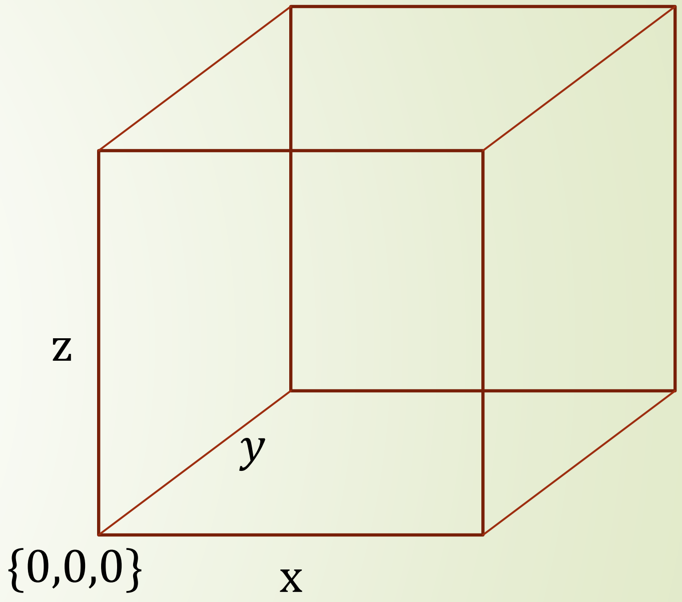
Location Determination

Framework and Technologies

Meaning of Location

- Three Dimensional Space
- Reference Coordinate System
  - Global – GPS
  - Local
  - Application Specific
- Multiple References
  - Ability to Map
- Notation
- $X = \{x, y, z\}$

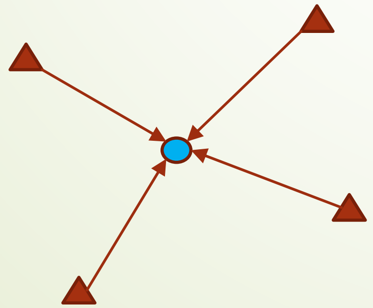


Location Uses

- All levels of accuracies have applications
- Outdoors
  - Navigation
    - Automobiles/ Road Vehicles
    - Aircrafts
    - Boats/Ships
    - Personal – walking/jogging/running
  - Targetting
  - Finding Hospitals/Gas Stations….
- Indoors
  - Advertising
  - Finding …
  -
- System based vs. device based

How

- Benchmarks
  - Known locations (Accuracy?)
  - Unknown Location WRT the location of Benchmarks
- What Form ??
  - Physical, marked locations
  - Location of devices



- What do I measure??
  - Proximity
  - Distance
  - Some function of distance
  - Direction
  - Some function of direction
- How many measurements
  - 3
  - 4
- Use Geometry
  - Triangulation
  - Trilateration

Desirable Features

- In Doors and Out Doors operation
- Independent of GPS
- Rapidly Deployable
- Agnostic to Frequency Band or Protocol
- Accurate
- Scalable
- …

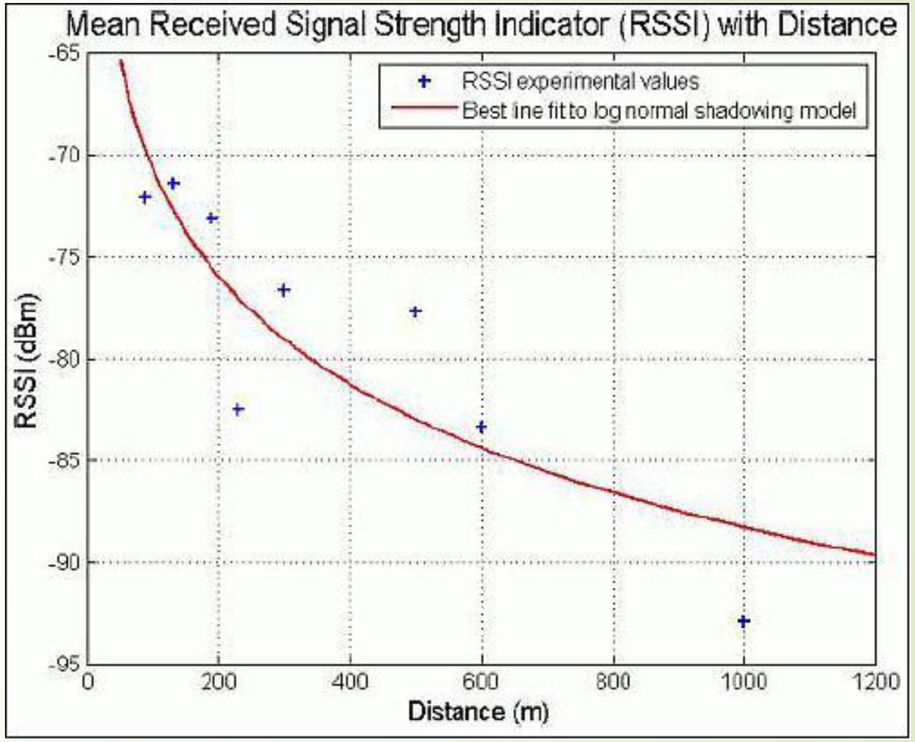
Proximity

- Detect the presence close to a known location
- RFID
  - Passive
    - Read by putting in a field of RF and reading the scatter pattern
    - Inventory Control
    - EZPass
  - Active
    - iBeacon
      - Using low power Bluetooth
    - Estimotes
    - ....

- How does Passive RFID approach compare with barcodes?
- FingerPrinting Based approach in WiFi Field

RF Field Based - WiFi

- AP – Generate Beacons 100 ms
- Can measure signal Strength
  - RSSI – Received Signal Strength Indicator
  - Included in spec to support handovers.
- RSSI – Relative scale or dbm
  - Most devices now report dbm
  - Range (-50 to -90 dbm)
  - Integer values only



Problem Formulation

- K Access Points
- Signal Field

$$S(X)$$

Where S is k dimensional vector and X is the location vector.

- Problem – The signal strength of K APs is measured by a device as signal vector S. Determine the location X where the device is

- Issues:
  - Is S an invertible function?
  - Does S have a closed form?
  - Is S deterministic or do the measurements vary with time

Signal Function

- Closed Form
  - Maxwell Equations
  - Affected by
    - Decay
    - Reflections
    - Refraction
    - Diffusion
    - Scattering
  - Some Approximations have been attempted
  - Outdoor – Cellular Phone
    - Accuracies ~200 meters
  - Indoor – WiFi
    - Accuracies 5-10 meters

- What should be K, the number of signal generators – APs.
- Most WiFi deployment is for supporting networking access and not for location.
- At a location one can only hear a small number of APs.
  - There are ~4500 APs on campus. How do we efficiently handle this 4500 dimensional function?

Stochastic nature of Signals

- Repeated measurements vary when nothing has changed
- There is some correlation among samples
- Signal Vector has to be treated as a stochastic vector
- As it is reasonable to assume that all APs operate independently the signals from them can be treated as independent random variables.
- Analytical models require the modeling of the randomness

FingerPrinting

- We can estimate the joint probability distribution of the signal vector

$$p\{S(X)\}$$

by empirical measurements

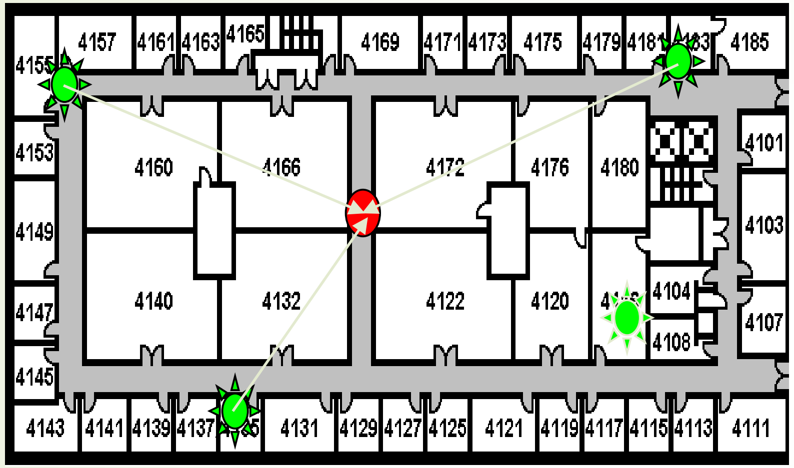
  - Discretize X and make measurements of S at known locations – a grid in X space
  - Treat the measurement points as benchmark points
  - Find the benchmark point closest to the device signal vector in signal space

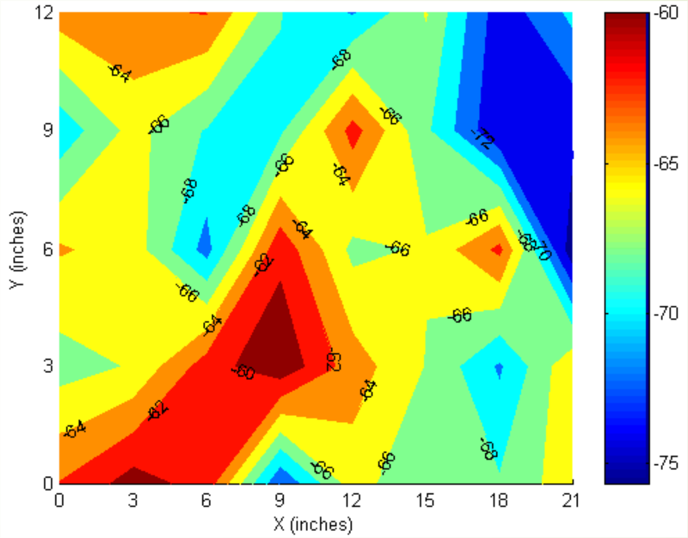
- May refine the location by determining a few closest benchmark points and interpolating

Horus: A WLAN-Based Indoor Location Determination System

Moustafa Youssef

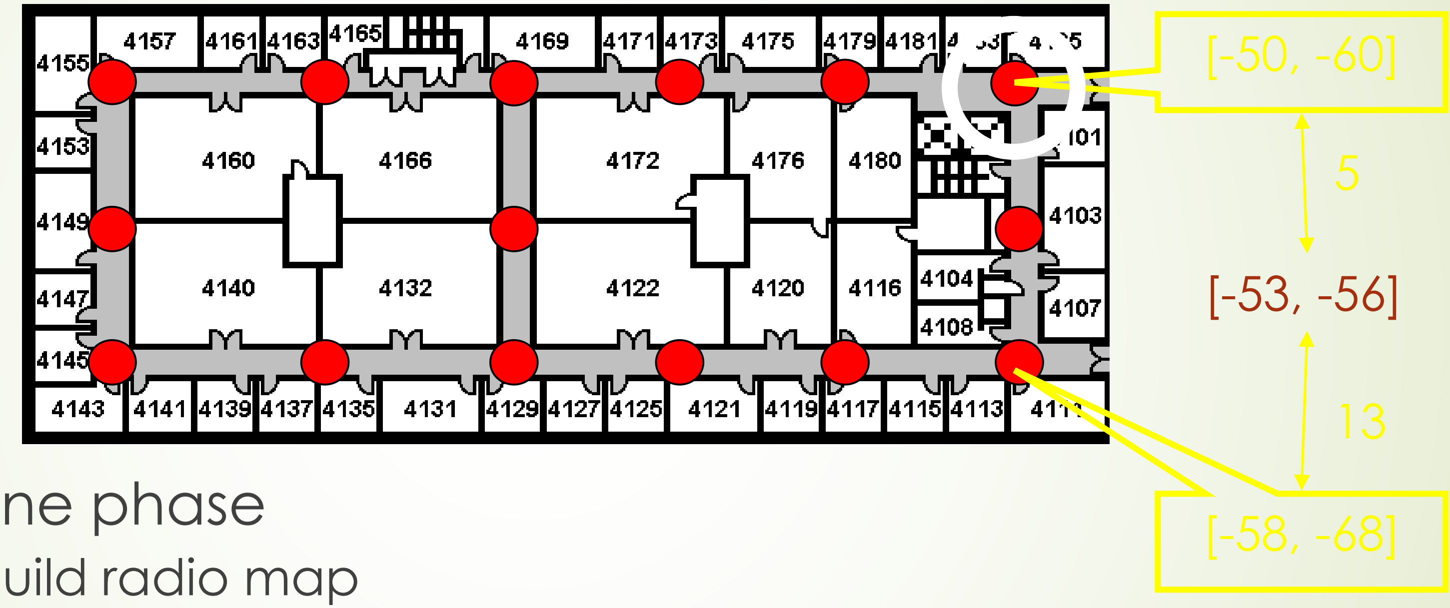
WLAN Location Determination (Cont'd)





- Signal strength= *f(distance)*
- Does not follow free space loss
- Use lookup table ≡ **Radio map**
- Radio Map: signal strength characteristics at selected locations

WLAN Location Determination (Cont'd)



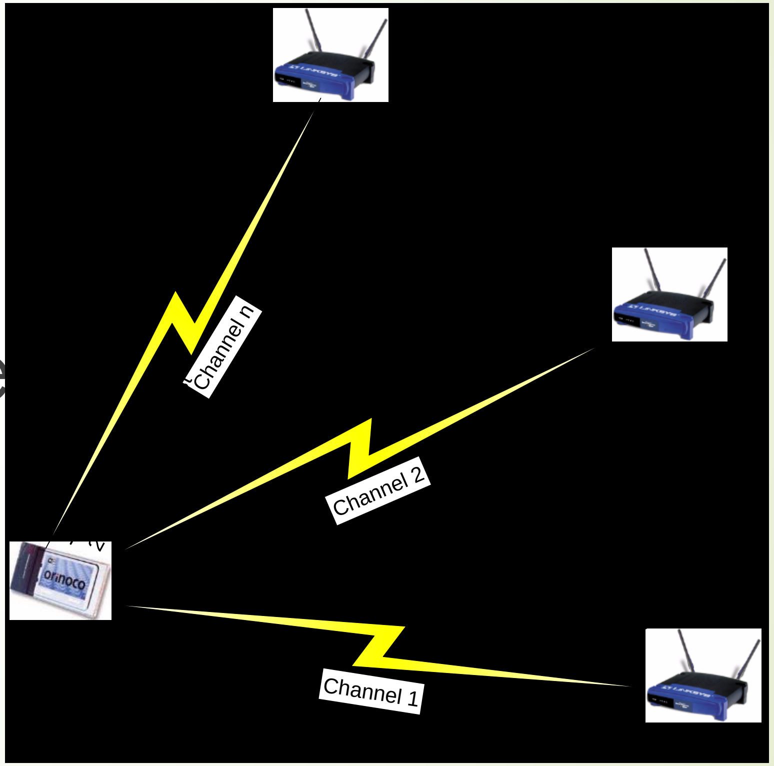
- Offline phase
  - Build radio map
  - Radar system: average signal strength
- Online phase
  - Get user location
  - Nearest location in signal strength space (Euclidian distance)

Horus Goals

- High accuracy
  - Wider range of applications
- Energy efficiency
  - Energy constrained devices
- Scalability
  - Number of supported users
  - Coverage area

Sampling Process

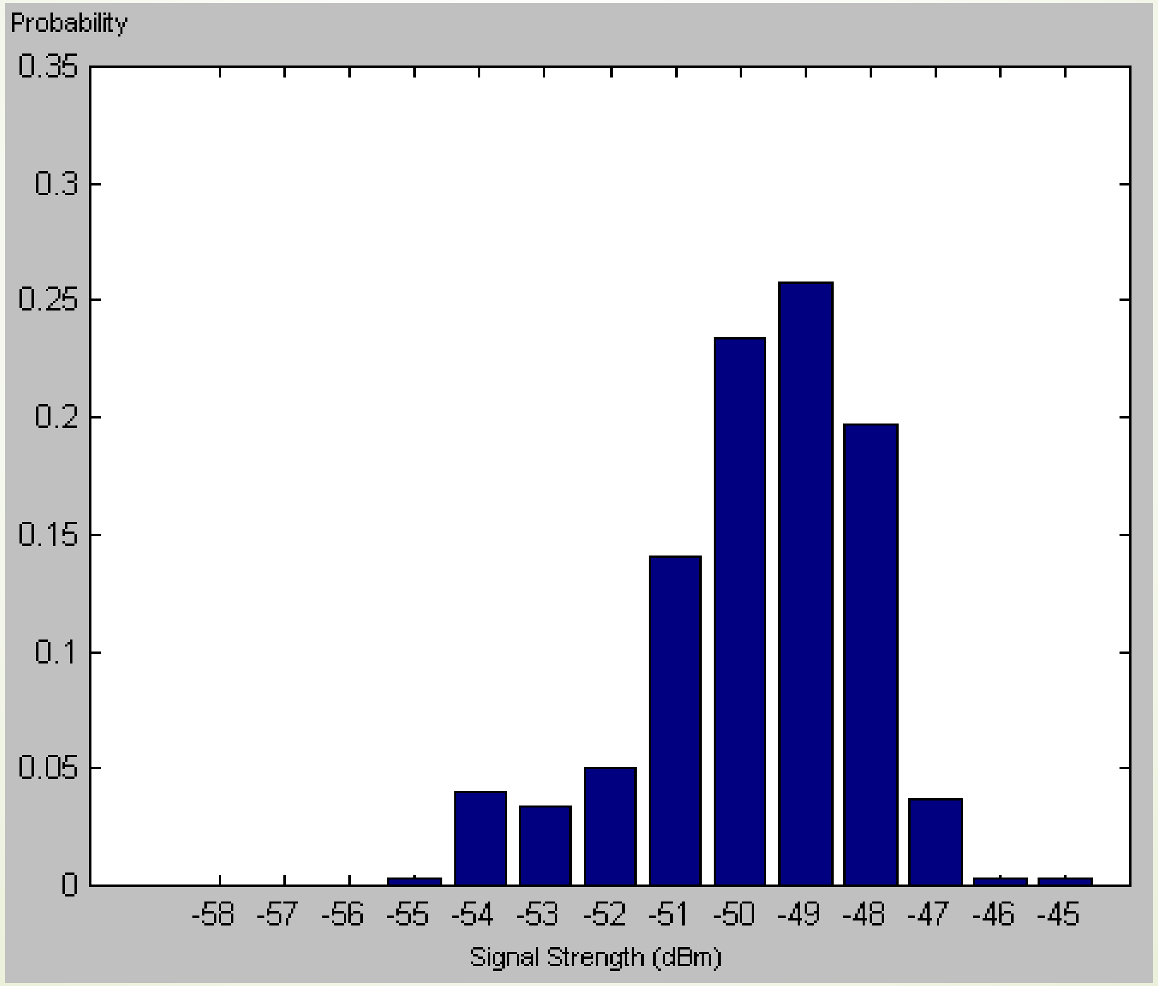
- Active scanning
  - Send a probe request
  - Receive a probe response



