Search Theory and Sequential Paging: Intelligent Mobile Location in Wireless Networks

Bhaskar Krishnamachari

"A" Exam Talk

School of Electrical and Computer Engineering
Cornell University

December 4, 2000

Prologue

Optimization Algorithms for Wireless Communications

Summary of Research with Prof. Wicker (Fall 1998 - Present)

- Survey of randomized search algorithms for wireless communications (in *Telecommunications Optimization: Heuristic and Adaptive Techniques* , John Wiley, 2000)

- Randomized search algorithms for design problems in cellular networks (*M.S. Thesis 1999*):

  - base station location design (*ECCAP 2000*)

  - minimum cost fixed network topology (*VTC-Fall 2000*)

- Theoretical analysis of randomized search algorithms (with Xi Xie, Prof. Selman, *CP 2000*)

Optimization Algorithms for Wireless Communications

Summary of Research with Prof. Wicker (Fall 1998 - Present)

- Improving turbo decoding via cross-entropy minimization (with Eoin Buckley, Prof. Hagenauer, *ISIT 2000*)
- Joint location area and paging zone design using randomized search algorithms (*manuscript in preparation*)
- Optimal sequential paging in cellular networks (with Michael Gau, Prof. Haas, submitted to the *IEEE/ACM Transactions on Networking, 2000*)
- Sequential paging on graphs (*ongoing work*)

Outline of Talk

- Search Theory
- Intelligent Mobile Location
- Sequential Paging for Cellular Networks
- Extension to Multi-hop Wireless Networks

Search Theory

Overview of Search Theory

- Concerned with the problem of finding a hidden target[1]
- Dates back to U.S. Navy Anti-submarine Warfare Operations Research during World War II
- Some of the techniques were motivated by the search for an H-bomb lost (!) in the Mediterranean near Palomares Spain in 1966, and the search for the missing U.S. nuclear submarine Scorpion in 1968
- One-sided searches and search games
- Continuous vs. discrete locations

[1] *Theory of Optimal Search*, L.D. Stone, 1975.

Search with Discrete Efforts

- Target is in one of n cells. Only one cell queried at each step.
- π_j = Probability that target is in cell j
- μ_j = Probability of a "miss" - i.e. the event that the target is in cell j, but a query in cell j doesn't succeed
- $\beta(j,k) = (1-\mu_j)(\mu_j^{(k-1)})$, Probability of first success on k^{th} query to cell j given that target is in cell j.
- A discrete search plan is a $\{1,\ldots,n\}$ valued sequence $x = \{x_i, i = 1, 2, \ldots\}$, x_i indicating which cell to query at on the i^{th} step.

Optimal Search Plans

- Let $r(j,k,x^*)$ be the number of queries out of the first k under plan x^* that are made to cell j.

- A theorem due to Blackwell & Stone shows that the following *greedy* plan maximizes the probability of locating the target on or before the m^{th} query (for any $m \geq 1$):

$$x^*_{k+1} = argmax(\beta(j, r(j,k,x^*) + 1) \cdot \pi_j)$$

- Informally: **query the cell j which maximizes the detection probability**

Optimal Search Plans (contd.)

- Another theorem by Stone shows that the uniformly optimal plan also minimizes the expected number of queries. If there is a cost associated with querying each cell, then the plan which minimizes the expected cost is to **query the cell that maximizes the ratio of detection probability to cost**. This plan will incur a finite expected cost.

- If we assume that the probability of a miss on a query, $\mu_j = 0$, then the optimal plan is to query the cells in decreasing order of location probabilities.

Intelligent Mobile Location

Locating Mobile Users in a Cellular Network

Users move from cell to cell in a non-deterministic manner, and when a call arrives for a given user, the network needs to determine its location in order to set up the call. A three-step process is involved:

- the mobile provides *location updates*,
- the network sends *paging messages* at the time of call arrival, and
- the mobile sends an *acknowledgement* of its location in response to paging.

Operating Costs

- Radio bandwidth utilization

- Intra-network signaling

- Database maintenance

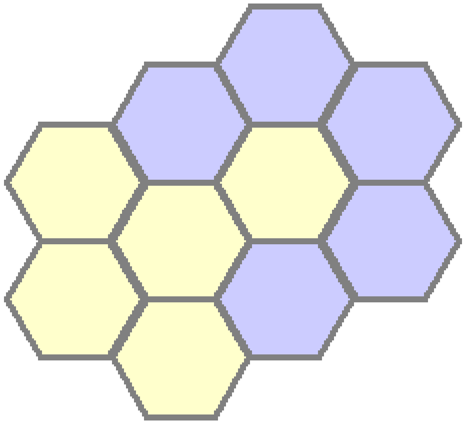
- Mobile energy consumption

Two Extremes

- Mobiles provide location updates every time they enter a new cell (no need for paging)

- Every cell in the entire network is paged each time a new call arrives (no need for location updates)

