WiGroup: A Lightweight Cellular-assisted Device-to-Device Network Formation Framework

Yufeng Wang†, Hongxu Zhang†, Chiu C Tan†, Xiaojiang Du†, Bo Sheng‡
† Department of Computer & Information Sciences, Temple University
‡ Department of Computer Science, University of Massachusetts Boston
Email: {Y.F.Wang, hongxu.zhang, cctan, dux}@temple.edu
shengbo@cs.umb.edu

***Abstract*—Cellular networks will periodically experience instances where the network capacity is insufficient to meet network demand. This paper proposes *WiGroup*, a group formation algorithm to help cellular networks organize mobile devices into groups to reduce cellular network overhead while limiting disruption to the end user. WiGroup is based on extending device-to-device (D2D) communications, a standard current supported by the cellular network operators, to create device-to-group (D2G) communications. Experimental results indicate that our framework is able to reduce up to 37% of the direct cellular connections. Also, trace-driven results show its ability of consistently reducing direct cellular connections by as high as 28%.**

I. Introduction

Smartphones today are no longer merely used to make phone calls, but also function as a content consumption device for watching videos, browsing the web, gaming, vlogging, and so on. As one of the key networks responsible for transporting a large component of this traffic, cellular networks are struggling to keep up with the data demands from these ever ubiquitous mobile phones [1], [2].

In response to this problem, there has been increased interest in solutions to facilitate mobile devices to communicate directly with each other to reduce the workload on the cellular network. One promising direction is *device-to-device*, or D2D, communications. D2D is a technology to simplify direct mobile-to-mobile communications [3]; rather than relying on the cellular infrastructure to route data from one mobile phone to another, a cellular base station will facilitate the creation of a direct link between two mobile devices to allow them to communicate with each other, bypassing the cellular infrastructure. This D2D connection can be performed over licensed cellular spectrum [4], or via an unlicensed spectrum like WiFi-Direct [5]. From the cellular providers' point of view, D2D can improve spectrum utilization and user satisfaction [6], as well as make the cellular infrastructure more resilient by way of enabling communications in disaster scenarios where cellular towers are overloaded or damaged. In fact, 3GPP is planning to adopt D2D for LTE release 12 for use by first responders in public safety networks [7]. From the end users' viewpoint, D2D can improve energy efficiency if communications are shifted from cellular networks to WiFi, primarily due to the lower energy consumption of WiFi radio hardware. More importantly, the entire D2D process is transparent to the end user.

In this paper, we expand the idea of D2D from just two devices to multiple devices. In other words, instead of connecting two phones together, the cellular provider will create small groups of phones that can communicate with each other without relying on cellular infrastructure. This larger D2D-network is suitable for applications like opportunistic connection sharing [8], [9], [10], [11], where mobile devices share the cellular connectivity with other nearby devices so as to reduce the number of direct connections with the cellular tower, and take advantage of caching opportunities. We term this larger D2D network as D2G (device-to-group) network.

In this paper, we propose *WiGroup*, a practical framework of create these larger D2D networks. The key features of WiGroup include (1) a lightweight group formation algorithm that allows a cellular provider to regulate the number of groups to form without collecting additional information from mobile devices; (2) an incentive mechanism to encourage mobile devices to participate. Our simulation results indicate the WiGroup framework can reduce the load on cellular networks without sacrificing user QoS. We also tested the overhead of WiGroup on an operational WiMAX tower to determine the feasibility of WiGroup on 4G eNodeBs.

II. Related work

Direct inter-mobile device communication, like mobile ad hoc networks (MANETs) [12] or delay-tolerant networks (DTN) shares some of the characteristics with D2G networks. However, there are two main differences. First, D2G networks are typically a single hop network, which means that data routing is relatively simple. Second, D2G networks can rely on the underlying infrastructure network to perform certain operations such as device discovery, connection management, synchronization, handoff, and so on. However, D2G networks are designed mainly to reduce the number of cellular connections so that the network congestion and failures will be alleviated [8].

Network traffic is considered to have a large amount of redundancy, even among different users when they get access to similar content [13]. Numerous mechanisms, which are also known as deduplication techniques, have focused on eliminating such network redundancy. Deduplication is different to classic data compression technique such as GZIP [14], since it detects duplicate data across objects other than within objects. [15] proposed a mechanism to locate identical and similar sources for data objects using a constant number of lookups and inserts a constant number of mappings per object, so that redundancy during the file downloading can be efficiently eliminated. A more fine-grained deduplication technique is network deduplication [16], [13], which is a protocol independent approach for identifying redundant bytes in network traffic. [13] reports their mechanism can deliver average bandwidth savings of 15-60% for enterprise and university access links as well as the links connecting busy web servers.

As the mobile traffic grows explosively [17], redundancy elimination on mobile devices has attracted a large amount of attention [18], [19], [20]. Implementation of redundancy elimination techniques used in wired network is not feasible for mobile devices due to the power, speed, memory and storage limitations. [18] proposes asymmetric caching, which allows mobile devices to selectively feedback appropriate portions of its cache to the traffic source with the intent of improving the redundancy elimination efficiency. [19] provides a redundancy elimination system that uses both object and chunk based deduplication. In [20], it proposes that the server directly identify duplicate traffic for mobile users. However, above work all consider the deduplication based on single user. More importantly, they either require service provider or an extra middle-box to participant into the redundancy elimination.

III. Motivation and Background

The goal of WiGroup is to create D2G networks to help reduce the overhead on cellular network during periods where demand for cellular resources outstrips supply. For this is occur, all involved parties need to obtain some benefit from participation. The involved parties are: the group member (GM), the cellular base station (BS), and the group owner (GO).

The GM benefits from D2G in several ways. First, cellular providers generally place a data cap on users (e.g. Verizon, a major US cellular provider, has a cap of 5 GB per month). A GM participating in a D2G will avoid incurring cellular data usage, since it is using WiFi to communicate with the GO [21]. Other benefits include improved power savings, WiFi radio draws less power than cellular [22], [23], and improved response time via caching [24], [25].

