Power-aware metrics for adaptive routing in mesh networks

Alexandra Alexeeva, Eugene Kalishenko

Open Source & Linux Lab

http://osll.fruct.org

Outline

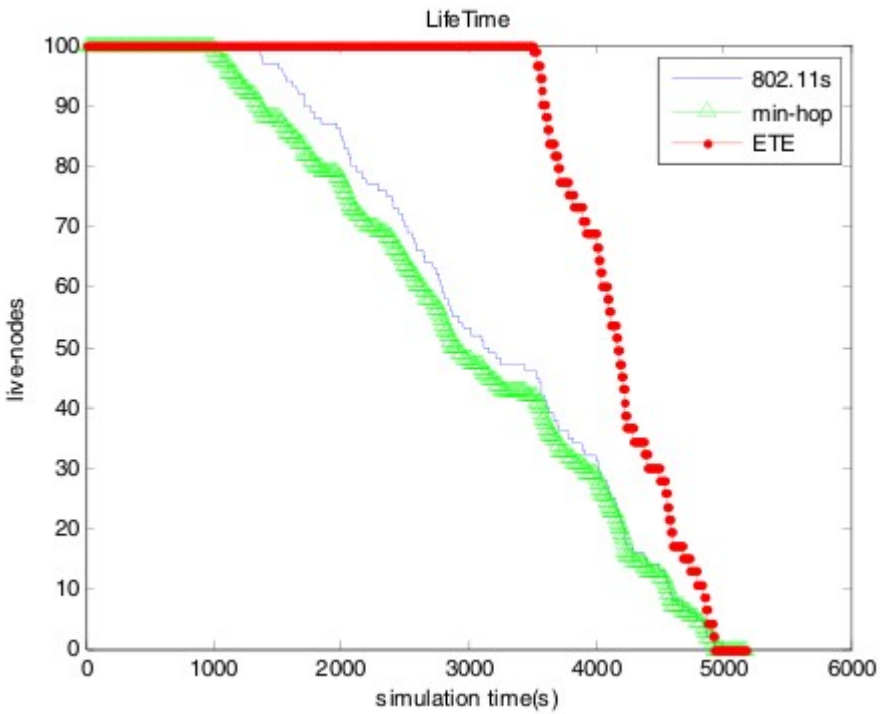
- **Why do we need this?**

  As mobile computing requires more computation as well as communication activities, energy efficiency becomes a critical issue for battery-operated mobile devices.

- **What do we want?**

  The lifetime of the whole net-work to be as long as possible

  That means ideally, all the nodes die sumultaneously.



Problem Statement

- Analyze the existing power-aware metrics for ad-hoc and mesh networks
- Develop an algorithm of power-aware routing using 802.11s features
- Simulate the approach using NS-3
- Implement the battery capacity visualisation support for Netanim (visualizator for NS-3)

