STRAW - An integrated mobility & traffic model for vehicular ad-hoc networks

David R. Choffnes &
Fabián E. Bustamante
Department of Computer Science,
Northwestern University

www.aqualab.cs.northwestern.edu

C3 - Car-to-Car Cooperation

- An opportunity
  - Computers everywhere - ~100 in a BMW 7x
  - Increased interconnectivity (adv. in wireless communication)
  - A large & well spread platform – growing vehicle population
  - Good & accessible location information

Distributed systems based on car-to-car cooperation

- Some example applications
  - Traffic advisory
  - Mobile sensor network for recognizance
- A few key properties
  - Self-organizable & infrastructure independent
  - Natural scalability
  - Highly resilient to failure

A challenge for experimenters

- C3 – a mobile ad-hoc network over vehicles
  - Infrastructureless networks (ad-hoc networks)
  - Nodes act as routers finding/maintaining routes to others
  - Nodes are capable of movement & can be connected dynamically in an arbitrary manner (MANETS)
- The challenge - How can we play with our ideas for such systems?
  - Real-world experimentation
    - Currently no test-bed available
    - Hard to explore scalability
    - Classical problem with repeatability

An experimenters' challenge

- Emulation
  - Uses real sw/hw in simulated environment to ensure accuracy
  - Higher scalability, but still limited
- Network simulation (e.g. NS-2, GloMoSim, SWANS)
  - Scalable to large number of nodes
  - Easy to vary system configuration
  - Repeatability
  - …
- Desirable simulation characteristics
  - Close correspondence with real world - trace-based simulation? …
  - Generalizable - should enable a wide range of scenarios
  - Feedback loop - enable self-steering (e.g., traffic advisory)
  - Scalability - interesting problem instances

Outline

- C3 motivation & approach
- Mobility models for MANETs
- STRAW design and implementation
- STRAW performance
- Conclusion and future directions

