Haggle: Tracing Pocket Switched Networks

James Scott, Intel Research Cambridge

Includes work by:
Augustin Chaintreau, Chen-Nee Chuah, Jon Crowcroft, Christophe Diot, Timur Friedman, Richard Gass, Pan Hui, Jason LeBrun, Jeremy Leguay, Anders Lindgren, Meng How Lim

Outline

- Our scenario: Pocket Switched Networking
- Intel Mote trace of human connectivity
- Drive-by WiFi trace of
- Issues/tips with collecting traces

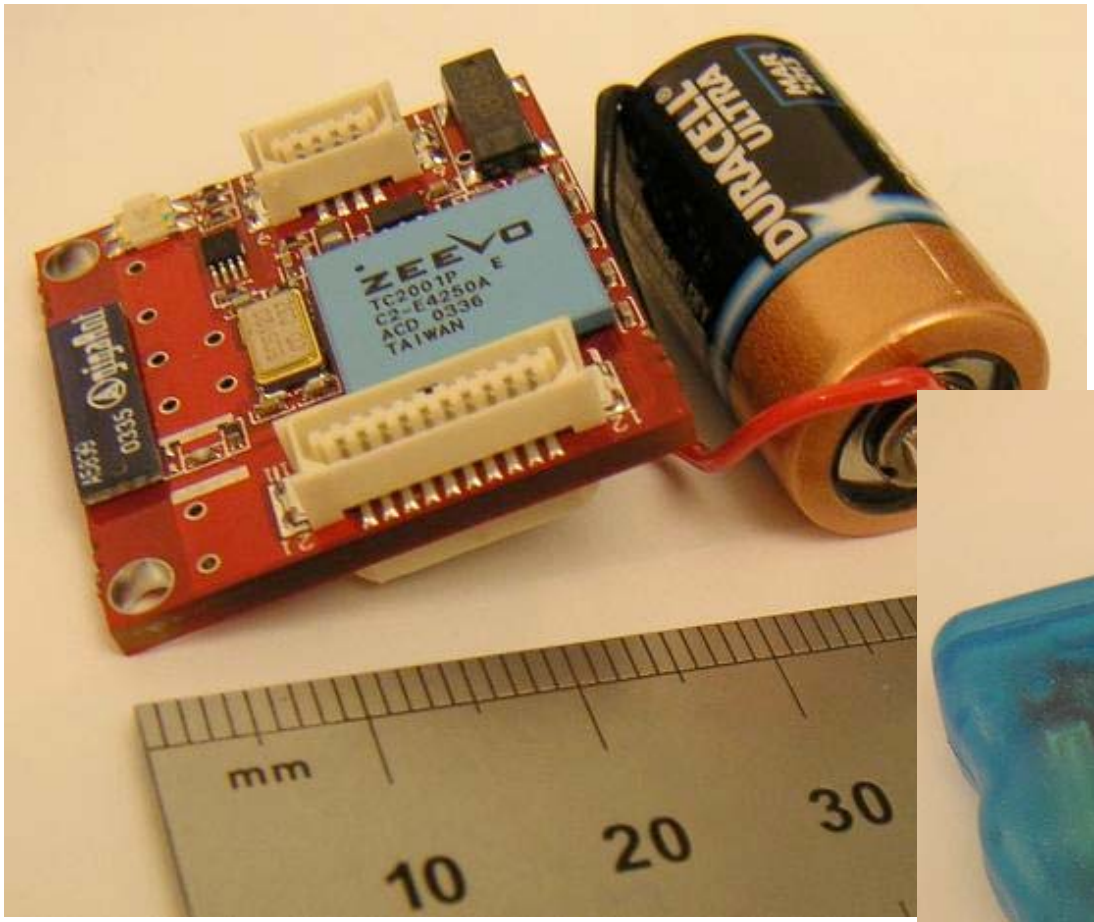
Pocket Switched Networking

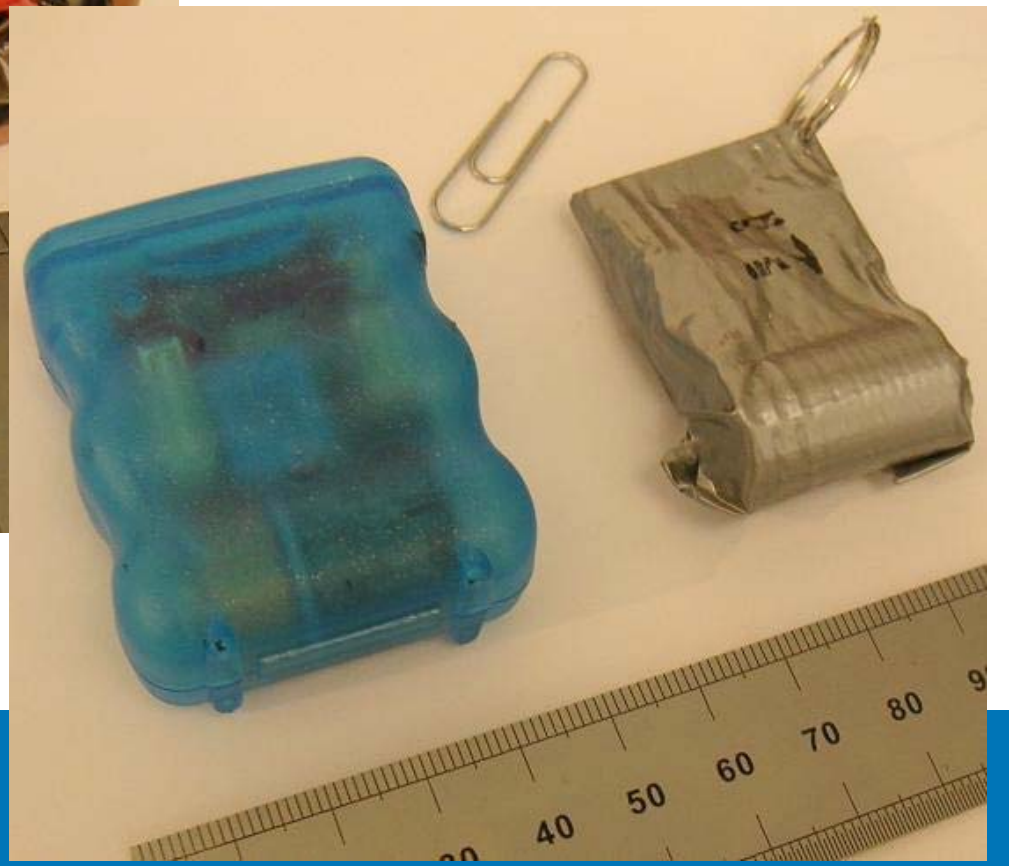
- Current networking architecture works only if infrastructure available
- But is unusable when there is no infrastructure
  - E.g. cannot receive/send email or get webpages
  - Not making use of plentiful local bandwidth
- Scenario: Pocket Switched Networking
  - Mobile users carrying always-on devices in their pockets
  - Connection opportunities with infrastructure and with neighbours directly
  - Make use of *both* types of opportunity

Need for traces in PSN

- To design for PSN, we need to understand the properties of users' contacts with each other, and with access points
  - How often, how long, with whom
- No fully suitable traces
  - Dartmouth and UCSD WiFi AP traces can be used to estimate user-user contacts
  - RealityMining trace from MIT useful, but came later
- We decided to measure
  - User-user contact patterns for real user groups
  - Drive-by performance of WiFi to discover properties of contacts

