Lightweight network interface selection for reliable communications in multi-technologies wireless sensor networks

Brandon Foubert
Nathalie Mitton
21 April 2021

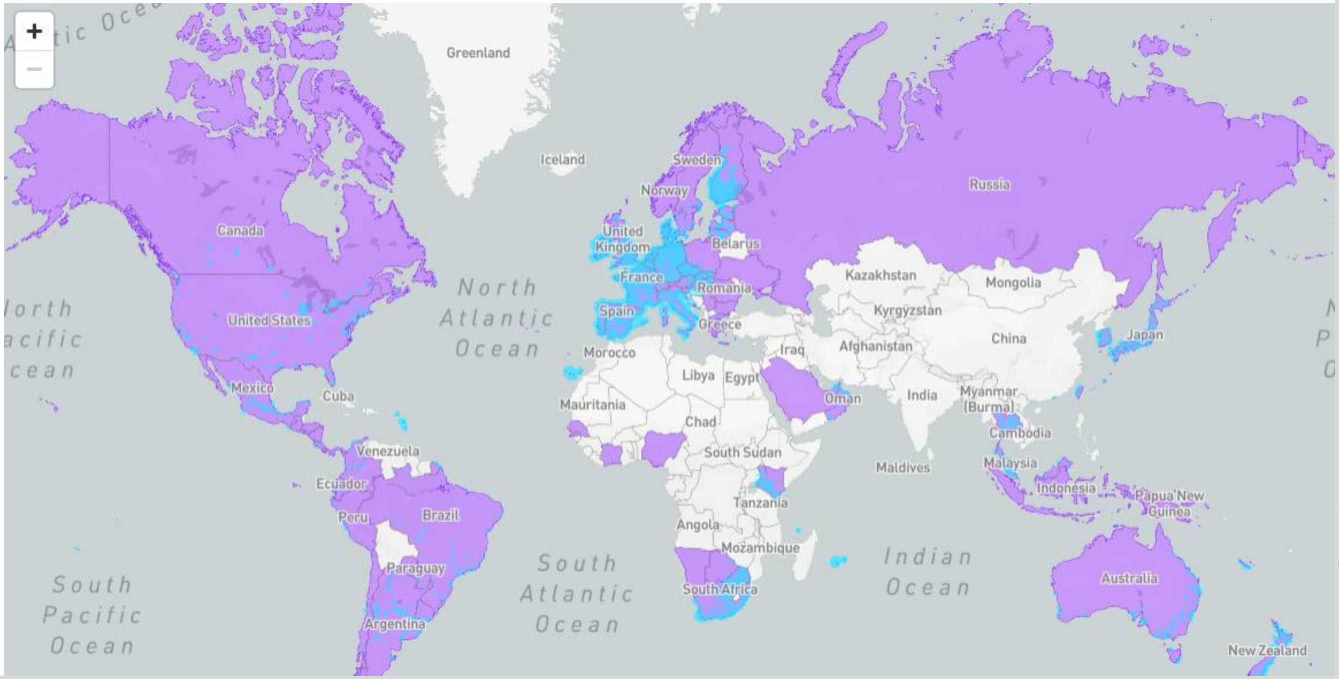
Wireless sensor networks: a tool to help farmers

- Automate the collection of climate data
- Prevent risks (e.g. frost)
- Decision making (e.g. better use of pesticides)

https://www.terre-net.fr/actualite-agricole/economie-social/article/revenus-agricoles-2017-en-hausse-202-133225.html

Each Radio Access Technology (RAT) have limitations, *e.g.* Sigfox

- Max 12 bytes per message
- Max 140 messages per day
- Limited worldwide coverage

