A Pattern Recognition System for Handoff Algorithms

K. Daniel Wong, *Member, IEEE*, and Donald C. Cox, *Fellow, IEEE*

***Abstract*—In wireless cellular systems, handoff algorithms decide when and to which base station to handoff. Traditional handoff algorithms generally cannot keep both the average number of unnecessary handoffs and the handoff decision delay low. They do not exploit the relative constancy of path loss and shadow fading effects at any given location around a base station. This information can in fact be used to improve the efficiency of handoff algorithms, as we do in our new handoff algorithms using statistical pattern recognition. Handoff algorithms with both a negligible number of unnecessary handoffs and a negligible decision delay can therefore be realized.**

***Index Terms*—Cellular, handoff, handover, pattern recognition, wireless.**

I. Introduction

THE ABILITY to communicate while in motion is at the heart of the current popularity of cellular service. In order to provide these mobile communication services, cellular systems need to provide handoff services to users.[1] Handoff refers to any change in wireless channel used. It is alternatively known as handover [1] or automatic link transfer (ALT) [2]. There are generally two main reasons for handoff. Handoffs are performed for link quality maintenance, and for reducing interference in the system. Handoffs therefore support the provision of mobility and are an essential component of cellular systems.

Traditional handoff algorithms (THO) in existing systems generally use relative signal strength as a main component of the handoff decision process. The relative signal strength of a base station is the difference in signal power level associated with that base station from that associated with the serving base station. Whenever the relative signal strength of a given base station rises above a threshold, Δ dB, handoff is performed to that base station. Because of fading effects, the relative signal strength of base stations other than the serving base station can be positive for brief periods of time when it may not really be necessary to handoff to those base stations. Handoff in such cases wastes network resources and is undesirable. Therefore, the threshold Δ is typically set at a few dB, introducing hysteresis into the handoff process, to reduce the frequency of unnecessary handoffs. While hysteresis reduces the frequency of unnecessary handoffs, it also increases the decision delay. Large decision delay is undesirable, like frequent unnecessary handoffs. The smaller Δ is, the more frequent the unnecessary handoffs, but the larger Δ is, the larger the decision delay. There is therefore an inherent tradeoff between number of unnecessary handoffs and decision delay with THO [3]. In terms of the handoff algorithm performance measures to be introduced in Section II-A, THO cannot be *accurate* and *timely* at the same time.

We introduce new handoff algorithms to overcome the shortcomings of THO discussed above, and to meet the additional challenges presented to handoff algorithms by microcellular systems. The new handoff algorithms use statistical pattern recognition techniques to exploit a characteristic of radiowave propagation which is not exploited by THO. In particular, with THO, the use of previous knowledge about the propagation characteristics of the environment is limited and general. Our pattern recognition handoff algorithms (PRHO), however, train on the signal strength measurements at the locations in which handoffs may be desired, and hence acquire specific knowledge about the propagation characteristics at these locations. Specifically, the signal strength measurements are comprised of relatively deterministic components (path loss and shadow fading) and random components (Rayleigh fading) whose means are the deterministic components. The reasonable assumption is made that each time a user travels along the same stretch of road, the measured signal power from a particular base station[2] has the same unique deterministic component, but a different random component. Pattern classes can therefore be set up to yield a one-to-one correspondence with a user being at or near these locations. Matching these patterns could accurately signal the need for a handoff, without incurring much delay. Hence, pattern recognition techniques can be applied to provide more accurate and timely handoff decisions than is possible with THO. Handoff algorithms with few unnecessary handoffs and low decision delay can thus be realized.

Previous work on using pattern recognition techniques for handoff-related purposes include [4]–[6]. The pattern classes in [5] and [6] are associated with *serving base stations*. As long

Manuscript received February 7, 1999; revised November 3, 1999. This research was supported in part by Nortel and the Stanford University Center for Telecommunications.

K. D. Wong is with Telcordia Technologies, Red Bank, NJ 07701 USA (e-mail: dwong@research.telcordia.com).

D. C. Cox is with Stanford University, Stanford, CA 94305 USA (e-mail: dcox@nova.stanford.edu).

Publisher Item Identifier S 0733-8716(00)05283-5.

[1]By "user," we mean equipment that uses a cellular system, such as a cellular phone. "Mobile," "mobile terminal," and "subscriber unit" are other synonymous terms used in the literature.

[2]In some configurations, signal strength measurements of the signals from users are made at base stations. Throughout this paper, whenever we refer to measurements at users *from* particular base stations, the statements can apply to these alternative configurations as well, by simply reversing the direction of the measurements.

as a user's measured signal strength patterns are classified as belonging to the pattern class associated with its serving base station, it does not handoff to another base station. Once the patterns are classified as belonging to the pattern class associated with another base station, the algorithms might initiate, or begin considering, handoff to the other base station. This differs from our work in that our pattern classes are associated with *stretches of street*. Reference [4] uses hidden Markov models for pattern recognition of signal strength patterns for user location. It also mentions that the proposed pattern recognition techniques could possibly be used for various system functions like handoff. More recent work on pattern recognition for handoff algorithms is described in [7], which uses probabilistic neural networks and attempts much more extensive tracking of user location than do the algorithms in this paper.

Handoffs which are consistently both accurate and timely can result in higher capacity and better overall link quality than what is available with today's systems. Furthermore, with continual demands for more system capacity, there is a trend toward smaller cells, also known as microcells. Handoffs are more critical in systems with smaller cells, because for a given average user speed, handoff rates tend to be inversely proportional to cell size.

The rest of this paper is structured in four parts. Section II states the problem we are addressing, discusses issues involved, and briefly surveys previous work on handoff algorithms. We have briefly introduced the concepts of using statistical pattern recognition to recognize when a particular handoff may or may not be necessary, to improve the performance of handoff algorithms. These pattern recognition concepts are formalized, more carefully defined and explained in Section III. Performance and simulation results of the new algorithms are discussed in Section IV.

II. The Handoff Decision Problem

A. Toward Quantifying Performance of Handoff Algorithms

The main objectives of handoffs are link quality maintenance and interference reduction. There is a network cost with each handoff that occurs. Therefore, a third objective of handoff algorithms is keeping the number of handoffs low. Consequently, to meet all three objectives, a handoff algorithm should initiate a handoff if and only if the handoff is necessary. We define the *accuracy* of a handoff algorithm to refer to how well the algorithm does this. An accurate handoff algorithm is one which initiates handoffs only when necessary and does not initiate handoff when handoff is not necessary. The timing of the handoff initiation is also important. There can be deleterious effects on link quality and interference is the initiation is too early or too late. We define the *timeliness* of a handoff algorithm to refer to how well the algorithm times the handoff. A timely handoff algorithm is one which initiates handoffs neither too early nor too late.

1) Comparing Handoff Algorithms—The Canonical Handoff Scenario: The canonical handoff scenario is a simple but important handoff situation that has been studied in some detail in the literature (for example, [8]–[10]). Illustrated in Fig. 3, it involves two base stations in the same street separated by some distance, with possible obstructions like trees. The effects of interference from other base stations or users is usually not considered.[3] There is only one user, moving in a straight line between the two base stations at constant speed, starting at one base station and ending at the other. The two arrows labeled "Pattern 1" and "Pattern 2" have to do with pattern placement when PRHO is used, and may be ignored at this time. Simulation comparisons of PRHO with THO in the testbed of the canonical handoff scenario will be discussed in Section IV.

2) Heuristic Choices: Both large-scale and small-scale fades are frequently encountered in mobile wireless communications [12]. As a result, the following challenges arise.

1) It is difficult for handoff algorithms to be both accurate and timely at the same time. The use of a hysteresis threshold, Δ, by THO (traditional handoff algorithms) was introduced in Section I. THO trades off accuracy (for large Δ) with timeliness (for small Δ).
2) There is a challenge in quantifying accuracy or timeliness, i.e., to define objectively optimal answers, for a given situation, to the following questions. How many handoffs should occur, and to which base stations? When should the handoffs occur?

In this section, the focus is on the second of the above points. Some quantifying of accuracy or timeliness is necessary in order for comparisons to be made between different handoff algorithms. Note that it is not always obvious when the handoff(s) should occur, and to which base station. Even if perfect knowledge of various measurements, both past and future, are available, it still depends on the relative importance of the three objectives mentioned above, and on other system factors. Therefore, quantifying of handoff performance is often heuristic, based on rules of thumb which are generally useful.