Hardware for contact trace: Intel Mote





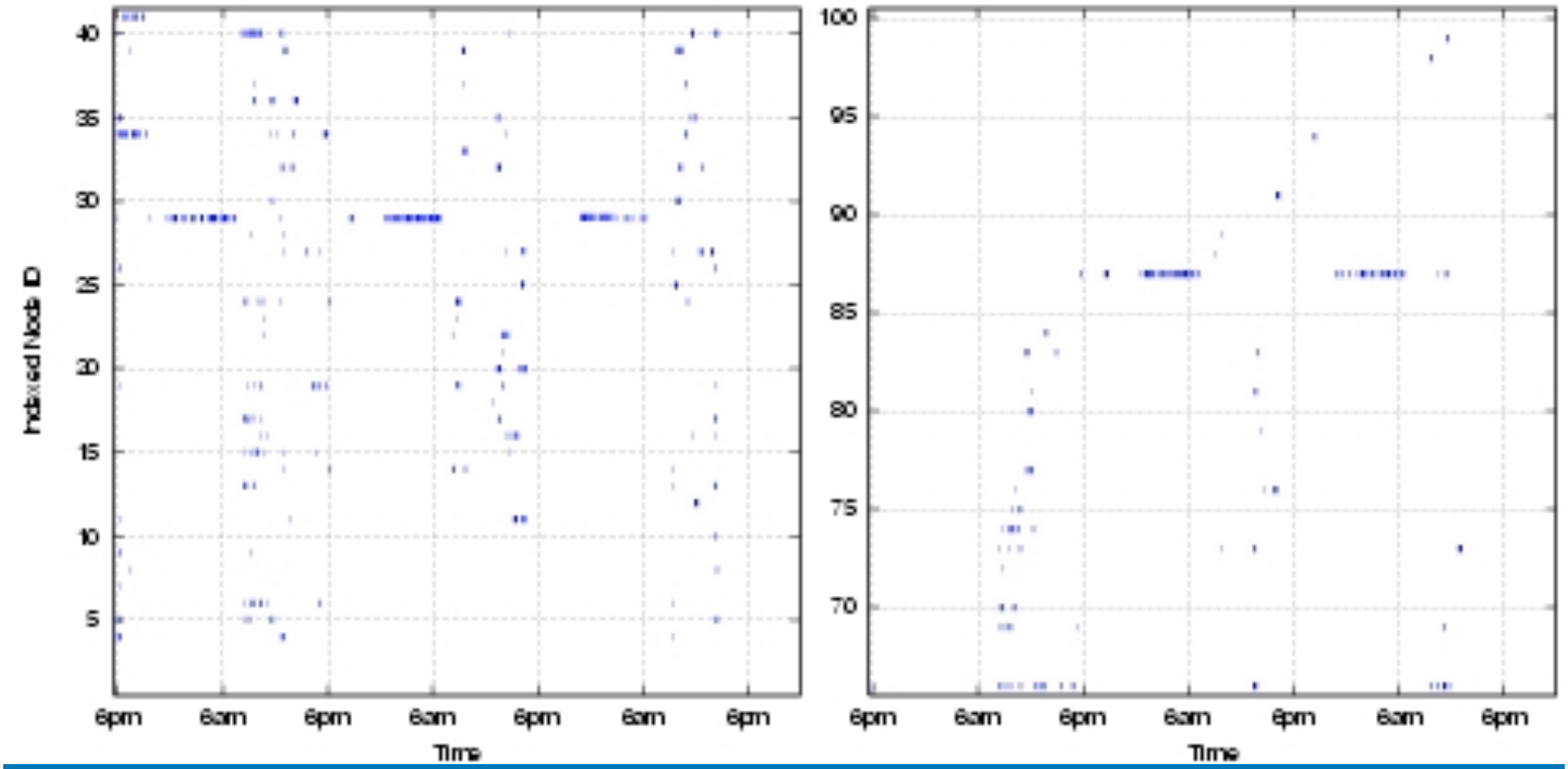
Trace gathering

- Deploy to a user population
- See and log iMotes and other discoverable Bluetooth devices every 2-10 minutes (depending on expt)
- 9 experiments so far (*4 on crawdad so far*)
  - *20 motes, 3 days, Intel employees*
  - *20 motes, 3 days, Cambridge PhD students*
  - *50 motes, 5 days, INFOCOM 2005 attendees*
  - 50 motes, 5 days, Hong Kong school students
  - 50 motes, 5 days, Hong Kong “random” group
  - 50 motes, 5 days, UC Davis buses
  - *80 motes, 20 days, Cambridge undergrads (with some stationary nodes)*
  - 100 motes, 5 days, INFOCOM 2006 (with some stationary nodes)
  - 100 motes, <1 day, Paris Roller Tour