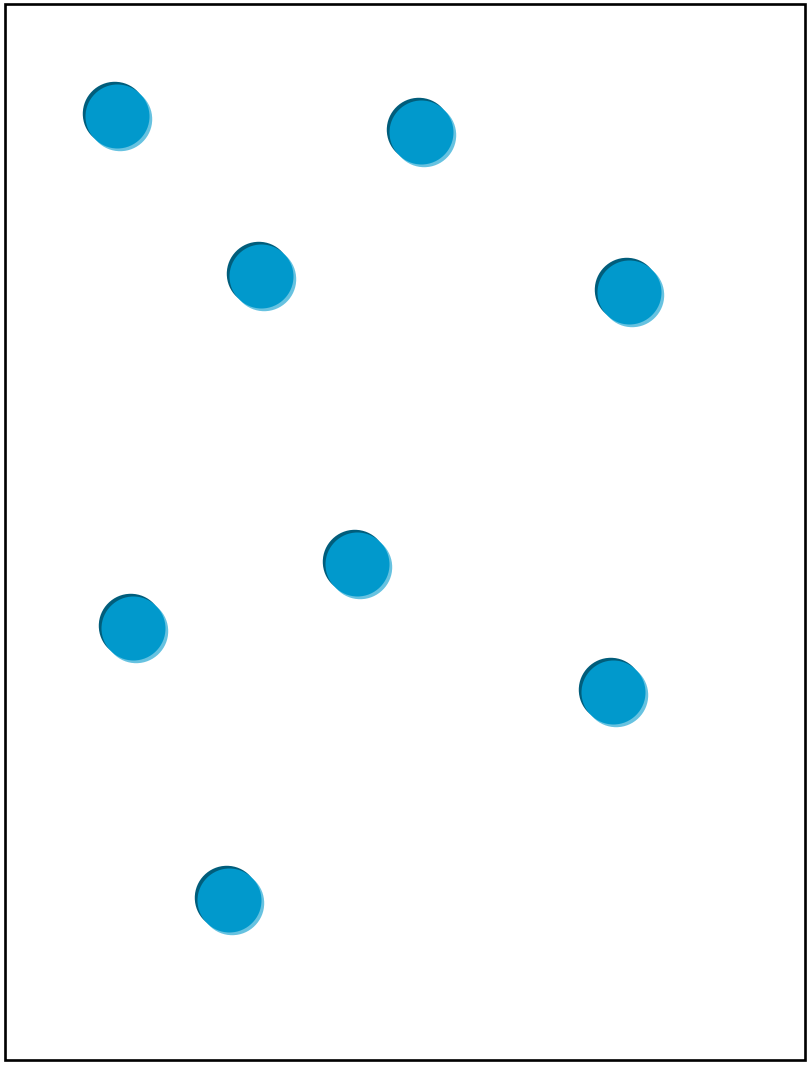
AquaLab

The importance of a mobility model

- **Mobility – key component of MANET simulators & emulators**
  - Mobility constraints (e.g., streets)
    - Affects velocities and distances between cars, which affects radio transmission
  - Nodes should physically interact with one another
    - E.g., avoid collisions
  - Central to “feedback loop” in many scenarios
    - Cars can change trajectory in response to data
  - What’s commonly used?
    - Random waypoint, Mobility traces, ...

Random Waypoint considered harmful

- Random Waypoint (RWP)
  - Benefits
    - Simple
    - Common
    - Low overhead
  - Disadvantages
    - NOT representative of mobility for worst-case or general-case performance
    - Nodes cannot interact wrt mobility
    - Encourages use of open field simulation



