A Rate-Distortion Framework for Information-Theoretic Mobility Management

Abhishek Roy[1], Archan Misra [2] and Sajal K. Das[1]

[1] CReWMaN Lab, CSE Department, University of Texas at Arlington {aroy, das}@cse.uta.edu

[2] IBM T. J. Watson Research, Hawthorn, New York archan@us.ibm.com

Abstract

A practical information theoretic framework is developed for studying the optimal tradeoff between location update and paging costs in cellular networks. The framework envisions quantization of location information into a Registration Area (RA) level granularity, followed by the use of an entropy-coding technique to decrease the location update rate. The rate distortion theory of lossy quantization is identified as an appropriate measure for capturing the optimal tradeoff between a mobile's update rate and its location uncertainty. Based on LZ-78 compression, two different RA-level location update algorithms (RA-LeZi and LeZi-RA) have been developed, both of which asymptotically approach this rate-distortion bound. By allowing for quantization loss in the mobile node's movement pattern, this framework can reduce the overall update cost below the entropy bound associated with the original loss-less LeZi-Update mobility management algorithm. Simulation results demonstrate a sharp decrease ($\sim 50\%$) in the update cost, at the expense of a minor ($\sim 25\%$) increase in the overall location management costs. The key essence of this framework lies in its practical applicability, because today's wireless networks already track the mobile user at an RA-level granularity.

I. Introduction

Algorithms for tracking the location of mobile nodes in a wireless cellular infrastructure essentially utilize two fundamental operations:

- *Update:* whereby a mobile node (MN) proactively informs the network of its current cellular coordinates, thus reducing its locational ambiguity.
- *Paging:* whereby the cellular system potentially searches for the MN in all *plausible cells*, i.e., those cells in which the MN currently has a non-zero probability of residence.

Optimizing the *total* location management cost usually involves trading off the costs associated with these two functions against each other (a lower location update cost implies a higher paging cost, and vice versa). These two costs are often inter-dependent in a non-linear way. In general, MN*-specific* algorithms, where the location update and paging strategies are individualized for each MN's movement history, outperform *global, threshold-based* management algorithms, where all (or groups of) users generate location updates at the same cell transitions. Most MN-specific algorithms, however, assume specific movement models, and do not allow an MN to tradeoff between the update and paging costs in a model-independent fashion.

The work of Abhishek Roy and Sajal K. Das is supported by NSF grants IIS-0121297 and TxTEC.

LeZi-Update [5] is perhaps the first *model-independent* approach to design an optimal location tracking scheme in cellular networks. By modeling the MN's movement as a (piece-wise) stationary, ergodic stochastic sequence, the LeZi-Update algorithm presents an optimal MN-specific approach to mobility management. The movement history of the MN is represented by a sequence of symbols, *where each symbol corresponds to a cell that the MN has traversed*. It then uses an entropy coding algorithm with on-line learning (the Lempel-Ziv ($LZ78$) compression scheme) to reduce the location update cost per symbol to an asymptotically optimal value (defined by the *entropy* of the stochastic process). No other location management approach can *accurately* convey this movement sequence at a smaller cost. The existing LeZi-Update scheme is, however, based on the *lossless* $LZ78$ text compression algorithms, and does not allow the individual user to trade off the location update and paging costs against each other. Moreover, LeZi-Update is purely an MN*-specific* algorithm, which is unable to exploit the global registration-area (RA) based location management infrastructure currently deployed in cellular PCS networks. Based on these observations, our work attempts to answer the following two questions:

(1) Is it possible to further reduce the location update cost incurred by LeZi-Update (below the entropy bound), and if so, how does one modify the location management algorithms in an optimal way?

(2) Is it possible to combine the LeZi-Update approach with the current RA-based approach to offer significantly lower mobility management costs, without disbanding the current PCS infrastructure?

In this paper, we have developed an information-theoretic model to capture the fundamental bounds on the trade-off between location update and paging, which is applicable to *any practical* location update scheme. The model is also constructive, since it allows us to use a variety of coding techniques from information theory to practically achieve these bounds. The concept of *rate distortion* functions from information theory is used to formalize the notion of how the location update cost of an MN can be arbitrarily reduced, at the expense of correspondingly higher paging costs. In essence, we investigate the use of *lossy quantization* algorithms with the LeZi-Update framework, such that the network does not necessarily receive the MN's true movement history, but a "reasonably close" approximation to it. Like the LeZi-Update approach, our tradeoff framework is also model-independent and *adapts* to the MN's movement history.

Our approach combines the LeZi-Update technique with the current RA-based management scheme. In this approach, we preserve the current PCS design, whereby cells are statically partitioned into a set of RAs. We have developed two different algorithms, RA-LeZiand LeZi-RA, to express the MN's movement history into two different sequences of symbols, where the *symbols refer to the registration areas*, rather than the individual cell. By applying $LZ78$ compression on each of these RA-based symbol sequences, we have reduced the location update cost of the MN below its entropy bound (in terms of its cell-level uncertainty), albeit by sacrificing on the precision of the reported sequence. Both the update algorithms correspond to a combined quantization and entropy coding technique. Information theoretic concepts are used to express the reduction in the update rates, and the increase in the paging costs, with the variation in the size of each RA. In particular we have shown that in practical systems with reasonable bounds on the paging latency, our algorithms are able to track the MN's location with an overall cost very close to the optimal LeZi-update strategy, but at much lower update cost.

Section II briefly summarizes the existing location update and paging schemes. The practical estimation of location uncertainty in a lossy wireless environment is mathematically formulated in Section III. Subsequently, we have developed the new, near-optimal location management framework, and the two proposed location update strategies in Section IV. Simulation results in Section V empirically demonstrate the advantages and relative performance of our two proposed strategies. Section VI concludes the paper with pointers to future research.

