Lecture 1: Mobility Management in Mobile Wireless Systems

Ing-Ray Chen

CS 6204 Mobile Computing

Virginia Tech

Mobility Management

- Location Management
  - Search: find a mobile user’s current location
  - Update (Register): update a mobile user’s location
  - Location info: maintained at various granularities (cell vs. a group of cells called a registration area)
  - Research Issue: organization of location databases
    - Global Systems for Mobile (GSM) vs. Mobile IP vs. Wireless Mesh Networks (WMN)
- Handoff Management
  - Ensuring that a mobile user remains connected while moving from one location (e.g., cell) to another
  - Packets are routed to the new location

Handoff Management

- Decide when to handoff to a new access point (AP)
- Select a new AP from among several APs
- Acquire resources such as bandwidth channels (GSM), or a new IP address (Mobile IP)
  - Channel allocation is a research issue: goal may be to maximize channel utilization, satisfy QoS, or maximize revenue generated
- Inform the old AP to reroute packets and also to transfer state information to the new AP
- Packets are routed to the new AP

Tradeoff in Location Management

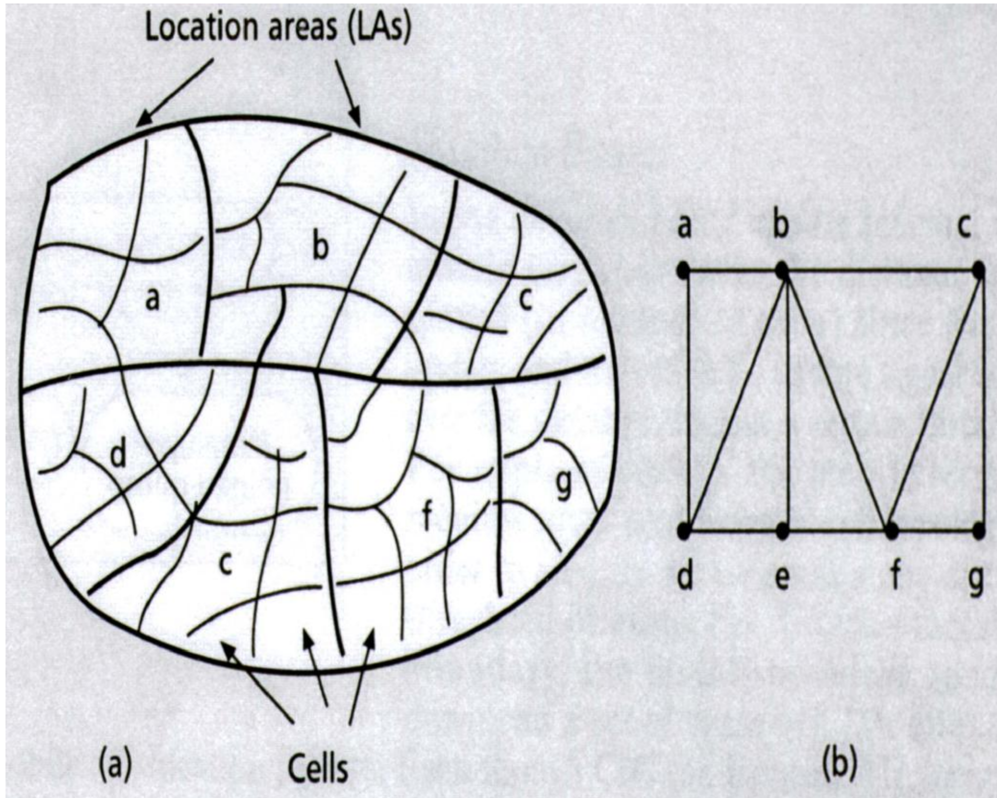
- Network may only know approximate location
- By location update (or location registration)
  - Network is informed of the location of a mobile user
- By location search or terminal paging
  - Network is finding the location of a mobile user
- A tradeoff exists between location update and search
  - When the user is not called often (or if the service arrival rate) is low, resources are wasted with frequent updates
  - If not done and a call comes, bandwidth or time is wasted in searching

Registration Area (RA) in the Basic HLR-VLR Scheme

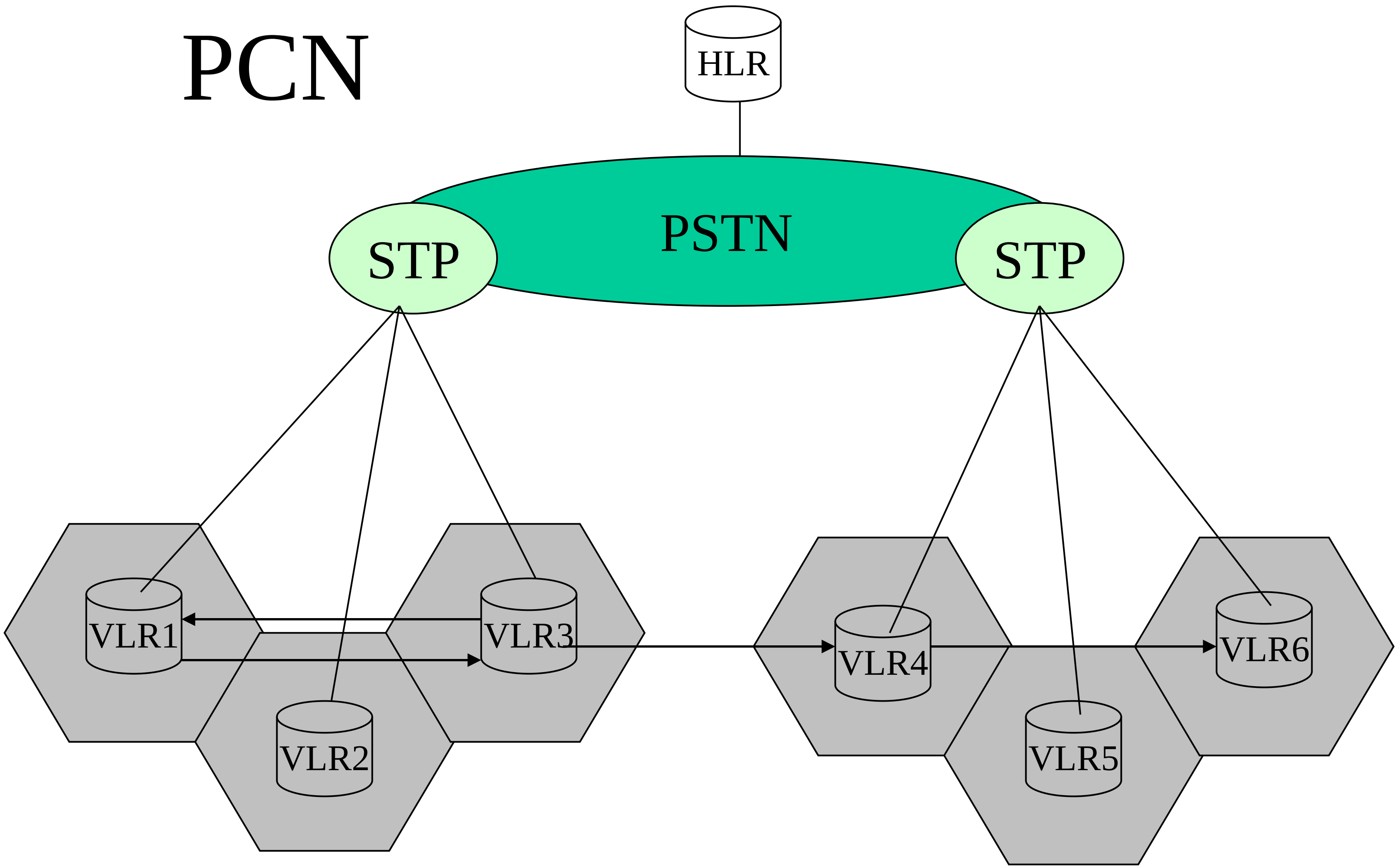
- Current Personal Communication Service (PCS) networks (i.e., cellular networks) use RA-based basic HLR-VLR schemes:
  - The service coverage area is divided into registration areas (RAs), each with a visitor location register (VLR)
  - Each RA covers a group of base stations (cells).
  - A user has a permanent home location register (HLR)
  - Base stations within the same RA broadcast their IDs
  - If ID is sensed different by the mobile terminal, then a cell boundary is crossed and a location update is sent to the VLR of the current RA.
  - When crossing a RA boundary, an update is sent to the HLR.
  - A search goes by HLR->VLR->cell->paging (by the base station)