Current Systems



- Divide the network into location areas (LA), each consisting of multiple cells

- Mobile provides a location update each time it enters a new location area

- Network pages all cells in the current location area

Location Update Schemes

These can be loosely classified as static versus dynamic:

static

- location areas (individualized)
- designated reporting centers

dynamic

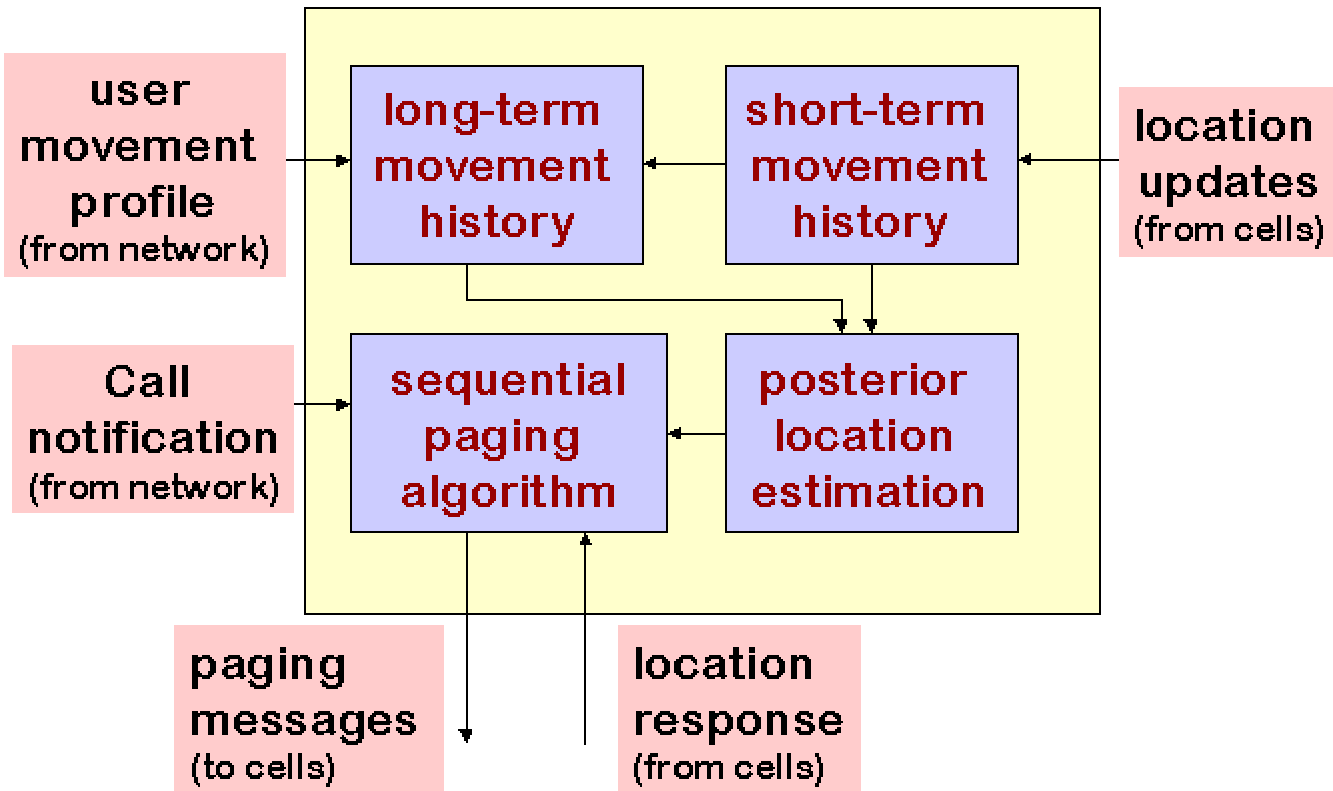
- movement-based
- time-based
- distance-based
- predictive distance-based

Intelligent Communication Systems

- Traditional Systems: Make assumptions about operating environment such as aggregate user movement tendencies, energy consumption, channel conditions etc. in planning the system.

- Intelligent Systems: Identify and monitor key decision variables and incorporate the decision-making process into the operational system, making it adaptive.

An Architecture for Intelligent Mobile Location



Sequential Paging for Cellular Networks

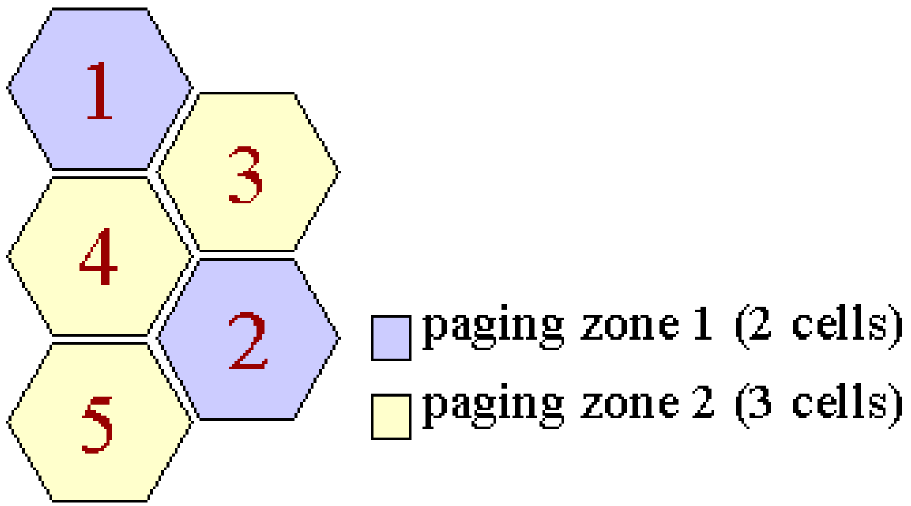
Sequential Paging

- The idea here is to partition location area into paging zones based on estimated user location probabilities and page these zones one by one. This trades off potentially increased delay for a reduction in the average cost of paging.