II. Related Work on Location Management

Given the large amount of work on location tracking in cellular systems, we briefly discuss only the broad categories of location management solutions. Cellular PCS networks generally cluster a group of cells into *registration areas* (RAs), such that an MN's location uncertainty is confined into its last reported RA. The mobile node (MN) performs proactive location updates only when it changes its current RA, and not on every cell-change. To resolve the MN's precise location within its current RA, the network pages the MN simultaneously in all the cells within that RA. Various strategies, such as simulated annealing [8], have been used to adjust the RA partitions to minimize the cumulative location update and paging costs. While it is well-known that larger RAs imply lower update and higher paging costs, the PCS framework is, however, far from optimal. MN-specific location update strategies can be broadly classified into three groups based on the different thresholds used: distance-based [1], time-based [12] and movement-based [2], whose relative performance has been analyzed in [4].

Rose and Yates [11] have pointed out that the naive approach of paging the mobile simultaneously, over all cells with a non-zero residence probability, is an overkill. They have shown that, in absence of any constraint, the expected paging cost is minimum when the cells are paged in decreasing order of occupancy probabilities. Even when the paging process is subject to a maximum delay constraint, they use an approximate dynamic programming-based solution to demonstrate how an optimal paging sequence can significantly lower the expected paging cost (using a uniform residence probability distribution). The basis of cluster paging [11] and directional paging [4] lies in the underlying assumption that the occupancy probabilities decrease omni-directionally with increasing distance from the last known (updated) location. More recently, the idea of profile-based paging [10], [5] has also been formulated.

LeZi-Update [5] has used lossless LZ78 data compression algorithms to develop the most generalized information-theoretic framework for optimal location tracking. However, none of the above-mentioned approaches has explicitly considered mechanisms to tradeoff between the location update and paging costs, by appropriately modifying the accuracy of the mobile's update sequence. Supporting such a tradeoff mechanism is important, since mobile nodes with identical movement patterns can have very different battery capacities or paging delay constraints. A major novelty of our scheme lies precisely in its ability to allow for loss of information during the location update process.

III. Rate Distortion Theory and its Implication in Mobility Management

Our proposed location management framework expresses the movement of a mobile in an abstract symbolic space. Unlike the existing LeZi-Update, where every symbol represents a particular cell, our location tracking framework works at an RA-level granularity. In other words, every cell is quantized to its nearest possible RA, and the set of RAs are represented by a sequence of symbols. Accordingly, the framework is practical, universal and can even support networks having geometric location information.

Let the lower-case symbols $a, b, c, \ldots$ denote the individual cells of the cellular system, while the upper case symbols represent the registration areas. From a cell-level standpoint, the movement history of the mobile node (MN) is represented as a random sequence $X^n = \{x_1, x_2, \ldots, x_n\}$, where each element x_i is a particular symbol from the alphabet $\mathcal{C}$, where $\mathcal{C}$ is the set of all cells. When expressed at the RA-level granularity, the MN's movement history appears as a random sequence $Y^n = \{Y_1, Y_2, \ldots, Y_n\}$, where each Y_i is a symbol from the alphabet $\mathcal{R}$, where $\mathcal{R}$ is the set of RAs. For any RA, Y, let $cell(Y)$ denotes the set of cells comprising that RA. Similarly, for any cell, x, let $Q(x)$ denote the ID of the corresponding RA. We assume that the system has N cells, partitioned into M different RAs. The mobility of MN essentially creates an uncertainty of its location. Shannon's entropy [7] is the most fair measure to estimate this uncertainty.

Definition 1: The entropy $H(X)$ of a discrete random variable X, with probability mass function $p(x)$, is defined by

$$H(X) = -\sum_{x \in \mathcal{C}} p(x) \lg\, p(x) \tag{1}$$

The limiting value "$\lim_{p \to 0}\, p\, \lg\, p\, =\, 0$" is used in the expression when $p(x) = 0$.

Result *1: It is impossible to reconstruct any (piece-wise) stationary, ergodic, stochastic sequence at the receiver end at a cost less than its entropy $H(X)$. Shannon's entropy provides the estimate the minimum amount of information needed to transmit a coded sequence in a lossless manner.*

However, most of the practical systems are inherently lossy and thus the receiver can only construct an approximation of the original sequence. This insight forms the basis of our work.

The essence of our scheme lies in a lossy learning of the MN's movement pattern, i.e., allowing for some distortion in the location update process. From the system's perspective, the MN generates location updates, not at a cell-level granularity, but based on a *quantized* version of it. Indeed, the many-to-one mapping from multiple cells to a specific RA, via the function $Q(x)$, can be viewed as a scalar quantization of the original symbol sequence. The MN's quantized movement history is then the stochastic process $\mathcal{Y}^n = \{Y^n\}$, where the repetitive patterns add piece-wise stationarity. Even if the system receives the sequence Y^n without any error, the original sequence can never be recovered, implying that the transmission process results in a distortion cost. Let us assume that the system maps each RA value, Y_i, to a particular cell $x_i \in cell(Y_i)$, according to a function $x_i = W(Y_i)$. In general, the fidelity of this quantization process is expressed via a *distortion function*, $d(.)$, that expresses the cost between an input vector X^n and a quantized vector $W(Y^n)$ = $\{W(Y_1), W(Y_2), \ldots, W(Y_n)\}$.

Definition 2: If the sequence $X^n = (x_1, x_2, \ldots, x_n)$ a the realization of the ergodic stochastic process $\mathcal{X}^n$, and $W(Y^n)$ is the lossy estimate of this sequence, then the distortion is estimated as $d_n(X^n, Y^n)$, where $d_n(X^n, Y^n) = 0$ if $X^n = W(Y^n)$ and $d_n(X^n, Y^n) > 0$ otherwise.