Registration Areas in a PCN

- The figure on the right shows a PCS network
  - RA topology
  - RA graph model



PCN



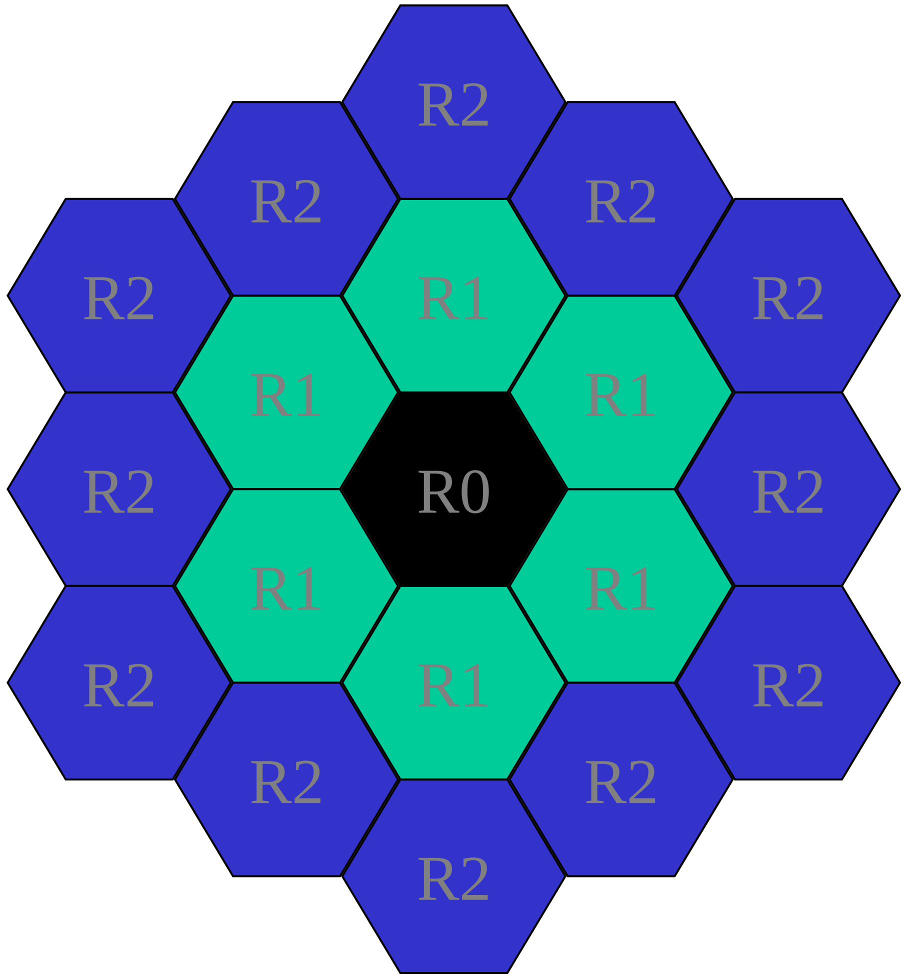
PSTN: Public Switched Telephone Network

STP: Service Transfer Point

HLR: Home Location Register

VLR: Visitor Location Register

Hexagonal Network Coverage Model for PCN



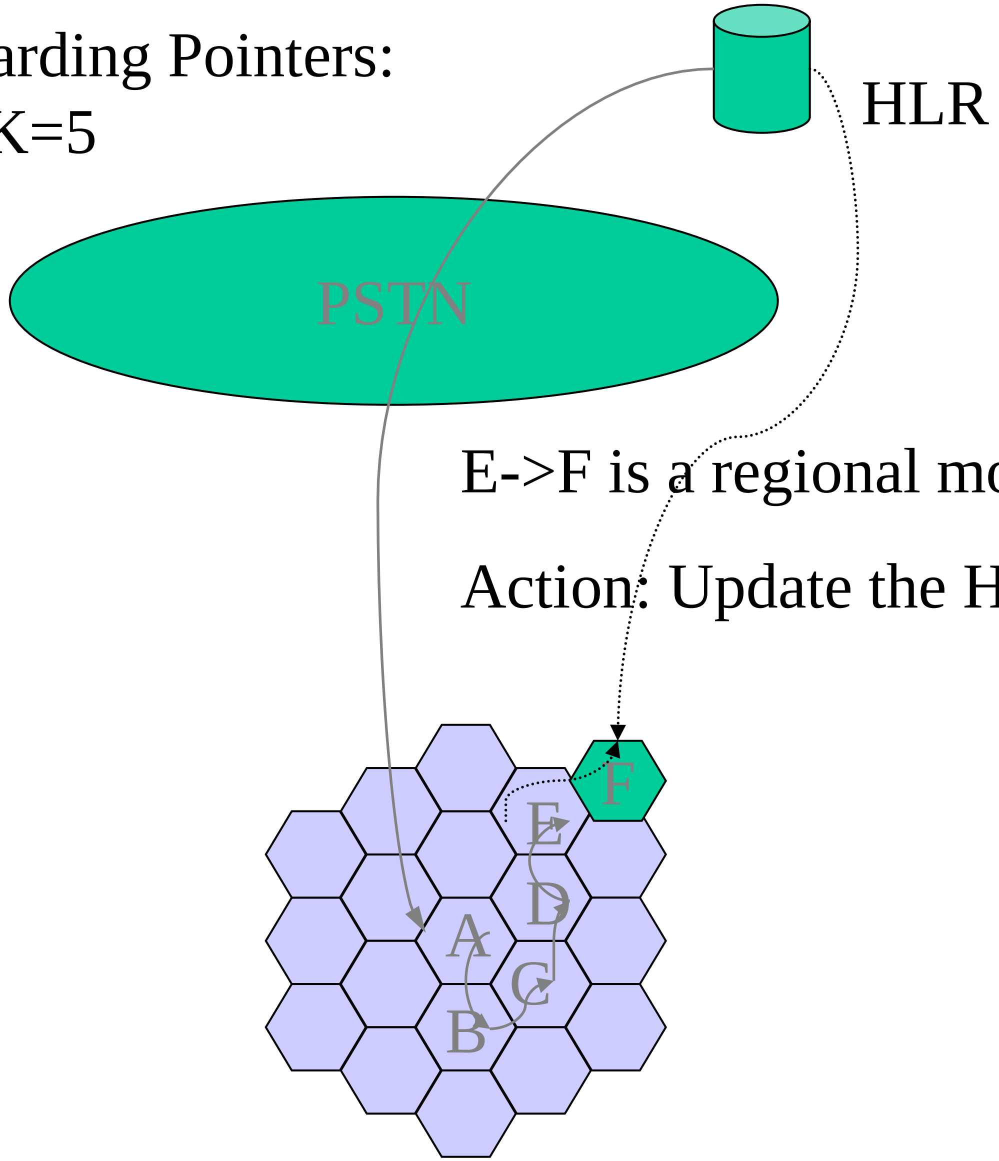
Forwarding Pointers (Ref [1])

- Update (upon crossing a RA boundary)
  - When the length of forwarding pointers < K:

    Set up a pointer between the two involving VLRs
  - When the length of forwarding pointers = K:

    Update information to the HLR
- Search
  - HLR -> VLR_0 -> …. -> VLR_i -> cell ->paging

Forwarding Pointers:
with K=5



E->F is a regional move since length = 5

Action: Update the HLR of the new address F

A->B->C->D->E are all local movements since the length of the forwarding chain is less than K=5

Action: Put a forwarding pointer between the two involving VLRs

How Big Should K be for Forwarding Pointers?

