HA NOI UNIVERSITY OF SCIENCE AND TECHNOLOGY
SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

BK.AI

—

Intelligent Communication Networks

Presenter: Nguyen Phi Le

Members

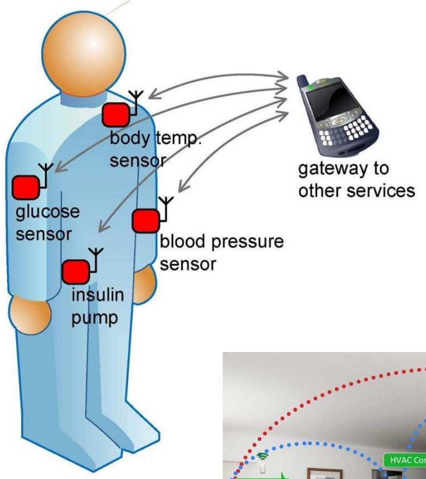
- Permanent members:
  - Tran Hai Anh
  - Do Phan Thuan
  - Nguyen Thanh Hung
  - Nguyen Phi Le (leader)

- + 40 students
  - Ph.D.: 5
  - Masters: 8

Main research topics

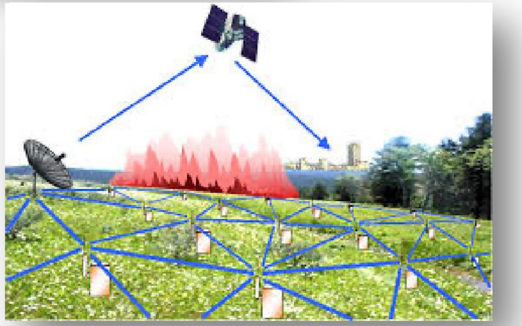
- Wireless sensor networks
- MEC, V2X
- Crowdsensing

Wireless sensor network



Small scale WSNs

- Body area network
- Smart home

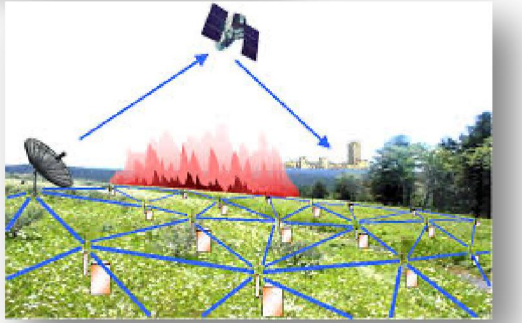


Large scale WSNs

- Environment monitoring
- Disaster focast
- Battle field surveillance
- …

Wireless sensor network

- Routing protocols
- Charging algorithms

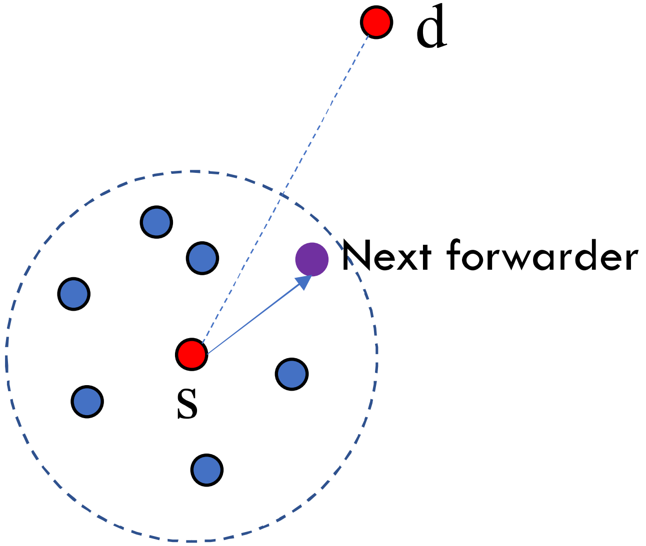


Large scale WSNs

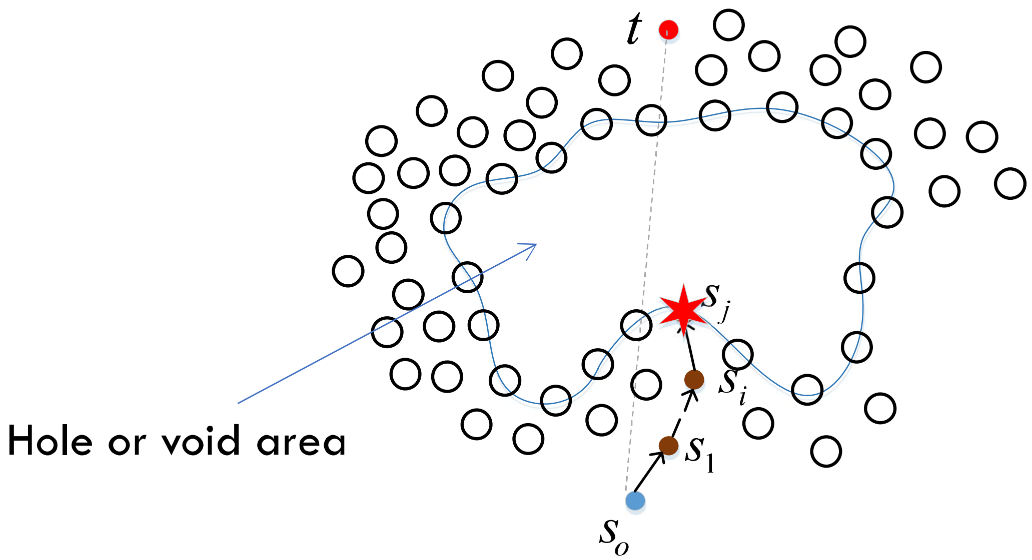
- Environment monitoring
- Disaster focast
- Battle field surveillance
- …

Wireless sensor network (WSNs)

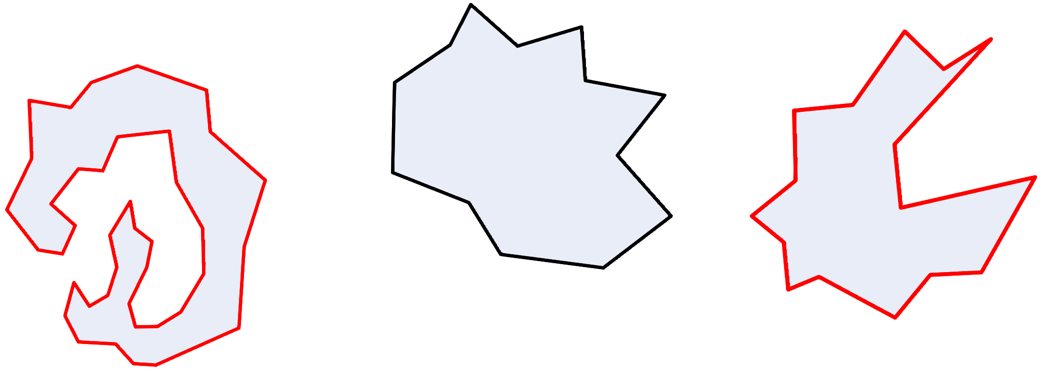
- Hole bypassing problem



Local minimum problem



s_j has no neighbour closer to the destination than itself

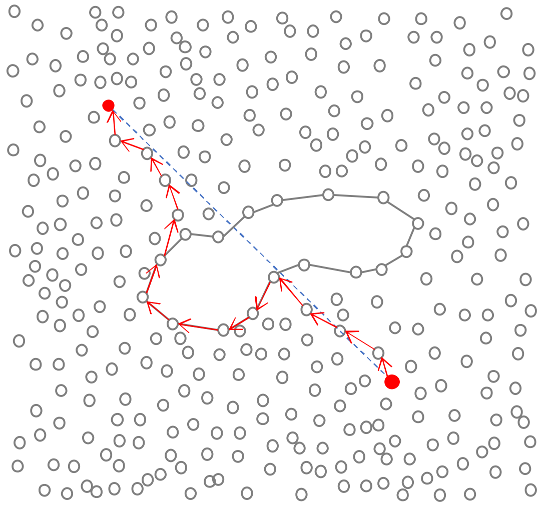


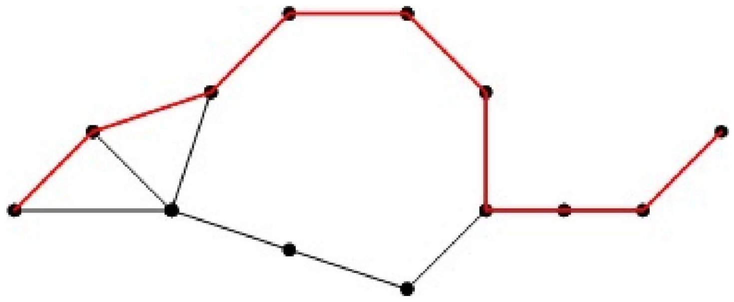
Multiple holes

Research questions

- How to alleviate the holes
- How to balance traffic among the nodes
- How to guarantee QoS constraints