AquaLab

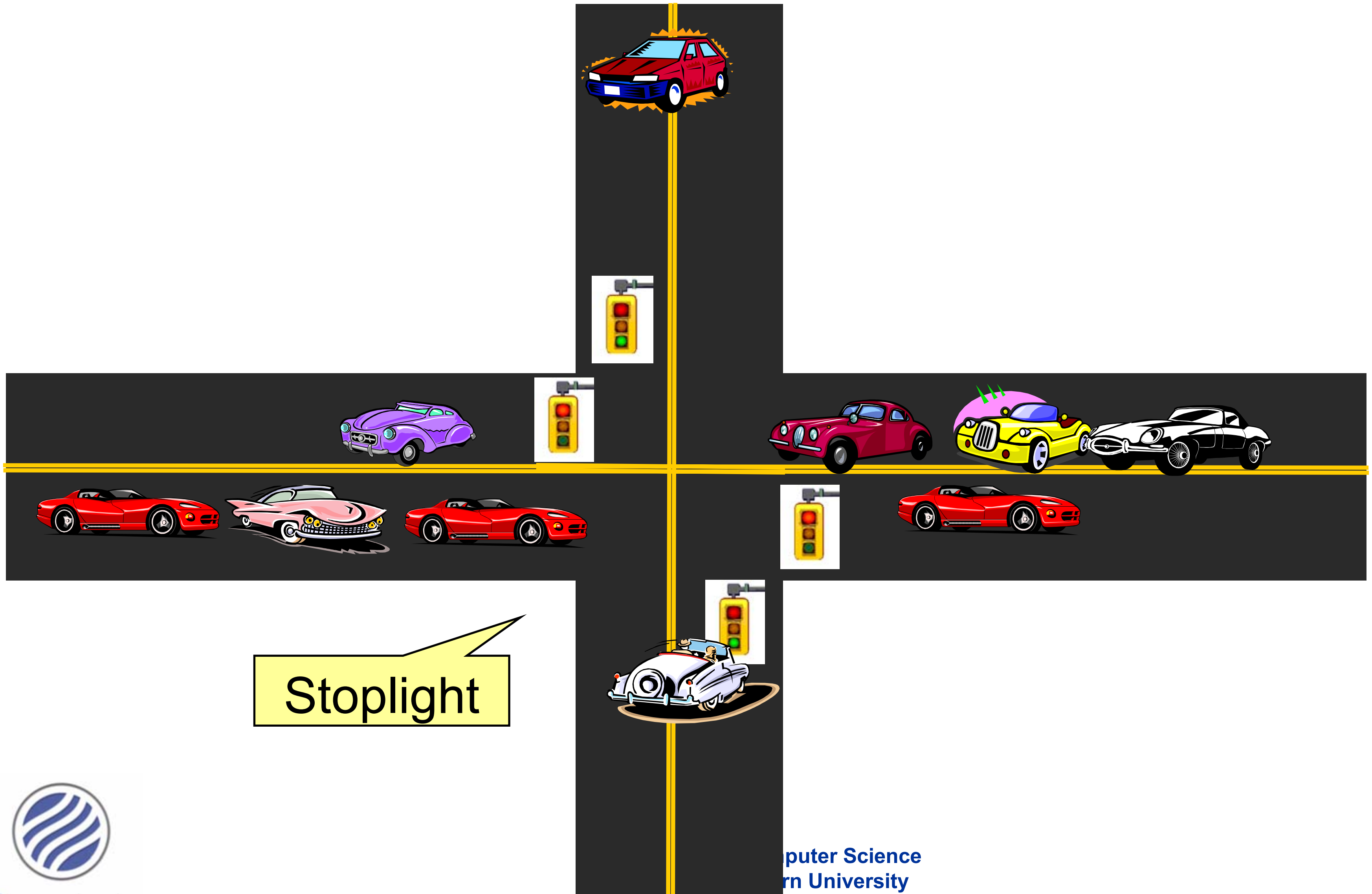
RWP effects on wireless communication

- Every position on map is a waypoint with equal probability
  - Average number of neighbors is relatively uniform over the field
- Nodes generally cannot leave the field
  - Routes generally live longer
- Arbitrary stopping points and stopping times
  - Affects route lifetimes arbitrarily
- Arbitrary speeds and speed distributions

Mobility traces

- Advantages
  - Represents real motion
  - Little overhead in simulation
- Disadvantages
  - Difficult to obtain
  - Rarely distributed (legal issues)
  - Difficult to generalize
  - Does not allow feedback loop

Vehicular motion



Car mobility & wireless communication

- Nodes tend to spend more time at intersections
  - Increases interference in this region
  - Increases number of unconnected pairs
- Buildings further reduce connectivity between nodes on different streets
- Nodes often traveling in opposite or orthogonal directions
  - Short interaction time window
- Vehicular congestion slows nodes
  - Can stabilize topology, but can reduce overall connectivity
- We need a new mobility model: STRAW

Outline

- C3 motivation & approach
- Mobility models for MANETs
- STRAW design and implementation
- STRAW performance
- Conclusion and future directions

AquaLab

STRAW (Street Random Waypoint)

