AccuLoc: Practical Localization of Performance Measurements in 3G Networks

Qiang Xu[1] Alexandre Gerber[2] Z. Morley Mao[1] Jeffrey Pang[2]
[1]University of Michigan and [2]AT&T Labs Research

ABSTRACT

Operators of 3G data networks have to distinguish the performance of each geographic area in their 3G networks to detect and resolve located network problems. This is because the quality of the "last mile" radio link between 3G base stations and end-user devices is a crucial factor in the end-to-end performance that each user experiences. It is relatively straightforward to measure the performance of all IP traffic in the 3G network from a small number of vantage points in the core network. However, the location information available about each mobile device (*e.g.*, the cell sector/site that it is in) is often too stale to be accurate because of user mobility. Moreover, it is impractical to collect fine-grained location information about all mobile devices on an on-going basis in large 3G networks due to expensive measurement overhead. Thus, it is a challenge to accurately assign IP performance measurements to fine-grained geographic regions of the 3G network using off-the-shelf components. Fortunately, previous studies have observed that human mobility patterns are very predictable. In this paper, we exploit this predictability to develop a novel clustering algorithm grouping related cell sectors that accurately assigns IP performance measurements to fine-grained geographic regions. We present results from a prototype in a real 3G network that shows our approach provides more accurate performance localization than existing approaches. Eventually, we can either narrow down individual IP performance measurements into only 4 candidate cell sectors consistently with the accuracy of 70% over one week based on a one-day snapshot of fine-grained 3GPP events, or increase the accuracy 20% comparing with site-level accuracy through lightweight handover statistics hourly collected at RNCs. Using our approach, we improve the anomaly detection based on IP performance measurements by reducing both false positive and false negative. Our study also sheds light on the mobility patterns of 3G devices.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Wireless communication; C.2.3 [**Network Operations**]: Network monitoring; C.4 [**Performance of Systems**]: Measurement techniques

General Terms

Experimentation, Measurement, Performance

Keywords

Cellular Network Architecture, Localization, Performance Anomaly Detection

1. INTRODUCTION

Mobile applications over 3G networks are among the fastest growing classes of network applications today. Network operators thus have a substantial interest in monitoring the performance of IP data traffic on their 3G data networks. In particular, operators would like to continuously monitor which geographical regions in their 3G networks are performing well and which ones are performing poorly. This is because the quality of the "last mile" radio link between 3G base stations and end-user devices is a crucial factor in the end-to-end performance that each user experiences. Unfortunately, due to protocol, equipment, capacity, and cost limitations, it is not trivial to accurately associate an end-to-end performance measurement to the 3G network path that it traversed. We redress this problem in this paper by developing a novel clustering algorithm and evaluating a prototype system in a real 3G wireless network.

The collection of IP-level statistics, such as packet or flow records, is crucial for an operator to understand the end-to-end performance of its users because they are basic to compute metrics such as end-to-end throughput, RTT, and loss. Due to the standard organization of 3G data networks, such as the UMTS network shown in Figure 1, it is impractical for operators of large 3G networks to collect IP-level statistics from a user that can readily be associated with the geographical region where he is located. Vendor equipment in a 3G network does not typically support the capture of IP-level flow statistics because IP packets are carried in an opaque lower-layer tunnel from the end-user device all the way to the Gateway GRPS Support Node (GGSN). Since the capital and labor costs associated with deploying additional monitoring equipment and backhaul capacity at all Radio Network Controller (RNC) or Serving Gateway Support Node (SGSN) locations is prohibitive, monitoring of IP-level statistics is only practically performed at the GGSNs, which are located in only a handful of different locations. However, from the perspective of a GGSN, it is not possible to determine when a mobile device moves from one cell sector to another because handover signaling information is not propagated up. The GGSN only observes the cell sector where the user began his session, the cell sector when the user moves far enough away from his original location that his network path traverses a different SGSN (each SGSN generally covers an entire metro area), or when the device changes from 3G to 2.5G coverage. Because users are often mobile, handovers are frequent and the GGSN will often have a stale view of where users are currently located. This staleness makes it non-trivial to associate current IP-level performance measurements to the cell sectors that they are associated with using only information collected at the GGSN.

RNCs, which observe all handovers that users experience, can collect accurate information about which cell sector each user is using at all times. However, not all equipment supports such fine-grained user tracking. Moreover, because RNCs are geographically distributed, significant additional long-haul capacity would be needed to collect all handover events in a centralized location. Thus, in large 3G networks, it is only practical to collect such information infrequently or in aggregate form. For example, the RNCs in the UMTS network that we study in this paper only collect hourly handover statistics per cell sector, rather than real-time cell sector information per user. This information indicates how many users move from cell sector to cell sector in aggregate, but not where any user is at any point in time.

In this paper, we develop a system called **AccuLoc** that takes these two sources of 3G network data — IP-level flow records collected at GGSNs and aggregate handover statistics at RNCs — and accurately associates end-to-end metrics to different fine-grained regions of the 3G network. **AccuLoc** leverages the observation that human mobility patterns are typically predictable and most users do not tend to move long distances at short time scales. Thus, it is possible to cluster related cell sectors together using aggregate handover information. By associating end-to-end measurements to the cluster in which a user began his session, we can accurately associate these measurements with fine-grained geographic areas covered by the 3G network. To evaluate **AccuLoc**'s accuracy in associating the end-to-end metrics to fine-grained regions, we have developed a prototype system in a real 3G wireless network and evaluate it on data from a large metropolitan area. We show that by clustering based on aggregate handover statistics, **AccuLoc** is more accurate than naïve forms of clustering, such as clustering purely by geographic proximity.

To our best knowledge, our study is the first one quantifying the localization inaccuracy of IP-level statistics at GGSNs due to stale views. and leveraging human mobility patterns for cellular operators to achieve better localization. Through the design and development of **AccuLoc**, we make the following five contributions:

- We characterize the localization inaccuracy for mapping the IP-level statistics at the GGSN to different fine-grained network elements, *i.e.*, cell sectors, base stations, RNCs, and LACs. The localization accuracy is around 20% at the granularity of cell sector level. Even if at higher aggregation levels, the accuracy is only around 50% at the cell-site level and 70% at the RNC level. The low accuracy is because using cell sites and RNCs to determine which cell sectors are related cannot capture the dynamics of user moving behaviors, which motivates us to obtain human mobility patterns in advance and leverage it for locating IP-level statistics accordingly.

- We propose two measurement-driven solutions for **AccuLoc** to build human mobility patterns in the terms of which cell sectors are strongly related, *i.e.*, identifying clusters of related cell sectors that subscribers have very high probability commuting within individual clusters, and leverage the mobility knowledge to accurately localize IP-level statistics. The two solutions have different advantages and complement each other from aspects of the localization accuracy and the overhead for conducting human mobility patterns. Since human mobility patterns are dynamic, capturing the variability of mobility patterns is critical for **AccuLoc** to outperform other naïve solutions.

- Our first solution, *i.e.*, **BIGRAPH**, requires a snapshot of 3GPP signaling events at RNCs which is expensive for long-term collection. In order to construct the human mobility patterns, **BIGRAPH** groups related cell sectors into small clusters. **BIGRAPH** can locate IP-level statistics into only 4 cell sectors with the accuracy of 70% over one week and 50% after the snapshot of 3GPP signaling events is 5.5-month old. Note that the mapping IP-level statistics to the correct RNC is around 70%, but one RNC usually contains 200 – 300 cell sectors, which is significantly larger than **BIGRAPH**'s clusters.

- Our second solution, *i.e.*, **HANDOVER**, relies on hourly aggregate handover statistics at cell sectors instead of expensive 3GPP signaling events at RNCs. **HANDOVER** performs as an alternative to **BIGRAPH** on condition that the collection of 3GPP signaling events is not supported or is restricted. Since it is an inherit tradeoff between the overhead of measurement and localization accuracy. **HANDOVER** is not as accurate as **BIGRAPH**. However, inferring the mobility patterns from lightweight handover statistics, **HANDOVER** still achieves reasonable localization accuracy. Compared with intuitive solutions such as grouping sectors purely by cell sites, **HANDOVER** can overall increase the accuracy 20%.

- Based on the information inferred from either **BIGRAPH** or **HANDOVER** of which cell sectors are related, **AccuLoc** re-assign the measured IP-level statistics in order to accurately associate the end-to-end performance metrics to the correct fine-grained network elements, *i.e.*, cell sectors, cell sites, RNCs. Through the re-assignment, the performance metrics observed at GGSNs are more close to the ground truth ones directly measured at RNCs. Applying **BIGRAPH** in performance anomaly detection, **AccuLoc** achieves both the lowest false positive and negative compared with solutions based on other forms of clustering sectors.

The rest of this paper is organized as follows: §2 describes the architecture of cellular networks and associated protocols, followed by §3 explaining the main data sources for the input, the ground truth, and the evaluation. §4 proposes the two solutions, **BIGRAPH** and **HANDOVER** adopted by **AccuLoc** to build the knowledge of human mobility patterns. §5 quantifies the performance of **AccuLoc** in associating metrics to fine-grained network locations. We discuss the generalizability of **AccuLoc** on other types of networks in §6. Related work is discussed in §7 and we conclude our study in §8.

