UbiSim:

A Framework for Simulation of Ad Hoc Networks in Realistic Urban Environment

UBIROADS'09 Workshop - June, 26th 2009

Guillaume-Jean Herbiet <guillaume.herbiet@uni.lu>

Pascal Bouvry <pascal.bouvry@uni.lu>

uni.lu
UNIVERSITÉ DU LUXEMBOURG

Outline

- Context and rationale
- UrbiSim principles and architecture
- Main features
- Future work
- Conclusion

Models for MANET simulation

- Model parameters impacting the network characteristics and performances
- Environmental factors:
  - Propagation model (fading, shadowing, ...)
  - Mobility constraints (pathways, speed limits, physical constraints)
- User-dependent factors:
  - Users destinations and path followed
  - Interaction between users (traffic pattern)

uni.lu UNIVERSITÉ DU LUXEMBOURG

Why this contribution?

- Development of handheld and vehicular communicating devices
  - Network of urban users with characteristic behavior (heterogenous mobility)
  - New interactions as ground for new communication protocols
- Requires a novel approach to mobility modeling/simulation
  - Reproduce the constraints of urban environment
  - See the users behind the equipments
  - Be easily extendable

Overview of current models

- Simple but unrealistic synthetic models [Camp02]
  - e.g. Random Waypoint [Yoon03]
- More realistic models take into account
  - laws of physic (in speed/direction change [Bettstetter01])
  - the environment (constrain movement to streets, presence of buildings [Jardosh03])
  - social behavior [Herrmann03] of user (habits, interests)
- Trace-based models [Koberstein08]

None of them proposes a complete approach

UrbiSim is...

UrbiSim is...

- A generic framework for ad hoc simulation in urban environments

UrbiSim is...

- A generic framework for ad hoc simulation in urban environments
- Extends existing simulators as independent plugin
  - does not rely on any specific command/library
  - work with many C/C++ based network simulators

    (NS-2, OPNET, OMNeT++)

UrbiSim is...

- A generic framework for ad hoc simulation in urban environments
- Extends existing simulators as independent plugin
  - does not rely on any specific command/library
  - work with many C/C++ based network simulators

