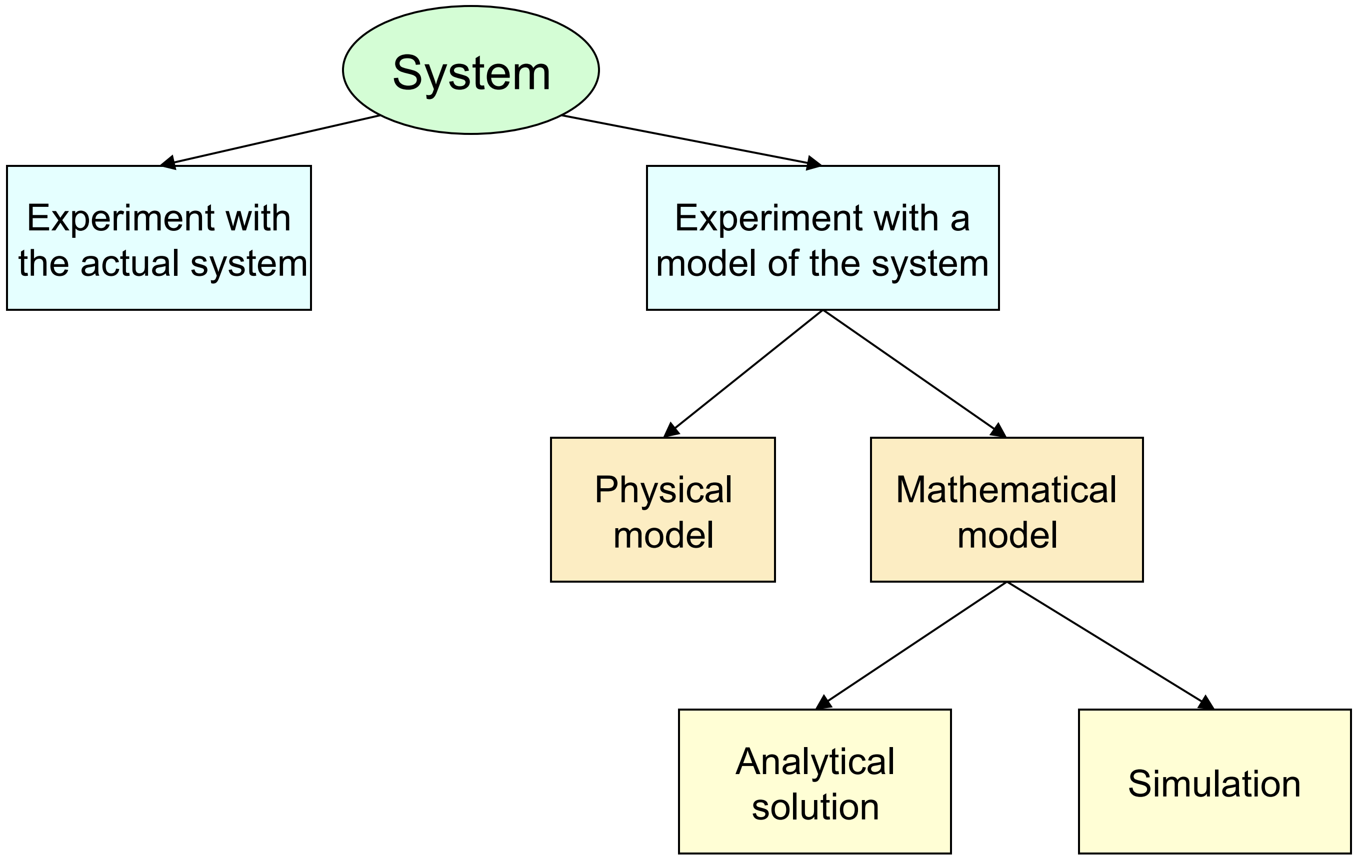
Simulation & the Simulation of Wireless Networks

