# Improving the latency of the Probe Phase during 802.11 Handoff


William Arbaugh

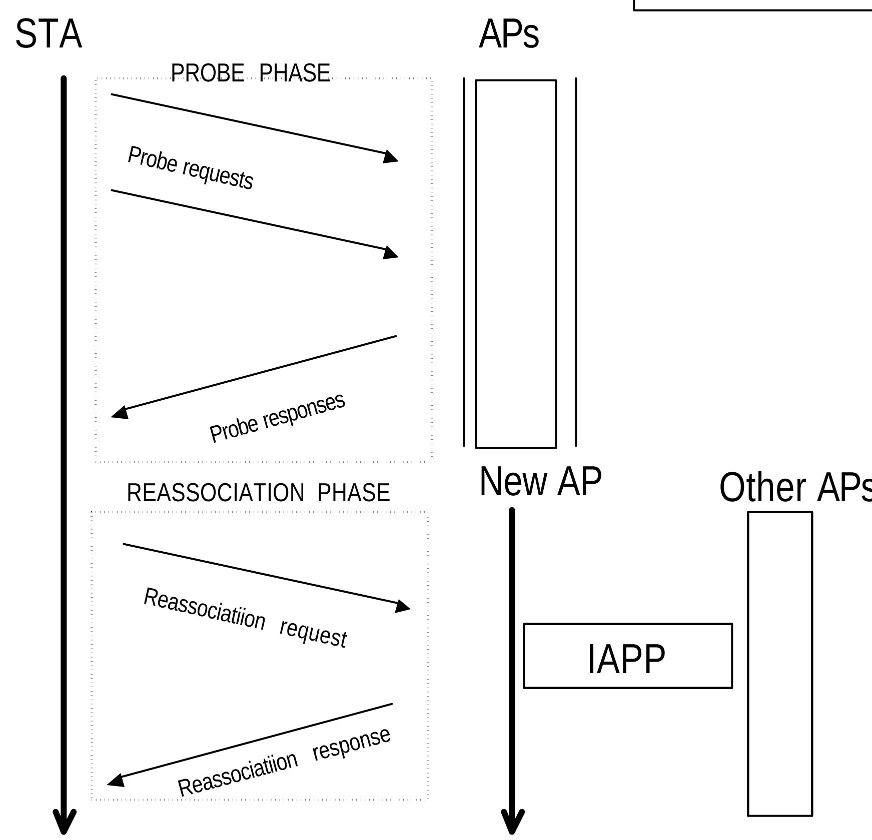
*University of Maryland*

*College Park*


UNIVERSITY OF MARYLAND

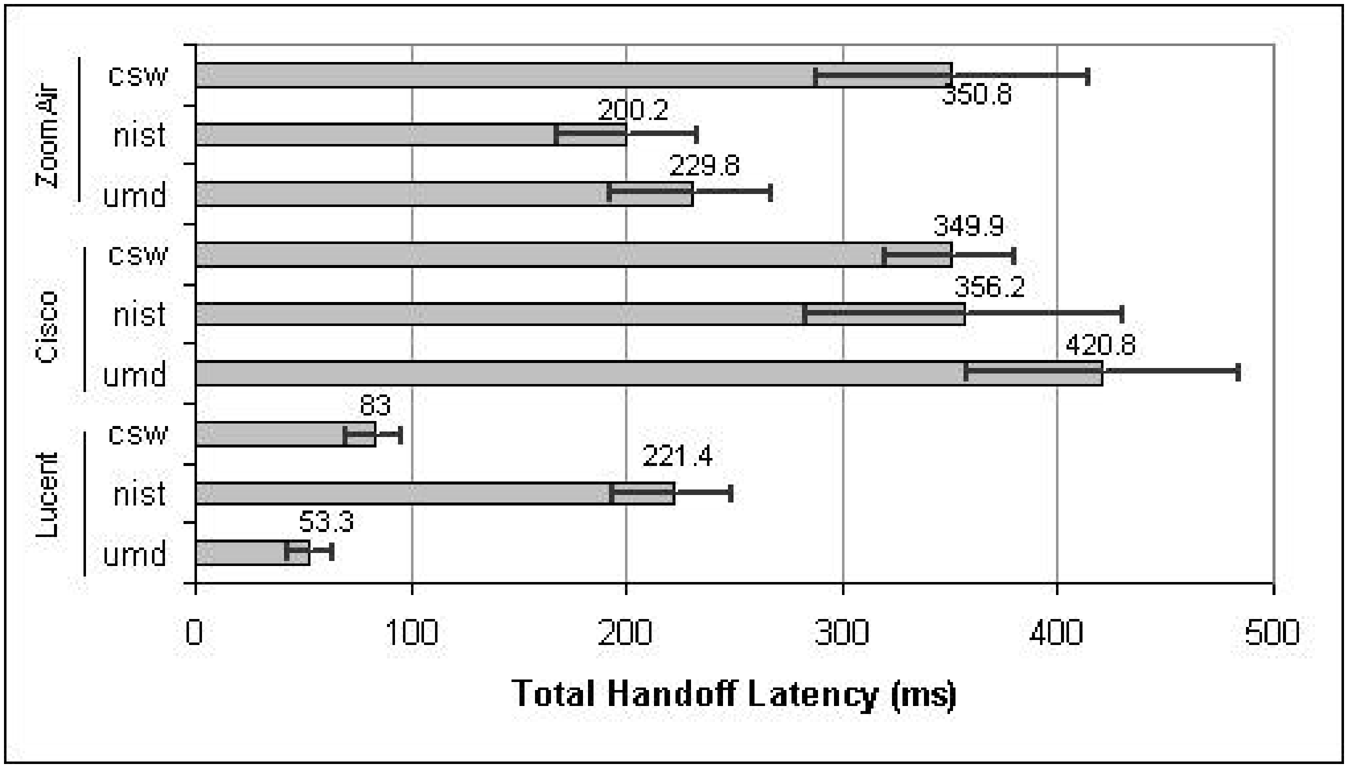
# The Handoff Procedure

- Probe Phase
  - STA scans for APs
- Reassociation Phase
  - STA attempts to *associate* to preferred AP

