Understanding RFID Counting Protocols

Binbin Chen[#] Ziling Zhou[^#] Haifeng Yu[^]

Advanced Digital Sciences Center
^ National University of Singapore

MobiCom 2013

Many applications need counting

RFID technology
enables
large-scale counting

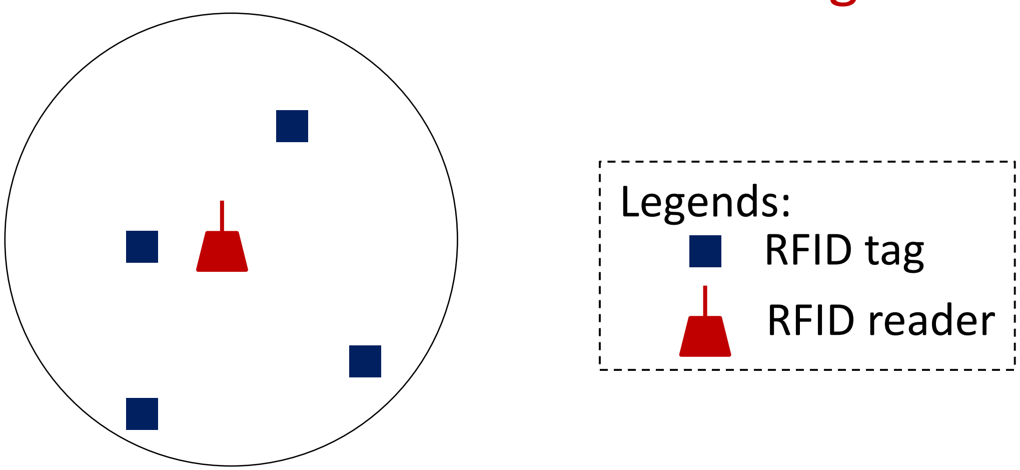
RFID counting problem

(a simple single-set version)

- One reader and n tags
- They run a protocol to get an $\hat{n} \approx n$
  - Getting the exact n is expensive $\Longrightarrow$ Randomization helps
- Guarantee: $|\hat{n} - n| \leq \varepsilon n$ holds (say, with 90% probability)
  - Here, ε bounds the relative error

See paper for generalizations:

e.g., a reader moves around to extend coverage



Existing RFID counting research

- An impressive arsenal of techniques

Protocol	Venue
UPE	MobiCom'06
EZB	INFOCOM'07
LOF	PerCom'08 / TPDS'11
(Enhanced) FNEB	INFOCOM'10
PET	ICDCS'11 / TMC'12
ART	MobiCom'12
ZOE	INFOCOM'13

- The central design goal:

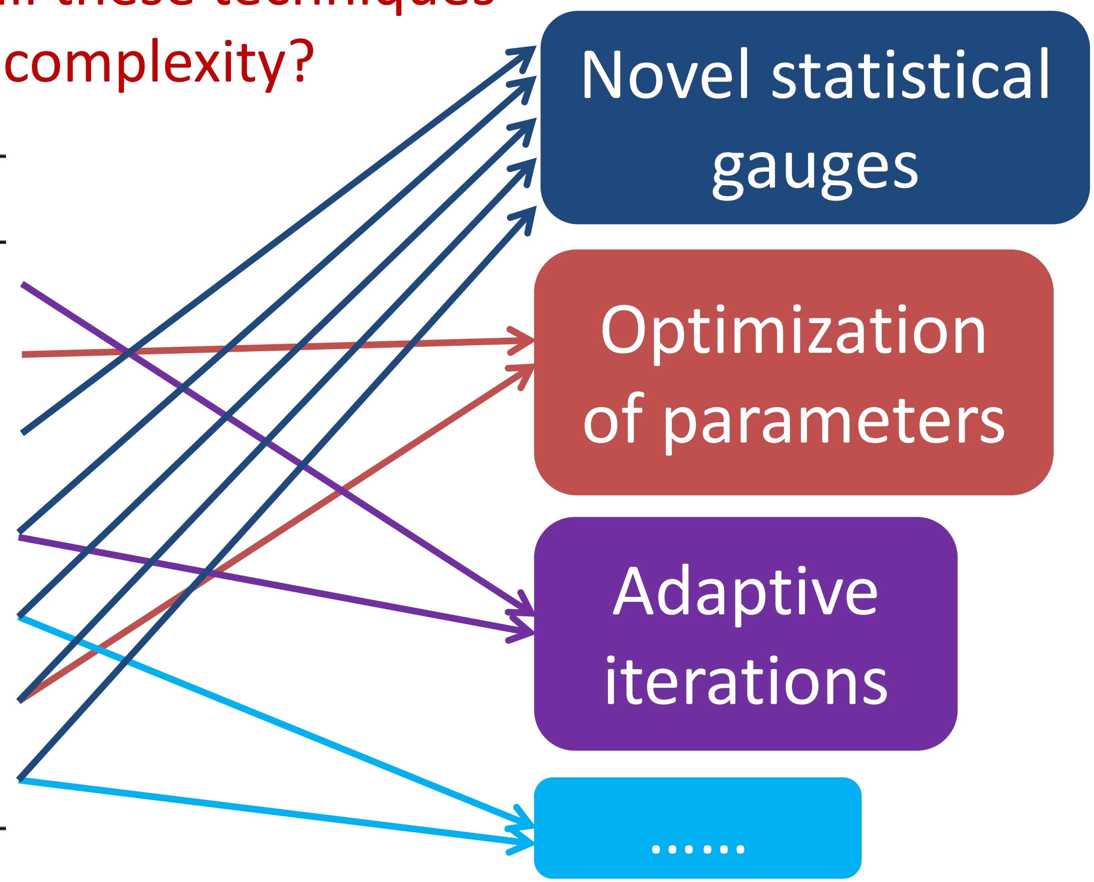
  Reduce ***time overhead*** & provide the guarantee