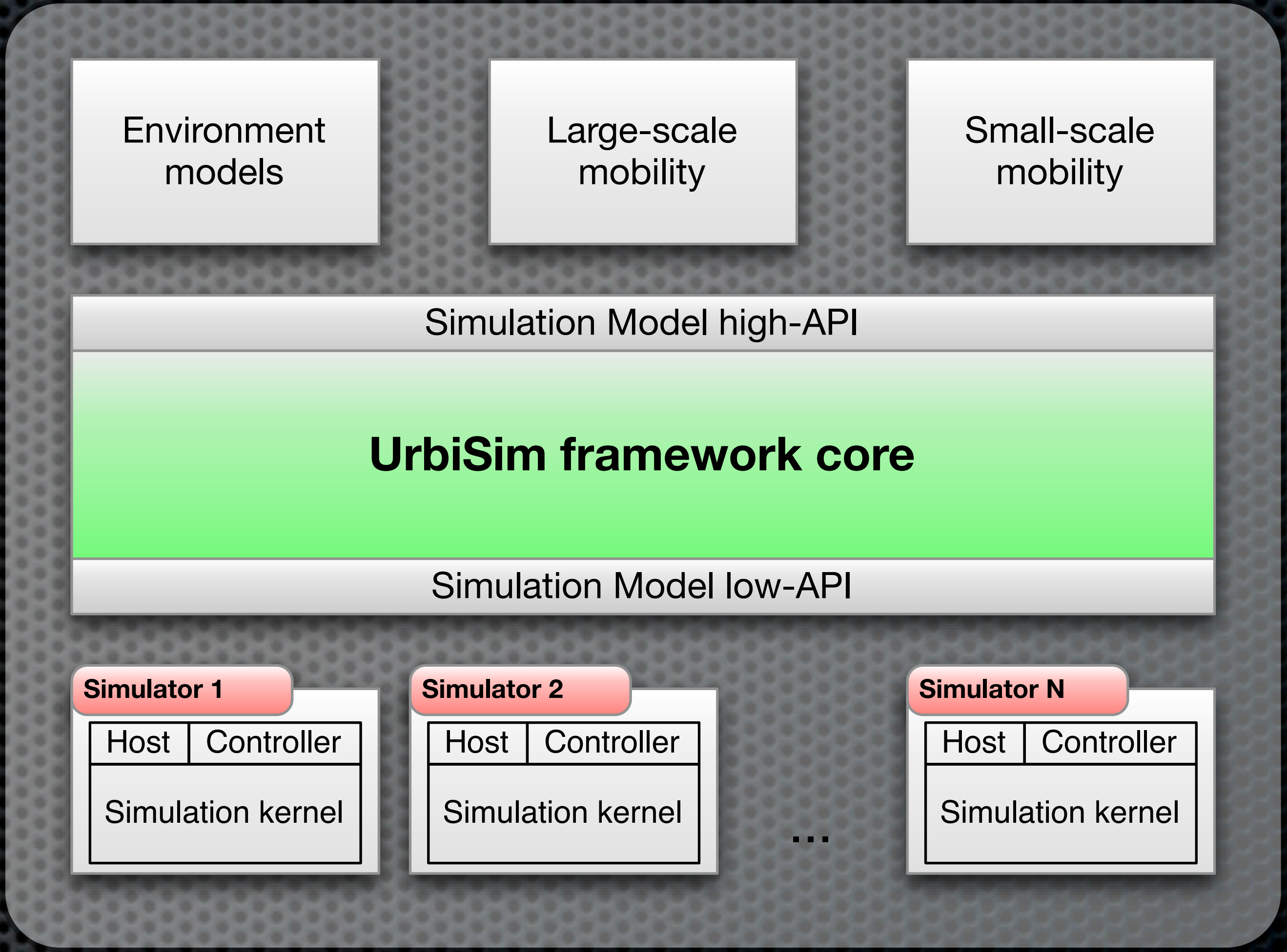
    (NS-2, OPNET, OMNeT++)
- Open source (available on UNI.LU Gforge)

UrbiSim is...

- A generic framework for ad hoc simulation in urban environments
- Extends existing simulators as independent plugin
  - does not rely on any specific command/library
  - work with many C/C++ based network simulators (NS-2, OPNET, OMNeT++)
- Open source (available on UNI.LU Gforge)
- Under development
  - not ready for extensive simulation campaigns so far

UrbiSim architecture

- high-API: interface with
  - Environment models (road types, spots, etc.)
  - Large scale mobility (nodes behavioral profile)
  - Small scale mobility (nodes physical model)
- low-API: interface with
  - Simulators node models (updates position of nodes in simulators)
  - Simulators “controllers” (updates radio links between nodes in simulators)