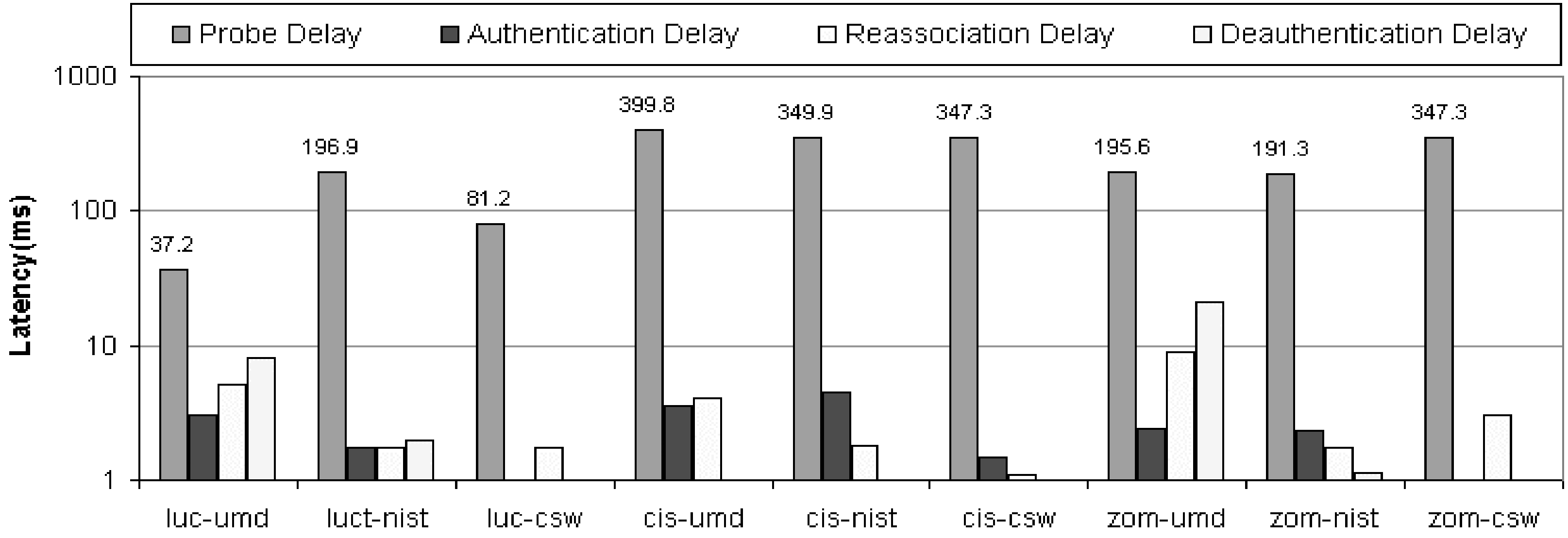


# The Handoff Procedure – Probe Phase

- Empirical Results:
  - High latencies
  - Large variation



# Probe Phase is Dominate Factor



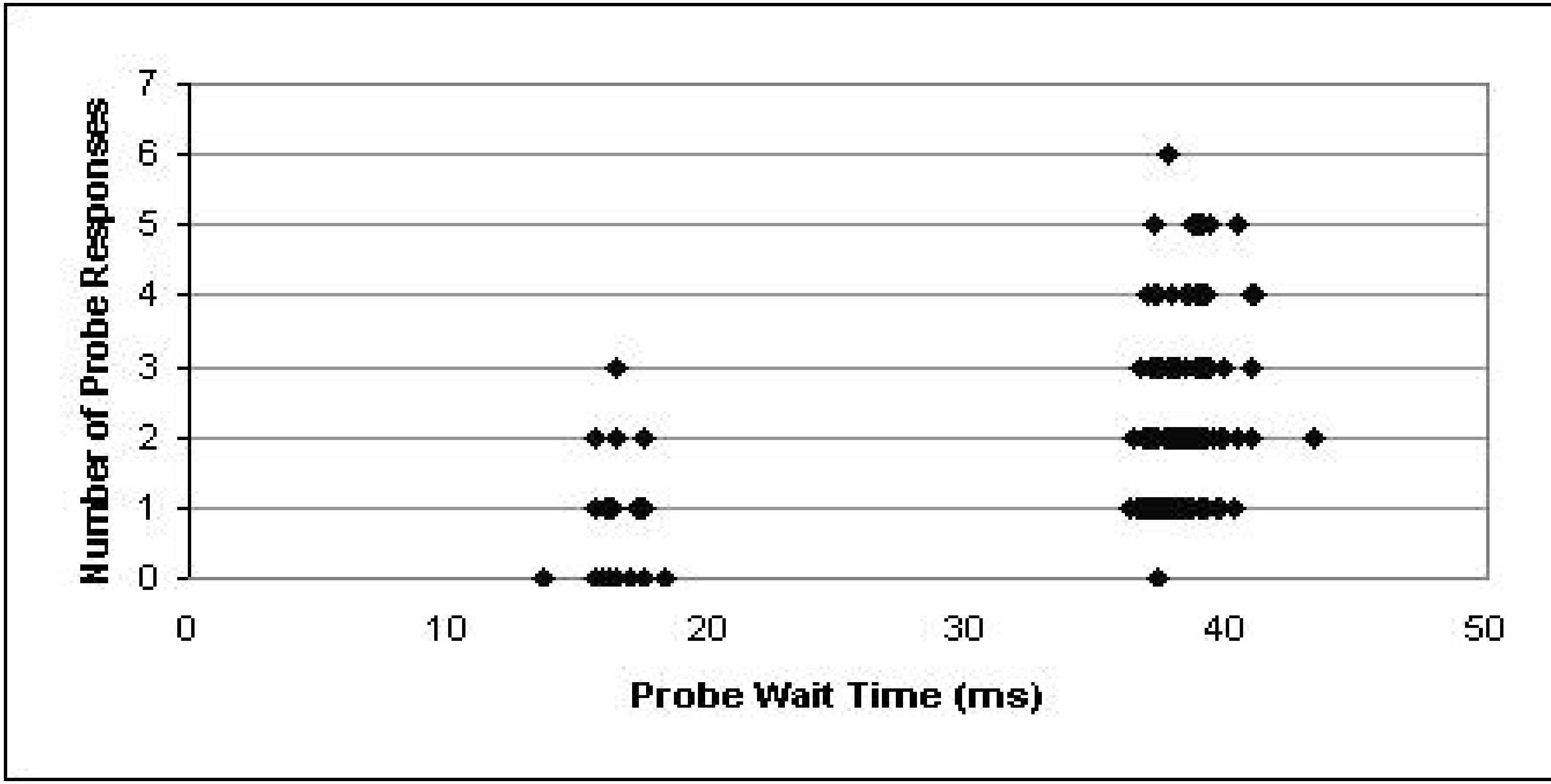
# Probe Phase Bounds

- Bounds of Probe Phase defined by standard

N * MinChannelTime <= t <= N * MaxChannelTime

When N is the number of channels, and t is the total measured probe time.

