Predicting Handoffs in 3G Networks

Umar Javed, Dongsu Han, Ramon Caceres†,
Jeffery Pang†, Srinivasan Seshan, Alexander Varshavsky†
Carnegie Mellon University, †AT&T Labs - Research

1. INTRODUCTION

Cellular data networks have recently seen an explosion in their usage due to the widespread deployment of 3G technologies and the rapid proliferation of smartphones. People are increasingly using their smartphones on the go and expect always-on, high quality connectivity at all times.

A key network primitive that enables continuous connectivity in cellular networks is *handoff*, or the transfer of a device's connection from one cell sector to another. Although handoffs are necessary for mobile devices to maintain connectivity, a recent study [8] showed that handoffs generally cause short-term disruptions in application performance. Similarly, simulation studies [2] have shown that handoffs could degrade performance of real-time applications such as VoIP. Moreover, some networks are prone to making handoff decisions suboptimally [8], incurring overhead for both applications and the infrastructure when performing unnecessary handoffs.

Given the negative impact of handoffs on application performance, applications could benefit from the ability to predict impending handoffs with reasonable accuracy, and modify their behavior to counter the performance degradation that accompanies handoffs. Recent work [6] has shown that handsets can make effective use of similar performance predictions to reduce energy consumption. In addition, the network infrastructure could utilize the ability to predict conditions that are likely to require handoffs to improve handoff decisions and resource allocation. However, predicting future handoffs is challenging because they are a function of unforeseeable attributes such as user mobility and changes in the environment.

In this paper, we study predictability of handoffs based on the data from a major US cellular provider's radio network. We start by showing that attributes that reflect recent mobility history, connected cell density, and signal strength variation are correlated with future handoff rates (§4). Then, we develop a two-phase machine learning framework that uses a combination of these attributes to predict the occurrence and the frequency of handoffs in the near future (§5).

Our evaluation on a large number of real handoff traces shows promising results. Using information available at either the handset or the network infrastructure, we are able to achieve 80% accuracy in predicting the occurrence of a handoff in the near future – much better than the 53% accuracy achieved by a predictor. Overall, we believe that our analysis provides a promising start towards a practical framework that handsets could use to accurately predict handoffs and better tolerate the performance disruptions that can accompany them.

2. HANDOFFS IN UMTS NETWORK

We consider handoffs in a UMTS (Universal Mobile Telecommunications System) network, the most common 3G standard in the world. In a UMTS network, a user device connects to a set of *cells*. A cell is defined by the area covered by a single antenna on a physical basestation (typically a few km^2) and its frequency (e.g., 850Mhz or 1900Mhz). Each basestation typically has 3-6 cells, and is managed by a Radio Network Controller (RNC). A single RNC controls tens to hundreds of basestations. In order to maintain IP connections while users are mobile, all IP traffic from a device is tunneled through the RNC to the core network.

At any point in time, a user device may be within the coverage area of multiple cells. When a device begins transmitting or receiving data, it establishes a connection with one or more of these cells, typically the ones with a sufficiently high signal-to-noise ratio (SNR). The set of cells a device is connected to, which must all be on the same frequency, is called the *active set*. In the network we examine, the size of the active set is limited to 4. While connected, cells may be added or removed from the ac-

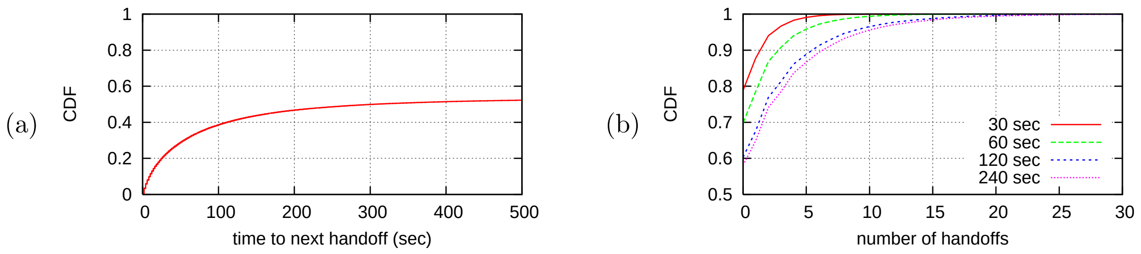

Figure 1: Distributions of (a) time to next handoff over all times in all traces, and (b) number of handoffs in four time interval sizes.

tive set as their SNRs change. This process is called *soft handover*. In addition, it may disconnect from all cells in the active set and instead connect to cells on a different frequency. This event is called *inter-frequency handover*. At each point in time, only one cell in the active set, called the *serving cell*, will actually transmit data to the device.[1] This is typically the cell with the highest SNR. When data ceases to be sent or received for a short time period (typically a few seconds), the device disconnects from all cells.

