Simulation in Large Scale Sensor Networks

Barney Maccabe
University of New Mexico

Santa Fe, New Mexico 17 Oct 2006
"I have finally reached the point in my career where I can live on the results of my graduate students"

Collaborators

- **UNM** - sensor network systems software
  - James Horey, PhD Student
  - Jean-Charles Tournier, Postdoc
  - Eric Nelson, MS student
- **LANL** - sensor network system simulation
  - Sean Brennan, MTS (& UNM PhD student)
  - Angela Mielke, project lead
- **SNL** - multi-scale system simulation for MPPs
  - Rolf Riesen
  - Keith Underwood
Multiscale Simulation for Systems Software

- Use simulation to study systems software approaches
  - fault tolerance: simulate a large system and inject faults
  - impact and viability of systems software at scale: how does it work
- More “emulation” than “simulation,” for now
- Traditional use of simulation in sensor networks
  - application testing
  - ad-hoc routing protocols
Basic simulation tools

- **ns-2** - network simulation
  - packet level simulation, great for studying changes to TCP/IP
  - plug-ins for wireless protocols
  - ad-hoc wireless routing protocols
- **Simple Scalar processor simulation**
  - instruction set simulation
  - great for studying branch prediction and out-of-order execution
- **Mambo**
More

- Tiny OS simulation (TOSSIM)
  - http://www.tinyos.net/
  - develop applications, small number of nodes
- EmStar
  - http://cvs.cens.ucla.edu/emstar/
  - sensor networks involving Linux-based platforms
- SWANS
  - http://jist.ece.cornell.edu/
  - like ns2, better scalability, Java apps
- and a whole host of others...
Sensor Network Environment

- Motes - sensing
  - 1,000’s
  - limited compute, communication and, power

- Stargates - compute
  - 100’s
  - more compute, memory, power, and communication

- Computation in the network
Software Stack

- Programming
  - writing the functions that will run on the nodes of a sensor network

- Tasking
  - distributing functions to the nodes of a sensor network

- Groups
  - Static and dynamic
  - Collective operations
Basic Simulation

- Goal: evaluate software stack
- Kensho API library
- Each sensor simulated by a process (linked with the Kensho library)
- Wireless communication provided by a broadcast daemon
- Sensor reads come from environment simulator
Fire Detection: Group Operations

smaller groups: higher resolution, more communication
Motion Tracking: Dynamic Groups

Group is defined by magnetometer activation

Tracking Accuracy

The object starts at position $p_3$ and moves at a velocity of $w_13$ units per time interval.
N-Sim: Scalable Simulation

- Goal: run large scale sensor network applications without modification
- Be able to control communication and node faults
- Order of magnitude speedup
- Initial focus on typical software stack: TinyOS and Linux
- Long term will be used to study the Kensho/Kaizen stack
Virtualization using Xen

- Xen provides base level virtualization
  - sharing CPU between multiple OSes
  - Dom 0 provides access to other resources (e.g., network)
- Type 1
- para-virtualization
N-sim: Scalable simulation for sensors

- Simulate each sensor or stargate using a separate virtual machine
- Direct execution of applications
- On modified operating systems
- Simulation daemon runs in Dom 0
  - network and sensing
Scaling: Cluster of nodes

- Use multiple cluster nodes
- Simulation Daemons coordinate and exchange communications
- MPI for communication and coordination
Full system simulation for MPP design
Multiscale issues

- Want full, cycle accurate, simulation
  - from detailed processor simulation
    - device drivers
  - to large scale application
  - including all software layers
    - impact of changing implementation of services
- Also want simple, proof of concept simulation
  - fast, clean and simple
Thanks

- **LANL & NNSA**
- **SSL UNM**
  - Prof Patrick Bridges
  - Postdocs
    - Patricia Crowley (now at Gonzaga)
    - Patrick Widener