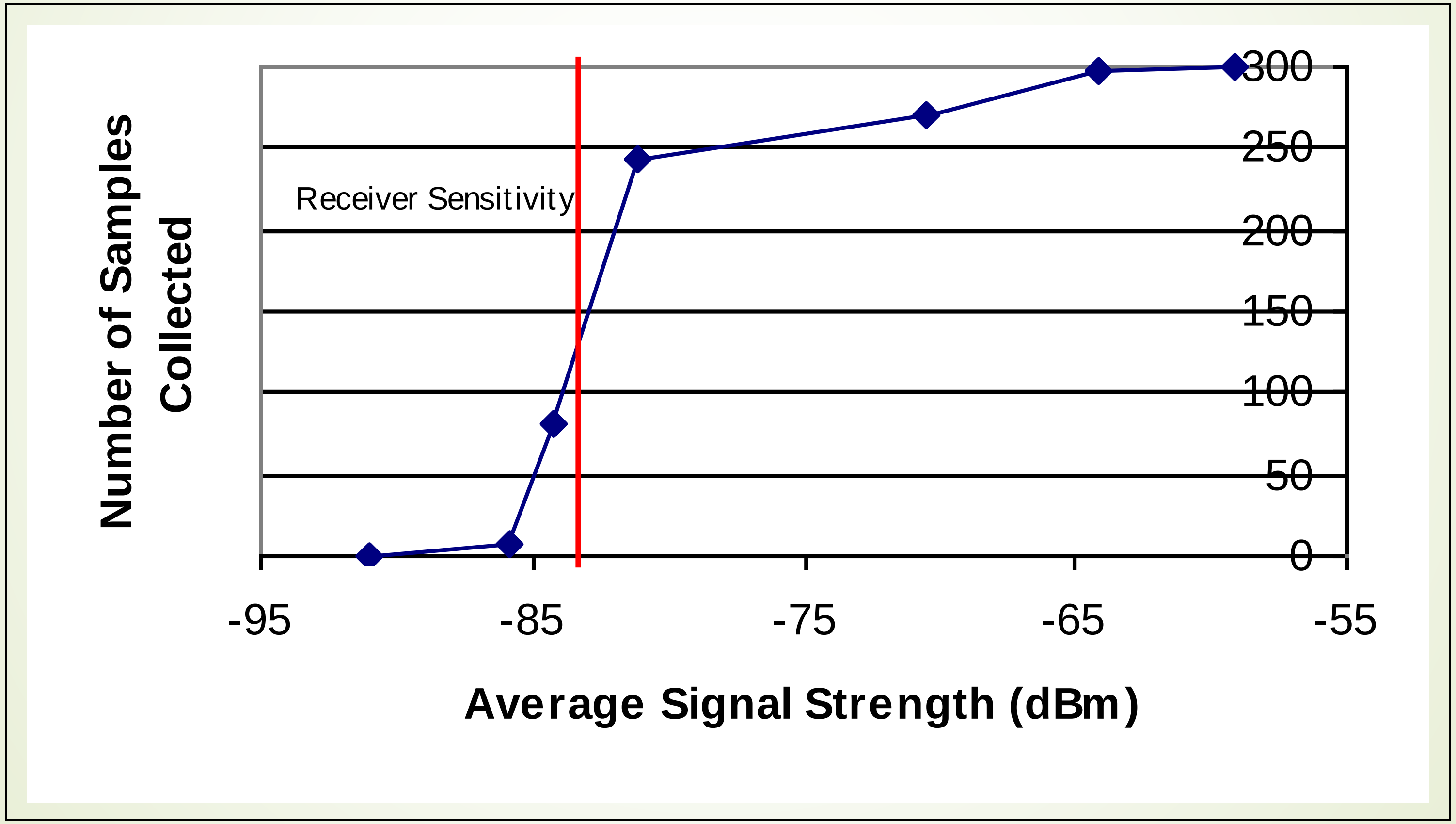
Signal Strength Characteristics

- Temporal variations
  - One access point
  - Multiple access points
- Spatial variations
  - Large scale
  - Small scale

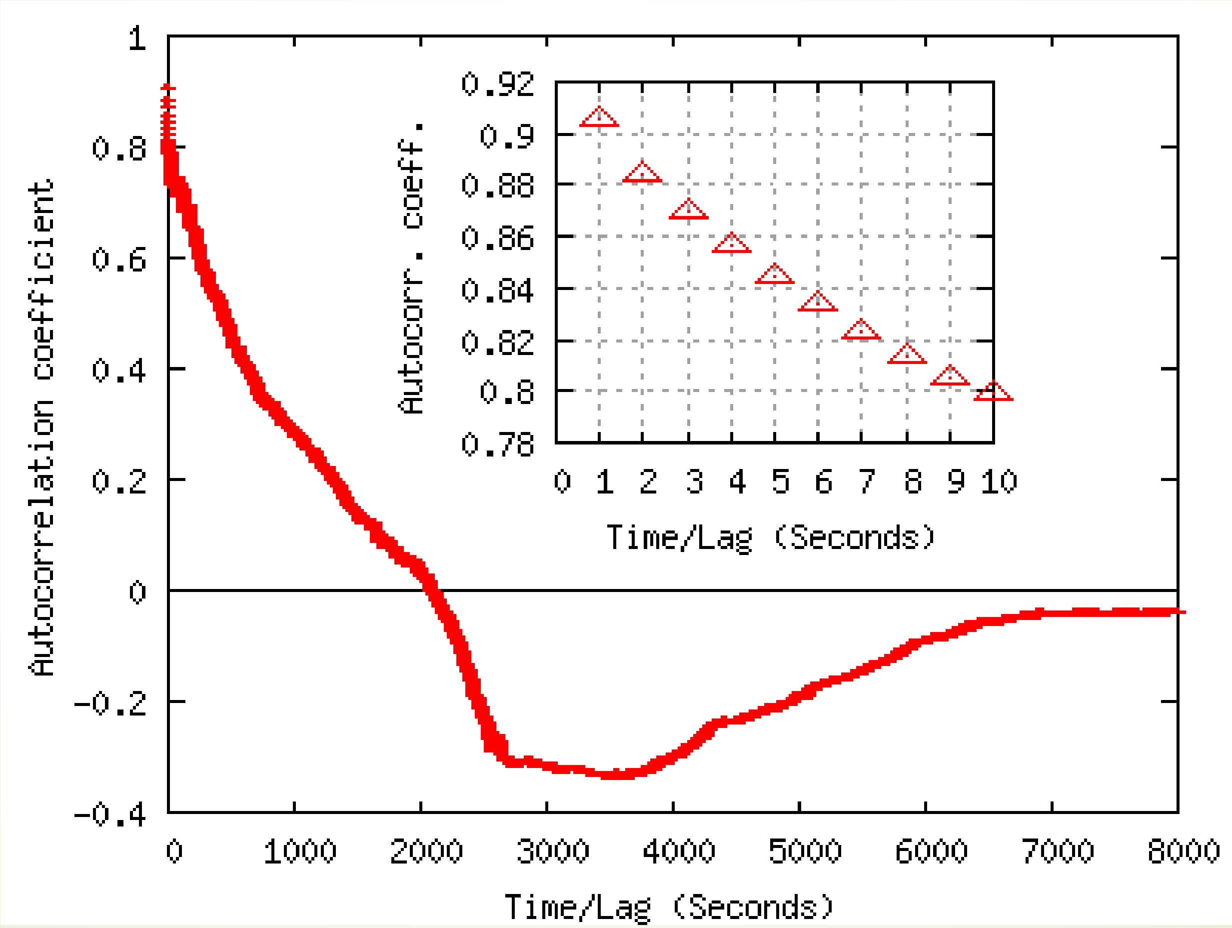
Temporal Variations



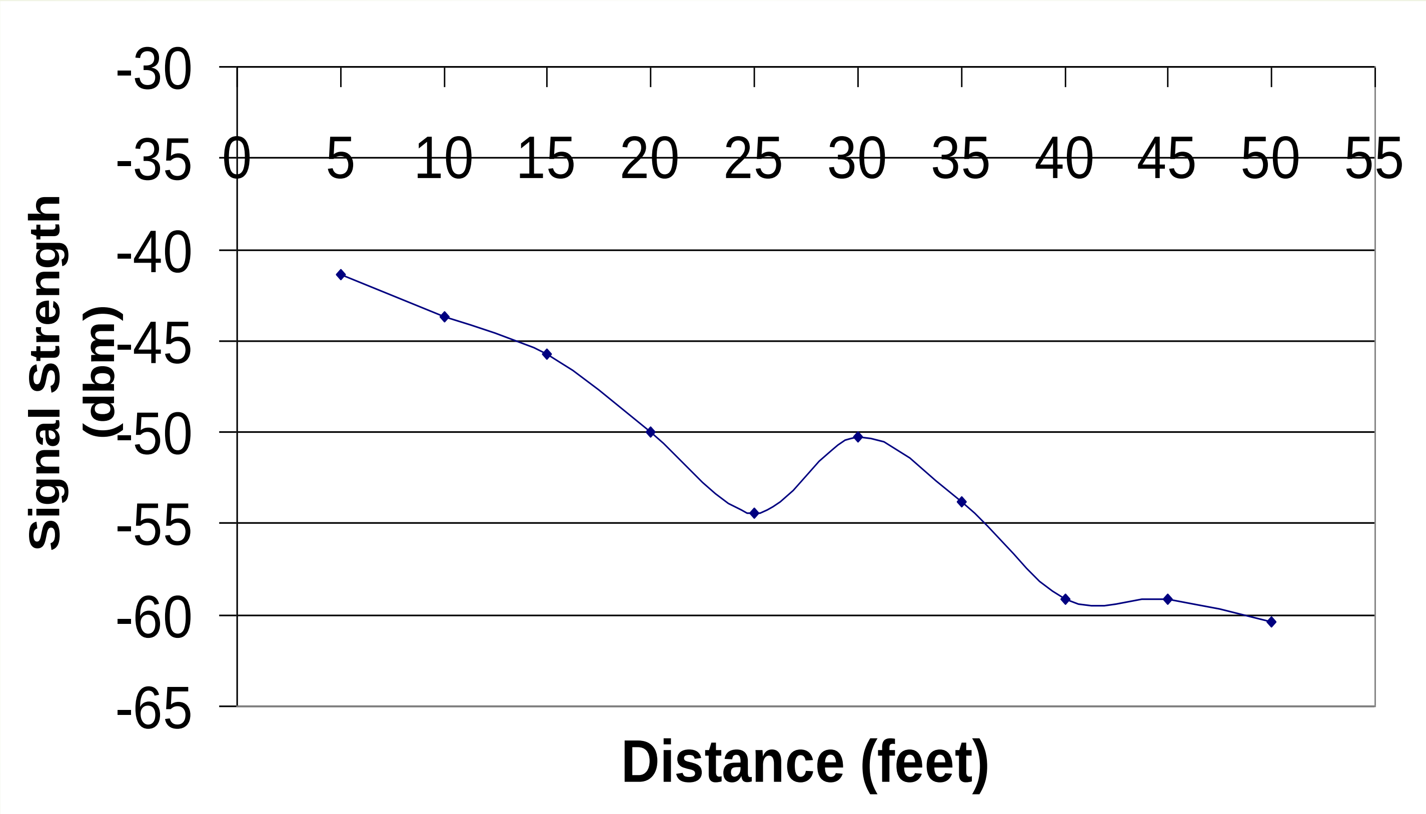
Temporal Variations



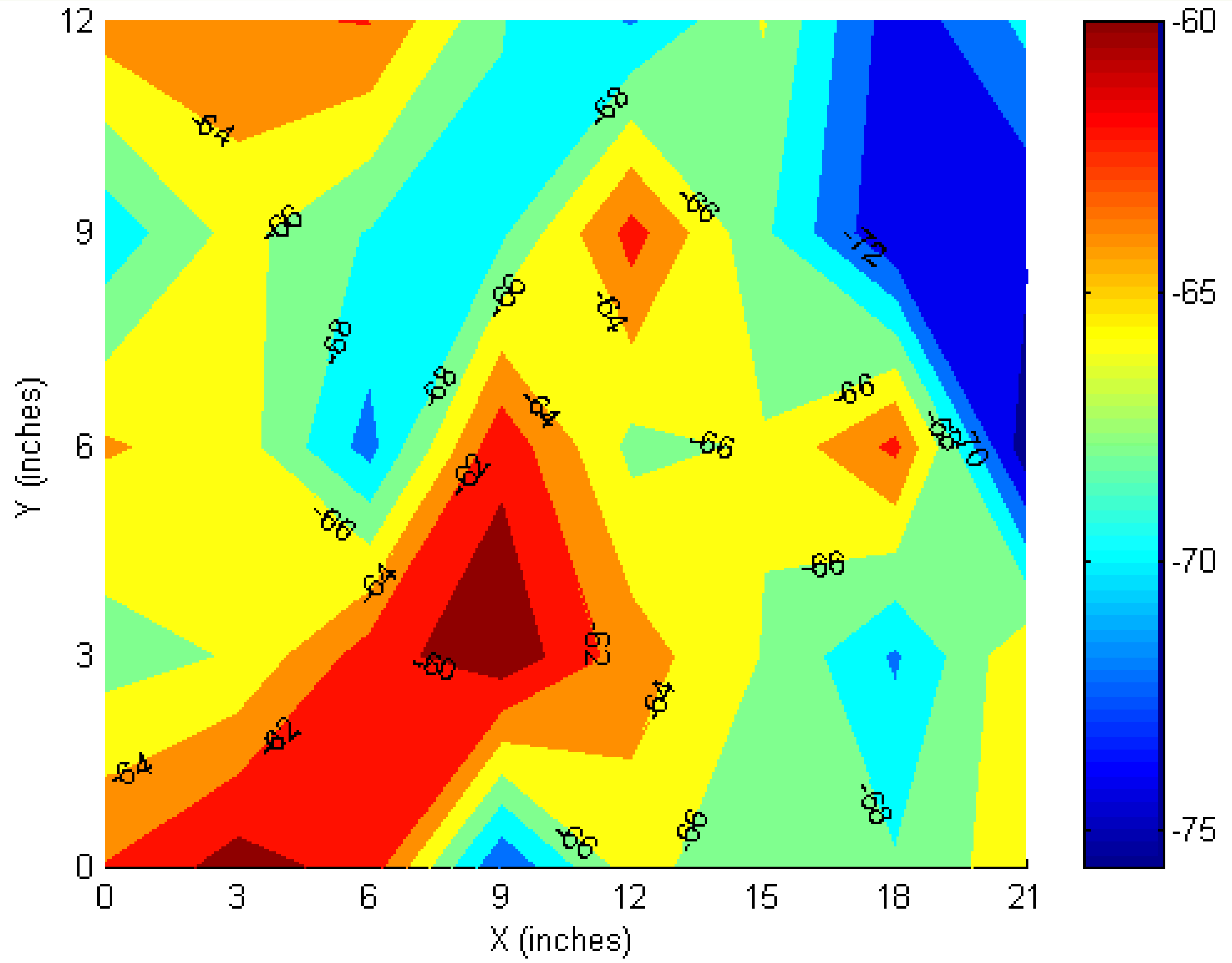
Temporal Variations: Correlation



Spatial Variations: Large-Scale



Spatial Variations: Small-Scale



Testbeds

- A.V. William's
  - 4th floor, AVW
  - 224 feet by 85.1 feet
  - UMD net (*Cisco* APs)
  - 21 APs (6 on avg.)
  - 172 locations
  - 5 feet apart
  - *Windows XP Prof.*