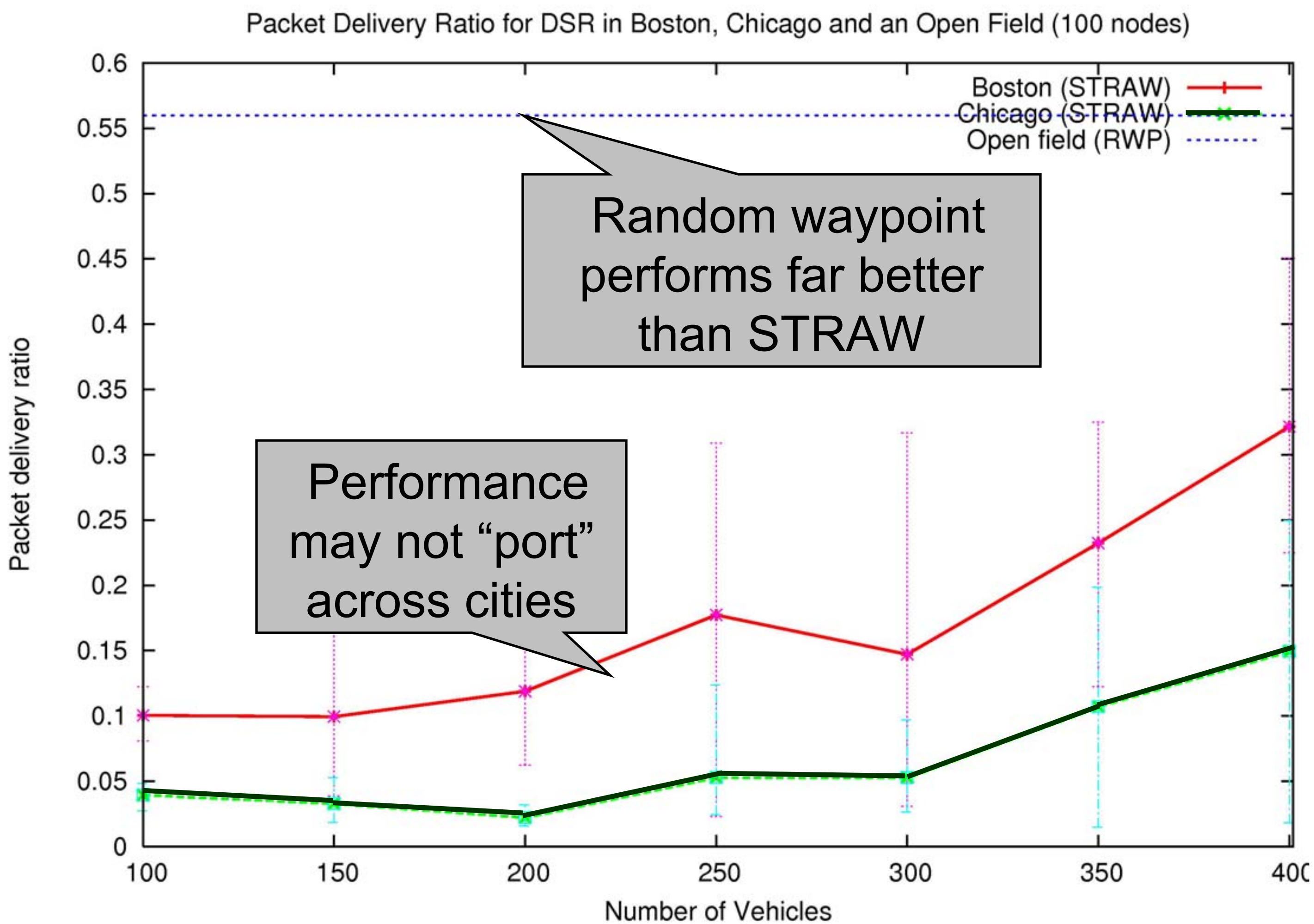
- Node movement incorporates
  - Car-following model
  - Speed limits
  - Traffic control
  - Multiple lanes
- Loads free map data for entire US (easily extended to load data from other map sources)
- Low overhead
  - Insignificant for “simple” model
  - Bounded by vehicular capacity of region for shortest-path origin-destination (OD) calculations
- Easy to configure