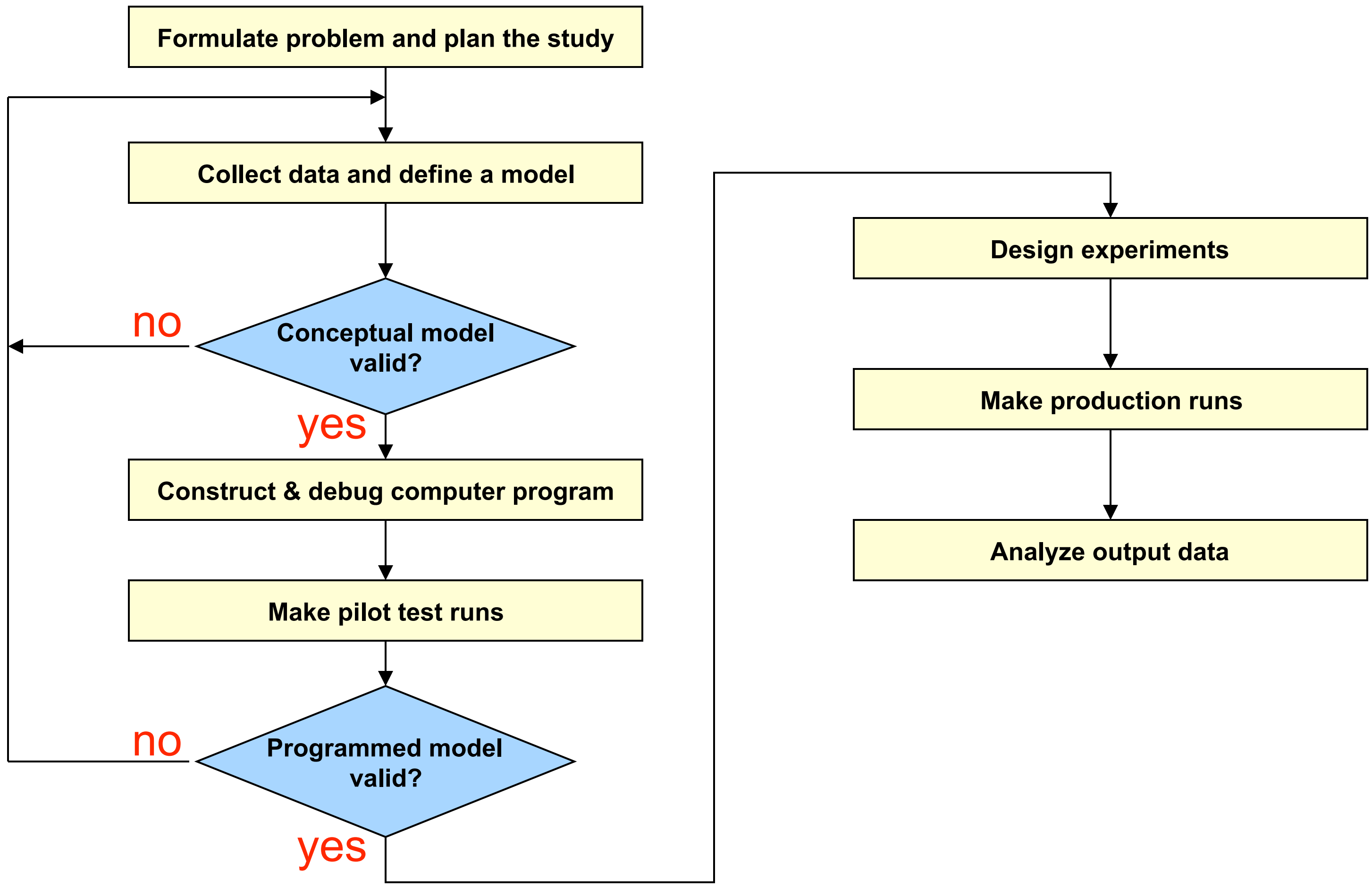
L. Felipe Perrone
Assistant Professor of Computer Science
Bucknell University

Studying a system



Source: *"Simulation Modeling and Analysis"*, Averill M. Law and W. David Kelton, 3rd Edition, 2000.

Steps toward a Simulation study



Source: *"Simulation Modeling and Analysis"*, Averill M. Law and W. David Kelton, 3rd Edition, 2000.

Steps toward a Simulation study

Formulate problem and plan the study

↓

(INPUT) MODELING

↓

VERIFICATION

↓

Make pilot test runs

↓

VALIDATION

↓

Design experiments

↓

Make production runs

↓

OUTPUT ANALYSIS

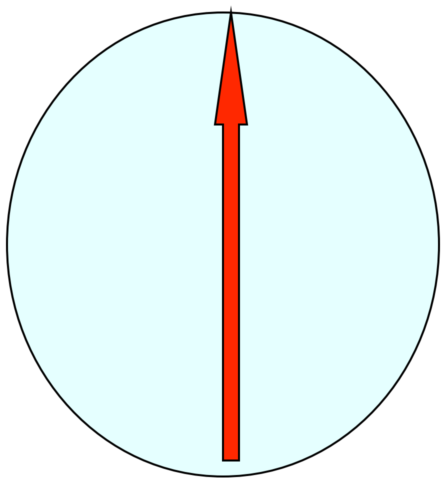
Simulation

- Continuous Simulation
- Discrete-Event Simulation
  - Next-event time advance
  - Fixed-increment time advance
- Combined Discrete-Continuous Simulation
- Monte Carlo Simulation

Continuous Simulation

State variables: The models are described by quantities that characterize the state of the system at some point in time.

Time: In reality, time is continuous, but a computational representation must work with a *discretization* of time. The system evolves in timesteps.