- **FLA**
  - 3rd floor, 8400 Baltimore Ave
  - 39 feet by 118 feet
  - LinkSys/Cisco APs
  - 6 APs (4 on avg.)
  - 110 locations
  - 7 feet apart
  - Linux (kernel 2.5.7)

Orinoco/Compaq cards

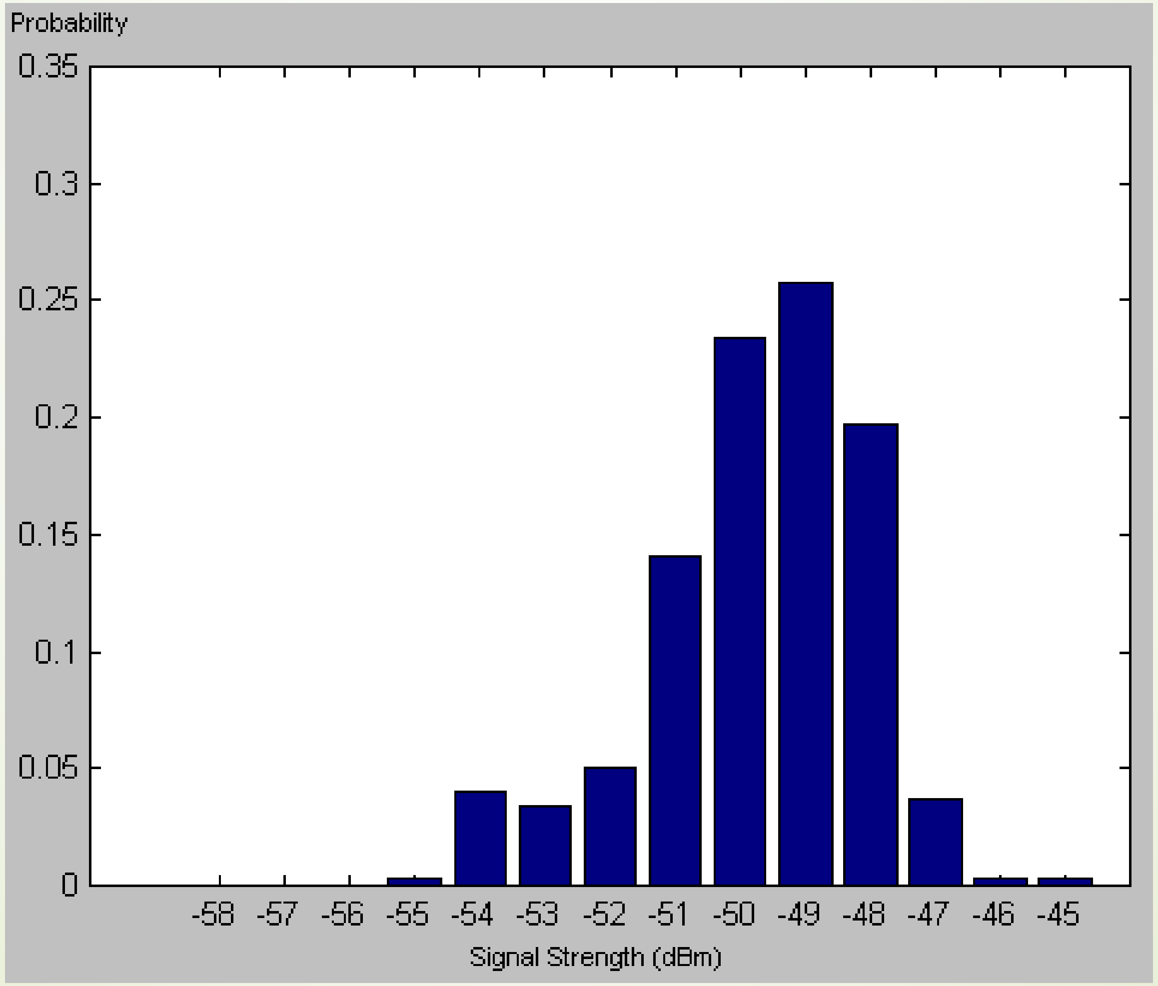
Horus Components

- Basic algorithm [Percom03]
- Correlation handler [InfoCom04]
- Continuous space estimator [Under]
- Locations clustering [Percom03]
- Small-scale compensator [WCNC03]

Basic Algorithm: Mathematical Formulation

- x: Position vector
- s: Signal strength vector
  - One entry for each access point
- s(x) is a stochastic process
- P[s(x), t]: probability of receiving s at x at time t
- s(x) is a stationary process
  - P[s(x)] is the histogram of signal strength at x

Basic Algorithm: Mathematical Formulation

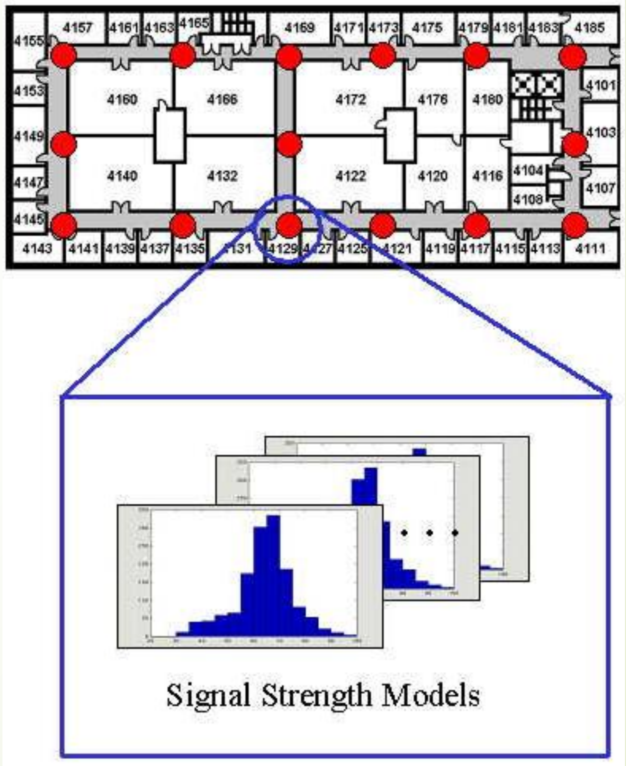


Basic Algorithm: Mathematical Formulation

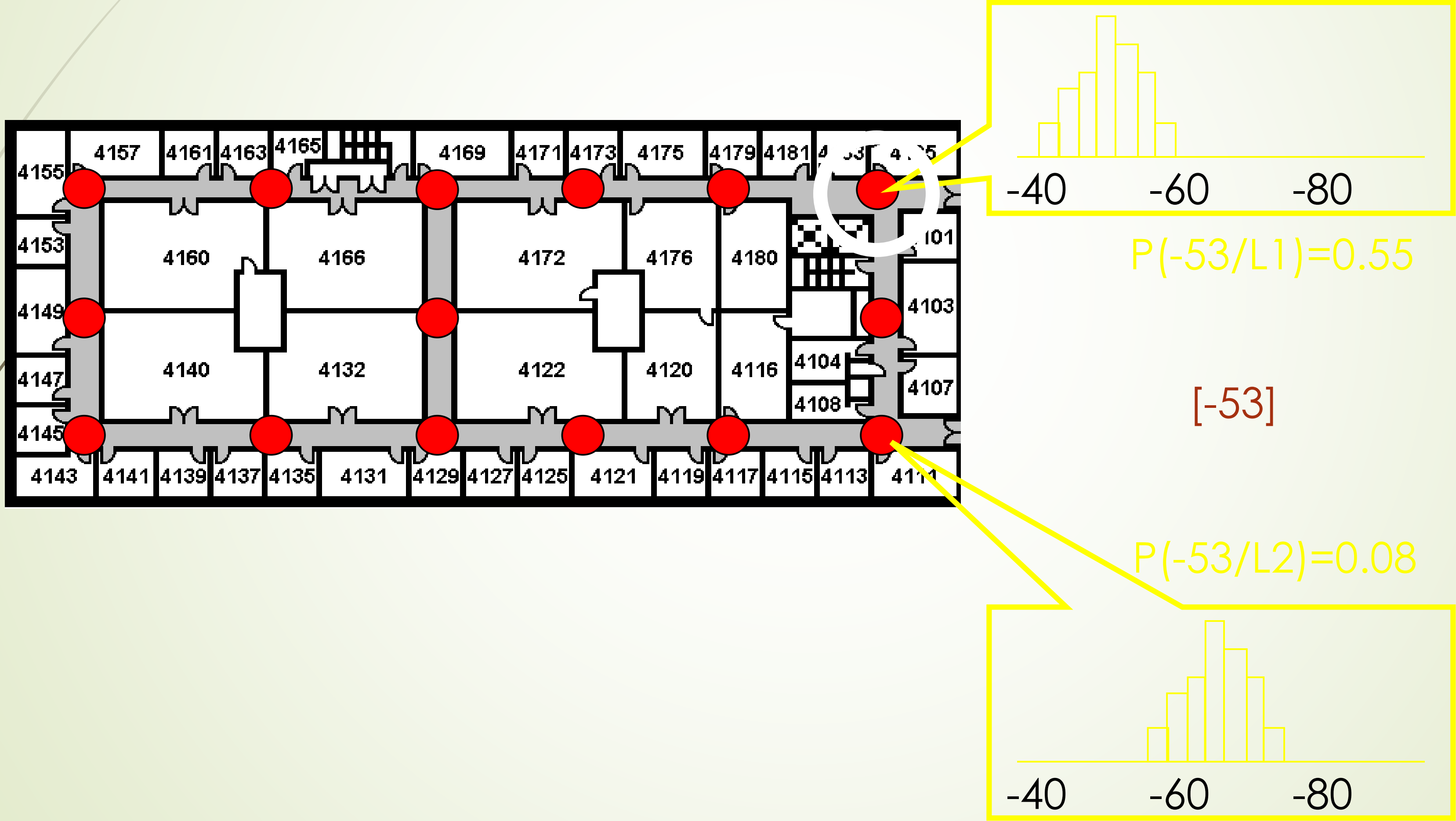
- $Argmax_x[P(x/s)]$
- Using Bayesian inversion
  - $Argmax_x[P(s/x).P(x)/P(s)]$
  - $Argmax_x[P(s/x).P(x)]$
- P(x): User history

Basic Algorithm

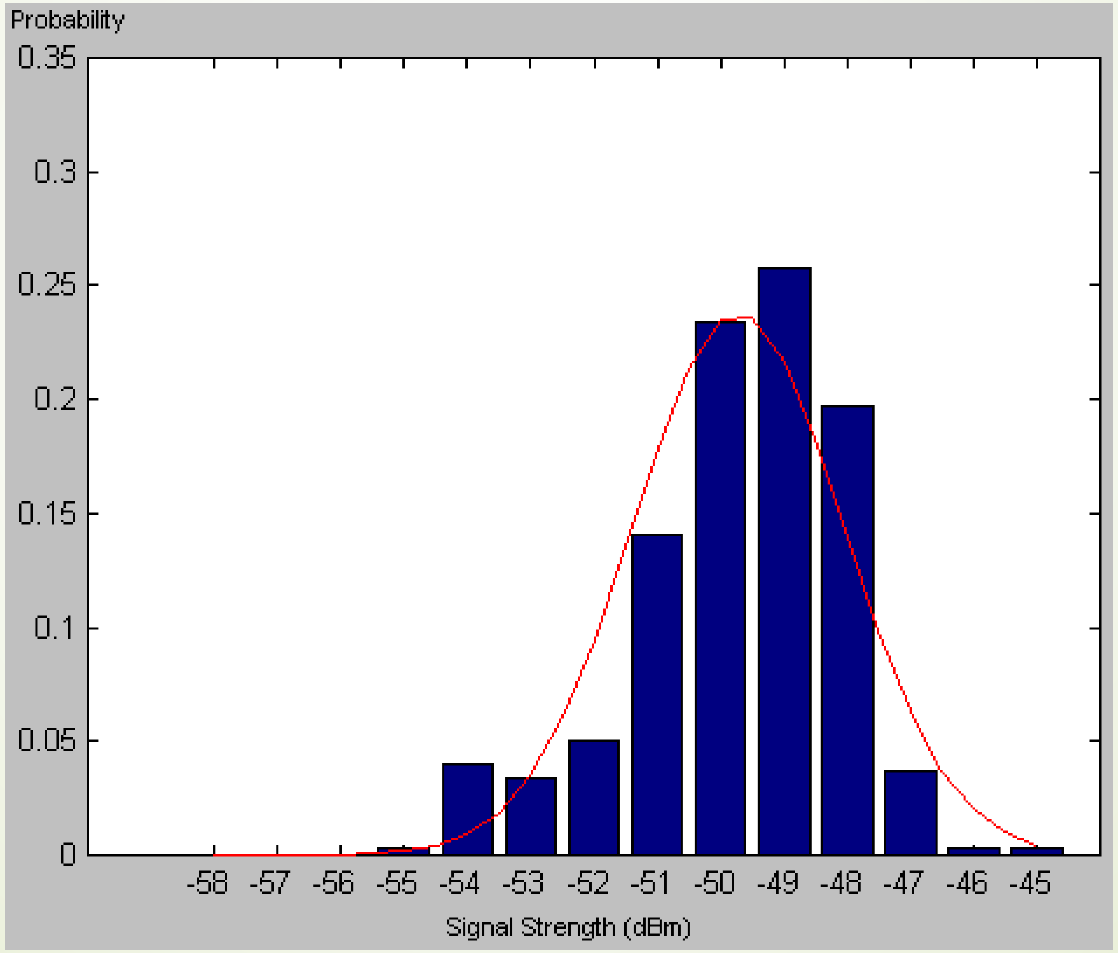
- Offline phase
  - Radio map: signal strength histograms
- Online phase
  - Bayesian based inference



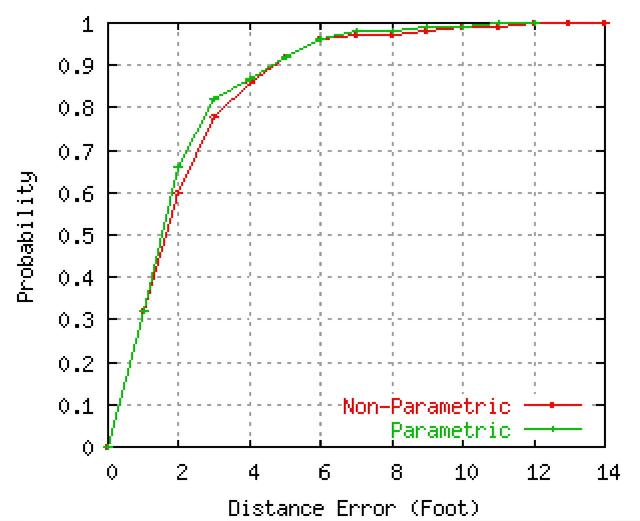
WLAN Location Determination (Cont'd)



Basic Algorithm: Signal Strength Distributions

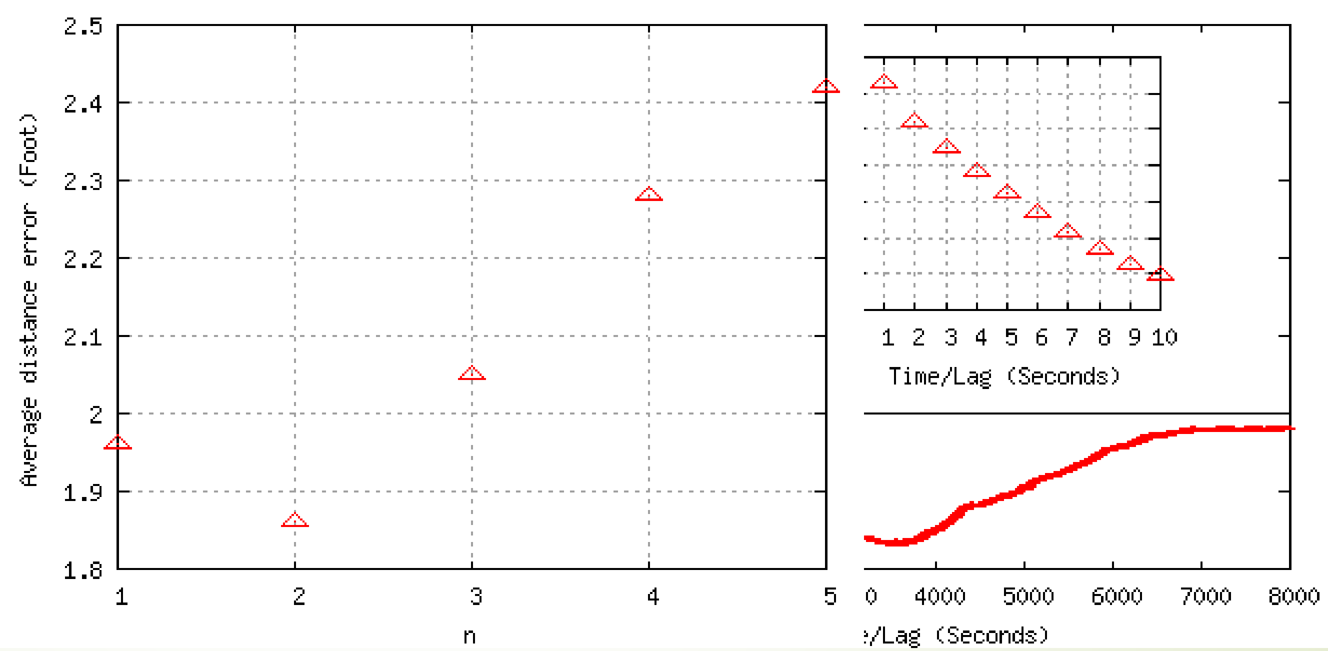


Basic Algorithm: Results



- Accuracy of 5 feet 90% of the time
- Slight advantage of parametric over non-parametric method
  – Smoothing of distribution shape