Basic abstractions for vehicular motion

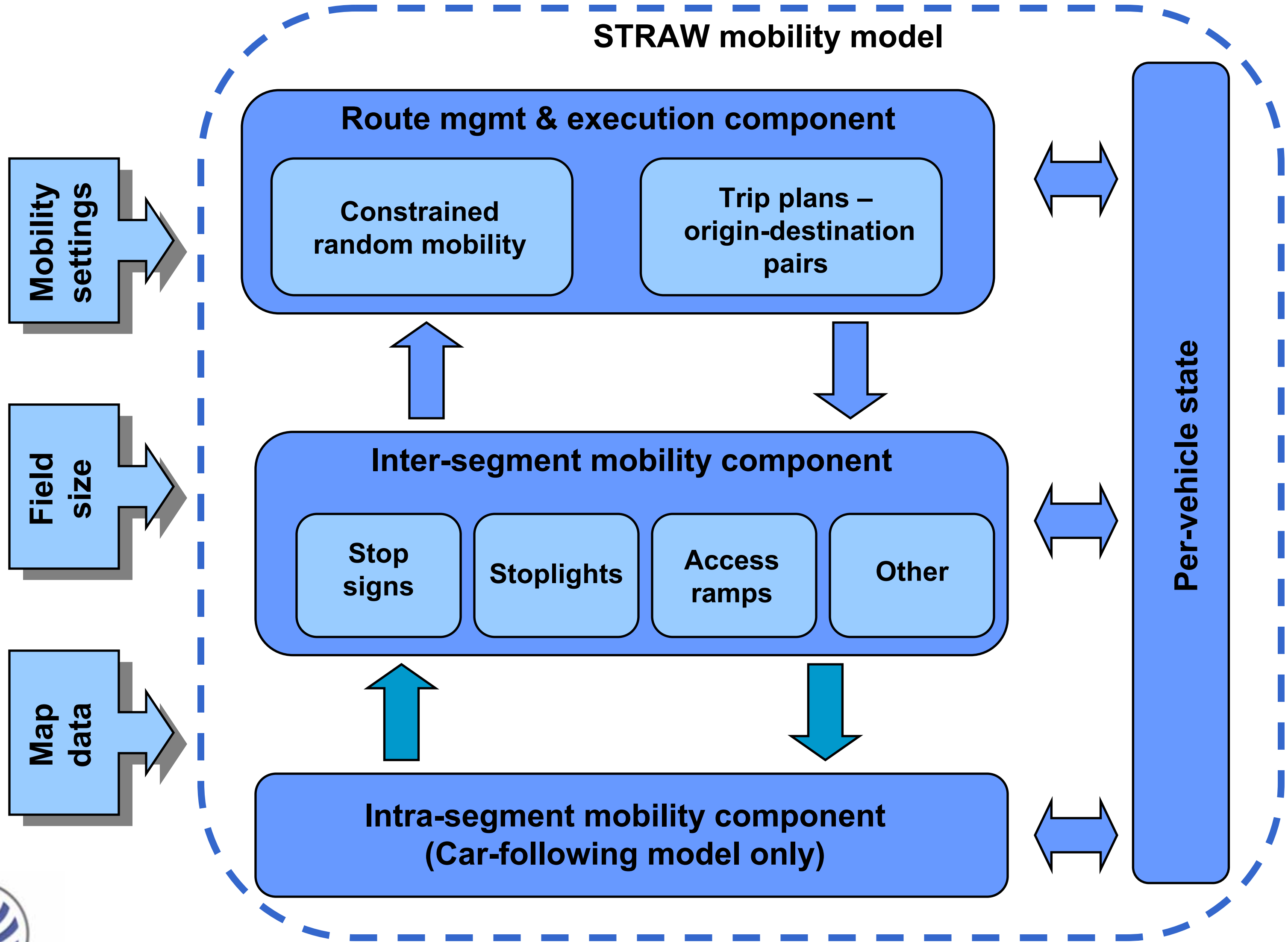
- Vehicle (containing finite length)
- Road Segment
  - Shape
  - Length
  - Width
  - Name
  - Average maximum speed (speed limit)
  - Class
  - Address
- Intersection

AquaLab

Some implications of STRAW



STRAW organization



STRAW initialization

- Loads road segments, names & shapes for rectangular region
  - Organized into a quad tree, with intersections at the leaves
    - Intersections contain a list of associated road segments
    - Manage inter-segment mobility
  - Number of lanes, speed limit, traffic control inferred from “road class”
- Nodes placed on random streets & lanes
  - If a node is already in that lane, put new one behind last node in the lane
  - All nodes start with zero velocity

Intra-segment mobility

- Car-following model
  - Speed-based following distance
- Cruising speed
  - Each vehicle moves at a speed distributed about the speed limit for the current segment
- Acceleration/Deceleration
  - Currently linear, can be extended to any curve made available

Inter-segment mobility

- Precondition for all segment changes - there is capacity on the next segment
- Timed stoplights
  - Wait times inferred from road classes
- Stop signs
  - Drivers take turns crossing the intersection
- Highway merge
  - No need to stop if there is room
- Nodes gradually slow down to a stop before the intersection if they cannot cross it

Route management & execution

- Simple STRAW
  - Turn with certain probability
  - Insignificant overhead
- STRAW with OD
  - Pick a series of origins and destinations
  - Overhead scales with size of region
  - Uses efficient A* shortest path search
  - Supports flows of vehicles from an origin to a destination
    - Vehicles removed from communication participation when they leave the map