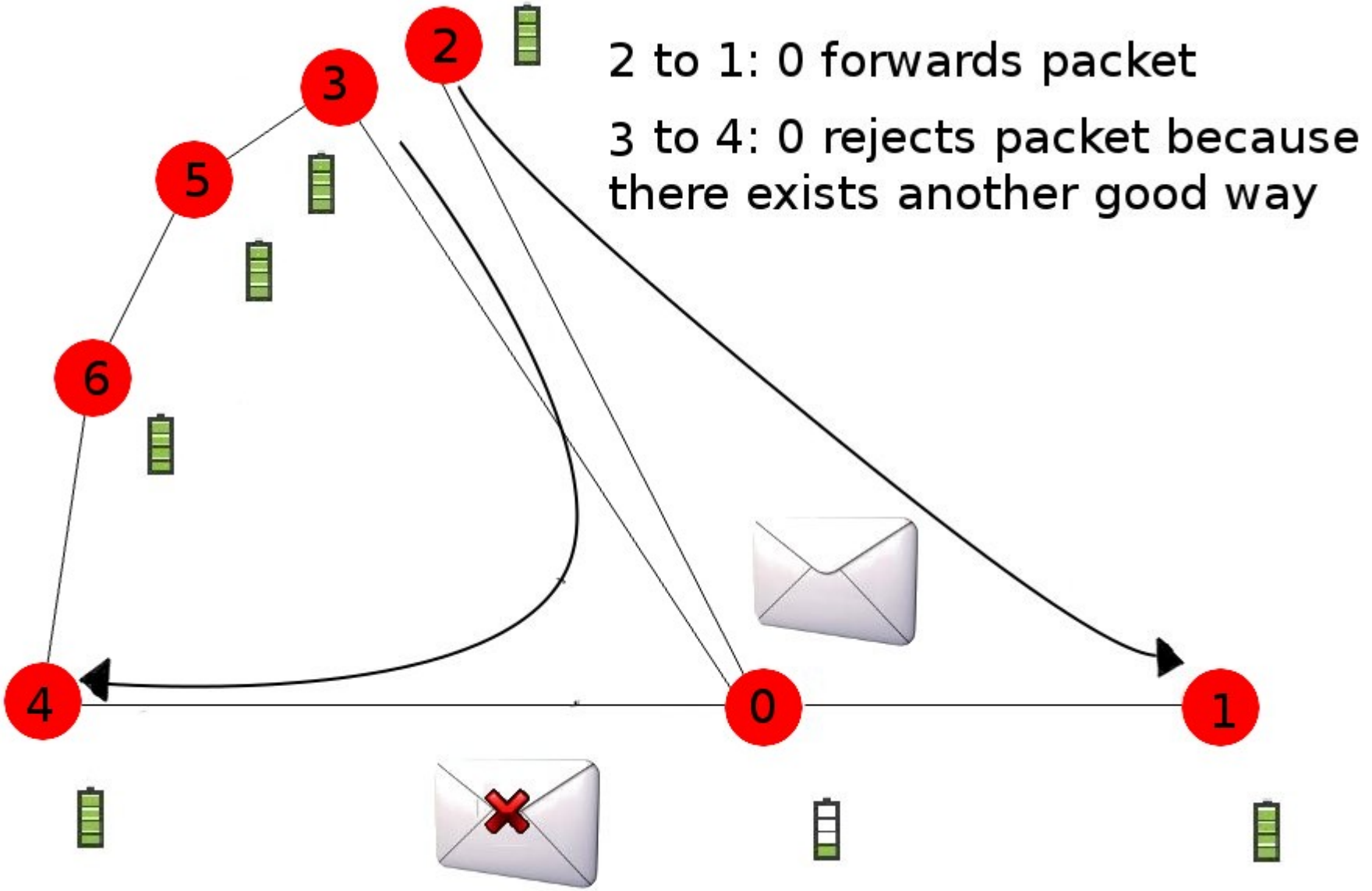
Analysis of existing metrics

The main ideas in the area developed until now:

- Using min-hop metrics
- Using the remaining capacity on host
- Using the speed of energy consumption of a host
- Using complicated physical models of a battery

Some approaches use a barrier to switch between different algorithms.

Mesh standart capabilities



An approach to power-aware routing

We want to use multiple metrics:

- Expected lifetime of a node
- Expected transmission energy
- Expectation of whether the destination node is reachable through other paths

There is an attractive way to combine all that metrics – to use fuzzy routing.