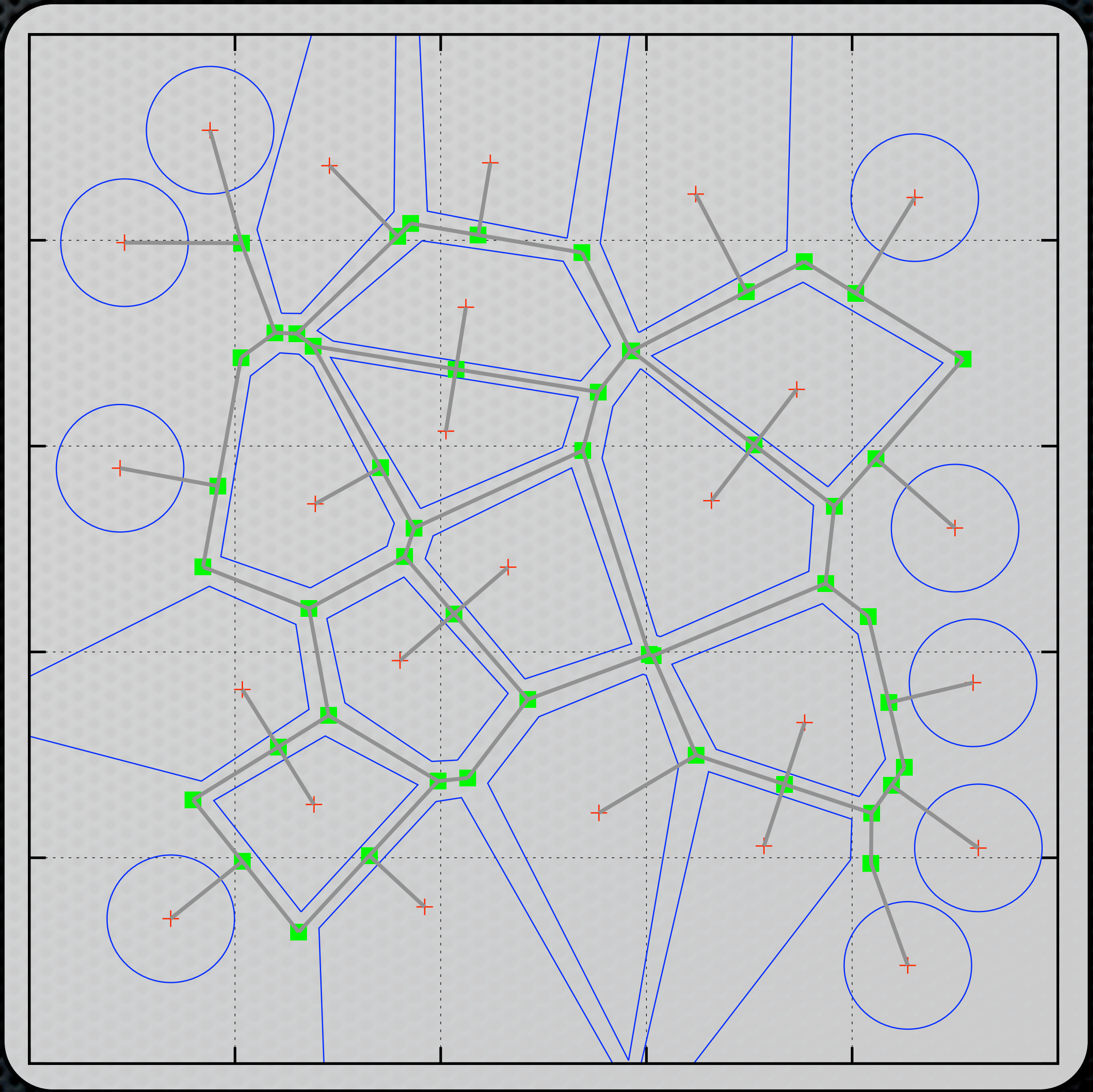
- Framework core:
  - manages all components

UrbiSim principles

- Users move from building (*spot*) to building using streets or *pathways*:
    - User "profile" decides of buildings to visit and entry time
    - Can be based on real world traces
    - Controlled randomness (simulation reproducibility)
- Pathways form a weighted directed graph with streets as edges:
    - Weight is time to traverse a street
    - Shortest path between building is chosen (BGL)
- Users move inside spot, wait and aim for next destination