- Suppose that the cost "saving" due to forwarding for a location update operation is τ where τ is the cost of accessing a remote registrar (approximately).
- The "increased" cost per search operation is $K\tau$ to follow the forwarding pointers of length K.
- Let λ be the call arrival rate (incurring search) and σ be the mobility rate (incurring location update). Then the increased cost due to search operations per unit time is $\lambda K\tau$, while the cost saving due to update operations per unit time is $\sigma\tau$
- When $\sigma\tau > \lambda K\tau$, or $\sigma/\lambda > K$, it makes sense to have forwarding pointers. In other words, K should be bounded by σ/λ, the reciprocal of λ/σ, or the reciprocal of the call to mobility ratio (**CMR**)

What's the problem of this analysis?

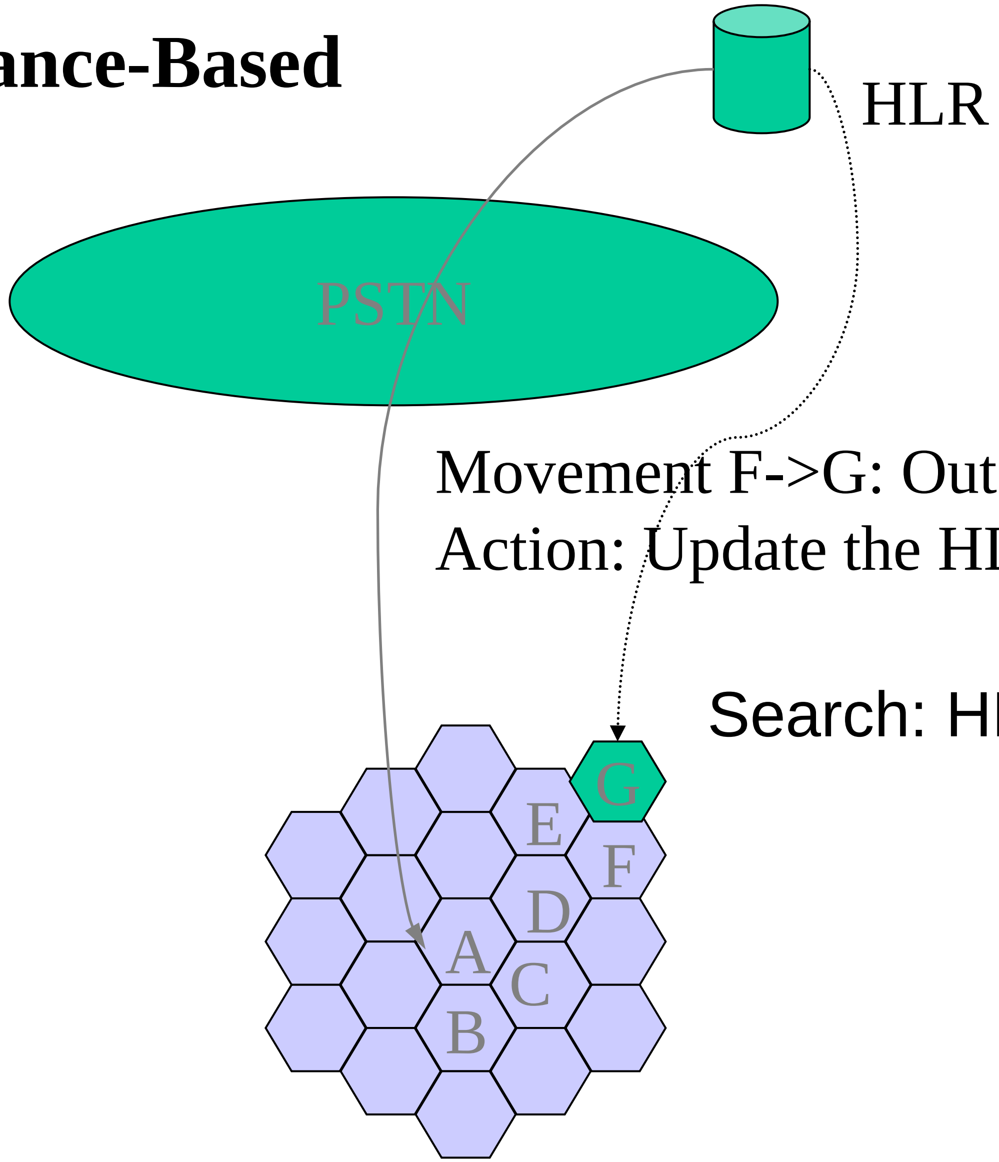
Dynamic Location Update

- Location update algorithms can be static or dynamic
  - With static, an update is triggered because of crossing of RA boundaries, e.g., the basic HLR-VLR scheme
  - With dynamic, update or not depends on a user's "dynamic" call and mobility patterns
- Dynamic Location update schemes:
  - Time-Based, Movement-Based, Distance-Based

Dynamic Location Update Schemes

- Time-Based: A mobile terminal updates in every T time units
- Movement-Based: A mobile terminal counts the number of boundary crossings and performs the update when a threshold is exceeded (e.g. M=6)
  - Forwarding pointers can be considered as a variation of it
- Distance-Based: A mobile terminal tracks the distance (in terms of RAs) it has moved since the last update
  - Update is performed when a distance threshold is exceeded
  - Mobile terminal needs some knowledge of the network topology
  - Local Anchor can be considered as a variation of it.