Perimeter approach





- ➢ Using right hand rule to locate the hole
- ➢ Routing the packet along the hole's boundary

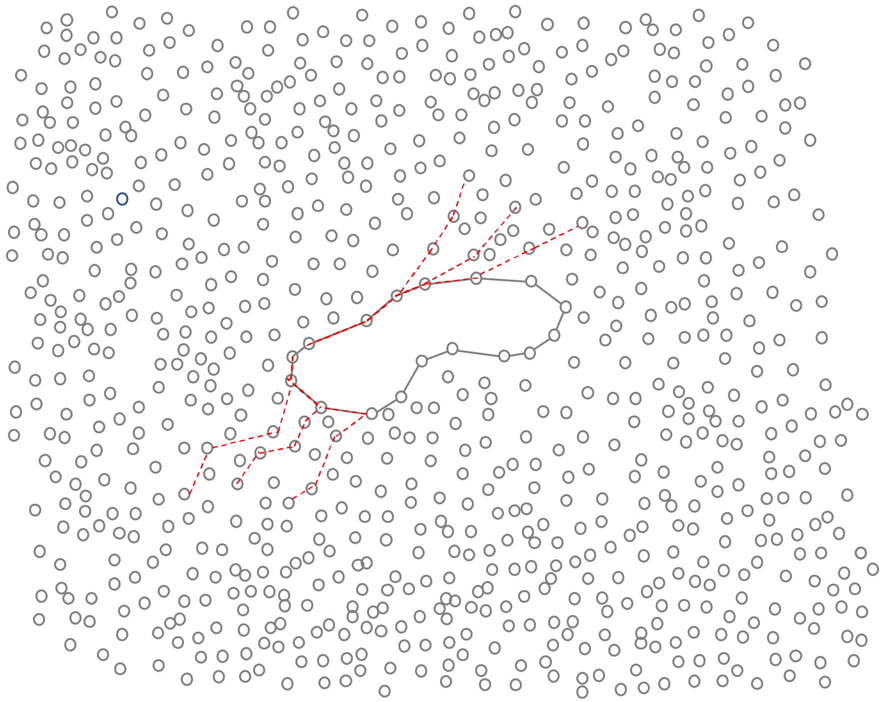
[Qing Fang, Infocom 2004]

- ➢ Using planar graph to locate the hole
- ➢ Routing the packet along the hole's boundary

[B.Karp, Infocom 2000]

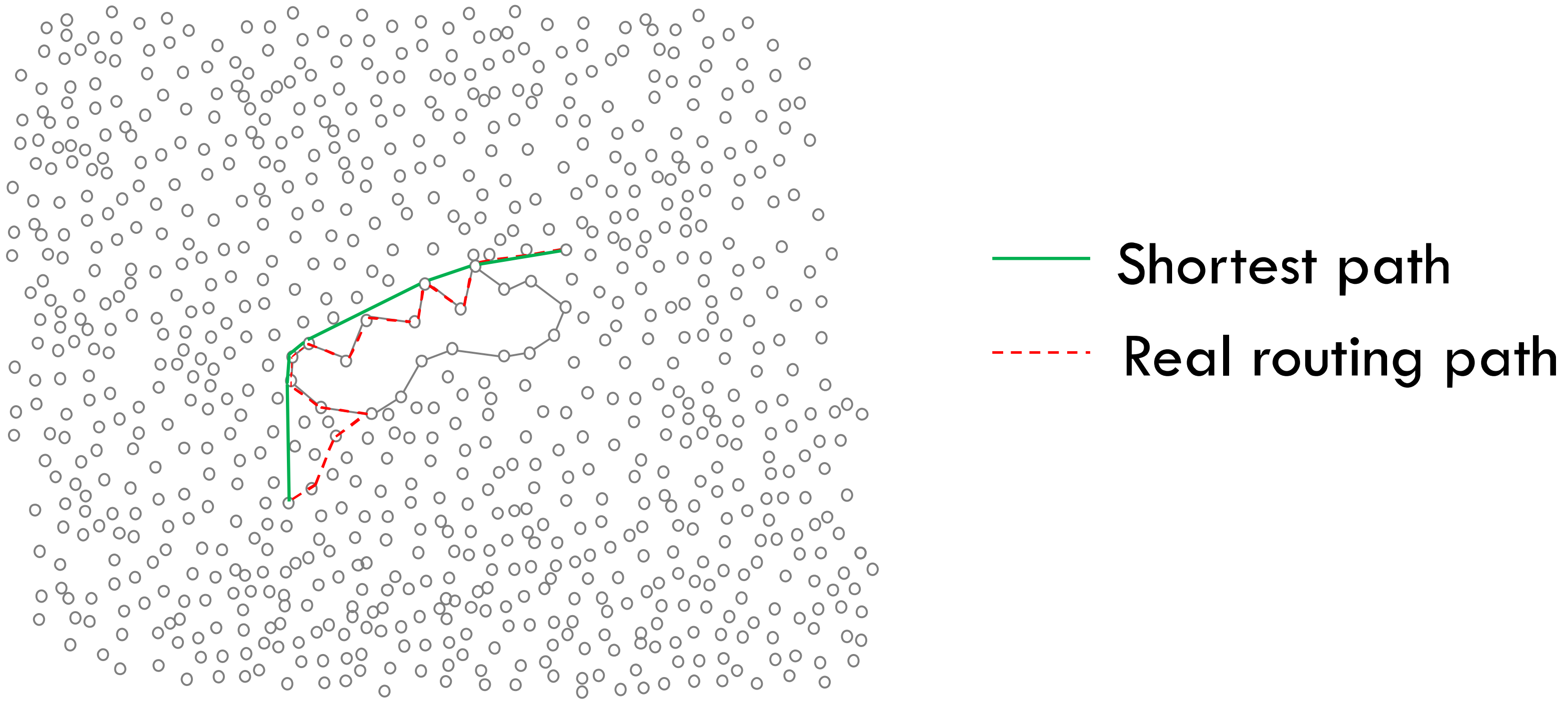
Perimeter approach

- Problems
  - Load imbalance



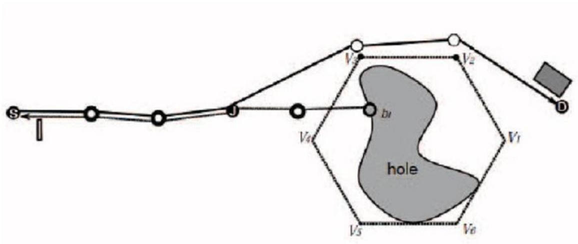
Perimeter approach

- Problems
  - Routing path enlargement

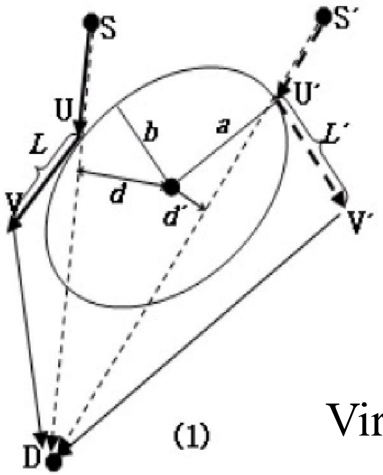


Forbidden area approach

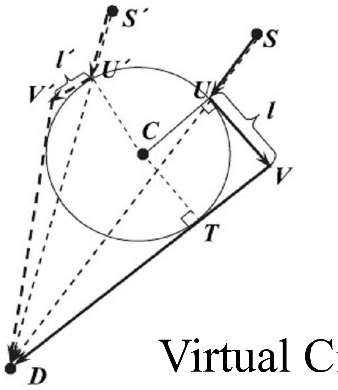
- The nodes know about the presence of hole in advance
- The routing path is determined based on the hole information



Virtual hexagon [H.Choo, ICOIN’11]



Virtual ellipse [Y.Tian, ICC’08]

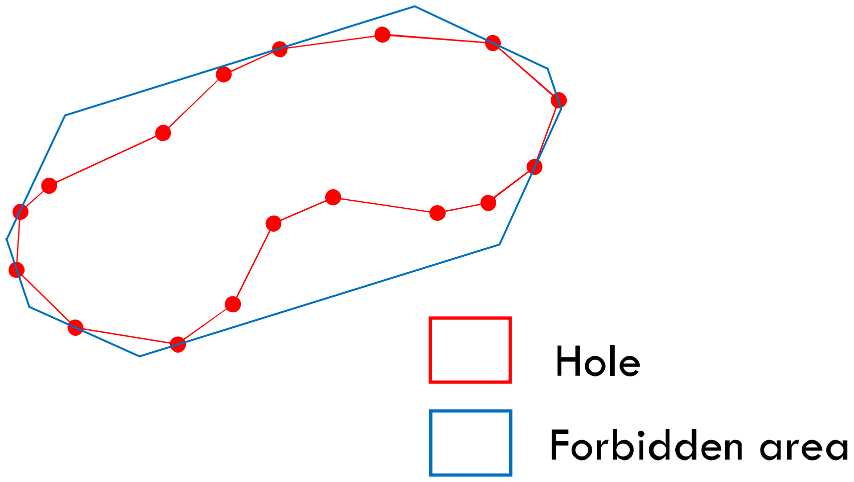


Virtual Circle [F.Yu, JCN 2009]