In the canonical handoff scenario, one handoff should usually occur from BS 0 to BS 1 somewhere in between the two base stations. So the number of handoffs that occurs between BS 0 and BS 1 is the chosen performance measure in some papers [5], [10], regardless of where the handoffs occur. Other papers also consider timeliness of the handoffs to be important. Various conceptions of the optimal handoff location include: a) the geometrical midpoint of the two base stations; b) a location specified by a (deterministic) function of signal strength along the path [9], [13]; and c) a location specified probabilistically in terms of signal strength statistics and other quantities [3], [14].

B. A Framework for Accuracy and Timeliness

Our chosen criterion for accuracy and timeliness is defined in the following framework. A given user travel path is broken down into N_s segments, where N_s is large enough that local mean signal strength from any base station remains relatively constant in each segment; and at most one handoff can reasonably occur in each segment. The base stations which may be involved in handoff are ordered and numbered, e.g., BS 0, BS

[3]The effects of interference *can* be included in the model, as in Austin and Stüber [11], which is an extension of Vijayan and Holtzman's model for the canonical handoff problem [3].

1, etc. A handoff *location–destination pair* is a segment index and a base station index indicating the occurrence of a handoff to the corresponding base station in the corresponding segment. A handoff sequence is a sequence of location–destination pairs that describes the handoff behavior of the user travelling along the given path. An optimal handoff sequence can be defined as one which maximizes a cost-reward function [15]:

$$r(\mathcal{Y}, \boldsymbol{S}) = \max_j \sum_{s=0}^{s=N_s-1} \boldsymbol{S}_{B_{js},s} - C \times H(\mathcal{Y}_j) \qquad (1)$$

where

$H(\mathcal{Y}_j)$	is the number of handoffs in sequence $\mathcal{Y}_j$,
C	is the cost of doing a handoff,
s	is the segment index,
j	is the sequence index,
B_{js}	is the serving base station at segment s of the sequence $\mathcal{Y}_j$, and
$\boldsymbol{S}_{i,s}$	is the local mean signal strength of base station i at segment s.

The summation term would be maximized if the "best" base station (with highest local mean signal strength) is the serving base station in every segment. However, maximizing the summation in this way may imply many handoffs, and the negative term reduces the attractiveness of sequences with many handoffs.

C. Real-Time Decisions

Handoffs algorithms operating in real time have to make decisions without the luxury of repeated uncorrelated measurements, or of future signal strength information. Therefore, it is difficult for the algorithms to maximize (1). Nevertheless, in our comparisons of the performance of PRHO and THO, (1) is used as the "objective" standard for accurate and timely handoff behavior, providing a basis for comparison of the algorithms.

III. PRHO Design, Training, and Usage

Signal strength patterns are collections of signal strength measurements. The patterns consist of measurements of consecutive samples of signal strength measured from one or more base stations as a user is moving along streets in a cellular system. The structure in such signal strength patterns, and in particular their repeatability, motivates the use of signal strength patterns to provide user location information which may be useful as input to the handoff decision process. Although other quantities like estimates of bit error rate (BER), frame error rate, and signal-to-interference ratios can also be used as inputs to a handoff decision process, in this paper we focus on the use of signal strength measurements. This is because they are relatively easy to measure and because the techniques presented here can be easily extended to working with patterns incorporating other quantities in addition to signal strength.

The fundamental basis of our approach is discussed in Section III-A. The assumptions we make and the terminology we use in discussing signal strength patterns are introduced in Sections III-B and III-C. The sliding window pattern classification of signal strength patterns is developed in Section III-D, where our template-based statistical pattern recognition technique is introduced.

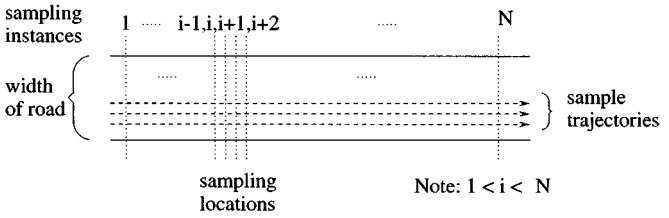


Fig. 1. Sample trajectories of a measurement vehicle used in training to obtain samples of signal strength to estimate optimal handoff location–destination pairs. The sampling locations are also illustrated.

A. The Basis of Our Approach

Large-scale fading effects in cellular or microcellular systems typically consist mainly of reflection, refraction, and scattering from objects like buildings and trees, which are relatively stationary over periods of days or weeks. Therefore, for any given location in a cellular system, the local mean signal power level is relatively fixed. Furthermore, users in cellular or microcellular systems in cities tend to move in very regular, determined paths, namely along lanes of streets or along sidewalks. These two factors result in considerable *regularity* of structure, and *repeatability* in the signal strength measurements (of beacon signals of base stations) which are made by users in cities.

Consider a situation in which a user moves along a particular stretch of street, and then another user a little while later moves along the same stretch of street. Both users make signal strength measurements from several base stations around them as they move along the street. Let the measurements made by the first user be consecutive samples of signal strength denoted by $\{\gamma_{0,i}\}$ for $i = 0, 1, \ldots, N-1$, and the measurements made by the second user be denoted by $\{\gamma_{1,i}\}$ for $i = 0, 1, \ldots, N-1$. If the sample points are aligned in space, as those in any two sample trajectories in Fig. 1 are, then $\{\gamma_{0,i}\}$ and $\{\gamma_{1,i}\}$ can be modeled as random vectors, both of which have the same N-dimensional joint distributions. Consider a third set of measurements, $\{\gamma_{2,i}\}$, made along any other stretch of street.

Given an appropriately chosen metric space, it is expected that the measurement vectors possess two fundamental properties: the distance between $\{\gamma_{0,i}\}$ and $\{\gamma_{1,i}\}$ is small; and the distances between $\{\gamma_{0,i}\}$ and $\{\gamma_{2,i}\}$, and between $\{\gamma_{1,i}\}$ and $\{\gamma_{2,i}\}$, are both significantly larger than the distance between $\{\gamma_{0,i}\}$ and $\{\gamma_{1,i}\}$. This regularity and repeatability in the signal strength measurements is not exploited in current cellular systems. We have investigated ways to exploit it using statistical pattern recognition techniques.

B. Assumptions About the Signal-Strength Samples

Consecutive signal strength samples of the small-scale fading are approximately independent if spaced more than half a wavelength apart [12]. Since our algorithm is designed to work with independent samples of the small-scale fading, it is desirable for the spacing between consecutive samples to be greater than half a wavelength. However, the spacing should not be very much greater than half a wavelength, in order not to lose useful data. The spacing between consecutive samples should therefore be fixed and small, and have dimensions of length. Note that this is *spatial sampling*, not the more common *temporal sampling*, where the spacing between consecutive samples has

dimensions of time. Another reason for using spatial sampling is that it makes the pattern classification simpler, since samples are easier to "align" in doing pattern matching.

Spatial sampling is feasible if user velocity information is available. This may be obtained if there were a device attached to the odometer of a vehicle. However, this would not be possible for pedestrian microcellular systems. Alternatively, other velocity adaptive sampling schemes could be used to obtain spatial samples. The level or zero crossing rates of the in-phase and quadrature components of the envelope of the received signal are functions of user velocity [12]. Reference [16] describes velocity estimation techniques using level crossing rates (or zero-crossing rates). Vehicle velocity is also proportional to the maximum Doppler frequency, f_m, of the received signal [12]. Therefore, estimates of maximum Doppler frequency can also be used to estimate user velocity. One estimator [8] uses a measure of squared deviations of signal strength sample measurements in dB, and is based on the properties of the covariance of adjacent signal strength samples. Yet another method for maximum Doppler frequency estimation [8], and consequently for vehicle velocity, is performing a spectrum or autocorrelation analysis and inferring f_m from this.

To reduce the large variation in signal strength caused by small-scale fading, the measured samples are averaged. The term *signal strength indicator* (SSI) will describe the preprocessed sample values (spatial samples averaged over a few samples, e.g., over 6 samples for the simulations discussed in Section IV). SSI will be in dB unless otherwise stated. SSI can be related to the uplink and measured by a base station, or related to the downlink and measured by a user. We will use "SSI associated with a base station" to mean either of these cases when the difference is irrelevant to a particular discussion. Note that we do *not* assume that small-scale fading is "averaged out." Although a convenient assumption to make, it is not a good model of reality [17]. Note that SSI can be used as the basis of signal strength patterns because it shares the structural regularity properties of unaveraged signal strength measurements, but with less variability.