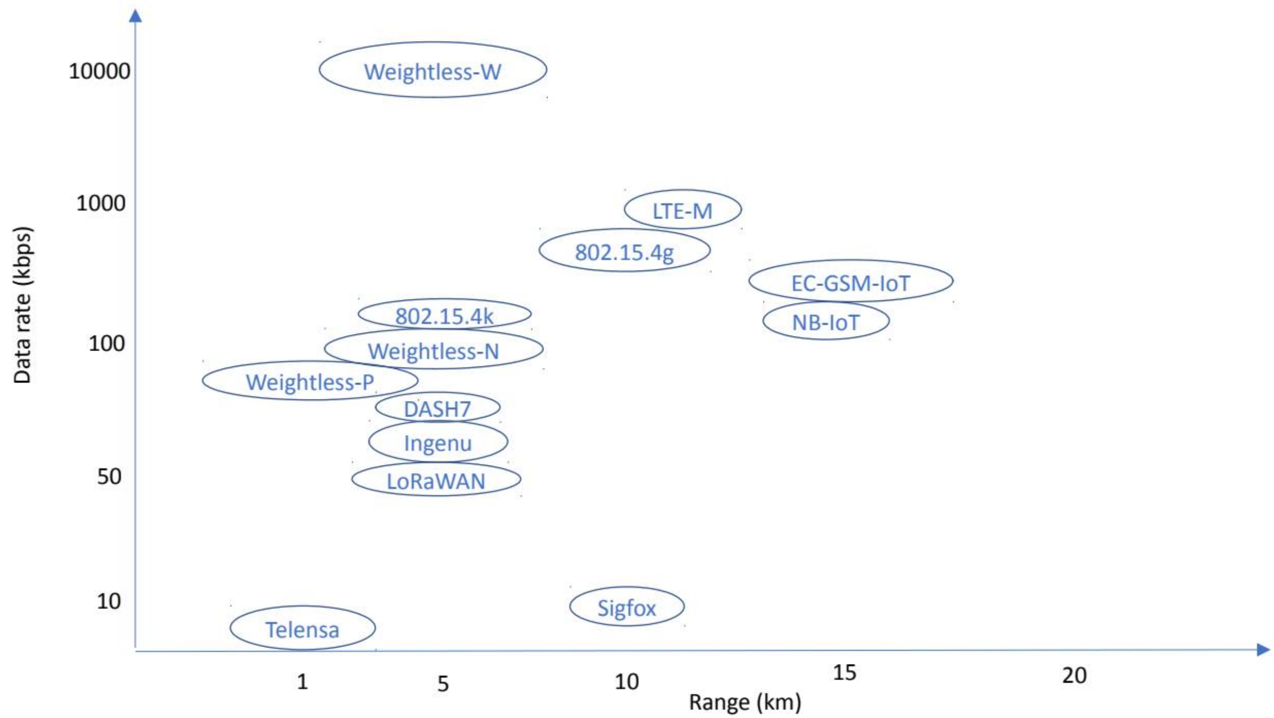


https://www.sigfox.com/en/coverage

Inria

Many RAT, always a trade-off

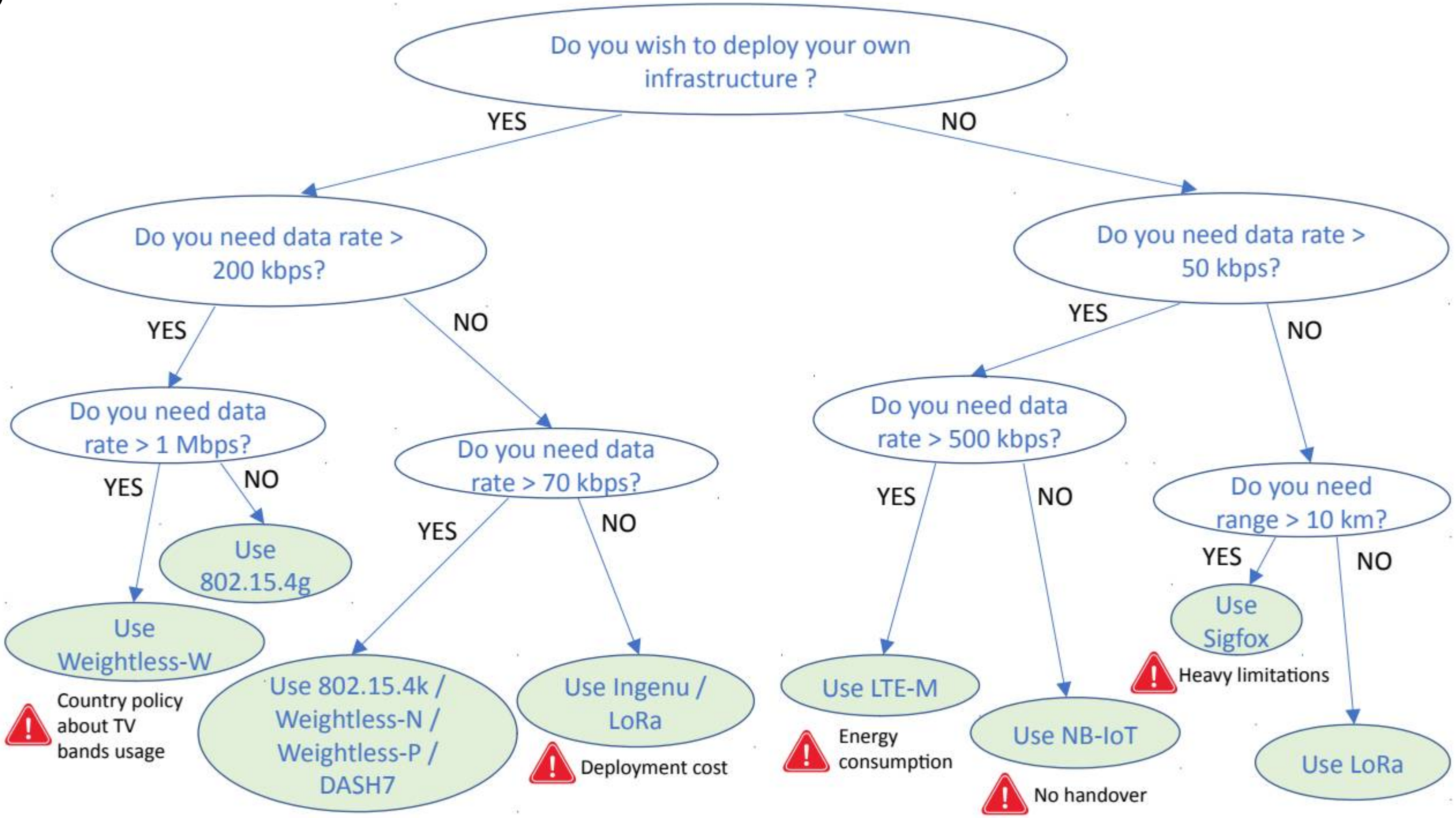
- Spectrum congestion
- Energy consumption
- Financial cost
- Coverage
- Range
- ...