Outline

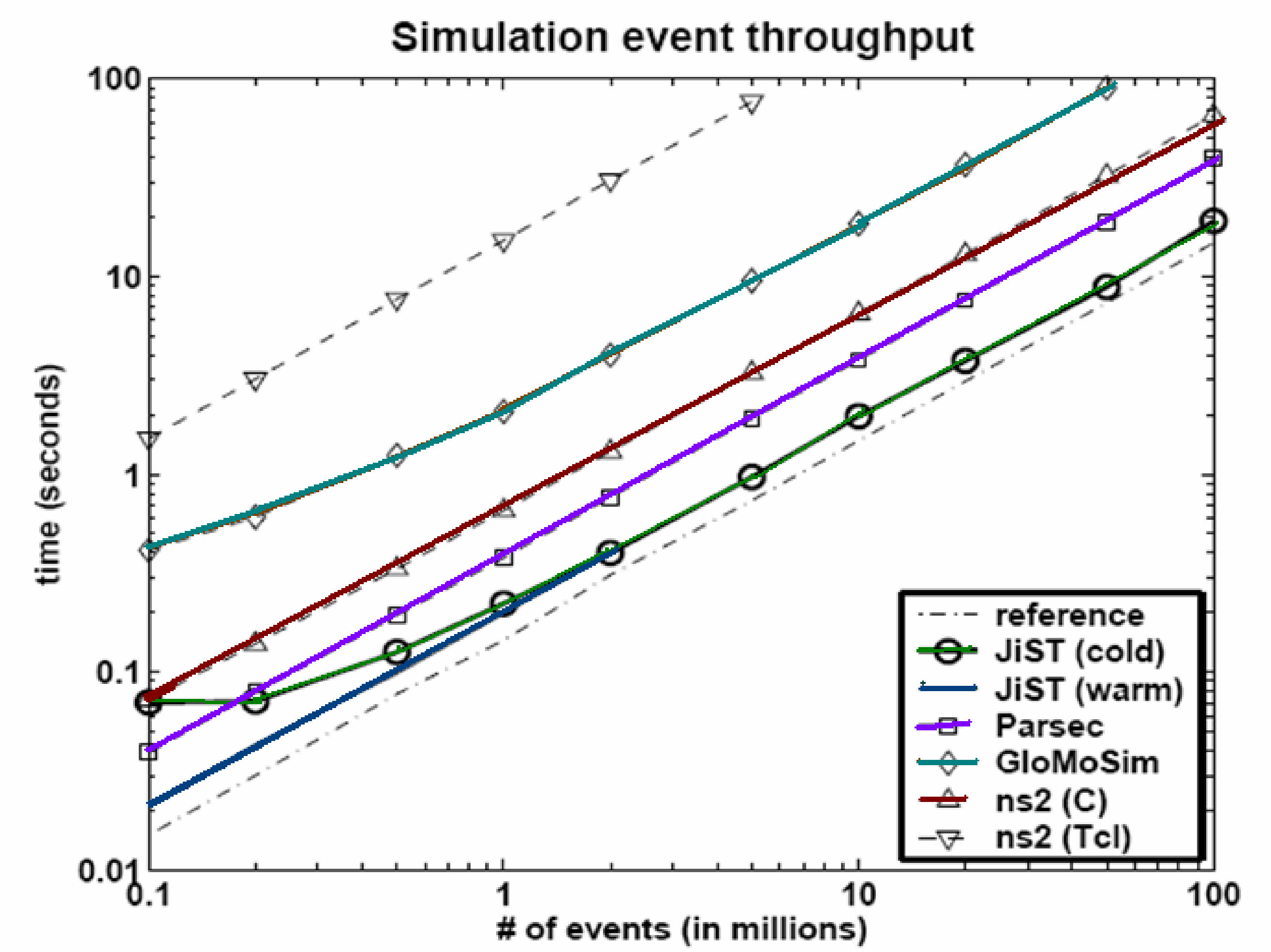
- C3 motivation & approach
- Mobility models for MANETs
- STRAW design and implementation
- STRAW performance
- Conclusion and future directions

STRAW current implementation

- Implemented as part of SWANS (Scalable Wireless Ad-hoc Network Simulator)
- SWANS features
  - Built atop Java in Simulation Time (JiST)
  - Automates porting to Java application code to the simulator
  - Very efficient and scalable discrete event sim
  - Natural programming model
  - Very modular and extensible, supports all important MANET abstractions