The BS benefits from D2G by being able to satisfy more users within each geographic cell. The BS needs to allocate resources to every user. When there are too many users in a cell, the BS cannot service additional users, since available resources have been allocated. D2G reduces the number of devices directly connected to the BS, thus reducing the connection failures and timeouts [8].

The GO acts as a hotspot to the GMs. In this role, the GO not only increases its cellular data utilization, but also increases its energy consumption, since it now has to operate both cellular and WiFi radios, and has less opportunities to sleep to conserve power. There are no benefits to the GO from participating in a D2G.

A. Incentives to Participate

It is clear that a D2G network does not benefit everyone all the time, necessitating the need to design incentives together with the group formation algorithm. The advantage of a D2G network, as opposed to a conventional ad hoc network, is that we can take advantage of the cellular infrastructure when designing any incentive algorithms. Since all traffic flows through the BS, the BS is able to collect information to make better decisions, such as how many GOs should there be, which mobile users should act as GOs, which as GMs, and so on. The accounting for incentives is also easier, since the cellular provider already performs the necessary billing functions. There is also more trust in the system, since all users trust the BS.

To keep our solution practical, we adopt the following guidelines in designing our algorithm.

1) The BS will only provide incentives to encourage the formation of D2G networks when it has insufficient capacity to handle that many direct connections. This implies that that the BS will regulate the number of D2G networks being formed, since the BS does not want to pay for incentives when it has sufficient capacity.
2) Each mobile device is independent. The BS cannot compel a device to become a GO or GM, nor control any aspects of a device's behavior (e.g. mobility, network access patters, etc.). The BS can only inform a device that it has been selected to be a GO or a GM, and the device is free to act as it wishes. There is no feedback communication from the GM/GO back to the BS.
3) We restrict the information we can use to what an actual 4G BS can collect under normal operations. This means we can assume that the BS can learn information like the MAC address, destination IP address, signal strength, and so on, but not information like power levels, GPS coordinates, overhead MAC address, etc, since these are not part of common network operations

Finally, we assume that all parties, the BS, GO, and GM, are honest, and will not attempt to cheat each other. Thus, cheating actions like a GO that never forwards traffic for GMs are out-of-scope of this paper. We assume that all mobile devices participating in a D2G is able to support operations such as setting up a soft AP, issuing IP address, and so on, that are necessary for tethering support. This can be done by using standards like WiFi-Direct, which are available on Android phones, or Multipeer on Apple's iOS phones.

B. Integration with cellular infrastructure

We envision WiGroup to be executed by eNodeBs in the cellular network. A basic 4G (LTE/WiMAX) network consists of two main components: a *radio access network* (RAN) and a *core network* (CN) [26]. The RAN consists of base stations (eNodeBs) that communicate directly with the end users' mobile devices. The CN consists of three main entities: the *packet delivery network gateway* (P-GW) that is responsible for allocating IP addresses and maintaining QoS, the *serving gateway* (S-GW) that performs end user accounting (data usage, minutes, etc.) and anchor for voice data, and the *mobility management entity* (MME) responsible for mobility and bearer management for the end user. The P-GW and S-GW are known as CSN-Gateway and ASN-Gateway in WiMAX. 4G networks use the concept of *bearers* to regulate end user QoS. Latency-sensitive applications like VOIP are assigned guaranteed bit rate (GBR) bearers, while less sensitive applications like web browsing are not, meaning there is no fixed bandwidth resources allocated to it.

D2G takes advantage of recent advances in systems powering 4G networks such as intelligent base stations [27], [28], where base stations have computational resources and are programmable, and C-RAN [29], [30], [31], [32], where conventional base station processing functions, *basebase units* (BBUs), are migrated away from the RAN to a backend data center. Our vision of D2G is to take advantage of the additional computing capability of the base station to improve the performance and robustness of cellular networks.

A basic D2G network consists of multiple mobile devices organized into a group, a basic scenario is shown in Fig. 1. All members in the group, denoted as **GM**s, group members, are located within the same cell region, meaning they are all communicating with the same cellular tower base station (**BS**). Each group has a single group owner (**GO**), which is also a mobile device. All GMs are connected to the GO, and access the BS (and wider Internet) through the GO. All communications within the group also go through the GO. The role of the GO, in essence, is similar to that of an WiFi infrastructure access point (AP), performing operations like assigning IP address and so on. A D2G will also be able to perform actions like caching, collaborative downloads, and so on, if they are browsing or watching the same web or videos, to even reduce cellular traffic. A BS can accomodate multiple D2G networks. Our WiGroup algorithm is used to facilitate the formation of D2G networks by identifying suitable mobile devices to serve as GOs and members of groups.

IV. WiGroup Protocol

Our WiGroup protocol helps a BS to organize mobile devices within its cell into one or more D2G networks. The goal is to reduce the number of devices directly connected to the BS by aggregating them into D2G networks so as to reduce the workload on the BS. We organize time into *epochs*, where in each epoch, the BS will run the WiGroup protocol to provide incentives to form the desired number of D2G networks. The incentives take the form of *tokens*, which will expire at the end of each epoch. The essential WiGroup protocol has the following three phases given below.

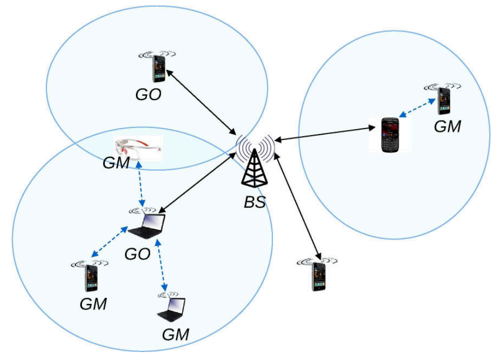


Fig. 1. D2G networks in which heterogeneous mobile devices reside.