Elastic forbidden area

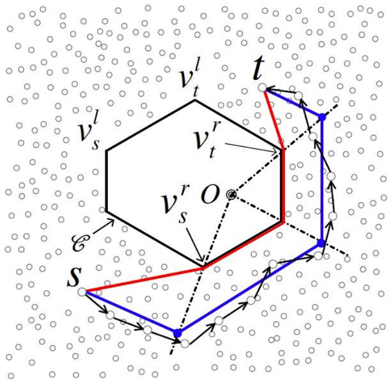
$1+\varepsilon$: <u>predefined</u> routing path stretch upper bound

- Forbidden area (CORE POLYGON):
  - An equiangular polygon with n vertices
  - Covers the hole
  - Each edge contains at least one hole's vertex
  - The number of the vertices (i.e. $n \geq 3$) satisfying that $\frac{1}{\sin\frac{(n-2)\pi}{2n}} < 1+\varepsilon$

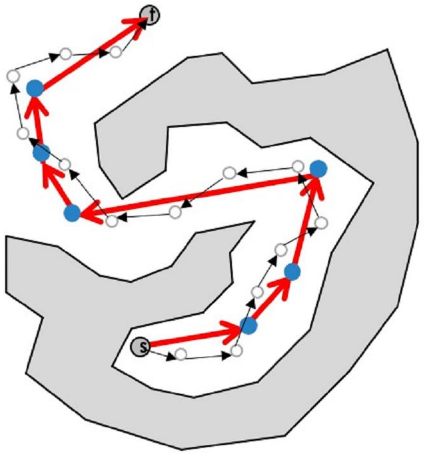


- Routing path stretch does not exceed $1+\varepsilon$
- The information to represent the forbidden area is $O(n)$

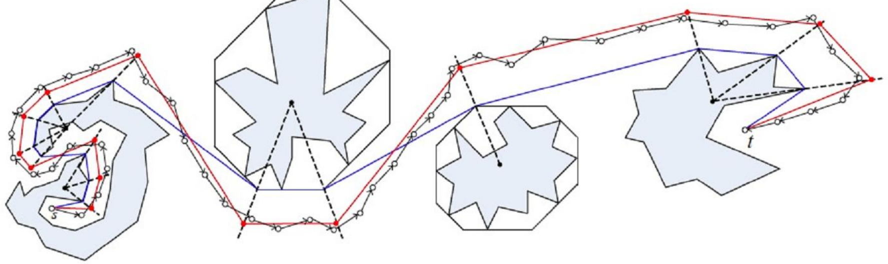
Elastic forbidden area



Bypassing one hole



Routing inside a hole

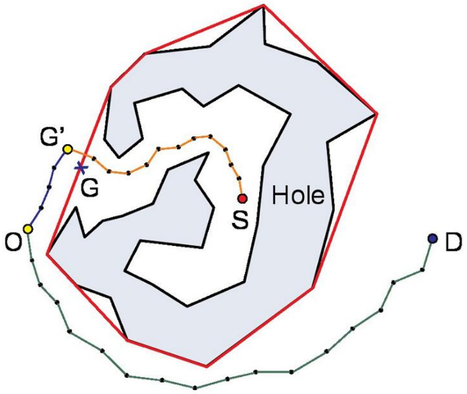


Bypassing multiple holes

Phi Le Nguyen et al., "Constant Stretch and Load Balanced Routing Protocol for Bypassing Multiple Holes in Wireless Sensor Networks", The 16th IEEE International Symposium on Network Computing and Applications (NCA 2017), Boston, USA

- forbidden area
- shortest path bypassing the forbidden area
- magnified path → virtual routing path
- real routing path

Phi Le Nguyen et al., "Load Balanced and Constant Stretch Routing in the Vicinity of Holes in WSNs", the 15th Annual IEEE Consumer Communications & Networking Conference, IEEE CCNC 2018, Las Vegas, USA



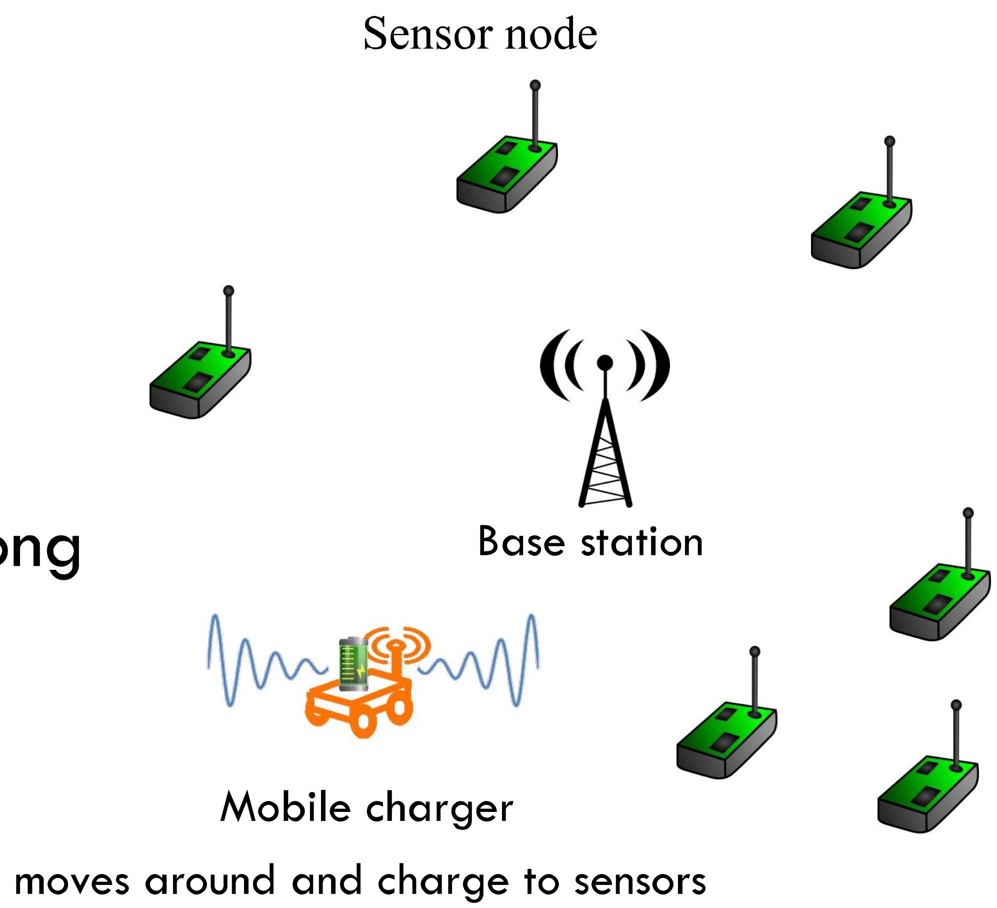
Using Q-learning [ICPADS 2019]

Phi Le Nguyen, Yusheng Ji, Zhi Liu, Huy Vu, Khanh-Van Nguyen, "Distributed Hole-Bypassing Protocol in WSNs with Constant Stretch and Load Balancing", Computer Networks journal 129 (2017), pp. 232-250

$$R(N,B) = -\ \alpha_1 * \left(1 - \frac{E_r(B)}{E_i(B)}\right) - \alpha_2 * \frac{d(B)}{s} - \alpha_3 * \frac{1}{1 + e^{h(B)}}$$

Wireless rechargeable sensor networks (WRSNs)

- Research questions
  - Where should the MC go?
  - How long should the MC stay at each charging point?
  - How many MC do we need?
  - How many depots do we need?
  - Is there any charging plan that can prolong the network lifetime forever?
- Techniques
  - Meta-heuristic algorithms
  - Approximation algorithms
  - Reinforcement learning



Wireless rechargeable sensor networks (WRSNs)

- Key ideas
  - Using exact algorithm to determine the optimal charging time at the next charging point
  - Using Q-learning to determine the next charging point
- Algorithm's overview
  - The MC determines the optimal charging time at every charging point
  - Calculates charging points' reward, updates the Q-table
  - Selects the next charging location whose Q value is the highest