Call for fundamental understanding

- Diverse views on which design aspects are important

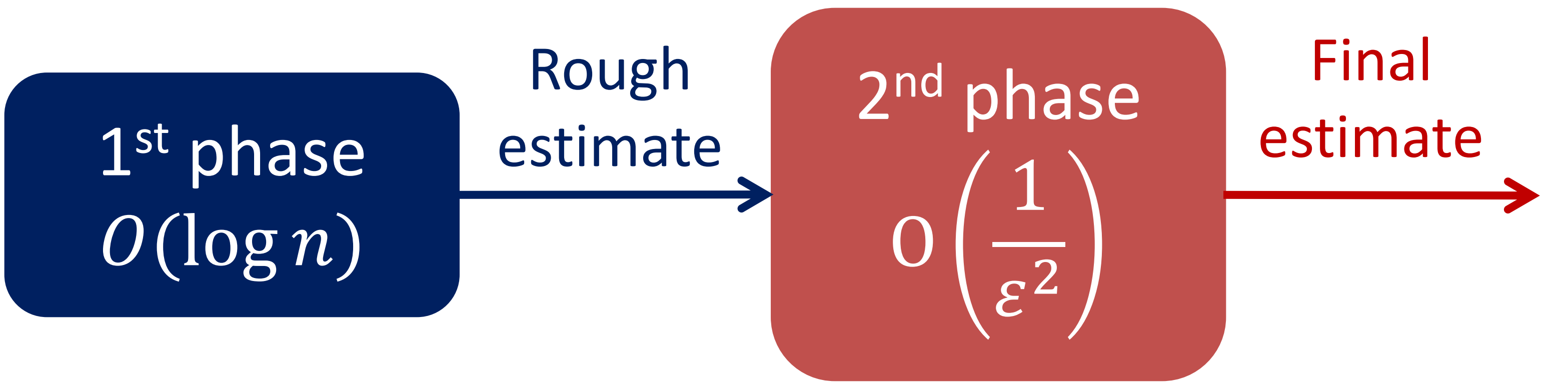
Should we combine all these techniques despite the resulting complexity?

Protocol
UPE
EZB
LOF
(Enhanced) FNEB
PET
ART
ZOE



Our central thesis for RFID counting

The *overlooked* key is to have two phases:



Other techniques proposed in the literature are less important than originally thought

Note:

- the $\log n$ term can be reduced to a $\log\log n$ term

The inspiration

- Novel **lower bounds** for RFID counting protocols:

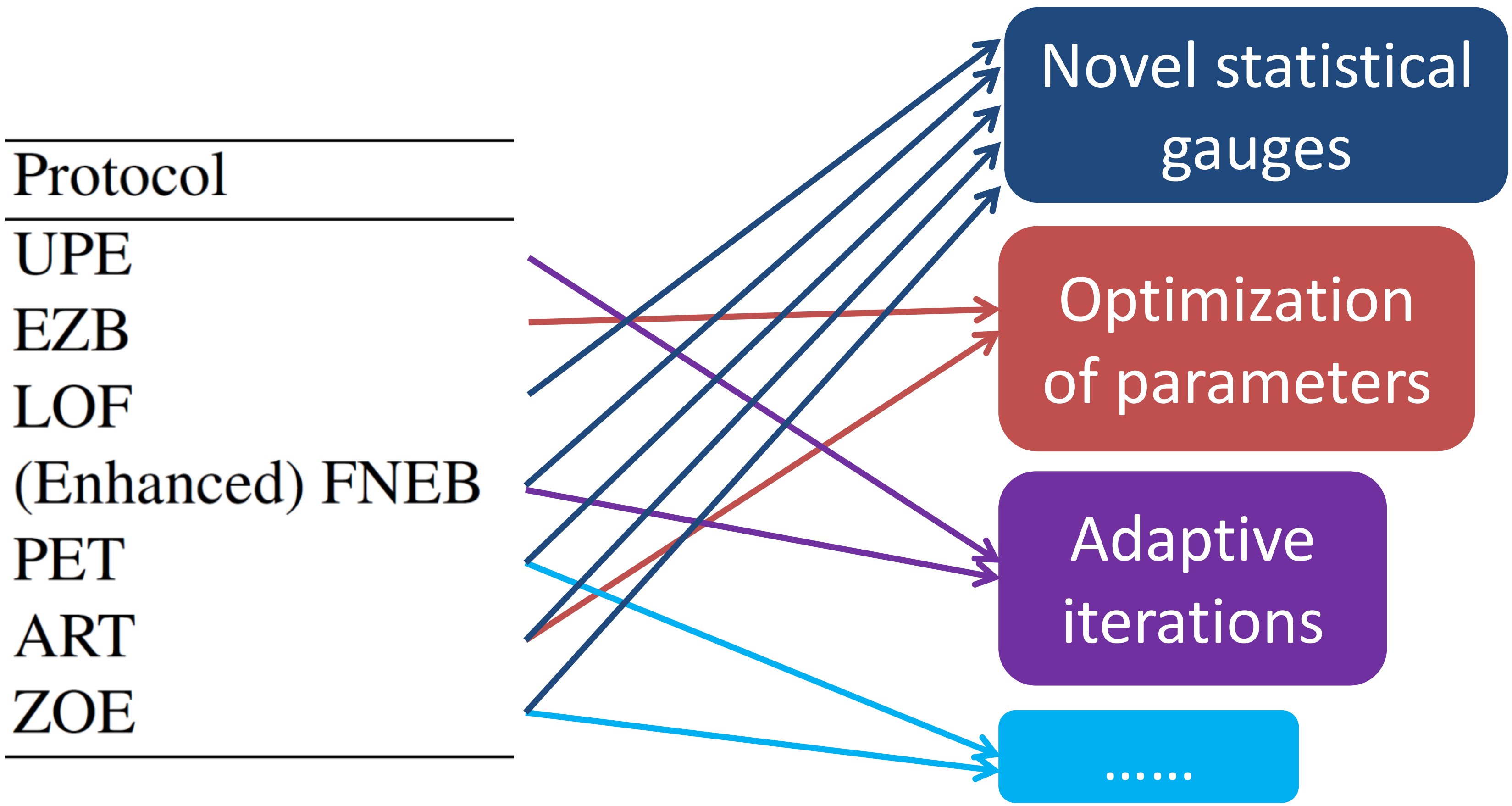
> *(Rough)* ***Theorem***:
> For single-set RFID counting, no protocol can estimate with $< \varepsilon$ relative error while incurring $o\left(\log\log n + \frac{1}{\varepsilon^2 log\frac{1}{\varepsilon}}\right)$ overhead

- Our proof leverages a recent breakthrough result in communication complexity [Chakrabarti & Regev, STOC'11]
  - They proved GHD (Gap Hamming Distance) problem is hard
  - We connect RFID counting and GHD by a novel reduction
  - Details in paper

Validating our thesis

- Examine the importance of other techniques
- Apply our thesis to design better protocols