Distance-Based

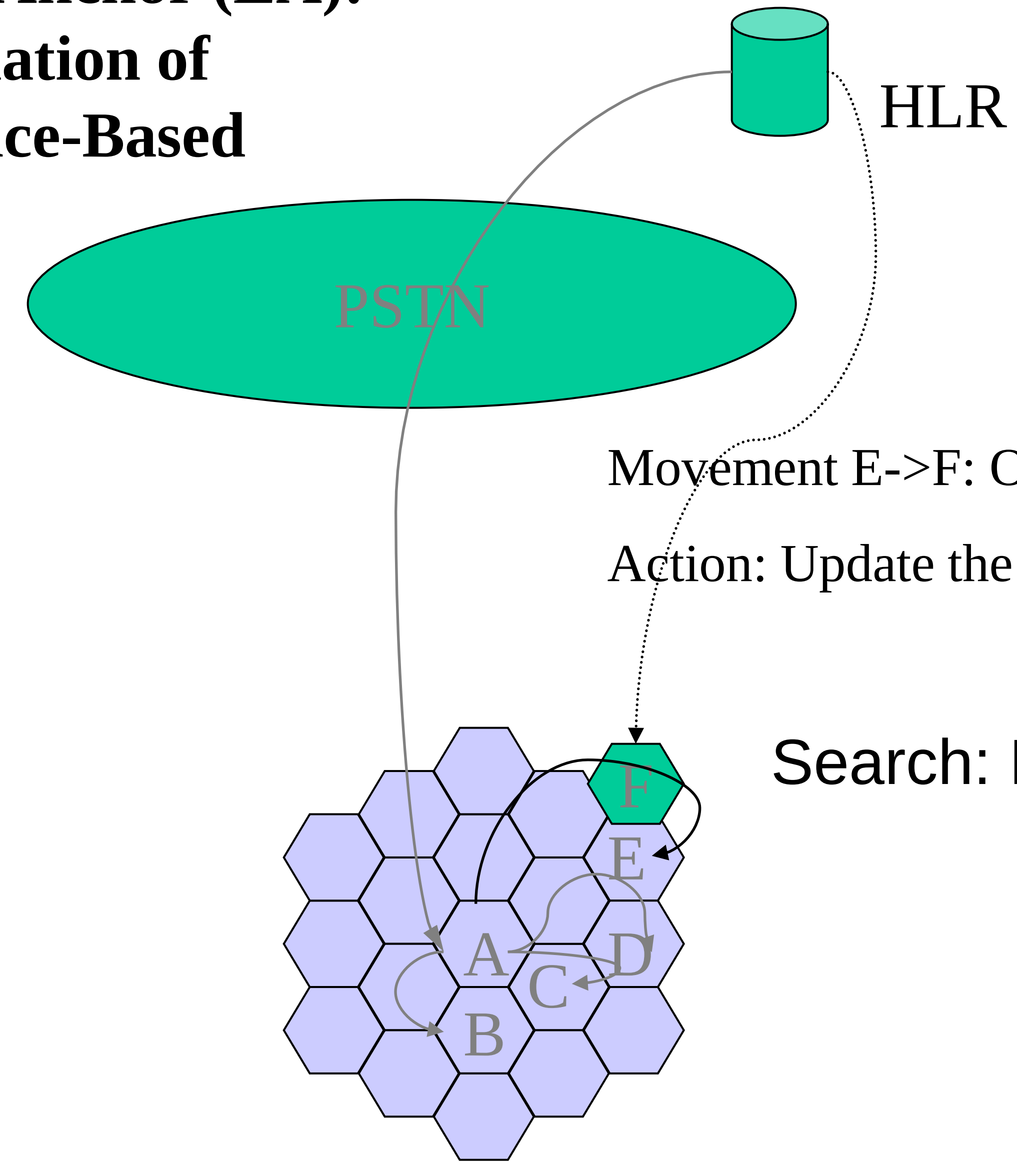


Movement F->G: Outside of 2-RA Distance
Action: Update the HLR of the new address G

Search: HLR-> last VLA->paging

Movements A->B, B->C, C->D, D->E, E->F: all within 2-RA distance
Update Action: None

Local Anchor (LA): A Variation of Distance-Based



Movement E->F: Outside of a 2-RA Anchor Area

Action: Update the HLR of the new anchor F

Search: HLR -> LA -> current VLR
-> paging

Local Movement: within the 2-RA Anchor Area

Update Action: Update the LA of the new VLR

This is the difference between the LA scheme vs. a pure distance-based scheme

Update Time Interval for Time-Based Schemes

- Design parameter: T (the time interval for performing a location update)
- Assume a search operation performs an expanding ring search
- Let λ be the call arrival rate and σ be the mobility rate.
- The maximum area to be searched for is a circle with the radius of $\sigma * \min(1/\lambda, T)$ cells.
- Normalize each search operation with a cost of $\sigma * \min(1/\lambda, T)$
- Normalize the cost of each update operation with a cost of 1
- The cost of time-based management per unit time is:

$$C = \lambda * \sigma * \min(1/\lambda, T) + 1/T$$

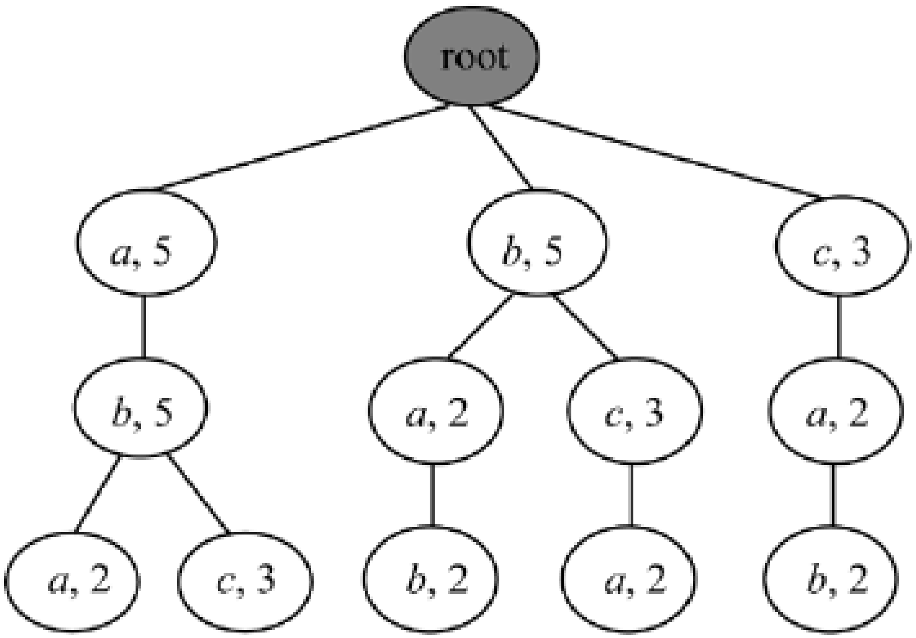
- When $1/\lambda < T$ (frequent arrivals), $C = \sigma + 1/T \approx \sigma$
- When $1/\lambda > T$ (infrequent arrivals), $C = \lambda * \sigma * T + 1/T$. Take $dC/dT=0$, $T_{opt} = 1/\mathrm{sqrt}(\sigma\lambda)$, implying that when either σ or λ decreases, T_{opt} increases

LeZi Update

- Based on a compression algorithm by Ziv and Lempel
- LeZi is a **path-based** update algorithm by which the movement history, not just the current location, is sent in an update message from the mobile user to the location database server
  - The history has a list of IDs of the zone (RA or cell) the mobile terminal has crossed
  - The location database keeps the history in compact form by means of a search tree structure (called a **LeZi trie**)
    - Can be part of the user's profile
  - On a call arrival, prediction of the current cell is given and selective paging is performed

Location Prediction Example

- Input sequence: … ababcabcababc (c is the current cell)
  - The LeZi Trie so far:



  - What's the next cell?
    - P_a: Probability that the next cell is *a*

- Estimate the probability of the next cell using 0, 1 and 2 last cells history:
- P_{0a}=5/13, P_{0b}=5/13, P_{0c}=3/13
- P_{1a}=1, P_{1b}=0, P_{1c}=0 (after c)
- P_{2a}=1, P_{2b}=0, P_{2c}=0 (after bc)

$$P_a = w_2 \cdot P_{2a} + w_1 \cdot P_{1a} + w_0 \cdot P_{0a}$$
$$P_b = w_2 \cdot P_{2b} + w_1 \cdot P_{1b} + w_0 \cdot P_{0b}$$
$$P_c = w_2 \cdot P_{2c} + w_1 \cdot P_{1c} + w_0 \cdot P_{0c}.$$

Per-User Location Caching

Lazy Caching: Cached information is updated on location search

Q: When is it beneficial to cache a callee's location in a caller's cell or a registration area?

- Let a callee's CMR be λ/σ, representing the number of search operations between two consecutive update operations.
- Let P_h be the cache hit ratio.
- First search after an update operation will result in a cache miss, after which the remaining (CMR-1) search operations will result in cache hits. Thus P_h =(CMR-1)/CMR.
- Let A_l be the local access cost and A_r be the remote access cost. Then $A_l + (1- P_h) A_r < A_r$, i.e., $P_h > A_l / A_r$, or, equivalently, CMR $> A_r /(A_r - A_l)$, meaning that caching is beneficial if user's call-to-mobility-ratio CMR $> A_r /(A_r - A_l)$.

Per-User Location Caching

<u>Eager Caching:</u> Cached information is updated on location update

Q: When is it beneficial to cache a callee's location in a caller's cell or a registration area?

- Let T be an observation period
- Let λi be the average number of calls in cell i for the mobile user during T
- Let σ be the number of location updates by the mobile user during T
- Let α be the cost savings when a local lookup succeeds
- Let β be the cost of updating a cache copy.
- The location of the mobile user is cached at cell i only if the cost of savings outweighs the cost of location update, i.e.,

$$\alpha * \lambda i > \beta * \sigma$$

Working Set under Eager Caching

- Working Set under eager caching: the working set of a mobile user *m* is the set of registrars (e.g., cells or RAs) that maintain location information about *m*
- A sliding window of length T is maintained by the system to estimate λi and σ for mobile user *m*
- When a new operation occurs, the working set membership is dynamically maintained as follows:
  - When the operation is a search operation from registrar *i*: If registrar *i* is not in the working set and if $\alpha * \lambda i > \beta * \sigma$ is true, then add registrar *i* to the working set
  - When the operation is an update operation: All the registrars in the working set are evaluated and if $\alpha * \lambda i > \beta * \sigma$ is not met, then delete registrar *i* from the working set.