# Cisco 340 Probe Wait per Channel Clusters



# Current Methods to Reduce Probe Time

- Beacons are sent at 1mbps on adjacent channels so a form of binary search can be used, i.e. only probe non-overlapping channels.

  PROBLEM: Still must probe a subset of available channels.

- Passive or active probing done during idle time.

  PROBLEM: Support for fast moving STA's and/or those with heavy loads.

# Problem Becomes Worse with 802.11a

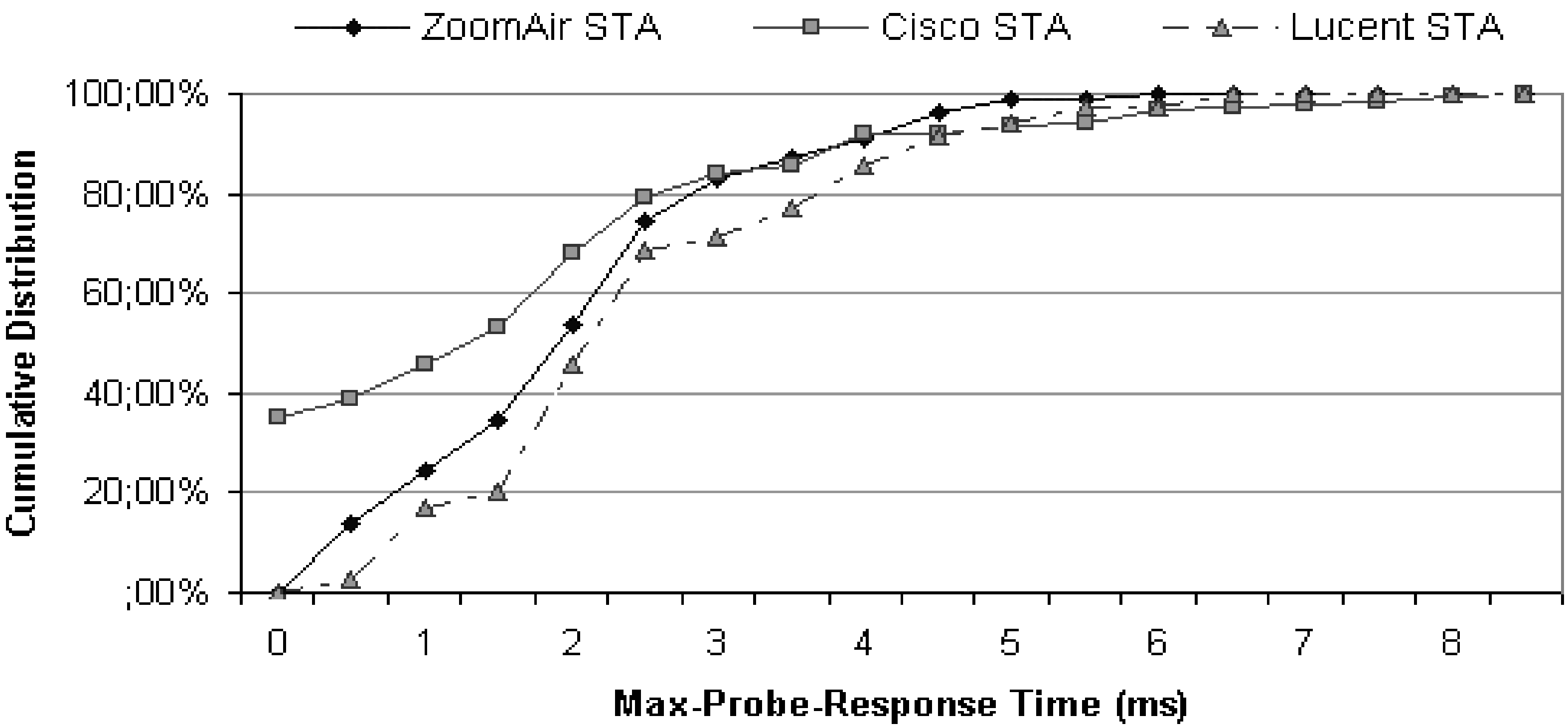
- 8 non-overlapping channels.
- Multiple current probe phase times by 2.67!!!