O_t	All the GOs at epoch t
M_t	All the GMs at epoch t
I_t	All the non-GO/GM devices at epoch t
Z_t	GM candidates at epoch t
v_t	Token value at epoch t
b_i^t	Bandwidth usage of device i at epoch t
B_i	Bandwidth capacity of device i
c_i	Time duration of device i's connection to the BS
c_T	Time duration threshold for GO candidates
s_i	Received signal strength
θ	Monetary loss for each over-threshold direct connection
τ	Targeted number of direct connections to BS

TABLE I
NOTATIONS

Phase 1: At the start of epoch t, the BS uses the network conditions (number of direct connections, failed connections, etc.) collected in epoch $t-1$ to determine whether and how to group devices. If the BS is able to reach its objective (IV-A), the BS will stop and waits for the next epoch to occur. Otherwise, the BS will proceed to the next phase.

Phase 2: The BS will first estimate the number of GOs and GMs necessary to achieve a reduction in the workload (Subsection IV-B). Then, the BS will determine which of the devices should act as GOs, and which as GMs, based on the connection quality and prior history of each device (Subsection IV-C).

Phase 3: The BS will inform the devices selected as GO candidates to start offering tethering service. The BS will also issue tokens to the selected GM candidates. When the GM candidates decide to connect to a particular GO, they will "pay" the GO using the tokens, which are valid for the duration of that epoch.The GO will return the tokens back to the cellular provider later to cash in on the incentives. The BS will adjust the parameters at the beginning of each epoch based on performance in the previous.

A. Overview

The goal of BS is to keep the direct connections below a threshold (τ) so that the overall network quality is well maintained, while it pays as little compensation as possible to GOs. If there are more than τ direct connections to the BS, the quality-of-service of the whole network will degrade. We assume that every extra connection beyond τ will bring monetary loss of θ, which could be caused by the leave of users because of poor network quality. Therefore, we formulate the target of BS as:

$$Min(\theta \cdot (|O_t| + |I_t| - \tau) + v_t \cdot \sum_{i \in M_t} b_i) \tag{1}$$

In which two parts of cost for BS are involved: Punishment Cost ($\theta \cdot (|O_t| + |I_t| - \tau)$), which represents the extra cost to BS because of extra direct connections. And Compensation Cost ($v_t \cdot \sum_{i \in M_t} b_i$)), which represents the cost that BS need to pay for aggregating devices into groups, so that the overall direct connections could be reduced.

B. Striking the right balance of GO/GMs

It is important to be aware of the information that a BS knows about the devices within its cell, and what it does not. A BS is aware of the destination of the network connections, length of connection, bandwidth used (b), and bandwidth capacity (B), for all the connected devices. Bandwidth used refers to the amount of data transmitted to/from the device per unit time, where as the capacity refers to the theoretical maximum amount of data a device could have uploaded/downloaded per unit time given its channel quality to the BS. The BS is unaware of the actual location of any of the devices. Thus, the BS cannot answer questions like, where the GOs are located, how many GMs are near a GO, and so on. This is important, since the cell coverage could be a couple of miles, encompassing hundreds of devices.

We would like to avoid a mismatch between the balance of GOs/GMs. There are four possible mismatches (1) undersupply of GOs with oversupply of GMs; (2) undersupply of GOs and GMs; (3) oversupply of GOs paired with undersupply of GMs; (4) oversupply of both GOs and GMs. Case 1 has too many GMs attempting to connect to a small pool of GOs, which will cause GOs to start rejecting the GMs since a device can only comfortably support a limited number of users. Cases (2) and (3) are both not ideal, since neither helps the BS reach its objective (Eq. 1). Case (4) should be avoided. It can actually increase the number of direct connections, since devices could be splintered into more D2G networks than necessary, bringing more cost to BS.

Number of GO candidates. The proper number of GOs are supposed to provide enough bandwidth resources for GMs, helping BS reach its objective. Note that the BS cannot accurately predict the usage of each devices in the network. The BS estimates the overall bandwidth requirement, TR_t, of all devices in its cell, by using historical information in epoch $t-1$. The BS then picks k devices which utilizes only less than u ($u \in (0, 1)$) of their bandwidth capacities, where k satisfies

$$\sum_{j \in O_t} B_j \geq TR_t, \quad TR_t \leftarrow \sum_{i \in O_{t-1}+I_{t-1}} b_i^{t-1} \tag{2}$$

Number of GM candidates. The number of GMs has significant affect on the objective of the BS as we discussed. Unfortunately, the BS does not have pre-knowledge of whether a GM candidate will eventually join a GO during the epoch t. Therefore even the BS could know the number of GM candidates needed to reach its objective at t, it could not tell who would certainly join Wigroups if notified. Thus, the BS needs to tune the number of GMs candidates to be notified based on performance in epoch $t-1$ (Section IV-C1), in order to reach its objective.

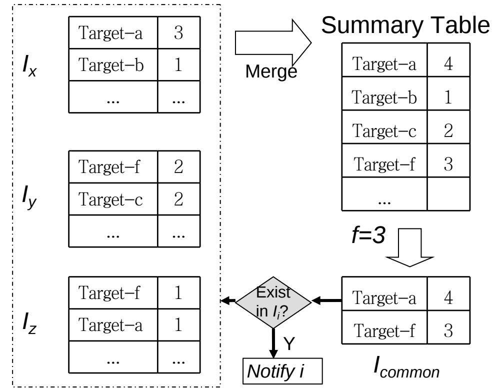


Fig. 2. BS merges information from $\boldsymbol{I}_x$, $\boldsymbol{I}_y$ and $\boldsymbol{I}_z$ into a single Summary Table (ST). In ST, only records with counter value no less than 3 are kept, labeled as common interests among these three devices.

C. WiGroup algorithm

Besides determining the number of GOs/GMs, we need to determine *which* devices should serve in what role. An ideal GO i should have (1) a relatively longer-term connection to the BS; (2) good network connection to the BS (good received signal strength indicator, RSSI); and (3) sufficient bandwidth capacity to support enough GMs.

The BS cannot control (1) and (2), but can try to improve (3) by selecting devices that share the common interests to form D2G networks. So that the bandwidth could be saved on GO by eliminating redundant traffic, also improving the response time for GMs. This is achieved by exploring the browsing history $\boldsymbol{I}_1, \boldsymbol{I}_2 ... \boldsymbol{I}_n$ of devices. The BS records the data from each device in tables shown in Fig. 2. We consider an target as part of common interests $\boldsymbol{I}_{common}$ if is has been visited by at least f times. Then the comparison between $\boldsymbol{I}_{common}$ and individual browsing history determines whether a device is selected as a GM candidate.