- If delay is not an issue, then the scheme that minimizes expected paging cost is to page all the cells in the location area one by one in decreasing order of location probability.

- But if we have a worst case delay constraint of w, then partition the location area into w *paging zones*, page all the cells in zone 1 simultaneously first, if user not found, then page all users in zone 2, and so on.

An Example



Cell	1	2	3	4	5
prob.	0.3	0.3	0.25	0.1	0.05

An Example (contd.)

- n = 5 cells in location area, C = {1, 2, 3, 4, 5}
- cell-wise user location probabilities:

i	1	2	3	4	5
π_i	0.3	0.3	0.25	0.1	0.05

- Number of paging zones w = 2
- Paging zones: $Z_1 = \{1, 2\}$, $Z_2 = \{3, 4, 5\}$
- Number of cells in each paging zone: $n_1 = |Z_1| = 2$, $n_2 = |Z_2| = 3$
- Zone location probabilities: $p_1 = 0.3 + 0.3 = 0.6$, $p_2 = 0.25 + 0.1 + 0.05 = 0.4$
- Average paging delay: $\overline{D} = 1 * p_1 + 2 * p_2 = 1 * 0.6 + 2 * (0.4) = 1.4$
- Average paging cost: $\overline{L} = n_1 * p_1 + (n_1 + n_2) * p_2 = 2 * (0.6) + 5 * (0.4) = 3.2$

What is the optimal partition?

Performance Measures For Sequential Paging

- Worst-case paging delay : w rounds / call arrival
- Average delay : $\overline{D} = \sum_{i=1}^{w} i * p_i$ rounds / call arrival
- Average paging cost : $\overline{L} = \sum_{i=1}^{w} (\sum_{j=1}^{i} n_j) * p_i$ cells paged / call arrival

Some Basic Results

- A partition of a location area into paging zones that minimizes either the average delay or the paging cost is such that a less probable cell is never paged before a more probable one.
- The minimum cost of paging obtained using w paging zones is less than or equal to that obtained using fewer than w paging zones.

Minimizing Average Paging Cost under Worst-case Delay Constraint

min $\overline{L}$, subject to: w is a fixed natural number

- Definition: $h[k, e]$ is the minimum average paging cost that can be achieved by dividing only the first e cells (numbered in decreasing order of location probability) into k paging zones. The solution we want is $h[w, n]$.

- The following recursive relation holds:

$$h[k+1, e] = \min_{j} h[k, j] + e \cdot \sum_{j+1}^{e} \pi_i$$

- Initial conditions: $h[1, j] = j \cdot \sum_{i=1}^{j} \pi_i$ for all $j \leq n$.

Minimizing average paging cost

under average delay constraint

min $\overline{L}$, subject to: $\overline{D} \leq \alpha$, α a positive real number, and w fixed

- "This problem is not amenable to solution via dynamic programming owing to the constraint on E[D]." Rose & Yates, 1995
- "(This problem) is NP-complete." - Abutaleb and Li, 1997

The Knapsack Problem

There are n paintings in a museum, each of positive integer value v_i and weight w_i, and the burglar can carry away no more than a total weight of W. What algorithm should the burglar employ to maximize the total value of the paintings he steals from this museum?

The optimal burglar knows dynamic programming

- Let $OPT(i, w)$ denote the maximum total value that can be stolen using a subset of the first i paintings $\{1, \dots, i\}$ with maximum allowed weight w. We want the quantity $OPT(n, W)$.

- Consider the k^{th} gold bar:

  a) If it is not taken by the burglar, then $OPT(k, W) = OPT(k-1, W)$.

  b) If it is taken by the burglar, then $OPT(k, W) = OPT(n-1, W-w_k) + v_k$.

  The burglar should choose the better of these two options. Hence the following recursion holds:

  $$OPT(k, W) = max(OPT(k-1, W), v_k + OPT(n-1, W-w_k))$$

- Now fill in a table of size W by n, using the initial condition: $OPT(1, w) = v_1$, if $w_1 \leq w$, else 0.

Minimizing Average Paging Cost under Average Delay Constraint

min $\overline{L}$, subject to: $\overline{D} \leq \alpha$, α a positive real number, and w fixed

- Definition: $h^{\dagger}[k, e, \beta]$ is the minimum average paging cost that can be achieved by dividing the first e cells into k paging zones, with a maximum average paging delay of $\beta \leq e$. We seek $h^{\dagger}[w, n, \alpha]$.

- Now:

$$h^{\dagger}[k+1, e, \beta] \;=\; \min_{j} h^{\dagger}[k, j, \beta - (k+1) * \sum_{j+1}^{e} \pi_i] + e * \sum_{j+1}^{e} \pi_i$$

- Initial Condition: If $\sum_{1}^{j} \pi_i \leq \beta$, then $h^{\dagger}[1, j, \beta] = j * \sum_{1}^{j} \pi_i$, else ∞.