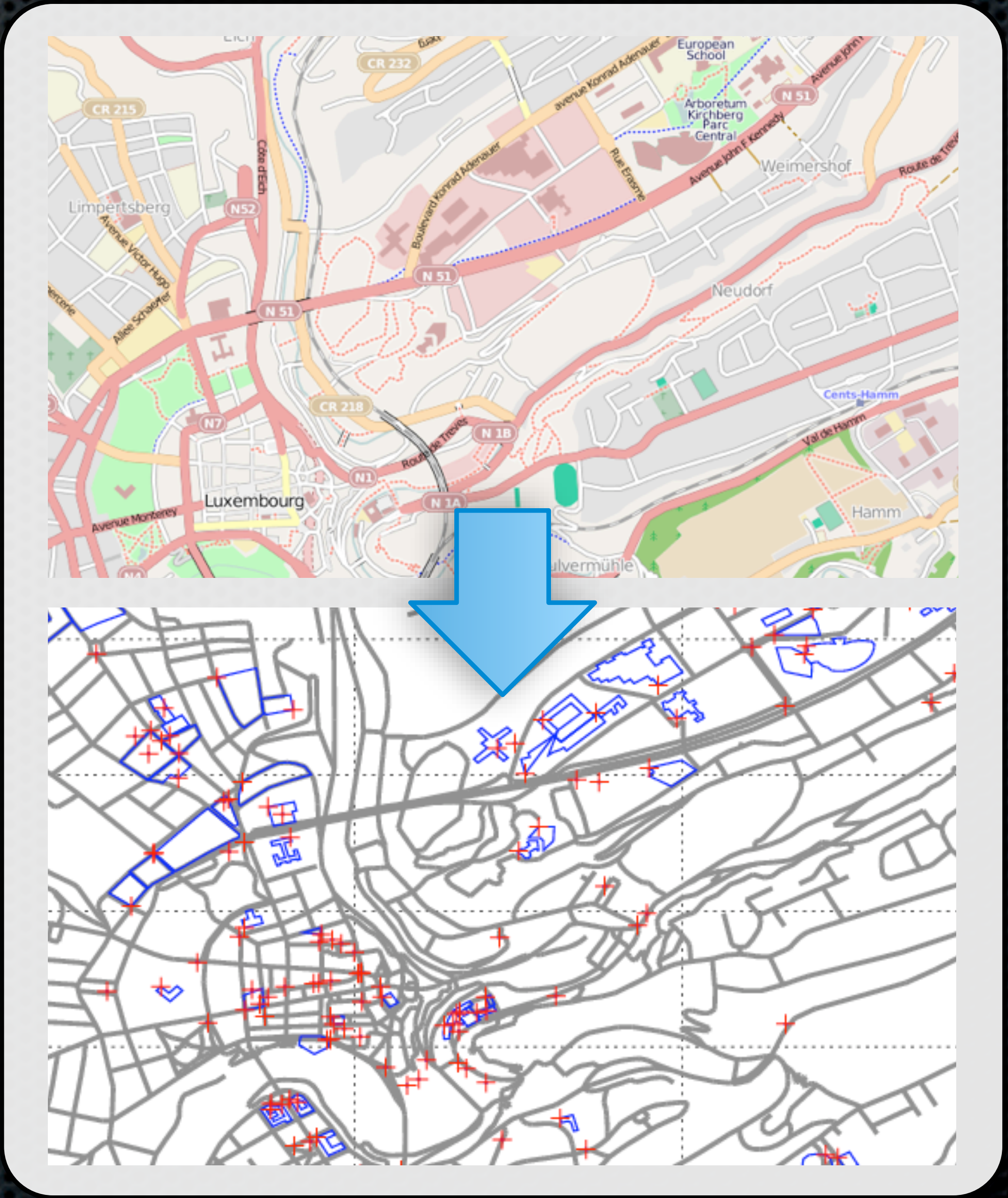
Urban environment generation

- Radom placement of building "centers"
- Generation of streets using Voronoi graph (perpendicular bisectors)
- Buildings fill space between streets with a margin (streets width)
- Buildings can also be round
- Access to "center" from the closest point of the closest street
- Outputs XML config file (compliant with defined DTD) + graphical output



Real-world urban environment

- Translation from OpenStreetMap XML to UrbiSim XML
- Import pathways from streets with specificities (max speed, unidirectional, etc.)
- Import spots from POI with type and shape
- Generation of additional spots (for users home, companies, etc.)
- Extension to usage of city amenities: parkings, bicyle rental stations, bus stops, etc.