BS merges $\boldsymbol{I}$ of all connected devices into a single Summary Table (ST). In ST, only if a counter value is no less than f, the correspondent target is labeled as part of common interests and kept. Otherwise, it will be removed from ST. Eventually the records left in ST is $\boldsymbol{I}_{common}$, which will be used to be compared with $\boldsymbol{I}$ (where $i = 1...n-k$). If $\boldsymbol{I}_i$ contains $\boldsymbol{I}_{common}$, BS considers device i as a potential GM.

Details of our algorithm for BS are shown in Alg. 1.

At epoch t, BS firstly updates the connection information (O_t and I_t) according to the current connection conditions. Then estimates total bandwidth requirement TR_t by summing up the total traffic during epoch $t-1$ and deducing the amount of bandwidth that current GOs can provide. Then it selects more GO candidates, which must satisfy (1) has a connection

Algorithm 1: BS Operation

Update O_t by removing any $i \in O_{t-1}$ but decides to quit;
Update I_t by adding new coming devices/quited GOs and removing departured devices;
$TR_t \leftarrow \sum_{i \in O_{t-1}+I_{t-1}} b_i^{t-1}$;
for *Device* $i \in O_t$ **do**
    $TR_t = TR_t - \sum_{i \in O_t} B_i$
end
Sort j ($j \in I_t$ & $j \notin O_t$) descendingly based on s_j ;
$k = |O_t|$;
for *Device* $j \in I_t$ *and* $j \notin O_t$ *with* $c_j > c_T$ **do**
    if $TR_t > 0$ **then**
        if $b_j^{t-1} < u \cdot B_j$ **then**
            $O_t = O_t \oplus j$;
            $I_t = I_t \ominus j$;
            $TR_t = TR_t - B_j$;
            $k = k + 1$;
        end
    end
    else
        Break;
    end
end
$\boldsymbol{I}_{common} \leftarrow$ Common interests extracted from $\boldsymbol{I}$;
$Z_t = \emptyset$;
for *Devices* $i \in I_t$ **do**
    if $\boldsymbol{I}_{common} \in \boldsymbol{I}_i$ **then**
        $Z_t = Z_t \oplus i$;
    end
end
Notifies $\omega \cdot |Z_t|$ devices to join D2G by giving tokens valued v_t;

duration longer than a predetermined threshold c_T, (2) has strongest received signal strength s_i, (3) bandwidth utilization is less than u. Then these chosen GO candidates will be notified.

1) Tunning the number of GM candidates: After settling all GOs, the BS starts to pick GM candidates and updates Z_t. Consider the fact that chosen GM candidates are not guaranteed to join the Wigroup, BS is facing uncertainties which make it hardly to make an optimal decision on the number of GM candidates. In this case, a tunning parameter ω is introduced to adjust the number of GM candidates in each epoch. According to Eq. 1, too few GMs will result a higher value of $|I_t|$ and raise the Punishment Cost. If few of the selected GM candidates in epoch $t-1$ join Wigroup, BS will raise ω at epoch t. While there are too many GMs, they could bring higher Compensation Cost. Then BS will decrease ω. It could happen that increasing number of GMs do not bring correspondent traffic increase, which will result ω unchanged.

2) Tunning the Token price: As the incentive for GOs to participate, the value of tokens v_t significantly affects the objective of BS. If GOs receive insufficient compensation, they choose to quit, resulting too few GOs in the Wigroup. Consider that in epoch $t-1$, a device i generates b_i amount of traffic, needs to pay $p \cdot b_i$ to the BS. Let us denote this as $Cost_A$. If i becomes a GO and has M_i GMs connected, it will need to pay for both its own traffic, as well as that from all M_i GMs. At the meantime, it will receive tokens from connected GMs, based on their traffic requests. We consider that i could eliminate redundant traffic among GMs. We use $U()$ to represent the actual amount of data transmitted between i and the BS after redundancy elimination. So that the current cost is denoted as $Cost_B$. So we have

$$\begin{aligned} Cost_A &= p \cdot b_i & (3) \\ Cost_B &= p \cdot U(\sum_{j \in M_i} b_j + b_i) - v_t \cdot \sum_{j \in M_i} b_j + \epsilon & (4) \end{aligned}$$

Where ϵ represents miscellaneous cost (Extra power drain, etc). We see that while i needs to pay more to BS for transmitting more data, it gains tokens from GMs connected. So for i, the cost of acting as a GO needs to be less than the cost of not being a GO, which is $Cost_B < Cost_A$:

$$v_t \cdot \sum_{j \in M_i} b_j > p \cdot U(\sum_{j \in M_i} b_j + b_i) - p \cdot b_i + \epsilon \tag{5}$$

We use $R()$ to represent the redundant amount of data, which can be eliminated on i. Then we have $b_i = U(b_i) + R(b_i)$. formula above can be written as:

$$\begin{aligned} & v_t \cdot \sum_{j \in M_i} b_j > \\ & p \cdot (\sum_{j \in M_i} b_j + b_i - R(\sum_{j \in M_i} b_j + b_i)) - p \cdot b_i + \epsilon \end{aligned} \tag{6}$$

$$v_t > p - p \cdot \frac{R(\sum_{j \in M_i} b_j + b_i)}{\sum_{j \in M_i} b_j} + \frac{\epsilon}{\sum_{j \in M_i} b_j} \tag{7}$$

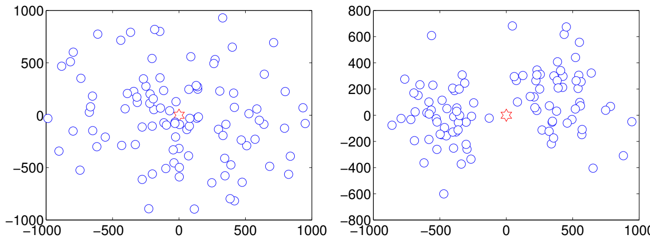
We can tell that the token value is related to the amount of shared data among GMs and their connected GO, this also motivates the BS to pick GM candidates that share common interest. Eq. 7 indicates the necessity for a device to continue acting as a GO. Otherwise, the existing GO will quit in epoch t. In this case, BS will consider raise v_t in order to motivate enough number of GOs. Another alternative is to identify devices that truly share common traffic as well as in the vicinity, and let them join the same GO. This way might be more efficient for BS to reach its objective. However, it requires information such as actual locations of devices which is unavailable on BS as we discussed(Section IV-B).