- Runs in $O(wn^2 A)$ time, where A is a constant representing the total number of discrete values the average delay can take on.

Other Problem Formulations

We have also developed polynomial-time sequential paging algorithms that are optimal in terms of:

- minimizing the weighted mean of average paging cost and average delay (with and without constraints on worst case delay)
- minimizing penalties due to paging cost, average delay, and worst case delay
- minimizing worst case delay with constraint on average paging cost
- minimizing the average paging delay under constraint on average paging cost

Performance of Sequential Paging

- We need a measure of the performance gain obtained using sequential paging. Let the normalized reduction in average paging cost be defined as follows for a given location probability distribution $\Pi_n = \{\pi_1, \pi_2 \ldots, \pi_n\}$:

$$\Lambda_{\Pi_n}(w) = \frac{\overline{L}_1^{\Pi_n} - \overline{L}_w^{\Pi_n}}{\overline{L}_1^{\Pi_n}}$$

- $\Lambda_{\Pi_n}(w) = 0$ when $w = 1$ (all cells paged simultaneously). We would like it to be as close to 1 as possible, since this represents the greatest gain using sequential paging.

Performance of Sequential Paging (contd.)

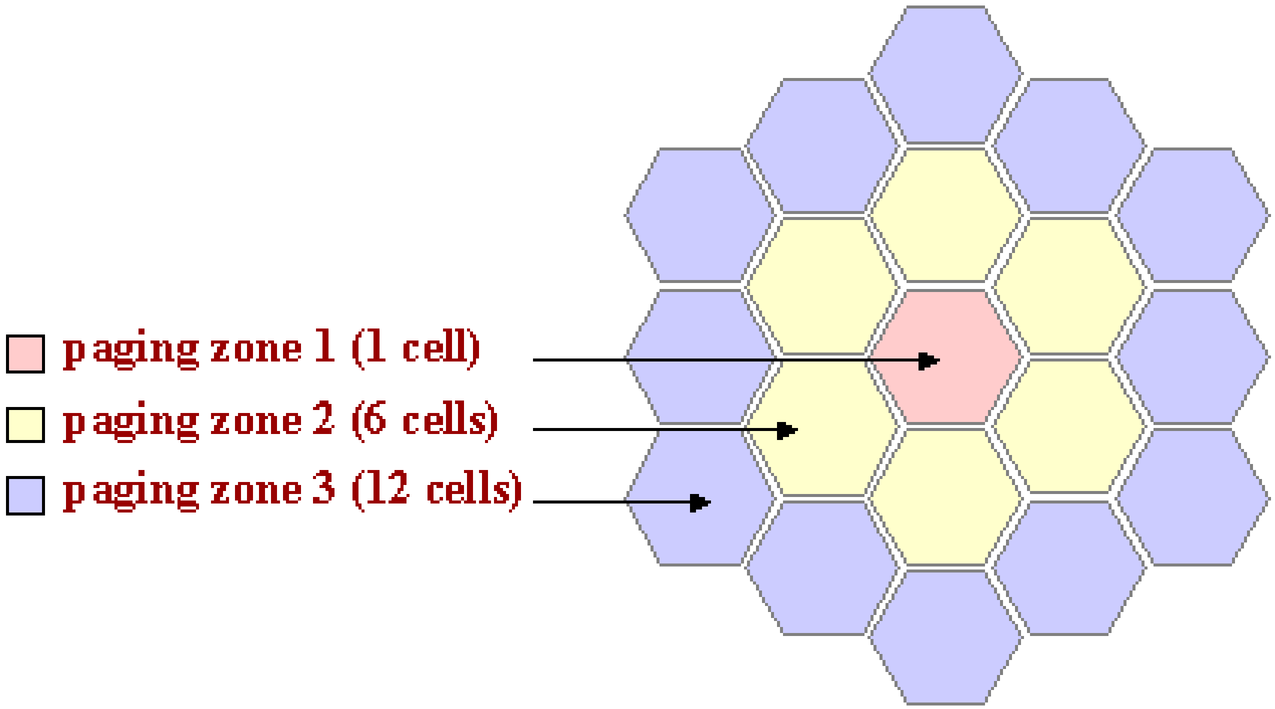
Theorem 1: Of all the possible location probability distributions, the minimum average paging cost and the average paging delay (using a fixed number of paging zones) are the worst when the user is equally likely to be in any cell, i.e. for a uniform distribution U_n.

Therefore $\Lambda_{\Pi_n}(w) \geq \Lambda_{U_n}(w)$ and $\overline{D}_w^{\Pi_n} \leq \overline{D}_w^{U_n}$.

