Bandwidth Reservation using WLAN Handoff Prediction

Libo Song[†], Udayan Deshpande[†], Ulas Kozat[‡], David Kotz[†], Ravi Jain[‡]
[†]Dartmouth College, Hanover, NH 03755
{lsong, udayan, dfk}@cs.dartmouth.edu
[†]DoCoMo USA Labs, San Jose, CA 95110
{kozat, jain}@docomolabs-usa.com

1. INTRODUCTION

Many network services may be improved or enabled by successful predictions on users' future mobility. The success of predictions depend on how much accuracy can be achieved on real data and on the sensitivity of particular application to this achievable accuracy. We investigate these issues for the case of advanced bandwidth reservation using real WLAN traces collected at Dartmouth College Campus [1].

In our system model, we envision a wireless network in which users associate with one access point (AP) or cell at a time, and can change their associations from one AP to another one as needed to remain connected. Such reassociation can be caused by a roaming user or it is possible that the user is stationary, and the changes in connection quality may be the main cause. This sequence of handoffs for each user, indicating the time and AP of each move (or "OFF" when the user leaves the network), is their movement history. We assume that there is a centralized or distributed mechanism in place for (i) collecting the history, (ii) performing online predictions, and (iii) distributing prediction results to relevant application agents.

In the case study, we focus on VoIP as the application of interest; we evaluate mobility predictors for advance bandwidth reservation to maintain VoIP service quality after handoffs. We measure the performance using application-specific *call drop rate* and *call block rate* metrics. The results show that intelligent prediction can lead to significant reductions in the rate at which active calls are dropped due to handoffs with marginal increments in the rate at which new calls are blocked.

2. PREDICTORS

We designed a CDF time predictor that produces the probability that the time of the next move is less than (or greater than) a given value. It does so by computing the observed cumulative distribution function (CDF) of the historic values, and using the CDF to measure the probability of a given value appearing in the distribution.

Consider a history H of values $v_1, v_2, \cdots, v_n$. Suppose V is the random variate, which outputs the actual values in H, and P is its distribution. The CDF predictor computes the observed CDF function $\hat{P}$ of V from the histogram, that is,

$$\hat{P}(V < v) = \frac{1}{n}\sum_{i=1}^{n} I(v_i < v), \tag{1}$$

where I is the indicator function. In a similar fashion, we can compute the probability of values occurring in range $a \leq V < b$, by simply computing $P(a \leq V < b) \approx (\hat{P}(V < b) - \hat{P}(V < a))$.

We combine the CDF time predictor with location predictors from our previous work [2] for integrated location and time predictions; we call it the *MarkovCDF Predictor*. This integrated predictor outputs a vector of probabilities, one for each AP that the user will move to within a certain time threshold. Therefore, we can make reservations according to the probabilities of predicted locations and times. These predictors build their internal tables on per-user basis, but it is equally possible to build aggregate tables from all users' movement histories. We name them the *MarkovCDF Individual* predictor and *MarkovCDF Aggregate* predictor, respectively.

We introduce a simple "straw-man" predictor, the *Neighbor Graph Predictor*, to compare with our MarkovCDF predictor. Using users' current neighbor locations as the prediction is an obvious way to predict future locations. The predictor outputs a vector of probabilities, one for each AP that is the neighbor of the current AP. The probabilities depend only on the locations that the user visited in the history. Mishra et al. [3] present an algorithm to dynamically build a user's neighbor graph to cache context for fast handoffs. In our case study we use one month's traces to construct a directed graph representing transition history and then use the graph for the Neighbor Graph Predictor to predict the probabilities on the second month's traces.

3. CASE STUDY

In our Case Study, we simulate a wireless network that is capable of supporting roaming telephone users. When a user has an ongoing call and moves from one access point to another, we refer to that call as a *handoff call*. When a user initiates a call while associated with an AP, we refer to that call as a *new call*. All calls require dedicated bandwidth at their current AP. If the AP lacks the bandwidth for a new or handoff call, the call fails: a failed handoff call is a "call drop" and a failed new call is a "call block". The literature often assumes that call drops are much more frustrating to users than call blocks, so the goal of mobility prediction in this particular application is to reserve bandwidth, in advance of handoffs, to reduce call drops at the expense of a small increase in call

Mobicom05 Poster: A project of the Center for Mobile Computing at Dartmouth College: http://cmc.cs.dartmouth.edu. This research was supported by DoCoMo USA Labs.

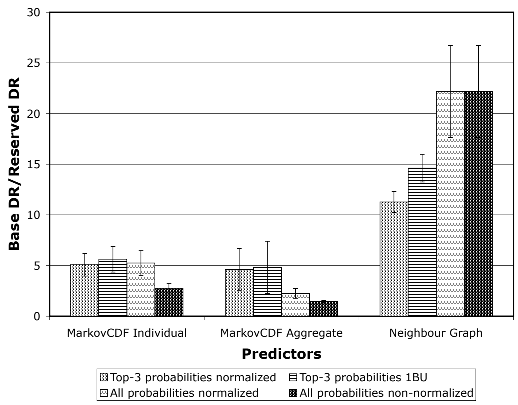

Figure 1: The ratio (Base *DR*/Reserved *DR*) for all the predictors with training. Higher ratio is better.