Correlation Handler



- Need to average multiple samples to increase accuracy
- Independence assumption is wrong

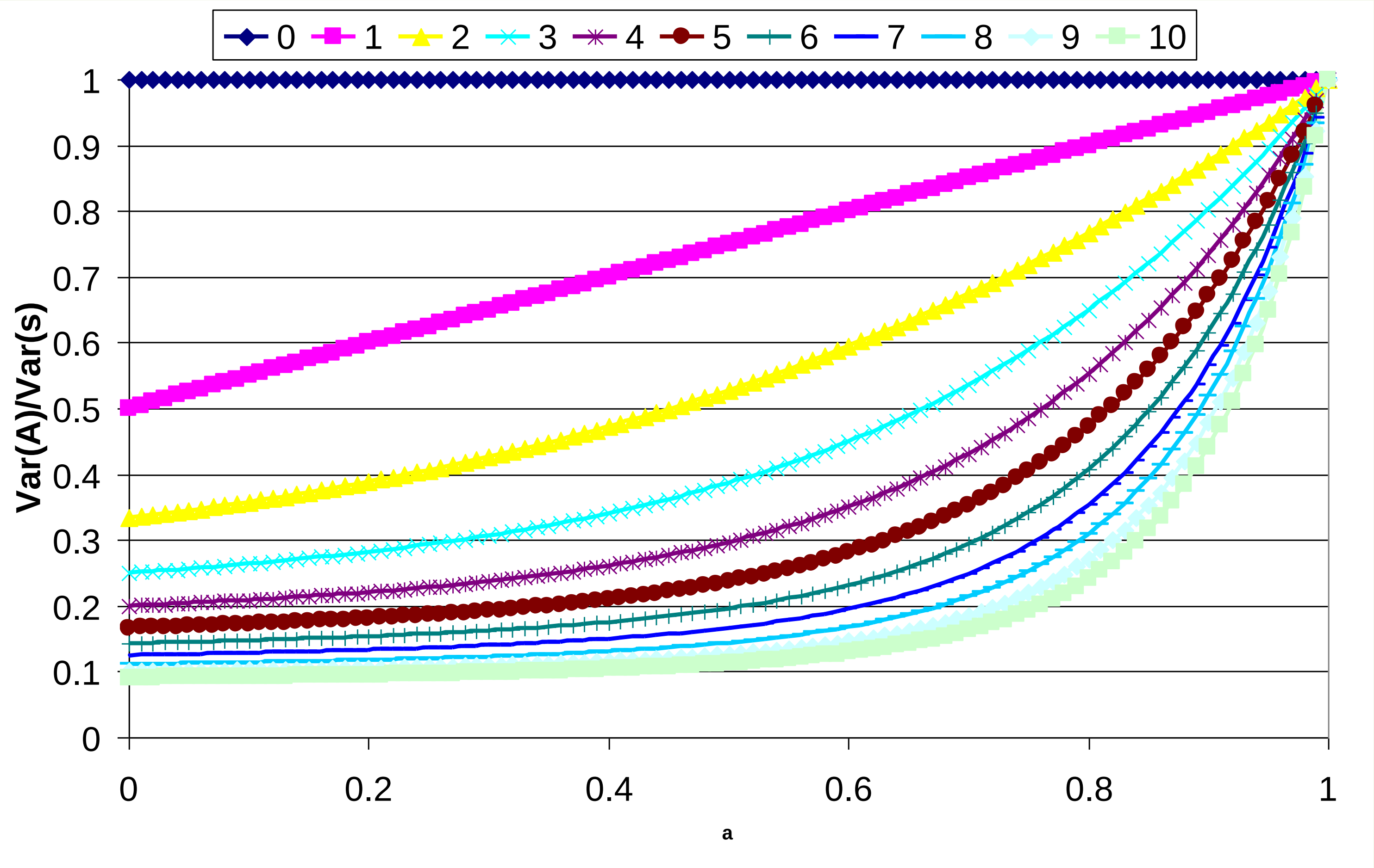
Correlation Handler: Autoregressive Model

- $s(t+1)=\alpha.s(t)+(1-\alpha).v(t)$
- α: correlation degree
- $E[v(t)]=E[s(t)]$
- $Var[v(t)]= (1+\alpha)/(1-\alpha)\ Var[s(t)]$

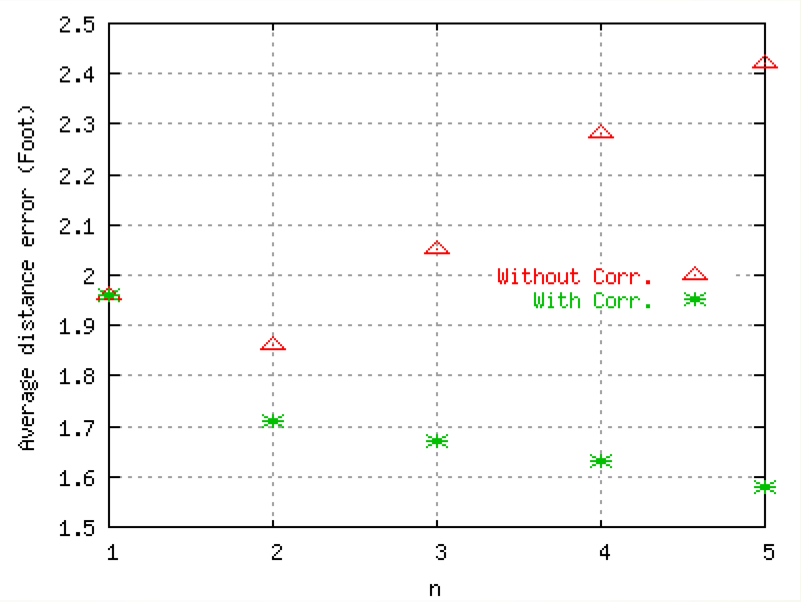
Correlation Handler: Averaging Process

- $s(t+1)= \alpha.s(t)+(1-\alpha).v(t)$
- $s \sim N(0, m)$
- $v \sim N(0, r)$
- $A=1/n\ (s_1+s_2+\ldots+s_n)$
- $E[A(t)]=E[s(t)]=0$
- $Var[A(t)]= m^2/n^2 \{ [(1-\alpha^n)/(1-\alpha)]^2 + n+ 1-\alpha^2 * (1-\alpha^{2(n-1)})/(1-\alpha^2) \}$

Correlation Handler: Averaging



Correlation Handler: Results



- Independence assumption: performance degrades as n increases
- Two factors affecting accuracy
  - Increasing n
  - Deviation from the actual distribution

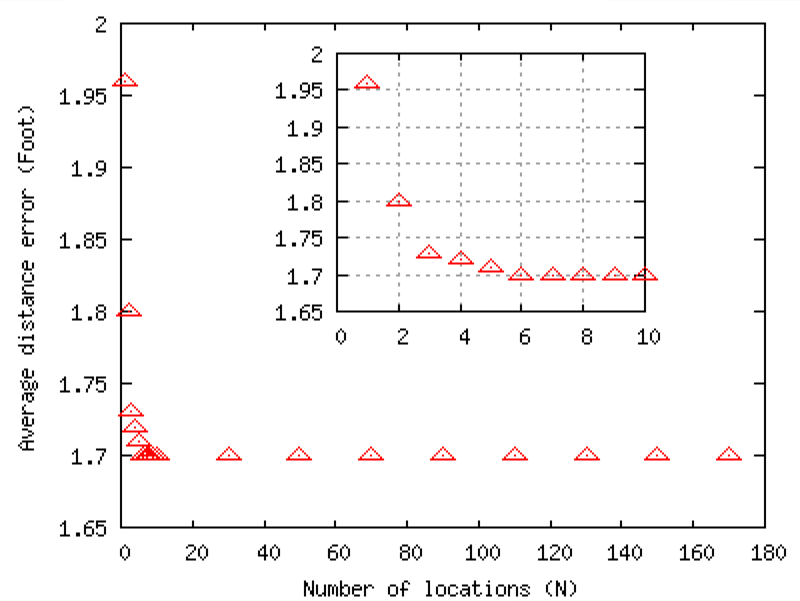
Continuous Space Estimator

- Enhance the discrete radio map space estimator
- Two techniques
  - Center of mass of the top ranked locations
  - Time averaging window

$$x \leftarrow \frac{\sum_{i=1}^{\min(N,\|\bar{\mathbb{X}}\|)} p(i) * \bar{\mathbb{X}}(i)}{\sum_{i=1}^{\min(N,\|\bar{\mathbb{X}}\|)} p(i)}$$

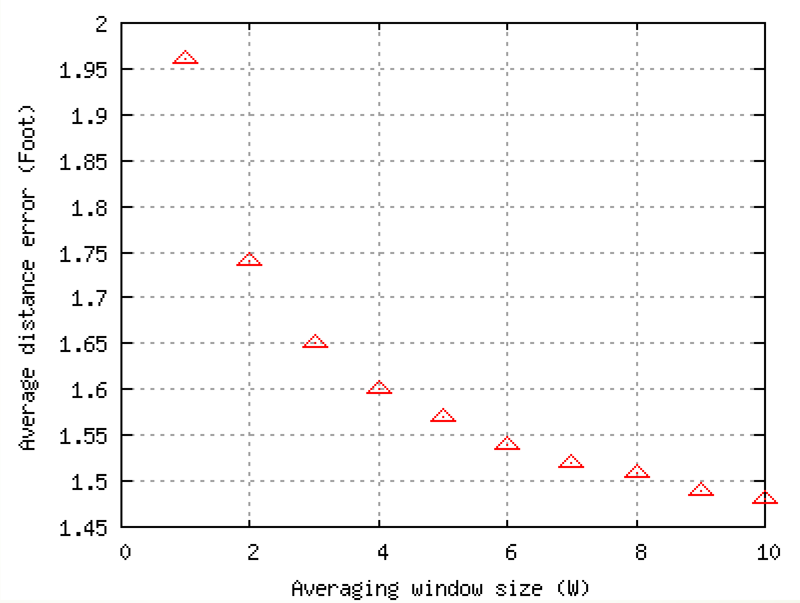
$$\bar{x_t} = \frac{1}{\min(W,t)} \cdot \sum_{t-\min(W,t)+1}^{t} x_i$$

Center of Mass:
Results



- N = 1 is the discrete-space estimator
- Accuracy enhanced by more than 13%

Time Averaging Window: Results

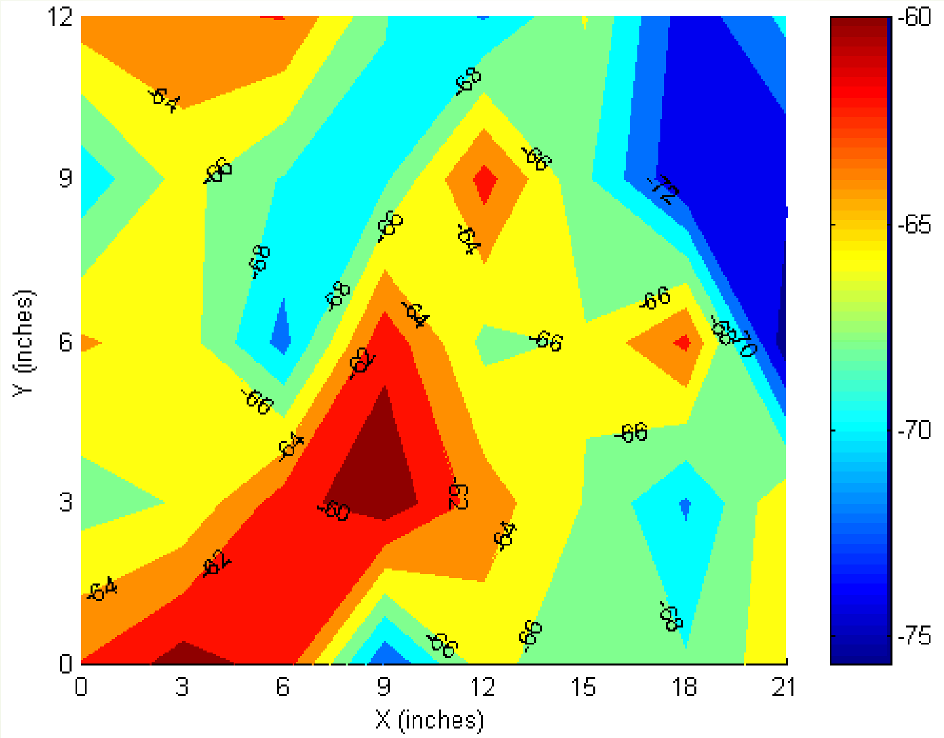


- N = 1 is the discrete-space estimator
- Accuracy enhanced by more than 24%

Horus Components

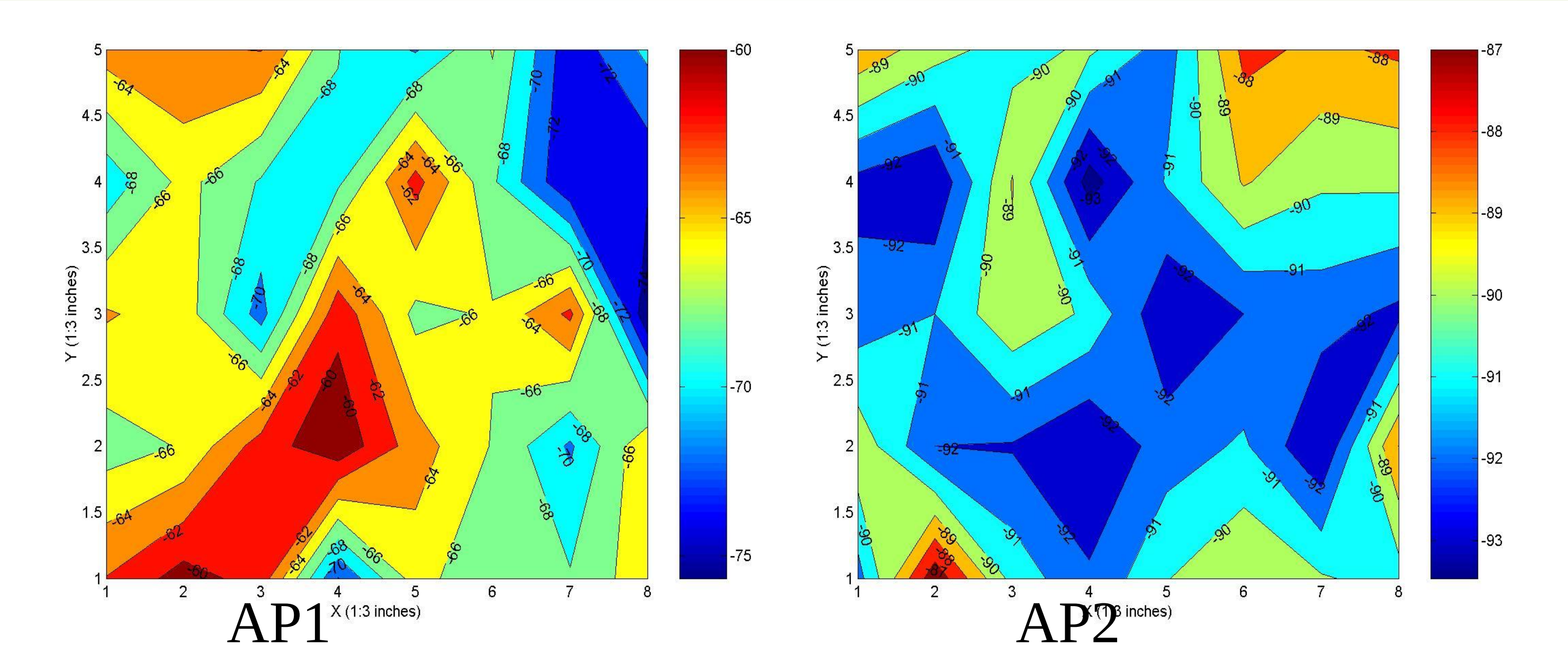
- Basic algorithm
- Correlation handler
- Continuous space estimator
- Small-scale compensator
- Locations clustering

Small-scale Compensator



- Multi-path effect
- Hard to capture by radio map (size/time)