Inria

Hard to find the best fitted RAT

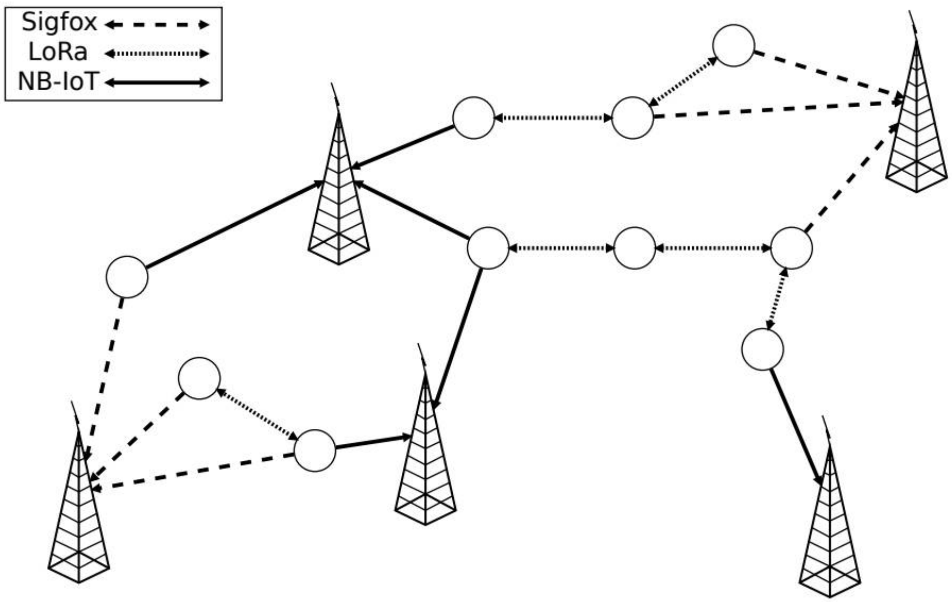
- So why not use several?



Introducing multi-technologies networks, with multi-RAT nodes

- Multi-RAT multi-hop networks
- Several radio links between two neighbors
- Several use cases (e.g. monitoring, video)

