Algebraic Computations in Anonymous VANET

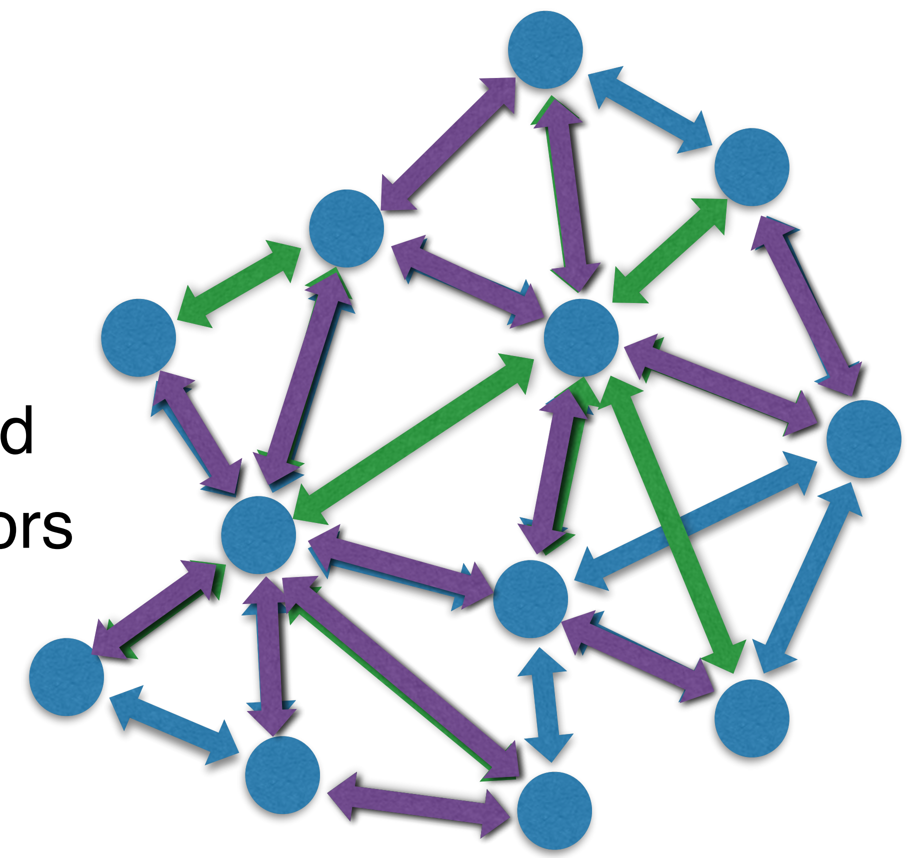
Dariusz R. Kowalski
Augusta Univ.

Miguel A. Mosteiro
Pace Univ.

Austin Powlette
Pace Univ.

NETYS 2024

Anonymous Dynamic Networks (ADN)



- **Fixed set of n nodes**
  - No identifiers or labels
- **Synchronous communication :** At each round
  - a node broadcasts a message to its neighbors
  - receives the messages of its neighbors
  - executes some local computation
- **1-interval connectivity**
  - communication links may change from round to round, but
  - at each round the network is connected
- **Distinguished nodes**
  - $0 < \ell < n$ "counted" network nodes with ℓ known to the algorithm.

  (Counting not solvable with $\ell = 0$ or ℓ unknown.)

Congested ADN with Opportunistic Connectivity

- **Fixed set of n nodes**
  - No identifiers or labels, internal memory limited to $O(\log n)$ bits
- **Synchronous communication :** At each round
  - a node broadcasts a $O(\log n)$ bits message to its neighbors
  - receives the messages of its neighbors
  - executes some local computation
- **T-connected**
  - communication links may change from round to round, but
  - the union of T snapshot graphs is connected.
- **Distinguished nodes**
  - $0 < \ell < n$ "counted" network nodes with ℓ known to the algorithm.

  (Counting not solvable with $\ell = 0$ or ℓ unknown.)

privacy!

inexpensive devices!

mobility!

Computation framework suitable for VANET:

Anonymous Vehicular Adhoc Networks (A-VANET)

Restricted Methodical Counting

[Kowalski-Mosteiro,22]

Algebraic computations require to know or compute n (Counting Problem).

Moreover, many are doable with a counting algorithm (e.g. AVG, SUM).

We focus on the Counting Problem.

RMC key ideas:

- ℓ counted network nodes and $n - \ell$ uncounted network nodes.
- try network size estimates $k = 2^i(\ell + 1)$ and binary search after estimate $k > n$.
- all nodes share some “potential” values for some function of k iterations.
- counted nodes remove potential every now and then to evaluate k.
- carefully designed alarms allow to detect correct or wrong k.