2. UMTS BACKGROUND

In order to understand the difficulty in locating IP performance measurements in UMTS networks, it is useful to have an understanding of how a UMTS data network is structured.

Network Elements and Architecture. Figure 1 shows the logical architecture of a UMTS data network according to the 3GPP standard. As depicted, a UMTS network is hierarchical. At the root of the network is a Gateway GRPS Support Node (GGSN). In practice, there are multiple GGSNs, but they are located in only a

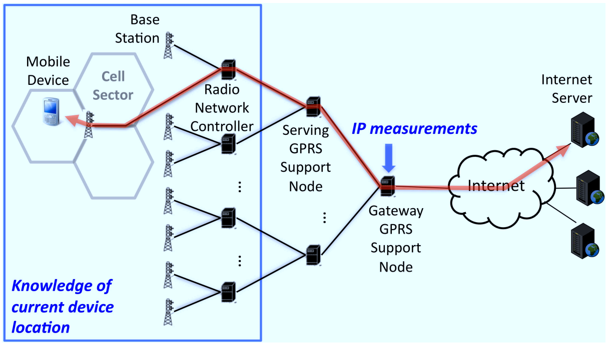


Figure 1: Logical architecture of a UMTS network.

handful of locations [25]. Due to their limited number of physical locations, it is relatively straightforward to monitor all IP traffic in the UMTS network at these locations. At the leaves are mobile devices (user equipment (UE), in 3GPP parlance), which connect to the UMTS network in a particular cell sector. Each base station (NodeB or cell site, in 3GPP parlance) has multiple cell sectors, one for each antenna attached to its cell tower. Typically these point in different directions and/or operate on different frequencies. Base stations send their data traffic to Radio Network Controllers (RNCs), which forward traffic to Serving GPRS Support Nodes (SGSNs), which, in turn, send the traffic to GGSNs. The GGSN sends and receives traffic from the Internet.

IP Tunneling. An important characteristic of UMTS networks is that IP traffic sent by mobile devices is tunneled to the GGSN using lower layer 3GPP tunneling protocols. As a consequence, none of the intermediary nodes in the UMTS network can directly inspect the sent IP packets and a mobile device's IP address is "anchored" to the GGSN, regardless of where it moves in the network. This characteristic ensures that the mobile device can maintain its IP address (and thus, its IP connections) even as it is mobile. In this paper, we will focus on the tunnel between the SGSN and the GGSN, which is called a PDP Context and uses the GPRS Tunneling Protocol (GTP) (GTP-U to carry data traffic and GTP-C for signaling control messages).

Session Establishment and Mobility. When a mobile device first connects to the UMTS network, the PDP Context that carries its IP traffic is set up. At this point, the originating cell sector and RNC is reported to the GGSN via GTP-C protocol. When a mobile device moves to a different sector, the path its data takes through the UMTS network changes.[1] RNCs manage the operation of handovers when a mobile device moves from one sector to another (*e.g.,* by coordinating base stations and other RNCs). However, to avoid unnecessary signaling overhead, the change of cell sectors is not reported to the higher in the hierarchy. Thus, the GGSN is not informed that a mobile device has moved unless the SGSN in its network path changes. This can occur for two reasons: (1) it moves far enough away that the SGSN changes — typically into a different metro area; or (2) the device changes from 3G to 2.5G, WiFi,or vice versa. The second scenario causes an SGSN change because the 2.5G hierarchy is different from the 3G hierarchy. This scenario typically occurs if a device moves from 3G areas that cover primary urban and suburban areas to 2.5G areas that cover less populated areas. In addition, the PDP Context is destroyed after an inactivity period of 2–4 hours or if the device is turned off. Note that the PDP Context remains alive even if the device is idle. Since smartphone applications may send periodic keep-alives or "push" notifications, a PDP Context may persist for hours or even days. Therefore, the initial cell sector reported to the GGSN when a device first sets up the PDP Context often is not the sector in which the device currently is connected.

[1]In practice, a device can be connected to multiple nearby sectors at the same time. This set of sectors, typically 1 to 4 in size, is called the active set. While all sectors in the active set coordinate to receive uplink data sent by the device, only one, the serving cell, transmits downlink data to the device at a given time. This is typically the sector with the highest signal-to-noise ratio. Since the vast majority of data is downlink traffic, in this paper we are only concerned with identifying the serving cell correctly.

3. DATA SOURCES

There are several sources of data in a UMTS network that we can utilize to measure the end-to-end performance experience of each cell sector. In this section, we describe these sources and the data sets we use to evaluate our prototype system. Note that due to privacy concerns, without compromising the usefulness of the results, we have anonymized the device identifier, *i.e.,* IMSI and IMEI.

3.1 Continuously Collected Data Sets

We are primarily interested in the performance of IP data traffic — *e.g.,* the throughput, RTT, loss, etc. of IP data flows. These metrics can be extracted from statistics captured about each IP flow [7, 10, 20]. Ideally, we would like to be able to collect IP flow data such that each IP flow can be mapped to the cell sector where it originates from or is destined to. Unfortunately, as described in Section 2, this is not trivial due to lack of available data sources. This section describes the data sources that are available on a regular basis.

Real-time IP Flow Records. It is relatively straightforward to capture IP flow data from all 3G traffic at all GGSNs because they are few in number. In the large UMTS network that we study, measurement infrastructure [9] is in place to capture IP flow records similar to Netflow records [8] in near real time. GTP-C signaling messages, described in the next paragraph, are used to map each IP flow to the originating or destination device, which is identified by its anonymized IMSI and IMEI. Hereafter, we call these IP flow records **IPFlowRecords**.

PDP Context Setup Messages. Similarly, it is straightforward for the same infrastructure to capture the signaling messages sent between SGSNs and GGSNs via the GTP-C protocol. Most importantly, PDP Context Setup messages that are exchanged when a device initially establishes a PDP Context indicate the initial sector that the device communicates with. PDP Context Update messages may also indicate a device's sector when it moves far enough away from the original sector so that the SGSN changes. Without any information collected outside the GGSN locations, these are the best estimates of device location that are available. Hereafter, we call the estimates of device location derived from these messages **PDPSetupLocations**.

Aggregate Handover Counters. Each RNC keeps track of the current cell a device is using as part of normal operation. However, due to vendor limitations and resource constraints, this information is not recorded. Instead, it is typically only practical to keep aggregate statistics about each sector. For example, the total number of connections, total number of disconnects, etc. One aggregate statistic that we can leverage is the total number of handovers between two sectors. In other words, for each pair of sectors (A,B), a counter is kept that indicates the number of handovers from A to B processed per hour. From these handover statistics, we can infer

dataset	availability	duration	description
IPFlowRecords	continuously	1 day	real-time IP flow records collected at the GGSN
PDPSetupLocations	continuously	1 week	sector information in PDP Context Setup messages collected at the GGSN
HandoverCounters	continuously	1 week	hourly aggregate handover counters for each sector pair reported by RNCs
RNCGroundTruth	infrequently	1 week	ground truth sector information for each device from 3GPP events collected at RNCs

Table 1: Datasets used in the evaluation of AccuLoc.

the aggregate mobility behavior of devices. Hereafter, we call this set of handover counters **HandoverCounters**.

3.2 Ground Truth Data

Some RNC equipment can record the current sector that each device is using at fine time scales. More specifically, some equipment can capture all 3GPP signaling events at the RNC level, such as handover events. However, this recording places additional load on RNC equipment that can interfere with normal operation, as the CPU, memory, and storage constraints of RNC equipment are not designed for continuous operation of such recording. Recording is typically only enabled for troubleshooting. In addition, the volume of such data is substantial (tens of GB per day for a single RNC), so backhauling the data to a central data collector for correlation with **IPFlowRecords** requires investment of additional resources. Finally, not all RNC vendors support such recording. Therefore, although it is possible to collect such data from a small number of RNCs periodically (*e.g.,* once every few days), continuous collection to support real-time performance localization is not possible.

In order to evaluate different approaches to our localization problem, we collect a sample of this "ground truth" data. Hereafter, we refer to this data as **RNCGroundTruth**.

3.3 Evaluation Data Sets

To evaluate our prototype and other approaches to the localization problem, we use one contiguous week of data in July 2010 for each of these datasets: **PDPSetupLocations**, **HandoverCounters**, **RNCGroundTruth**. To evaluate the accuracy of performance measurements based on these localization approaches, we use one day of **IPFlowRecords** data during this week. The data we examine covers all 3G sectors in the greater Los Angeles area. We note that not all areas have 3G coverage (some only have 2.5G coverage). However, since our focus is on 3G performance, we do not consider data from 2.5G sectors.

The datasets are summarized in Table 1.

4. LOCALIZATION

As we described in the previous section, it is common that the originating cell sector/cell site/RNC of performance measurements captured at the GGSN as it is impractical to collect fine-grained location information about all mobile devices on an on-going basis in large 3G networks.

In this section, we first characterize the localization inaccuracy based upon **RNCGroundTruth**, then propose two solutions adopted by **AccuLoc** to group related sectors together that take advantage of the predictability of human mobility patterns, and finally evaluate the performance of **AccuLoc** in terms of localization accuracy at the end of this section.