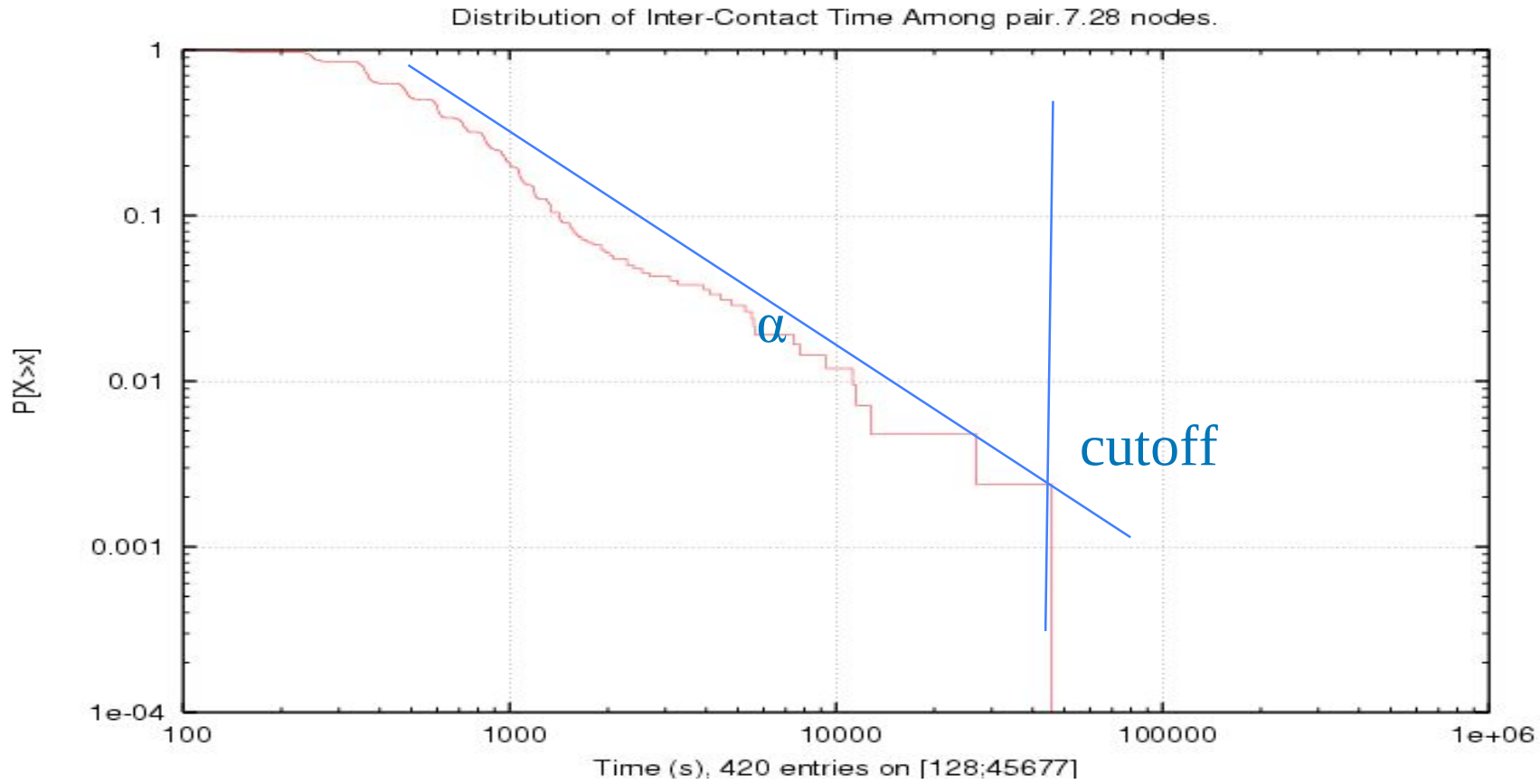
iMote trace sample (INFOCOM 05)