RMC Epoch Example

epochs:

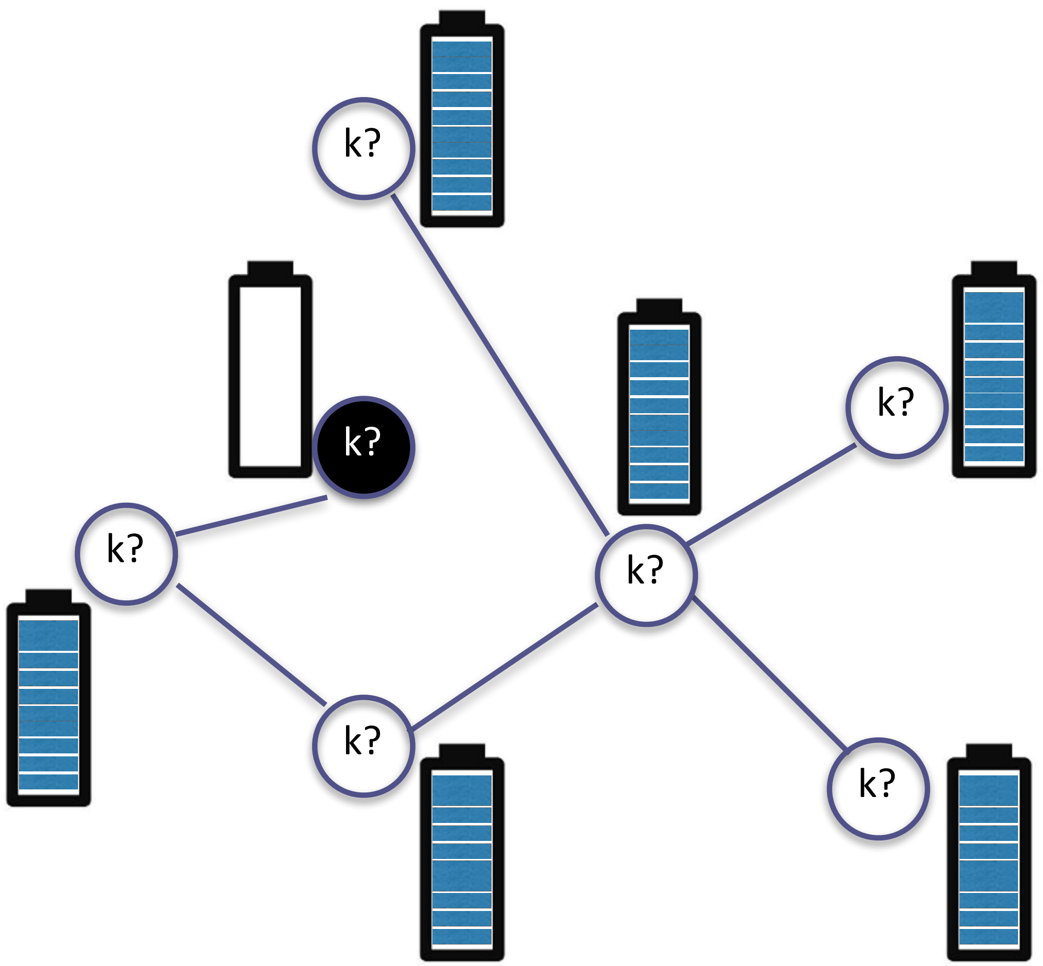
- one for each estimate k
- initialize potentials: $\Phi_{uncounted} = \ell, \Phi_{counted} = 0$

$p(k)$ phases:

(to let counted remove “enough” potential ρ)

$r(k)$ rounds:

(to “average” the current potentials Φ)



RMC Epoch Example

epochs:

- one for each estimate k
- initialize potentials: $\Phi_{uncounted} = \ell, \Phi_{counted} = 0$

$p(k)$ phases:

(to let counted remove “enough” potential ρ)