4.1 Characterizing the Inaccuracy of Initial Sectors

Understanding the duration of PDP Contexts, frequency of handovers, and their relationship to each other is important both for understanding performance localization problem we address in this paper and to shed light on user mobility patterns with respect to cellular networks in general. For example, the persistence of the same location in a PDP Context suggests how long modern smartphones are active and remain in the same metro area, as PDP Contexts only change if a user changes SGSN due to travel to another metro area, or departure from the 3G coverage area. Frequency of handovers suggests how often user mobility and environmental changes cause local radio characteristics to change substantially in the immediate vicinity (within 1-2 km), as they typically occur only when the sector with the best signal-to-noise ratio changes. These characteristics are important for a range of cellular applications so we study them in detail in this section.

GTP-C signaling messages, *i.e.,* **PDPSetupLocations**, during the initial PDP Context Setup provide the location of mobile devices at the time a device is turned on or after several hours of inactivity. 3GPP events collected from RNCs, *i.e.,* **RNCGroundTruth**, provide the location of the mobile devices every 2 seconds in terms of the cell sector where each device is located. Comparing the location information of **PDPSetupLocations** with that of **RNCGroundTruth**, we can evaluate how accurately **PDPSetupLocations** estimates where mobile devices currently are in.

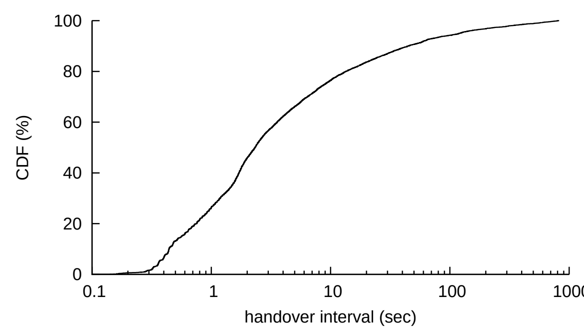


Figure 2: Interval between two consecutive handovers on device.

One type of 3GPP events in **RNCGroundTruth** record the occurrences of handover. Figure 2 depicts the CDF of the interval between two consecutive handovers on individual devices. 80% of handovers happen within 10 seconds after the last handover. Figure 3 shows the CDF of the age of PDP Contexts of the **RNCGroundTruth** records. Given a **RNCGroundTruth** record, we can discover the beginning of its corresponding PDP Context from **PDPSetupLocations** via the anonymized device identifier. The timestamp of **RNCGroundTruth** record minus the timestamp of the PDP Context leaves the age of PDP Context. The majority of PDP Contexts fall in the range of 1 hour – 1 day. Given the presence of so frequent handovers and the longevity of PDP Contexts, GGSNs will inevitably miss many sector changes in a single PDP Context. We can infer the potential high probability of that the cur-

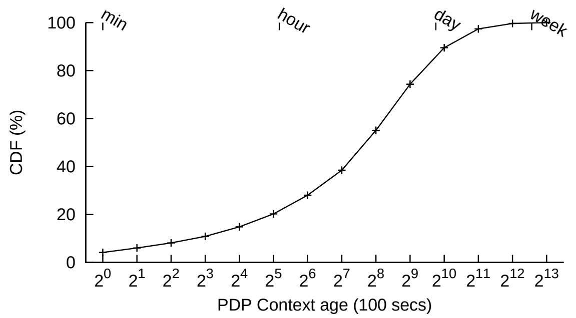


Figure 3: Age of PDP Context age of RNCGroundTruth records.

rent cell sector differs from the sector where the PDP Context Setup starts.

Operators are interested in determining the network path that each IP flow takes through the UMTS network. This is so that when IP flows exhibit problems, the problems can be isolated to a particular cell sector, cell site, or RNC. As a device moves away from its initial cell sector, the cell site and RNC through which its IP packets traverse will change. To understand how quickly device mobility causes the network path to change, we examine how accurately the initial cell sector represents the current network path at each level of the UMTS network hierarchy.

We define the accuracy of a particular level of the hierarchy cell sector/cell site/RNC as the percentage of time that the initial element the path traverses is the same as the actual element that the path traverses. For example, if the current sector is the same as the initial sector 20% of the cases, the accuracy at the cell sector level is 20%. We expect that accuracy at higher levels of the hierarchy (*e.g.*, RNC) will have higher accuracy, as a device has to move a greater distance before the path its packets take no longer traverse that element. Since devices are likely to move farther away from their initial locations over time, the older a PDP Context is, the more likely its setup location is inaccurate. Thus we show the inaccuracy of **PDPSetupLocations** with PDP Context age.

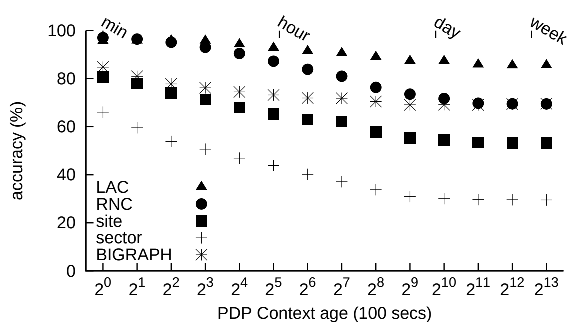


Figure 4: Accuracy over PDP Context age / BIGRAPH's accuracy over PDP Context age.

We evaluate the accuracy at each level of the UMTS network hierarchy: cell sector, cell site, and RNC. In addition, we evaluate the accuracy of each location area code (LAC). A LAC is the set of sectors that are paged when a mobile is idle and the network must search for it (*e.g.*, for an incoming call). Since a device must wake up when moving from one LAC or another to update its status, network planners attempt to group sectors in a LAC so that inter-LAC movement is rare. However, this also means that a LAC typically covers a large number of sectors and is not granular enough to pin point geographically constrained performance problems. For the elements in each aggregation level, cell sites cover 3 – 6 sectors, and RNCs and LACs cover about hundreds of sectors.

Figure 4 shows the accuracy over the PDP Context age at different levels of hierarchy. As mentioned, we can obtain the PDP Context age for each **RNCGroundTruth** record. Aggregating **RNCGroundTruth** records of the same PDP Context age together, we can identify the probability that the current cell sector is the same as the initial cell sector during PDP Context Setup, *i.e.*, accuracy, for each value of PDP Context age. It is expected that the accuracy is reasonably high if the PDP Context age is less than 1 minute. However, the sector-level accuracy decreases very fast as the PDP Context age increases, which verifies our previous inference from Figures 2 3. After the PDP Context has been activated for hours, the accuracy at the sector level is around 20% to 30%, which implies that 70% to 80% of the end-to-end performance measurements at the GGSN are assigned to incorrect cell sectors. As expected, the site-/RNC-/LAC-level accuracy is higher than the sector-level accuracy. However, the site-level accuracy is only 50% to 60% after the PDP age rises to hours, which means mobile devices have better than even odds of moving out of its current cell site several hours after the PDP Context starts. The RNC-level accuracy is 70% to 80% and the LAC-level accuracy is around 90%. Note that a typical cell site includes 3 – 6 sectors. one RNC or one LAC contains hundreds of sectors. Thus, it is too coarse-grained to use these aggregations to locate a device to a very granular geographic region.

However, just because these hierarchical clusters of sector, *i.e.*, cell site, RNC, or LAC, are not very accurate in locating measurements, it doesn't mean we cannot discover a better manner of aggregating related sectors. For example, perhaps sectors from two neighboring sites form a good cluster because they cover two areas that subscribers frequently commute back and forth. In general, if movement patterns are common amongst many subscribers, then we expect that we can learn the patterns and group related sectors into clusters accordingly (comparing with clustering by site, RNC, or LAC) such that these clusters are small and accurate.

One way movement patterns can be similar is if users do not move very far away from the sector in which they started. The geographic distance between the base station recorded by **PDPSetupLocations** and the ground truth base station by **RNCGroundTruth** estimates the distance a user has moved. If these distances are generally small, then human mobility patterns are revealed to some degree. Figure 5 shows the physical distance between cell site by the **PDPSetupLocations** and the cell site by **RNCGroundTruth**. Even if the time after the PDP Context has been initialized one day, the distance is still small. The maximum median error distance is 1.65km although some subscribers can still move away for more than 10km. This consistent small distance implies that most subscribers only move within a small geographic area. So if we can discover which set of sectors are always related (*i.e.*, those between which users frequently move), we can group them together so that if we want to predict the current sector of performance measurements in **IPFlowRecords** based on **PDPSetupLocations**, we can have a small set of candidate sectors but in very high confidence. This technique can be beneficial in practice in detecting performance anomalies and narrowing down problems into a small number of sectors.

Note that Figure 4 shows that the site-level accuracy is poor, which means subscribers often move in an area served by more than one cell site. However, subscribers moving across sites do not necessarily means they moving across many sectors. Subscribers may always move across a few sectors but these sectors may be in different cell sites. In the next section, we describe how we can learn these small clusters of related sectors, regardless of cell site.