In this paper, we define a ***handoff*** to be a change in a device's serving cell, as this change physically alters a device's network path and, thus, can impact performance. It also involves signalling overhead. Note that a handoff is distinct from the process of soft handover, which may add or remove cells from a device's active set without actually changing the serving cell. However, handoff includes all inter-frequency handover events. Finally, note that there can be no handoffs when a device is idle, as it does not have any serving cell. Handoff decisions are based on a deterministic function of recent SNR measurements reported by the device and the load on each cell [7]. While the network can determine whether a handoff will occur at the current time instant, future handoffs are affected by less predictable factors such as user mobility and environmental changes, which affect channel quality.

3. DATA SET

To analyze the predictability of handoffs of real users, we collected 1 day of anonymized event logs from several RNCs in a major U.S. cellular operator in May 2011. These RNCs control a significant fraction of the base-stations in a large U.S. city. We analyze events from three logs the RNC maintains: the ServCellLog, which records the serving cell and active set for each device every two seconds when the device is active; the SNRLog, which records the SNR values of all cells measured by a device in each measurement report sent to the RNC; and the SOHOLog, which records the soft handover events (i.e., additions and removals from each device's active set). Each log record is timestamped and devices are anonymously identified by an irreversible hash of the device's IMSI, which is unique per SIM card. All device and subscriber identifiers are anonymized to protect privacy without affecting the usefulness of our analysis. Furthermore, the data set does not permit reversing the anonymization or re-identification of subscribers.

For each device, the aforementioned events are captured when its radio is in the HSPA active state (i.e., in the DCH state described in [4]). We define a *trace* to be the sequence of events from a device for a contiguous active time. We determine a handoff occurred in a trace whenever the serving cell changes between two consecutive events in ServCellLog. To eliminate boundary effects of short traces, we concentrate on long traces and, thus, filtered out all traces that were shorter than 5 minutes. In total, we analyzed 5,000 traces. In these traces, we saw about 60,000 handoff events between approximately 1,200 distinct cells.

We note that although we analyze handoffs from events seen at RNCs, these events could be made available to the applications running on a device as well. Some events are already exposed via smartphone APIs (e.g., SNR values), while we believe others could be made available from cellular device drivers. This paper motivates making this information available via API.

To illustrate properties of handoffs in our dataset, Figure 1(a) plots the cumulative distribution function (CDF) of the time to next handoff for a random sample in our traces. The CDF levels off at about 55% because 45% of the points either fall within a trace that has no handoffs or fall after the last handoff in a trace. This plot suggests the following

[1] HSPA devices, which generate the vast majority of traffic in the UMTS network we examine, have a single serving cell, but older devices receive data from all cells in the active set simultaneously.