C. Definitions, Notation, and Terminology

A *pattern* is "a distinctive arrangement of structural elements" [18]. For the case of signal strength patterns, the structural elements are the SSI's, and the distinctive arrangements are a result of the structural characteristics of local mean signal power level (Section III-A) that we exploit. We define a *feature matrix* to be an $\mathcal{M} \times N_p$ matrix of SSI's with the following properties: each row contains consecutive SSI's associated with one base station, and the SSI's in any two different rows are measured at approximately the same time. The space in which feature matrices lie is the *feature space*. Feature matrices are a compact and convenient representation of signal strength patterns. The two fundamental properties of the signal strength patterns (Section III-A) enable pattern classes to be defined according to similarity or closeness according to some distance metric. Members of a *pattern class*, then, belong to a set of patterns each of whose feature matrix lies within a region in feature space. Our *pattern classification* problem consists of identifying the region in feature space into which a given pattern falls.

Each pattern class is associated, in one-to-one correspondence, with three objects: a stretch of street, a set of base stations, and a pattern action. Members of each class can be represented by feature matrices which contain SSI's from certain base stations within the corresponding set, made along that corresponding stretch of street. The unique pattern class specification of a specific pattern class specifies the following.

- *Pattern Stretch:* This is a stretch of street (N_p samples long) for which the system wants to be informed whenever a user traverses it. For example, this pattern may signal the need for a handoff-related action. Representative members of the class (template patterns of signal strength measurements) are obtained by making measurements along the pattern stretch. The components of the template patterns are the SSI vectors measured from surrounding base stations while a user is traveling along that pattern stretch. The list of base stations included, and the ordering of the base stations, are specified in the *included base station set*, defined below. For convenience, *locating a pattern* at a particular location will sometimes be used to mean defining a pattern class whose pattern stretch is at that location. Pattern stretches are inherently directional, going from the *stretch start* to the *stretch end*.
- *Pattern Action:* This is the action associated with the pattern class, such as "Change billing rate" or "Reduce transmit power." The action may also be handoff-related, such as "Handoff from base station 0 to base station 1."
- *Included Base Station Set:* This is a set of $\mathcal{M}$ base stations. Vectors of SSI measurements from these $\mathcal{M}$ base stations made while traversing the pattern stretch can be arranged into a feature matrix of the pattern class as follows:

$$\boldsymbol{P} = \begin{bmatrix} \text{SSI vector of 1st base station} \\ \text{SSI vector of 2nd base station} \\ \vdots \quad \vdots \quad \vdots \\ \text{SSI vector of } \mathcal{M}\text{th base station} \end{bmatrix}. \tag{2}$$

A few other useful concepts and some other terminology are as follows.

- *Pattern Instance*[4]: A pattern instance is a feature matrix obtained by observing the SSI vectors from the included base station list over a pattern stretch. Types of pattern instances include the following.
  1) *Training pattern instance* or *template*. This is a pattern instance which is observed in training. If there are several templates (e.g., L, where $L > 1$) for a particular class, they will be indexed by the *instance number* $(1, \ldots, L)$.
  2) *Sliding window pattern instance*. This is a pattern instance within a block of parallel sliding windows, as illustrated in Fig. 2 by the sliding window pattern instance $\boldsymbol{P_0}$. Fig. 2 is described in Section III-D.

[4]This is also known as a *sample* in pattern recognition terminology. But we use this term to avoid confusion between samples in this sense and samples in the sense of individual signal strength measurement samples.

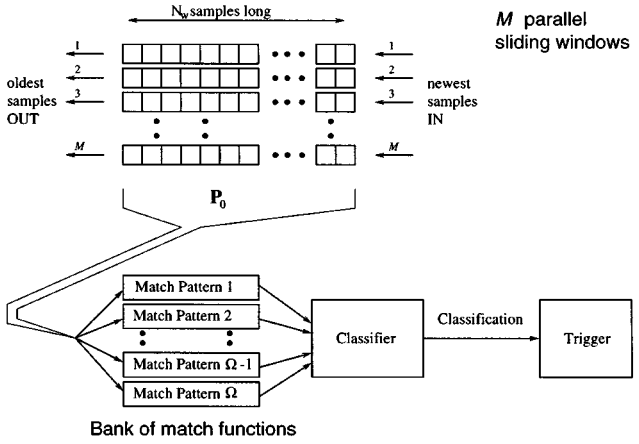


Fig. 2. How patterns from the parallel sliding windows are matched with stored patterns.

- *Pattern set.* This is the set of all pattern classes associated with a particular base station. These pattern classes are the ones whose pattern actions apply to users for whom the particular base station is the serving base station. Distributed pattern classification, with independent processing in the different base stations, can be implemented, as discussed in Section III-D. Each base station handles the patterns in its own pattern set. The size of the pattern set of base station x is denoted by $[\Omega]_x$.

Elements of pattern matrices are represented by $\boldsymbol{P}^{(c)}_{l,m,n}$, where $c \in 1,\ldots,\Omega$ is the pattern number, $l \in 1,\ldots,L$ the index of the pattern instance (or $l = 0$ if it is not a training pattern instance), $m \in 1,\ldots,\mathcal{M}$ the base station index, and $n \in 1,\ldots,S$ the sample index ordered from the stretch start to the stretch end. Then SSI vectors are denoted by $\boldsymbol{P}^{(c)}_{l,m}$ and pattern instances are denoted by $\boldsymbol{P}^{(c)}_{l}$. Where necessary to distinguish between patterns or variables refering to different pattern sets, we write $[\,]_q$ around the pattern or variable, where q is the pattern set index. Examples are $[\boldsymbol{P}^{(c)}_{l,m,n}]_q$ and $[\Omega]_q$.

D. Sliding Window Pattern Recognition

Sliding window pattern recognition can be described in three stages.

- *Design stage:* the pattern stretches are picked, perhaps after optimal handoff location–destination pairs are estimated.
- *Training stage:* for each pattern stretch, training pattern instances are observed and stored as templates of the class.
- *System operation:* whenever the sliding window is updated, the sliding window pattern instance is classified. A handoff-related action may then be initiated.

Each stage is elaborated upon below.

1) Design: Certain locations in the system are identified, for which PRHO is desired. These may be stretches of street for which THO performs poorly. In order to train PRHO, desired handoff location–destination sequences must be chosen. These may be chosen based on (1). However, the optimal handoff location–destination sequence as defined by (1) cannot be accurately found in a real system, because small-scale fading makes $\boldsymbol{P}$ difficult to obtain. Hence $\boldsymbol{P}$, and consequently, $r(\mathcal{Y},\boldsymbol{P})$, must be estimated. This may involve making several training runs through the street and making measurements at the same set of locations each time (on-the-fly training is an alternative which will be elaborated in the following section). Local mean signal power level estimators are employed to estimate the elements of $\boldsymbol{P}$. The optimal (MMSE unbiased) local mean signal power level estimator for Rayleigh fading environments [17] is useful for this purpose. Practical procedures for estimating (1) using estimates of $\boldsymbol{P}$, including efficient search algorithms to perform the maximization in (1), can be found in [15].

2) Obtaining the Training Samples: For each location where a handoff is from a particular base station is desired, a pattern class is defined in that base station's pattern set, where: a) the pattern stretch is located along the anticipated travel path, with the stretch end at the desired handoff location; b) the pattern action is a handoff to the corresponding base station; and c) the pattern templates are obtained by special training/measurement runs along the pattern stretch.

a) Alternative design/training: If making special training/measurement runs through streets is prohibitively inconvenient, an alternative is to obtain the estimates during prior actual system usage as users travel down the street and handoffs are initiated using threshold techniques. New algorithms would need to be developed to extract patterns from such measurements, through clustering or other means. The advantages of this alternative are that special training/measurement runs are unnecessary and that the patterns may learn to be more responsive to actual system handoff needs. The disadvantages are added complexity (the training module needs to be more intelligent, for instance) and difficulty in also using the patterns for user location applications like E-911. In this paper, we focus on the approach where special training/measurement runs are required.

3) System Operation: Each base station oversees its pattern set of Ω patterns. The pattern set is different for each base station, which handles pattern classification for only the pattern classes in its pattern set, independent of the other base stations. The included base station set of all Ω patterns in a base station's pattern set is the same set of $\mathcal{M}$ base stations (however, the set may be different from that of the patterns of other base stations), but the patterns are otherwise independent, with their own pattern stretches and actions. Several training pattern instances of each class are used as templates.