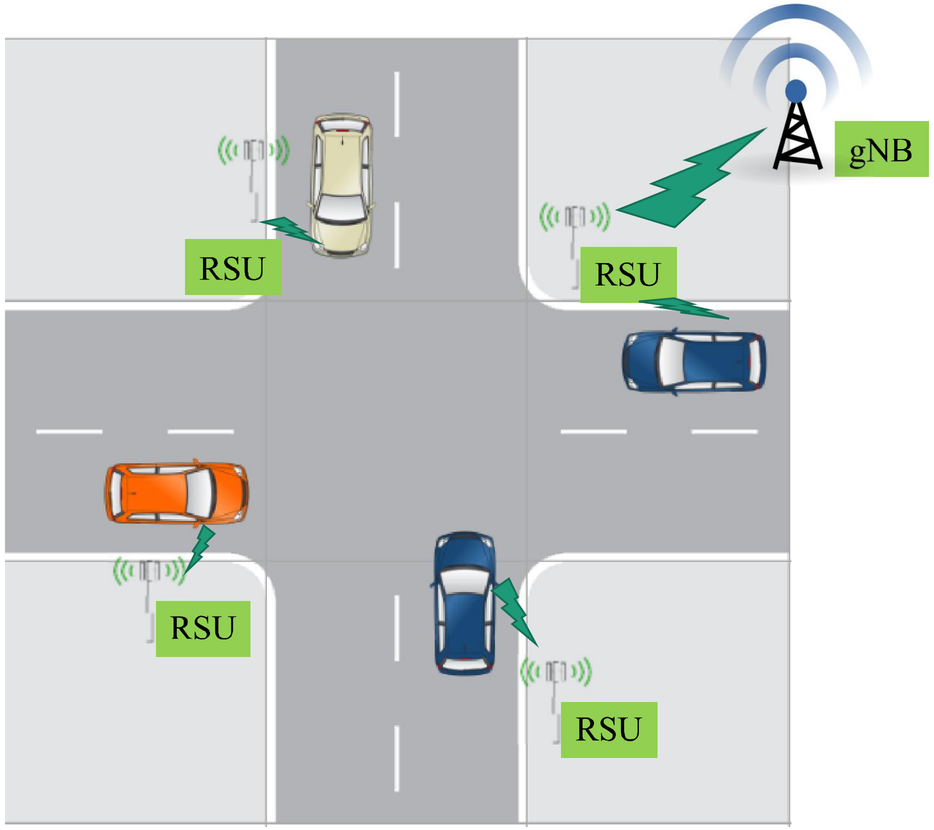
La Van Quan, **Phi Le Nguyen**, Thanh-Hung Nguyen, Kien Nguyen, "Q-learning-based, Optimized On-demand Charging Algorithm in WRSN", The 19th IEEE International Symposium on Network Computing and Applications (NCA 2020), 24-27 November 2020, Virtual conference.

Wireless rechargeable sensor networks (WRSNs)

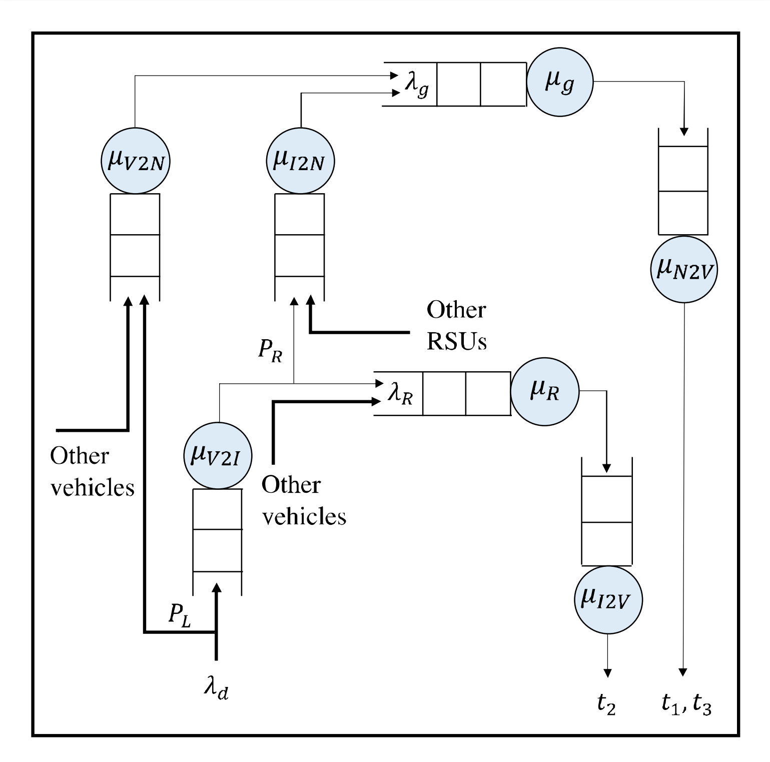
- Ongoing work
  - Multiple chargers
  - Minimizing the number of chargers to prolong the network lifetime forever
  - Optimizing the positions of the depots
  - Horizontal charging
    - Could the sensors charge to each others
- New techniques
  - Deep RL
  - GNN

MEC, V2X

- Application example: Vehicles need to send sensor information to Local Access Data Network (LADN) such that local digital map could be built
- Computation offloading: LADN could be located at RSU or gNB
- Communication offloading: uplink traffic could be sent to RSU or gNB
- Research topics
  - Offloading strategy
- Techniques
  - Mathematical modelling
  - Reinforcement learning
  - Approximation algorithms
  - Meta-heuristic algorithms



MEC, V2X

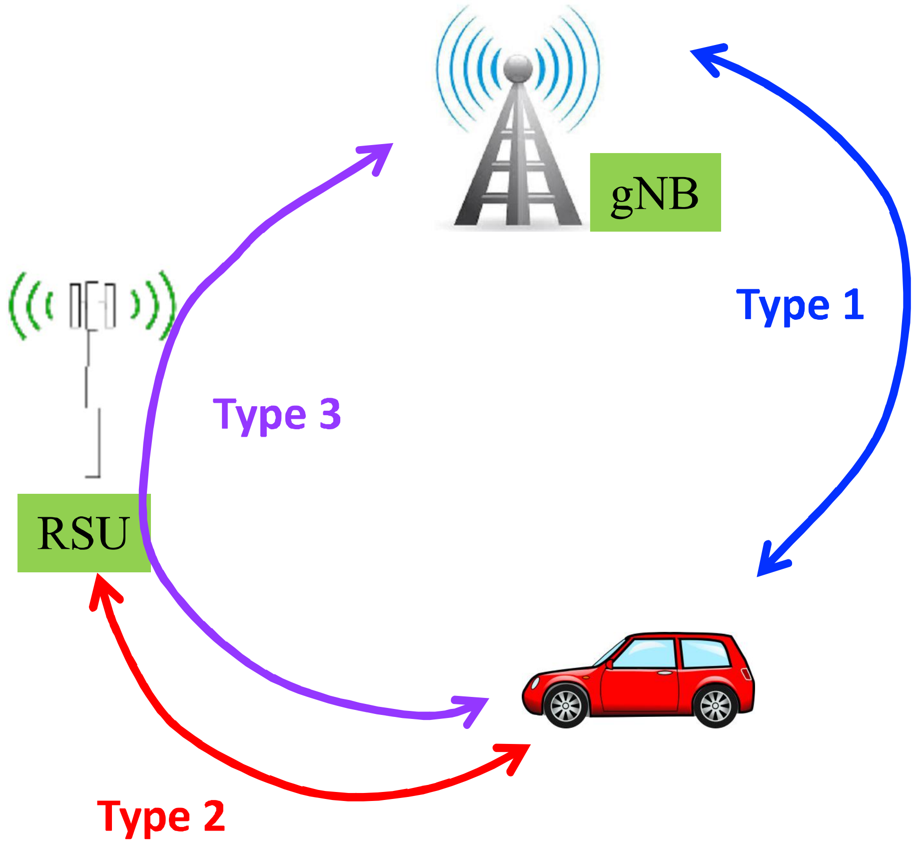


- Given
  - Topology: # of RSUs=N under 1 gNB
  - Vehicle arrival and departure rate (to a RSU): λ_v, μ_v
  - Packet (sensing data) arrival rate from each vehicle: λ_d
  - Service capacity: μ_R, μ_g
  - Link bandwidth: $\mu_{V2I}, \mu_{V2N}, \mu_{N2V}, \mu_{I2V}$
- Offloading in 3-tier network
  - With probability P_L, send to gNB
  - With probability P_R , RSU transfers task the gNB
  - Link bandwidth: μ_{I2N}, μ_{N2I}
- Objective
  - Minimize the average latency

Phi Le Nguyen, Ren-Hung Hwang, Pham Minh Khiem, Kien Nguyen, Ying-Dar Lin, "Modeling and Minimizing Latency in Three-tier V2X Networks", The 2020 IEEE Global Communications Conference (IEEE GLOBECOM), 8–10 December 2020, Taipei, Taiwan