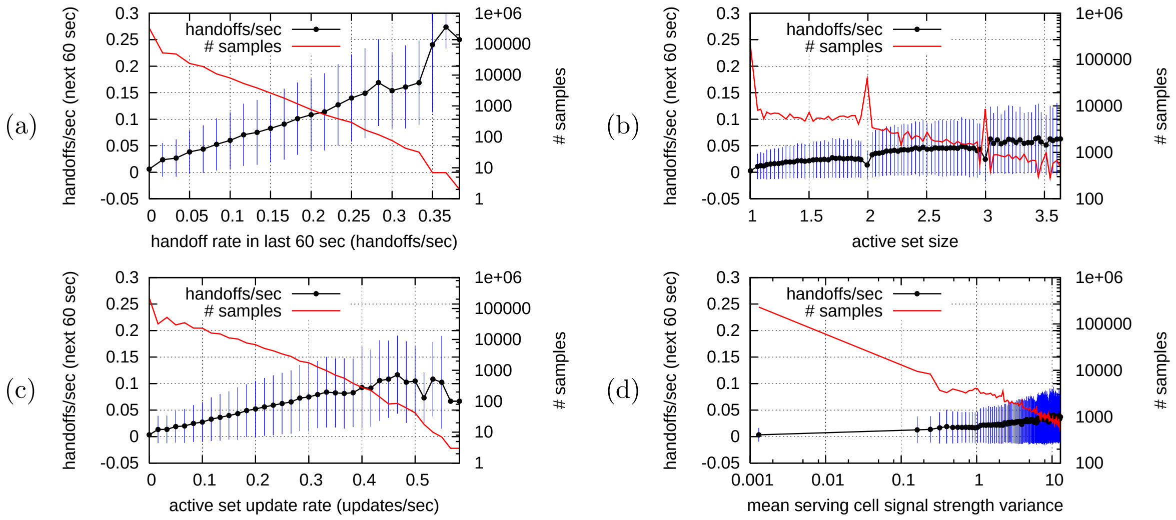

Figure 2: The handoff rate in the next 60 seconds compared with (a) the handoff rate in the past 60 seconds, (b) the size of the active set, (c) the rate of updates to the active set, and (d) the mean signal strength of the serving cell. For clarity, we quantize the samples into groups and plot the mean and standard deviation of each group.

naive handoff predictor: estimate the likelihood of a handoff occurring within t seconds as the percentile that corresponds to t in the plot. For instance, the naive predictor would estimate the likelihood of a handoff occurring within the next 30 seconds to be always 22% and within 200 seconds to be 47%.

Figure 1(b) plots the CDF of the number of handoffs in all 30, 60, 120 and 240 second time intervals in our traces. The figure shows that there is no handoff in about 80%, 70%, 60%, and 60% of 30, 60, 120, and 240 second intervals, respectively. On the other hand, some intervals have many handoffs. For example, 1% of 30 second intervals have more than 5 handoffs. Thus, applications may want to predict not only *whether* a handoff will occur, but also the *frequency* of handoffs.

4. DISCRIMINATIVE ATTRIBUTES

The key to predicting future handoffs is uncovering attributes that have a strong correlation with future handoffs rates. In this section, we describe four attributes that we found to have a correlation with future handoffs, namely *past handoff rate*, *size of the active set*, *active set update rate* and *signal strength variation*. Each of these attributes could be obtained both on the handset and at the RNC, making our prediction technique useful for both application developers and network operators.

To show the correlations, we averaged an attribute value over a historical time window of h seconds and plotted it against the handoff rate of the next t seconds. Although we experimented with various historical and future window sizes, we show results for only $h = 60$ seconds and $t = 60$ seconds due to space limitations. The correlations remained similar for other choices of h and t as well.

For each plot in Figure 2, we grouped data points into clusters based on the attribute values and plotted its mean future handoff rate with its standard deviation. We also plotted the size of each group on the second y axis. We next describe each of the attributes in detail.

4.1 Past Handoff Rate

We hypothesized that handoff rates of the recent past is a good predictor of handoffs in the near future. This is because in short time scales one's mobility pattern is likely to stay the same. Figure 2 (a) depicts the correlation between the past handoff rate and the future handoff rate. The figure shows that the correlation between the past and future handoff rates is quite pronounced (correlation coefficient of 0.56), which indicates that historical handoff rates are potentially a strong predictor. As we show in § 5, this is indeed the case.

4.2 Size of the Active Set

Recall that an active set contains up to 4 cells that are above a certain signal strength threshold. A large number of cells in the active set implies more options in choosing a serving cell, and thus a stronger potential for handoffs.

Figure 2 (b) shows the correlation between the size of the active set and the future handoff rate. Although the figure shows a positive correlation (correlation coefficient of 0.39), there are two factors that reduce the effectiveness of the active set size at predicting future handoff rates: large standard deviations and the dips that occur when the active set size is 1, 2 and 3. Large standard deviations are the result of the active set sizes not reflecting the mobility of the handset. The dips that occur at integral values are due to a large number of stationary users that fall within these three groups. This is because stationary users tend to have a constant active set size and no handoffs. We verified that groups that correspond to integral set sizes indeed contain a large fraction of points with no handoffs.