clock

state variables

$x(t) =$ predator population

$y(t) =$ prey population

initial conditions

$x(0)$

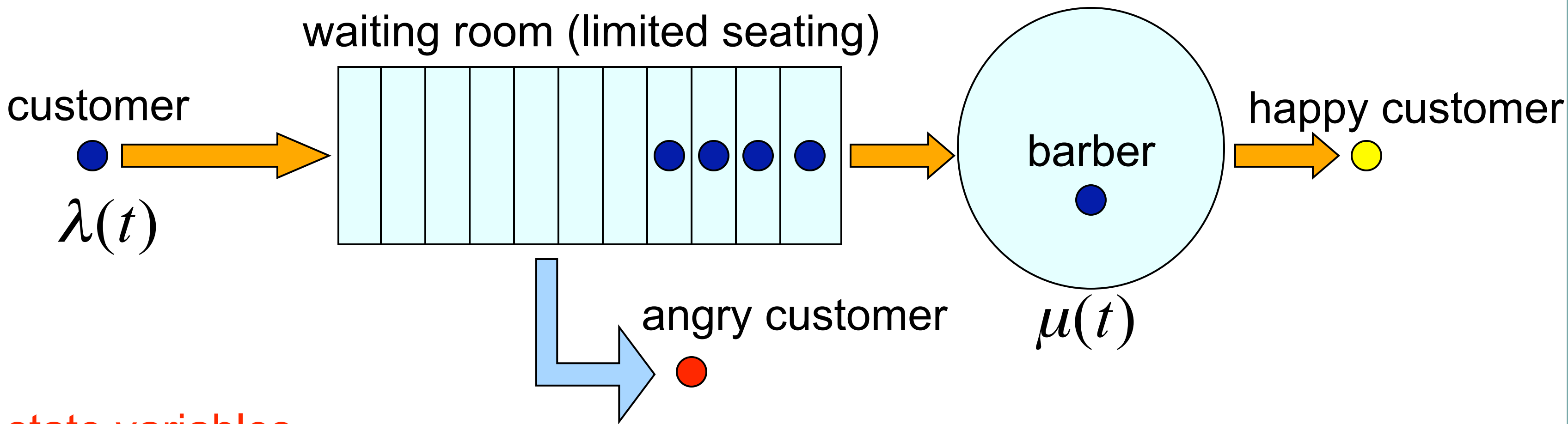
$y(0)$

evolution of the system

$$\frac{dx}{dt} = rx(t) - ax(t)y(t)$$

$$\frac{dy}{dt} = -sy(t) - bx(t)y(t)$$

Discrete-Event Simulation



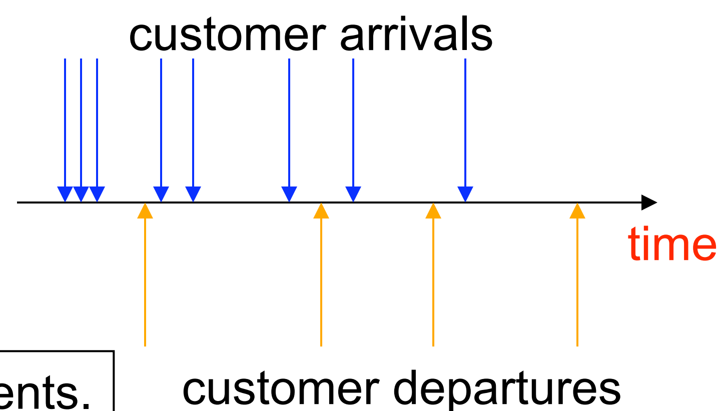
state variables

$\lambda(t)$ = arrival rate at time t

$\mu(t)$ = service rate at time t

$wc(t)$ = number of waiting customers (0..k)

$b(t)$ = number of customers in service (0..1)



The system evolves in random time increments.

Simulation helps when:

- The system is too complex to be evaluated analytically.
- Simulation allows one to study the system under varying scenarios of operating conditions.
- The scenarios can be controlled better in a simulation than in the real world.
- Simulations allows us to study the evolution of the system over a long period of time in a short amount of real time.

Simulation, however, is not trivial

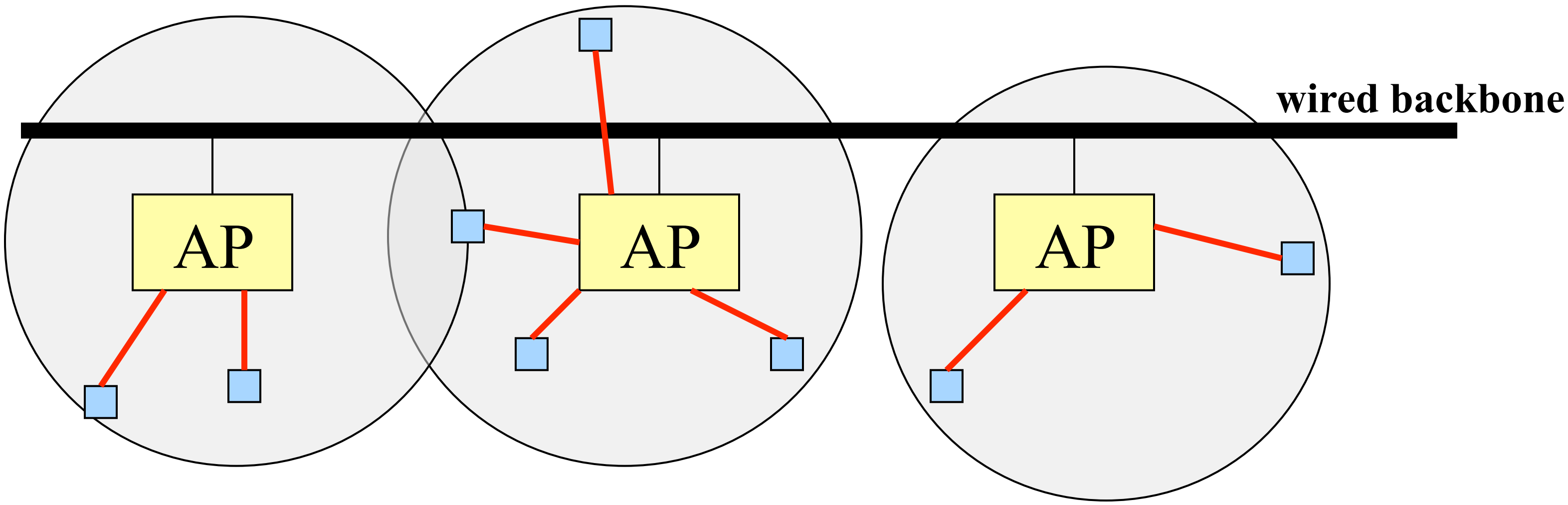
- Modeling requires extended experience with the real system.
- Models must be ***validated*** so that one can trust the simulation results.
- Once a model is constructed, a computational implementation must be derived and ***verified***.
- Simulation produces estimates of the metrics in the model. One needs to be careful in the interpretation of these results.
- It is tempting to collect large amounts of data. Analyzing all this body of data is time consuming and requires good methodology.

- The simulation of very large models is computationally expensive: one needs a very fast computer or perhaps several fast computers.