Latency derivation

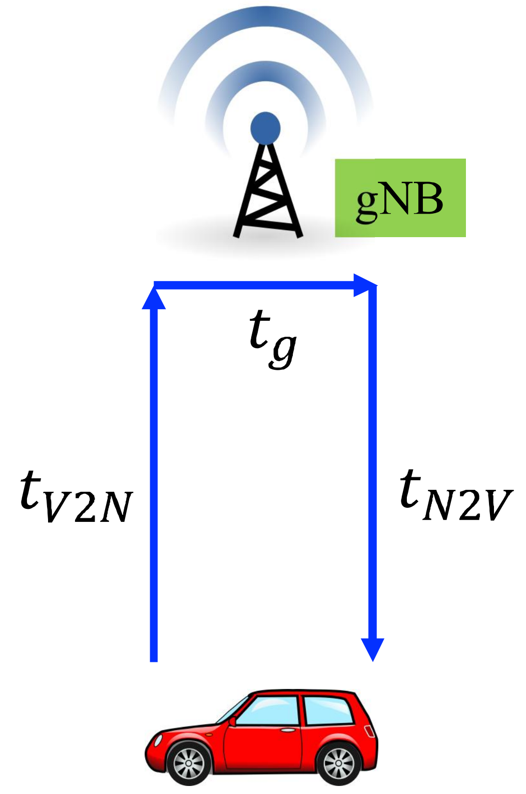
- **Type 1:** Tasks that are offloaded directly to the gNB
- **Type 2:** Tasks that are offloaded to the RSU and then processed at the RSU
- **Type 3:** Tasks that are offloaded to the RSU and then forwarded to the gNB



Latency derivation: Type 1

- $t_1 = t_{V2N} + t_g + t_{N2V}$
  - t_{V2N}, t_{N2V}: transmission time from vehicle to the gNB and vice versa
  - t_g: processing time at the gNB

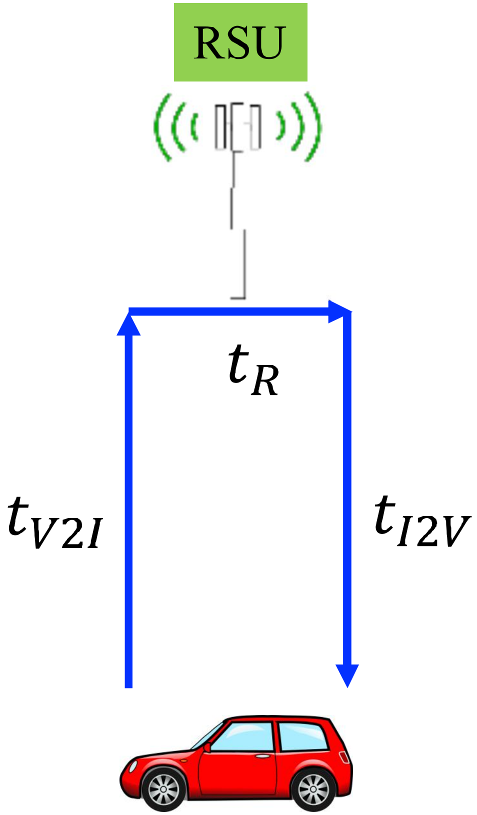
$$
\begin{aligned}
t_1 = & \frac{1}{\mu_{V2N} - \frac{\rho_v}{1-\rho_v}\lambda_d N P_L} && \leftarrow t_{V2N} \\
& + \frac{1}{\mu_g - \frac{\rho_v}{1-\rho_v}\lambda_d N P_L - \frac{\rho_v}{1-\rho_v}\lambda_d N (1-P_L) P_R} && \leftarrow t_g \\
& + \frac{1}{\mu_{N2V} - \frac{\rho_v}{1-\rho_v}\lambda_d N P_L - \frac{\rho_v}{1-\rho_v}\lambda_d N (1-P_L) P_R} && \leftarrow t_{N2V}
\end{aligned}
$$



Latency derivation: Type 2

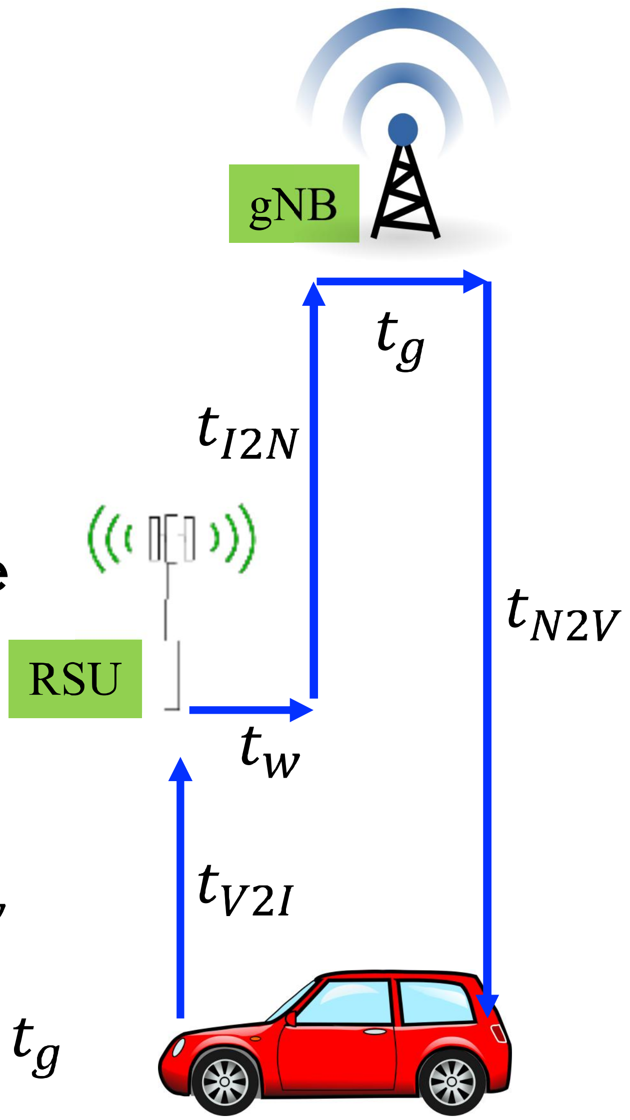
- $t_1 = t_{V2I} + t_R + t_{I2V}$
  - t_{V2I}, t_{I2V}: transmission time from vehicle to the RSU and vice versa
  - t_R: processing time at the RSU

$$
\begin{aligned}
t_2 = & \frac{1}{\mu_{V2I} - \frac{\rho_v}{1-\rho_v}\lambda_d(1-P_L)} && \longleftarrow \text{- - -} \quad t_{V2I} \\
& + \frac{1}{\mu_R - \frac{\rho_v}{1-\rho_v}\lambda_d(1-P_L)(1-P_R)} && \longleftarrow \text{- - -} \quad t_R \\
& + \frac{1}{\mu_{I2V} - \frac{\rho_v}{1-\rho_v}\lambda_d(1-P_L)(1-P_R)} && \longleftarrow \text{- - -} \quad t_{I2V}
\end{aligned}
$$



Latency derivation: Type 3

- $t_1 = t_{V2I} + t_{I2N} + t_g + t_{N2V}$
  - t_{V2I}: the transmission time from vehicle to the RSU
  - t_{I2N}: the transmission time from the RSU to the gNB
  - t_{N2V}: the transmission time from the gNB to the vehicle
  - t_w: the waiting time at RSU
  - t_g: processing time at the gNB



$$t_3 = \frac{1}{\mu_{V2I} - \frac{\rho_v}{1-\rho_v}\lambda_d(1-P_L)} + \left(\frac{1}{\mu_R - \frac{\rho_v}{1-\rho_v}\lambda_d(1-P_L)(1-P_R)} - \frac{1}{\mu_R}\right) \quad \leftarrow t_{V2I} + t_w$$

$$+ \frac{1}{\mu_{I2N} - \frac{\rho_v}{1-\rho_v}\lambda_d N(1-P_L)P_R} + \frac{1}{\mu_g - \frac{\rho_v}{1-\rho_v}\lambda_d N P_L - \frac{\rho_v}{1-\rho_v}\lambda_d N(1-P_L)P_R} \quad \leftarrow t_{I2N} + t_g$$

$$+ \frac{1}{\mu_{N2V} - \frac{\rho_v}{1-\rho_v}\lambda_d N P_L - \frac{\rho_v}{1-\rho_v}\lambda_d N(1-P_L)P_R} \quad \leftarrow t_{N2V}$$

