reason is that LocalTree only chooses one MU as a puller to download all video data packets from the BS and then disseminate the packets to other MUs. Such a high power consumption may make an MU unwilling to cooperate with other MUs due to the risk of becoming the puller.

The LocalTree algorithm outperforms our scheme in terms of the total network power consumption as shown in Fig. 5. This is mainly due to the fact that LocalTree allows an MU to broadcast a data packet to multiple neighboring MUs via a single transmission, which reduces total number of data transmissions. We argue that, however, using wireless broadcast may not be practical in realistic scenarios because most current P2P applications use application-level multicast.

Fig. 6 illustrates how the outgoing capacity of the WiFi links affects the MU’s power consumption. Given that the cellular links are less power efficient, an MU would prefer to cooperate with neighboring MUs to exchange streaming content over WiFi links. However, such cooperation is limited by the capacity of WiFi networks. In other words, when the WiFi outgoing link capacity is small, only limited amount of streaming content can be exchanged locally and each MU has to rely on the cellular network to retrieve additional streaming segments, leading to high cellular network usage (Fig. 7) and high power consumption (Fig. 6). On the other hand, the power consumption decreases with an increase in the WiFi capacity, as more content can be exchanged locally. The maximum power consumption reaches its minimum when the WiFi outgoing capacity is around 12 Mbps. At this point, the WiFi capacity is no longer the constraining factor. How much data to be downloaded from the BS is determined by the streaming rate \( S \) (Fig. 7) since a group of MUs only needs to download one copy of the streaming content over the cellular network and can exchange the content locally, resulting in minimum power consumption per MU.

VI. CONCLUSION AND FUTURE WORK

In this paper, we studied the scenario where a group of nearby mobile devices share video streaming content, orig-
in a sports arena using their smartphones to watch instant video. For instance, there could be hundreds of co-located users in a sports arena using their smartphones to watch instant replays. We plan to develop customized iterative algorithms so that large problem instances can be solved efficiently. When developing such iterative algorithms, we will also consider the dynamics of the network, i.e., mobile users joining/leaving the networks (known as peer churn). The iterative algorithms should keep the networks stable and only invoke a minor reconfiguration of the networks when peer churn happens.

Moreover, when formulating the problem, we made the assumption that the original streaming can be divided into sub-streams of arbitrary data rates. In reality, the data streaming can only be divided into a limited number of sub-streams of equal data rate (as in multiple description coding [24]). Considering such an issue may possibly introduce integer constraints into the optimization problem. We will investigate the problem to see whether the problem becomes NP-complete and design efficient heuristic solutions.

REFERENCES