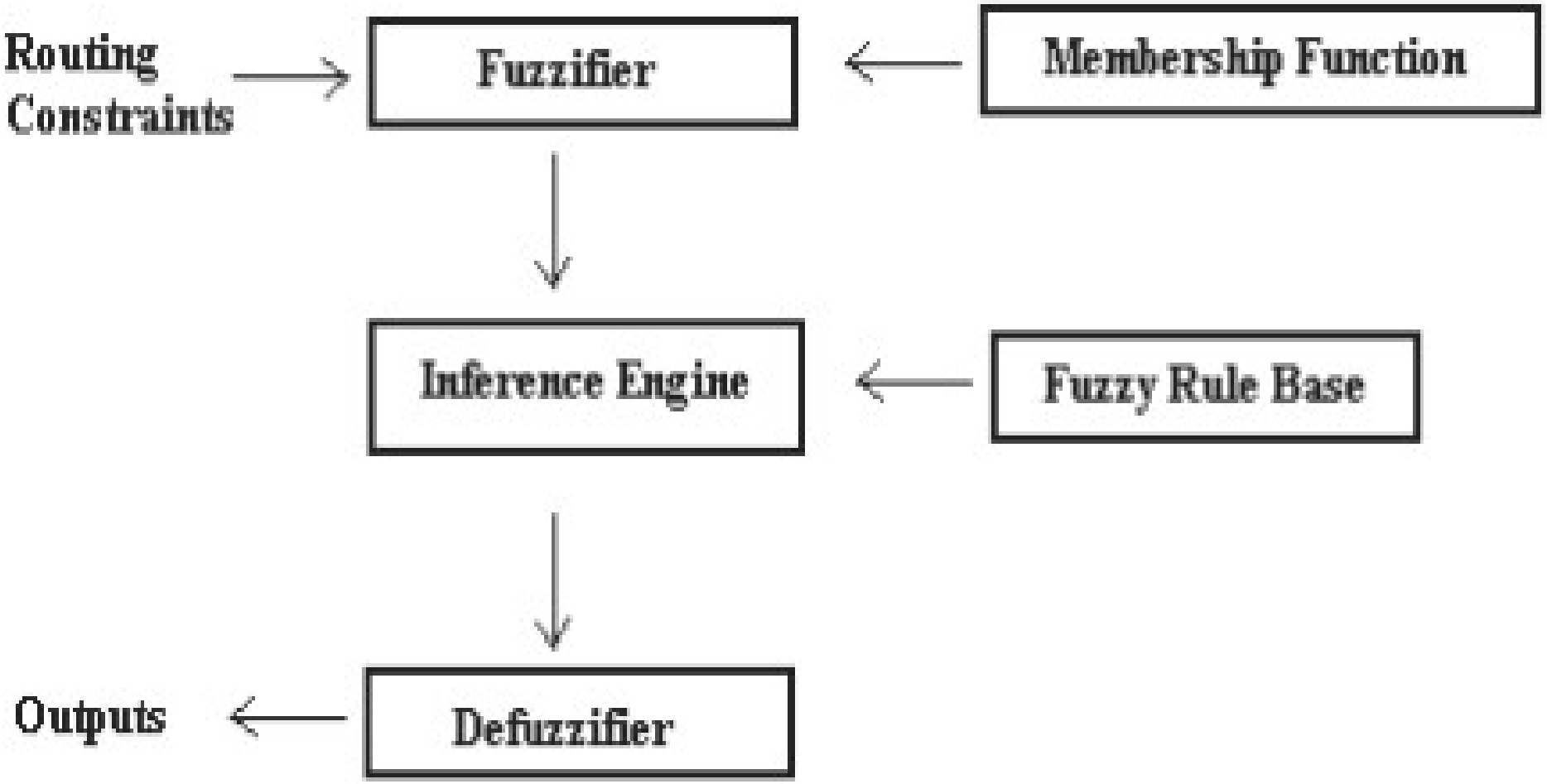
Classical routing

- Router metrics can contain any number of values that help the router determine the best route.
- A router metric is typically based on information like path length, bandwidth, load, hop count, path cost, delay etc.
- Metric is a number that agregates many parameters

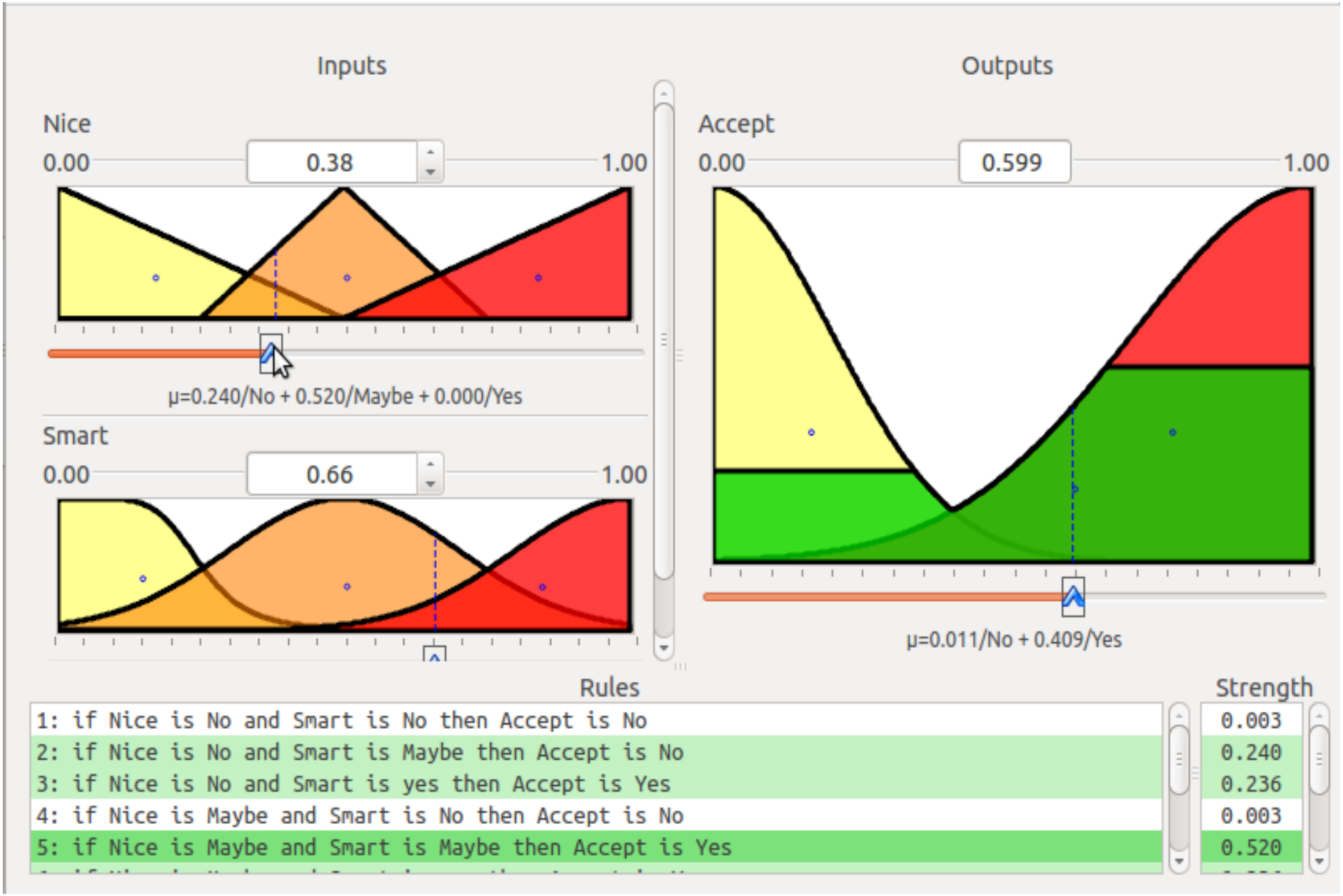
Network Destination	Netmask	Gateway	Interface	Metric
0.0.0.0	0.0.0.0	192.168.0.1	192.168.0.100	10
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1	1
192.168.0.0	255.255.255.0	192.168.0.100	192.168.0.100	10
192.168.0.100	255.255.255.255	127.0.0.1	127.0.0.1	10
192.168.0.255	255.255.255.255	192.168.0.100	192.168.0.100	10