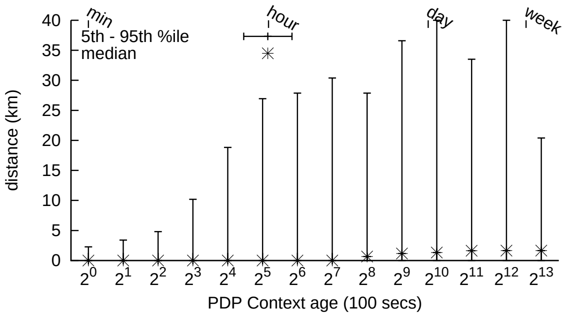


Figure 5: Distance from the site by PDPSetupLocations to the site by RNCGroundTruth.

4.2 Practical Localization Solutions

As we observed in the previous section, using only static information to cluster sectors, such as by the cell site each sector belongs to, is not accurate because they do not capture dynamic human mobility patterns. We expect that a good heuristic is able to learn the movement behavior of users and leverage this knowledge to predict the current sector more accurately.

As mentioned in Section 3, there are two available sources of data collected at RNCs in addition to the **IPFlowRecords** and **PDPSetupLocations** data collected at the GGSN: **HandoverCounters** and **RNCGroundTruth**. **HandoverCounters**, aggregate counters of handovers between each sector, are collected continuously. **RNCGroundTruth**, precise location information based on RNC event information, cannot be collected continuously, but can be sampled from small segments of the network every few days. In this section, we propose two algorithms to build mobility models using these data sources, given their collection and granularity constraints: **BIGRAPH** and **HANDOVER**. Using **RNCGroundTruth**, **BIGRAPH** has higher data collection overhead and is more accurate. However, we find that the algorithm using **HandoverCounters**, *i.e.*, **HANDOVER**, also provides acceptable accuracy. Both solutions can be formulated as a version of the sparsest cut of a graph cut problem [14], so we first describe how to formulate each graph. The practical algorithm to solve the sparsest cut problem on each graph is the same.

4.2.1 BIGRAPH: *a solution based on one-time snapshot of* RNCGroundTruth

Since there are often common travel routes between common areas that most people follow, we conjecture that subscribers that begin their PDP Contexts in the same sector tend to move to similar sectors. In addition, as we see in Figure 5, most devices do not move very far away from their initial sector. Based on these observations, we expect that there will be small clusters of sectors close to each other that can be grouped together. Subscribers have high probability to commute within these sectors in the same cluster.

BIGRAPH prerequisites (i) a snapshot of **RNCGroundTruth**; and (ii) the corresponding snapshot of **PDPSetupLocations** whose PDP Contexts covers the records in the snapshot of **RNCGroundTruth**. **BIGRAPH** attempts to learn these clusters by creating a graph of the relationships between the initial sectors in **PDPSetupLocations** and the current sectors where devices are located in **RNCGroundTruth**. Since **RNCGroundTruth** can only be collected infrequently, **BIGRAPH** builds a model of these clusters based on a set of training data (*e.g.*, collected over one day from all RNCs in a greater metro area), then the model can be used to predict relationships in future data when **RNCGroundTruth** is not available.

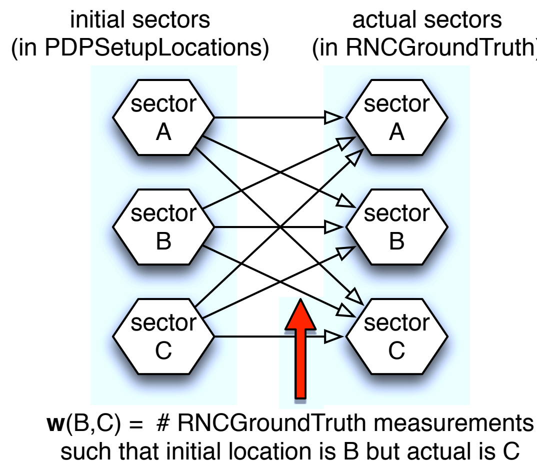


Figure 6: BIGRAPH constructing the bipartite graph.

As shown by Figure 6, the **BIGRAPH** builds a bipartite graph connecting **RNCGroundTruth** and **PDPSetupLocations**. Let us denote the graph as $G = (U,V,E)$, where vertex u in U represents a sector in **PDPSetupLocations** and u in V represents a sector in **RNCGroundTruth**. Let $w(u,v)$ be the number of **RNCGroundTruth** records (once every 2 seconds when a user is active) such that **RNCGroundTruth** reports a subscriber is in v and **PDPSetupLocations** says the subscriber is in u. $M(u)$ returns the vertices in V that corresponds to the sector that u represents in U. Edges in G that have high weights are strongly related (*i.e.*, lots of users move from those source sectors to those other sectors). Thus, we would like to cluster strongly related nodes in V together.

Let a clustering $C_1, C_2, \ldots, C_n$ each be a disjoint subset of V. The accuracy of a clustering of sectors (for any clustering method) is computed as $\frac{1-E(C_1,C_2,\ldots,C_n)}{N}$, where N is the number of **RNCGroundTruth** records, and $E(C_1,C_2,\ldots,C_n)$ is the sum of $w(u,v)$ for all (u,v) such that $M(u)$ is in one cluster C_i and $M(v)$ is in another cluster C_j, *i.e.*, this sum counts the number of **RNCGroundTruth** records that get assigned to the incorrect cluster. Location accuracy is thus maximized when $E(C_1,C_2,\ldots,C_n)$ is minimized.

Therefore, given constraints on the size of clusters, the goal is to minimize the edges that cross clusters. For example, if we want to clusters to be of size 4, then we want to cut the graph such that each connected component is only size 4 and the weight of the edges that cross connected components is minimized. We can merge the vertices u and $M(u)$ to make the problem be a sparsest cut problem. We describe how to solve the sparsest cut problem below in §4.2.3

4.2.2 HANDOVER: *a solution based on hourly* HandoverCounters

Some RNC equipment is incapable of building **RNCGroundTruth**, and in some cases, it is imprudent to do so because it may interrupt normal operation. Thus, we formulate a second solution as an alternative to **BIGRAPH**, **HANDOVER** that uses only **HandoverCounters**, the aggregate handover counters. The motivation behind **HANDOVER** is that the handover counts between different sectors represents how frequently subscribers move between those sectors. Thus a graph with edges weighted by handover counts approximates the degree of movement between sectors.

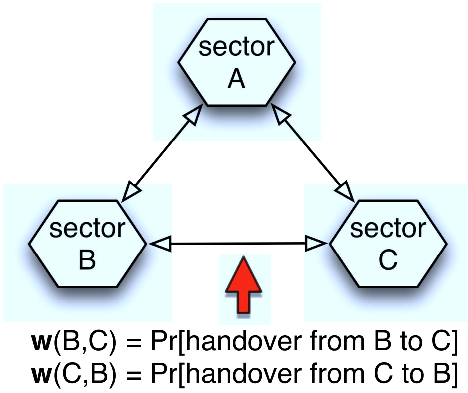


Figure 7: HANDOVER constructing the graph.

Similar to **BIGRAPH**, **HANDOVER** also refers to a graph (shown by Figure 7) to keep the relationship between sectors via weighted edges. However, **HANDOVER** does not require **RNCGroundTruth** which is hard to collect continuously. Instead, **HANDOVER** prerequisites only **HandoverCounters**, aggregated handover counters from RNCs which can be collected continuously. Each edge in the graph reflects the probability for subscribers from the source sector moving to the destination sector based on these counters (*i.e.,* $\Pr[\text{handover from } A \text{ to } B] =$ the ratio of handovers from A to B and the number of devices that entered A, either by starting there or via a handover). As with **HANDOVER**, our goal is to cut the graph into clusters with minimum edge cut given the constraint on the cluster size.

We note that **HANDOVER** may not perform as well as **BIGRAPH** because it does not distinguish sectors where devices begin PDP Contexts and sectors where they move into. In addition, it does not take into account the duration that users spend in each sector. For example, suppose we want a cluster size of 2. If there is a high likelihood of handovers between A and B, and B and C, but devices spend very little time active in B (*e.g.,* because it is a highway sector), then we will be more accurate clustering A and C together without B, as even though there is frequent movement between A and C to B, there will not often be activity (*i.e.,* measurements). In terms of performance monitoring, we do not care where users are when they are not active.

4.2.3 Solving the sparsest-cut problems

Once the graph containing the information of mobility patterns is constructed, both **BIGRAPH** and **HANDOVER** will be formulated as a sparest cut problem, *i.e.,* cutting the graph into connected components (clusters) of at most a particular size. In practice, we want the size of clusters to be as small as possible because we want to quickly identify the right sector for isolated performance problems. However, the smaller the cluster, the more information will be lost since we end up with ignoring the movement between sectors across clusters. We formulate our clustering into a recursive sparsest cut process. The sparest cut process is a bi-partition of the vertices in the graph that minimizes the ratio of the number of edges across the cut and keeps the two halves balanced. We recursively apply the sparest cut algorithm on both subgraphs until the size of these subgraphs hits the constraint on the cluster size. As the sparest cut is known to be a NP hard problem, we use an existing approximation algorithm, *i.e.,* Kernighan-Lin [14], to split the graph into two and recursively repeat it on both subgraphs until the size of the subgraphs satisfies the constraint on the predefined cluster size.