Small-scale Compensator: Small-scale Variations

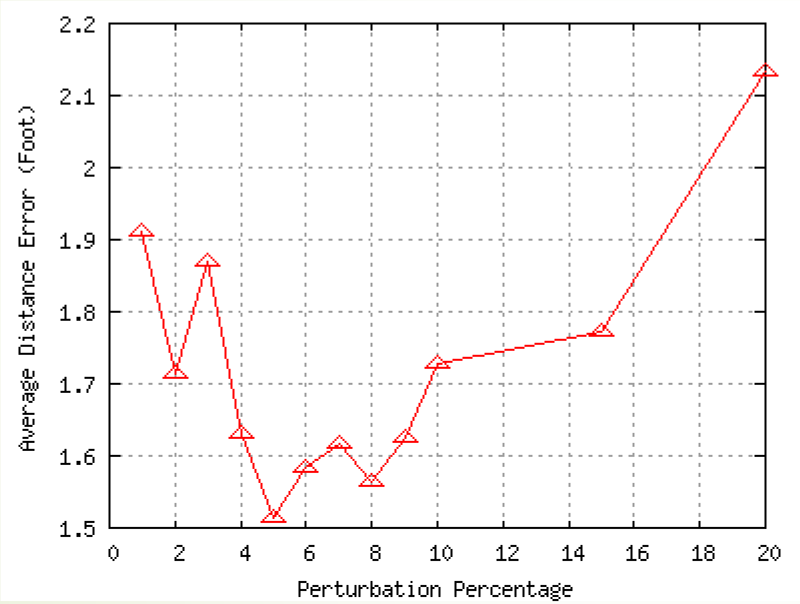


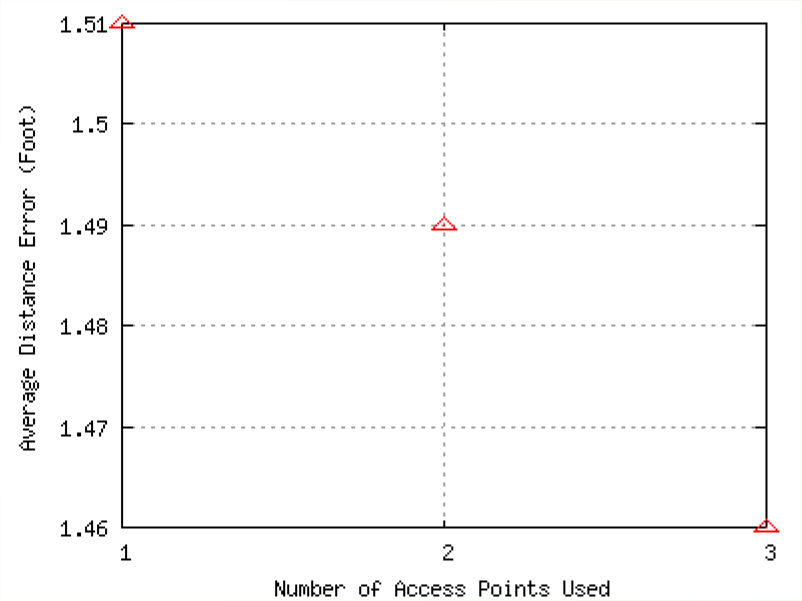
- Variations up to 10 dBm in 3 inches
- Variations proportional to average signal strength

Small-scale Compensator: Perturbation Technique

- Detect small-scale variations
  - Using previous user location
- Perturb signal strength vector
  - $(s_1, s_2, \ldots, s_n) \rightarrow (s_1 \pm d_1, s_2 \pm d_2, \ldots, s_n \pm d_n)$
  - Typically, n=3-4
- d_i is chosen relative to the received signal strength

Small-scale Compensator: Results





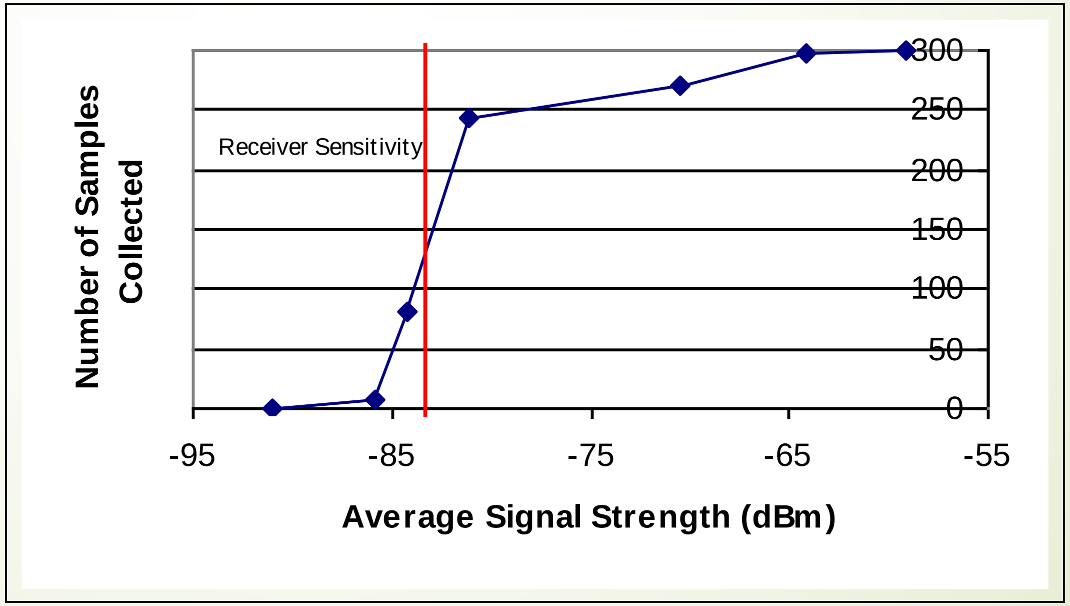
- Perturbation technique is not sensitive to the number of APs perturbed
- Better by more than 25%

Horus Components

- Basic algorithm
- Correlation handler
- Continuous space estimator
- Small-scale compensator
- Locations clustering

Locations Clustering

- Reduce computational requirements
- Two techniques
  - Explicit
  - Implicit



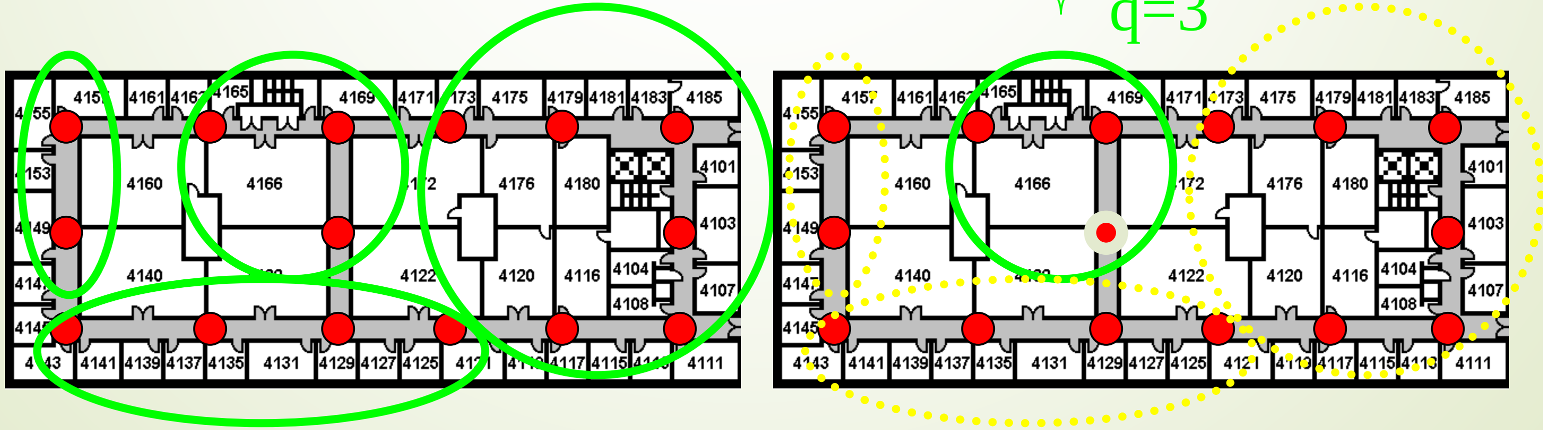
Locations Clustering: Explicit Clustering

- Use access points that cover each location
- Use the *q* strongest access points

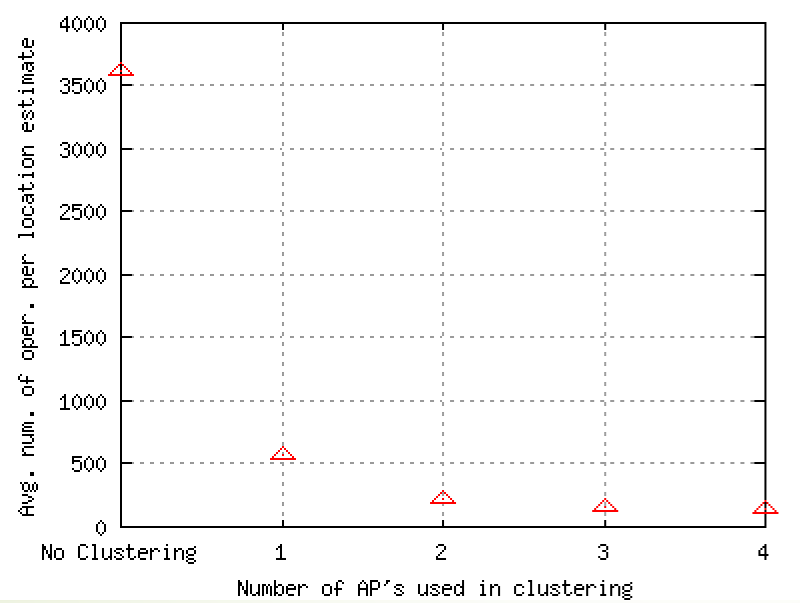
S=[-60, -45, -80, -86, -70]

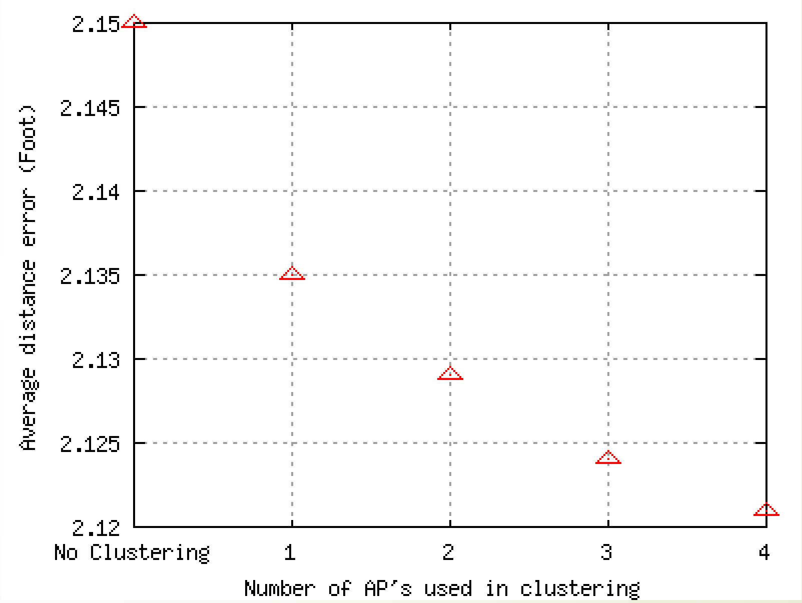
S=[-45, -60, -70, -80, -86]

q=3



Locations Clustering:
Results- Explicit Clustering





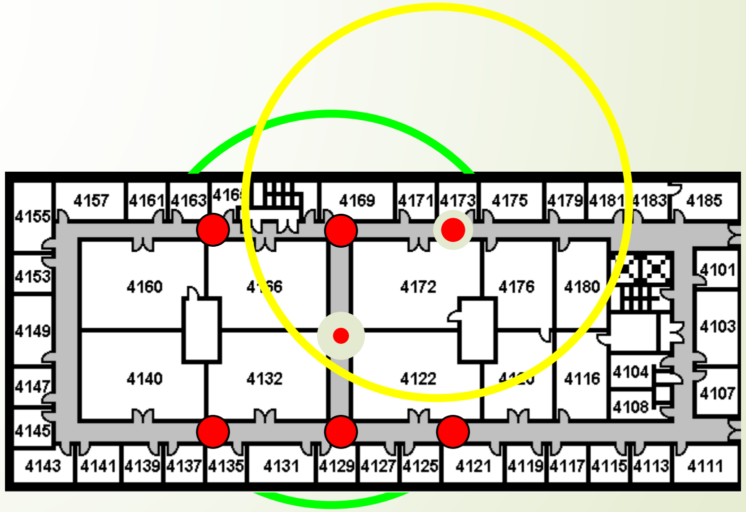
- An order of magnitude enhancement in avg. num. of oper. /location estimate
- As q increases, accuracy slightly increases

Locations Clustering: Implicit Clustering

- Use the access points incrementally
- Implicit multi-level clustering

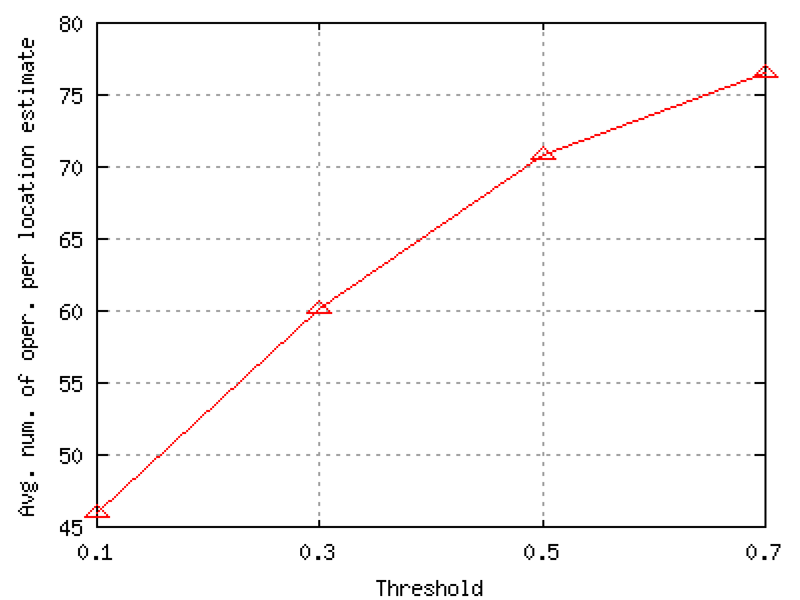
S=[-60, -45, -80, -86, -70]

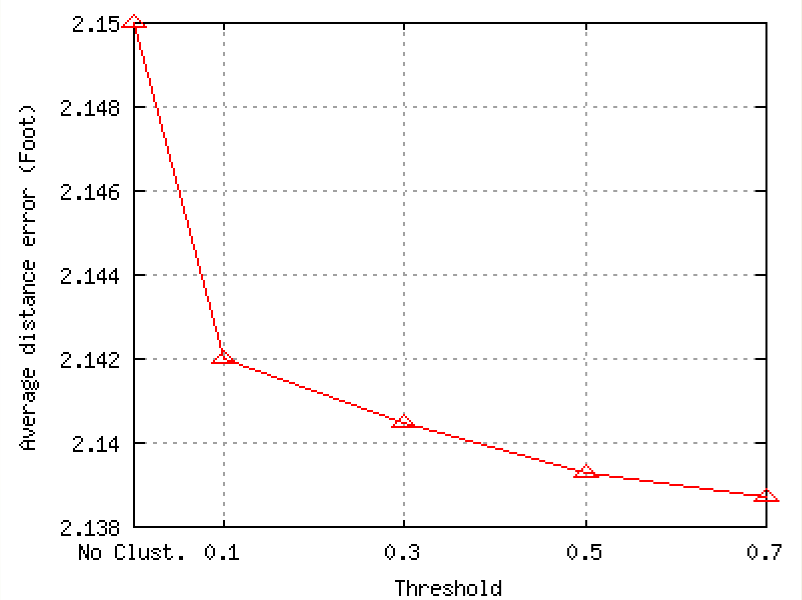
S=[-45, -60, -70, -80, -86]



Locations Clustering:

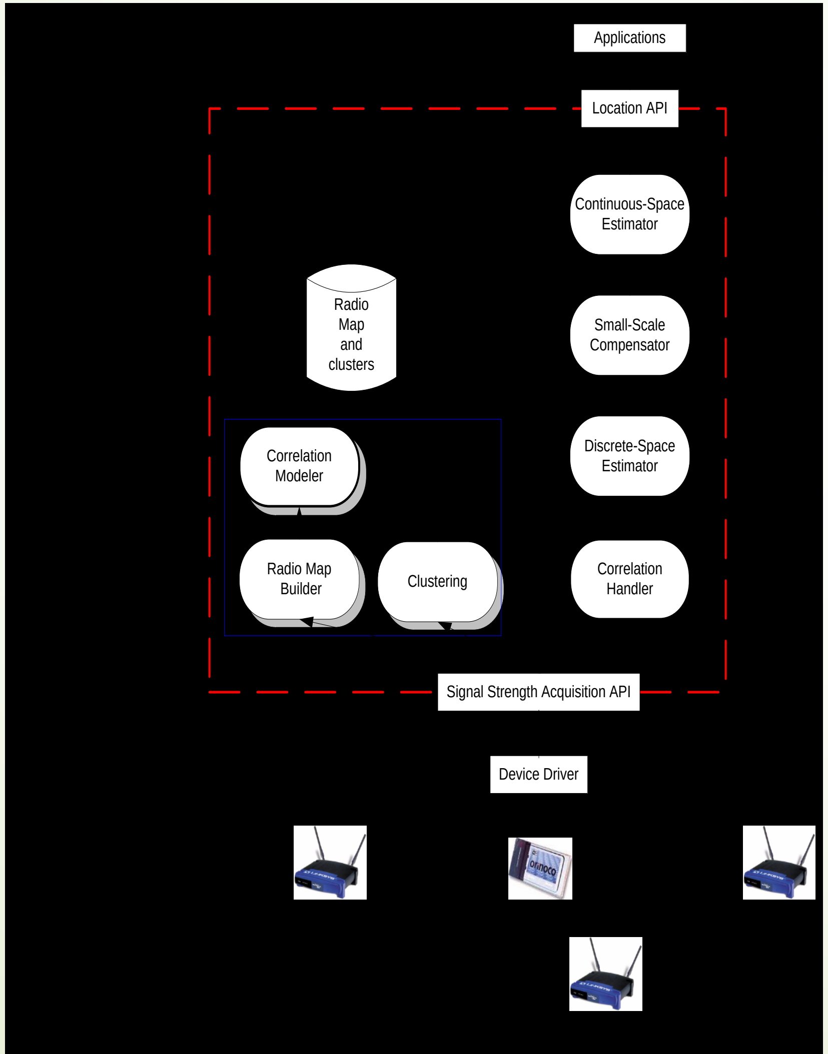
Results- Implicit Clustering



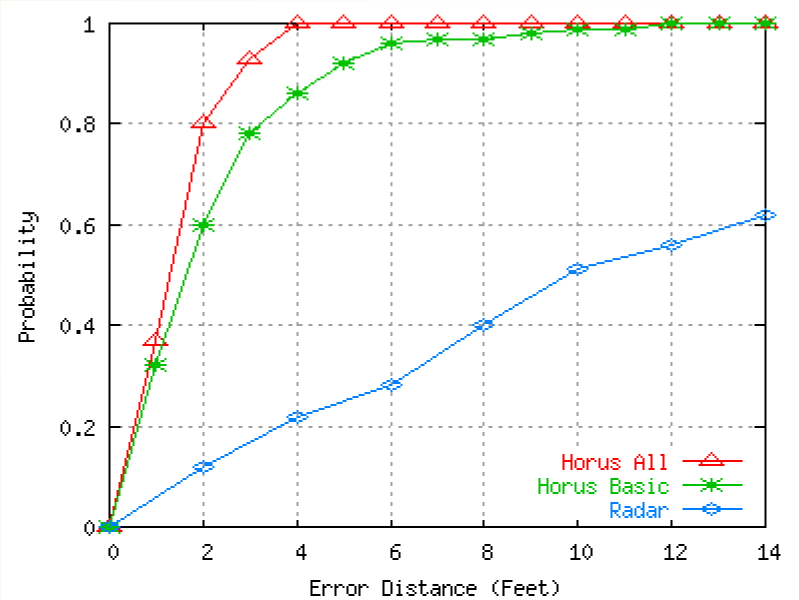


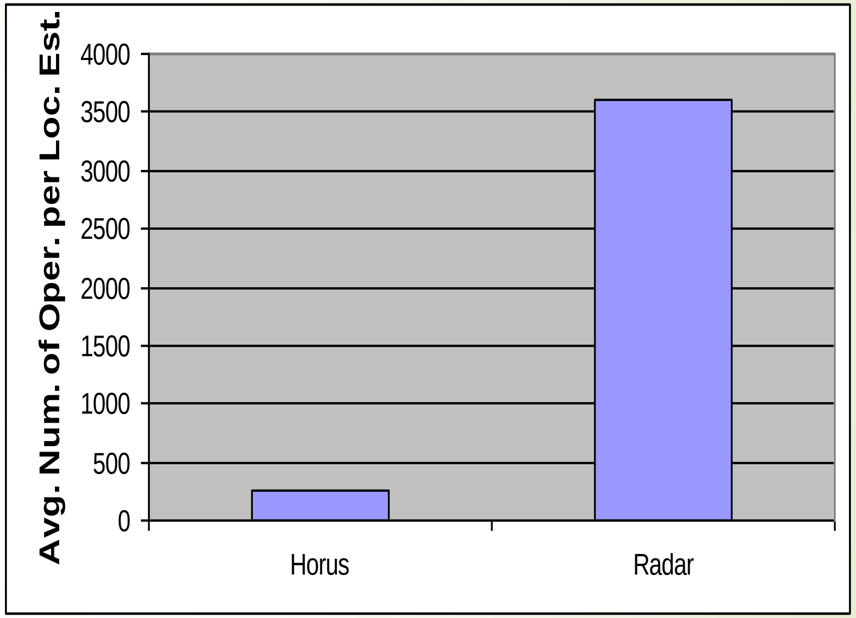
- Avg. num. of oper. /location estimate better than explicit clustering
- Accuracy increases with Threshold

Horus Components



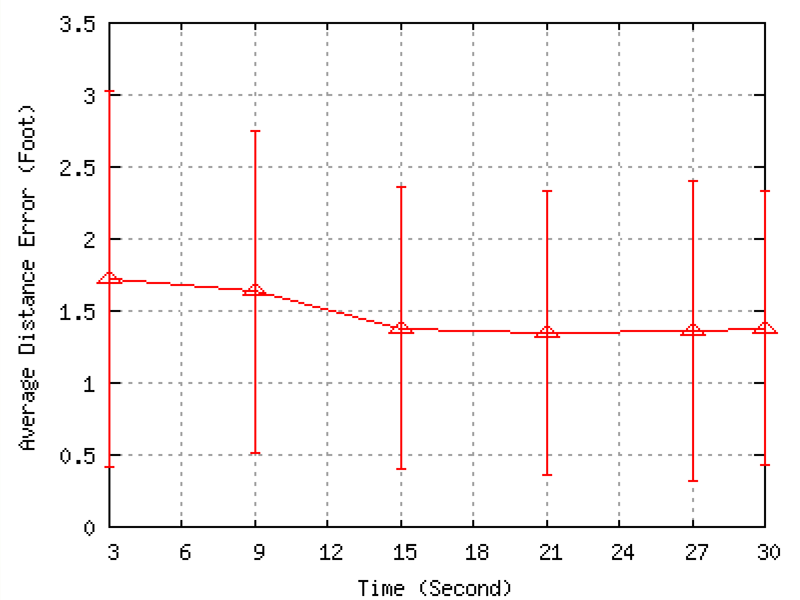
Horus-Radar Comparison





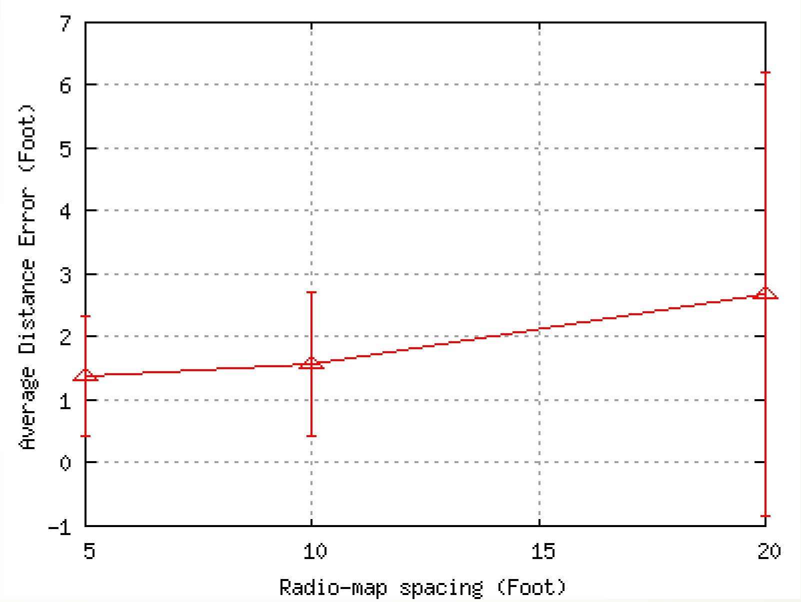
	Median	Avg	Stdev	Max
Horus (all components)	1.28	1.38	0.95	4
Horus (basic)	1.6	2.16	2.09	18.08
Radar	9.74	13.15	10.71	57.67

Training Time



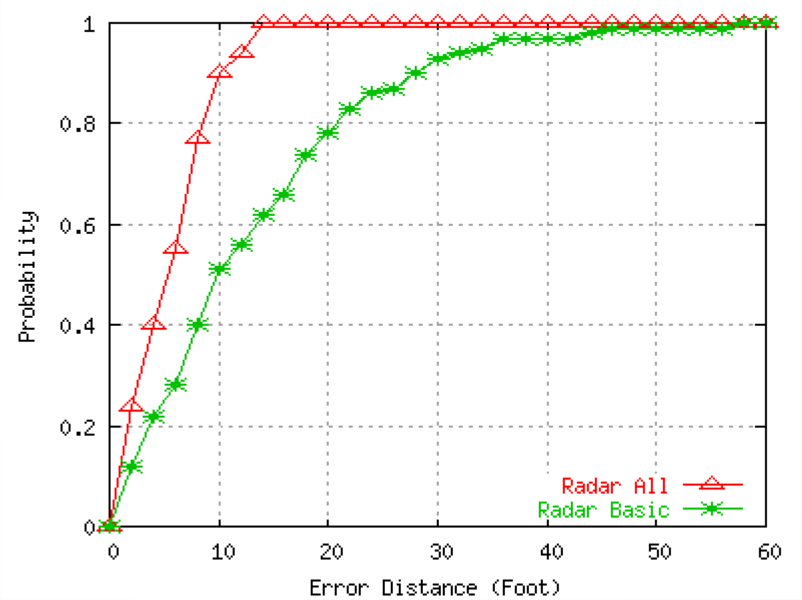
- 15 seconds training time per location

Radio map Spacing



- Average distance error increase by as much as 100% (20 feet)
- 14 feet gives good accuracy

Radar with Horus Techniques



- Average distance error enhanced by more than 58%
- Worst case error decreased by more than 76%

Conclusions

- The *Horus* system achieves its goals
- High accuracy
  - Through a probabilistic location determination technique
  - Smoothing signal strength distributions by Gaussian approximation
  - Using a continuous-space estimator
  - Handling the high correlation between samples from the same access point
  - The perturbation technique to handle small-scale variations
- Low computational requirements
  - Through the use of clustering techniques

Conclusions (Cont’d)

- Scalability in terms of the coverage area
  - Through the use of clustering techniques
- Scalability in terms of the number of users
  - Through the distributed implementation
- Training time of 15 seconds per location is enough to construct the radio-map
- Radio map spacing of 14 feet
- Horus vs. Radar
  - More accurate by more than 11 feet, on the average
  - More than an order of magnitude savings in number of operations required per location estimate
- Horus vs. Ekahau

Conclusions (Cont'd)

- Modules can be applied to other WLAN location determination systems
  - Correlation handling, continuous-space estimator, clustering, and small-scale compensator
- Applied to Radar
  - Average distance error enhanced by more than 58%
  - Worst case error decreased by more than 76%
- Techniques presented thesis are applicable to other RF-technologies
  - 802.11a, 802.11g, HiperLAN, and BlueTooth, …

Locus

- Indoor location anywhere on College Park Campus
- Based on Wi-Fi RSSI
- ~ 4500 Access Points
- Floor accuracy >95%
- Location Accurate to the room
- Being integrated with M-Urgency

Flying Turtle

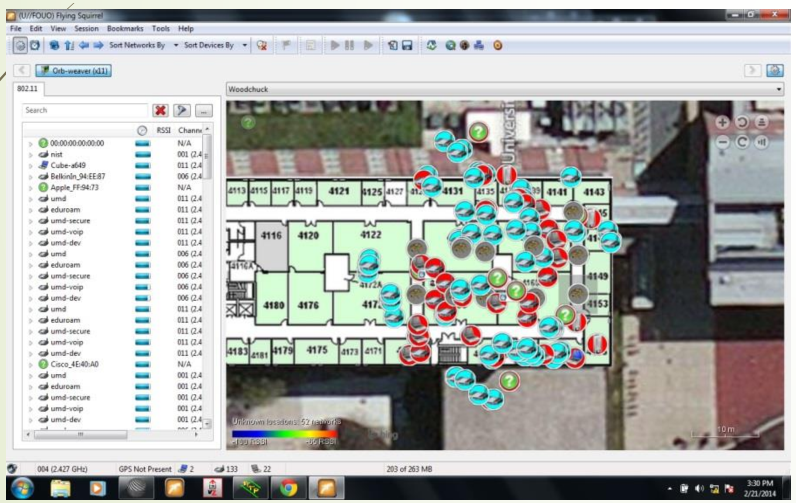
Locating indoors

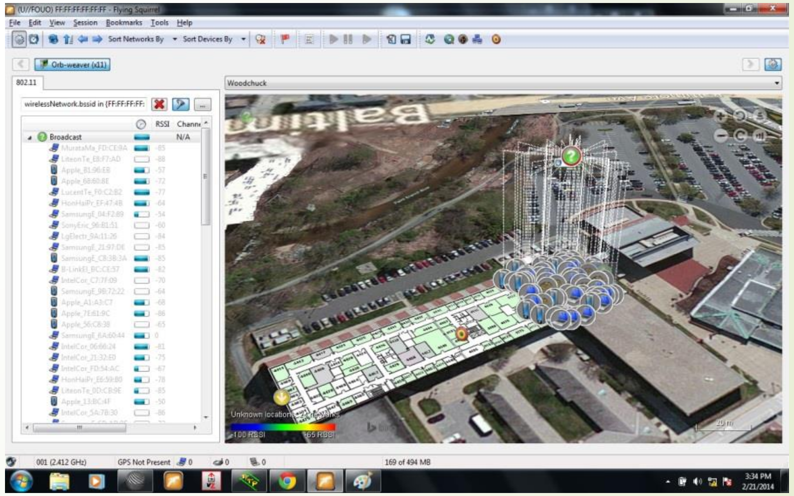
Flying Squirrel – NRL Project

- Goal
  - Real-time discovery, analysis, and mapping of IEEE 802.11a/b/g/n wireless networks
  - Use passive listeners
  - Extensive analytics

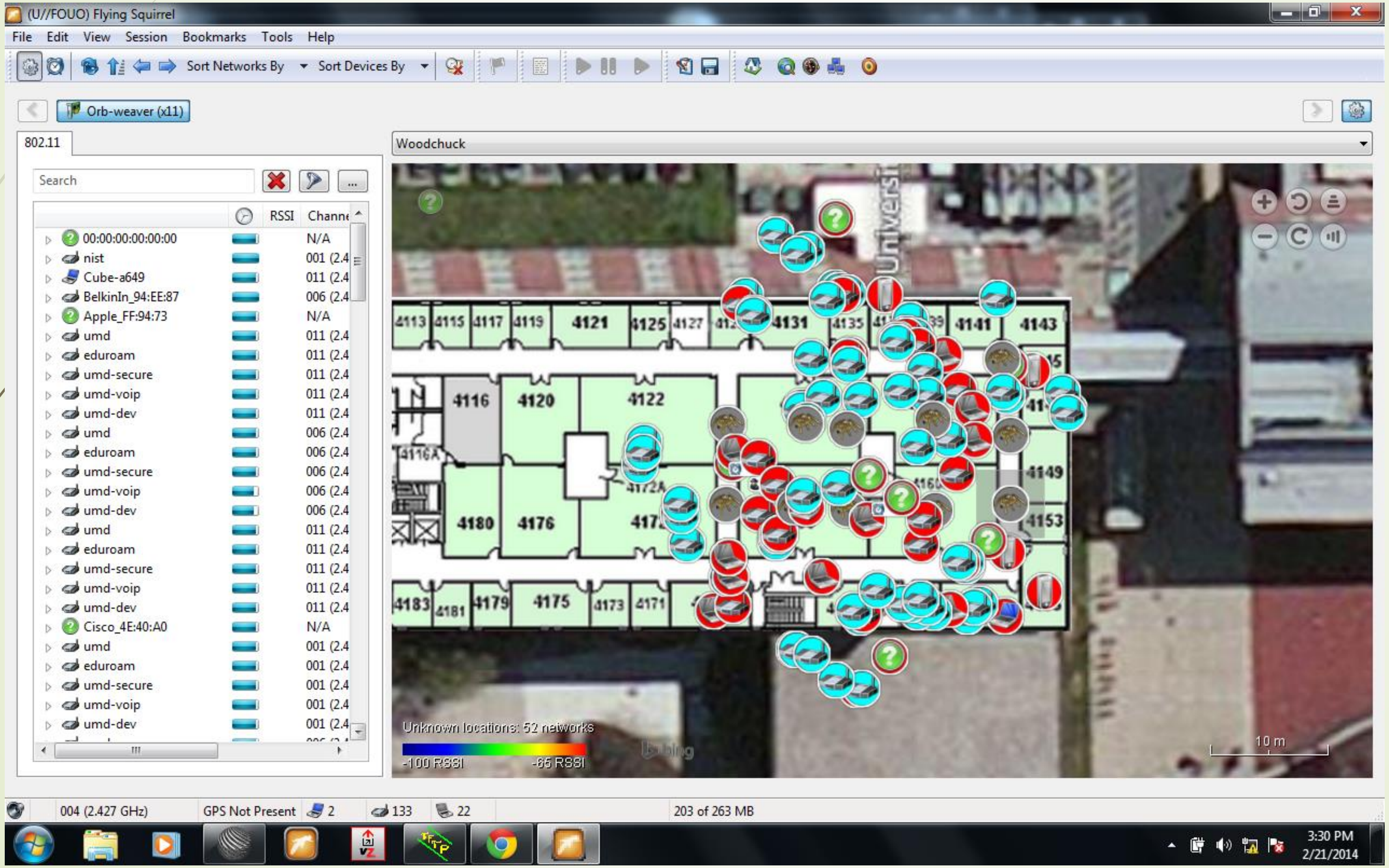
Flying Turtle

- 20 sensors on 4100 wing of AVW
  - compose approx. 20 ft, 20 ft grid points.