V. Evaluation

We use a combination of simulations and 4G testbed to validate the WiGroup systems design.

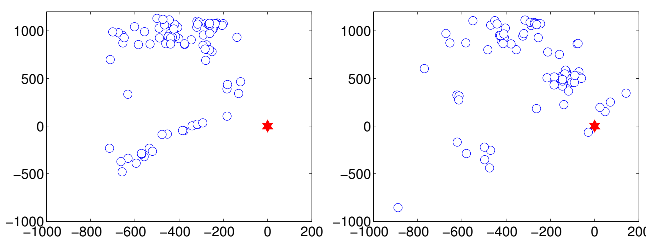
A. Simulation Setup

We run simulations on both synthetically static model and mobility trace to determine the performance of WiGroup. For the synthetic model, we consider the coverage radius of BS to be 1,000 m and the Wifi communication range as 100 m. Such



(a) Random sceniario (b) Crowded-sparse sceniario

Fig. 3. Sample topologies with 100 devices reside of 2 scenarios



(a) Initial (b) Terminated

Fig. 4. Initial and terminated (90 min) locations of users. Red hexagram indicates the monitor.

configurations are conservative compared with performance tests in realistic environment (∼3 km to 5km [33], and up to 200m [34], respectively). We assume that the communication qualities are stable enough within this radius so that each node is guaranteed a base channel capacity which is normalized as one. Also consider the heterogeneity of devices, the maximum channel capacity B varies by adding a random number which is smaller than one to the value of base channel capacity. Data rate is also generated randomly but within each device's maximum channel capacity. The mobility trace is collected in KAIST [35]. It consists of GPS locations of 92 users over time. We tested the trace for duration of 90 minutes, which is long enough for mobile device group connection.

As a light-weighted adaptive solution, our system will work without being limited by the geo-locations. We consider the following two geo-location scenarios in the synthetic model. (1) **Random**: Nodes are randomly distributed within the BS coverage area. (2) **Crowded-Sparse**: There can be locations inside the BS cell where nodes are closely and densely located. This could be resulted from crowed events, such as political protests, public demonstrations and even community parties. Fig. 3 shows an example geo-locations of devices in our experiments.

The mobility trace shows user distribution more similar to the Crowded-Sparse distribution. The initial and final locations of these users are shown in Fig. 4. The geo-locations are collected from the reference point at (0,0). There are 1 and 8 users not shown since they are far from the others in the initial and terminated time, respectively.

RSSI of each device is generated based on the free-space transmission loss prediction model [36], which indicates that the signal strength is negatively related to the square of distance from the device to the BS. This model though is not applicable to all scenarios in reality, details of how various models are applied are out of our scope. We generate RSSI for all the nodes based on their distances to the BS, and multiply it by a random number in $[0, 1]$ to adapt possible variations in the wireless environment.

B. Simulation Results

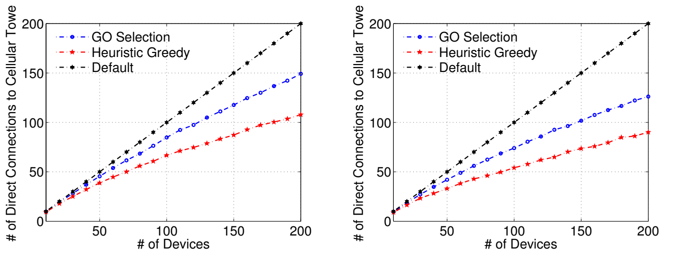
1) Performance improvement on BS: The goal of WiGroup is to reduce number of direct connections on BS below a required threshold at minimum cost. Implicitly we know that increasing number of GOs will provide more opportunities for other devices to aggregate their network traffic. However, k cannot also be too large. If we consider two extreme cases: BS selects no GOs versus selecting every device as a GO. In both cases, BS will not gain any benefits because there will be no direct connection reduction.

Ideally BS can select a proper set of GOs that are within the communication range of selected GMs. To analyze it more generally, we can assume that all devices within this network share enough common interest so that they can join D2G networks as long as they are in the communication range of a GO. Then the Heuristic approach [37] provides a potential greedy solution to such a problem by dividing them into k clusters in which the maximum inter-cluster distance is minimized. If we assume that each GO has good enough network quality and capacity, then this solution will help aggregate as many as possible potential GMs because it tries to make each device stay close enough to a GO.

We use the number of direct connections to the BS as the metric to evaluate effectiveness performance of GO selection. We compare our GO selection algorithm with that from Heuristic. Note that in Heuristic solution, it is required to give the exact number of GOs. To make a fair comparison, we explore it by manually tuning k from 1 to n and choose the one that can claim the best performance.

Here we consider all devices are qualified to join Wigroup if they are in vicinity of any GOs. As shown in Fig. 5(a), the performance based on our GO selection criteria claims 3% to 25% of direct connection reduction in random distribution case. The heuristic approach has better performance by reducing 4 - 21 % more direct connections. In the other case, the reduction varies from 5% to 37%, which is higher than that in the random case and also only at most 18% less than the heuristic approach, as shown in Fig. 5(b). This is because devices are in closer vicinity in the crowded environment, which makes it easier to aggregate and reduce the performance differences between the two solutions.