  $99.32 <= t <= 1067.46$
- <u>**No way**</u> to do real time synchronous applications and support hand-offs.

# Another Solution

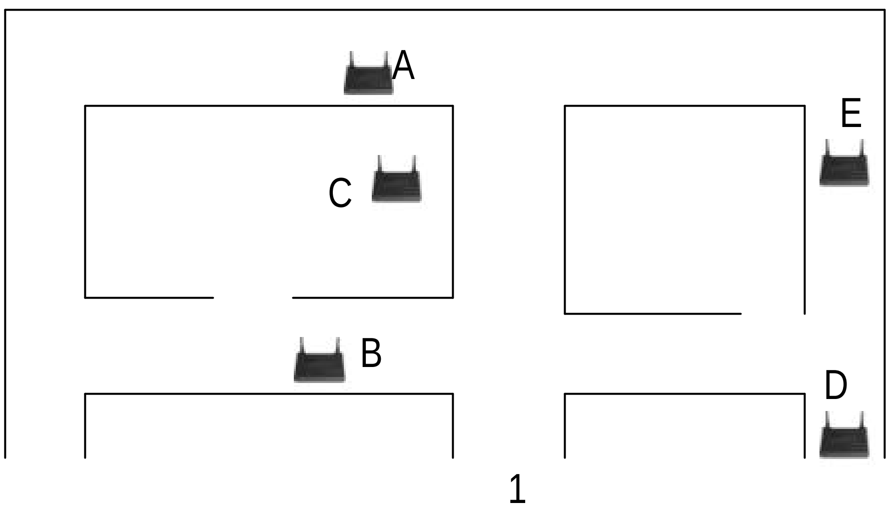
- Instrument APs to support knowledge of neighboring APs, e.g. Neighbor graphs
- STA maintains an optimal channel time
- Now Probe (Actively or Passively) during idle time or prior to roam only on the neighboring AP channels and only wait the optimal channel time.

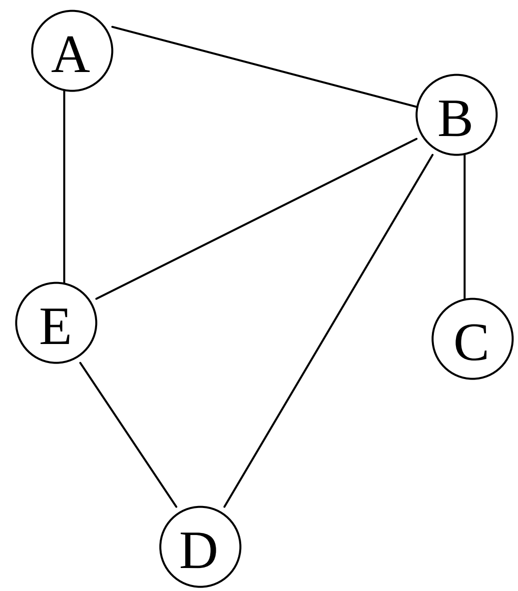
# Optimal Channel Time



# AP Neighborhood Graph

- Two APs *i* and *j* are neighbors if
  - Exists a path of motion between *i* and *j* such that it is possible for a mobile STA to perform a *reassociation*
  - Captures the '*potential next AP*' relationship
  - Distributed data-structure i.e. each AP maintains list of neighbors