Theorem 2: The following are true:

- $\lim_{n\to\infty} \overline{D}_w^{U_n} = \frac{w+1}{2}$
- $\lim_{n\to\infty} \Lambda_{U_n}(w) = \frac{1}{2}(1 - \frac{1}{w})$

Cluster/Selective Paging

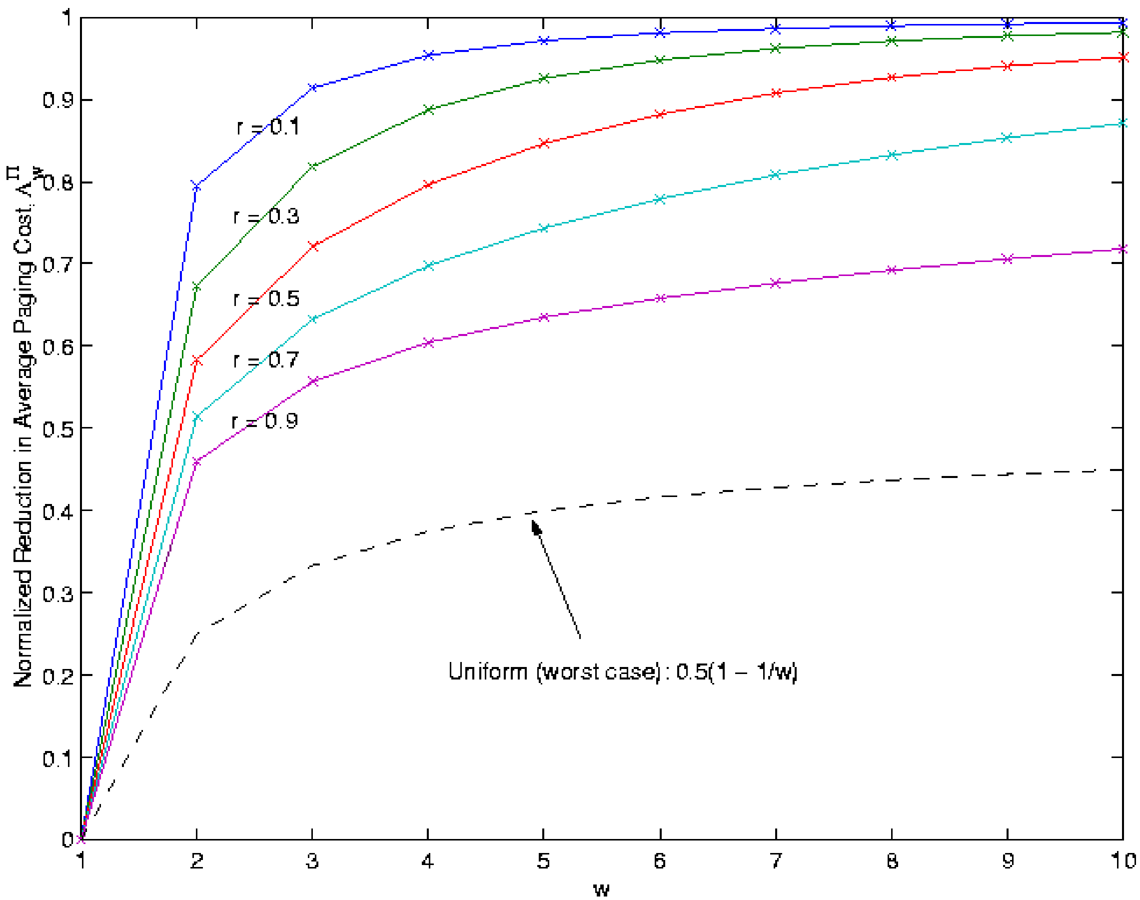


Geometrically Distributed Zone Location Probabilities

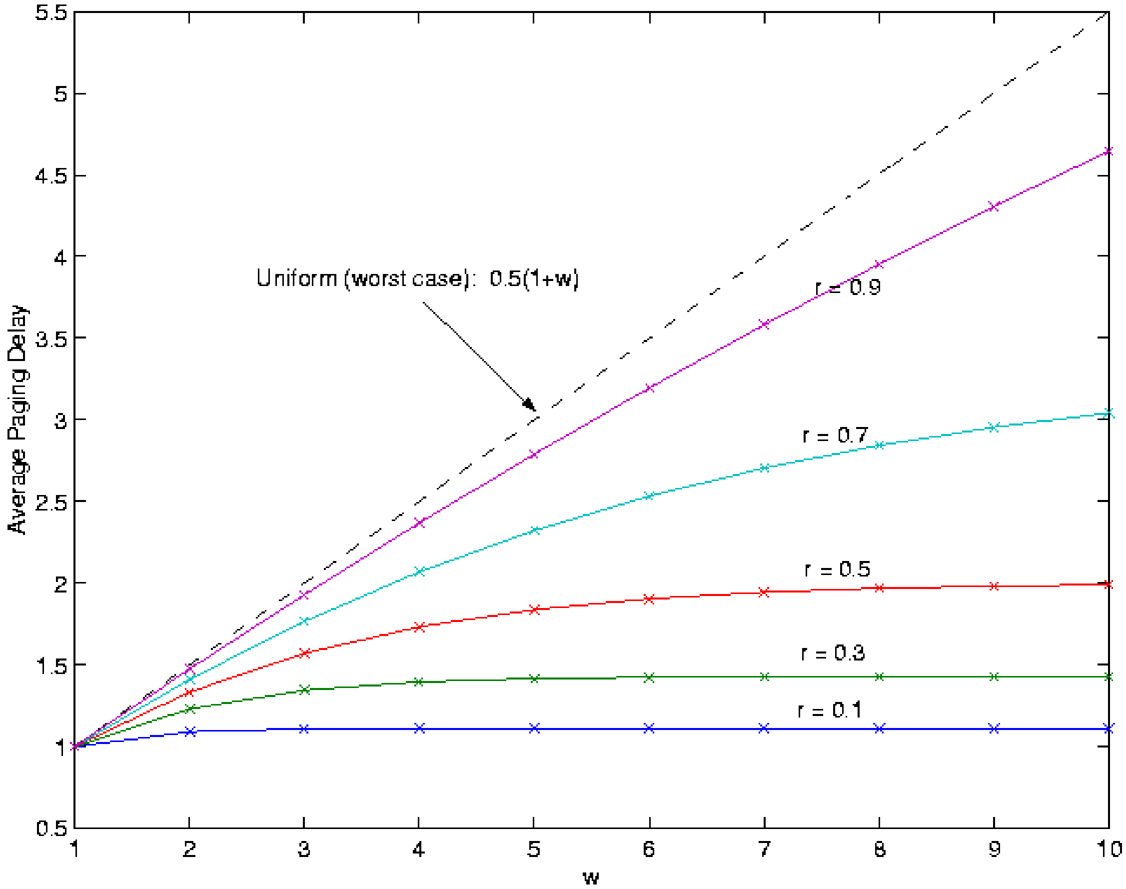
- Theorem 3: If the user location probability decreases as we move outwards from the center cell, and each cell in a given ring has the same user location probability, then cluster/selective paging is the optimal sequential paging scheme.