Fig. 2 illustrates how sliding window pattern recognition works in the system operation phase. For each user, a sliding window of SSI measurements is kept at the serving base station. At regular intervals, the sliding window is updated, as the oldest samples from the $\mathcal{M}$ base stations are discarded and a fresh set of samples is transmitted from the user terminal to replace them. The contents of the sliding window are a sliding window pattern instance, and an attempt is made to classify it as belonging to one of the Ω pattern classes, or none of them (the null classification). The null classification will normally be very likely, especially if Ω is small, because most of the time the user will then not be in one of the locations corresponding to a pattern class.

Decision rules like the nearest neighbor decision rule [19] can be used for classification. The classification method we use has

a nearest neighbor decision rule with a null classification option. The null classification is made whenever distance to the nearest neighbor exceeds a *match threshold*. Otherwise, the class of the nearest neighbor is selected. The selected distance metric between points in pattern space is a normalized squared Euclidean distance. The distance of a sliding window pattern instance, $\boldsymbol{P}_0$, to a pattern class, $\boldsymbol{P}^{(c)}$, is taken to be the minimum of the normalized squared Euclidean distances between $\boldsymbol{P}_0$ and each of the L members (templates) of $\boldsymbol{P}^{(c)}$:

$$M\left(\boldsymbol{P}^{(c)}, \boldsymbol{P}_0\right) = \frac{1}{N_p \mathcal{M}} \min_{l \in 1,2,\ldots,L} \sum_{m=1}^{\mathcal{M}} \sum_{n=1}^{N_p} \left(\boldsymbol{P}^{(c)}_{l,m,n} - \boldsymbol{P}_{0,m,n}\right)^2. \quad (3)$$

The distance computations may be performed by *match functions*. Fig. 2 illustrates a bank of match functions, one for each of the Ω pattern classes used as input to the classifier. If the sliding window pattern instance is classified as belonging to a particular pattern class, instead of the null class, the information is passed to the trigger. The trigger implements the pattern action associated with that pattern class.

4) Pattern Classification Variations: The basic classifier compares the results of all the match functions with a match threshold. When there is none below the threshold, the classification is the null class (no match to any particular class), and no action is taken. When there are one or more below the threshold, the smallest is picked. A variant classifier that is used in our simulations has one additional criterion. Treating each base station component of each pattern as a separate subpattern, we require that each subpattern also has to satisfy a match threshold criterion.

IV. Handoff Decisions Using Pattern Recognition

THO and PRHO are now compared in a common handoff situation, the canonical handoff scenario, previously introduced in Section II-A. For THO, handoff occurs to the other base station when the relative SSI of that base station (i.e., relative to that of the serving base station) exceeds the hysteresis threshold.

A. Pattern Placement

For fair comparison with THO, two pattern classes are associated with the canonical handoff problem. One of them, $[\boldsymbol{P}^{(c_0)}]_0$, is the c_0th pattern in the pattern set of base station 0 (BS0) and its pattern stretch points from BS0 to base station 1 (BS1), for handing off to BS1. The other, $[\boldsymbol{P}^{(c_1)}]_1$, is in the pattern set of BS1 and its pattern stretch points in the reverse direction, for handing off to BS0, as shown in Fig. 3. For convenience, $[\boldsymbol{P}^{(c_0)}]_0$ and $[\boldsymbol{P}^{(c_1)}]_1$ are refered to simply as "Pattern 1" and "Pattern 2" in the figure. Having these two patterns provides for fairer comparison with THO than if there were only one pattern class, just for handing off to BS1, because multiple back-and-forth handoffs can occur with PRHO in this setup (when match thresholds are set too large), as with THO. The patterns are located on either side of the designated handoff point, which is the optimal handoff location according to (1).

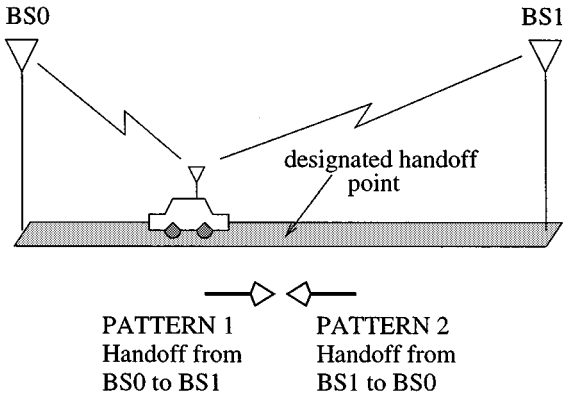


Fig. 3. The Canonical Handoff Problem Setup for PRHO. Without the patterns in the picture, this would be the canonical handoff scenario which is often used in the literature as an unofficial testbed for handoff algorithm performance. The user moves in a straight line between two base stations, and one handoff somewhere near the midpoint of the two base stations is desired.

TABLE I
Selected Simulation Parameters for PRHO Simulations

Category	Parameter	Value
Geometrical	Spatial sampling interval	1/6 meters
	(City) Block length	100.0 meters
Algorithm for PRHO	Sliding window size, N	50 SSI samples
	Size of pattern sets, Ω	4
	Match threshold, T	variable
	Number of base station components, $\mathcal{M}$	5
Propagation	Standard deviation (σ) of lognormal fading	3.0
Preprocessing	Number of independent samples in input	6
Simulation	Multipath fading tr. per shadow fading tr.	20
	Shadow fading traces	1000

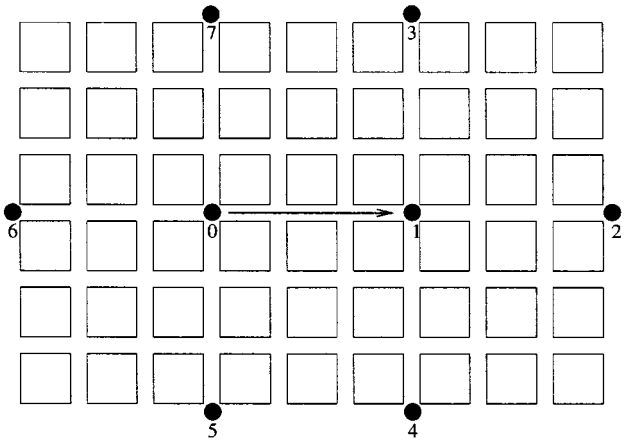


Fig. 4. Base station arrangement for computer simulations. The squares are city blocks and the circles are base stations. The arrow between BS 0 and BS 1 indicates the path of motion of the user. $\mathcal{M} = 5$ for both base stations. When the serving base station is BS0, the included base station set is $\{0, 1, 5, 6, 7\}$. When the serving base station is BS 1, the included base station set is $\{0, 1, 2, 3, 4\}$.

B. Performance Comparisons of PRHO and THO

Simulations were run to compare the performance of PRHO with that of THO. The results demonstrate that PRHO performs very well in comparison with THO. Table I lists selected simulation parameters. The base station arrangement is shown in Fig. 4. In the simulations, handoffs happen instantaneously (no network delays). The propagation model used is from [20]. For each simulation (involving one set of parameter settings), 1000

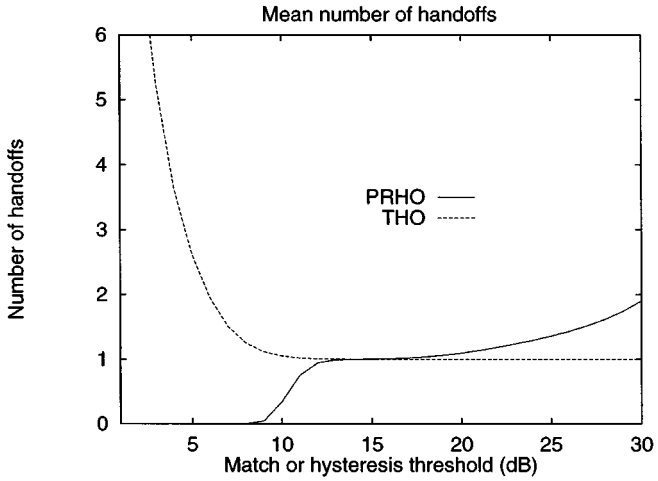


Fig. 5. Mean number of handoffs versus either match or hysteresis threshold (depending on algorithm used).