4.3 Localization Accuracy of Clustering Solutions

In order to evaluate the performance of our two solutions, we refer to the accuracy, *i.e.,* the fraction of measurements in **RNCGroundTruth** whose locations agree with the initial location from at the granularity of clusters.

4.3.1 BIGRAPH*'s accuracy in one-time snapshot*

As mentioned in §4.2.1, **BIGRAPH**'s accuracy can be obtained by comparing the location of each measurement in **RNCGroundTruth** against the initial location of the corresponding records reported by **PDPSetupLocations**. Note that the location is at the granularity of **BIGRAPH**'s clusters.

First, **BIGRAPH** computes the clusters via recursive sparest cut on a training data set whose **RNCGroundTruth** and **PDPSetupLocations** records were collected on July 21. Eventually, the recursive sparest cut will end with a set of clusters whose size are pre-decided. Second, we evaluate the accuracy on a evaluation data set at the granularity of these clusters from the training set. The **RNCGroundTruth** and **PDPSetupLocations** records on July 22 serve as evaluation data set. The training data set and the evaluation data set are very close to each other in time, so the human mobility patterns are up-to-date. Note that we expect **BIGRAPH** achieving its best performance when the human mobility pattern is most up-to-date.

Figure 4 compares **BIGRAPH**'s accuracy with site-/RNC-/LAC-level accuracy over the PDP Context age. The site-/RNC-/LAC-level accuracy is the accuracy at the granularity network elements of site/RNC/LAC. We can observe that **BIGRAPH**'s accuracy is significantly better than the site-level accuracy. In this comparison, the cluster size of **BIGRAPH** is 4, while the average number of sectors for all cell sites is 3 – 6. So, **BIGRAPH** uses smaller cluster size but achieve much better accuracy than using cell site to predict candidate sectors. **BIGRAPH**'s accuracy is even comparable to the RNC-level accuracy, but one RNC usually consists of 200 – 300 sectors.

As we mentioned in §4.2.3, cluster size is a constraint on the recursive sparest cut. The smaller the cluster size, the more information is lost by the cutting. However, smaller cluster size is beneficial in practice as it reduces the overhead to narrow down the candidate sectors, *i.e.,* it improves the localization granularity. Figure 8 shows the impact of cluster size on the accuracy. We can imagine that if the cluster size is ∞, the accuracy will be close to 1. When the cluster size is 1, which is equal to the sector-level accuracy. When the cluster size is 4, 8, or 16, the accuracy highly rises, which confirms our expectation that subscribers usually move within a small number of sectors.

Figure 9 explains why our small clusters perform better than cell sites, or even RNCs under some scenarios. We count the number of unique cell sites that a single **BIGRAPH**'s cluster includes. From Figure 9, we can observe that each cluster covers 2 cell sites in average when the cluster size is 4. So, the reason for the high accuracy of **BIGRAPH** is that **BIGRAPH** can flexibly capture the dynamics of human mobility patterns without restricted by static geographic regions, *e.g.,* cell sites.

Now we consider some issues regarding the deployment of **BIGRAPH** in 3G networks. In the deployment of **BIGRAPH** algorithm, there are 3 dimensions in fact: accuracy, cluster size, and number of measurements. The "number of measurements" relabels and rescales the "PDP Context age" in Figure 4, but the "num-

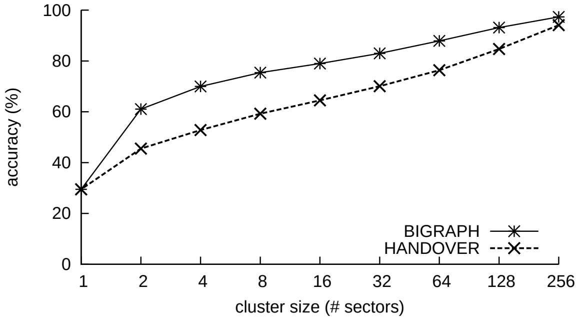


Figure 8: BIGRAPH's and HANDOVER's accuracy over cluster size.

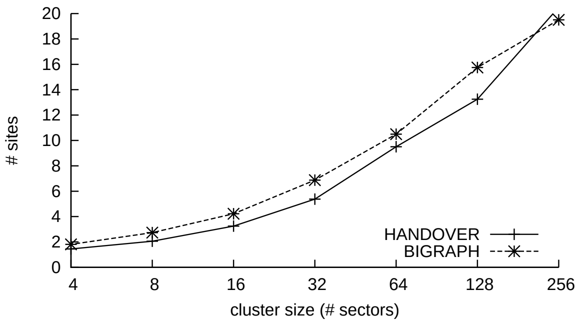


Figure 9: # cell sites covered by one BIGRAPH/HANDOVER cluster.

ber of measurements" is more straightforward for cellular network providers to operate. In practice, given a minimum PDP Context age, we can have the fraction of measurements accordingly so that we can estimate the measurement overhead for collecting **RNCGroundTruth**. Figure 10 shows the accuracy under change over the other two dimensions: number of measurements and cluster size.

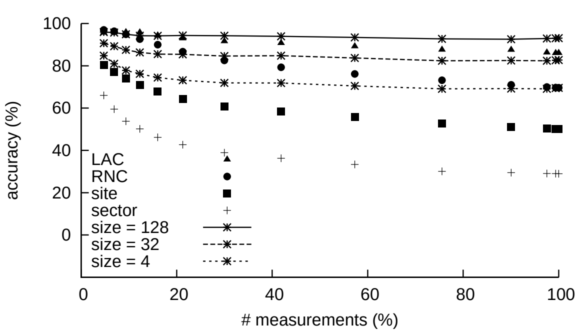


Figure 10: BIGRAPH: # measurements, accuracy, cluster size.

4.3.2 **BIGRAPH**'s *accuracy over short term*

As the evaluation data set of July 22 and training data set of July 21 are too close to each other in time, it is still uncertain how **BIGRAPH** performs if time changes. Note that being accurate over longer time is very necessary for **BIGRAPH** because collecting **RNCGroundTruth** is expensive and building the training data set is supposed to be infrequent consequently. It is more efficient if **BIGRAPH**'s snapshot clusters can keep reasonably accurate for a long time.

Although Figure 4 shows that **BIGRAPH**'s accuracy is still stable over long PDP Context age, we have to investigate **BIGRAPH**'s performance over time even longer than typical PDP Context ages. In practices, it is acceptable that we collect one snapshot of **RNCGroundTruth** once a week. So, in this section, we evaluate **BIGRAPH**'s accuracy over the week of July 21. We compare the similarity of clusters over different days in the week. Similarly to §4.3.1, we still use the same training data set, but have the separated evaluation data sets from July 18th to July 23 (excluding July 21 and July 22). From Figure 11, we can observe consistently high accuracy after we apply the clusters from the training data set on the 4 separated evaluation date sets in that week. When the cluster size is limited to 4, the accuracy can be consistent around 70% over one week.

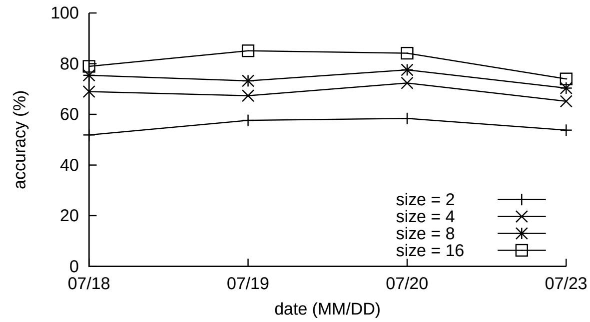


Figure 11: BIGRAPH's accuracy over one week.

4.3.3 **BIGRAPH**'s *accuracy over long term*

In §4.3.2, we observe that **BIGRAPH** is able to perform consistently well over a week. We want to push the time difference between the training data set and the evaluation data set even longer to discover the worst case of how old the clusters can be acceptable after we apply it on **PDPSetupLocations** records.

We collect the data sets of **RNCGroundTruth** and **PDPSetupLocations** again on Dec 5, 2010 and serve them as an evaluation data set, which is 5.5 months later after the training data set on July 21. Then we check the accuracy by applying the clusters of the training data set on the evaluation data set. Figure 12 shows that the accuracy is still reasonable. When the cluster size is 4, **BIGRAPH**'s accuracy is higher than site-level accuracy. Using clusters of 32 sectors, **BIGRAPH** can achieve the accuracy comparable to RNC-level accuracy. Therefore, although **BIGRAPH** requires expensive overhead in **RNCGroundTruth** collection, one snapshot of **RNCGroundTruth** can be still acceptable after several months.

4.3.4 **HANDOVER**'s *accuracy in real time*

Serving as an alternative of **BIGRAPH** on condition that **RNCGroundTruth** is not available, **HANDOVER** compute clusters in real time based on lightweight **HandoverCounters**. Once per hour, **HANDOVER** updates its graph, sparest cuts the graph, and ends up with a set of clusters. **HANDOVER** works differently from **BIGRAPH**, so does the evaluation for the performance of **HANDOVER**.