4.3 Active Set Update Rate

Changes in the active set, such as an addition or a removal of a cell, may be good indicators of mobility and, consequentially, of future handoff rates. Figure 2 (c) shows a positive correlation between the mean active set update rate and the future handoff rate (correlation coefficient of 0.5). The decrease in correlation when active set update is greater than 0.48 is due to the small sample size (less than 100 points per group) as shown on the second y axis.

Note that changes to the active set may be caused by removal and additions of cells with similar signal strengths due to signal strength fluctuations. These changes may not lead to future handoffs. Indeed, in § 5, we show that using active set update rates results in relatively high false positive rates.

4.4 Signal Strength Variation

Since the instantaneous handoff decisions made by the RNC depend upon the signal strength fluctuations of the cells in the active set, there might be a correlation between recent fluctuations in the signal strength of the serving cell and the future handoff rate. Figure 2 (d) plots the correlation between the mean serving cell signal strength variance and the future handoff rate. Unfortunately, the correlation is relative small (correlation coefficient of 0.18). Indeed, as we show in § 5, this attribute turned out to be the worst predictor of future handoff rates. This may be due to the fact that the variance of the signal strength of the serving cell is irrelevant unless the signal strength falls below a certain threshold.

5. PREDICTING HANDOFF BEHAVIOR

This section describes our handoff prediction algorithm and its evaluations. We first present our prediction algorithm, which is divided into two parts: predicting the occurrence of a handoff and its frequency. Then, we answer four questions related to the accuracy of the predictor:

1. What's the accuracy of our occurrence predictor? (§5.2.1)
2. How much history do we need for accurate prediction? (§5.2.2)
3. How sensitive is the accuracy to the future window size? (§5.2.3)
4. What's the overall accuracy of our predictor including the frequency predictor? (§5.2.4)

5.1 The Learning Algorithm

Handoff predictions are made over a future time window of a certain length. We use the features described in §4 to train a two-stage cascading predictor. The first level decides whether there is an impending handoff in the time window considered or not (occurrence predictor). If the window is large enough such that more than a few handoffs could occur, we apply the second binary predictor which classifies each handoff period as a period of high or low handoff frequency, which we define in §5.2.4 (frequency predictor).

Both our prediction classifiers use AdaBoost [1] with decision stumps for the training phase. We have chosen to use boosting since it has been shown to work well for a variety of classification tasks [9, 5]. We tried other approaches, such as Logistic Regression and naive heuristic-based methods, and found that AdaBoost provides the best results. We also obtained results with a single multiclass classifier (no handoff, high frequency handoff, low frequency handoff) instead of our two-stage predictor and found that the latter results in higher accuracy.

5.2 Evaluation

For evaluation, we trained our predictor on one-fourth of the data with 5-fold cross-validation[2] and used the remaining for testing purposes.

5.2.1 Predicting the Occurrence

We first test the accuracy of the handoff occurrence predictor. We use three metrics for accuracy: 1) *overall accuracy*, the percentage of test examples predicted correctly, 2) *false negative rate*, the percentage of actual handoff events in the test set wrongly classified as non-handoffs, and 3) *false positive rate*, the percentage of actual non-handoffs mistakenly predicted as handoffs.

Table 1 shows the performance of our **combined**

[2] In k-fold cross-validation, the training data set is partitioned into k parts, and k training and testing iterations are performed, each time using $k-1$ partitions used for training and one used for testing. The final classifier is the average of these k runs.

predictor along with other simple predictors over a future window of 60 seconds. Specifically, the table shows the prediction performance of each attribute when it is the only feature used in the learner as well as the performance of a naive predictor from § 3. Each attribute is computed over 10, 30, and 60 seconds of history to best capture temporal variations (we explain this choice in §5.2.2).

Attribute	Accuracy	False Positive	False Negative
Past handoff rate	79.5%	11.14%	42.0%
Mean active set size	76.1%	11.5%	51.0%
Active set update rate	76.5%	15.3%	43.7%
Signal strength variance	75.7%	13.4%	49.5%
Combined predictor	80.3%	11.6%	38.3%
Naive predictor	53.4%	31.1%	69.0%

Table 1: Prediction performance of individual features, combined features and the naive predictor.