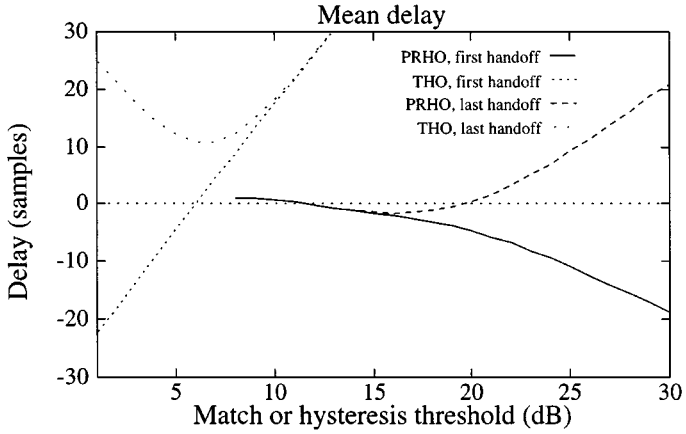


Fig. 6. Mean delay versus match or hysteresis threshold (depending on algorithm).

different shadow fading traces (different "streets") are generated, and for each of those, 20 different multipath fading traces are generated. The algorithms (for particular parameter settings) are simulated in these fading conditions and various statistics are collected. The statistics include delay and number of handoffs, which will be explained shortly. Our default choice of $\sigma = 3$ for our simulations, while appearing small, is actually a "worst case" value. In general, larger values of σ help the performance of PRHO because the forward and backward pattern would be expected to be further apart in pattern space. Other simulations have confirmed this expectation, as well as the utility of the SSI averaging length of 6 for both PRHO and THO. For PRHO, the tradeoff in choice of SSI averaging length is that for too small a value, algorithm performance is severely degraded by the increased variability but for too large a value, the patterns become too long. For THO, the tradeoff is between too much delay and too many handoffs.

Mean number of handoffs (occurring while a mobile station is moving along a canonical handoff track) is plotted versus threshold for both THO and PRHO in Fig. 5. For THO, the threshold referred to is the hysteresis threshold, whereas for PRHO, the threshold referred to is the pattern match threshold. The two thresholds are different quantities and should not be confused with or compared with one another. Instead, we compare the performance and sensitivity of the two types of algorithms with respect to their respective thresholds.

It can be seen from Fig. 5 that there are multiple handoffs performed when THO is used with hysteresis thresholds of less than 10 dB. Above 10 dB, the "ping-pong" effect is practically eliminated. For PRHO, there are two undesirable regions and one desirable region. Below about 10 dB, the match threshold is too low to match for necessary handoffs, leading sometimes to complete failure to handoff. Above 15 dB, the threshold is too high to avoid "ping-pong" by alternatively matching $[\boldsymbol{P}^{(c_0)}]_0$ and $[\boldsymbol{P}^{(c_1)}]_1$, sometimes erroneously. In the desirable region, approximately between 10 and 15 dB, there is only one handoff.

From the plots of mean number of handoffs alone, the advantage of using PRHO is not apparent. In fact, it appears more cumbersome to use, because there is a desirable range of threshold settings in between two undesirable ranges, whereas THO eliminates unnecessary handoffs for high enough thresholds. However, the difference becomes clear in the delay plots. We define handoff delay to be the number of samples after the optimal handoff location that it takes for the handoff to occur. If a handoff occurs before the designated handoff point is reached, it is assigned a negative delay value. *First-handoff delay* is the number of segments after the optimal handoff location at which the first decision is made to handoff, and *final-handoff delay* is the number of samples after the optimal handoff location at which the last decision is made to handoff. The interval between the first and final handoff is nonzero if multiple back-and-forth handoffs occur.

In Fig. 6, it can be seen that PRHO is clearly better in terms of delay. For small values of the match threshold, no handoff occurs, and so no delay value is plotted. In the region where the first-handoff delay and final-handoff delay line coincide, only one handoff occurs. For PRHO, there is no delay in the decision process. In fact, there is a slight negative delay caused by early handoffs. The slightly early handoffs are because of pattern matches being declared before the actual match points, because a threshold is used. Only for larger values of match threshold are there multiple handoffs, but even then with little delay. THO, on the other hand, cannot avoid having either multiple handoffs or excessive delay. When the threshold is low, multiple handoffs occur, and when the threshold is high, delay is large.

The mean number of handoffs is plotted parametrically against the mean handoff delay in Fig. 7. There are two curves for each handoff algorithm, representing first and last handoff delays as in Fig. 6. For the THO curves, the parameter varied is the hysteresis threshold. For the PRHO curves, the parameter varied is the match threshold. The point where either $T = 1$ or $T = 30$ (or 20, for THO) is labeled. The threshold T is the parameter which varies from 1 to 30 (20 for THO) along each curve. Each of the printed points on the curves represents an integer value of T. Of the two algorithms, it is PRHO which gets closest to the desired region around (0, 1) in the plot, i.e., no delay and one handoff. The plots labeled THO in these parametric plots resemble plots in [21] which discusses THO.

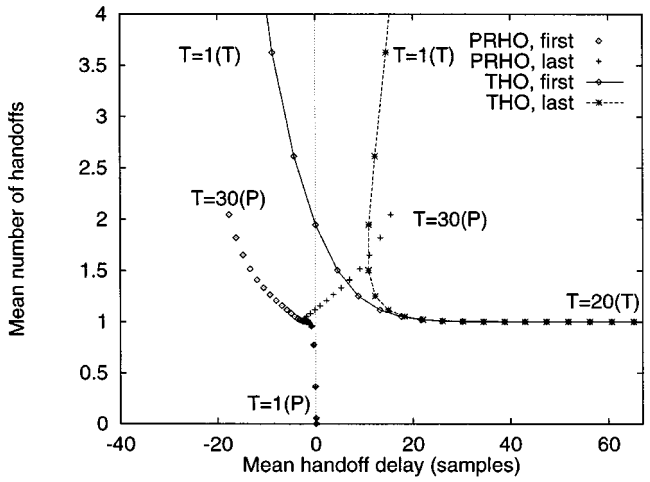


Fig. 7. Mean number of handoffs versus Mean handoff delay.

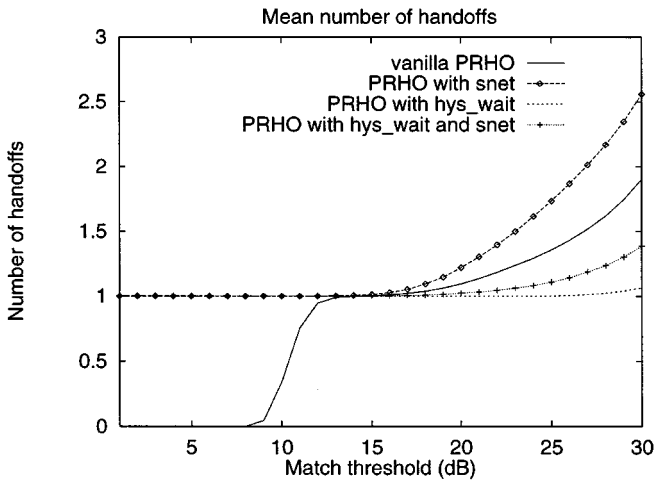


Fig. 8. Refinements in PRHO.

Reference [21] suggests that the optimum threshold setting for THO is where the "knee" of the curve is, i.e., $T \approx 7$ from Fig. 7 in this case. Finding the "knee" of the curve is not easy, but [21] suggests that points around it might be acceptable too.

Two advantages of PRHO over THO are readily seen.

1) PRHO is more robust than THO. For a reasonable range of match thresholds, the mean number of handoffs is clustered around one, with very little delay. As the match threshold increases, the mean number of handoffs increases, but not very rapidly. Picking one of those points appears less sensitive to error than finding the "knee" of the curve for THO. It should therefore be easier to train and adjust thresholds for PRHO than for THO, and PRHO can be expected to be more forgiving of slight parameter misadjustments than can THO. For low values of (PRHO) match threshold, matches are missed and the mean number of handoffs is below one. Therefore, low values of match threshold should be avoided.
2) If the optimal settings for THO can be found, performance is still not quite as good as PRHO, although it can be somewhat close.

C. Refinements

The basic version of PRHO has the problem that, on a few occasions, the classifier fails to classify the situation correctly when the user is near the pattern stretch. As a result, the handoff action is never triggered and the call would eventually be dropped. Also, depending on the match threshold settings (as seen in Fig. 5, for example), there might be one or more additional handoffs back and forth between the base stations. Some of these additional handoffs may be unnecessary, and may be the result of misclassification. This is especially bad when the user ends up handing off back to the original serving base station, in which case the call may eventually be dropped. For illustrative purposes, consider a match threshold of 15 dB, and the rest of the parameters as in Table I. Our simulations have found that the probability of no handoff is approximately 2×10^{-4}, and the probability of two handoffs occuring (the second one back to the original serving base station) is approximately 4×10^{-3}.