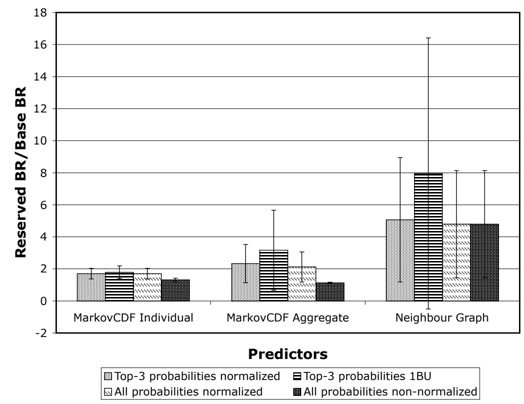

Figure 2: The ratio (Reserved *BR*/Base *BR*) for all the predictors with training. Lower ratio is better.

blocks. Specifically, we define the drop rate and the block rate

$$DR = \frac{\text{number of dropped calls}}{\text{number of attempted call handoffs}},$$

$$BR = \frac{\text{number of blocked calls}}{\text{number of attempted calls}}.$$

4. RESULTS

We evaluated the MarkovCDF Individual, the MarkovCDF Aggregate, and the Neighbor Graph predictors using two months' real mobility traces collected at our campus-wide wireless network. We use the first month's traces to train our MarkovCDF predictors and build the neighbor graph for the Neighbor Graph predictor. The two months' traces include 545 APs and 6,181 users. We also used exponentially distributed call duration and inter-call time to simulate the voice calls.

We implemented four reservation schemes, which differ in the way they use the probabilities returned by predictors. We explain the reservation schemes and the corresponding results below:

- **All probabilities, non-normalized.** We reserve bandwidth proportional to the probabilities returned. The sum of the probabilities is not necessarily 1 for the MarkovCDF predictors, because the sum is the probability that user will move to any location within a certain finite time threshold. The sum for the Neighbor Graph Predictor is 1.

- **All probabilities, normalized.** We reserve bandwidth proportional to the normalized probabilities returned by the predictor. The sum of the probabilities is 1. For Neighbor Graph Predictor, the normalization does not the probabilities.

- **Top-3 normalized probabilities.** We make reservations at the three most probable APs, proportional to the normalized probabilities returned by the predictor. The sum of the three probabilities is 1.

- **Top-3 probabilities, 1BU.** We make reservations of 1 bandwidth unit (BU) at each of the three most probable APs.

Figure 1 shows the ratios of Base *DR* and Reserved *DR*, where the Base *DR* is the drop-rate without using any reservation scheme, and the Reserved *DR* is the drop-rate with reservations. The Figure 2 shows the ratios of Reserved *BR* and Base *BR*, where the Base *BR* is the block-rate without using any reservation scheme, and the Reserved *DR* is the block-rate with reservations. The

Neighbor Graph Predictor reduces the drop-rate the most, while it increases the call block-rate the most. Although the low drop-rate of the Neighbor Graph Predictor is desirable, the high block-rate wastes bandwidth resources. There is a trade-off between the gain of reduced drop-rate and the loss of increased block-rate. We do gain substantial improvement using the MarkovCDF predictors while not worsening the BR as much. The MarkovCDF Aggregate predictor does not improve the *DR* as much as the MarkovCDF Individual does, while worsening the *BR* to a greater extent in some cases. The reason for this result may be because the aggregate predictor does not add much in terms of accuracy of the prediction, but introduces larger probabilities for some of the reservation schemes, resulting in higher over-provisioning of bandwidth.

We observe that using the "Top-3 normalized probabilities" scheme is the best of the four resource reservation schemes for the MarkovCDF predictor, because it concentrates resources at the top three APs that the user is most likely to roam to. The "Top-3 probabilities with 1 BU" scheme reserves excessive resources and causes an increase in the block rate. In the case of the MarkovCDF Individual predictor, the "all probabilities normalized" reservation scheme performs similar to the "Top-3 normalized" scheme because, it is likely that the sum of the top three probabilities is equal to or close to 1. In case of the MarkovCDF predictors, the "all probabilities non-normalized" scheme does not improve the *DR* greatly neither does it worsen the *BR* greatly because, the probabilities returned by the predictor are low.

5. REFERENCES

[1] David Kotz and Kobby Essien, "Analysis of a campus-wide wireless network," *Wireless Networks*, vol. 11, pp. 115–133, 2005.

[2] Libo Song, David Kotz, Ravi Jain, and Xiaoning He, "Evaluating location predictors with extensive Wi-Fi mobility data," in *Proc. of INFOCOM*, March 2004, pp. 1414 – 1424.

[3] Arunesh Mishra, Min-ho Shin, and William A. Arbaugh, "Context caching using neighbor graphs for fast handoffs in a wireless network," in *Proc. INFOCOM*, March 2004, pp. 351–361.