- If each successive ring has a geometrically decreasing probability of user location, then expressions for average paging delay and average paging cost are easy to derive analytically.

Normalized Reduction in the Average Cost of Paging



Average Paging Delay



Sequential Paging for Cellular Networks: Future Work

- Characterize the sensitivity of these schemes to errors in the location probability estimates.

- Look into ensemble sequential paging, i.e. paging multiple mobile users at once.

- Develop Bayesian techniques for obtaining accurate location estimates.

- Develop software mobility models in order to simulate the performance of these algorithms under realistic settings.

Extension to Distributed Multi-hop Wireless Networks

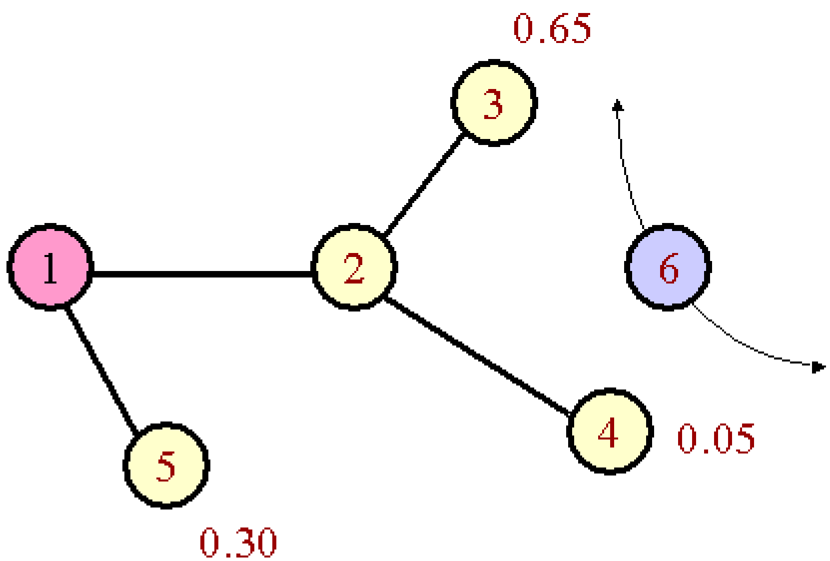
Distributed Multi-hop Wireless Networks

- Evolved from DARPA packet radio program in the early 1970s; often referred to as ad hoc networks. They do not require much infrastructure, and can be deployed rapidly.

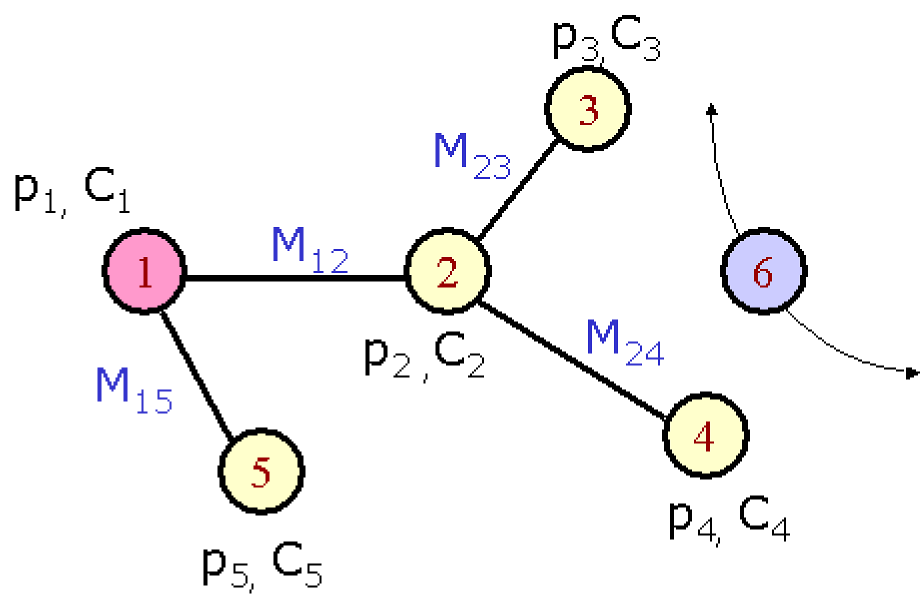
- Applications: disaster recovery, home/office/campus local area networks, law enforcement, military communications, sensor networks, special events such as conferences, festivals etc.