A sample of SWANS performance

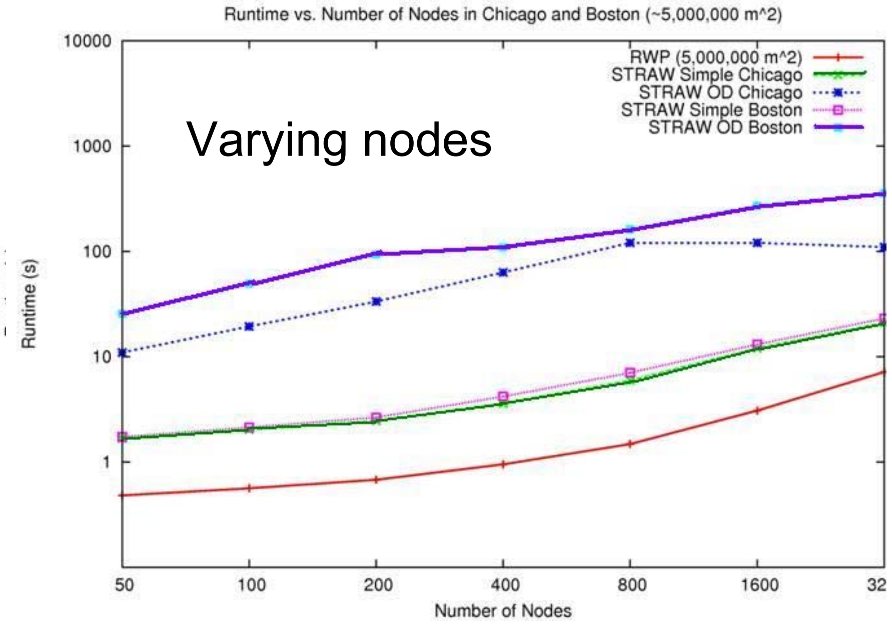
Log-log scale

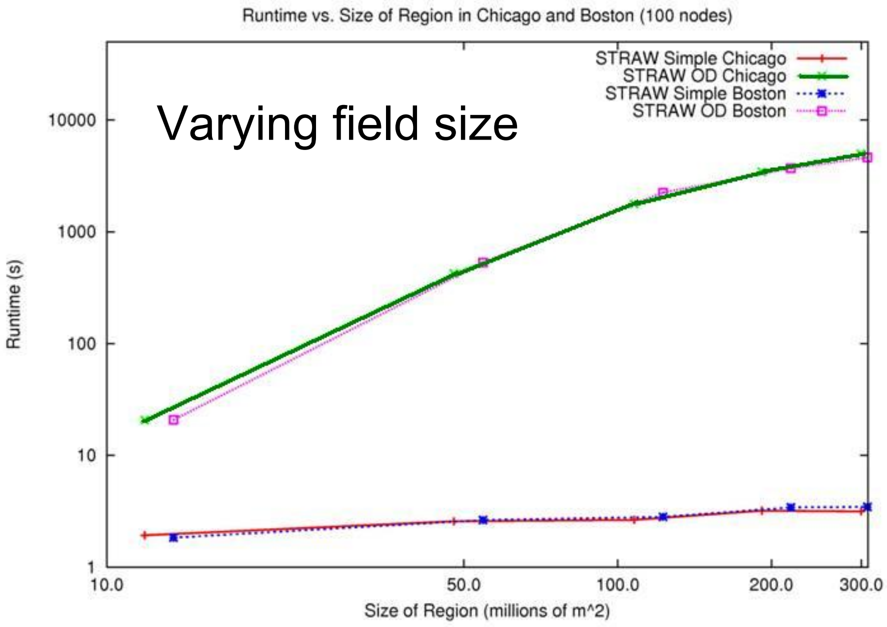


From: BARR, R. *An efficient, unifying approach to simulation using virtual machines*. PhD thesis, Cornell University, 2004.

How much does STRAW cost?