Figure 13 shows **HANDOVER**'s real-time accuracy. Similar as **BIGRAPH**'s accuracy, **HANDOVER**'s accuracy is the probability that the current locations of a measurement of **RNCGroundTruth**

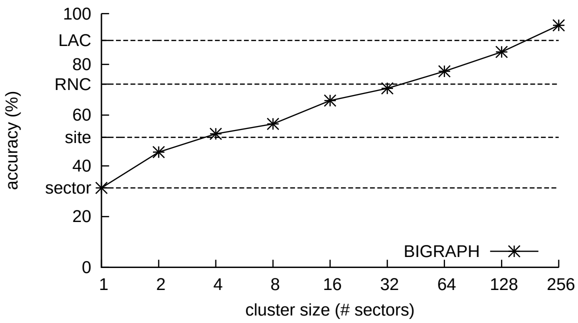


Figure 12: BIGRAPH's accuracy over 5 months.

agrees with the corresponding initial location by **PDPSetupLocations** at the granularity of **HANDOVER**'s clusters. We observe that **HANDOVER** is consistently and significantly better then the site-level accuracy. Also, we can see that the accuracy is always higher in the earlier hours of day, which is probably due to users less movement during the early morning.

Previous Figure 8 shows that **HANDOVER**'s performance is worse than **BIGRAPH**, which is expected because **RNCGroundTruth** in **BIGRAPH** captures more information than **HandoverCounters** in **HANDOVER** such as the start of PDP Contexts. Further confirmed from Figure 9, **HANDOVER**'s clusters have slightly smaller coverage on the number of cell sites than **BIGRAPH**'s clusters.

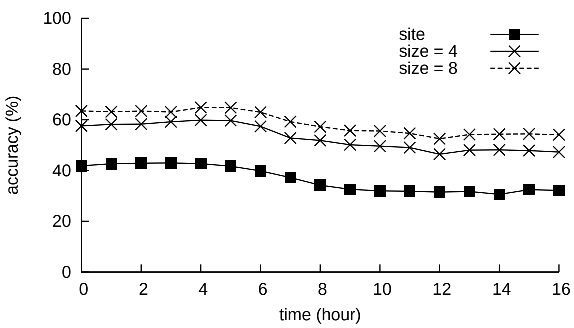


Figure 13: HANDOVER's accuracy over hours.

The impact of cluster size on **HANDOVER** is reflected by previous Figure 8. Same as **BIGRAPH**, the smaller **HANDOVER**'s cluster size is, the more information is lost by the cutting. Figure 8 shows that the accuracy degradation if **HANDOVER** serves as an alternative of **BIGRAPH**. **BIGRAPH**'s accuracy is 20% higher than **HANDOVER**'s when cluster size is from 2 – 8.

4.4 Naïve Heuristics Perform Bad

Besides **BIGRAPH** and **HANDOVER**, we investigate another three solutions as well based on some straightforward heuristics. These solutions may be potential solutions under some extreme cases, such as when **BIGRAPH**'s or **HANDOVER**'s prerequisites at RNCs are totally unavailable.

MAX. This approach is based on either **BIGRAPH**'s bipartite graph or **HANDOVER**'s graph. Before the approach starts, each sector itself is a cluster. Once started, each cluster repeatedly absorbs the outside sector that has the maximum edge cost with the sectors inside this cluster. One cluster stops from growing as soon as it hits the constraint on cluster size. The whole absorbing process stops if all clusters are determined.

CLOSE. It is similar to **MAX**, but does not depend on any knowledge except the GPS location of cell sectors. Since cell sectors are geographic areas directionally covered by cell sites, we use the GPS location of the corresponding cell sites to estimate the geographic coverage of cell sectors. Each cluster repeatedly absorbs the sector with the closest physical distance to it until it hits the restricted the size.

THRE. This approach is based on either **BIGRAPH**'s or **HANDOVER**'s graph. It directly filters out those edges whose weight are smaller than a predefined threshold. The filtering will potentially leave the graph a set of disjoint clusters. The average cluster size, the maximum cluster size, and the accuracy is affected by the threshold. Note that one disadvantage of this approach is the uncertainty on the cluster size: the size could be very different across clusters as shown by

Figure 14. **THRE**'s maximum cluster size is much larger than the average cluster size, which indicates the imbalance of the graph is a natural. Large cluster size increases the difficulty to narrow down performance to fine-grained network locations in practices. Increasing the threshold on the minimum cost can reduce the maximum cluster size, but it results in high inaccuracy as well.

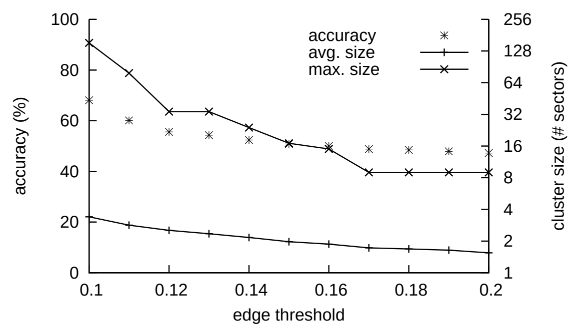


Figure 14: THRE's accuracy and average/maximum cluster size over threshold.

4.5 Pros and Cons

Figure 15 shows the accuracy of all these five solutions under the impact of cluster size. Comparing the accuracy, **BIGRAPH** $>$ **HANDOVER** $>$ **MAX** $\approx$ **CLOSE** $\approx$ **THRE**. When the cluster size is 1, all five solutions have the same accuracy as the sector-level accuracy. **BIGRAPH** performs better than site-level accuracy when its cluster contains more than two cell sectors. With the cluster size of 4, **HANDOVER** can perform as good as site-level accuracy. The accuracy of **MAX**, **CLOSE**, and **THRE** is not sensitive to cluster size. Even if the cluster size is 64, these three solutions still cannot achieve the RNC-level accuracy.

Table 2 summarizes the properties of the five solutions. **BIGRAPH** has the best accuracy, but with the highest measurement overhead. In order to reduce the overhead, we can create snapshot of **RNCGroundTruth** periodically. As we discussed before, one-day snapshot can perform very well without accuracy degradation over one week. So we think **BIGRAPH** is still a stable solution. **HANDOVER** requires the **HandoverCounters** which is lightweight. Since **HandoverCounters** are not captured at GGSNs, it is only hourly collected. We think **HANDOVER** is a more stable solution than **BIGRAPH**. The overhead and stability of **MAX** and **THRE** depend on the input graph, either **BIGRAPH**'s or **HAN-**

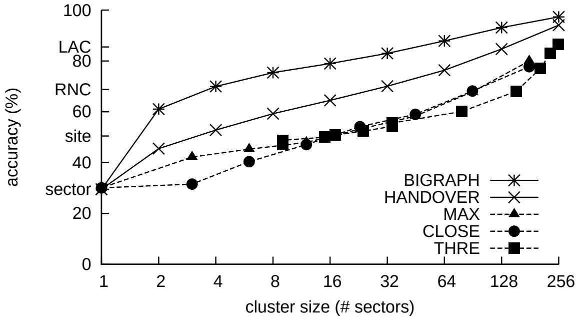


Figure 15: Accuracy of the five solutions.

solution	overhead	accuracy[1]	stability
BIGRAPH	**RNCGroundTruth**	70%	1 week – 5 months
HANDOVER	**HandoverCounters**	53%	real-time
MAX	**RNCGroundTruth/ HandoverCounters**	41%	1 week – 5 months / real-time
CLOSE	sectors' GPS	31%	static
THRE	**RNCGroundTruth/ HandoverCounters**	<48%	1 week – 5 months / real-time
LAC	-	83%	static
RNC	-	67%	static
site	-	51%	static

[1]: the accuracy if the cluster size == 4 is applicable.

Table 2: Pros and Cons of the five solutions.

DOVER's graph. **CLOSE** and site-/RNC-level estimation do not require any dynamic information.

5. PERFORMANCE RE-ASSIGNMENT

Once we have the knowledge of which sectors are related by clustering, we have high confidence on the traversed network elements for performance measurements logged by **IPFlowRecords**. In this section, we want to associate end-to-end performance metrics captured by **IPFlowRecords** to different network elements with high accuracy based on the clusters obtained from **BIGRAPH** or **HANDOVER**.

5.1 Performance Inference

Since the performance metrics captured by **IPFlowRecords** can be assigned to incorrect locations due to localization inaccuracy, our observation of which parts within networks are performing well and which parts have poor performance may be incorrect. We want to leverage the knowledge of related sectors and infer the actual performance of each network element by associating the measured end-to-end performance metrics with all cell sectors in the same cluster.

Measured Performance. **IPFlowRecords** monitors end-to-end performance, *e.g.*, RTT, loss, throughput, *etc.*, at GGSNs for each IP flow. In this measurement infrastructure, although the reported network location by **IPFlowRecords** can be inaccurate, end-to-end performance is comprehensively recorded at GGSNs.