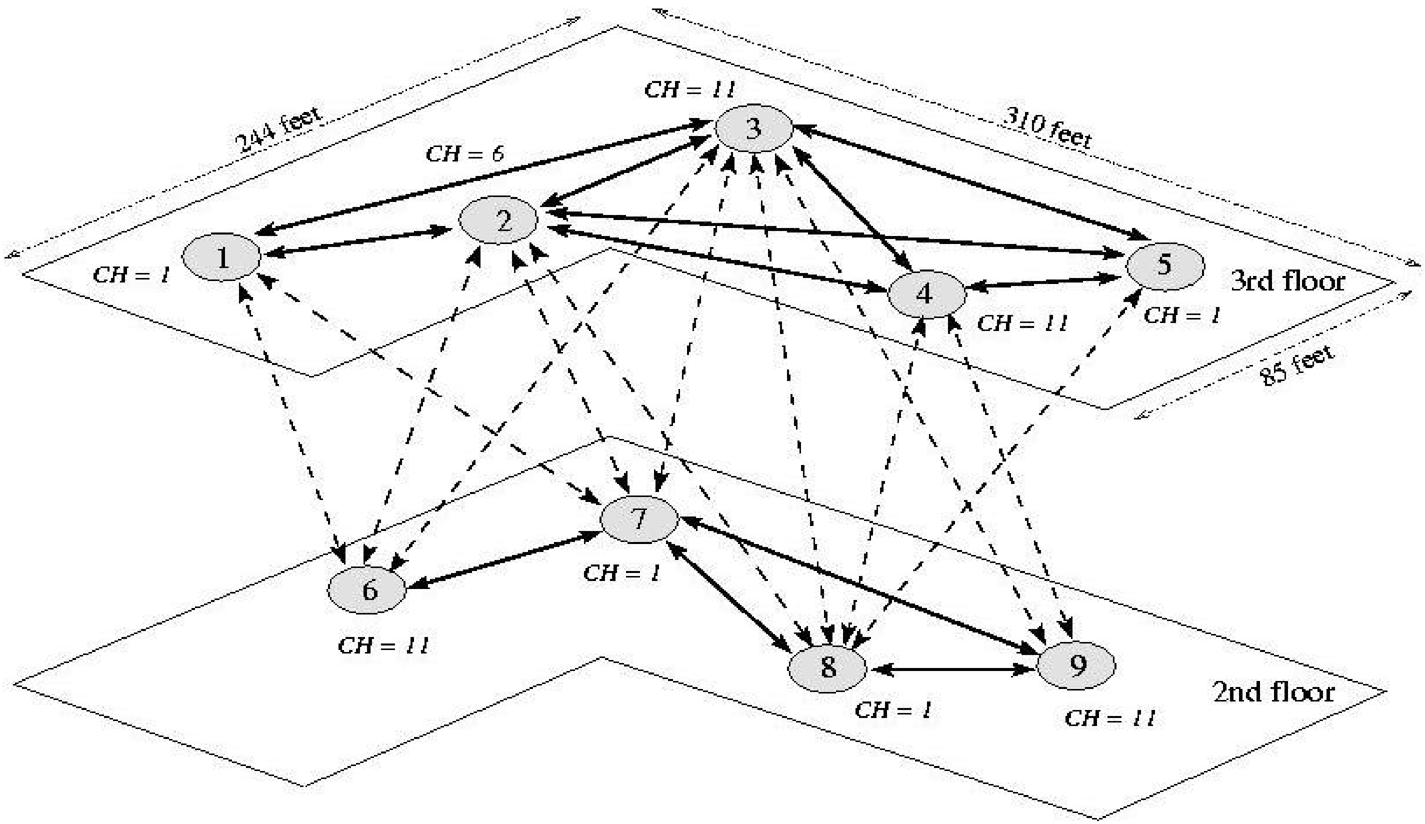


# AP Neighborhood Graph – Automated Learning

- Construction
  - Manual configuration for each AP or,
  - APs can learn:
    - If STA *c* sends *Reassociate Request* to AP *i*, with old-ap = AP *j*
      - Create new neighbors *(i,j)* (i.e. an entry in AP *i*, for *j* and vice versa)
    - Learning costs only one '*high latency handoff*' per edge in the graph
    - Enables mobility of  APs, can be extended to wireless networks with an ad-hoc backbone infrastructure

# Experimental Neighbor Graph



# Example

- STA is associated with AP3
  - | Neighbors(AP3) | = 8
  - | UniqueChannels(Neighbors(AP3)) | = 3
- STA probes on three channels and waits a MinChannelTime of 3ms and a MaxChannelTime of 7ms

9 ms <= t <= 21 ms

# Conclusions

- Current Probe times are not adequate for multimedia applications
- Probe times in 802.11a will be over two times worse.
- Using Neighbor graphs and an optimal channel wait can reduce the Probe phase significantly.

# Future Work

- Extending Neighbor Graphs to Interworking, i.e. AAA to AAA communications.
- Trying to start an IRTF working group on WiFi handoffs.
- Light up the beltway