The system is D-$semifaithful$ if $d_n(X^n, Y^n) \leq D$. In general, while $d_i(.)$ can be any legitimate cost function, most practical applications use a scalar (or per-symbol) distortion measure, with

$$d_n(X^n, Y^n) = \frac{1}{n} \sum_{i=1}^{n} d_i(X_i, W(Y_i)), \tag{2}$$

where $d_i(.)$ is a scalar distortion measure (e.g., squared distance or Hamming distance). From a stochastic viewpoint, we are interested in the expected distortion $E[d_n(X^n, Y^n)]$ over all plausible input sequences X^n. For the location update problem, this distortion cost should relate to the eventual paging cost, since the loss of information about the MN's precise movement history will eventually manifest itself in a less-optimal paging strategy. While we are not concerned with the definition of specific distortion functions, it should be clear that the larger the size S of an RA, the larger should be the distortion value D, since the quantization error is proportionately greater. This measure of fidelity forms the basis of rate distortion theory [7]:

Definition 3: For any distortion $D \geq 0$ and $n \geq 1$, the n^{th}-order rate-distortion function of $\mathcal{X}^n$ is given by

$$R_n(D) = \inf_{(X_1^n, Y_1^n),\, E[d_n(X_1^n, Y_1^n)] \leq D} I(X_1^n, Y_1^n) \tag{3}$$

where $I(X_1^n, Y_1^n) = p(x_1^n, y_1^n) \lg \frac{p(x_1^n, y_1^n)}{p(x_1^n)p(y_1^n)}$ denotes the mutual information between the sequence and its distorted estimate, $p(x_1^n, y_1^n)$ is the $n^{th} - order$ joint probability of two random variables x and y and the infimum (inf) is taken over all jointly distributed random vectors (X_1^n, Y_1^n) with values in $\mathcal{C}^n \times \mathcal{R}^n$. The limiting operational value is achieved as $R(D) = \lim_{n\to\infty} (1/n)\, R_n(D)$.

$R(D)$ essentially indicates a lower bound on the symbol-generation rate (update cost) of any feasible scheme that transmits the sequence $\mathcal{X}^n$ without incurring a distortion cost larger than D. To ensure lossless reproduction of the random process $\mathcal{X}^n$, the update rate cannot be lower than the entropy, i.e., $R(0) = H(X)$. However, by allowing for progressively larger values of the error cost D, we can reduce the update rate/symbol arbitrarily below this entropy bound. From our perspective, the most important result is that [13]:

Result *2: For memoryless sources, cascading the scalar quantizer with a conventional (lossless) entropy coding algorithm is nearly optimal in that it yields a rate very close to the rate-distortion bound.*

In the next section, we shall use this result and the rate-distortion framework to present two alternative approaches for combining the use of registration areas with LeZi style updates.

IV. Comparative Picture of Two Different Algorithms

We now present two alternative location management strategies that exploit this notion of rate-distortion coding to reduce the MN's location update cost below the entropy bound, at the expense of increasing the paging cost. The two strategies can be viewed as two different approaches to scalar quantization of the underlying movement sequence of the MN, and allow us to tradeoff between the location update and paging costs in two different ways. Both approaches to quantization-based location update operate in two stages. In the first stage, each cell associated with the mobile user is represented by its corresponding registration area (RA). In other words, the movement of the mobile user is *quantized* to a coarser RA-level granularity. Note that the RAs are themselves not user-specific, but are statically defined for all users. In the second stage, this (user-specific) quantized sequence is processed in chunks according to the LeZi-Update algorithm. Both the schemes, however, use an identical paging process, which is described in Section IV-C.

A. *The LeZi-RA Location Update Algorithm*

```
initialize dictionary := NULL
initialize phrase w:= NULL
loop
  wait for next symbol (cell) x_i
  quantize the symbol to get the RA Y_i = Q(x_i)
  if(w.Y_i in dictionary)
    w := w.Yi
  else
    encode <index w,Y_i>
    add w.Y_i to dictionary
    w := NULL
  endif
forever
```

Fig. 1. Encoder of the LeZi-RA scheme in mobile

```
initialize dictionary := NULL
loop
  wait for next codeword
  decode phrase
  add phrase to dictionary
  increment frequency for every prefix
            of every suffix of phrase
forever
```

Fig. 2. Decoder at the system

In the first location-update approach, which we call ***LeZi-RA***, every change in the MN's cell x_n is mapped to the corresponding RA Y_n, given by $Y_n = Q(x_n)$. The MN then acts as an encoder, performing LZ78 compression on this quantized sequence Y^n. Thus, the movement pattern "$x_1x_2x_3\ldots$" is first quantized as "$Y_1Y_2Y_3\ldots$", and eventually reaches the location register as a sequence "$C(w_1)C(w_2)C(w_3)\ldots$," where w_is are non-overlapping, distinct segments of the string (sequence of RAs) "$Y_1Y_2Y_3\ldots$" and $C(w)$ is the encoding for segment w. For example, let us assume that the RAs A, P, Z and E respectively consist of cells (a, b, c), (p, q, r), (x, y, z) and (e, f, g). Now, let the movement of MN in the cellular granularity is given by the string ζ_I = "a, b, c, c, p, p, q, r, r, x, x, y, z, z, f, g, e, g, a, a, b, c, p, p, q, r, e, e, g, x, y, y, z". This movement sequence is now quantized in the RA-level to obtain the lossy sequence ζ_Q = "A, A, A, A, P, P, P, P, P, Z, Z, Z, Z, Z, E, E, E, E, A, A, A, A, P, P, P, P, E, E, E, Z, Z, Z, Z." The algorithm for generating updates (at the MN) and for processing them (at the location registers) is described in Figures 1 and 2. Following these algorithms, the lossy sequence is incrementally parsed as A, AA, AP, P, PP, PZ, Z, ZZ, ZE, E, EE, AAA, APP, PPE, EEZ, ZZZ. This symbol-wise context model is efficiently stored in a dictionary implemented as a search trie. Figure 3 shows these different phrases with their frequencies, where the frequency of every symbol is incremented for *every prefix of every suffix* of each phrase [5]. The incremental parsing accumulates larger and larger sequences of RAs in the dictionary, thereby accruing estimate of probabilities of all possible orders.