Ground Truth Performance. In order to quantify how inaccurate the measured metrics could be and how much benefit can be obtained from our performance re-assignment, we first have to build a ground truth of the performance of each cell sector over time. Similar to the correlating approach in §4.2.1, we search the correct sector for each measurement in **IPFlowRecords** from **RNCGroundTruth** based on the anonymized subscriber identifer and timestamp. Although the **IPFlowRecords**'s reported sector can be inaccurate, via **RNCGroundTruth**, we can still label the measured metrics to where they are.

Re-assigned Performance. In §4, we have already proposed clustering solutions **BIGRAPH** and **HANDOVER** for discovering related sectors. At time T, let $M(s_1)$ be the measured performance of s_1, $G(s_1)$ be the ground truth performance of s_1, $P(s_1)$ be the re-assigned performance of s_1, and s_2, s_3, and s_4 be the sectors in the same cluster with s_1. We re-assign sector s_1 the performance $P(s_1)$ with the average of $M(s_1)$, $M(s_2)$, $M(s_3)$, and $M(s_4)$. Therefore, for each sector, at any time, we can re-assign performance to the sector based on the measured performance of sectors in its clusters.

To evaluate the benefit of performance re-assignment, we compare the measured RTT and the re-assigned RTT with the ground truth RTT in Figure 16. Figure 16 shows the relative difference from the ground truth. In the re-assigning, we investigate the re-assigned RTT across different forms of clusters, *i.e.*, cell site, **BIGRAPH**'s clusters, **HANDOVER**'s clusters. Because the measured RTT can be considered as the re-assigned RTT when each sector itself is a single cluster, to make our presentation consistent with that in §4, hereafter, we label the measured performance as "sector"-level re-assigned performance in legends.

According to Figure 16, the measured RTT can be very different from the ground truth RTT. More than 20% sectors have the error difference of more than 40%, which encourages us to locate the measured performance better. After the re-assignment, the difference from the ground truth RTT becomes smaller. The median RTT difference decreases by 10% from 16%. Using **BIGRAPH**'s or **HANDOVER**'s clusters, the RTT difference is very close to the one using cell sites, but both **BIGRAPH**'s and **HANDOVER**'s clusters have only 4 clusters while the average size of cell sites is larger.

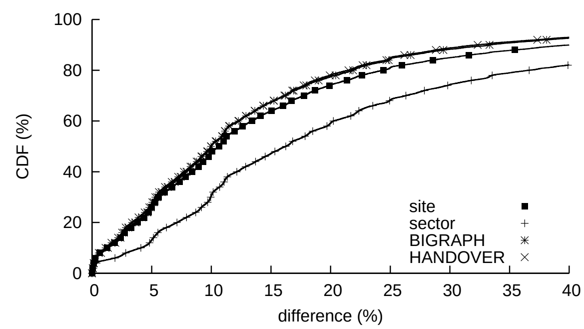
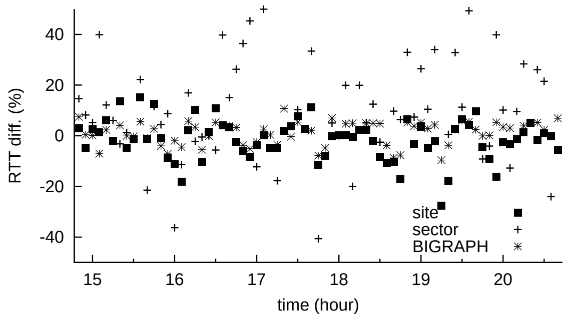


Figure 16: Difference across the re-assigned, measured, and ground truth performance.

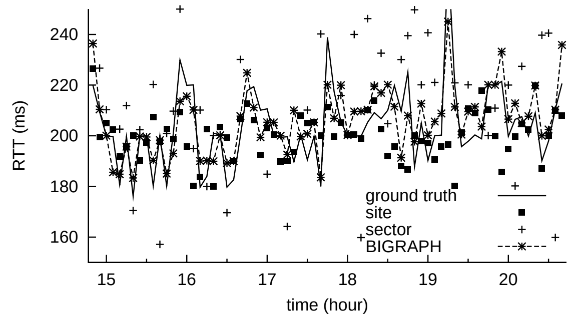
5.2 Anomaly Detection

The relative difference of RTT is only one dimension for evaluating the performance re-assignment. How accurate the re-assignment reacts to the performance change is of particular interest as well because monitoring exceptional performance increases or decreases is one way for cellular operators to detect anomalies.

Figure 17 shows an example that re-assignment does well in capturing RTT spikes by comparing the re-assigned RTT and the measured RTT with the ground truth one over a few hours. Figure 17(a) depicts the relative RTT difference over time, which confirms the observation from Figure 16: before the re-assignment, the relative difference has many outliers larger than 40%, but the RTT is mostly within +/-20% ground truth RTT after the re-assignment.



(a). Relative difference of RTT



(b). Absolute RTT

Figure 17: How the re-assigned performance reacts to performance changes over time.

Instead of the relative difference, Figure 17(b) compares the absolute RTT over Figure 17(a). Many times, RTT spikes in ground truth is well captured by the re-assigned RTT based on **BIGRAPH**'s clusters, but failed by the measured performance and the re-assigned one based on cell sites. The measured performance is erratic and may potentially have significant inaccuracy. Hereafter, to ease reading, we do not include the re-assigned performance based on **HANDOVER**'s clusters if it has similar patterns to **BIGRAPH**.

Given that the re-assigned performance is fairly close to the ground truth and also captures the RTT spikes well, we expect it can play an important role in anomaly detection in practice where anomaly alerts happen when the performance observed in real time drops or increases significantly, *e.g.,* 25%. Assuming X is a reasonable threshold to determine anomalies, if we observe the performance is above 100% + X or below 100% - X of the last performance update, we consider some anomaly has occurred. Then we quantify and compare the false positive and false negative of using the re-assigned performance and the measured performance for anomaly detection. Note that the X is the threshold for the ground truth performance, but for the re-assigned and the measured ones, we can tune the threshold for each in practices.

Figure 18 illustrates the false positive/negative over the threshold of using re-assigned performance to detect performance anomalies. In terms of false positives, using cell sites and **BIGRAPH**'s clusters to re-assign the performance have similar false positive, but **BIGRAPH**'s clusters have slightly lower false positives. In terms of false negatives, both the re-assigned performance based on **BIGRAPH**'s clusters and the measured performance have the lowest false negative, which is significantly lower than the false negative of the performance re-assigned based on cell sites. The re-assigned performance based on **BIGRAPH**'s clusters is much better than the one re-assigned based on cell sites although both are based on related sectors. It is because **BIGRAPH**'s clusters can locate subscribers more accurately and leverages the human mobility patterns better consequently.

In summary, re-assigning the performance based on **BIGRAPH**'s clusters can achieve the lowest false positive and false negative.

6. GENERALIZABILITY

The split between the radio network where fine-grain location information is known (RNCs and base stations) and the core network where IP-level metrics are easily collected (SGSNs and GGSNs) is fundamental in UMTS networks. This is because the RNC is designed to handle all device mobility. While infrequent sector information is reported to the core network when a device moves into a new location area and for accounting purposes (e.g., those in CDR records used for billing), this information is typically similar to the accuracy of that in PDP Context Setup messages: they may only be able to distinguish an RNC or LAC (which can cover hundreds of sectors in a metro area). Partly, this is because it is not efficient to propagate all the cell switch signaling information to the core network and the radio network control and date planes are coupled together. HSPA devices (the majority of smartphones and data cards today) may change their serving cell every 2 – 20 ms in order to prevent degradation due to fast-fading in the wireless channel.

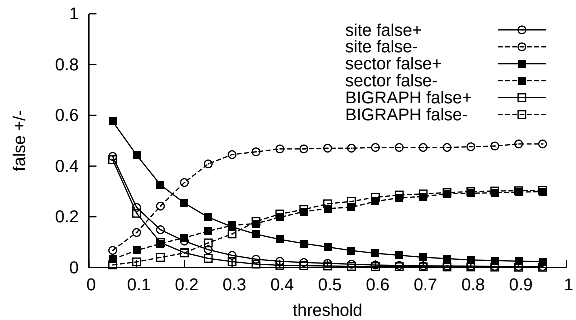


Figure 18: Anomaly detection via the re-assigned performance.

The deployment of Long Term Evolution (LTE) networks may alleviate the problem discussed in this paper to some extent. In LTE networks, the control plane signaling that was performed by the RNC in UMTS is instead performed by a new entity called the Mobility Management Entity (MME). User data, rather than being routed through the MME, is sent directly from the base station to the core network. This is made possible by LTE's all-IP architecture and a decoupling of the radio network control and data plane. Thus, IP measurements collected in an LTE core network can, in principle, be easily correlated with the base station that originated the traffic. Nonetheless, sector level information is still not available to the core network and, as we have seen in our study, related sectors may not belong to the same base station. Furthermore, LTE core networks are still logically hierarchical like UMTS networks: base stations send data to a Serving Gateway which in turn forward it it to a Packet Data Network Gateway before it exits to external networks such the Internet. Thus, if the Serving Gateways are distributed physically, the cost of deploying IP measurement infrastructure at all ingress points of the core network may still be prohibitive. For example, in the UMTS network we studied in this paper, the number of physical locations housing GGSNs is two orders of magnitude smaller than the number of physical locations housing RNCs so that signaling and handover latencies are acceptable. It remains to be seen how large-scale LTE networks will be deployed.