One solution is to use a loose relative-signal-strength-based threshold as a failsafe mechanism. This is a "safety net" to catch the errors that overcomes both problems mentioned above. The mean number of handoffs of the algorithm with this failsafe mechanism (*PRHO with snet*) is shown together with the mean number of handoffs of the basic PRHO described earlier (*vanilla PRHO*) in Fig. 8. As expected, *PRHO with snet* has more handoffs occurring, since there is always a final handoff occurring to correct one of the abovementioned cases if necessary. In the region where the match threshold is too low for meaningful pattern classification, the failsafe mechanism is the reason why the mean number of handoffs is one. With a well-chosen match threshold (around 15 dB), the failsafe mechanism is seldom, if ever, triggered.

In a sense, the comparison with THO so far has been unfair to PRHO. With THO, there is hysteresis to inhibit handing off back to the previous serving base station after a handoff occurs, whereas we have not included hysteresis with PRHO. In fact, there is more reason to inhibit backward handoffs with PRHO, since there is more certainty about the correctness of the decision to handoff than with THO. One way to add hysteresis to PRHO is to set a tighter match threshold on the pattern for handoff back to the previous serving base station for a period of time after a handoff. The plot in Fig. 8 shows this hysteresis-wait mechanism (*PRHO with hys_wait*) with the tighter match threshold being 5 dB below the regular match threshold and the wait period being twice the size of the sliding window. This mechanism significantly reduces the number of unnecessary handoffs even for high match thresholds.

Finally, the performance of a version of PRHO with both the hysteresis-wait and failsafe mechanisms is also shown in Fig. 8 (*PRHO with hys_wait and snet*). For a match threshold of 15 dB, the mean first handoff delay is -1.9284, with standard deviation of 3.88, and the mean last handoff delay is -1.8356, with standard deviation of 4.1774. This combined algorithm has practically no unnecessary handoffs and negligible delay.

1) The Side-Street Scenario: Another handoff scenario in which PRHO has been simulated is the side-street scenario shown in Fig. 9. The user is not moving in a straight line between two base stations as before. The region where handoff

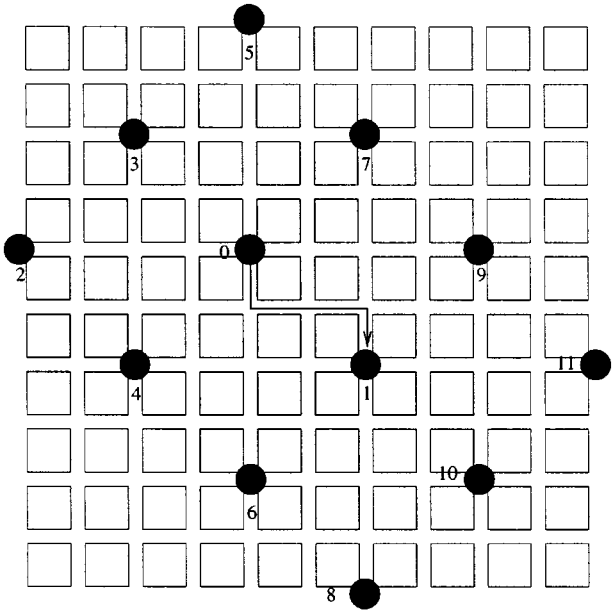


Fig. 9. Base station arrangement for side-street scenario.

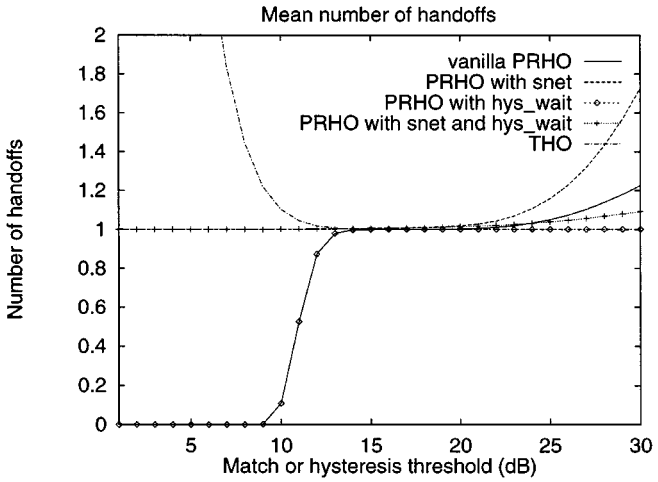


Fig. 10. Mean number of handoffs in the side-street problem.

will most probably need to occur is along the side street. The propagation model used in the simulations takes into account the fact that the radiowaves need to go around corners in situations like this, and computes the propagation accordingly [20].

The mean number of handoffs is examined for different handoff algorithms in Fig. 10, and the average delay is examined in Fig. 11. The behavior of vanilla PRHO and its variations is similar to that in the case of the canonical handoff scenario. Once again, there is practically no difference between the variations in the optimal region of match threshold between 13 and 19 dB where they all perform very well, with practically no unnecessary handoffs and negligible delay. The relative robustness of the algorithms to match thresholds is similar to that in the case of the canonical handoff scenario. As in that scenario, *PRHO with snet and hys_wait* appears to be the best choice.

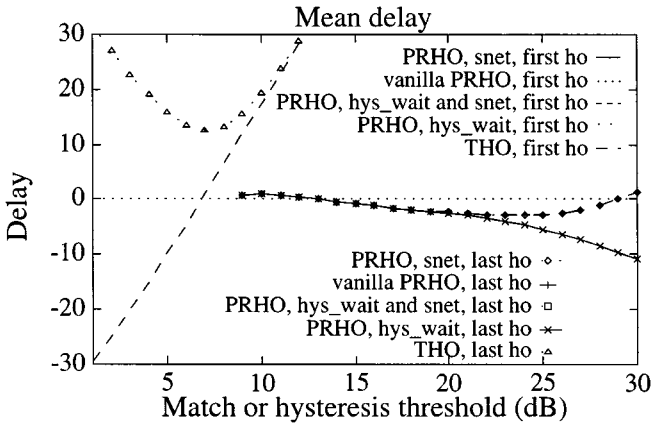


Fig. 11. Delay in the side-street problem.

For a match threshold of 15 dB, the mean first handoff delay is −0.8113, with standard deviation of 3.379 16, and the mean last handoff delay is −0.6708, with standard deviation of 4.6843. This combined algorithm again has practically no unnecessary handoffs and negligible delay.

V. Further Discussion

A. Handoff Control Structure

Handoffs may be either *intercell* if the old and new channels are at different base stations, or *intracell* if the old and new channels are at the same base station. Handoffs may also be either *soft* if there is simultaneous traffic channel communication during the handoff process, or *hard* if there is not. The control structure for handoffs in a system may be one of three kinds: with *network-controlled* handoffs (NCHO), measurements are made and control is performed at in the network of base stations; with *mobile-assisted* handoffs (MAHO), measurements are made by the users and transmitted to the serving base station, which performs handoff control and may also perform measurements; with *mobile-controlled* handoffs MCHO, measurements are generally performed at the user only, and control is performed at the user. NCHO and MAHO are *network-initiated*, while MCHO is *mobile-initiated*. In the case of intercell handoffs, when the handoff is mobile-initiated, the initiation may occur through signaling with the old base station or the new base station. Handoffs of the former type are *backward* handoffs, and those of the latter type are *forward* handoffs.