iMotes

Other devices



Result 1: Inter-contact time follows approximate power law



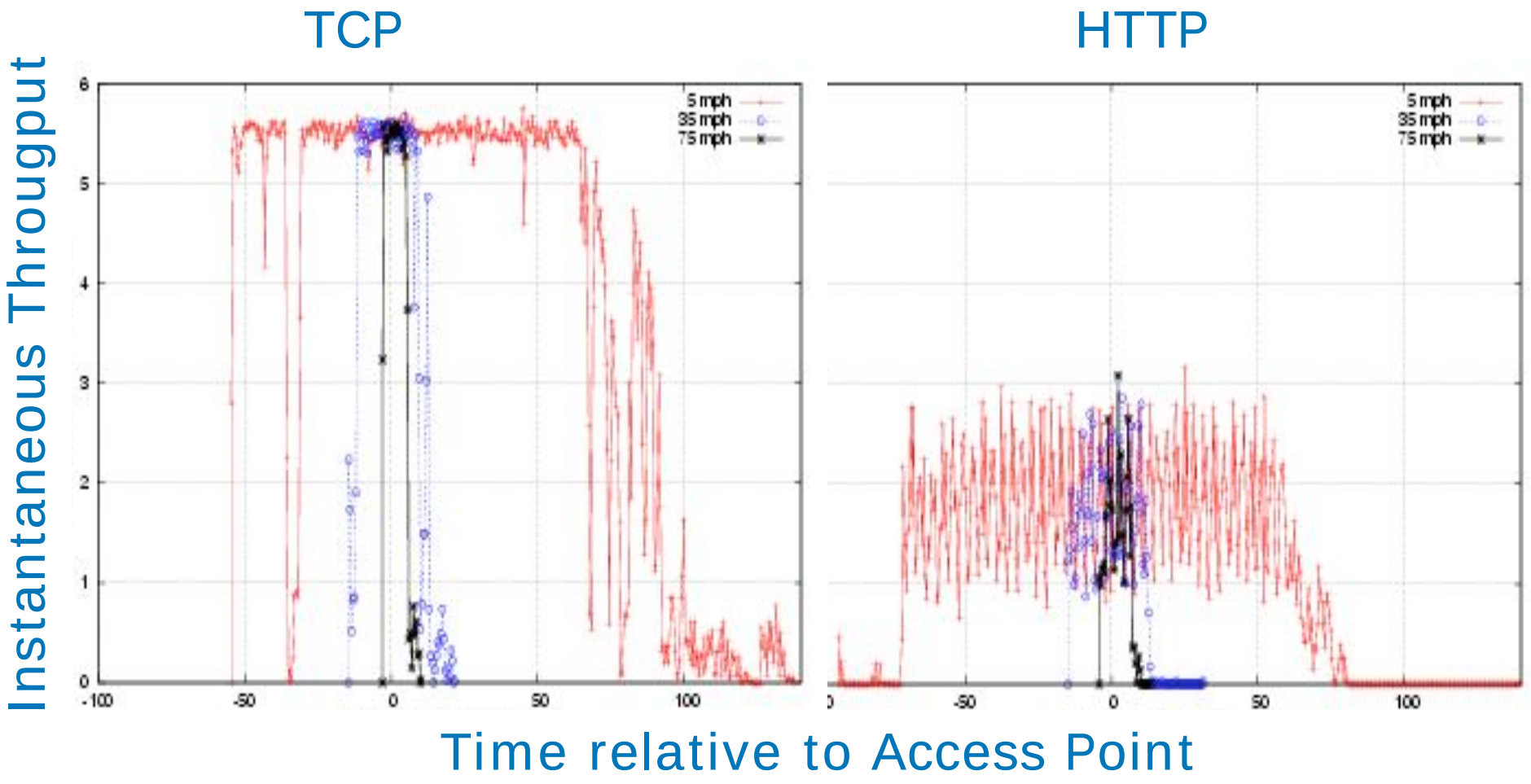
Result 2: Data is highly non-uniform

- Times of day have widely different alphas
- Individuals' sighting level varies widely
- Pairs of nodes' sighting level varies widely
  - Social relationships as well as "familiar stranger" effects
- Communities of nodes can be identified
  - with high mutual connectivity
  - i.e. lower-latency network paths when inside the community

In-Motion 802.11 Trace

- Drive past access point on long, straight, flat road
- Laptop on passenger's lap using built-in 802.11 interface and antenna
- Vary:
  - speed (5mph to 75mph)
  - data traffic (UDP, TCP, web)
  - backhaul (none, 1Mbit/s limit, 100ms delay, both)

Results – instantaneous throughput



Implications

- Link layer performs very well while in-range
- Association delay is not a problem, but DHCP, VPN, email login, etc are
- Application-layer protocols are interactive and waste a lot of bandwidth – big gain to be had optimising them for PSN
- Trace recently used [Hadaller, CHANTS 06] to show benefit of new MAC algorithm optimised for drive-by situations

So, you've decided to collect a trace

- Plan a series of increasing-size experiments
  - It never works 1st time, and often not 2nd or 3rd times
  - Expect hardware bugs, software bugs, and wetware bugs
- Document everything
  - Makes sense if you know you'll need >1 trace
  - Others may want to reproduce trace in their environment
- If possible, get trace users involved early
  - Can help with effort of collecting and processing trace
  - Make sure the traces suffices for other work if possible

Issues faced in collecting traces

- Consent and human subjects
  - Quite lightweight at Intel, more heavyweight elsewhere
  - Specific issue: we collect data about devices (e.g. mobile phones) of users who *haven't* consented
- Logistics of deploying hardware to users
  - Deployment, collection, etc