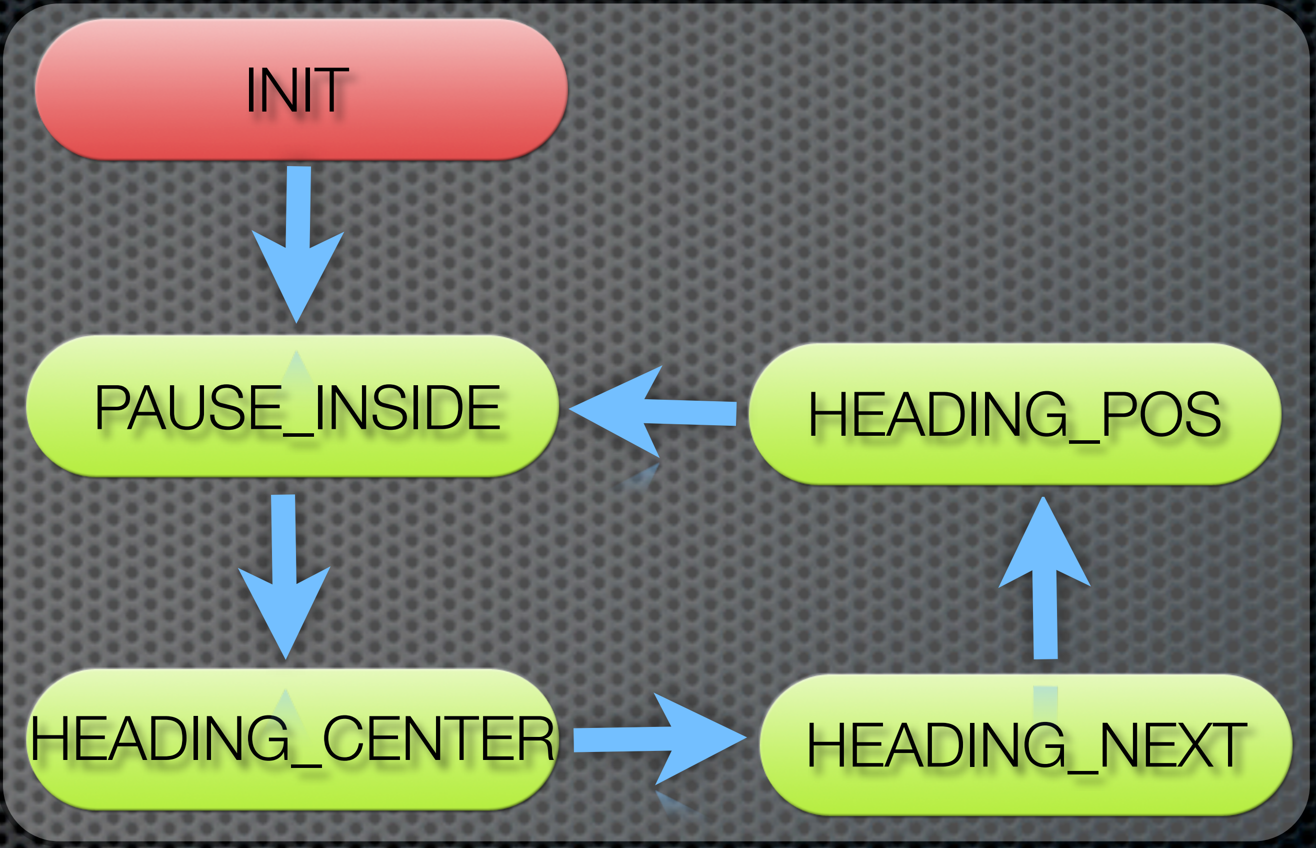
Social behavioral model for nodes

- Create a node profile (XML sub-tree) that will determine the list of visited buildings and the time of visit
  - Due to membership of certain groups
  - Individual preferences
- Mimmic users going to work, home, shopping, visit, ...
- Mimmic users going from building to building for classes, meetings, ...