→ Need a reliable RAT selection scheme



Route selection problem: multi attribute decision making

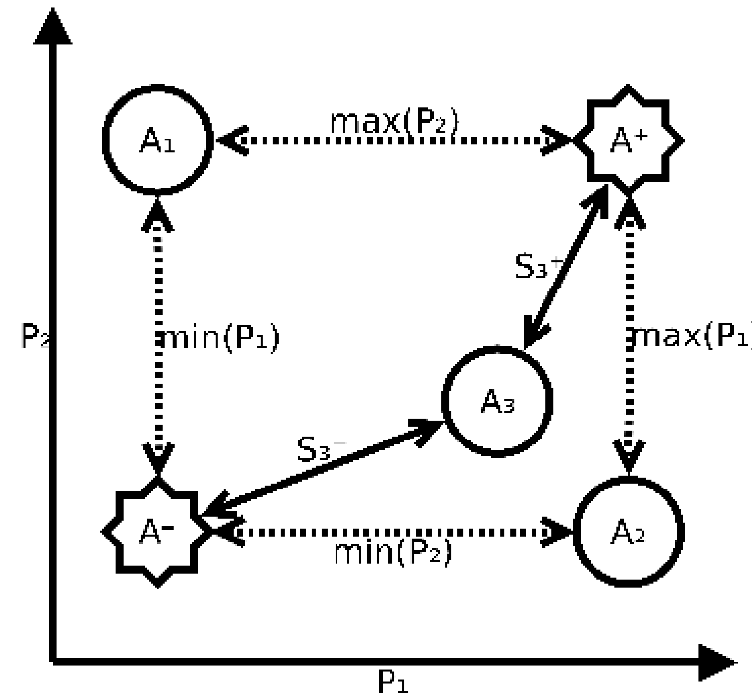
- A = {Ai, for i=1,2,...,n} the set of candidates
- P = {Pj, for j=1,2,...,m} the set of attributes
- W = {wj, for j=1,2,...,m} the weights of each attribute

MADM DECISION MATRIX

	P_1 w_1	P_2 w_2	...	P_m w_m
A_1	x_{11}	x_{12}	...	x_{1m}
A_2	x_{21}	x_{22}	...	x_{2m}
...	...	...	...	...
A_n	x_{n1}	x_{n2}	...	x_{nm}

Technique for Order Preference by Similarity to Ideal Solution (TOPSIS)

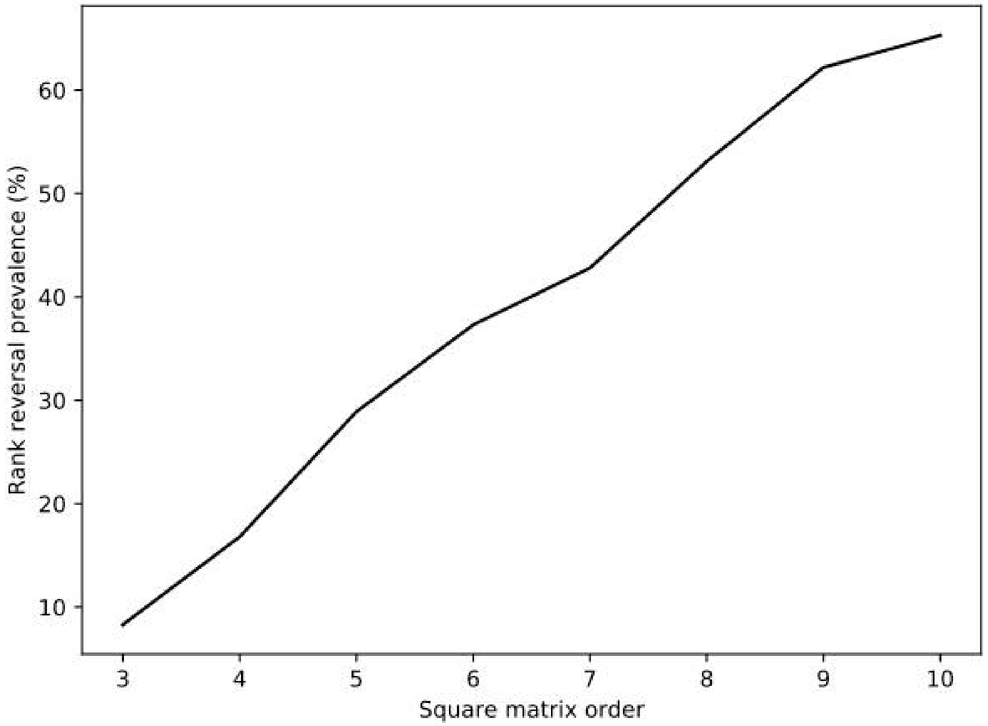
- Ranks each candidate from best to worst
- Ranks are determined based on:
  - closeness to best solution
  - farness to worst solution



Complexity and ranking abnormalities

- Complex calculations for WSN hardware
- Ranking alteration when removing candidates (e.g. route loss)