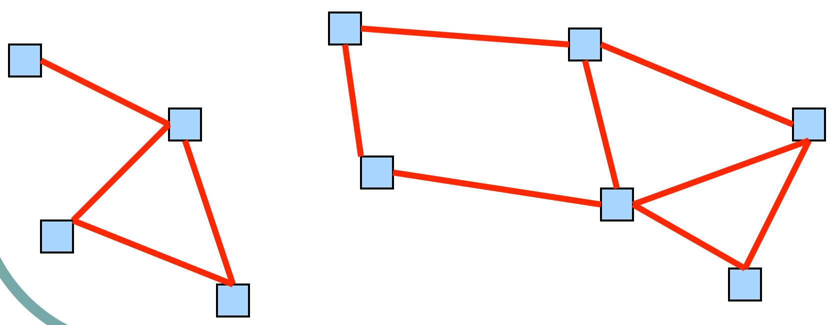
- Parallel or distributed simulation is, in fact, **very** hard.

Wireless Networks

Wireless Hot Spot or Fixed Infrastructure (IEEE 802.11 PCF)

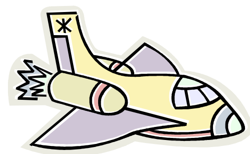


Wireless Ad Hoc (IEEE 802.11 DCF)

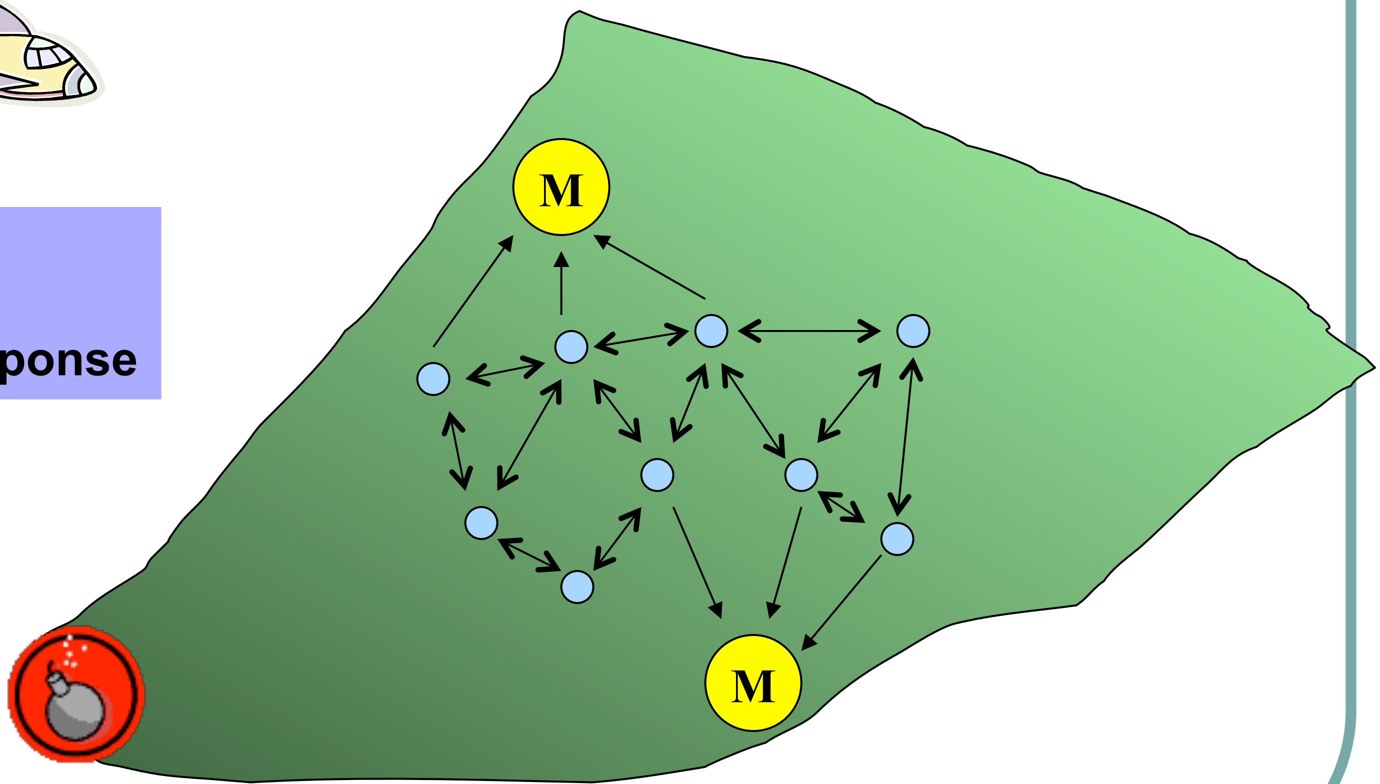


- Easy to deploy.
- Good in changing environments.
- Allows for node mobility.
- Self-configurable.
- Scalable?

Motivation: Sensor Networks



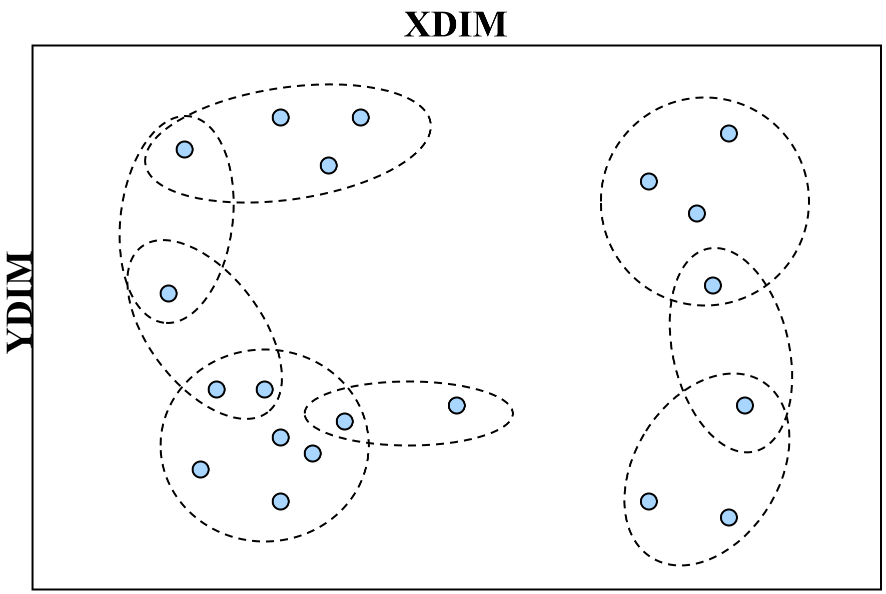
Intelligence, Surveillance, Emergency Response