Initial Observations



Our Approach

- Dynamic Fingerprinting/Radio Map
- With passive listeners
  - Can we provide accurate localization from measured signal strengths ?

Time Based Location

Ashok K. Agrawala
Director, MIND Lab
Professor, Computer Science
University of Maryland

Topics

- Location Determination
  - Horus and Locus
  - PinPoint
- Clock Synchronization
  - With Absolute Real Time

GeoLocation

- RSSI Based – Horus and Locus
- Accurate Time Stamping
- GeoLocation with accuracy in inches
- Clock Synchronization
  - Mapping Function Timing Protocol
  - Synchronization with Absolute Time
- Flying Turtle - testbed

PinPoint Technology - Basis

- Use a clock model
- Determine node to node distance by measuring time of flight of the signal

Clock Model

- The clock at a node is assumed to have drift stable over short periods.
- Hence clock time τ is related to the real time t by

$$\tau_a(t) = \beta_a(\alpha_a + t)$$

- where
  - α and β remain constant for the measurement phase.
  - β, the drift rate of the clock is no worse than 100 parts per million
  - τ is measured with a nanosecond resolution

Measurements for node pair A and B

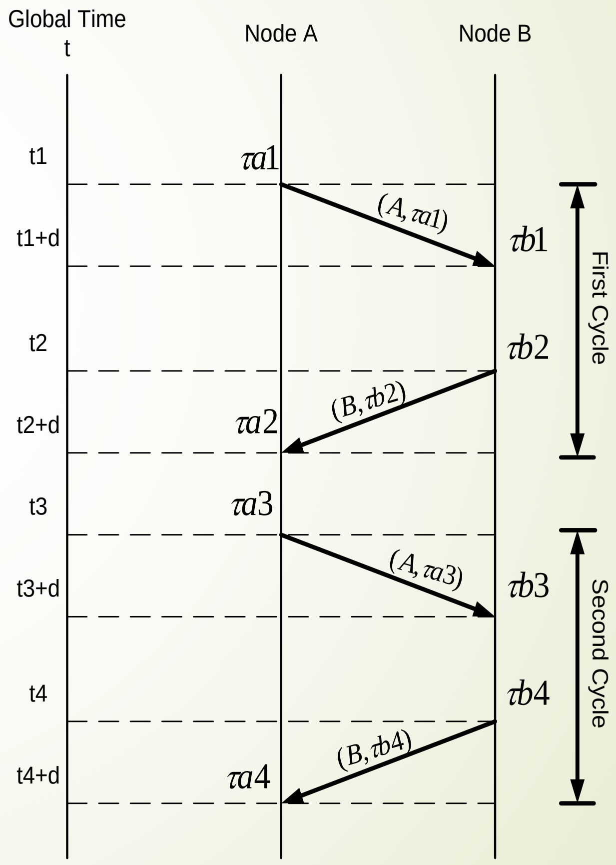
- Let

  $\tau a1$, $\tau b1$: tx and rx ts of first A msg

  $\tau b2$, $\tau a2$: tx and rx ts of first B msg

  $\tau a3$, $\tau b3$: tx and rx ts of second A msg

  $\tau a4$, $\tau b4$: tx and rx ts of second B msg



Calculations for node pair A and B

- Drift ratio

$$\frac{\tau_{a3}-\tau_{a1}}{\tau_{b3}-\tau_{b1}} = \frac{\beta_a\left(\alpha_a+t_3\right)-\beta_a\left(\alpha_a+t_1\right)}{\beta_b\left(\alpha_b+t_3+d\right)-\beta_b\left(\alpha_b+t_1+d\right)} = \frac{\beta_a}{\beta_b}$$

- Propagation delay

$$\beta_b d = \frac{(\tau_{b1}-\tau_{a1})+(\tau_{a2}-\tau_{b2})}{2} + \frac{1}{2}\left(\frac{\beta_a}{\beta_b}-1\right)\left(\tau_{a2}-\tau_{a1}\right)$$

- Remote clock reading

$$\tau_b\left(t\right) = \tau_{b1}-\beta_b d - \frac{\beta_b}{\beta_a}\tau_{a1} + \frac{\beta_a}{\beta_b}\tau_a\left(t\right)$$

$$t = \frac{\tau_a\left(t\right)}{\beta_a} - \alpha_a$$

Accurate Time-stamping

- Accuracy of distance measurement is directly related to the accuracy of timestamping
- Collaboration with Austrian Academy of Sciences
  - SMiLE 3 board

Block Diagram of SMiLE

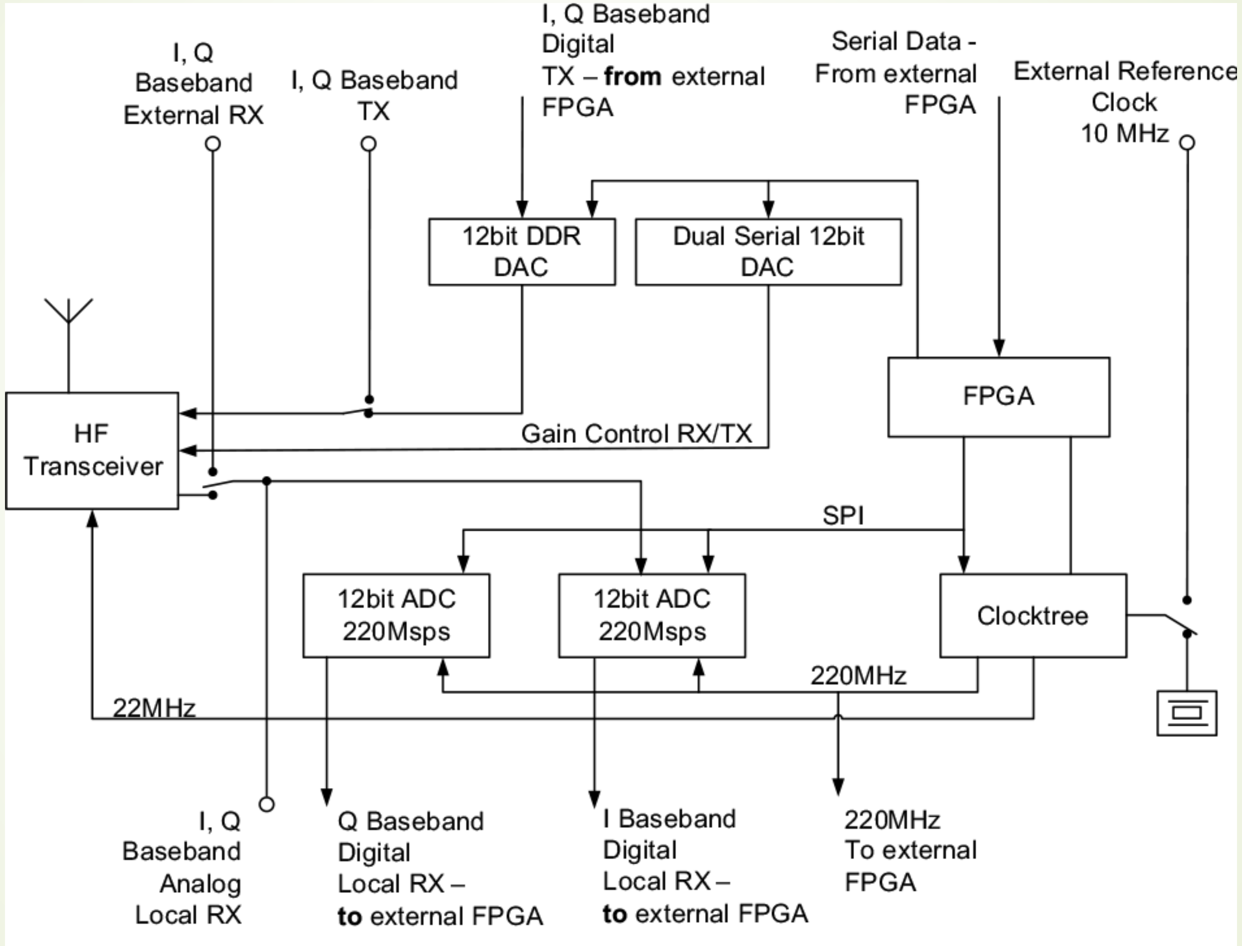


Figure 2. Block diagram of SMiLE PCB

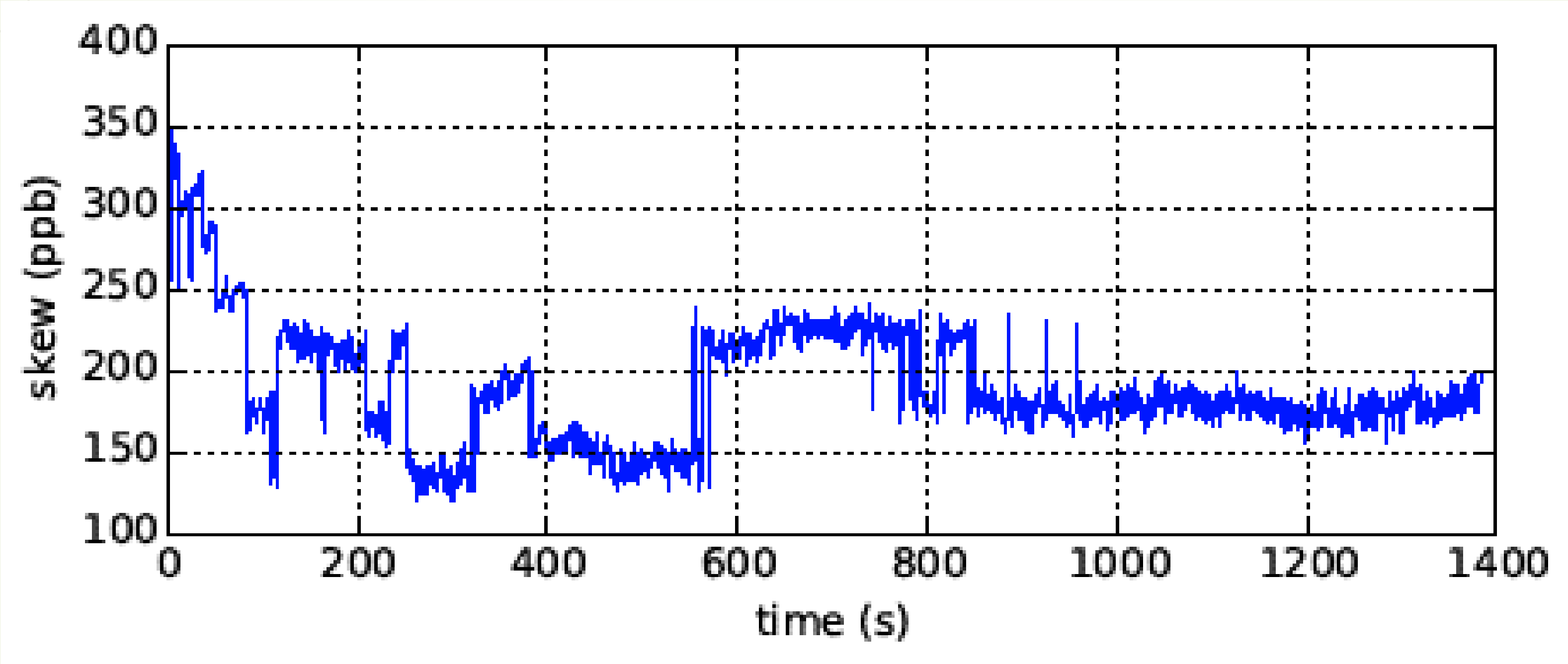
SMiLE Details

- Altera FPGA Cyclone III
- Max 2830 WiFi chip
- Sampling Rate = 44 MHz (22.75 ns Tick)
- Discretization 256 levels (22.75/256 = 88.77 ps)

Measurement Results

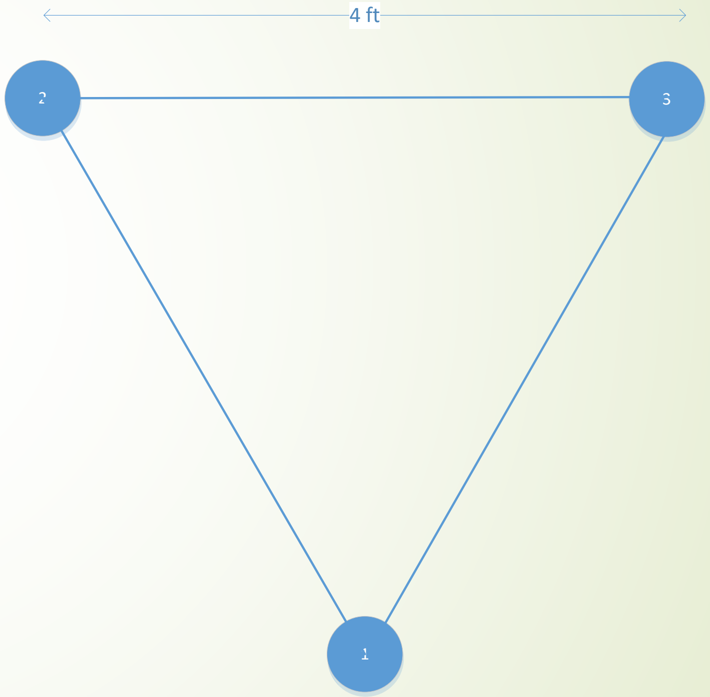
- Time Stamping
  - Tick time 88.77 ps (~2.66 cm)
  - Standard Deviation of Error – 0.97 ticks
  - Stable
- Clocks
  - Have variable drifts ~ (0.119 to 0.364 ppm)

Clock Drift (Skew)



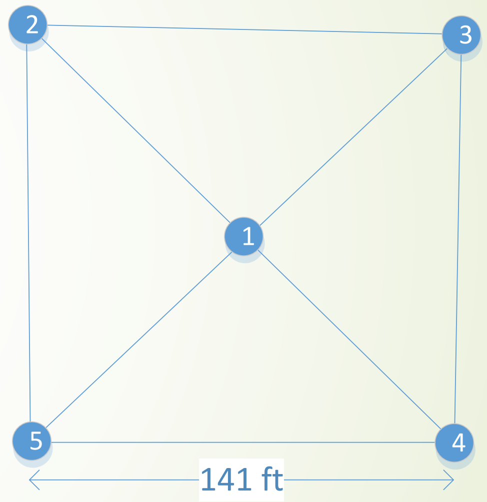
Distance Measurements

- Configuration



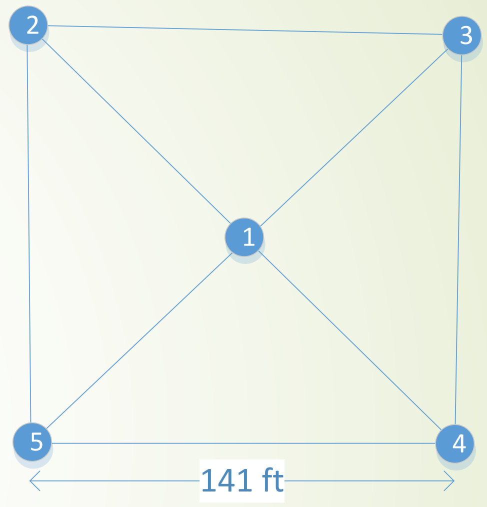
Distance Measurements

- Configuration
  - Outdoors
- Experiment
  - Nodes take turn is sending messages
  - 10ms interval