Problem formulation

- The average latency:
  - $\bar{t} = P_L t_1 + (1 - P_L)(1 - P_R)t_2 + (1 - P_L)P_R t_3$
- Minimize $\bar{t}$
- Subject to

$$\frac{\rho_v}{1-\rho_v}(1-P_L)\lambda_d \le \mu_{V2I}$$

$$\frac{\rho_v}{1-\rho_v}\lambda_d(1-P_L)(1-P_R) \le \mu_{I2V}$$

$$\frac{\rho_v}{1-\rho_v}\lambda_d P_L N \le \mu_{V2N}$$

$$\frac{\rho_v}{1-\rho_v}\lambda_d N P_L + \frac{\rho_v}{1-\rho_v}(1-P_L)\lambda_d N P_R \le \mu_{N2V}$$

$$\frac{\rho_v}{1-\rho_v}\lambda_d(1-P_L)P_R N \le \mu_{I2N}$$

$$\frac{\rho_v}{1-\rho_v}(1-P_L)(1-P_R)\lambda_d \le \mu_R$$

$$\frac{\rho_v}{1-\rho_v}\lambda_d N P_L + \frac{\rho_v}{1-\rho_v}(1-P_L)\lambda_d N P_R \le \mu_g$$

GA-based optimization

- Chromosome representation
- Fitness function
- Crossover
- Mutation

- Chromosome representation and fitness function
  - Chromosome: two genes representing values of P_R and P_L

P_R	P_L

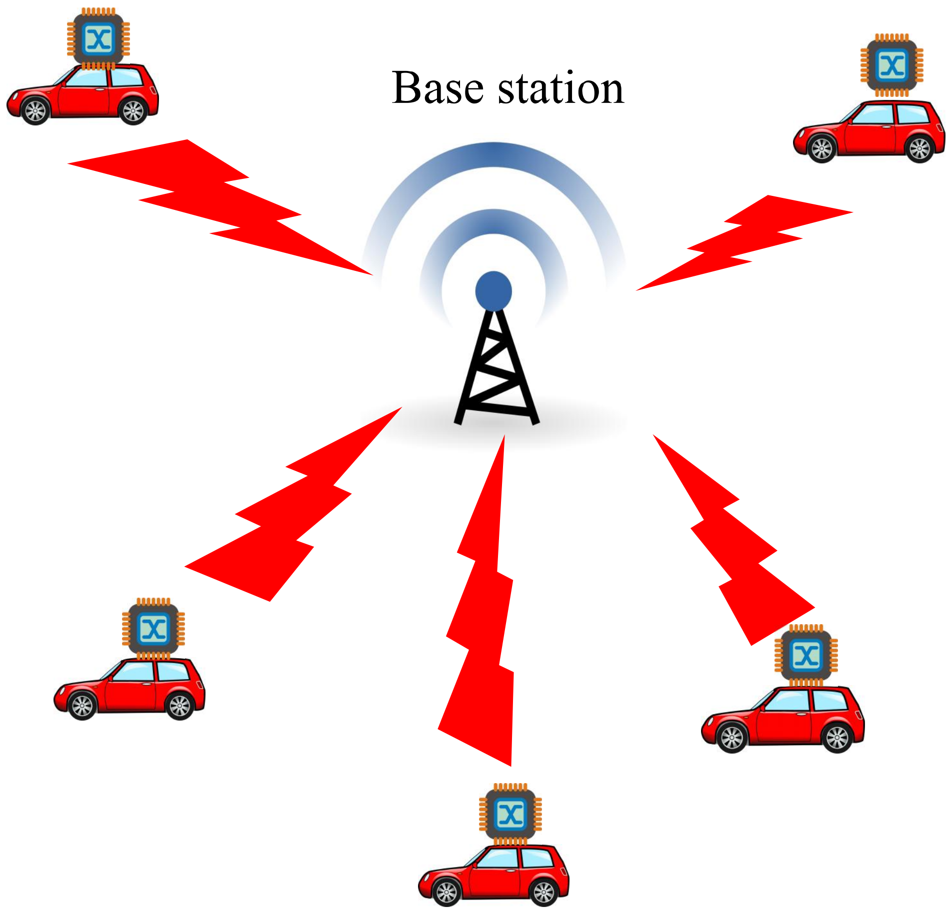
  - Fitness function: the average latency
    - $f(P_R, P_L) = P_L t_1 + (1 - P_L)(1 - P_R)t_2 + (1 - P_L)P_R t_3$

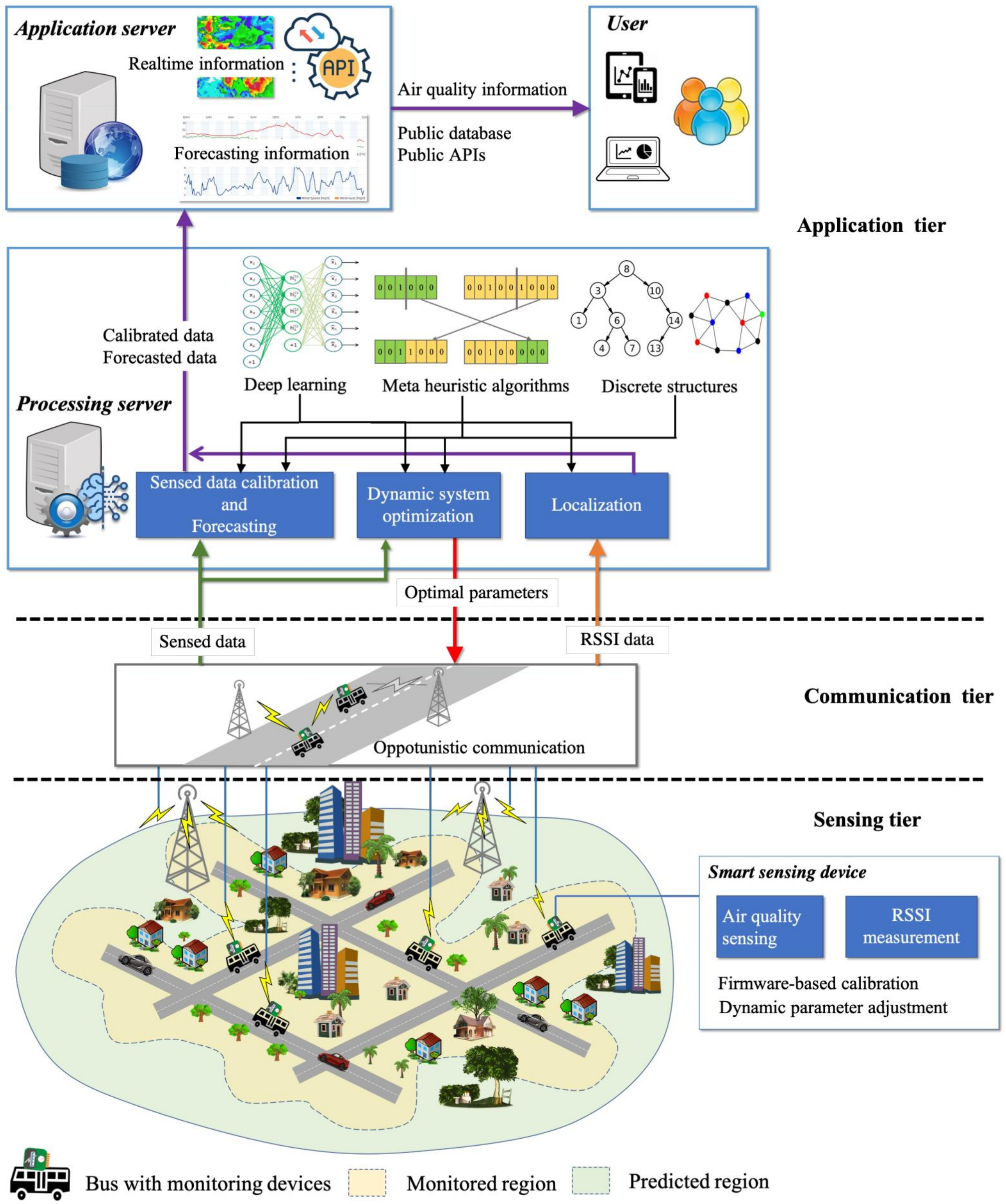
MEC, V2X

- Ongoing work
  - Horizontal offloading
  - More constraint
- Techniques
  - Reinforecement learning
  - Deep reinforcement learing
  - Fuzzy logic

Crowdsensing

- Scenario
  - Air quality monitoring decvices are mounted on vehicles
  - Devices collect and transfer sensory data to the base station
- Reasearch questions
  - How frequently should the devices measure?
  - When should the devices report to the base station?
- Objective
  - Minimize the communcation cost
- Constraints
  - QoS
  - Data quality
- Techniques
  - Mathematical modelling
  - Reinforecement learning
  - Deep learning