Technical Challenges

- **Energy constraints**: No wires, no power source.
- **Level of dynamics:** Weather, terrain, RF interference, network traffic.
- **Self-configuration:** neighbor discovery, routing tables, health of links.
- **Scaling:** Very large number of nodes complicates protocol design.

Structure of a Wireless Ad Hoc Network Model (macro view)



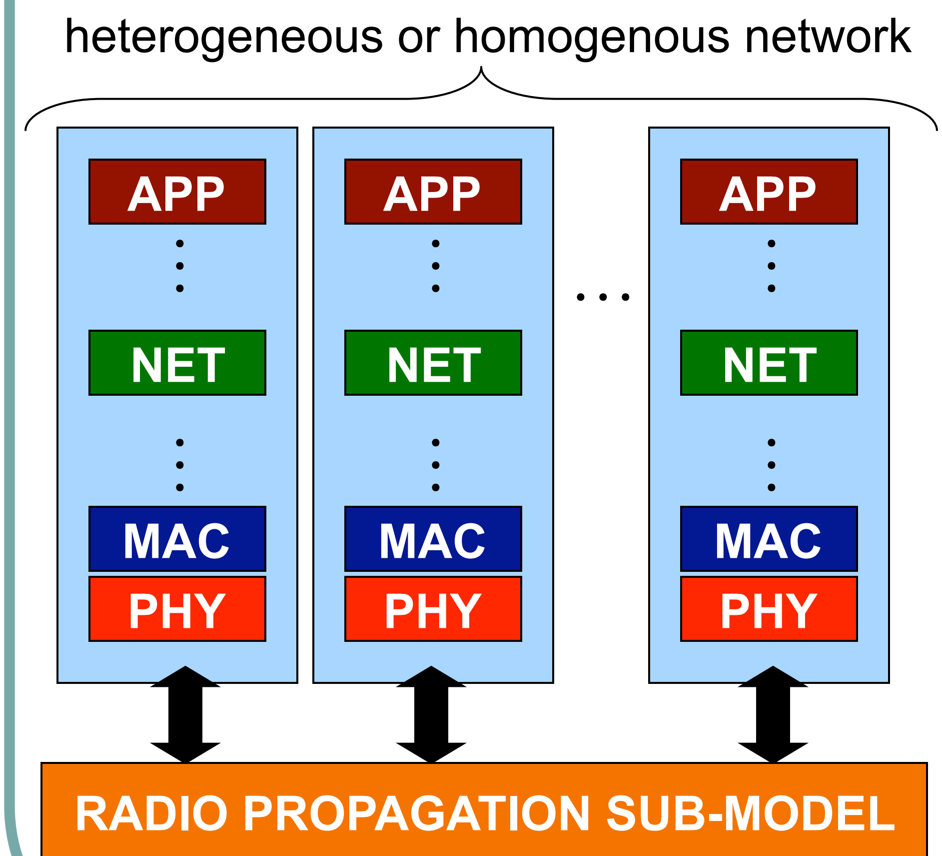
Environment Sub-models

Space:
geometry, terrain

Mobility:
single model, mixed models

Propagation:
computational simplicity (performance), accuracy (validity)

Structure of a Wireless Ad Hoc Network Model (micro view)



Network Node Sub-models

Physical Layer:
radio sensing, bit transmission

MAC Layer:
retransmissions, contention

Network Layer:
routing algorithms

Application Layer:
traffic generation or "direct" execution of real application

Security issues

- Desirable properties:

Confidentiality	Availability
Authenticity	Accessibility
Integrity	Self-organization
Freshness	Non-repudiation
Scalability	Flexibility

- As of today, the network can be vulnerable at multiple levels:
  - PHY: radio jamming.
  - MAC: DoS via fake requests or schedules.
  - NET: fake route advertisements (black hole attack).
  - A funny but scary notion: “caveman” attacks.

The need for simulation

- Protocol design has always been a tough problem. Protocol validation and verification have always been even tougher.
- We have a complex system that defies mathematical analysis.
- This system has several components tightly inter-connected: interactions complicate behavior.
- Experiments will call for repeatability and controllability.