In the LeZi-RA algorithm, the input symbol rate to the LZ coder equals the cell transition rate of the MN. However, LeZi-RA will outperform the simple RA-based reporting algorithm currently used in cellular PCS, since the actual update rate (the output of the LZ coder) adjusts to each MN's individual RA-level movement pattern.

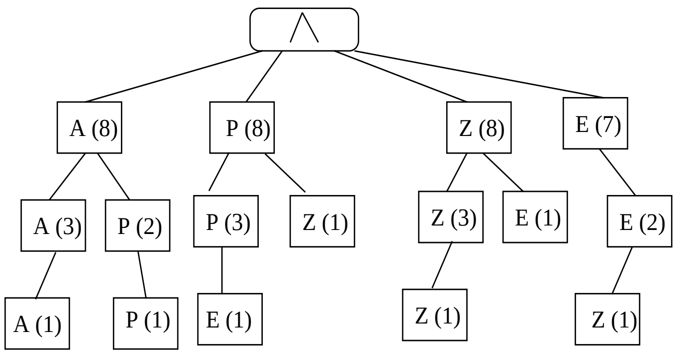

Fig. 3. Trie for the LeZi-RA Scheme

```
initialize dictionary := NULL
initialize phrase w:= NULL
loop
  wait for next symbol x_i
  if(Q(x_i) ≠ Q(x_{i-1}))
    quantize x to get Y_i = Q(x_i)
    if(w.Y_i in dictionary)
      w := w.Y_i
    else
      encode <index w,Y_i>
      add w.Y_i to dictionary
      w := NULL
    endif
  endif
forever
```

Fig. 4. Encoder of the RA-LeZi scheme in mobile

Result *3: The location update cost per symbol of the LeZi-RA algorithm is lower than the corresponding update cost of the original LeZi-Update algorithm.*

Proof: We first show that the entropy of the quantized sequence is smaller than the entropy of the original movement sequence of the MN. To see this, note that $H(X,Y) = H(X) + H(Y/X) = H(Y) + H(X/Y)$. Accordingly, we have

$$\begin{aligned} H(Y) &= H(X) + H(Y/X) - H(X/Y) \\ &= H(X) - H(X/Y) \leq H(X), \;\; \text{(since } H(Y/X) = 0, \text{as the } cell \rightarrow RA \text{ mapping is deterministic)} \end{aligned} \tag{4}$$

Since the LZ-compression algorithm asymptotically approaches the entropy rate of the input sequence, it follows that the update cost for LeZi-RA (i.e., $H(Y)$ bits/symbol) will be lower than that of LeZi-Update (i.e., $H(X)$ bits/symbol).

B. *The RA-LeZi Location Update Algorithm*

The second approach is called *RA-LeZi* and involves a further compression of the quantized sequence. In RA-LeZi, a new quantized sequence (RA value) is generated as input to the LZ coder only when the RA value changes from its previous value. Since each RA typically comprises multiple contiguous cells, the MN would typically make multiple cell transitions within the same RA. Note that, RA-LeZi essentially ignores the run length of these quantized symbols, and generates a new symbol only when the MN changes its current RA. The string ζ_I, formulated earlier, is now quantized into the lossy string, ζ_L = "A, P, Z, E, A, P, E, Z". Subsequently, the parsing results in the the A, P, Z, E, AP, EZ. Clearly, in this process, we are further reducing the input symbol rate to the LZ coder. Figure 4 provides the encoder of this new update mechanism; the decoder remains the same as described in Figure 2.

It is, however, interesting to observe that the relative performance of LeZi-RA and RA-LeZi is dependent on the precise movement model – one cannot make model-independent claims that one scheme will outperform the other. While LeZi-RA has a higher symbol input rate to the LZ compressor, this symbol sequence also has much higher correlation (since the MN will typically make multiple successive transitions in the same RA), thereby incurring a lower entropy cost *per input symbol*. On the other hand, while RA-LeZi has a lower symbol generation rate, these symbols will have lower correlation (greater randomness), since the RA-LeZi symbol sequence does not capture the self-transitions that occur within an RA. Accordingly, the entropy cost *per input symbol* will be higher than that of LeZi-RA. The overall cost, *per cell transition, or per time*, is then model-dependent, since it depends on the combination of the actual cell transition rate, and the correlation between successive cell transitions (whether or not an MN makes multiple transitions within an RA).

Our strategy of performing LZ compression on the RA-level sequence, rather than the cell level sequence, is motivated by the observation that the use of coarser-grained location coordinates *usually increases the regularity observed in the movement pattern* of typical mobile users. For example, in the Boring Professor Model [15] of user mobility, an MN usually travels between a set of well-defined points of interest (such as home, work or shopping center). While the paths between these points can, however, exhibit some randomness at cell-level granularity (e.g., the mobile user takes a detour to a travel agency on the way to work), these perturbations disappear when the user pattern is considered at a coarser (RA-level) granularity.