Our algorithms are applicable for both intracell and intercell handoffs. As described in this paper, the algorithms are applicable for hard handoffs. However, the framework is easily adaptable to working with soft handoffs as well. To reduce mobile terminal complexity, the pattern recognition is performed at base stations, and so mobile-controlled pattern recognition handoffs are excluded (but users can still initiate handoffs based on other criteria). The configuration discussed in this paper makes mobile assisted handoffs, with users making measurements and transmitting them to base stations. However, an alternative configuration is also possible, where surrounding base stations make measurements, and transmit them to the serving base station. Both forward and backward handoffs are

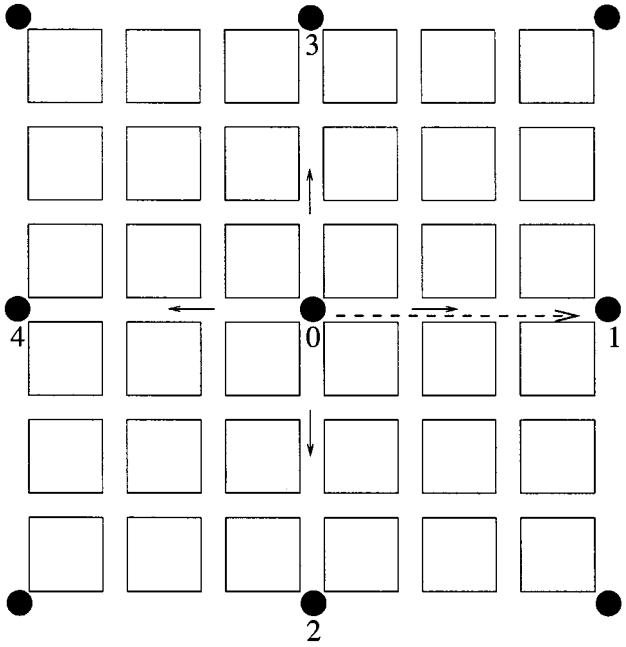


Fig. 12. Geometrical arrangement for computer simulations. The squares are city blocks and the circles are base stations. The solid arrows are the pattern stretches of the $\Omega = 4$ pattern classes. The dashed arrow between BS 0 and BS 1 indicates the path of motion of the user. $\mathcal{M} = 5$. When the serving base station is BS0, the included base station set is $\{0, 1, 2, 3, 4\}$.

possible. Nevertheless, if it is known in some situations that a forward handoff is definitely better than a backward handoff (or vice versa), that information can be included in the pattern action of the associated pattern class.

B. On the Selection of Match Thresholds

Unnecessary handoffs occur when the match thresholds are too *high*. Handoffs would then be triggered too often, sometimes because of false alarms. However, a genuine pattern match can be missed when the thresholds are too *low*. Handoffs would be rarely triggered, and sometimes not triggered when they should be. Therefore, careful setting of the match thresholds in PRHO is important. In this section, we describe a way to estimate the performance of the pattern recognition system without the need to simulate it. The method is illustrated for the scenario shown in Fig. 12. The user moves from BS 0 to BS 1, and is served by BS 0, which has four patterns in its pattern set, as shown in the figure. We estimate the mean-number-of-handoff performance, and compare the estimates with simulation results.

Note that in millions of simulation runs, the number of internal mismatches (between pattern classes associated with the same base station) is 0, for all reasonable parameters settings investigated. The reason is that since there are several base station components in each pattern, at least one, and probably more, of these base station components differ significantly (local mean signal power levels being mostly on the order of tens of dB different), so the patterns are far apart enough in pattern space that probability of internal mismatches is basically zero.

Simulation results are plotted against the match threshold in Fig. 13 for $\mathcal{M} = 5$ and $\Omega = 4$ as in Fig. 12; each of the 4 pattern classes contains patterns of size $N_p = 50$; $L = 3$ training patterns are obtained for each class, and the preprocessing averaging length is 6. There is also another curve (labeled *predicted values*) which is based on predicted values, where the prediction method is explained below. The closeness of the curves attests to the accuracy of the prediction method. Both curves show no pattern matches occuring for match thresholds which are too small. In the region approximately between match thresholds of 8 and 13 dB, pattern matches occur with a certain probability, increasing with the match threshold. For larger match thresholds, it is almost certain that pattern matches will occur. In all these cases, the pattern match is to the correct pattern class, i.e., the one pointing between base stations 0 and 1, in Fig. 12.

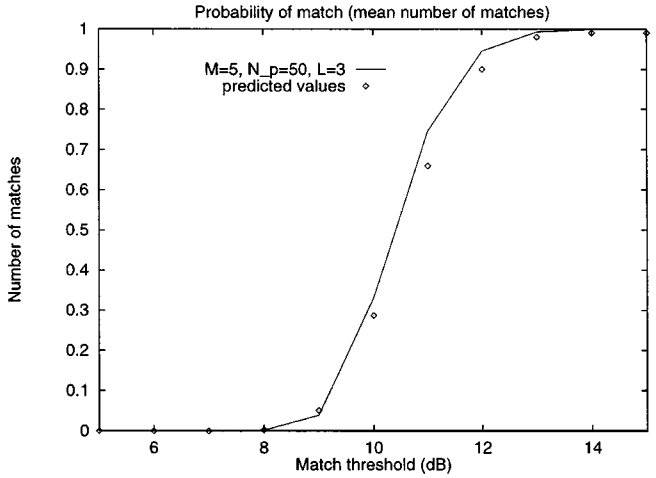


Fig. 13. Comparison of simulation results with predicted results.

Reasonably accurate estimates can be made of the locations (in terms of match threshold ranges) of the characteristic rise (the shape of the curve where it goes from 0 to 1), and of the slope. Estimates of pattern matching performance are based on mean and variance statistics at the match points. Mean and variance estimates are used, and a Gaussian distribution is assumed, to calculate the probabilities of being below the match threshold. The close correlation between the estimates and the sliding window pattern recognition simulation results justifies the assumption for this purpose.

Let α and $\sqrt{\alpha_v}$ be the mean and standard deviation of

$$X = \frac{1}{N_p \mathcal{M}} \min_{j=1,2,\ldots,L} \sum_{i=1}^{N_p \mathcal{M}} (Y_{j,1,i} - Y_{j,2,i})^2$$

where for each i, the set of random variables $Y_{j,1,i}$ and $Y_{j,2,i}$ is IID with distribution equivalent to that of the sample average of the appropriate number of signal strength samples. By Monte-Carlo methods, we obtain $\alpha \approx 8.8$ and $\sqrt{\alpha_v} \approx 1.56$. From this, it can be predicted that the characteristic rise would be around 8.8. Suppose that we also want to know $P(X < k)$ for $k = 5, 6, \ldots, 14, 15$, where X is a random variable with the distribution of M_1. Assuming X has a roughly Gaussian distribution with mean α and standard deviation $\sqrt{\alpha_v}$, $P(X < k)$ can be computed as

$$P(X < k) = \frac{1}{2}\left[1 + \operatorname{erf}\left(\frac{k - \alpha}{\sqrt{2\alpha_v}}\right)\right] \tag{4}$$

where $\operatorname{erf}(j)$ is the error function $(2/\sqrt{\pi}) \int_0^j e^{-\lambda^2}\, d\lambda$. Then $P(X < k)^5$ is the estimate, and is what is plotted for the *predicted values* curve.

Most of the time, the estimates are slightly *smaller* than the actual probability that pattern matches occur, for any given match threshold. This is because sliding window pattern matching is used, so there is a chance of matching happening slightly early or late as well, slightly increasing the actually probability. However, the estimates are close enough that they can be used by system operators to set parameters like match thresholds, if sliding window pattern recognition techniques were to be used.

C. Limitations of PRHO

PRHO has been found to perform better than THO. Two of the main potential limitations of PRHO are the processing requirements on the base stations and the need for training.

For each user that it serves, a base station would be continually attempting to find a match with all Ω of its pattern classes. This processing increases linearly with the number of users and with Ω. The core computation is the evaluation of (3), which can be performed very quickly, especially if parallel computations are used. It should be noted that our algorithm is less complex than that in [7], which associates a different pattern class with every short distance (e.g., 1 meter) of movement, requiring more signal processing for pattern classification as well as much greater storage space. Furthermore, our algorithm has the flexibility that not all users need be using PRHO—perhaps it could be reserved for users paying for premium service. Moreover, the number of pattern classes need not be large, if pattern classes are defined only for "difficult" spots that are known to give trouble to THO. In this case, PRHO with a hysteresis-based failsafe mechanism (as in Section IV-C) might be utilized.

The most accurate use of PRHO requires training of the pattern classes. Training need not inconvenience users if it is performed at offpeak hours. Furthermore, training is only required along the stretches of street associated with pattern classes, not along entire streets all around the base station, as in some other schemes [7]. In addition, once the training has been done, the pattern templates can be used for a long time—weeks or months—if there are no significant changes in the propagation environment (if the buildings along the street are not torn down, etc.). While there would be a cost associated with sending people out to make training measurements, this might be preferable to building new base stations for more coverage, as a solution for fixing "difficult" spots (where THO normally results in many dropped calls). An alternative approach to obtaining the pattern templates has also been discussed in Section III-D-2.