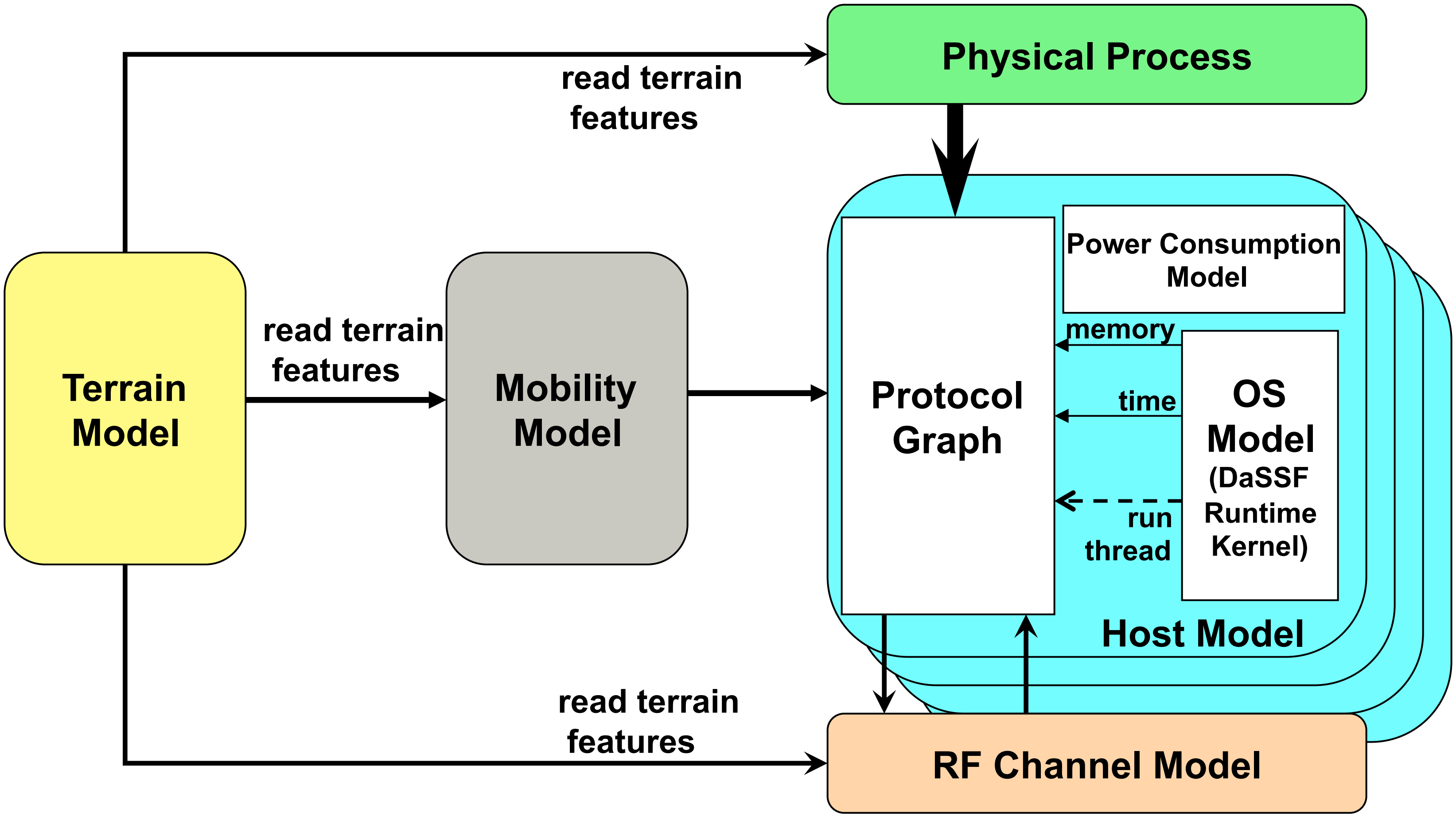
Rapid simulation: a tough goal

- Radio propagation: a continuous process in continuous time.
- Teletraffic: a discrete process in continuous time.
- The simulation must cope with time scales of very different resolution. Mixing them and achieving high performance could be a tough goal.
- Parallel simulation is an option.

Wish list for a Wireless Networks Simulator

- Detail
- Completeness
- Performance
- Scalability

The architecture of SWAN



SWAN model description: DML

```
model [
 arena [
  mobility [
   # nodes are stationary
   model "mobility.stationary"
   deployment "random" # uniform distribution
   xdim 1500 # width of virtual space
   ydim 1500 # length of virtual space
  ]
  network [
   netid 1 # wireless network 1
   model "network.fixed-range"
   cutoff 350 # signal cutoff distance
  ]
 ]
 propagation [
  model "propagation.friis-free-space"
  carrier_frequency 2.4e9
  system_loss 1.0
 ]
```

```
# network node: configuration example
 host [
  id 1
  graph [
   session [
    name "app" use "tstapp.sess-app-session"
    packet_size 512  iat 1.0 show_report true
    peer [ netid 1  hostid 2  iface 0 ]]
   session [ name "aodv“
    use "routing.aodv_sim.swan-aodv-session"
    netid 1 show_report true ]
   session [ name "net" use "net.ip-session"]
   session [ name "arp“ use "net.arp-session" show_report true ]
   interface [ id 0 # identification of network interface card
    netid 1 # identification of wireless network
    session [ name "mac" use "mac.mac-802-11-session"
     show_report true]
    session [ name "phy" use "phy.phy-802-11-session"
     bandwidth 11e6 accumulative_noise true
     interference_threshold -111.0
 ] ] ] ] ]
```

Where things get complicated

- **Physical Processes:** We need to simulate different physical phenomena accurately and rapidly.
- **RF Channel Model:** Propagation models are mathematically very complex. We need to abstract and take only the most relevant details so that the models scale.
- **Scale:** Large number of nodes consume large amounts of memory. Large number of nodes mean large number of computing threads adding a big burden to scheduling.
- **Direct execution:** Different code, potentially different behavior. We want to allow the simulator to run the same code that runs in the real system.

What we're doing with SWAN

- Evaluate routing protocols' robustness to dynamic changes in propagation conditions, scaling, etc.
- Evaluate the network's robustness to "***caveman attacks***".
- Evaluate the impact of ***best practices*** in the simulation of wireless ad hoc networks.

The development of SWAN

Project started in 2000.

First milestone: The simulation of 10,000 nodes running WiroKit, a proprietary routing algorithm developed by BBN Technologies.

Second milestone: Used in the development and experimental study of a high-performance model for 802.11b.