  - Return rate <100% (lost, broken, etc)
- Post-processing
  - Time often underestimated
  - Data extraction, synchronisation,
  - Anonymisation – seems trivial, but is often insecure

And the main issue is…

- Large time overhead
  - Time taken to decide on random waypoint: zero
  - Time taken to gather a 100-node iMote trace: ~3 person-months
  - We wait for a paper to be accepted before we publish a trace
  - Crawdad helps by handling requests for traces, and by generating citations to make it worth it

So why collect traces?

- *Trace first*: traces are the only way to really understand a problem that needs to be solved
  - Avoids assumptions that will come back to bite you
- *Trace last*: traces are the most realistic way to evaluate a solution without bringing in errors in evaluation
- *Trace together:* traces are a great way to evaluate different systems/algorithms against each other “in the wild”
  - Evaluations are too often for very narrow circumstances
  - Amortise trace collection cost among a community

Call for help: Tracing user behaviour

- Our work so far: measured network contacts between humans
- Still need trace of *traffic patterns*
  - Implicit assumption of uniform traffic
  - Google desktop logs?
  - Chicken and egg problem because of users
  - Simultaneous traffic and connectivity if possible
- More generally: need to understand the *users* in order to properly evaluate the utility of all our work
  - Does it *really* matter that packet X was dropped?
  - Is there *real* benefit if packet Y experiences less latency?


Questions?

Taster: Haggle demo

- Haggle is the network architecture we've designed from "clean slate" to operate in PSN environments
- Routing algorithms and protocols are being informed by iMote and vehicular traces
- Architecture allows ad hoc neighbour connectivity to be used as easily as infrastructure
  - Standard email and web apps when there are no AP
  - User doesn't have to change apps or manually configure
- Tuesday 3:30-5:30 demo session