Fuzzy logic

Fuzzy logic is a form of many-valued logic. It deals with reasoning that is approximate rather than fixed.



Fuzzy logic


Inputs
Nice
0.00
0.38
1.00
μ=0.240/No + 0.520/Maybe + 0.000/Yes
Smart
0.00
0.66
1.00
Outputs
Accept
0.00
0.599
1.00
μ=0.011/No + 0.409/Yes
Rules
Strength
1: if Nice is No and Smart is No then Accept is No
0.003
2: if Nice is No and Smart is Maybe then Accept is No
0.240
3: if Nice is No and Smart is yes then Accept is Yes
0.236
4: if Nice is Maybe and Smart is No then Accept is No
0.003
5: if Nice is Maybe and Smart is Maybe then Accept is Yes
0.520

Fuzzy routing

Fuzzy routing allows us to use several constraints and correlate them altogether in such way:

– multiple fuzzy coefficients as an output, choose between several scenarios.

– vary the threshold between the constraints dynamically

Rule base

Table 1 Rule base when Neighbor = “Closer”.

NeighborRate		Thermal				
		Empty	**Few**	**Half**	**Almost**	**Full**
Distance	**VeryClose**	VeryGood	VeryGood	Good	Fair	Bad
	Close	VeryGood	VeryGood	Good	Fair	Bad
	StartPoint	VeryGood	VeryGood	Good	Fair	Bad
	Far	VeryGood	Good	Fair	Bad	Bad
	VeryFar	VeryGood	Good	Fair	Bad	Bad

Table 2 Rule base when Neighbor = “Farer”.

NeighborRate	Thermal				
	Empty	**Few**	**Half**	**Almost**	**Full**

Constraints

– Expected lifetime of a node and Expected transmission energy

$$C_i = RBP_i / DR_i$$

$$DR_i = \alpha * DR_{old} + (1 - \alpha) * DR_{sample}$$

RBP – *remaining battery potential*
C_i *defines the lifetime of a host*

DR_i – *drain rate*
$\alpha = 0.3$

- For formulation of a fuzzy model we can use some physical characteristics of a battery.

  – Expectation of whether the destination node is reachable through other paths can be found through analysis of routing tables.No additional traffic needed.

NS-3

NS-3 is an opensource discrete event network simulator with many capabilities.