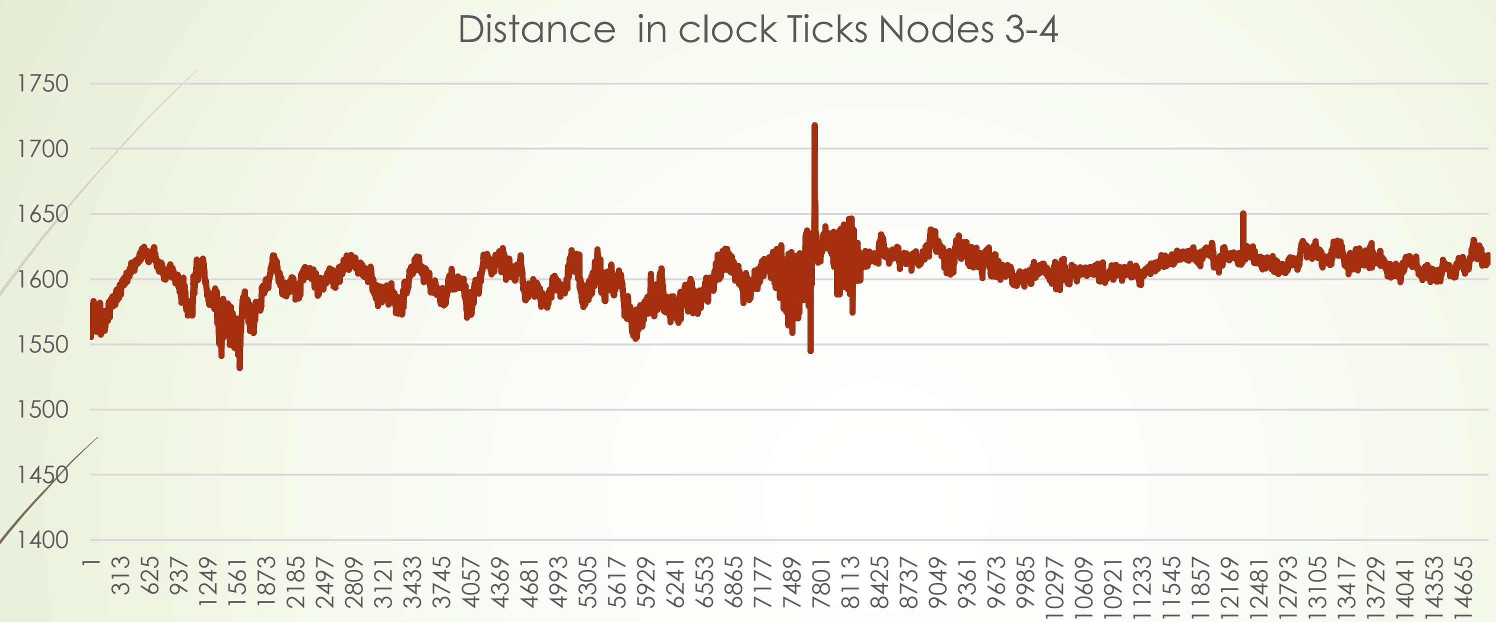


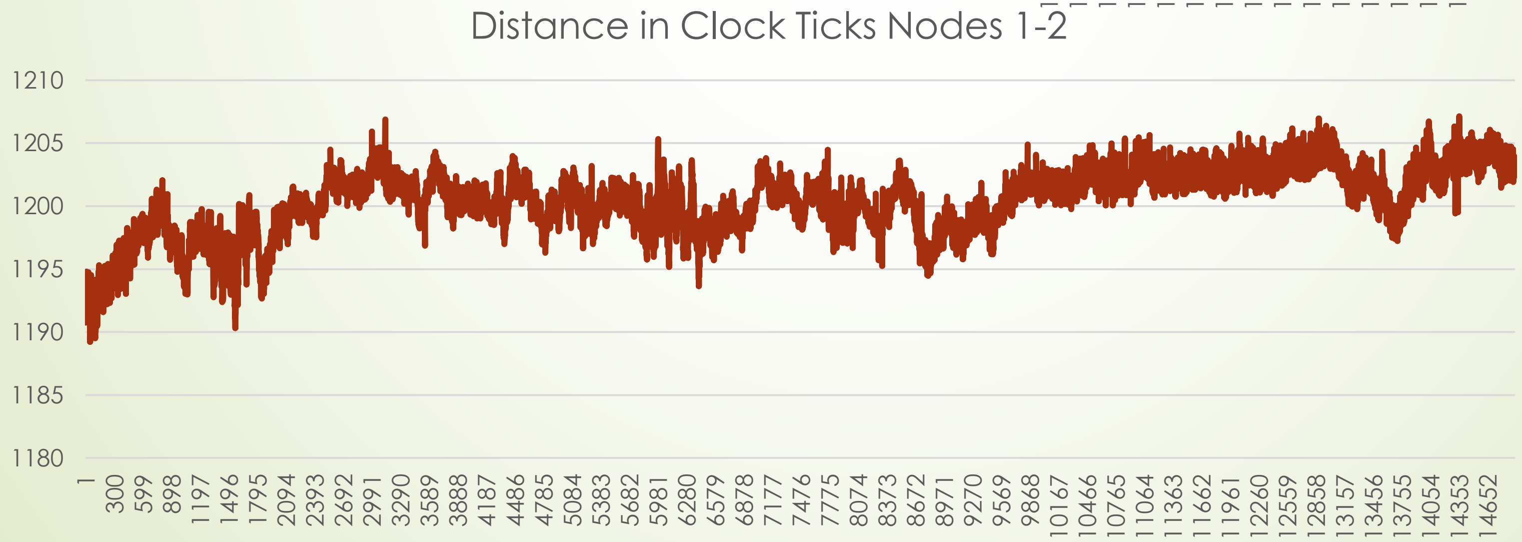
Distance Statistics

stats	path	in ticks	in feet		
mean	12	1200.3790	104.8188		
	13	1169.8708	102.1548		
	14	1170.1178	102.1764		
	15	1182.3626	103.2456		
	23	1681.8644	146.8628		
	34	1603.9611	140.0602		
	45	1656.1120	144.6141		
	52	1639.8012	143.1898		
	24	2352.6710	205.4386		
	35	2313.8754	202.0509		
	path	in ticks	in feet	in cms	in inches
std	12	2.5491	0.2226	6.7845	2.6711
	13	2.4626	0.2150	6.5544	2.5805
	14	3.4353	0.3000	9.1433	3.5997
	15	4.0475	0.3534	10.7725	4.2412
	23	8.9180	0.7787	23.7358	9.3448
	34	15.0450	1.3138	40.0432	15.7651
	45	11.0574	0.9655	29.4299	11.5866
	52	12.2881	1.0730	32.7055	12.8762
	24	3.9620	0.3460	10.5451	4.1516
	35	25.5180	2.2283	67.9175	26.7392



Distance


Distance in clock Ticks Nodes 3-4
1750
1700
1650
1600
1550
1500
1450
1400
1
313
625
937
1249
1561
1873
2185
2497
2809
3121
3433
3745
4057
4369
4681
4993
5305
5617
5929
6241
6553
6865
7177
7489
7801
8113
8425
8737
9049
9361
9673
9985
10297
10609
10921
11233
11545
11857
12169
12481
12793
13105
13417
13729
14041
14353
14665


Distance in Clock Ticks Nodes 1-2
1210
1205
1200
1195
1190
1185
1180
1
300
599
898
1197
1496
1795
2094
2393
2692
2991
3290
3589
3888
4187
4486
4785
5084
5383
5682
5981
6280
6579
6878
7177
7476
7775
8074
8373
8672
8971
9270
9569
9868
10167
10466
10765
11064
11363
11662
11961
12260
12559
12858
13157
13456
13755
14054
14353
14652

Implication

- ASIC Based Technique with accuracy in inches with sub second latencies
- Indoor Location
  - Multipath Effects need addressing

Clock Synchronization

- Mapping Function Based
- With Absolute Time

Mapping Function Based Synchronization

- Normal approach
  - Exchange signals
  - Determine corrections
  - Correct the local clock
- Our Approach
  - Use a free running local clock
  - Exchange messages to determine a mapping function
  - When time information is needed
    - Read time from local clock
    - Map it using a mapping function

Mapping Function

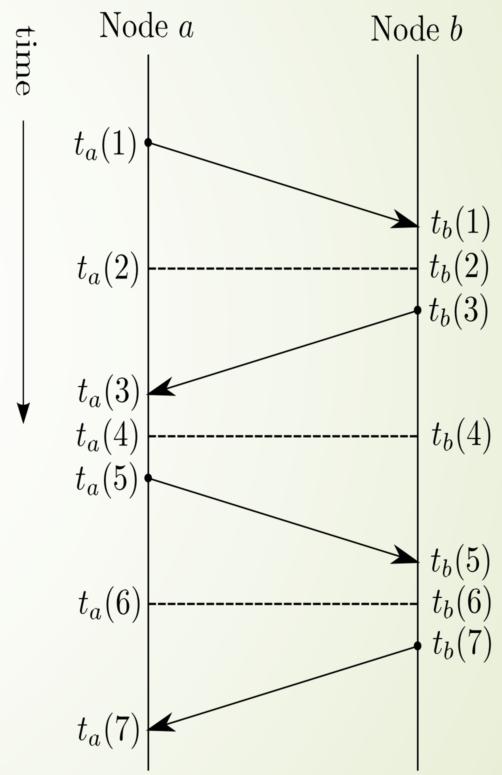
- Two nodes, a and b
- $\varphi_a(t) = t_a$
- $\psi_a(t_a) = t$
  - *Example*

$$\tau_a(t) = \beta_a(\alpha_a + t)$$

- $\varphi_{ab}(t_b) = t_a$
- $\psi_{ab}(t_a) = t_b$

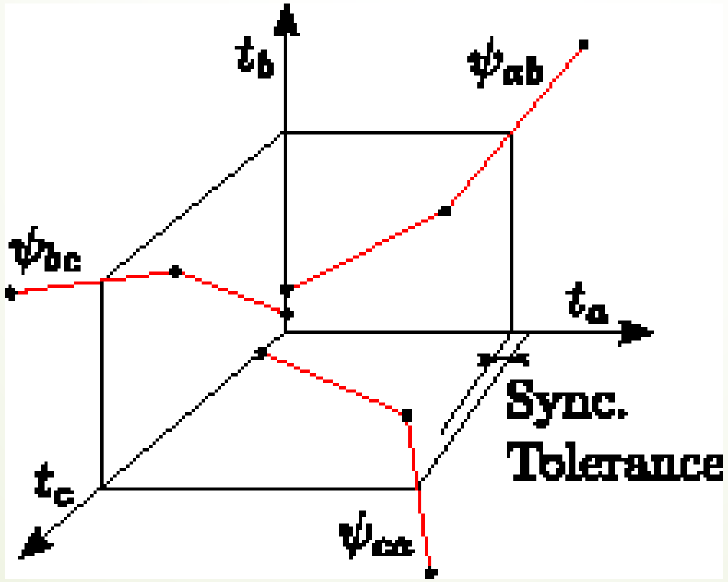
Approach

- Linear model of clock works well over short periods of time
- When exchanging messages, Time instants $t_a(2)$ and $t_b(2)$ are the same time instants in real time.
- Calculate and use a piecewise linear mapping function



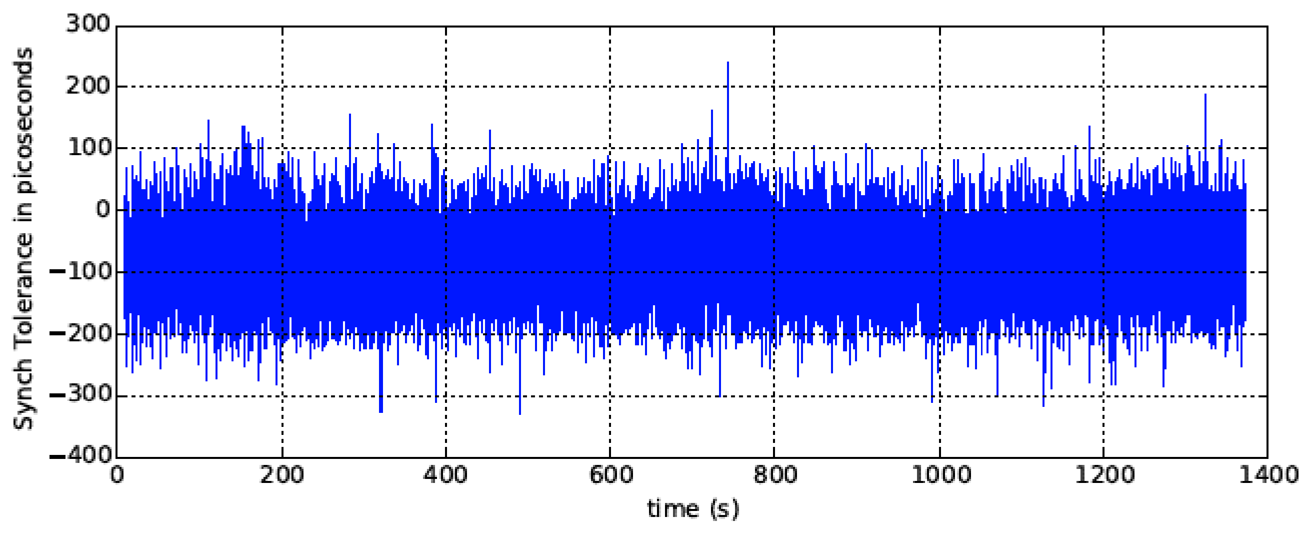
Synchronization tolerance

- How far is the time at a node compared to the mapped time?



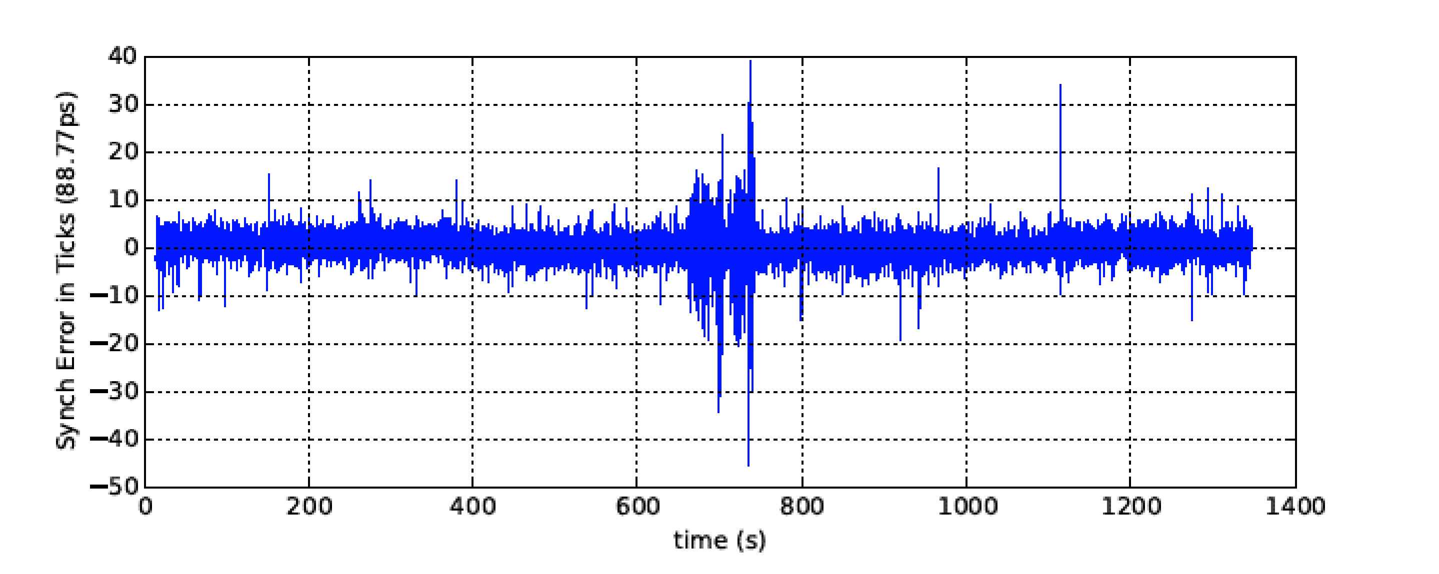
Synchronization Tolerance

- 3 nodes 4 ft apart
- Average ~ 80 ps, STD ~ 60 ps



Synchronization Tolerance

- Five Nodes – 123451 path



Synchronization with Absolute time

- Note that

$$\beta_b d = \frac{(\tau_{b1} - \tau_{a1}) + (\tau_{a2} - \tau_{b2})}{2} + \frac{1}{2}\left(\frac{\beta_a}{\beta_b} - 1\right)(\tau_{a2} - \tau_{a1})$$

- If we can measure d accurately we can determine β the drift rate with respect to real time

Two Approaches

- Over the air
  - The term d is a function of distance and the speed of light.
    - We can keep nodes at fixed distance
    - Speed of light through air changes as a function of temperature, pressure and humidity
    - Monitoring these we can determine the speed of light with an accuracy of one part in 10^9
    - As these parameters change slowly we can have a stable reference during a mission.
- Using a communications means with known delay
  - Fiber with measured delay