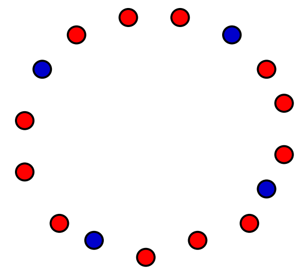
Replicating Location Information

- A mobile user's location information may be replicated to a number of registrars for fault tolerance (like eager caching)
- Two different organizations:
  - Flat: No structure exists among the registrars
  - Hierarchical: A multiple-level tree structure exists to organize location registrars

Replicating Location Information based on **Flat Organization**

- Consider having k replicas:
  - Separating them by equal distance: Placing k replicas at registrars i, (i+s) mod n, (i+2s) mod n, …, [i+(k-1)s] mod n where n is the total number of registrars and s = n/k.
  - What is the best value of k?
  - The update cost is k location registrars per update
  - The search cost is n/k location registrars accesses per search
  - The normalized overall cost per time unit is $C=k\sigma + (n/k)\lambda$, which is minimized when k_{opt}=sqrt(n*CMR)
- As CMR (i.e., λ/σ) increases, k_{opt} increases

Example:
k=4, n=16



Replicating Location Information based on **Hierarchical Organization**

- A tree of location registrars:
  - A location registrar (LR) that is a leaf node in the tree has information on all the mobile users in its RA
  - A non-leaf LR replicates location information in all the location registrars in the subtree rooted to it.
  - The root registrar in the tree has information on all the mobile users in the system.

Replicating Location Information based on **Hierarchical Organization**

- Search: Let the callers be in RA_i and the callee be in RA_j. Let LCA(i, j) be the registrar that is the **least common ancestor** of LR_i and LR_j. The registrars along the path from the leaf registrar LR_i to LCA(i, j) will be searched until the callee information is found.
- Update: If a mobile user moves from RA_i to RA_j, then location information is deleted in all the registrars along the path from RA_i to LCA(i, j) (except LCA(i, j)), and the location information is updated in all the registrars along the path from root to RA_j.
- The cost of both the search and update is O(log n) where n is total number of registrars in the tree

Replication + Forwarding (Ref. [2])

- Under the replication + forwarding hybrid scheme, the system maintains *N* replicas on *N* most calling VLRs, and in addition a forwarding chain length of *K*, for each user.
- How many replicas to be used depends on the user's CMR value and its call profile
- Each of the N replicas stores the identical content, i.e., a pointer to the first VLR of the forwarding chain
- The HLR determines the user's best (*N*, *K*) combination; the HLR knows exactly which calling VLRs keep a replica by consulting the user's call arrival profile.
- On a user mobility event, if *K* is reached, then the HLR is updated to point to the new VLR as in the basic HLR-VLR scheme, the forward chain is reset as in the forwarding scheme, and all N replicas are updated to store the location of the new VLR as in the replication scheme.

Replication + Forwarding Example

- Consider a mobile user's call pattern given as follows:
  - Hit ratio P=0% when N=0 (trivial condition),
  - Hit ratio P=50% when N=1,
  - Hit ratio P=70% when N=2, and
  - Hit ratio P=80% when N=3.
- For each CMR, determine the optimal forwarding length K under each of these possible (N, P) combinations
- For each CMR, determine the best (N, K) among all that will minimize the network communication cost
  - More N will incur more update cost but reduce search cost
- At runtime, based on the CMR, the HLR applies the best (N, K) to minimize the cost.

Mobile Terminal Paging

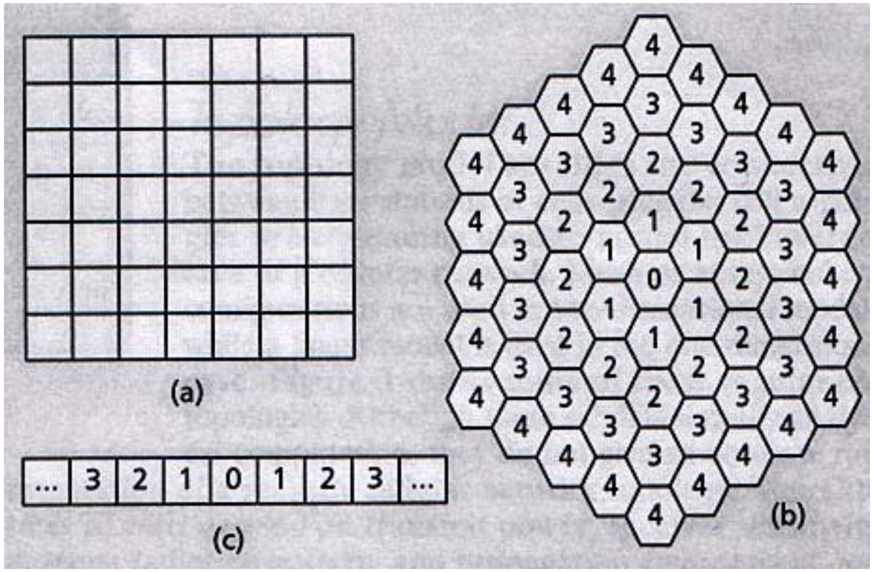
- A process by which the network determines the exact location of a particular mobile terminal
- Polling cycle or search iteration:
  - Polling signals sent over a downlink control channel where the mobile terminal is likely to be
  - If a reply is received before a timeout, the polling ends; otherwise, a new group of cells is chosen
  - A call is dropped when the mobile terminal is not located within an allowable time constraint
  - "Maximum paging delay" is the maximum number of polling cycles allowed to locate a mobile terminal
- The Paging cost is proportional to the number of polling cycles as well as the number of cells polled in each cycle

Terminal Paging (1)

- Blanket Polling
  - All cells within an LA are polled at once when a call arrives.
  - The mobile terminal is located in 1 polling cycle
  - Currently deployed on top of LA-based update schemes in existing PCN’s
  - Paging cost is high due to a large number of cells in an LA

Terminal Paging (2)