Table 1 shows that all of the attributes result in reasonable accuracy when used alone in the predictor, verifying our intuition about their predictive power. We also see that the combined classifier improves upon the overall accuracy and the false negative rate (% of actual handoff events missed). The combined predictor is also much more accurate than the naive heuristic-based predictor. Note that the false negative rate in all cases is much higher, and overall accuracy is biased towards the false positive rate. This is because handoff events are infrequent. For example, Figure 1 shows that around 70% of data points have no handoffs in the next 60 seconds.

Reducing False Negatives: A high overall accuracy may not be sufficient for a handoff predictor. For example, a streaming application may be able to tolerate false handoff predictions by using a little extra buffering but may not be able to tolerate missed handoffs because unexpected disruptions may cause a stall in the stream. In other words, the application may want to minimize the false negative rate possibly at the cost of a higher false positive rate.

We can reduce the false negative rates shown in Table 1 by assigning a higher cost to false negatives than false positives through Asymmetric Boosting [3]. In particular, we show results when the cost of a false negative is twice that of a false positive[3].

Table 2 shows that this *cost-sensitive prediction* results in a large reduction in the false negative rate at the expense of a modest increase in the false positive rate and a slight decrease in overall accuracy. It also shows that some attributes respond more sharply to the cost-sensitive predictor than others. For example, the false negative rate for **mean active set size** dropped precipitously from 51% to 19% with the introduction of the asymmetric cost values. In contrast, **past handoff rate** still has a higher false negative rate (30%) than its false positive rate (17%). In other words, it is conservative in predicting handoff occurrences even after cost-sensitivity is introduced. We conjecture that this is due to the relatively smaller variation in the **past handoff rate** values, as shown in Figure 2 (a). Attributes with high variance would be more responsive to cost-sensitive prediction and hence more aggressive in predicting the handoff occurrence.

Attribute	Accuracy	False Positive	False Negative
Past handoff rate	79.0%	17.0%	30.4%
Mean active set size	72.6%	31.1%	18.8%
Active set update rate	71.7%	32.4%	18.8%
Signal strength variance	71.8%	32.2%	19.0%
Combined predictor	78.1%	21.8%	22.0%

Table 2: Prediction performance of individual features and the combined predictor with a cost-sensitive classifier.

5.2.2 *How Much History is Needed?*

Each attribute discussed in §4 is computed over one or more recent time windows. Predicting based on more history can be beneficial because it will better capture temporal variations. However, the more history needed, the longer a handset will have to wait to make accurate predictions after initiating activity. Thus, we want to find the the minimum history needed for good predictive power.

To determine how much history is needed, Figure 3 plots prediction accuracy for the next 60 seconds (without the cost-sensitive modification) as we increase the amount of history used. To ensure that more history never reduces predictive power, each marked x-axis point x represents a predictor that uses attributes computed using x seconds of history *in addition to* attributes computed using x' seconds of history for all marked x-axis points $x' < x$. For example, the point at $x = 60$ shows the accuracy of a predictor using attributes computed over 5, 10, 30 and 60 second history windows. Figure 3 shows that using more history improves accuracy primarily due to a reduction in false negatives. This is because observing slightly more history is much more likely to accurately measure the true **past handoff rate** and hence capture the short-term mobility pattern better. Nonetheless, it is clear that the performance for all three error metrics doesn't improve much beyond 60 seconds of history. Hence, we use attributes computed on 10, 30, and 60 seconds of history in our classifier.

5.2.3 *Predicting over Different Time Windows*

[3] In general, we found that higher values caused convergence issues and lower values led to worse false negative rates.

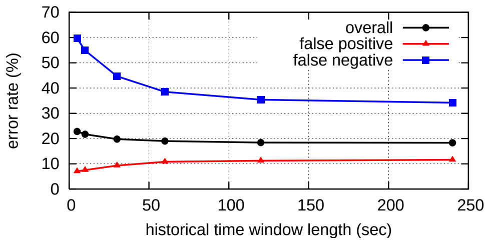

Figure 3: Handoff predictor performance as the number of historical windows is increased.