→ Caused by the euclidean normalization of values



Lightweight TOPSIS for WSN

- Simple and stable normalization algorithm

→ Use fixed bounds for each attribute

Algorithm 1 Lightweight normalization

Require: x_{ij} the raw value of each attribute j for each candidate i
  for each attribute P_j **do**
    if P_j is an upward attribute **then**
      B_j^+ is the upper bound of P_j
      $r_{ij} = \frac{x_{ij}}{B_j^+}$
    else if P_j is a downward attribute **then**
      B_j^- is the lower bound of P_j
      $r_{ij} = \frac{B_j^-}{x_{ij}}$
    end if
  end for
  return r_{ij} the normalized value of x_{ij}

TOPSIS complexity lowered

- Spares 5mn-2 operations (with mn the dimensions of the decision matrix)

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{n} x_{ij}^2}}$$

$$r_{ij} = \frac{B_j^-}{x_{ij}}$$

$$A^+ = [v_1^+ ... v_m^+]$$
$$A^- = [v_1^- ... v_m^-]$$
$$v_j^+ = max\{v_{ij}, i = 1, ..., n\}$$
$$v_j^- = min\{v_{ij}, i = 1, ..., n\}$$

$$v_j^+ = min\{v_{ij}, i = 1, ..., n\}$$
$$v_j^- = max\{v_{ij}, i = 1, ..., n\}$$

$$A^+ = [1...1]$$
$$A^- = [0...0]$$

$$v_j^+ = 1$$
$$v_j^- = 0$$

$$S_i^+ = \sqrt{\sum_{j=1}^{m} (v_j^+ - v_{ij})^2}$$
$$S_i^- = \sqrt{\sum_{j=1}^{m} (v_j^- - v_{ij})^2}$$

$$S_i^+ = \sqrt{\sum_{j=1}^{m} (1 - v_{ij})^2}$$
$$S_i^- = \sqrt{\sum_{j=1}^{m} v_{ij}^2}$$

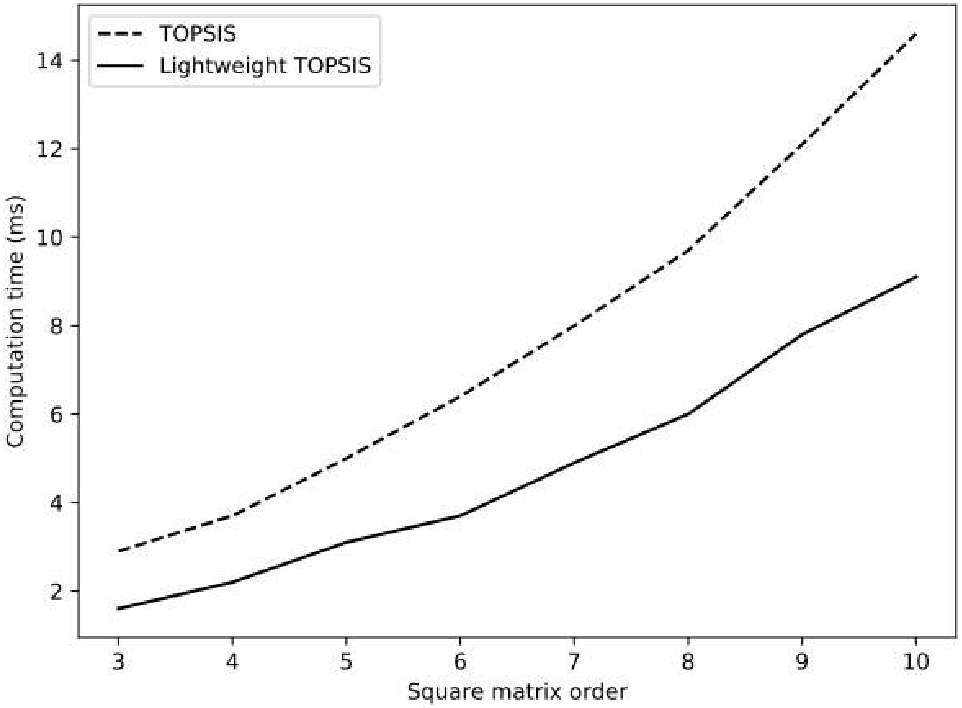
Hardware for experiments: Pycom Fipy

- Allows Python implementation (port of MicroPython)
- 5 RATs available

https://pycom.io/product/fipy/

Lightweight TOPSIS results

- Mean speed up of 39%
- 82% NIS similarity with vanilla TOPSIS
- Saves 448µJ per TOPSIS run (based on FiPy data-sheet)



Recap & future works

- TOPSIS method for multi-criteria selection in WSN
- Custom based method with simpler normalization algorithm
- Quicker execution time for almost identical results, thus energy savings

- Plan to use for route selection in multi-technologies routing

Thank you for your attention!
Any questions?

brandon.foubert@inria.fr