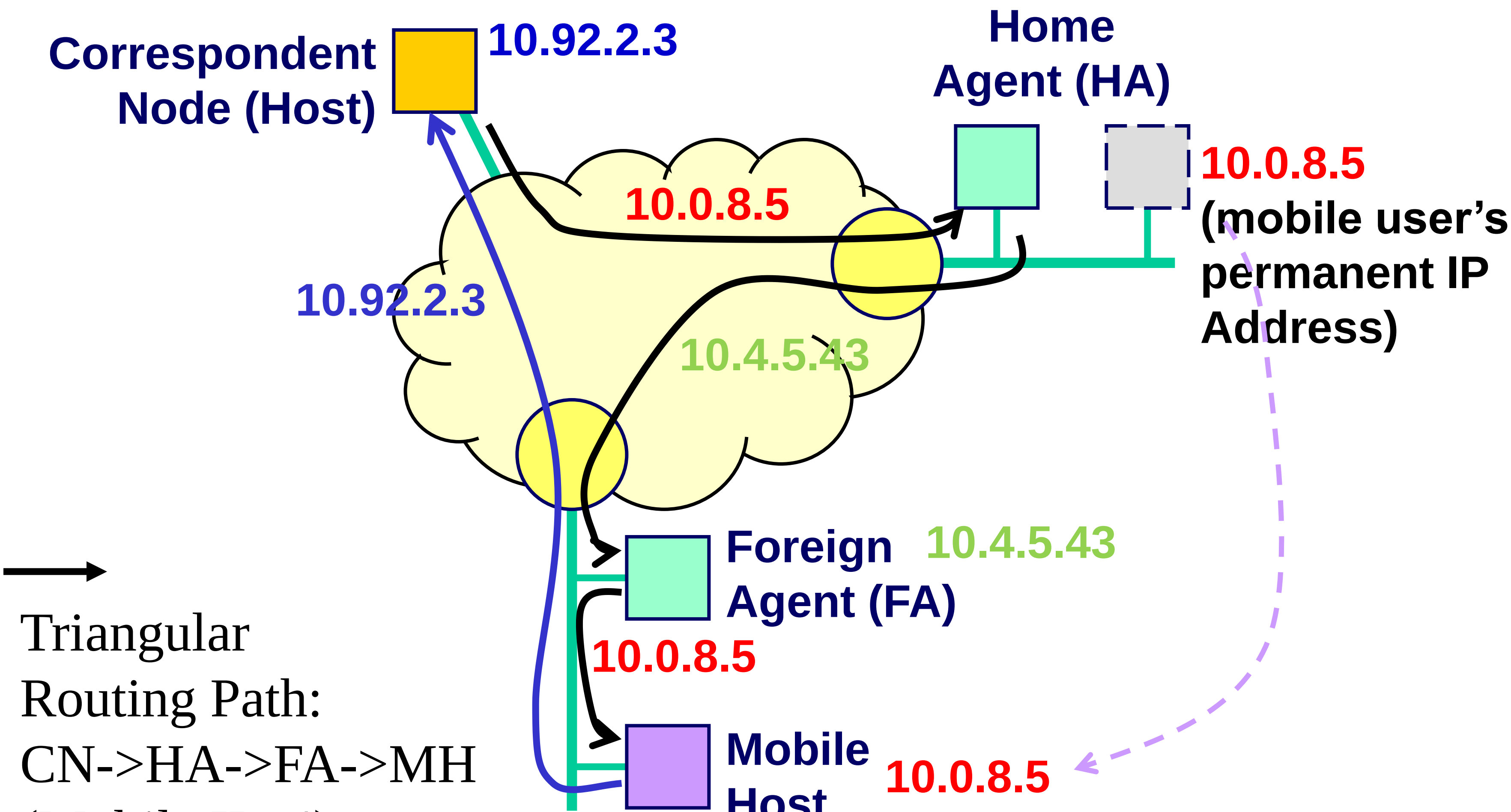
- Shortest-Distance-First (Expanding ring search)
  - Starts at last known mobile terminal location
  - Moves outward in a shortest-distance first order.



Terminal Paging (3)

- Sequential Paging Based on a User's Location Probability
  - Current location is predicted based on its location probability distribution (e.g., LeZi algorithm)
  - Groups of cells can be polled by selecting them with dynamic programming, when using a maximum paging delay constraint

Basic Mobile IP: Home Agent and Foreign Agent



Triangular
Routing Path:
CN->HA->FA->MH
(Mobile IPv4)

* Route Optimization (CN->MH) available in Mobile IPv6

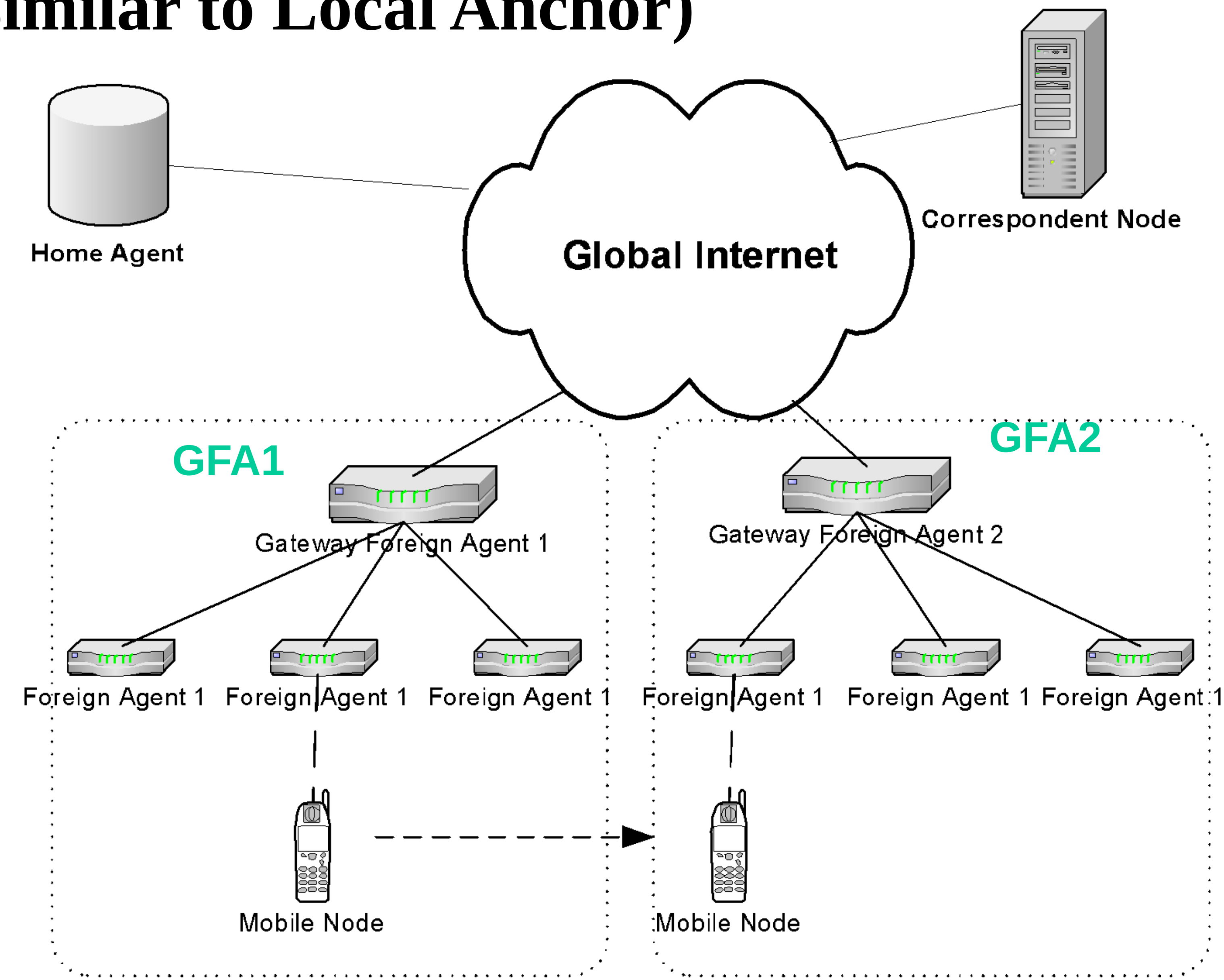
Mobile IP Operation

- A HA or FA advertises it availability using agent-advertisement messages
  - A mobile host may optionally solicit an agent-advertisement message
- A mobile host receives an agent-advertisement message and decides if it is on a FA or HA network
- If the mobile host is on a FA network, it obtains a care-of address (CoA) on the foreign network:
  - Foreign agent’s IP address
  - Co-located CoA statically or dynamically through DHCP (this is the only way in Mobile IPv6)

Mobile IP Operation

- **Binding Update**: Mobile host registers the new COA with its HA possibly via a FA
- **Tunneling**: HA intercepts datagrams sent to the mobile node's home address and tunnels datagrams to the registered COA
- Tunneled datagrams could be received:
  - By the foreign agent and delivered to the mobile host, or
  - directly to the mobile node (if a co-located COA is allocated)
- Mobile host can send datagrams directly back to the CN
- **Route Optimization**: In Mobile IPv6, MH also registers new COA with the CN which can communicate directly with MH without the overhead of triangular routing.