C. Paging Algorithm (for both LeZi-RA and RA-LeZi)

The paging process is identical for both the LeZi-RA and RA-LeZi update strategies, and essentially involves a *trie-traversal* to compute the unconditional residence probabilities of the MN at each RA present in the trie. The unconditional residence probabilities are computed using a three-stage approach. In the first stage, *prediction by partial match* (PPM) style blending techniques [14] are used to compute the probability

```
Initialize i := 0, Pr[ψ̄] := 0, h:=highest order
Initialize escape probability Pr_h^(e) := 1
While (i ≤ h)
       Search for ψ̄ at order h − i
       If (ψ̄ found)
          Compute its (h − i)^th order
          probability (Pr_{h−i}[ψ̄])
       else
            Pr_{h−i}[ψ̄] := 0
       End-if
       Compute the escape probability (Pr_{h−i}^(e))
       to order (h − i)
       Estimate the combined probability as
       Pr[ψ̄] := Pr[ψ̄] + Π_{j=h}^{h−i} Pr_j^(e) × Pr_{h−i}[ψ̄]
       i := i + 1
End-while
Compute the probability of individual RAs based on
their relative weights in the phrase
Sequentially probe the RAs in decreasing order
of normalized residence probabilities
Page the cells of an RA in any random order
(since they have identical residence probabilities)
```

Fig. 5. Paging Algorithm

of a particular phrase (sequence of RA). The algorithm, shown in Figure 5, starts from the highest order of context (leaf nodes in the trie) and *escapes* to lower order, until order-0 (the root) is reached. If ξ, $\mathcal{N}(\omega)$, $\mathcal{L}(\omega)$, $\mathcal{S}^k(\omega)$ and $\mathcal{P}(\omega)$ denote the last updated phrase, number of occurrences of a phrase ω, its length, k-th suffix, and prefix respectively, the probability of any phrase $\bar{\psi}$ can be estimated by the recursive formula:

$$Pr\left[\psi\right] = \frac{\mathcal{N}(\psi \mid \mathcal{P}(\mathcal{S}^k(\xi)))}{\sum_\omega \mathcal{N}(\omega \mid \mathcal{P}(\mathcal{S}^k(\xi))} + \frac{\mathcal{N}(\Lambda \mid \mathcal{P}(\mathcal{S}^k(\xi)))}{\sum_\omega \mathcal{N}(\omega \mid \mathcal{P}(\mathcal{S}^k(\xi)))} \times \Pr\left[\mathcal{S}^1(\mathcal{P}(\mathcal{S}^k(\bar{\xi})))\right]$$

for all k, where $1 \leq k \leq \mathcal{L}(\bar{\xi})$.

In the next stage, following the principle of *insufficient reason* [9], the phrase probability mass is distributed according to its *type* or *composition*. The probability of an individual symbol (RA), is thus essentially computed based on the relative weights of symbols on these phrases. Formally, the probability $\varrho(R_i)$ of each RA, R_i, is then obtained as

$$\varrho(R_i) = \sum_\psi \frac{\mathcal{N}(R_i)}{\mathcal{L}(\psi)} \times \Pr\left[\psi\right], \tag{5}$$

Unlike LeZi-Update, the unconditional residence probabilities in LeZi-RA and RA-LeZi apply not to individual cells, but to RAs. Accordingly, these unconditional probabilities are now equally distributed among the constituent cells. For example, if S_i denotes the set of cells constituting the RA Y_i, the unconditional probability ϱ_{Y_i} is distributed among these cells $\varrho(x) = \frac{\varrho(Y_i)}{S_i}$ $\forall x \in S_i$.

The optimal paging scheme, in the absence of any delay constraints, is to page the cells in the decreasing order of these unconditional residence probabilities. Due to the RA-based updates of LeZi-RA and RA-LeZi, it is clear that the unconditional residence probabilities for all cells belonging to a common RA will be identical. Accordingly, paging the cells in decreasing order of their residence probabilities is effectively identical to paging the RAs in the order of their *normalized residence probabilities* (residence probability divided by the RA size). Within each RA, the cells can be sequentially paged in random order.

It is interesting to compare the costs of paging algorithm employed by LeZi-RA and RA-LeZi with the paging costs of original LeZi-Update. In general, by dispersing the paging probabilities of an RA over all its constituent cells, our paging algorithm may either a) include cells in the paging sequence that the MN has never visited (and have zero residence probability under LeZi-Update), or b) change the order in which the cells are paged. As an example of the first possibility, consider an extreme situation, where the MN simply visits K cells (indexed in their optimal paging order), $c_1, c_2, \ldots, c_K$, *each of which lies in a different RA*, $Y_1, \ldots, Y_K$. Then, the expected minimum paging cost, P_{min}, is clearly given by

$$P_{min} = \sum_{i=1}^{K} i \times \varrho(c_i), \tag{6}$$

where $\varrho(c_i)$ is the unconditional residence probability in cell c_i. In our strategies, the unconditional residence probability of each cell c_i will be distributed among all the cells in the set $cell(Q(c_i))$. Assuming that each RA consists of S cells (i.e., $S_i = S, \forall i$), our paging sequence will consist of $K \times S$ cells. The order of paging the RAs will, however, *remain unchanged*, since the normalized residence probabilities are divided identically (by S) for each RA. Moreover, if the MN is indeed in cell c_i, it will be paged first in $(i-1) \times S$ cells (in each of $\{Y_1, \ldots, Y_{i-1}\}$), and, on average, $\frac{S}{2}$ cells in RA Y_i. Accordingly, the expected paging cost $E[P]$ is then given by:

$$\begin{aligned} E[P] &= \sum_{i=1}^{K} \left[S \times (i-1) + \frac{S}{2} \right] \varrho(c_i) \qquad (7) \\ &= \frac{S}{2} \sum_{i=1}^{K} \varrho(c_i) \times (2i-1), \\ &= \frac{S}{2} \times \{2\, P_{min} - \sum_{i=1}^{K} \varrho(c_i)\}, \ \text{(from Eq.6)} \\ &= \frac{S}{2} \times \{2\, P_{min} - 1\}, \quad \text{(since } \sum_{i=1}^{K} \varrho(c_i) = 1) \end{aligned}$$

We thus see that the paging cost in this case is S times the optimal paging cost. Clearly, this motivates us to consider a distortion measure (in Equation 2), that is proportional to S, the size of an RA. However,

Result *4: In general, the expected paging cost incurred by LeZi-RA or RA-LeZi can be arbitrarily high (trivially $N/2 \times P_{min}$, for a system comprising N cells) even if the optimal paging cost P_{min} is arbitrarily small (close to 1).* (This trivial bound is reached when the true residence probability is arbitrarily close to 1 in one cell, while our paging algorithm computes this probability to be uniformly distributed across all N cells).