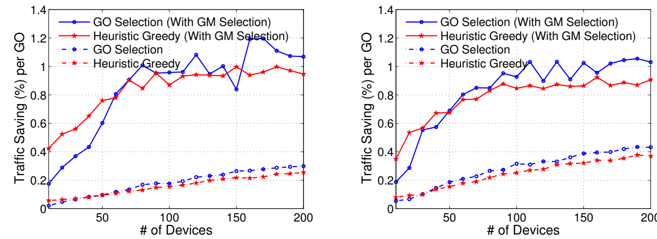
Though the heuristic solution could claim better performance as a way to evaluate our performance in above simulation, it is not feasible for implementation for the following reasons. First, there is no pre-knowledge on the number of GOs that can help claim its best performance. Note that though we can iterate all possible number of GOs for a simulation purpose, it is not practical in real systems since it will bring a nontrivial delay to the network. Second, as a location-



(a) Random

(b) Crowded-Sparse

Fig. 5. Performance Comparison between our algorithm and the Heuristic solution. The results are all averaged values of 50 independent executions.



(a) Random

(b) Crowded-Sparse

Fig. 8. Performance Comparison between our algorithm and the Heuristic solution. The results are all averaged values of 50 independent executions.

based solution, it must obtain the location information of every device. This becomes also unfeasible since it is hard to accurately fetch such information. Besides, movement of devices in wireless networks will bring heavy communication burden in the network for information update.

We then expand the test to the mobility trace by applying various settings. Fig. 6 shows the performance when we limit the maximum number of devices each GO can support in the group (denoted as ’capacity’). It can tell that when the capacity increases from 2 to 4, the overall performance becomes better by reducing direct connections from 9% to 13% in average. However, we note that when the capacity increases from 4 to 6, no further improvement can be achieved. This is because when the capacity of GO increases, the BS tends to selects less GO, covering smaller geo-space.

In addition, the performances have also been tested when the communication range varies. Since different D2D communications can be applied and result in different communication capabilities, which is represented as communication range here. Fig. 7 shows the performance under different D2D communication ranges (from 5m, 10m and 30m). The direct connections has been reduced by from 5% to 28% when communication range becomes longer.

Based on above results on the mobility trace, we can also tell that our algorithm can well adapt the movement of users by maintaining the direct connections at a stable number.

2) Traffic Reduction within groups: As discussed in section IV-C2, GO needs enough incentives to participate D2G networks. The amount of duplicate traffic on each GO also affects whether they are willing to serve for BS. We then conduct extended simulation on the static model by assuming that each device shares random percentage of duplicate data among their traffic. Therefore we propose that BS selects devices share enough amount of common interest with others to join in section IV-C. We conduct simulations on synthetic model. Fig. 8(a) and Fig. 8(b) shows the per GO savings of our algorithm and the heuristic solution under the circumstances that BS randomly chooses GMs versus prioritizes devices that share higher percentage of traffic over the others. We can tell that with GM selection, GOs are able to detect almost 3 times of more duplicate data, providing much stronger incentives for GOs.

Also it is shown that though the heuristic solution claims more GM participation in total, there is less duplicate data through each GO when there are more than 60 and 50 nodes in random and crowded-sparse scenarios, respectively. Our solution can provide better incentives for GOs.

C. Prototype Setup and Experimental Results

One key component of WiGroup is that the cellular base station needs to perform additional operations, e.g. tracking IP address, signal strength, and so on. We used an AirSpan base station operating on 2590Mhz frequency under an experimental licence from the FCC. It consists of WiMax antenna, an outdoor and indoor unit, a base station server (BSS). We implemented our WiGroup prototype code in the BSS. We did not implement the signaling mechanism between the base station and mobile devices, since we assume that those operations will be similar to the D2D standard.

We used TCPDump to collect the necessary source and destination IP addresses, and use SNMP to obtain the RSSI values of the connected mobile devices from the base station. The AirSpan base station contains a Click Router module, but we deliberately avoided using Click since it is not commonly found in commercial cellular base station deployments. In our experiment, we used a Lenovo ThinkPad L430 laptop equipped with a AW3 US300 WiMax USB adapter and an Samsung Galaxy S2 as the GO, respectively. For the GM, we use up to four Samsung Galaxy S1 running Android 4.3 OS.

We were interested in using the prototype testbed to examine two components of WiGroup algorithm, the estimation of number of GOs, the use of signal strength to select GOs and the effect of content similarity inside D2G on the network latency.

1) Estimating the Number of GOs: WiGroup algorithm estimates the number of GOs (in Eq. 2) by assuming that the unused bandwidth of a potential GO can be used to support the estimated bandwidth consumption of the other GMs. Only devices which utilize less than u ($u \in (0, 1)$) of their bandwidth capacities can be considered as GOs, so that they have enough spare bandwidth for GMs. In other words, if a single device can download 5 Mbps but it only downloads 1 Mbps, then that device as GO can support two GMs that are downloading at 2 Mbps. This simplifies the estimation, but ignores potential overheads, such as channel contention between GMs, hardware overheads, and so on.

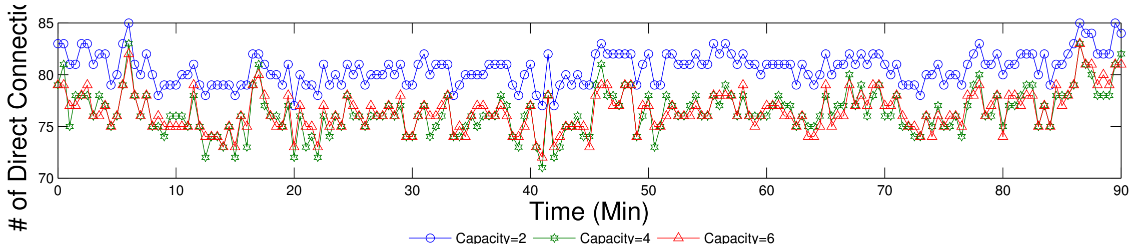


Fig. 6. The number of direct connections to the BS while devices have different maximum connection capacities. Communication range is 10m.