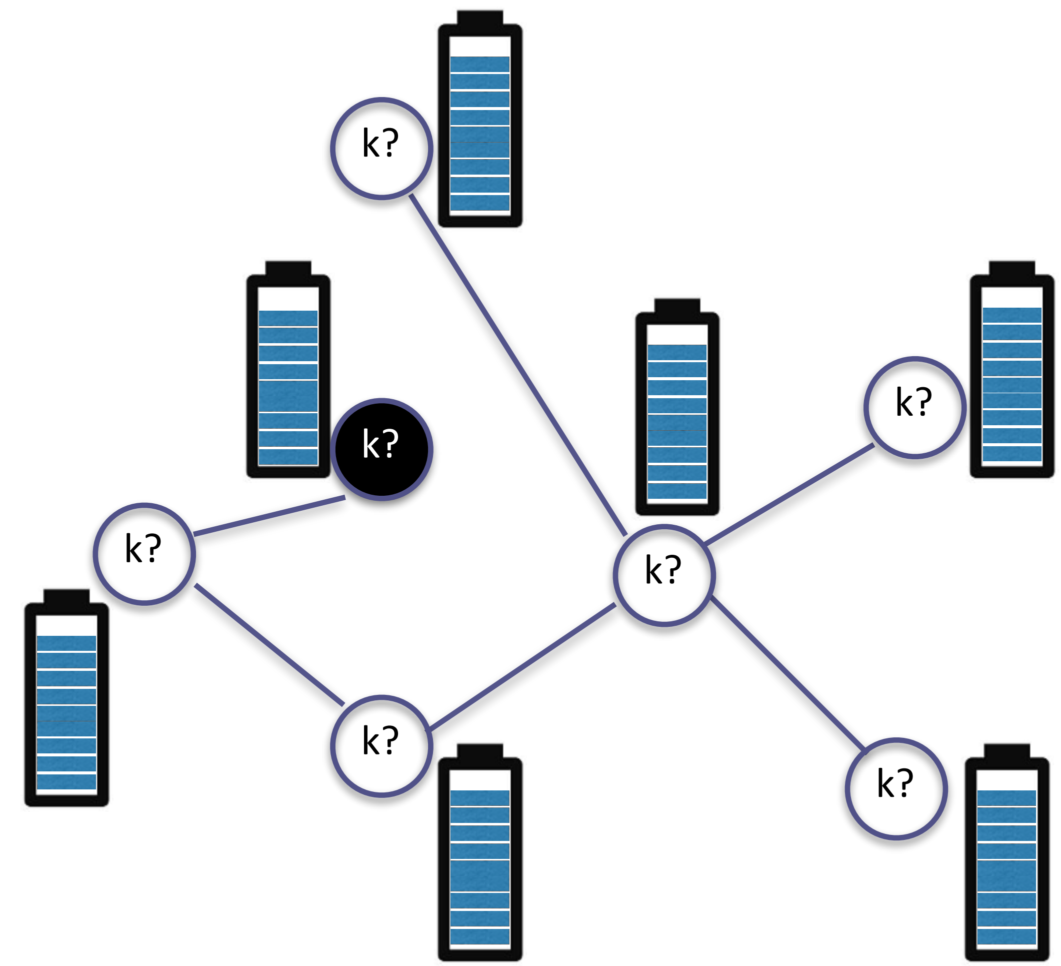
$r(k)$ rounds:

(to “average” the current potentials Φ)

mass distribution:

- broadcast Φ and receive neighbors’ Φ_i
- $\Phi = \Phi + \sum_{i \in N} \Phi_i / d(k) - |N| \Phi / d(k)$ truncated to $O(\log n)$ bits

- counted remove potential: $\rho = \rho + \Phi, \Phi = 0$



$\rho=$

RMC Epoch Example

epochs:

- one for each estimate k
- initialize potentials: $\Phi_{uncounted} = \ell, \Phi_{counted} = 0$

$p(k)$ phases:

(to let counted remove "enough" potential ρ)

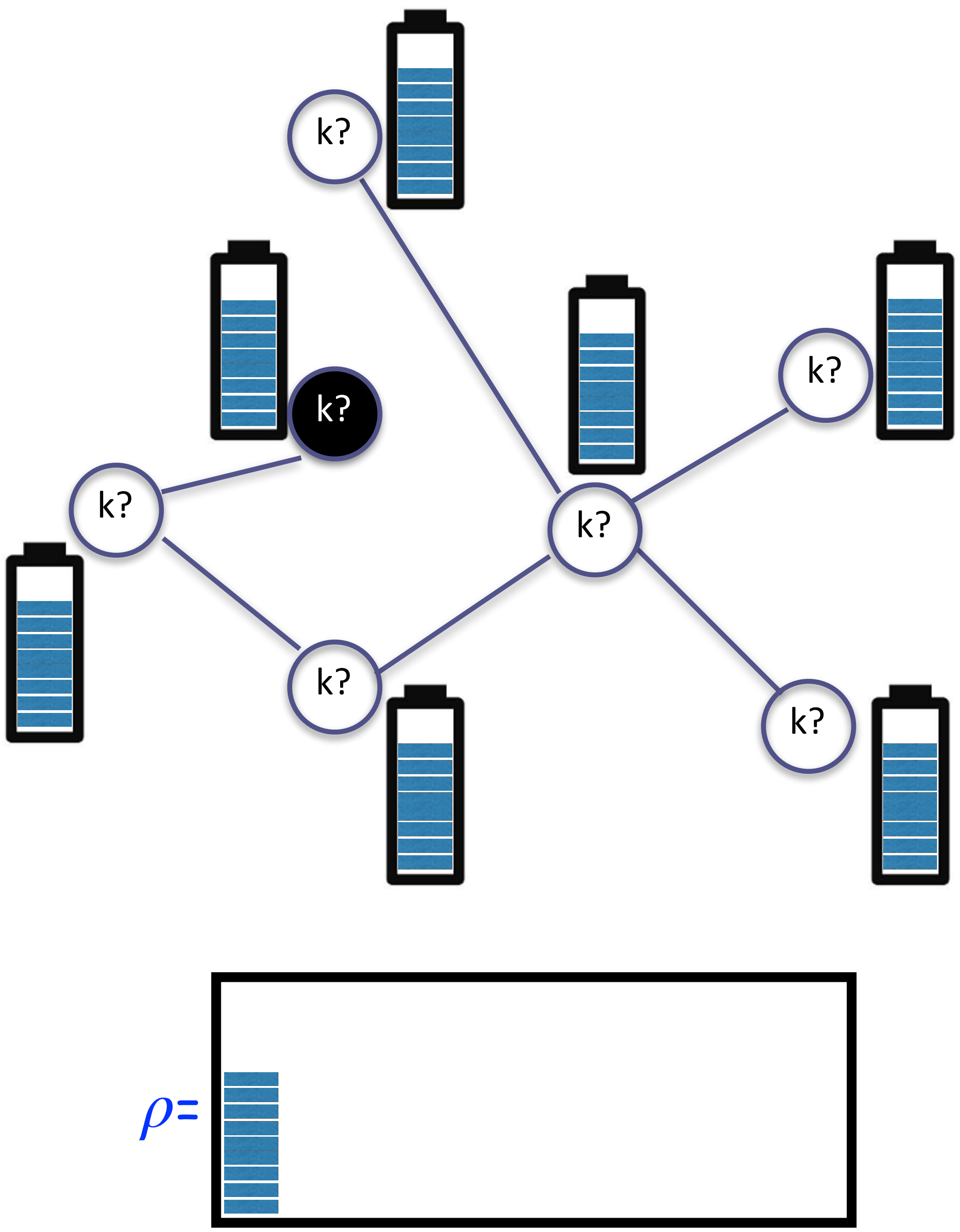
$r(k)$ rounds:

(to "average" the current potentials Φ)

mass distribution:

- broadcast Φ and receive neighbors' Φ_i
- $\Phi = \Phi + \sum_{i \in N} \Phi_i / d(k) - |N| \Phi / d(k)$ truncated to $O(\log n)$ bits

- counted remove potential: $\rho = \rho + \Phi, \Phi = 0$



RMC Epoch Example

epochs:

- one for each estimate k
- initialize potentials: $\Phi_{uncounted} = \ell, \Phi_{counted} = 0$

$p(k)$ phases:

(to let counted remove "enough" potential ρ)

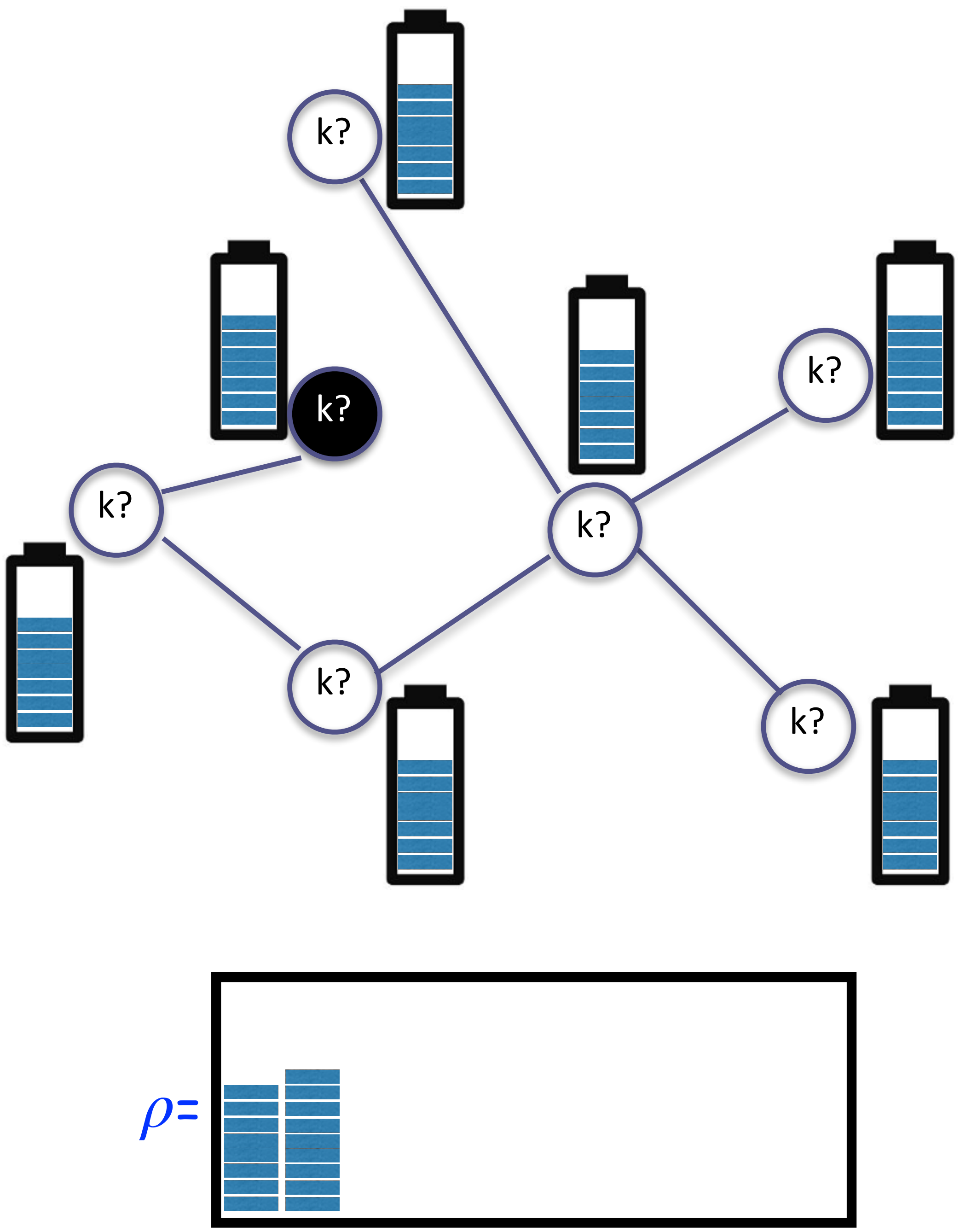
$r(k)$ rounds:

(to "average" the current potentials Φ)

mass distribution:

- broadcast Φ and receive neighbors' Φ_i
- $\Phi = \Phi + \sum_{i \in N} \Phi_i / d(k) - |N| \Phi / d(k)$ truncated to $O(\log n)$ bits

- counted remove potential: $\rho = \rho + \Phi, \Phi = 0$



RMC Epoch Example

epochs:

- one for each estimate k
- initialize potentials: $\Phi_{uncounted} = \ell, \Phi_{counted} = 0$

$p(k)$ phases:

(to let counted remove "enough" potential ρ)

$r(k)$ rounds:

(to "average" the current potentials Φ)

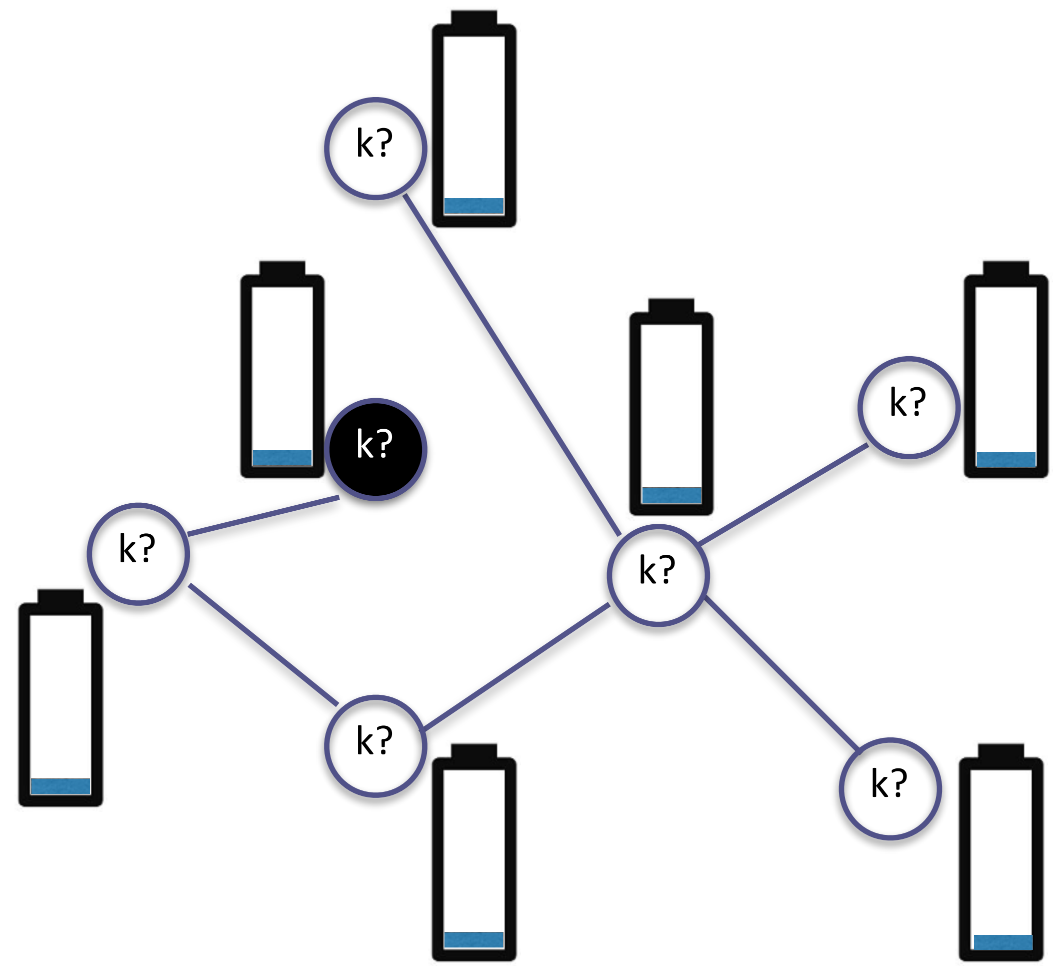
mass distribution:

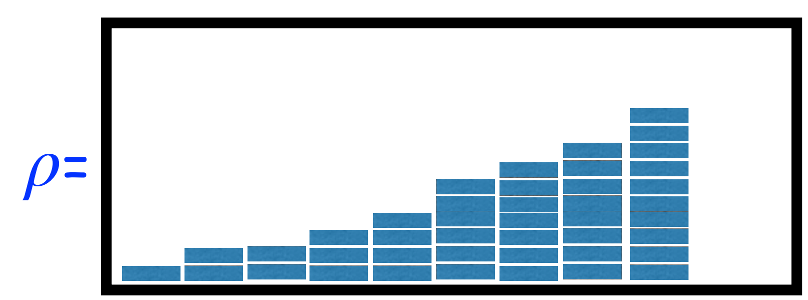
- broadcast Φ and receive neighbors' Φ_i
- $\Phi = \Phi + \sum_{i \in N} \Phi_i / d(k) - |N| \Phi / d(k)$ truncated to $O(\log n)$ bits

- counted remove potential: $\rho = \rho + \Phi, \Phi = 0$

- counted decide according to ρ
- counted notify if $k \geq n$
- try next k if needed

After $p(k)$ phases…





RMC theory and practice

RMC is simple (light implementation),

works with restricted resources, and tolerates disconnections

$\Longrightarrow$ suitable for A-VANET computations, but!

… the current theoretical analysis applies to adversarial dynamicity:

$$\tilde{O}\left(\frac{n^{1+2T(1+\epsilon)}}{\ell i_{\min}^2}\right) \Rightarrow \tilde{O}(n^{9+10\epsilon}) \text{ , for } T = 5,\ i \in \Theta(n),\ \ell \in O(1).$$

On the other hand, lower bounds known:

$$\Omega(\log n) \qquad \text{and} \qquad \Omega(\mathscr{D})$$

Large polynomial gap!! $\Longrightarrow$ we evaluate experimentally on real traffic.

<u>Hypothesis</u>: large polynomial speed-ups on real traffic.

Simulation Techniques

epochs:

- one for each estimate k
- initialize potentials: $\Phi_{uncounted} = \ell, \Phi_{counted} = 0$

$p(k)$ phases:

change # phases function until incorrect

(to let counted remove "enough" potential ρ)

$r(k)$ rounds:

change # rounds function until incorrect

(to "average" the current potentials Φ)

mass distribution:

- broadcast Φ and receive neighbors' Φ_i
- $\Phi = \Phi + \sum_{i \in N} \Phi_i / d(k) - |N| \Phi / d(k)$ truncated to $O(\log n)$ bits

- counted remove potential: $\rho = \rho + \Phi, \Phi = 0$

- counted decide according to ρ
- counted notify if $k \geq n$
- try next k if needed

change threshold functions until incorrect

Simulations

Input topologies:

- Extracted from traces of taxi trips in NYC with proximity defined by street location. (Massive database, technical challenge.)
- “Enhanced” path with proximity defined by short reachability. (Simulates highways.)

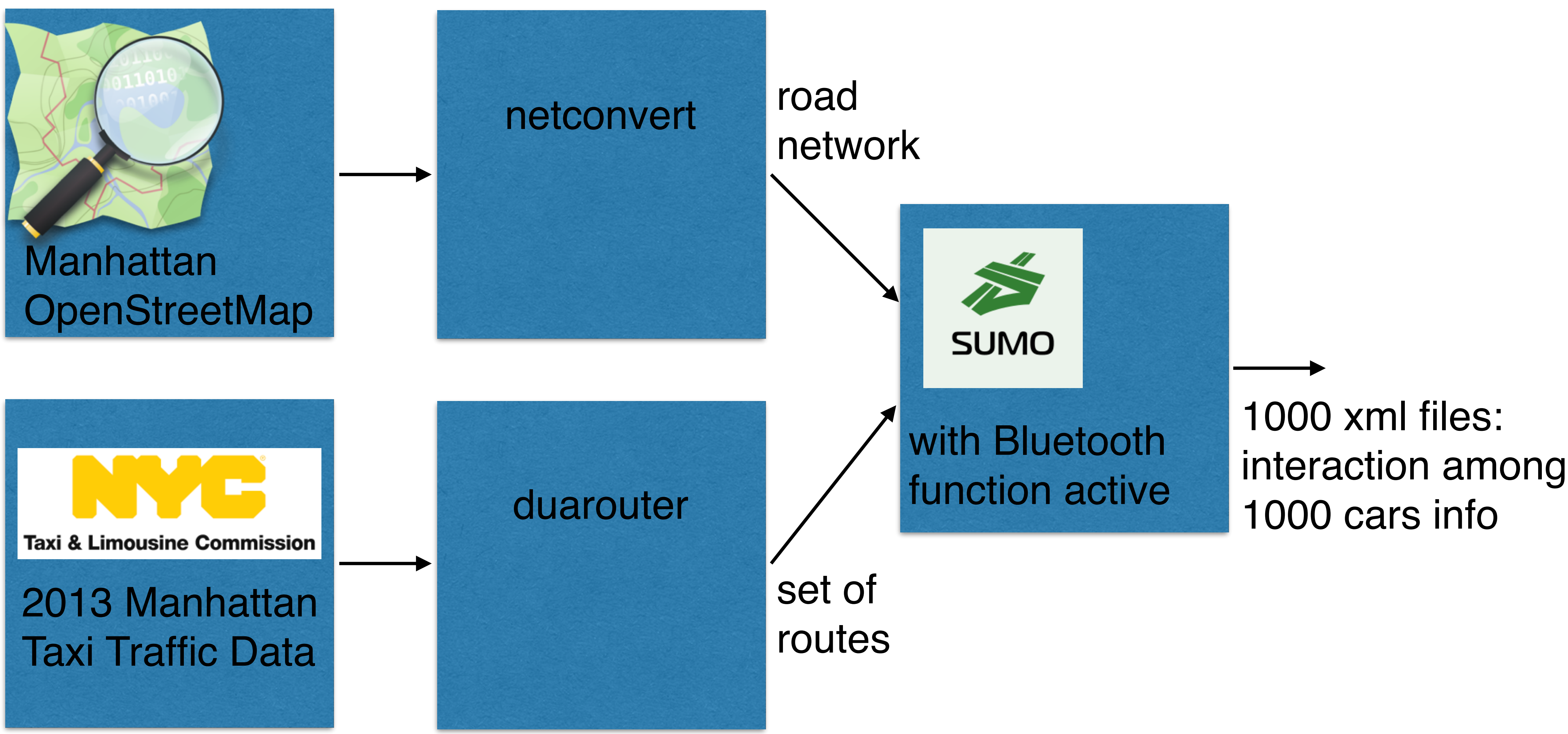
Parameters:

- $n = \{128, 256, 500, 600, \ldots, 1000\}$ network size
- $\ell = \{30, n/4 - 1\}$ counted nodes
- $T = \{1, 2.5, 5, 11\}$ connectivity
- Other algorithm parameters and asymptotic notation constants fixed to small values.

Evaluated against:

- RMC theoretical running time.

NYC Taxi Traffic Data Extraction



Results

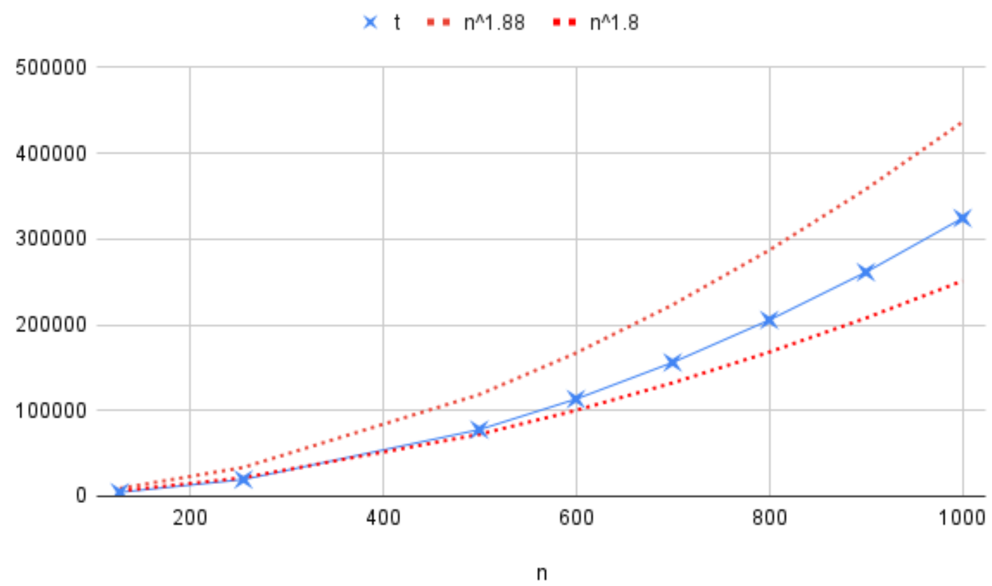


Fig. 1: NYC taxi traffic input for $\ell = 30$ and $\mathcal{T} = 5$.
Dotted lines correspond to functions of n bounding the running time.

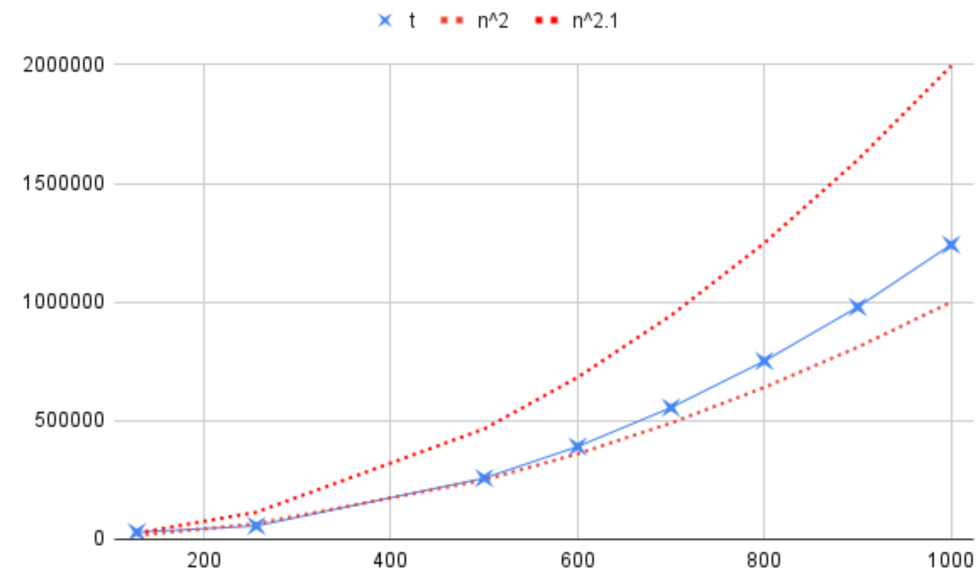


Fig. 2: Highway input for $\ell = 30$ and $\mathcal{T} = 11$.
Dotted lines correspond to functions of n bounding the running time.