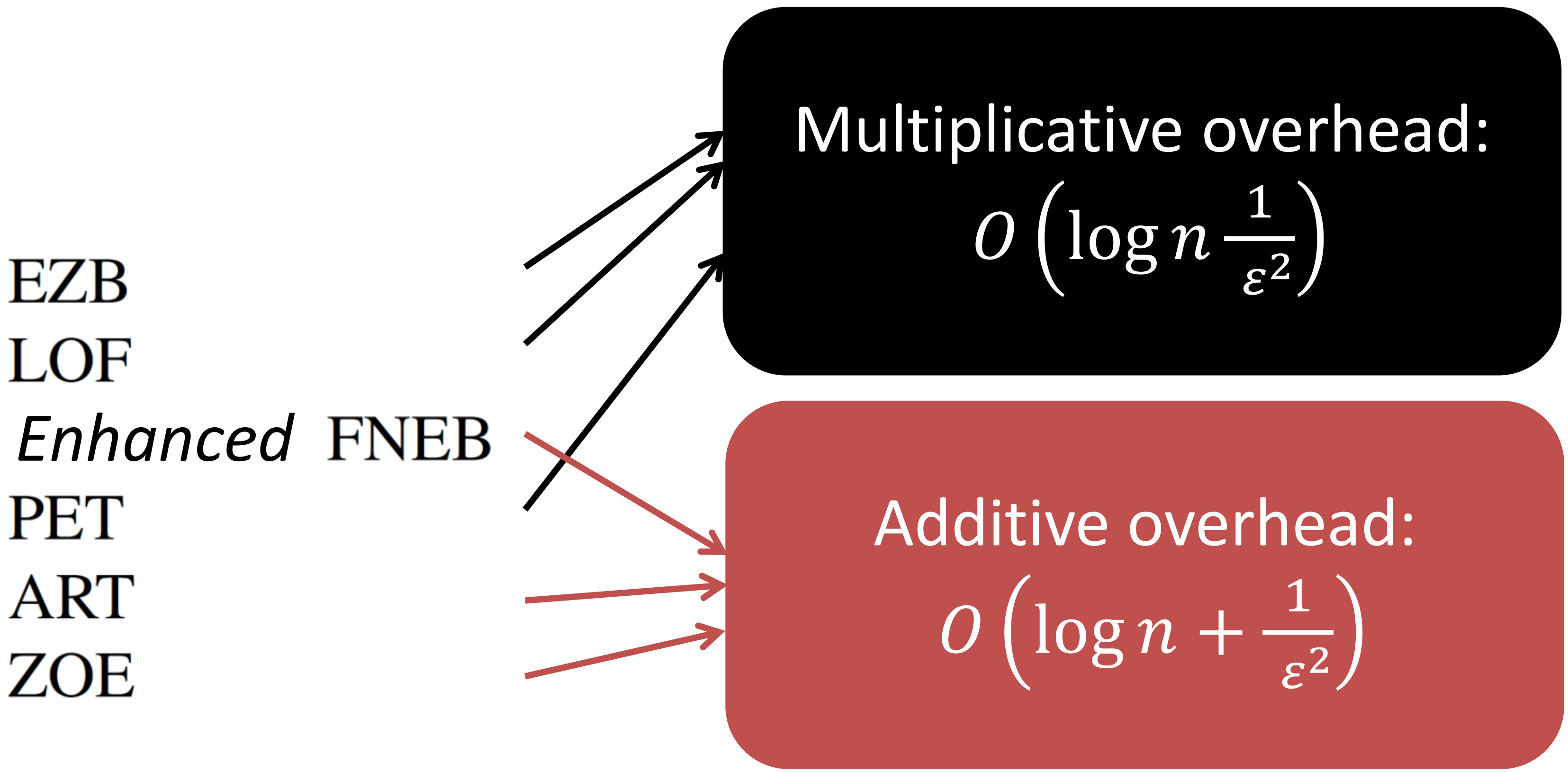
Existing literature:
diverse views about what are important



Let us step back, and take an asymptotic view of existing protocols

Such a comparison has not been done before

Two distinct groups



Note:

- Some protocols reduce the $\log n$ term to a $\log\log n$ term

Additive-overhead protocols are better

Enhanced FNEB

ART

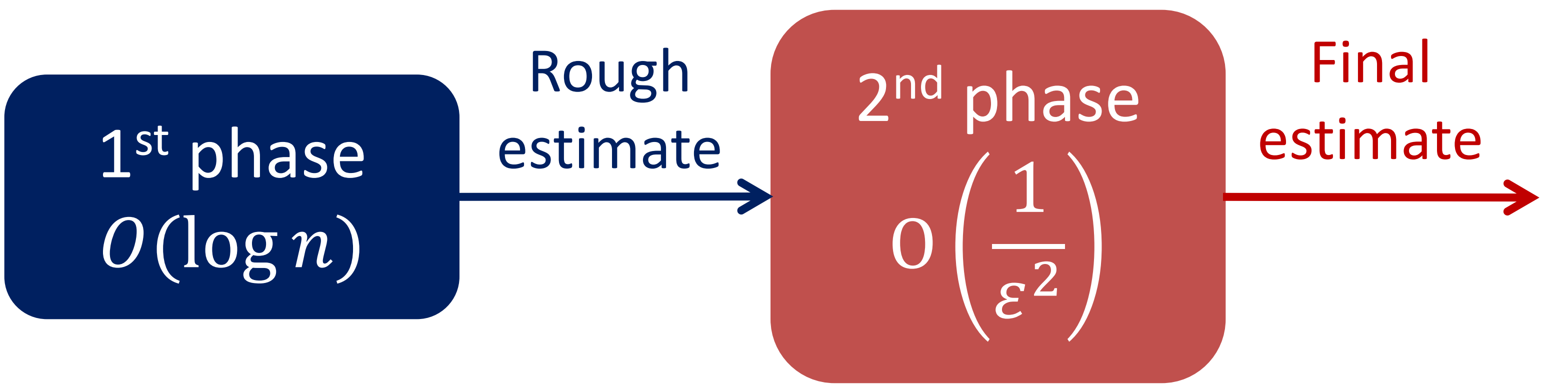
ZOE

Additive overhead:

$$O\left(\frac{1}{\varepsilon^2} + \log n\right)$$

How they achieve additive overhead?

- Despite their many differences (as originally emphasized), they all have a two-phase design:



Our thesis has not been discovered

Enhanced FNEB ('10)	Use of a **novel gauge**: the indices of the first non-empty slots
ART ('12)	Use of a **novel gauge**: the average run length of non-empty slots
ZOE ('13)	i) Unique design about the **gauge**: Each trial has a single slot ii) Two-phase design

They also employ other interesting techniques:

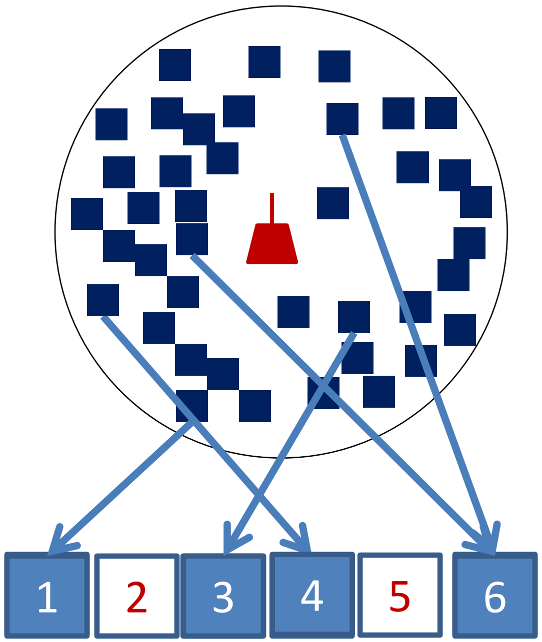
– involved optimizations, adaptive iterations …

Are these other techniques important?

Let us focus on the ***gauges***

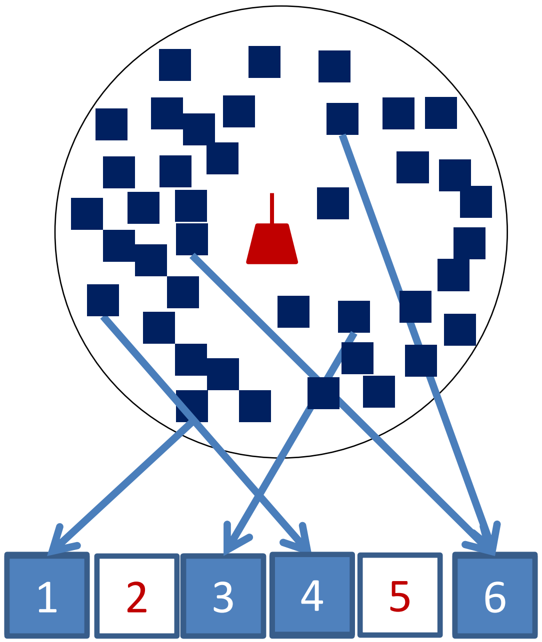
An old gauge of the early EZB ('07) protocol