The general process of creating a simulation can be divided into several steps:

- Topology definition
- Model usage
- Node and link configuration
- Execution
- Analysis (performance, traffic load...)
- Visualization

Visualization architecture

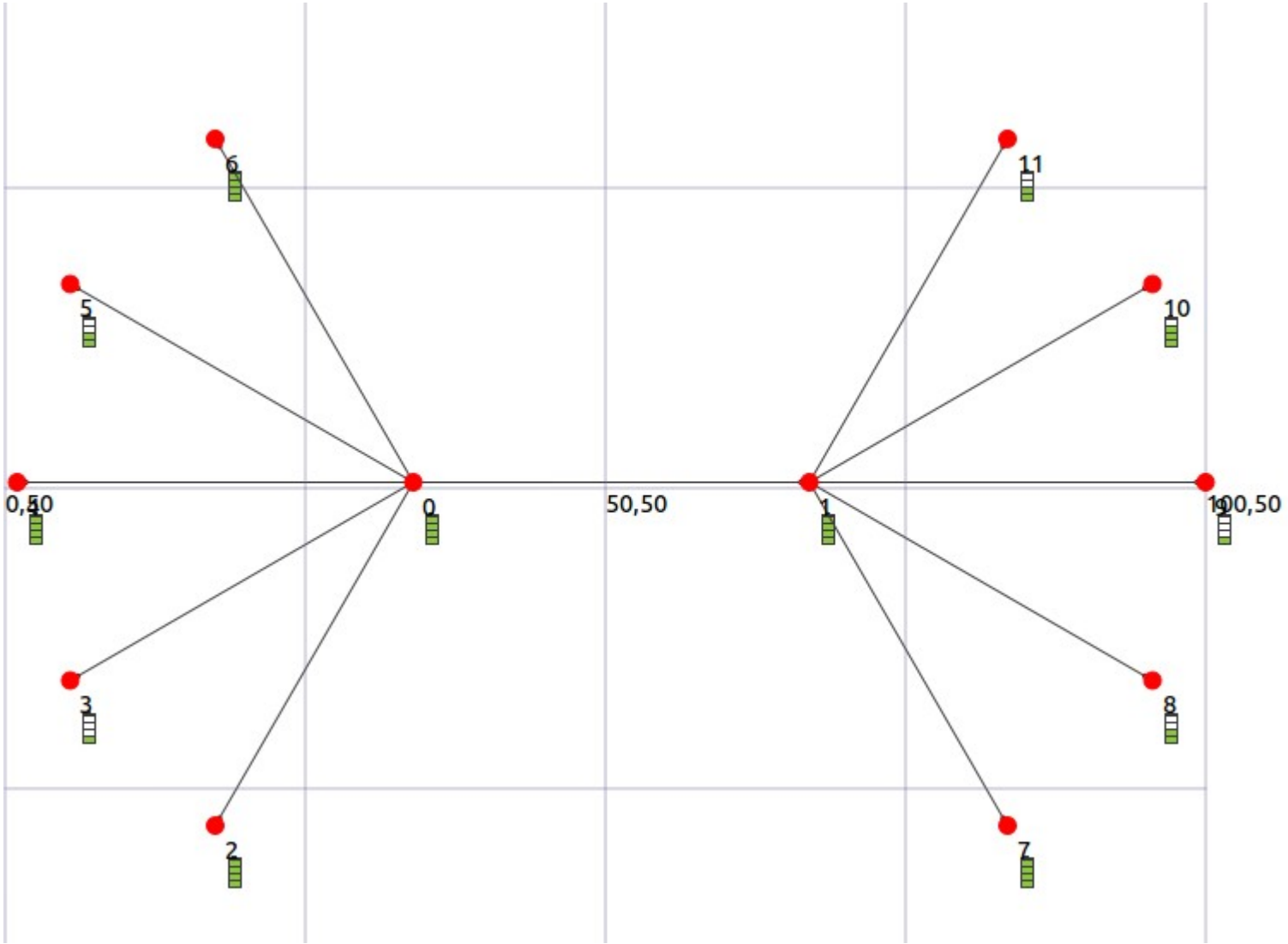
How an NS-3 model can be imported to NetAnim?

- use a typical NS-3 model
- add vizualization functionality (libraries, functions)
- The xml file is got from c++ code of the model using special plugin
- xml file can be opened in NetAnim

How events can be visualized?

- through adding visualization callbacks in model

Battery capacity visualization



Results & Further Work

Current Results

- There are some vague ideas and sketches of a metric
- Battery visualization is implemented
- Analysis of existing power-aware routing algorithms is performed

Further Work

- Decide wich routing type to use
- Finish the mathematical model
- Implement the model in NS-3
- Simulate the metric and analyze simulation results

Questions & Answers

Alexandra Alexeeva
kronshtein78rus@gmail.com
Open Source & Linux Lab,
http://osll.fruct.org, osll@fruct.org