7. RELATED WORK

Studies of human localization have strong motivations yielding insights into our society from perspectives such as urban planning, traffic forecasting, *etc.* [11, 22]. Particularly, in the mobile computing community, localization is critical in designing content delivery services, context-aware applications, *etc.* [16, 17, 21].

In location-aware studies, one critical issue is how to collect the ground truth for end-user locations. Some positioning systems such as GPS and SkyHook [1] are among the most well-known solutions for mobile users. However, these solutions do have limitations, *e.g.,* GPS is not necessarily available for all mobile devices, and SkyHook does not work well for tracking cellular users [5]. Moreover, getting GPS information requires user permission most of the time and cannot work for indoor localization. There are some localization techniques targeting individual users [2, 4, 6], while cellular network operational records are very helpful in investigating a large and representative set of cellular users [11, 3, 13, 23, 12].

Overall, previous location-aware studies can be classified into three categories:

Characterization. Characterizing the relationship between a user's interests and his mobility properties, *e.g.,* the real-time location, is critical for location-based services. Based on the IP flow traces from a cellular operator, Keralapura *et al.* investigated if there exists distinct behavior patterns among mobile users in web browsing behavior [13]. Trestian *et al.* characterized the relationship between user application interests and their mobility properties [23]. Via extensive field tests under different movement scenarios, Tso *et al.* presented an empirical study on the performance of mobile HSPA networks in Hong Kong [24]. Using aggregate statistics of hundreds of thousands of subscribers, Isaacman *et al.* demonstrated different mobility patterns in Los Angeles and New York City [12].

Modeling. Context-aware applications can benefit from location forecasting if there exists mobility patterns for individual users and the future locations are predictable from their mobility models. Kim *et al.* presented a method to estimate the physical location by associating it with access points in a wireless network [15]. Eagle *et al.* monitored the cell towers transitions on hundreds of subjects and built a network for cell towers. Through clustering on cell towers and discovering strongly related cell towers, they can predict each subject's subsequent movement [18].

More than merely modeling the mobility patterns accurately, the overhead of building the mobility patterns has been considered by Sohn *et al.*. [22]. They explored how coarse-grained GSM data from mobile phones can be used to discover high-level properties of user mobility properties.

Application. Achieving accurate localization is the prerequisite for location-based services. Among the studies related to providing such services, Nicholson *et al.* and Pang *et al.* generated the connectivity forecasts beyond the location prediction [19, 21]. Taking energy saving into consideration, Ananthanarayanan *et al.* proposed the solution of associating energy efficient bluetooth hotspots with the locations of WiFi access points to enhance WiFi performance.

Compared with all these studies, our study has the following two major differences. First, from the perspective of cellular operators, we focus on the performance metrics in cellular networks. Therefore, we are interested in accurately localizing subscribers to their correct network locations, rather than their physical locations. Second, designed for cellular operators, our system has to reduce the overhead for characterizing mobility patterns of all users in the target network. We face the challenge due to the inherent trade-off between the fine-grained measurement and measuring overhead.

8. CONCLUSIONS

Our work is motivated by the important problem faced by cellular network operators to identify which parts of their 3G networks are performing badly ad the radio links between base stations and cellular devices heavily impact the end-to-end performance.

Our work is the first to quantify the location inaccuracy for mapping the IP-level flow records at GGSNs to different fine-grained aggregation levels, and leverage human mobility patterns to achieve better performance localization. We developed clustering algorithms to group related clusters together, re-assigned the measured performance metrics to more accurately reflect the actual ones, and eventually captured performance anomalies effectively with lower false positive and false negative rates compared with other forms of clusters.

During the clustering, in order to balance the tradeoff between localization accuracy and measurement overhead, we proposed two solutions for clustering related sectors. One solution can locate a subscriber into only 4 cell sectors with the accuracy of 70% over 5 consecutive days, while the other one can update the knowledge of mobility patterns hourly, and increase the accuracy by at least 20% compared to solutions such as clustering purely by cell sites.

We believe that our work can be an important utility for cellular operators in for the purpose of performance monitoring, network maintenance, and anomaly detecting.

9. REFERENCES

[1] Skyhook. http://www.skyhookwireless.com.

[2] G. Ananthanarayanan and I. Stoica. Blue-Fi: Enhancing Wi-Fi Performance using Bluetooth Signals. In *Proc. ACM MOBISYS*, 2009.

[3] N. Antunes, C. Fricker, P. Robert, and D. Tibi. Metastability of CDMA cellular systems. In *Proc. ACM MOBICOM*, 2006.

[4] M. Azizyan, I. Constandache, and R. R. Choudhury. SurroundSense: Mobile Phone Localization via Ambience Fingerprinting. In *Proc. ACM MOBICOM*, 2009.

[5] M. Balakrishnan, I. Mohomed, and V. Ramasubramanian. Where's that Phone?: Geolocating IP Addresses on 3G Networks. In *Proc. ACM SIGCOMM IMC*, 2009.

[6] N. Banerjee, S. Agarwal, P. Bahl, R. Chandra, A. Wolman, and M. Corner. Virtual Compass: Relative Positioning to Sense Mobile Social Interactions. In *Proc. IEEE PERVASIVE*, 2010.

[7] J. But, U. Keller, and G. Armitage. Passive TCP Stream Estimation of RTT and Jitter Parameters. In *Proc. IEEE LCN*, 2005.

[8] B. Claise. Cisco Systems NetFlow Services Export Version 9. IETF RFC 3954, 2004.

[9] C. Cranor, T. Johnson, O. Spatscheck, and V. Shkapenyuk. Gigascope: A Stream Satabase for Network Applications. In *Proc. ACM SIGMOD*, 2003.

[10] A. Gerber, J. Pang, O. Spatscheck, and S. Venkataraman. Speed Testing without Speed Tests: Estimating Achievable Download Speed from Passive Measurements. In *Proc. ACM SIGCOMM IMC*, 2010.

[11] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi. Understanding Individual Human Mobility Patterns. 453(7196):779–782, June 2008.

[12] S. Isaacman, R. Becker, R. CÃąceres, S. Kobourov, J. Rowl, and E. Varshavsky. A Tale of Two Cities. In *Proc. Workshop on Mobile Computing Systems and Applications (HotMobile)*, 2010.

[13] R. Keralapura, A. Nucci, Z.-L. Zhang, and L. Gao. Profiling Users in a 3G Network Using Hourglass Co-Clustering. In *Proc. ACM MOBICOM*, 2010.

[14] B. W. Kernighan and S. Lin. An Efficient Heuristic Procedure for Partitioning Graphs. *Bell Systems Technical Journal*, 49, 1970.

[15] M. Kim, D. Kotz, and S. Kim. Extracting a mobility model from real user traces. In *Proc. IEEE INFOCOM*, 2006.

[16] Z. M. Mao, C. D. Cranor, F. Bouglis, M. Rabinovich, O. Spatscheck, and J. Wang. A Precise and Efficient Evaluation of the Proximity between Web Clients and their Local DNS Servers. In *Proc. USENIX Annual Technical Conference*, 2002.

[17] S. Mathur, T. Jin, N. Kasturirangan, J. Ch, W. Xue, and M. Gruteser. ParkNet: Drive-by Sensing of Road-Side Parking Statistics. In *Proc. ACM MOBISYS*, 2010.

[18] J. Q. N. Eagle, A. Clauset. Methodologies for Continuous Cellular Tower Data Analysis. In *Proc. IEEE PERVASIVE*, 2009.

[19] A. J. Nicholson and B. D. Noble. BreadCrumbs: Forecasting Mobile Connectivity. In *Proc. ACM MOBICOM*, 2008.

[20] J. Pahdye and S. Floyd. On inferring TCP behavior. In *Proc. ACM SIGCOMM*, 2001.

[21] J. Pang, B. Greenstein, M. Kaminsky, D. Mccoy, and S. Seshan. WiFi-Reports: Improving Wireless Network Selection with Collaboration. In *Proc. ACM MOBISYS*, 2009.

[22] T. Sohn, A. Varshavsky, A. Lamarca, M. Y. Chen, T. Choudhury, I. Smith, S. Consolvo, J. Hightower, W. G. Griswold, and E. D. Lara. Mobility Detection Using Everyday GSM Traces. In *Proc. ACM UBICOMP*, 2006.

[23] I. Trestian, S. Ranjan, A. Kuzmanovic, and A. Nucci. Measuring Serendipity: Connecting People, Locations And Interests In A Mobile 3G Network. In *Proc. ACM SIGCOMM IMC*, 2009.

[24] F. P. Tso, J. Teng, W. Jia, and D. Xuan. Mobility: A Double-Edged Sword for HSPA Networks. In *Proc. ACM MOBIHOC*, 2010.

[25] Q. Xu, J. Huang, Z. Wang, F. Qian, A. Gerber, and Z. Mao. Cellular Data Network Infrastructure Characterization and Implication on Mobile Content Placement. In *Proc. ACM SIGMETRICS*, 2011.