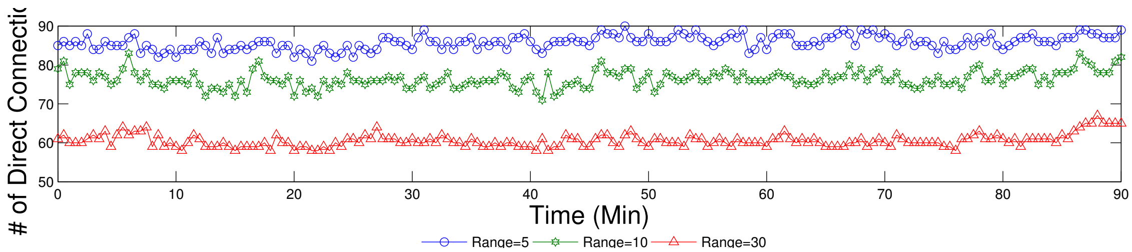
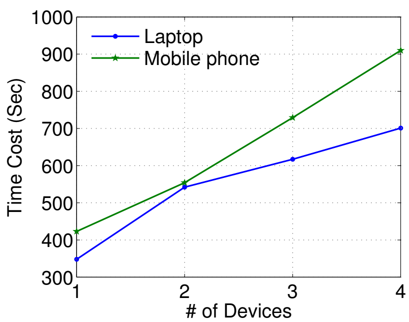
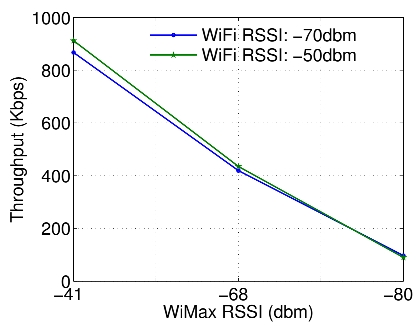


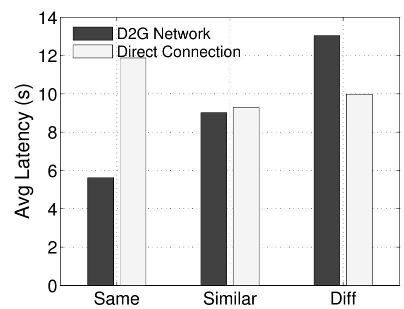
Fig. 7. The number of direct connections to the BS while devices support different communication range



(a) The time cost comparison



(b) Identify the best selection of parameter a.



(c) The average latencies comparison when a device is in the D2G versus direct connection

To better understand these overheads, we created two D2G networks. The first uses a WiMAX enabled Android smartphone as GO, and the second uses a WiMAX enabled laptop as GO. We let up to four regular Android phones act as GMs. Every time the whole D2G group downloads a fixed size of file which is 30MB from a remote web server. The download is triggered by GMs, which evenly fetch part of the file. i.e., the GM will download the whole 30MB if there are no other GMs. Otherwise, each GM will download 15, 10 and 7.5 MB of the file if there are 2, 3 and 4 GMs in the group. We measure the total time consumption from the start of the download to the completion, and plot the results in Fig. 9(a).

We see that, not surprisingly, as the number of GMs increases, the total time it takes also increases. However, there is a noticeable difference when the GO is a laptop, and when it is a smartphone. The results indicate that the resources necessary for the bridging and NAT operations can be significant between different devices, causing noticeable differences of up to about 30% when the number of GMs attached to a GO increases. In practice, this means that the estimation of the number of GMs should be more conservative.

2) Signal Strength in Selecting GOs: In the WiGroup algorithm, we select the GOs based on the signal strength to the cellular tower, and do not consider the signal strength to potential GMs around them. The advantage of our approach is that we avoid the overhead of collecting specific geographic locations for each device.

To explore how the WiMax link quality and the WiFi link quality affect the performance of D2G networks in terms of throughput, we conduct another experiment by varying the distance between GO and the WiMax tower or the location of the GO to get different levels of WiMax RSSI. For a certain WiMax RSSI value, we also vary the distance between the GO and GM to get different levles of WiFi RSSI. In this experiment, We have one GM connected to the GO and measure the performance on GM. We run wget on GM to download a big file in the remote server and measure the throughput between the GM and the server. Fig. 9(b) shows the experiment result. We see

From the result, we can tell that the WiMax link quality indeed has more influence on the performance of the D2G network than the WiFi link quality. Therefore, it is necessary to require GO to have a good network connection to the BS as illustrated in section IV-C.

D. Effect of Similarity inside D2G

The value of token and the amount of traffic redundancies together affect the incentives for GOs to participate in D2G networks. Besides, by taking advantage of caching, GMs could gain improved response time by joining D2G network, as we mentioned in section III. We hereby explore the improvement of response time on GMs in D2G networks. In this experiment, an Android v4.3 phone is set up as the GO, on which the polipoid [38] is installed as the D2G cache.

Shown as Fig. 9(c). We record and compare latencies the GM experiences, when it directly connects to the BS versus to the GO. It browses web contents which are totally different, similar and the same with what the GO has browsed. We measured the latencies in seconds of all targets visited and recorded the average. We are shown that when they browse the same web content, the GM only needs half of the time to get requested content, which is 100% speedup than that in the direct connection scenario. When they have part of the content the same (which we denoted as 'similar' in the figure), the D2G wins. However, D2G is not preferable when they fetch totally different information. This is why we propose that the BS needs to select devices who share common interests as GMs by checking their browsing history.

VI. Conclusion

We proposed to extend the device-to-device (D2D) concept to a *device-to-group* (D2G) concept, where the cellular base station will attempt to aggregate devices originally connected to the base station to reduce the workload on the base station. We proposed a WiGroup algorithm as a practical means of implementing D2G, and the results indicate that WiGroup is able to reduce the workload on cellular networks.

References

[1] New York Times, "iPhones Overload AT & T Network, Angering Customers," September, 2009.

[2] "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2013–2018," in *Cisco White Paper*, 2014.

[3] K. Doppler, M. Rinne *et al.*, "Device-to-device communication as an underlay to LTE-advanced networks," *Communications Magazine*, 2009.

[4] B. Raghothaman, E. Deng *et al.*, "Architecture and protocols for LTE-based device to device communication," in *Computing, Networking and Communications (ICNC)*. IEEE, 2013.