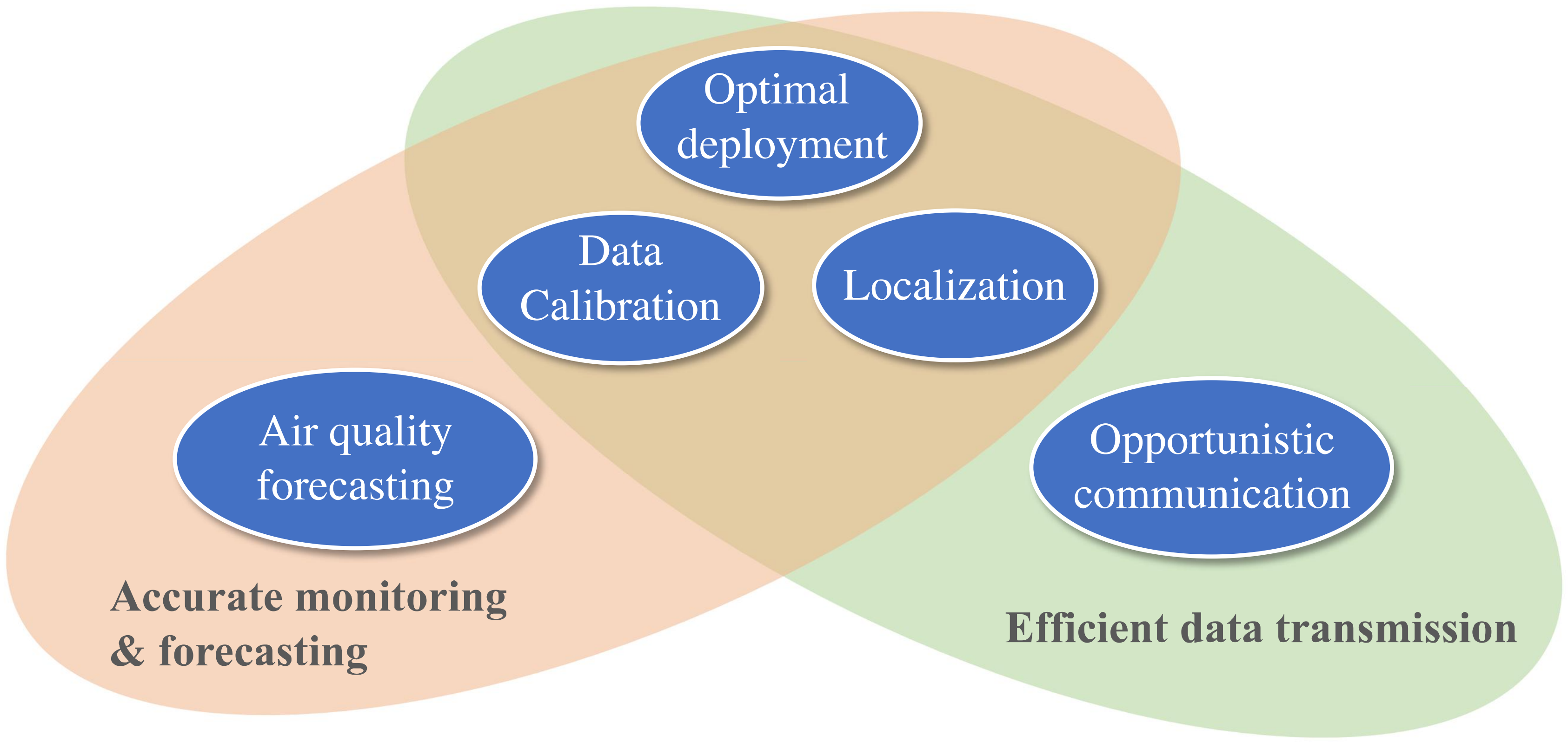
3-tier Fi-Mi Architecture

- **Sensing tier**
  - Collects **real-time** air quality data
  - Carried by air **monitoring devices** deployed on **vehicular devices** such as buses
- **Communication tier**
  - **Transfers data** between the monitoring devices and the servers
- **Application tier**
  - **Stores** the sensory data
  - **Forecasts** the **future** trend of the air quality
  - **Predicts** air quality in **un-monitored regions**
  - **Optimizes** the behaviors of the monitoring devices
  - Provides information to users through **smartphone applications** and **web portal**

Device implementation

- Finish prototyping
- Implementing real devices

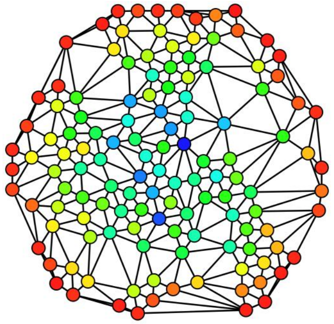
Research problems



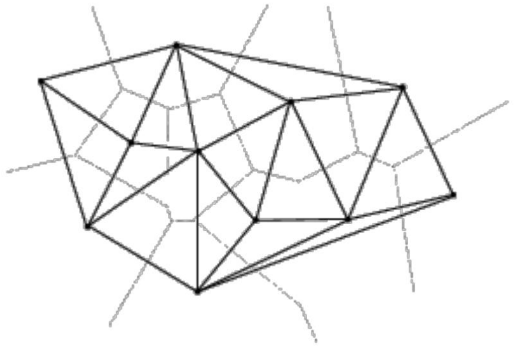
Research problem 1: Optimal deployment

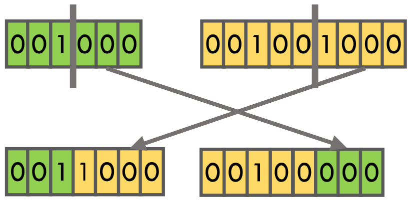
- **How many devices** are enough?
- **Which buses** to place the monitoring devices?
- **How frequently** should we measure?
- **When measuring** the air quality?
- Advanced problems
  - Bus routes change **dynamically**