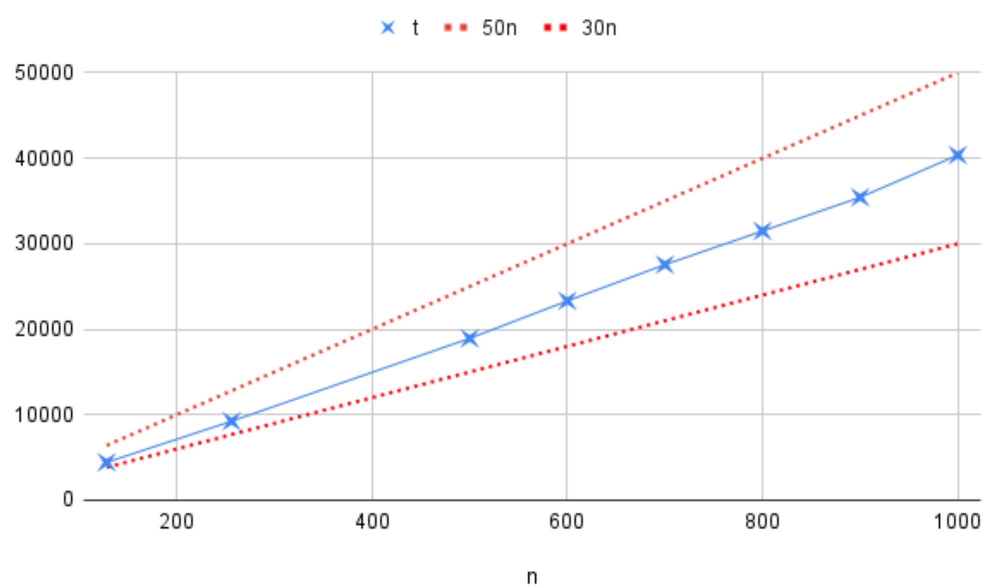


Fig. 3: NYC taxi traffic input for $\ell = n/4 - 1$ and $\mathcal{T} = 5$.
Dotted lines correspond to functions of n bounding the running time.

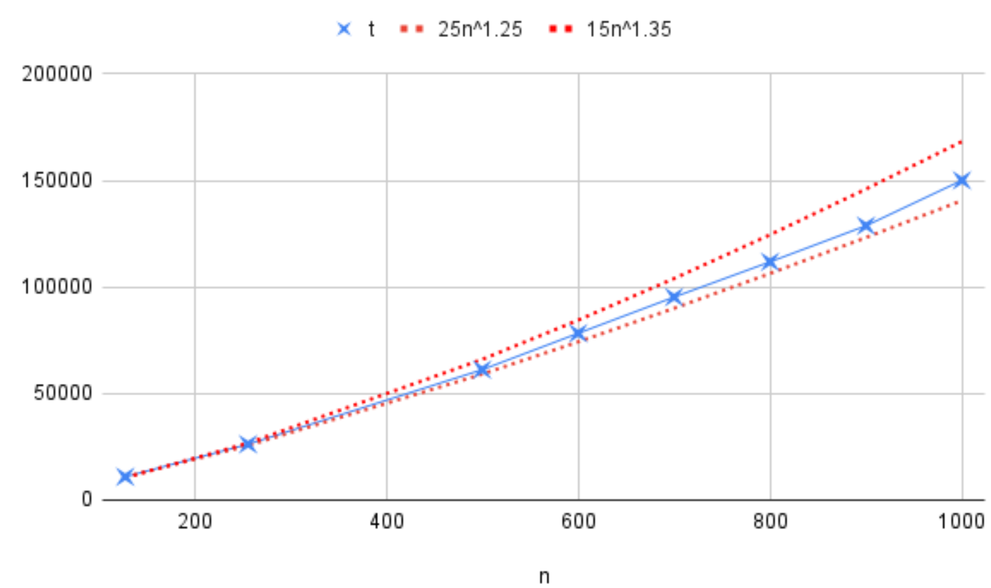


Fig. 4: Highway input for $\ell = n/4 - 1$ and $\mathcal{T} = 11$.
Dotted lines correspond to functions of n bounding the running time.

In the topologies tested, RMC is $\approx n^7$ times faster than the worst-case theoretical running time, confirming our hypothesis.

Thank you!

mmosteiro@pace.edu