[5] H. Zhang, Y. Wang, and C. C. Tan, "WD2: An Improved WiFi-Direct Group Formation Protocol," in *ACM MobiCom Workshop on Challenged Networks (CHANTS)*, 2014.

[6] T. Bansal, K. Sundaresan *et al.*, "R2D2: Embracing Device-to-Device Communication in Next Generation Cellular Networks," *Cell*.

[7] A. Pyattaev, K. Johnsson *et al.*, "3GPP LTE traffic offloading onto WiFi direct," in *Wireless Communications and Networking Conference Workshops (WCNCW)*. IEEE, 2013.

[8] M. Z. Shafiq, L. Ji *et al.*, "A first look at cellular network performance during crowded events," in *ACM SIGMETRICS*, 2013.

[9] B. Han, P. Hui, V. Kumar, M. V. Marathe, G. Pei, and A. Srinivasan, "Cellular traffic offloading through opportunistic communications: a case study," in *ACM Workshop on Challenged Networks (CHANTS)*, 2010.

[10] H. Luo, R. Ramjee, P. Sinha, L. E. Li, and S. Lu, "Ucan: a unified cellular and ad-hoc network architecture," in *ACM International Conference on Mobile Computing and Networking (Mobicom)*, 2003.

[11] S. Trifunovic, B. Distl, D. Schatzmann, and F. Legendre, "Wifi-opp: ad-hoc-less opportunistic networking," in *ACM Workshop on Challenged Networks (CHANTS)*, 2011.

[12] B. Tavli and W. Heinzelman, *Mobile Ad Hoc Networks*. Springer, 2006.

[13] A. Anand, C. Muthukrishnan *et al.*, "Redundancy in network traffic: findings and implications," in *Proceedings of the eleventh international joint conference on Measurement and modeling of computer systems*. ACM, 2009.

[14] J. Ziv and A. Lempel, "A universal algorithm for sequential data compression," *IEEE Transactions on information theory*, 1977.

[15] H. Pucha, D. G. Andersen, and M. Kaminsky, "Exploiting Similarity for Multi-Source Downloads Using File Handprints." in *NSDI*, 2007.

[16] S. C. Rhea, K. Liang, and E. Brewer, "Value-based web caching," in *Proceedings of the 12th international conference on World Wide Web*. ACM, 2003.

[17] "Ericsson Mobility Report," November 2013.

[18] S. Sanadhya, R. Sivakumar *et al.*, "Asymmetric caching: improved network deduplication for mobile devices," in *ACM International conference on Mobile computing and networking*, 2012.

[19] I. Papapanagiotou, R. D. Callaway, and M. Devetsikiotis, "Chunk and object level deduplication for web optimization: a hybrid approach," in *Communications (ICC)*. IEEE, 2012.

[20] R. Filipe and J. Barreto, "End-to-end data deduplication for the mobile Web," in *Network Computing and Applications (NCA)*. IEEE, 2011.

[21] N. Do, C.-H. Hsu, and N. Venkatasubramanian, "CrowdMAC: a crowdsourcing system for mobile access," in *International Middleware Conference*, 2012.

[22] N. Balasubramanian, A. Balasubramanian, and A. Venkataramani, "Energy consumption in mobile phones: a measurement study and implications for network applications," in *ACM Internet Measurement Conference (IMC)*, 2009.

[23] A. Sharma, V. Navda *et al.*, "Cool-Tether: energy efficient on-the-fly wifi hot-spots using mobile phones," in *International conference on Emerging networking experiments and technologies*, 2009.

[24] X. Hu, L. Meng, and A. D. Striegel, "Evaluating the Raw Potential for Device-to-Device Caching via Co-Location," 2014.

[25] M. Ji, G. Caire, and A. F. Molisch, "Wireless device-to-device caching networks: Basic principles and system performance," *arXiv preprint arXiv:1305.5216*, 2013.

[26] S. Sesia, I. Toufik, and M. Baker, *LTE: the UMTS long term evolution*. Wiley Online Library, 2009.

[27] Intel, "Increasing Mobile Operators Value Proposition With Edge Computing," http://networkbuilders.intel.com/docs/Nokia_Solutions_and_Networks.pdf.

[28] IBM, "Smarter wireless networks: Add intelligence to the mobile network edge." [Online]. Available: http://www-_01.ibm.com/common/ssi/cgi-_bin/ssialias?infotype=SA\&subtype=WH\&htmlfid=WSW14201USEN

[29] C. Mobile, "C-RAN: the road towards green RAN," *White Paper, ver*, 2011.

[30] K. Sundaresan, M. Y. Arslan *et al.*, "FluidNet: a flexible cloud-based radio access network for small cells," in *Proceedings of the 19th annual international conference on Mobile computing & networking*, 2013.

[31] S. Bhaumik, S. P. Chandrabose *et al.*, "Cloudiq: a framework for processing base stations in a data center," in *ACM International conference on Mobile computing and networking*, 2012.

[32] C. Liu, K. Sundaresan *et al.*, "The case for re-configurable backhaul in cloud-RAN based small cell networks," in *INFOCOM, 2013 Proceedings*. IEEE, 2013.

[33] WINLAB, "GENI - Open, Programmable WiMAX Base Station," http://www.winlab.rutgers.edu/pub/docs/focus/GENI-_WiMAX.html.

[34] W.-F. Alliance, "Wi-Fi Direct," http://www.wi-_fi.org/discover-_wi-_fi/wi-_fi-_direct.

[35] I. Rhee, M. Shin, S. Hong, K. Lee, S. Kim, and S. Chong, "CRAWDAD data set ncsu/mobilitymodels (v. 2009-07-23)," Downloaded from http://crawdad.org/ncsu/mobilitymodels/, Jul. 2009.

[36] C. Phillips, D. Sicker, and D. Grunwald, "A survey of wireless path loss prediction and coverage mapping methods," *Communications Surveys & Tutorials*, 2013.

[37] T. F. Gonzalez, "Clustering to minimize the maximum intercluster distance," *Theoretical Computer Science*, 1985.

[38] "Polipoid," Accessed in 02/2014, https://github.com/splondike/polipoid.