Graph theory

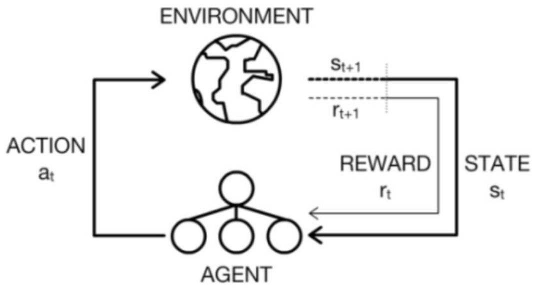


Computational Geometry



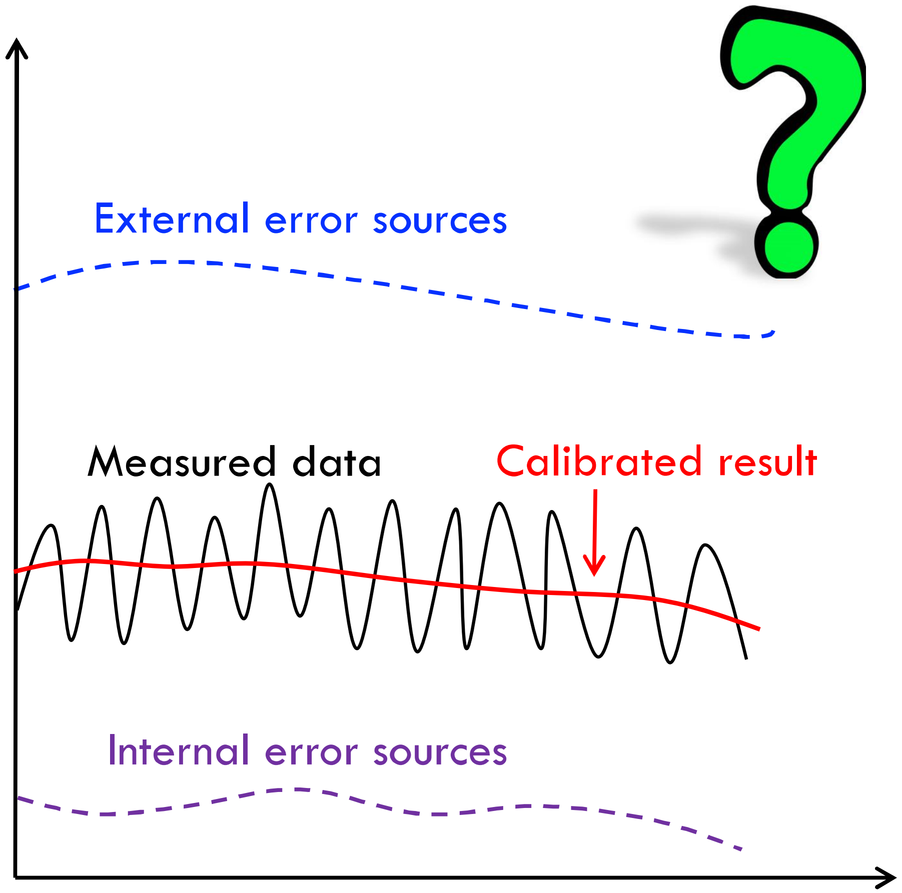


Meta-heuristic Algorithms



Reinforcement Learning

Research problem 2: Data calibration



Calibration period

Calibration method

Reference points

Regression methods

Expert knowledge

Deep learning

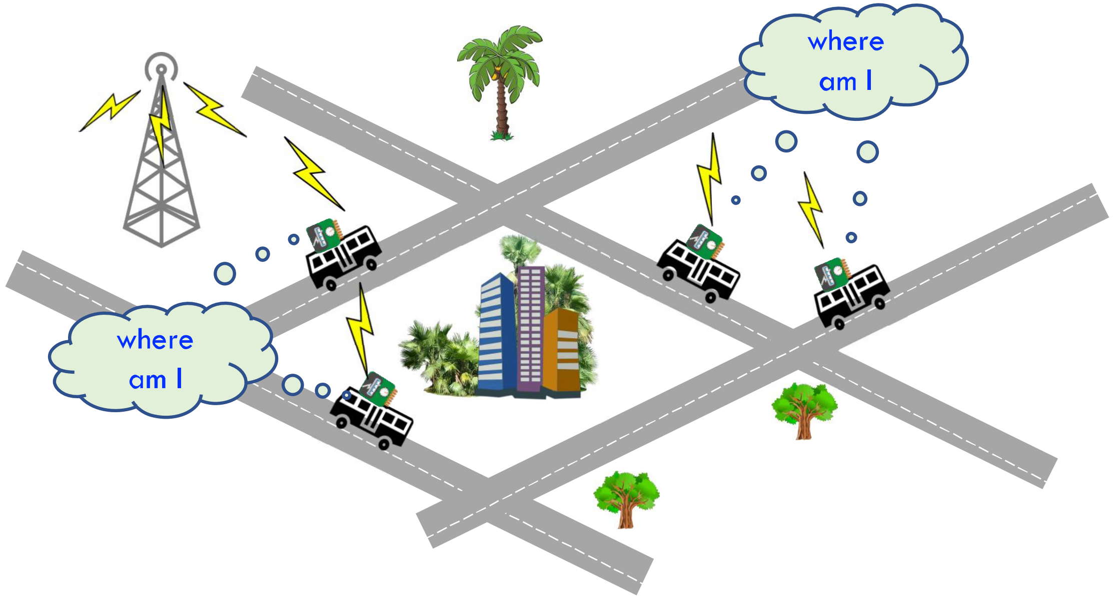
Static curve

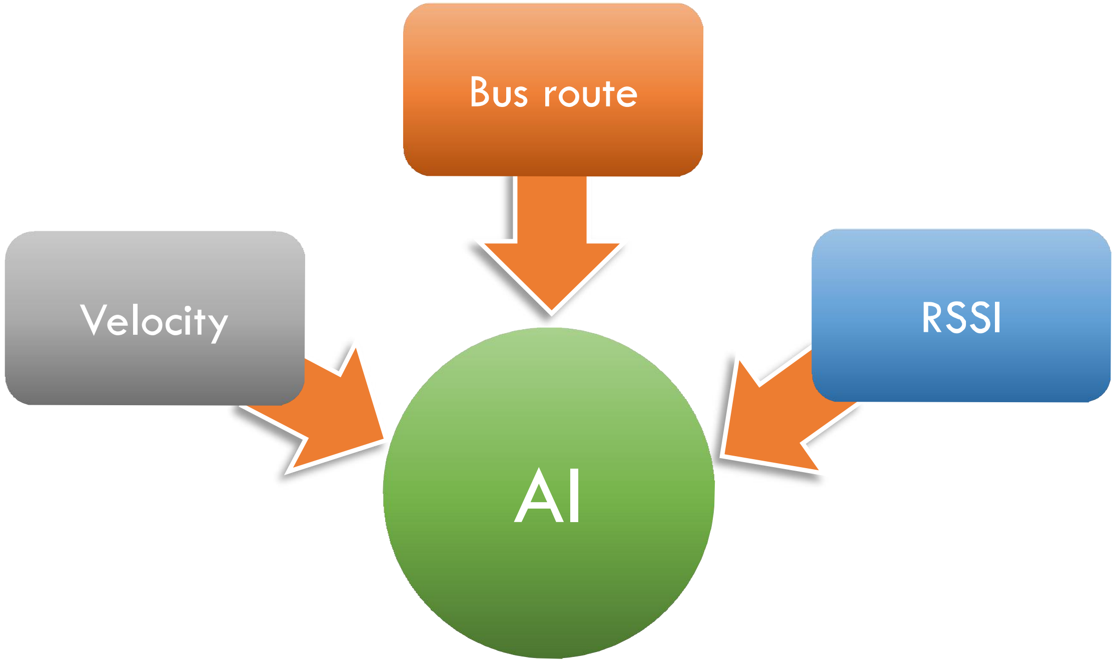
Equation

Research problem 3: Localization

- High accuracy
- Low cost

Research directions





Research problem 4: Air quality analysis and forecasting

Humidity

NOx

O_3

°C

SOx

Wind

PMx

Time

Which **features** should we use?

Data imputation

Outlier filtering

Model construction

Which **techniques** should we use?

Which **affects** the air quality?

Research problem 5: Opportunistic communication

Voluminous Data

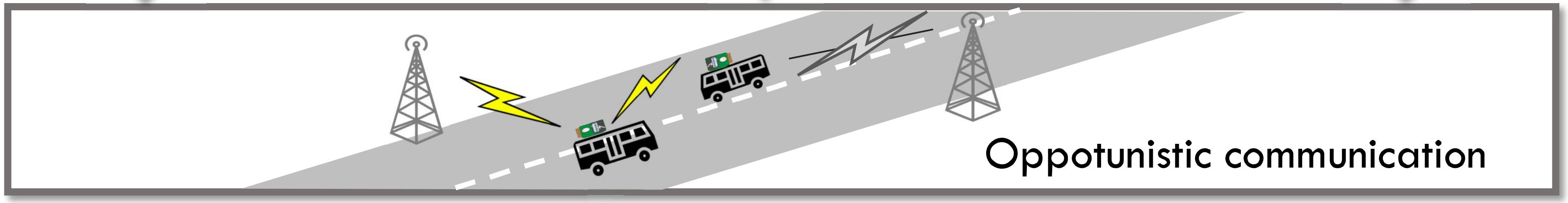
(Sensing, wireless, etc.)

Efficiency

(Cost, flexibility, etc.)

Quality of Service

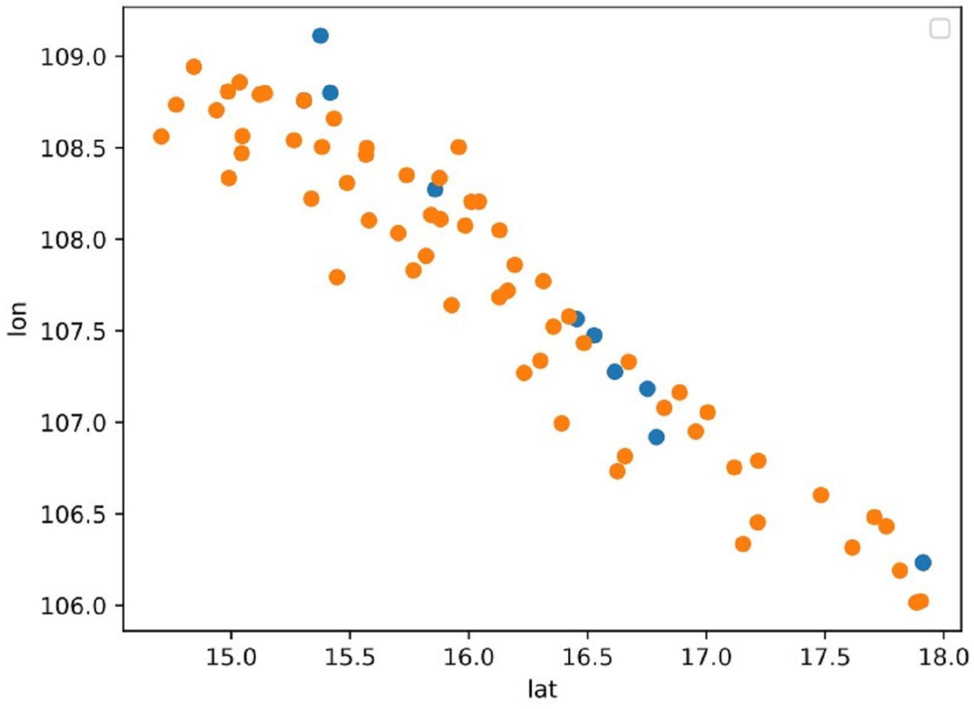
(Real-time, reliability, etc.)



- De-ossification of the TCP/IP stack
- Leveraging Softwarization technologies
- Context-based Communication

Prediction

- Temporal prediction
  - Using historical data to predict future information
    - PM2.5 prediction, water level prediction, …
- Spatial prediction
  - Using data collected by other stations to predict a targeted station



● Stations whose data is used to train the model

● Stations whose data will be predicted


Thank
You!