Regional Mobile IP for Mobility Management (similar to Local Anchor)

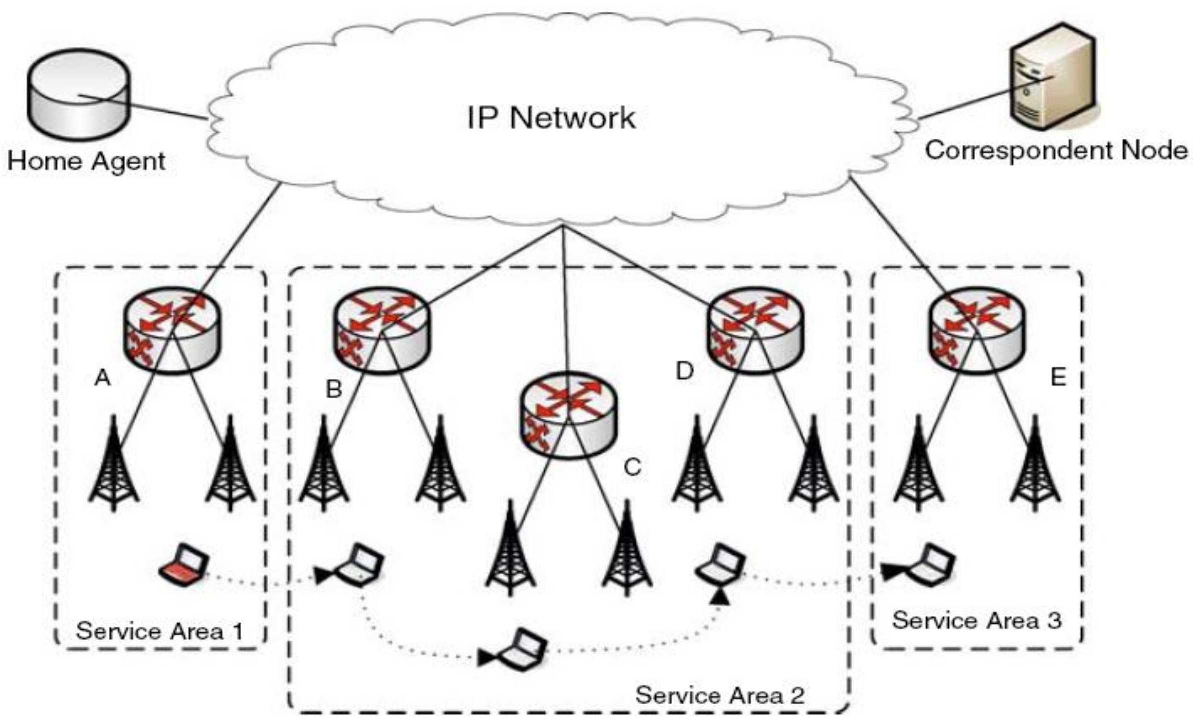


Hierarchical Mobile IPv6

- Mobile Anchor Points (MAPs) acting as GFAs structured in a hierarchical manner
- MAPs in HMIPv6 are statically configured and shared by all mobile nodes in the system
- **Local Subnet Handoff**: CoA registration to the local MAP upon a local subnet handoff within the MAP service domain area
- **MAP Domain Handoff**: Regional CoA (RCoA) registration to the HA and CN upon a regional handoff into a new MAP service domain
- Research issue: what's the right size of a MAP service area? How many levels?

Dynamic MAP (Ref [3])

- All IP routers can serve as MAPs on demand
- Mobile device selects a MAP dynamically based on its service rate to mobility rate ratio (SMR)
- Optimal DMAP service size is dynamic (see the figure below) and is determined by exploring the tradeoff between service management cost (for packet delivery) vs. location management cost (for location update)



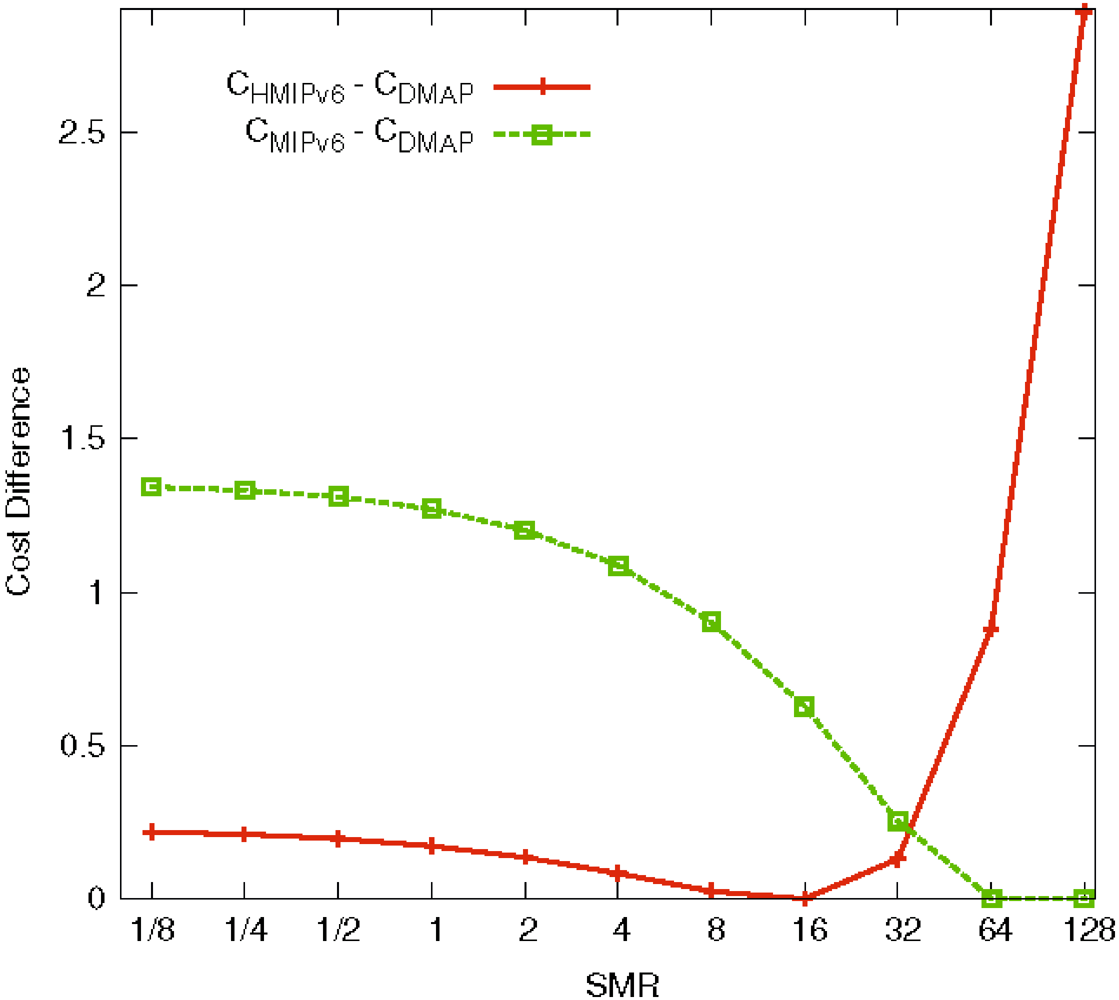
Performance of Dynamic MAP

- DMAP degenerates to basic MIPv6 when SMR becomes sufficiently large (the DMAP domain size decreases as SMR increases)
- DMAP degenerates to HMIPv6 when the MAP size is fixed as that in HMIPv6

SMR	C_{MIPv6}	$C_{\mathrm{HMIPv6}}(K_H = 4)$	$C_{\mathrm{DMAP(K_{opt})}}$	K_{opt}
0.1250	1.8750	0.7897	0.5522	34
0.2500	2.0000	0.9187	0.6892	32
0.5000	2.2500	1.1766	0.9619	28
1.0000	2.7500	1.6925	1.5034	22
2.0000	3.7500	2.7242	2.5758	16
4.0000	5.7500	4.7876	4.6960	11
8.0000	9.7500	8.9144	8.8859	7
16.0000	**17.7500**	**17.1681**	**17.1681**	**4**
32.0000	33.7500	33.6754	33.5475	2
64.0000	65.7500	66.6901	65.7500	1