- Reasonable runtime overhead





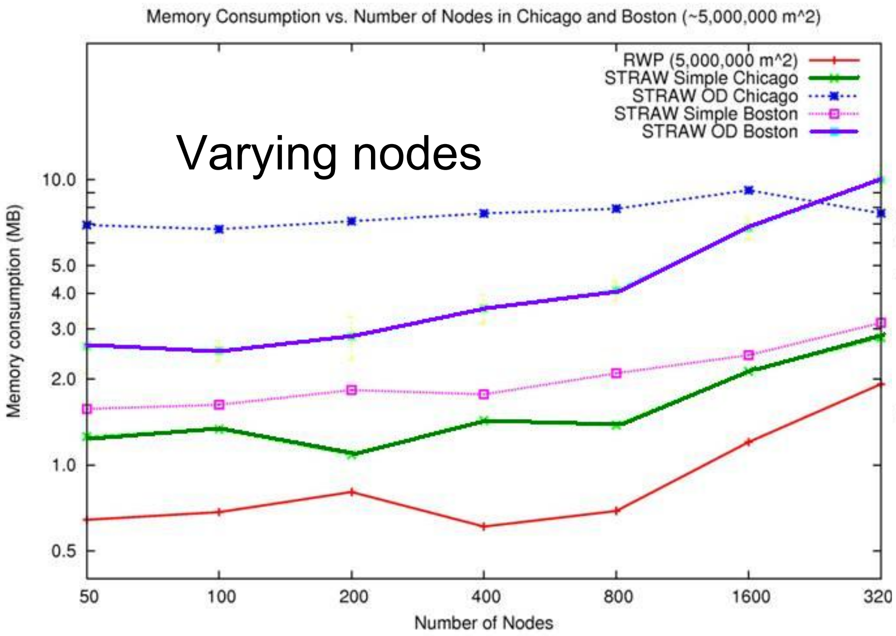
Runtime overhead (log-log scale)

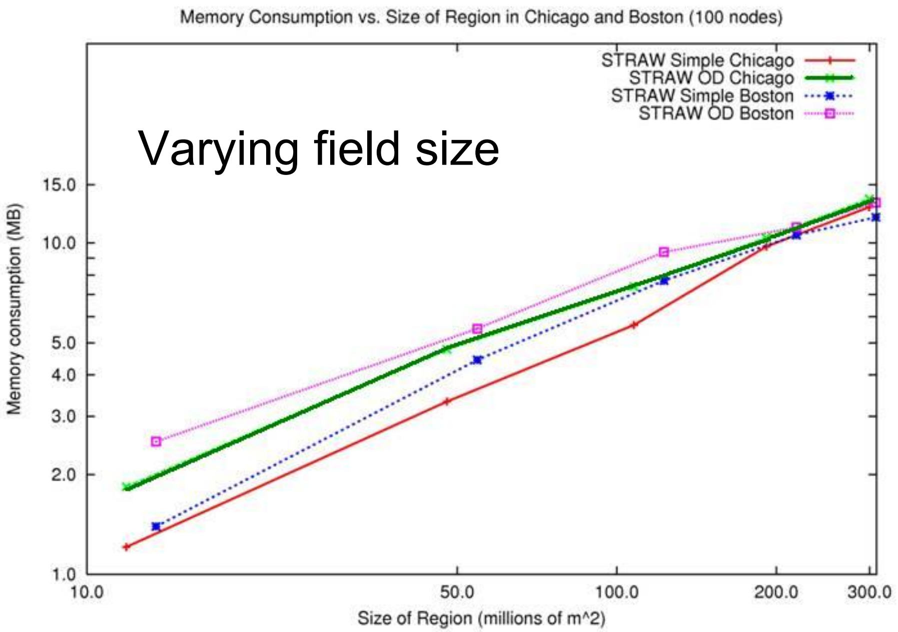
AquaLab

Dept. of Computer Science
Northwestern University

How much does STRAW cost?

- Reasonably small memory overhead





- Greater Chicago’s Cook County ≈ 92MB

AquaLab

Conclusion & Future directions

- Summary
  - Mobility significantly impacts network perf.
  - Performance varies with road plan
  - STRAW models VANET mobility with low overhead
- Future directions
  - STRAW implications on the effectiveness of location-based, mobility-aware communication protocols
  - Middleware for VANET applications
  - Content distribution on VANETs