- Here we assume that there is a backbone of *quasi-stationary* nodes, and that all other *mobile* nodes can be reached in at most one hop from one of the backbone nodes (e.g. MCDS routing, Das and Bhargavan 1997).

Locating a Mobile Node

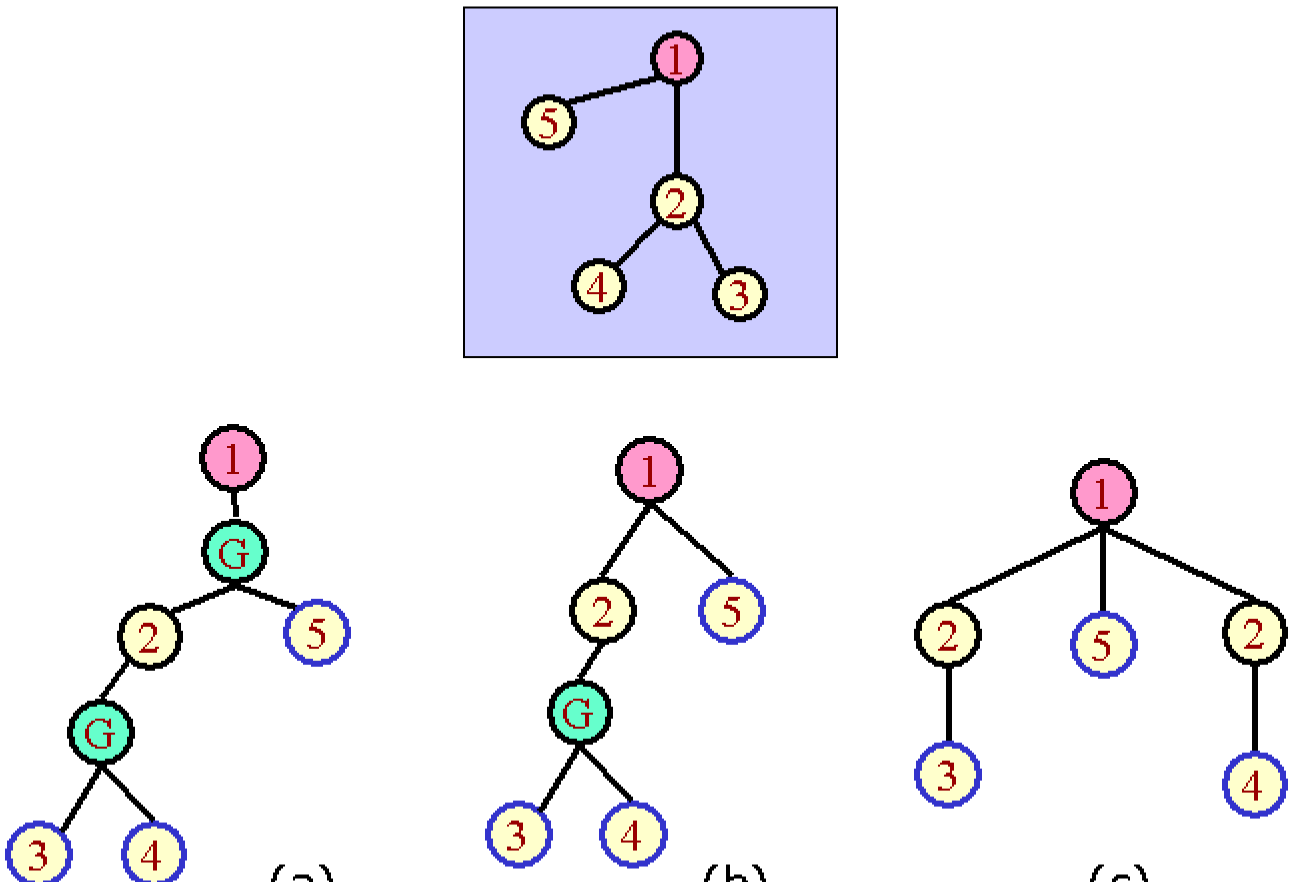


Sequential Paging on Graphs



- Graph-based formulation: Given a graph G = (V,E) without cycles (i.e. a tree), a source node $s \in V$, a target location probability distribution p_v over the node set V, message costs $M_e, \forall e \in E$, and paging costs $C_v, \forall v \in V$. Determine a search procedure to locate the target while minimizing the expected total cost (or some other cost function).