Proof: We prove this 'worst-case' result by construction. Consider a situation where the optimal paging sequence involves K cells, $\{c_1, c_2, \ldots, c_K\}$, each belonging to a distinct RA, with the residence probability of c_i being $\varrho(c_i)$. In general, it is possible to create smaller and smaller expected paging costs by skewing the distribution of $\varrho(i)$. Now, consider the situation where the size of each RA is different and $\propto \frac{1}{\varrho(i)}$, such that the normalized residence probabilities of all Y_is are equal. If the Y_is ($i = 1, \ldots, K$) now cover all the N cells of the cellular system, it follows that the residence probability of the MN, (from the viewpoint of our algorithm) is *uniformly distributed over all cells.* Accordingly, in the worst case, our expected paging cost can prove to be as high as $\frac{N}{2}$, even though the optimal expected cost is arbitrarily close to 1. Of course, this is an extremely pessimistic result, since this 'worst-case' result is unlikely to occur in any practical system.

C.1 Paging with Delay Constraints

Since we are specifically concerned with designing a practical mobility management scheme, we also consider a paging process subject to constraints in the maximum allowable paging delay. We assume that this constraint is expressed as Max, which indicates the upper bound on the number of sequential paging attempts. In such a constrained environment, our paging strategy essentially groups RAs into Max different clusters, $CL_1, CL_2, \ldots, CL_{Max}$, indexed in the order in which these clusters are paged. Following the results in [11], any RA belonging to cluster CL_j should have a normalized residence probability not greater than the normalized residence probability of any RA belonging to clusters $CL_i : \forall i < j$. All cells belonging to RAs in the same cluster are then paged simultaneously. The actual clustering of these RAs is performed using the approximation technique presented in [11].

V. Simulation Results

We now present results obtained using a discrete-event simulation framework that we developed for studying the movement of a mobile user. Synthetic traces of user's activities are dynamically generated and fed in. In order to consume memory space economically, while keeping the scheme fast and efficient, every trie is implemented as a hash-table, which is periodically refreshed. The user has a home and a work-place randomly chosen in different cells. The time spent by the user at every cell is normally distributed. The movement pattern of the user differs between weekdays and weekends/holidays. The call arrival process is Markov-modulated with three distinct states, each having its own Poisson arrival rate λ and normally distributed holding times with mean μ and variance σ. The states represent weekday daytimes ($\lambda = 0.2$ calls/hr, $\mu = 10$ min, $\sigma = 3$ min), weekday evenings ($\lambda = 0.3$ calls/hr, $\mu = 20$ min, $\sigma = 5$ min), and weekends ($\lambda = 0.5$ calls/hr, $\mu = 30$ min, $\sigma = 7$ min). The entire simulation results are based on an observation for a period of 12 weeks.

A. Performance with Unconstrained Paging

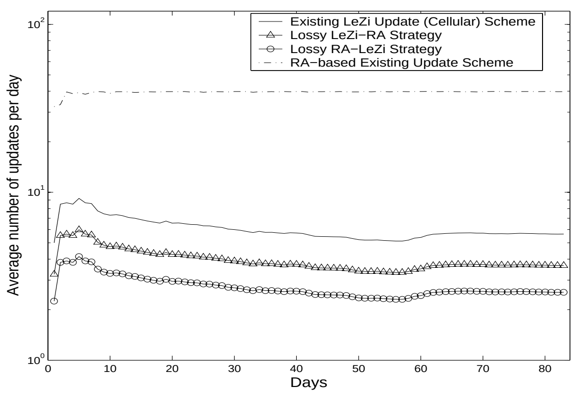

Fig. 6. Comparison of Different Update Costs (S=5)

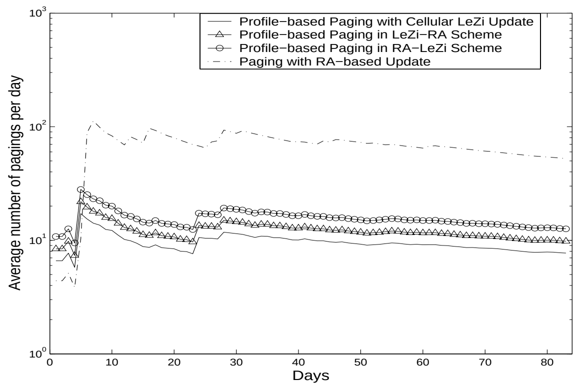

Fig. 7. Comparison of Different Paging Costs (S=5)

We first present the results of a typical study to compare the *update* and *paging* costs for four strategies – existing PCS-style RA-based updates, existing LeZi-Update (at cellular granularity), and our proposed LeZi-RA and RA-LeZi schemes. The plots in Figures 6, 7 and 8 correspond to the case where each RA comprises $S = 5$ cells. Figure 6 shows that both of the proposed LeZi-RA and RA-LeZi schemes result in significant savings in update cost, compared to existing RA-based update strategy and the original LeZi-Update scheme (cellular granularity). The update-cost of LeZi-RA (and RA-LeZi) is only $\sim 9\%$ (and $\sim 7\%$) of existing RA-based updates and $\sim 63\%$ (and $\sim 49\%$) of existing LeZi-Update. This is indeed, a direct consequence of lossy compression and quantization at a higher-level of granularity.