Heterogenous mobility with long-term interactions

Global (large-scale) mobility

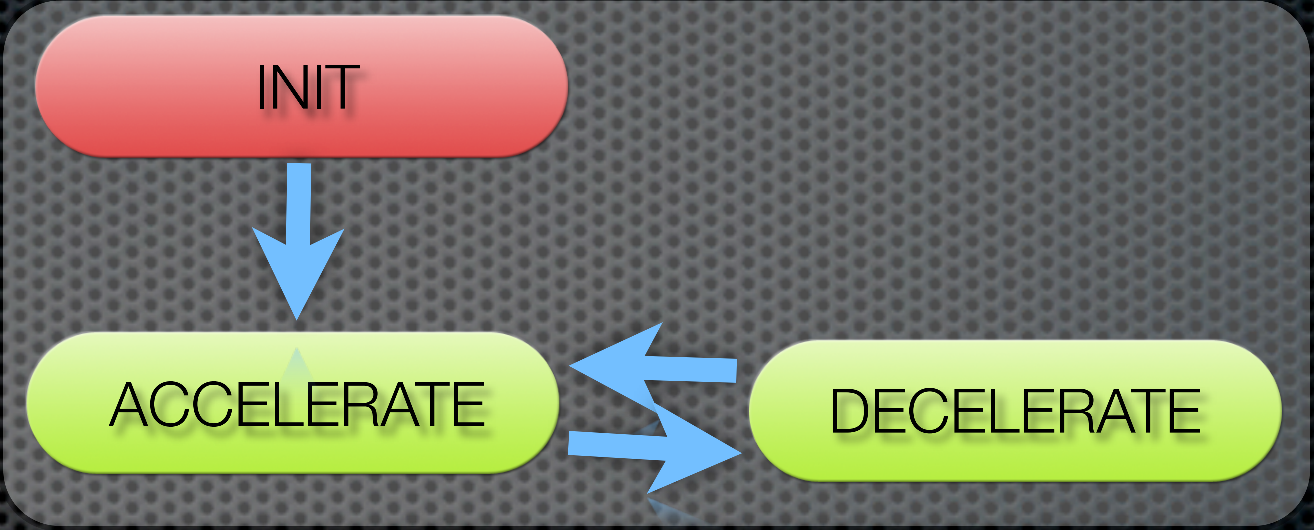
- Governed by a FSM
- Different mobility for in/out buildings
- On streets:
    - Speed on streets depends on traffic congestion
    - Pause time at intersections
    - Speed decay depending on turning angle