Third milestone: Used as substrate in the development of a simulator for Berkeley motes running TinyOS. Prototype constructed as proof-of-concept for framework on the eve of the release of nesC and major version update of TinyOS.

Fourth milestone: Used in the development and experimental study of lookahead enhancement techniques.

... and then came the million dollar question:

How accurate are SWAN simulations? Are we doing it right?

Validation by proxy bombed

We looked for simulation studies done with other simulators that we could use as reference to validate SWAN.

Roadblock: We found it very difficult to repeat previously published studies because we could not obtain information on *all* their settings (models and/or parameters). At times, we also failed to understand why certain parameter values had been chosen and perpetuated in the community.

Roadblock: We could not find incontrovertible evidence that the simulators used in those studies had been validated.

We resorted to comparing SWAN models to those of other simulators only to discover inconsistencies or errors in their models.

Crisis, what crisis?

Pawlikowski et al: **"On credibility of simulation studies of telecommunication networks"**. *IEEE Communications Magazine* 40 (1):

"An opinion is spreading that one cannot rely on the majority of the published results on performance evaluation studies of telecommunication networks based on stochastic simulation, since they lack credibility. Indeed, the spread of this phenomenon is so wide that one can speak about a deep crisis of credibility."

Crisis indeed...

Kotz et al. "**The mistaken axioms of wireless-network research**". Technical Report TR2003-467, Dept. of Computer Science, Dartmouth College, July, 2003:

"The 'Flat Earth' model of the world is surprisingly popular: all radios have circular range, have perfect coverage in that range, and travel on a two-dimensional plane. CMU's ns2 radio models are better but still fail to represent many aspects of realistic radio networks, including hills, obstacles, link asymmetries, and unpredictable fading. We briefly argue that key "axioms" of these types of propagation models lead to simulation results that do not adequately reflect real behavior of ad-hoc networks, and hence to network protocols that may not work well (or at all) in reality."

Why is it so difficult?

- Models for a wireless networks are complex and have many, many parameters. Articles in print can't afford to list all the parameters used in a study.
- There isn't a general consensus on the appropriate composition of the model (i.e. protocol stack) for wireless networks.
- We're not all speaking the same language all the time: people may refer to the name of a well-known model and actually implement a different one (the terminology is sometimes perverted).
- Some of the people doing simulations lack wireless networking expertise (improper modeling), while others who have that expertise don't understand much about simulation (improper output analysis).

Experimental scenario: everything counts in large amounts…

RF propagation: 2-ray ground reflection, antenna height 1.5m, tx power 15dBm, SNR threshold packet reception.

Mobility: density 7 neighbors per node, initial deployment triangular, stationary (pause=H, min=max=0), low (pause=60s, min=1, max=3), high (pause=0, min=1, max=10).

Traffic generation: variation of CBR; session length=60s, ist=20s, destination is random for each session, CBR for each session, packet size=512 octets, vary packet rates to produce 16kbps, 56kbps, and 300kbps.

Protocol stack: IEEE 802.11b PHY (message retraining modem capture), IEEE 802.11b MAC (DCF), ARP, IP, AODV routing.

Arena size: variable; changed according to the number of nodes simulated to maintain constant density of 7 neighbors per node.

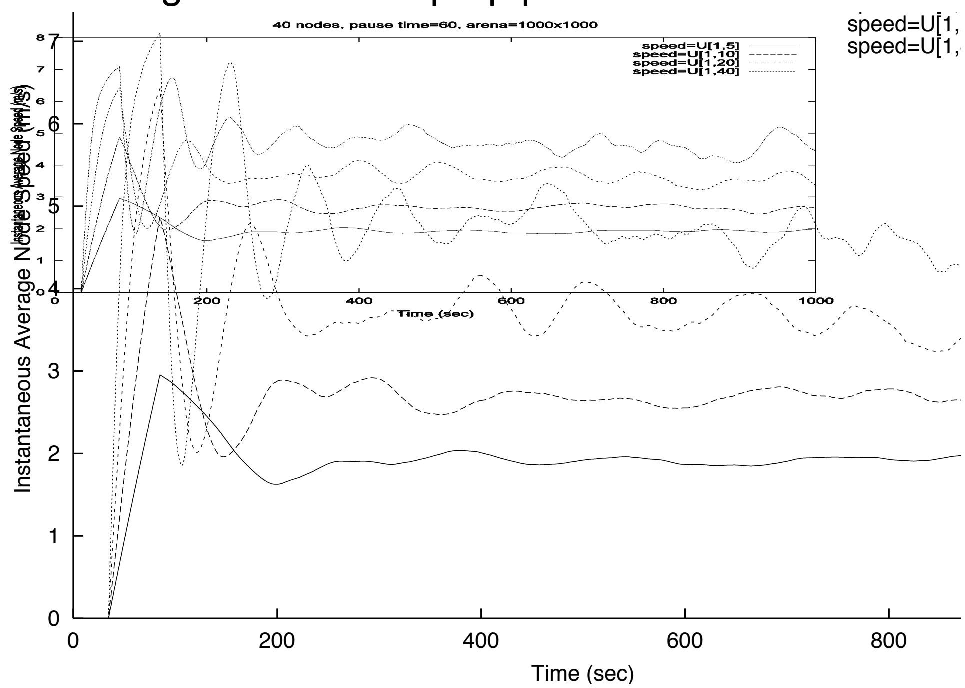
Replications: 10 runs with different seeds for every random stream in the model. For all metrics estimated, we produced 95% confidence intervals.

Scale: 20, 30, 40, and 50 nodes.

Case study: mobility model

Yoon et al. "**Random waypoint considered harmful**". *INFOCOM 2003*.