The penalty for these savings in the update cost occur in terms of higher paging cost. Figure 7 demonstrates that the paging cost associated with LeZi-RA and RA-LeZi schemes are bounded by ~ 1.8 times and ~ 2.3 times that of the optimal profile-based paging cost provided by LeZi-Update. However, the two schemes still result in only $\sim 21\%$ and $\sim 33\%$, respectively, of the existing paging cost associated with RA-based updates in current PCS systems. To obtain an overall picture of the performance of our newly proposed schemes, we have measured the total location management (location update + paging) cost of all the 4 schemes, for changing values of the location update to paging cost ratio (UPR). The higher the UPR, the higher is the cost of transmitting an update, relative to the cost of a single paging message. Figure 8 shows that the proposed LeZi-RA and RA-LeZi algorithms respectively, result in only ~ 25-29% and ~ 21-27% more than the optimal location management scheme (LeZi-Update), and also incur only ~ 25-30% of the total cost associated with the existing RA-based location management scheme. It can also be noted that while the LeZi-RA scheme outperforms the RA-LeZi scheme for low update to paging ratios, the relative performance of the two schemes is reversed at higher values of the UPR.

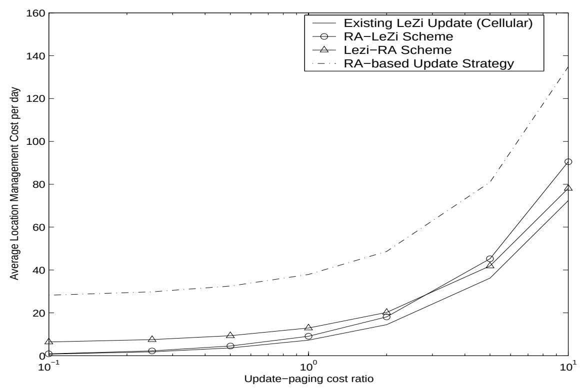

Fig. 8. Total Location Management Cost (S=5)

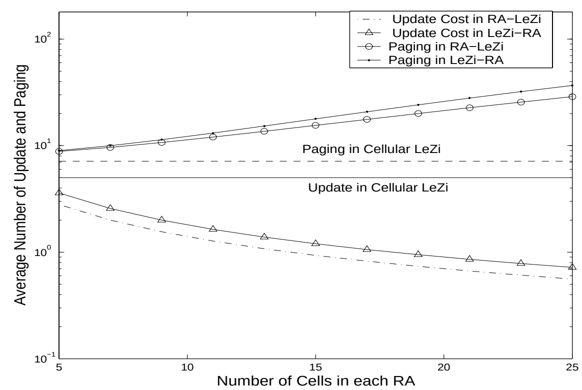

Fig. 9. Performance Tradeoff of RA-LeZi and LeZi-RA

To further illustrate the rate-distortion framework under which LeZi-RA and RA-LeZi allow the system to tradeoff between paging and update costs, we now present performance results for different values of S, the number of cells/RA. Figure 9 shows the update and paging cost for the four schemes, and shows how the LeZi-RA and RA-LeZi schemes result in progressively greater savings in update cost as S increases. Clearly, a larger S corresponds to a larger distortion value D, and thus lowers the rate-distortion bound $R(D)$ on the minimum feasible update rate. However, as Figure 9 illustrates, the savings in update cost come at the expense of higher paging costs. Clearly, a coarser quantization process (larger S) increases the loss of information about the MN's precise movement pattern, and hence results in a greater deviation from the optimal paging sequence. In high UPR scenarios (where the system is more concerned about conserving MN's energy by reducing the number of location updates), the use of the LeZi-RA or RA-LeZi strategy, together with larger values of S, can result in a location management algorithm with a much lower update rate, yet with an overall cost that is only marginally higher than the optimal LeZi-Update scheme.

Figure 10 is an alternative representation of Figure 9, in that it plots the *change* (or 'delta') in the update and

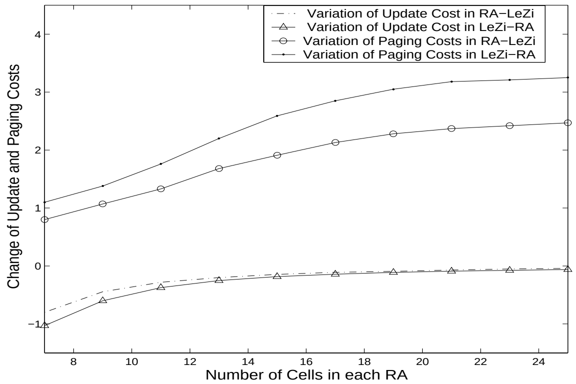

Fig. 10. Variation of Update and Paging Costs

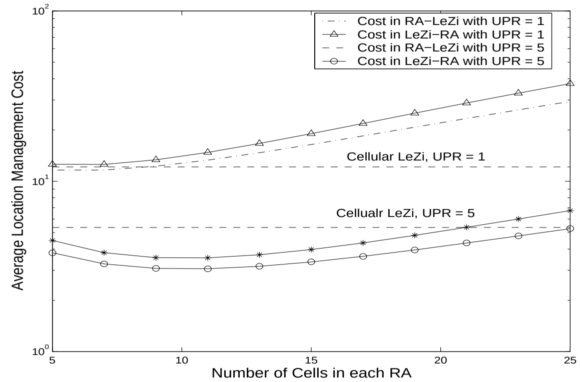

Fig. 11. Variation of total Location Management Costs

paging costs, with successive increases in the number of cells (S) in RAs. As the figure shows, the changes to both the update and paging costs are non-linear. (The negative increments for the update cost indicate that the update cost actually decreases with increasing S.) number of cells per RA). In particular, the rate of decrease in the update costs diminishes for increasing S, while the rate of increase in the paging costs is fairly linear in S. Figure 11 shows the total location management (update+paging) costs for two different UPRs with increasing number of cells per RA. The total location management cost of LeZi-Update is taken as the basis of comparison. Initially both LeZi-RA and RA-LeZi results in lower location management costs than cellular LeZi update, when $UPR = 5.0$. For $UPR = 1.0$ the total location management cost is initially almost same. With increasing cells/RA the update cost decreases but the paging cost increases. After a certain point ($S \approx 7\, for\, UPR = 1.0$ and $S \approx 15\, for\, UPR = 5.0$) the increase of paging cost outperforms the reduction of update costs; beyond this point, the total location management cost starts increasing and exceeds that of cellular LeZi-Update.