Finally, there is the issue of whether the required measurements can be adequately made by a user as it moves around. In particular, existing systems may not provide for measurements of signal strength of surrounding base stations as often and consistently as PRHO requires. However, future systems could be designed to provide for this.

VI. Conclusion

Traditional Handoff Algorithms (THO) trade off accuracy and timeliness (number of unnecessary handoffs versus delay and/or dropped calls). New handoff algorithms have been developed (called PRHO). It has been demonstrated that these handoff algorithms can avoid the accuracy/timeliness tradeoff and perform better than THO in a variety of handoff scenarios.

References

[1] S. T. S. Chia, "The control of handover initiation in microcells," in *Proc. IEEE Veh. Technol. Conf.*, St. Louis, May 1991, pp. 531–536.

[2] Motorola Inc. Final text for PACS licensed air interface (TAG 3) J-STD 014, June 1995.

[3] R. Vijayan and J. M. Holtzman, "A model for analyzing handoff algorithms," *IEEE Trans. Veh. Technol.*, vol. 43, pp. 351–356, Aug. 1993.

[4] O. Kennemann, "Pattern recognition by hidden markov models for supporting handover decisions in the GSM system," in *Proc. 6th Nordic Seminar Digital Mobile Radio Commun.*, June 1994, pp. 195–202.

[5] H. Maturino-Lozoya, D. Munoz-Rodríguez, and H. Twafik, "Pattern recognition techniques in handoff and service area determination," in *Proc. IEEE Veh. Technol. Conf.*, Stockholm, June 1994, pp. 96–100.

[6] N. Tripathi, J. Reed, and H. VanLandingham, "Pattern classification based handoff using fuzzy logic and neural sets," in *Proc. IEEE Int. Conf. Commun.*, Atlanta, June 1998, pp. 1733–1737.

[7] R. Narasimhan and D. Cox, "A handoff algorithm for wireless systems using pattern recognition," in *Proc. IEEE Int. Symp. Personal, Indoor, Mobile Radio Commun.*, Boston, Sept. 1998.

[8] J. M. Holtzman and A. Sampath, "Adaptive averaging methodology for handoffs in cellular systems," *IEEE Trans. Veh. Technol.*, pp. 59–66, Feb. 1995.

[9] O. E. Kelly and V. V. Veeravalli, "A locally optimal handoff algorithm," in *Proc. IEEE Int. Symp. Personal, Indoor, Mobile Radio Commun.*, Toronto, Sept. 1995, pp. 809–813.

[10] V. Kapoor, G. Edwards, and R. Sankar, "Handoff criteria for personal communication networks," in *Proc. IEEE Int. Conf. Commun.*, New Orleans, May 1994, pp. 1297–1301.

[11] M. D. Austin and G. L. Stüber, "Cochannel interference modeling for signal strength based handoff analysis," *Electron. Lett.*, vol. 30, no. 23, pp. 1914–1915, Nov. 1994.

[12] W. C. Jakes, Ed., *Microwave Mobile Communications*. New York: Wiley, 1974. Republished by IEEE Press.

[13] R. Rezaiifar, A. M. Makowski, and S. Kumar, "Optimal control of handoffs in wireless networks," in *Proc. IEEE Veh. Technol. Conf.*, Chicago, July 1995, pp. 887–891.

[14] B. Senadji, S. Tabbane, and B. Boashash, "A handover decision procedure based on the minimization of bayes criterion," in *Proc. IEEE Veh. Technol. Conf.*, Stockholm, June 1994, pp. 77–81.

[15] D. Wong, "Handoff algorithms using pattern recognition," Ph.D. thesis, Stanford Univ., June 1998.

[16] M. D. Austin and G. L. Stüber, "Velocity adaptive handoff algorithms for microcellular systems," *IEEE Trans. Veh. Technol.*, vol. 43, pp. 549–561, Aug. 1994.

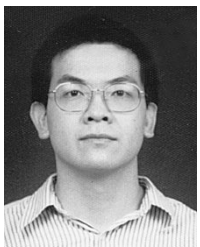
[17] D. Wong and D. Cox, "Estimating local mean signal power level in a Rayleigh fading environment," *IEEE Trans. Veh. Technol.*, vol. 48, pp. 956–959, May 1999.

[18] M. Nadler and E. Smith, *Pattern Recognition Engineering*. New York: Wiley, 1993.

[19] T. M. Cover and P. E. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inform. Theory*, vol. IT-13, pp. 21–27, Jan. 1967.

[20] J.-E. Berg, R. Bownds, and F. Lotse, "Path loss and fading models for microcells at 900 MHz," in *Proc. IEEE Veh. Technol. Conf.*, Denver, May 1992, pp. 666–671.

[21] A. Sampath and J. M. Holtzman, "Estimation of maximum doppler frequency for handoff decisions," in *Proc. IEEE Veh. Technol. Conf.*, Seacaucus, May 1993, pp. 859–861.



K. Daniel Wong (M'98) was born in Ipoh, Malaysia in 1971. He received the B.S.E. degree in electrical engineering (with highest honors) from Princeton University, New Jersey, in 1992, and the M.S. and Ph.D. degrees in electrical engineering (both from Stanford University, California) in 1994 and 1998, respectively. He joined Bellcore (now Telcordia Technologies) in 1998 as a Research Scientist and is currently working on wireless access to next generation networks. His research interests include handoff algorithms in cellular systems, wireless broadband technologies and network protocols, multi-carrier systems, MAC and routing protocols for wireless IP networks, and 3rd generation mobile systems. He is also a member of Tau Beta Pi, Sigma Xi, and Phi Beta Kappa.

Donald C. Cox (S'58–M'61–SM'72–F'79) received the B.S., and M.S. degrees in electrical engineering from the University of Nebraska in 1983. From 1960 to 1963, he did communications system design at Wright-Patterson AFB, Ohio.

From 1963 to 1968, he was at Stanford University doing tunnel diode amplifier design and research on microwave propagation in the troposphere.

From 1968 to 1973, his research at Bell Laboratories, Holmdel, NJ in mobile radio propagation and on high-capacity mobile radio systems provided important input to early cellular mobile radio system development, and is continuing to contribute to the evolution of digital cellular radio, personal communications systems and cordless telephones. From 1973 to 1983, he was Supervisor of a group at Bell Laboratories that did innovative propagation and system research for millimeter-wave satellite communications. In 1978, he pioneered radio system and propagation research for low-power wireless personal communications systems. At Bell Laboratories in 1983, he organized and became the Head of the Radio and Satellite Systems Research Department that became a Division in Bell Communications Research (Bellcore) with the breakup of the Bell System in 1984. He was Division Manager of that Radio Research Division until it again became a department in 1991. He continued as Executive Director of the Radio Research Department where he championed, led, and contributed to research on all aspects of low-power wireless personal communications, entitled, Universal Digital Portable Communications (UDPC). He was instrumental in evolving the extensive research results into specifications that became the U.S. Standard for the Wireless or Personal Access Communications System (WACS or PACS). In September 1993, he became a Professor of Electrical Engineering and Director of the Center for Telecommunications at Stanford University where he continues to pursue research and teaching of wireless mobile and personal communications. He was appointed Harald Tap Friis Professor of Engineering in 1994.

Dr. Cox is a member of the National Academy of Engineering, and is a Fellow of the IEEE, the AAS, and the Radio Club of America. He was awarded the IEEE 1993 Alexander Graham Bell Medal "For pioneering and leadership in personal portable communications;" was a co-recipient of the 1983 International Marconi Prize in Electromagnetic Wave Propagation (Italy); and received the Bellcore Fellow award in 1991, the IEEE 1985 Morris E. Leeds Award, the IEEE Communications Society 1992 L. G. Abraham Prize Paper Award and 1990 Communications Magazine Prize Paper Award, and the 1983 IEEE Vehicular Technology Society paper of the year award. He is a member of Commissions B, C and F of USNC/URSI, was an Associate Editor of the IEEE TRANSACTIONS ON ANTENNAS AND PROPAGATION (1983–1986), was a member of the Administrative Committee of the IEEE Antennas and Propagation Society (1986–1988), and was a member of the URSI Intercommission Group on Time Domain Waveform Measurements (1982–1984). He is author or coauthor of many papers and conference presentations, including many invited and several keynote addresses, and books. He has been granted over a dozen patents. He is a member of Sigma Xi, Sigma Tau, Eta Kappa Nu and Phi Mu Epsilon, and is a Registered Professional Engineer in Ohio and Nebraska.