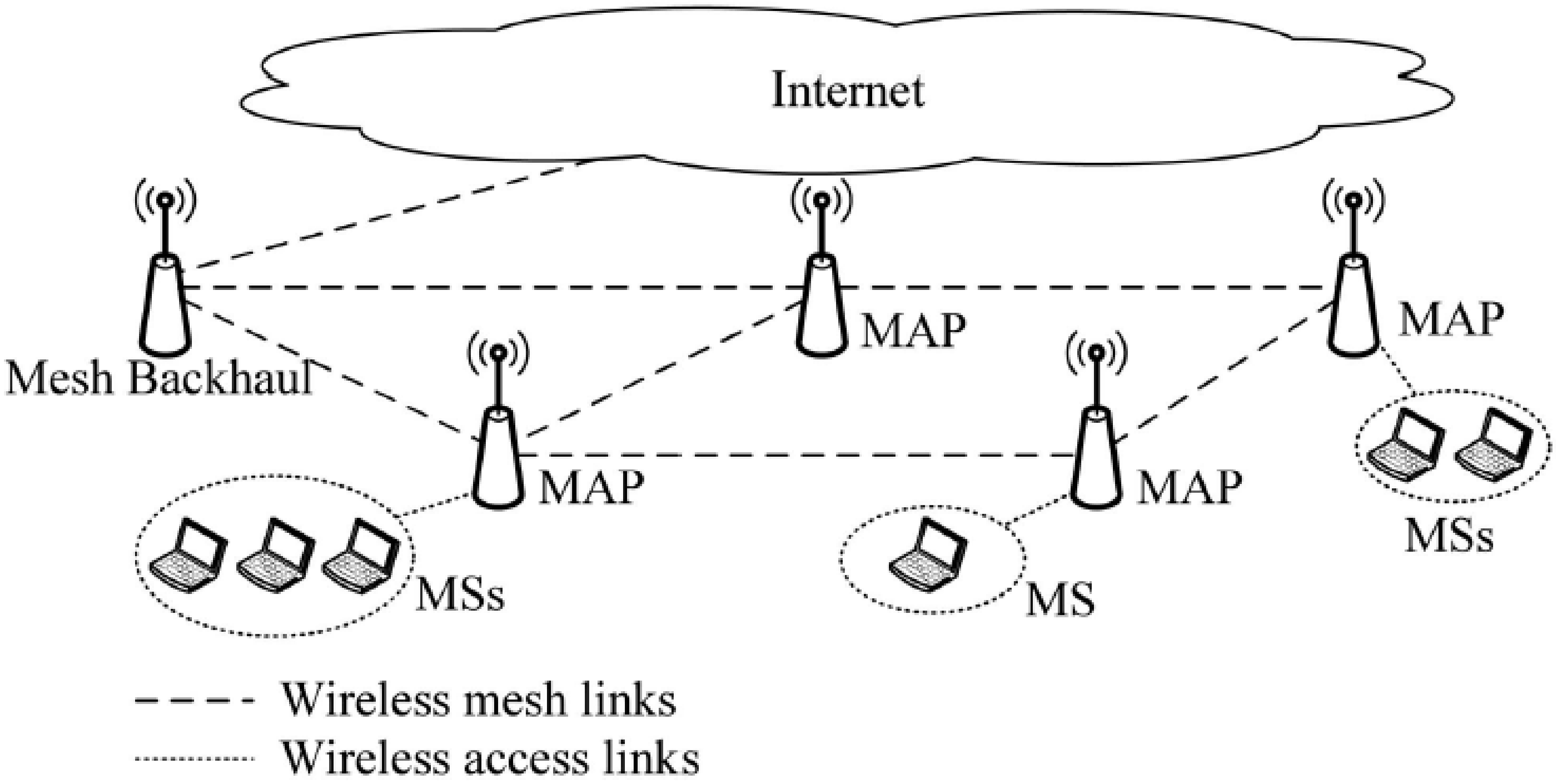
Dynamic MAP vs. MIPv6 and HMIPv6

- The cost difference between HMIPv6 and DMAP initially decreases as SMR increases until K_{opt} coincides with K_H at which point DMAP degenerates to HMIPv6, and then the cost difference increases sharply as SMR continues to increase.



Mobility Management in Wireless Mesh Networks (Ref [4])

- Wireless Mesh Networks (WMNs) have emerged as one of the major technologies for 4G high speed mobile networks
- Architecture:
  - A mesh backhaul gateway (MG) connects the WMN with the internet.
  - Stationary mesh access points (MAPs) or mesh routers (MRs) provide wireless network access service to mobile stations (MSs).


Internet
Mesh Backhaul
MAP
MAP
MAP
MAP
MSs
MS
MSs
Wireless mesh links
Wireless access links

WMN Mobility Management Mechanism (WMM)

- Based on location cache – a MS’s location information is cached at all MRs which perform both routing and location management functions
- Based on full replication and weak consistency – all MRs cache a MS’s location
- Based on opportunistic cache location updates through routing – A MR updates its cache about a MS when it routes a packet issued by the MS.

WMM: Location Management / Handoff Management

- **Location Management:** When a MS changes its MR, the new serving MR information is updated to other MRs opportunistically through routing when the MS sends a packet which routes through these relaying MRs.

- **Handoff Management:** During data transmission, if the MS changes from an old MR to a new MR, handoff management enables the old MR to forward user data to the new MR **based on pointer forwarding**

WMM: Location Cache Update

- **Opportunistic location cache updates -** When MRs route packets for an MS, they store the MS location information (in the packet header) in the location cache
- **Routing a packet to a MS**: a MR can correctly route the packets for a MS by referencing its location cache and routing table.
  - If there is no location information in the location cache, the packet is forwarded to the mesh backhaul gateway.
  - If the mesh gateway does not cache the MS location information when processing packet routing, a location query procedure (through flooding) is executed to obtain the MS location information
- Key issues for performance:
  - Minimize the location query probability because flooding is expensive
  - Control location cache updates based on a MS's SMR characteristics

Summary

- Mobility management includes location management and handoff management
- Location management must explore the tradeoff between location search cost and location update cost based on each user's call (service) and mobility characteristics:
  - Time-based
  - Movement-based (forwarding is a variation of it)
  - Distance-based (local anchor is a variation of it)
- Caching and replication techniques can be used to provide search efficiency and fault tolerance but must be used with care not to dramatically increase update costs. In many cases, caching and replication (or the hybrid of both) must base on each user's call (service) and mobility characteristics.
- Mobile IP supports mobility management in IP networks. Mobility management techniques for PCN can be applied to Mobile IP networks (e.g., Regional/Hierarchical Mobile IP/DMAP in MIP vs. forwarding/Local Anchor in PCN).
- Routing-based mobility management with full replication and opportunistic cache updates at mesh routers has been proposed for wireless mesh networks

References

Most of the materials are based on Chapter 2, F. Adelstein, S.K.S. Gupta, G.G. Richard III and L. Schwiebert, *Fundamentals of Mobile and Pervasive Computing*, McGraw Hill, 2005, ISBN: 0-07-141237-9.

Other References:

1. I.R. Chen and B. Gu, "A comparative cost analysis of degradable location management algorithms in wireless networks," *The Computer Journal*, Vol. 45, No. 3, 2002, pp. 304-319.
2. I.R. Chen and B. Gu, "Quantitative analysis of a hybrid replication with forwarding strategy for efficient and uniform location management in mobile wireless networks," *IEEE Transactions on Mobile Computing*, Vol. 2, No.1, 2003, pp. 3-15.
3. I.R. Chen, W. He and B. Gu, "DMAP: integrated mobility and service management scheme in Mobile IPv6 systems," *Journal of Wireless Personal Communications*, 2007.
4. D.W. Huang, P. Lin, C.H. Gan, "Design and performance study for a mobility management mechanism (WMM) using location cache for wireless mesh networks," *IEEE Transactions on Mobile Computing*, Vol. 7, No. 5, 2008, pp. 546-556.