B. Performance Under Paging Latency Bounds

Since practical systems will always impose bounds on the maximum acceptable paging latency, we now consider the performance of LeZi-RA and RA-LeZi in scenarios where the system imposes a bound on the number of sequential paging attempts. Since bounds on the paging latency do not modify the update behavior, we only consider the changes in the paging costs. Figure 12 presents the average paging cost (average number of individual paging messages per paging request) for LeZi-RA, RA-LeZi, and the original LeZi-Update strategy, for different values of Max, the maximum allowable number of sequential paging attempts. The plot shows that, under practical constraints on the maximum tolerable paging latency, both the RA-LeZi and LeZi-RA schemes generate only 5-15% more paging overhead than the optimal LeZi-Update scheme.

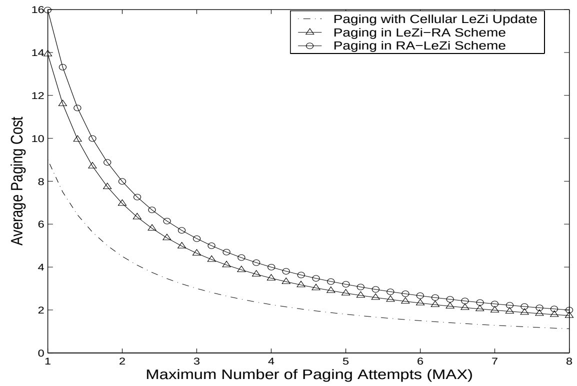

Fig. 12. Comparison of Different Paging with Delay Constraints

VI. Conclusions and Future Work

In this paper we have developed an information-theoretic framework for trading off the location update and paging costs in a near-optimal fashion. By considering the movement of an individual mobile node as an outcome of a stochastic process, the tradeoff problem is expressed in terms of rate-distortion functions. The framework demonstrates how the location update cost of a mobile node can potentially be lowered well below its entropy bound, at the expense of a corresponding increase in the paging cost. For any well-defined distortion measure D, the rate-distortion function $R(D)$, then expresses the minimum feasible update rate. We have then presented two different online, model-independent location update algorithms, called LeZi-RA and RA-LeZi, that leverage the existing RA-based PCS infrastructure, and first quantize the MN's location history at the RA-level granularity, and subsequently use an entropy coder to reduce the update rate to $R(D)$. Extensive simulation experiments demonstrate that both these schemes can reduce the update cost to $\sim 50\%$ of the existing LeZi-Update strategy, while causing only a minor increase in the overall (paging+ location update) location management cost. Simulations also demonstrate that these update costs can be lowered even further by increasing the size of each RA, effectively increasing the distortion between the mobile node's true mobility pattern and the pattern reported to the system. Even when we consider practical networks, which are subject to constraints on the maximum paging delay, both of our algorithms suffer from only a small increase in the paging cost. Our future interests lie in further refining this approach to develop lossy compression algorithms and pattern matching techniques for multi-system, heterogeneous wireless networks.

References

[1] I. F. Akyildiz and J. S. M. Ho, "Dynamic Mobile User Location Update for Wireless PCS Networks", *ACM Wireless Networks*, vol. 1, no. 2, pp. 187-196, July 1995.

[2] I. F. Akyildiz and J. S. M. Ho, "Movement-based Location Update and Selective Paging for PCS Networks," *IEEE/ACM Transactions on Networking*, vol. 4, no. 4, Dec. 1995.

[3] A. Bar-Noy and I. Kessler, "Tracking Mobile Users in Wireless Communication Networks", *IEEE/ACM Trans. on Information Theory*, vol. 39, no. 6, pp. 1877-1886, Nov. 1993.

[4] A. Bar-Noy, I. Kessler and M. Sidi "Mobile Users: To Update or not to Update", *Wireless Networks*, vol. 1, no. 2, pp. 175-185, July 1995.

[5] A. Bhattacharya and S. K. Das, "LeZi-Update: An Information-theoretic approach for personal mobility tracking in PCS networks," *ACM-Kluwer Wireless Networks* (WINET), vol. 8, no.2, pp.121-137, 2002.

[6] Y. Birk and Y. Nachman, "User Direction and Elapsed-time Information to Reduce the Wireless Cost of Locating Mobile Users in Cellular Networks", *Wireless Networks*, vol. 1, no. 4, pp. 403-412, Dec 1995.

[7] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley, 1991.

[8] I. Demirkol, C. Ersoy, M. U. Caglayan, H. Delic, "Location Area Planning in Cellular Networks Using Simulated Annealing", *Proc. of IEEE InfoCom*, vol. 1, pp.13-20, 2001.

[9] J. N. Kapur and H. K. Kesavan, *Entropy Optimization Principles with Applications*, Academic Press, 1992.

[10] G. P. Pollini and C.-L. I, "A Profile-based Location Strategy and its Performance, *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 8, pp. 1415-1424, 1997.

[11] C. Rose and R. Yates, "Minimizing the Average Cost of Paging under Delay Constraints", *Wireless Networks*, vol. 1, no. 2, pp. 211-219, July 1995.

[12] C. Rose, "Minimizing the Average Cost of Paging and Registration: A Timer-based Approach", *ACM Wireless Networks*, vol. 2, no. 2, pp.109-116, June 1996.

[13] N. Farvardin and J. Modestino, "Optimum Quantizer Performance For a Class of Non-Gaussian Memoryless Sources", *IEEE Transactions on Information Theory*, pp. 485-498, May 1984.

[14] J. Ziv and A. Lempel, "Compression of individual sequences via variable-rate coding," *IEEE Transactions on Information Theory*, vol. 24, no. 5, pp.530-536, September 1978.

[15] S. Rajagopalan and B. Badrinath, "An Adaptive Location Management Strategy for Mobile IP", *Mobile Computing and Networking*, pp. 170-180, 1995.