Since different applications may want to forecast the likelihood of handoffs over different time intervals, we evaluate prediction accuracy over several different prediction time windows in Figure 4. Each point (x, y) on an error curve shows the prediction error y in predicting whether there is going to be a handoff within the next x seconds. Also plotted in dotted curves are the error rates for the naive predictor. The results clearly show that our predictor is more accurate than the naive predictor for all prediction windows.

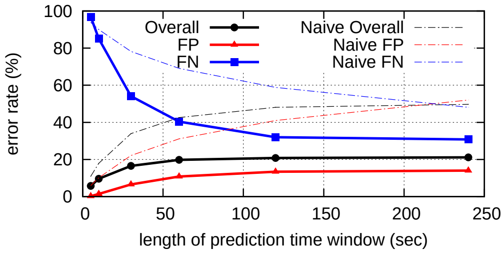

Figure 4: Handoff predictor performance as the prediction window length is increased.

5.2.4 *Predicting Handoff Frequency*

As we observed in §3, some time intervals have many more handoffs than others, so it is also useful to predict the *frequency* of handoffs. For example, a streaming application may benefit by sizing its buffer based on the frequency of potential disruptions. Hence, we implement a second-stage classifier that takes positive handoff predictions from the first-stage and classifies them into low- or high-frequency. The threshold for a high-frequency period is the median non-zero number of handoffs during the period for the entire dataset. For example, from Figure 1 (b), for the 60 second interval, 4 or more handoffs is considered high frequency.

Table 3 shows the results for this classifier for the next 60 seconds. It shows a large majority of intervals are predicted correctly (94% of low-frequency and 77% of high-frequency intervals). We also note that almost all (98.5%) of the false positives from the first-stage classifier (not shown) are classified into low-frequency by the second-stage, so adding this stage does not compound those errors.

		Actual	
		Low Freq.	High Freq.
Predicted	Low Freq.	**94.5%**	5.5%
	High Freq.	23.3%	**76.7%**

Table 3: Predicted vs actual handoff frequencies for the next 60 seconds. Bolded cells are correct predictions.

6. CONCLUSION AND FUTURE WORK

Predicting handoff is important for mobile applications and wireless network operators. This paper explored the possibility of accurately predicting future handoffs using a simple set of attributes that are readily available at the handset as well as at network. To this end, we analyzed several attributes that intuitively reflect the user mobility and the environment. Based on a large scale analysis, we showed that it is possible to predict impending handoffs with almost 80% accuracy.

We believe that this is a significant step towards understanding and predicting the behavior of individual handsets. We envision two important future works. We plan to study how mobile applications might use this ability to predict handoffs to enhance user performance or throughput. Also, we plan to study what kind of benefits would network providers get if they utilize such information.

7. REFERENCES

[1] Y. Freund. Schapire: Experiments with a new boosting algorithm. In *Proc. ICML*, 1996.

[2] P. Lunden, J. Aijanen, K. Aho, and T. Ristaniemi. Performance of VoIP over HSDPA in mobility scenarios. In *Proc. IEEE Vehicular Technology Conference*, 2008.

[3] H. Masnadi-shirazi and N. Vasconcelos. Asymmetric boosting. In *Proc. ICML*, 2007.

[4] F. Qian, Z. Wang, A. Gerber, Z. M. Mao, S. Sen, and O. Spatscheck. Characterizing radio resource allocation for 3G networks. In *Proc. IMC*, 2010.

[5] R. E. Schapire and Y. Singer. Boostexter: A boosting-based system for text categorization. In *Machine Learning*, pages 135–168, 2000.

[6] A. Schulman, V. Navda, R. Ramjee, N. Spring, P. Deshpande, C. Grunewald, K. Jain, and V. N. Padmanabhan. Bartendr: a practical approach to energy-aware cellular data scheduling. In *Proc. MobiCom*, 2010.

[7] S.-F. Su. *The UMTS Air-Interface in RF Engineering*. McGraw Hill, 2007.

[8] F. P. Tso, J. Teng, W. Jia, and D. Xuan. Mobility: a double-edged sword for HSPA networks: a large-scale test on Hong Kong mobile hspa networks. In *Proc. MobiHoc*, 2010.

[9] P. Viola and M. Jones. Fast and robust classification using asymmetric AdaBoost and a detector cascade. In *Proc. NIPS*, 2001.