- Demonstrates how a bad choice of parameters can lead to a mobile network that tends to become stationary (no steady state).
- Called out attention to the fact that the vast majority of simulation studies with wireless networks ignores the ramp-up period in their sub-models.



The impact of mobility transient on network metrics

We verified that using data deletion to avoid the mobility transient led to significant changes in relative error:

- from 5% to 30% in packet end-to-end delay,
- from 5% to 30% in the ratio of data to control packets sent,
- up to 10% in packet delivery ratio.

Interesting results with algorithms for estimation of when steady-state is reached were presented yesterday at WSC ’03:

Bause & Eickhoff. “**Truncation Point Estimation Using Multiple Replications in Parallel**”.

PS: Our paper shows that transients due to the ramp-up effect in traffic, further compromise the correctness of network metrics.

Lesson learned

The simulation framework should be flexible enough in the collection of statistics to allow for data deletion.

All the statistics we collect are stored in data types derived from a base class that takes truncation point in time as a parameter. Only the values recorded after the truncation point are kept.

In our experiments we ran several simulations just to determine the truncation point… Certainly, it would be beneficial to compute the truncation point on the fly, as suggest by Bause and Eickhoff.

Case study: composition of the protocol stack

- Broch et al. "**A performance comparison of multi-hop wireless ad hoc networking protocols.**" Mobicom '98.
  - States that the use of ARP in the protocol stack produces non-negligible effects in the simulation of a wireless network.
  - We found no mention to the use of ARP models in other simulation studies save for one other paper. Our inquisitiveness lead us to attempt to quantify the effect of ARP on the networking metrics our simulation estimates.

The impact of ARP

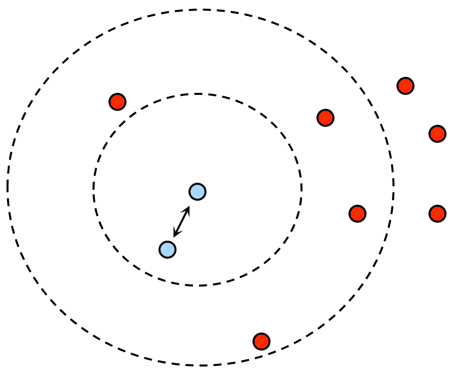
For 16kbps and 56kbps traffic loads, the relative error in end-to-end delay observed was as high as 16%.

Packet delivery ratio showed much less pronounced sensitivity: relative error went only as high as 1.6%.

The number of events in simulations with and without ARP we observed is comparable. The protocol contributes to the simulation with small processing load, and also with small additional memory requirement.

Case study: radio interference model

A common approach to reducing the complexity of interference computation is to limit, or truncate, the sensing range of a node. This range can be defined by a maximum path loss parameter. We have investigated two values: 106dB and 126dB.



For a given node, we can define a receiving range and a sensing range.

Results were consistent with what has been observed in the simulation of wireless cellular phone networks (Liljenstam & Ayani '98; Perrone & Nicol 2000):

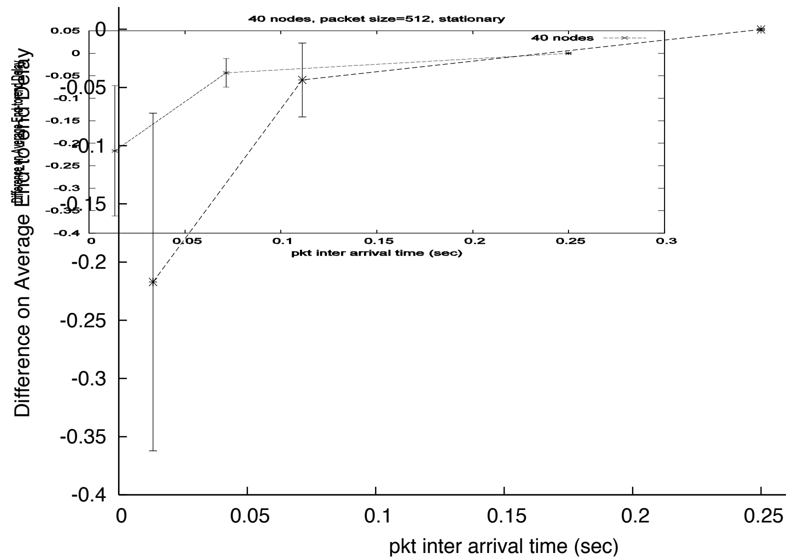
- truncation leads to a substantial reduction in number of events to process at the cost of a small relative error in network metrics.

A question of time

How long does one need to run a simulation in order to produce good estimates of the network metrics?

We have run simulations of 1000s after 500s of warm-up for mobility and traffic generation models. This choice, however, has proved to be insufficient to avoid problems…

At high-traffic loads, due to contention and interference, the estimates obtained for end-to-end delay exhibit very large confidence intervals indicating that a higher number of samples should have been taken.



Summary of lessons learned

Make an effort to get to know what is under the hood of the simulator. Assuming that every tool has been created by all knowing experts has high risks. Look for hard-coded parameter values.

Question and analyze every single parameter choice. Blindly using values that the majority of the studies have used is a temerity.

Stay true to well-known simulation methodologies for output analysis and work on narrowing those confidence intervals.

Attempt to piece together bleeding edge knowledge about models for wireless network simulations. Since much of the material is new, the pieces of the puzzle lie scattered across the board.

The published paper is not enough. It is necessary to keep a detailed record of the experiments' settings so that they can be replicated and built upon. Perhaps storing this data in a persistent website is the answer.

Work for the future

- Expand this study to provide a more complete analysis of the sensitivity of the simulation to different parameter settings and choices of sub-models.

- Automation of the generation of models for wireless networks: guide the user to build consistent combinations of choices in the parameter space.