Paging Trees

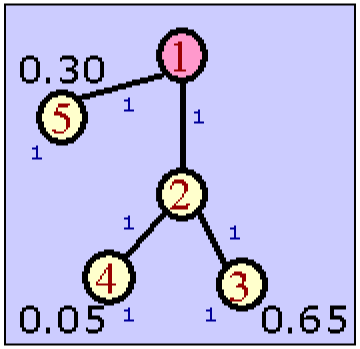


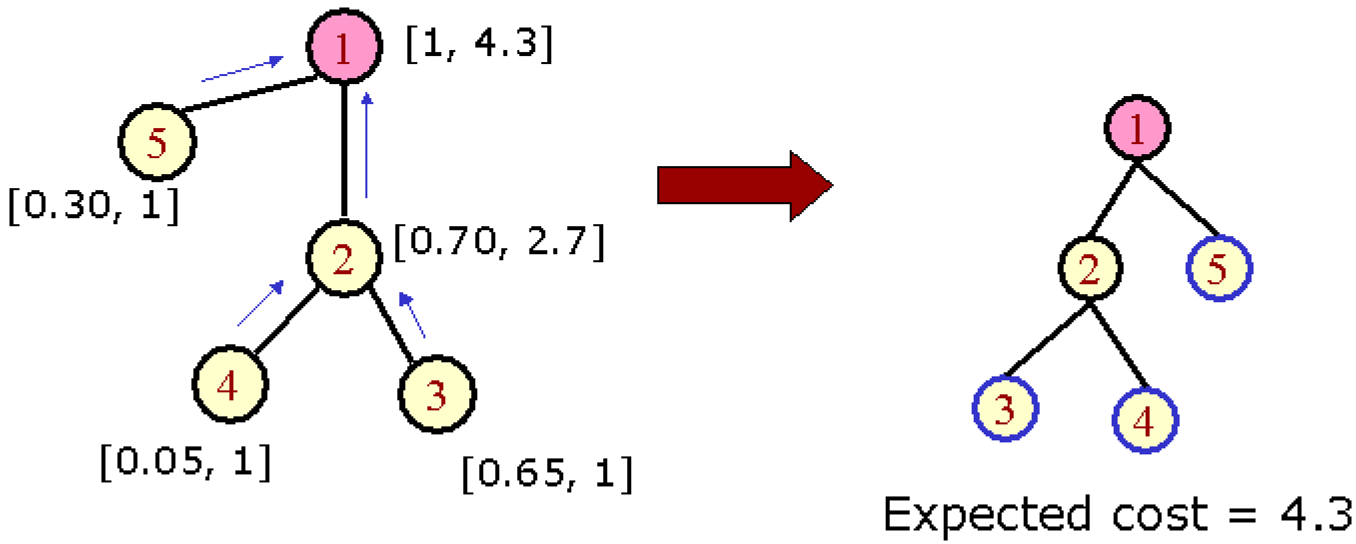
(a) (b) (c)

Distributed Sequential Query Forward Algorithm (DSQFA)

- Goal: minimize expected cost of paging and messaging, without delay constraints.
- Each node sends a query to each of its children (neighbors) in sequence, waiting for a response before querying the next child node. If mobile is located by a child, or if all children have been queried, it reports back to its parent.
- Query order: Recall result from Search Theory: **query the cell that maximizes the ratio of detection probability to cost**.
- Each node calculates its expected *forward* detection cost and forward detection probability by using the same information about its children nodes. This is then propagated backwards from the leaves all the way up to the source node.
- The algorithm is essentially a form of dynamic programming, reminiscent of the Bellman-Ford Algorithm.

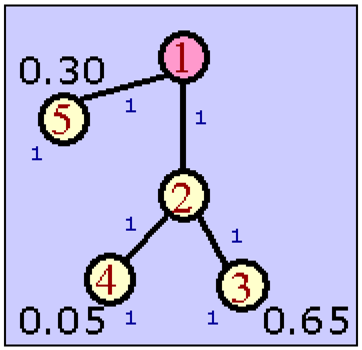
DSQFA example

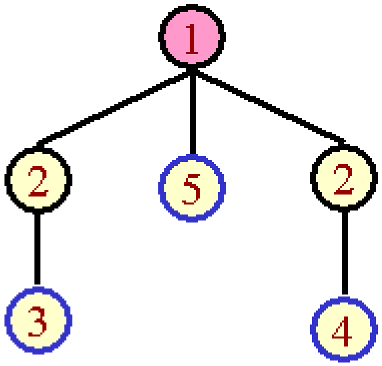




Is DSQFA Optimal ?

- If we require that each node can only be queried once, then it appears intuitive that DSQFA will yield the optimal paging tree, in terms of minimizing the expected total cost of messaging and paging. But if this restriction is removed, then there may be a lower cost paging tree:





Expected cost = 3.85

Future Work

- Design algorithms to obtain the optimal sequential paging scheme for a given graph.
- Consider formulations involving delay constraints.
- Obtain results on the estimation error sensitivity of sequential paging in multi-hop wireless networks.
- Study the optimization gains of these algorithms.
- Develop techniques for updating and maintaining probabilistic location estimates.

END OF TALK