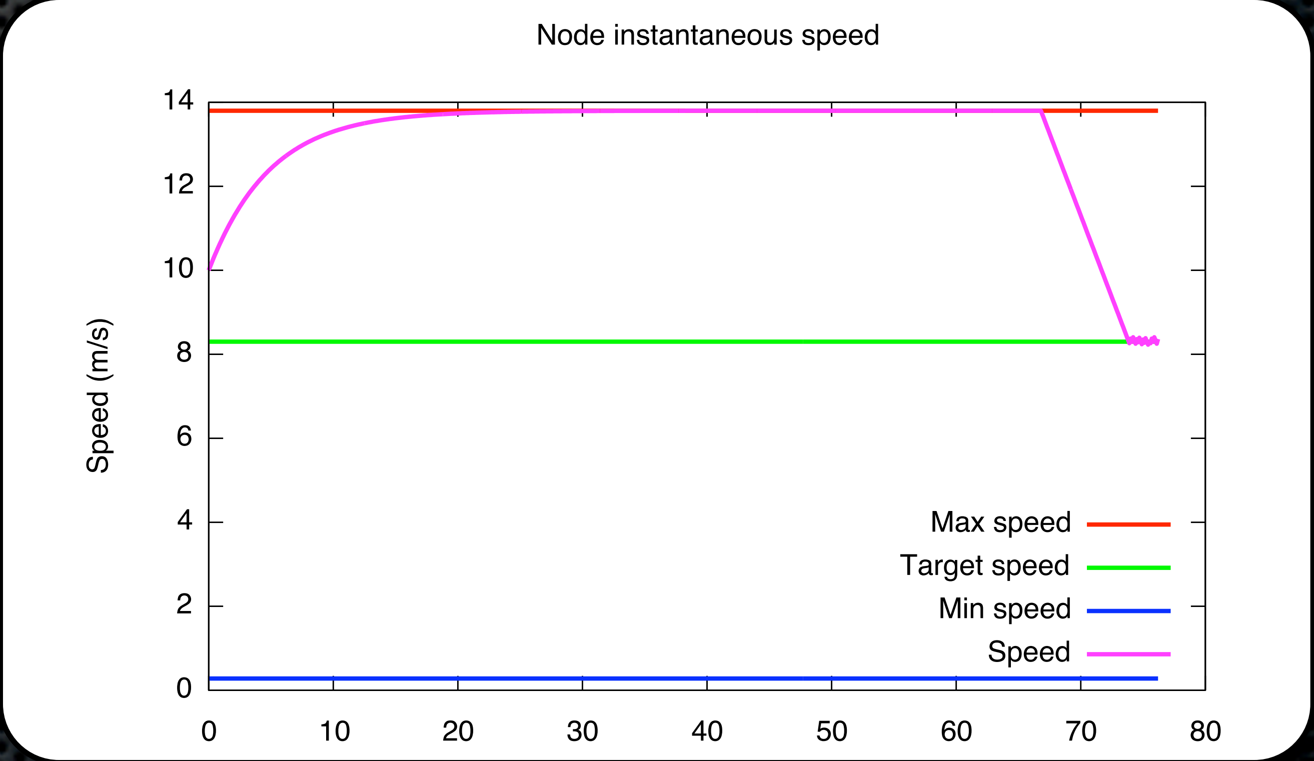


- In buildings:
    - Random walk at walking speed to random position inside building
    - Pause time depends on profile

Atomic (small-scale) mobility

- FSM to manage node speed whatever the global state
- Node accelerates to reach the street max speed
- Node breaks to reach the end-street intersection/turning speed
- Realistic physical speed models:
  - eg. exponential acceleration/linear braking
  - user are attached a small-scale model based on their vehicle type





Future work: Multimodal transportation

- Personal multimodal transportation:
  - Switch between car/bicycle/walk based on distance, fastest alternative, urban constraints (pedestrian zones)
  - Refine the path computation algorithm and use more information from OpenStreetMap database
- Public multimodal transportation:
  - Introduce buses, trains with dedicated or shared pathways
  - Refine the path computation algorithm and use more information from OpenStreetMap database
  - Introduce new “collective mobiles” with specific mobility pattern and “waiting points” (bus stops, stations, etc.)

Future work: Environment-aware propagation model

- So far, propagation model is simple free-space
  - Another important but different aspect of the model
- Add a log-normal shadowing component [Hekmat06]
  - Go beyond the circular transmission radius
- Include buildings in propagation model
  - Different attenuation to model density/material
  - Propagation matrix depending on node situation (in, out)

Need to investigate impact factor on topology

Conclusion

- Realistic modeling of node mobility in urban environment
- Social-aware behavior of the users
- Implementation as natural plugin for main network simulators
- Allows to study heterogenous mobility environments with long-term interactions

Thanks for your attention.

uni.lu
UNIVERSITÉ DU
LUXEMBOURG