- # of empty slots
  - More empty slots $\Longrightarrow$ less tags



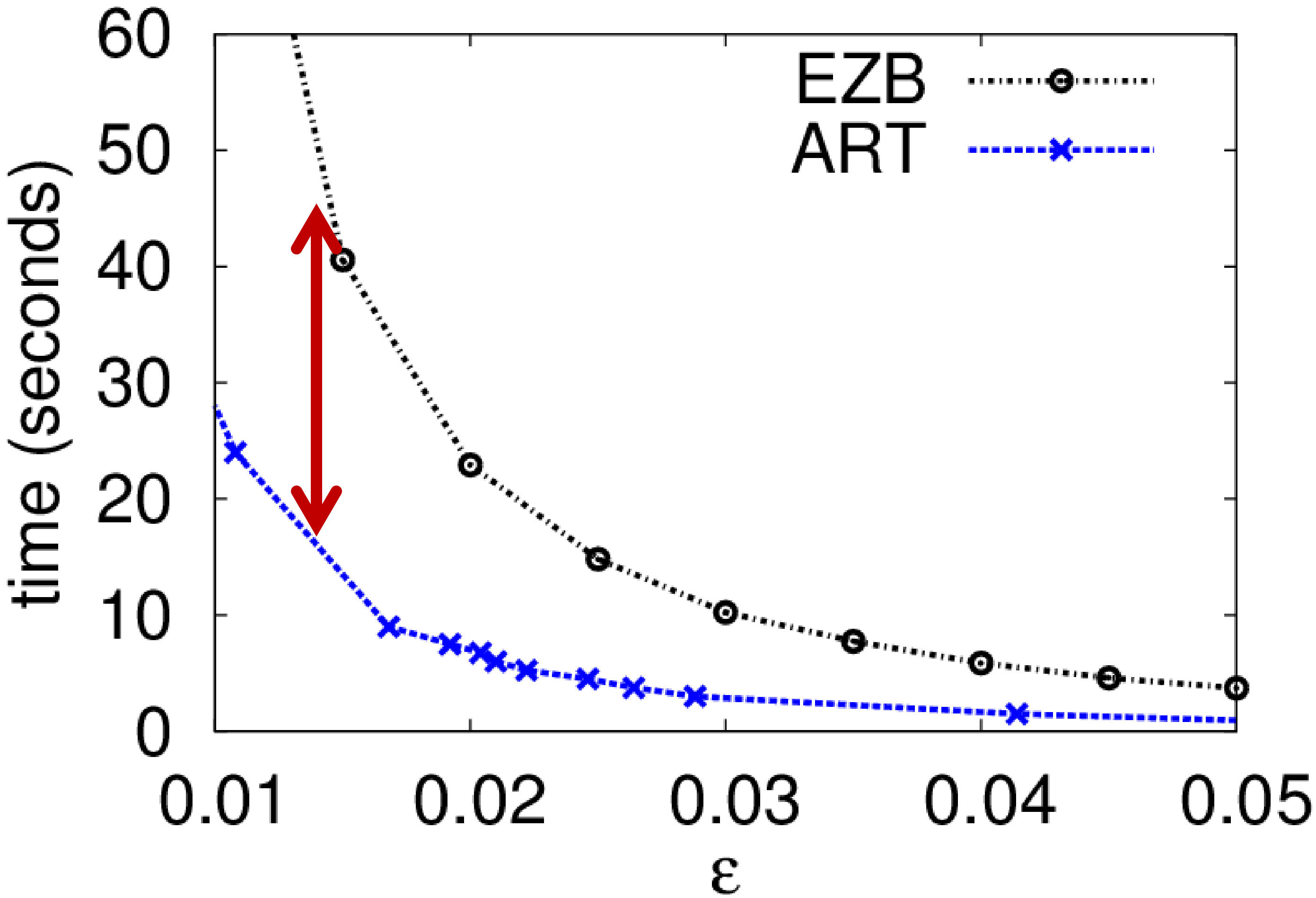
The novel gauges

- ART: average run length of non-empty slots
  - In the example: (1+2+1)/3
- FNEB: index of the first non-empty slot
- ZOE: still # of empty slots, but each slot is independent



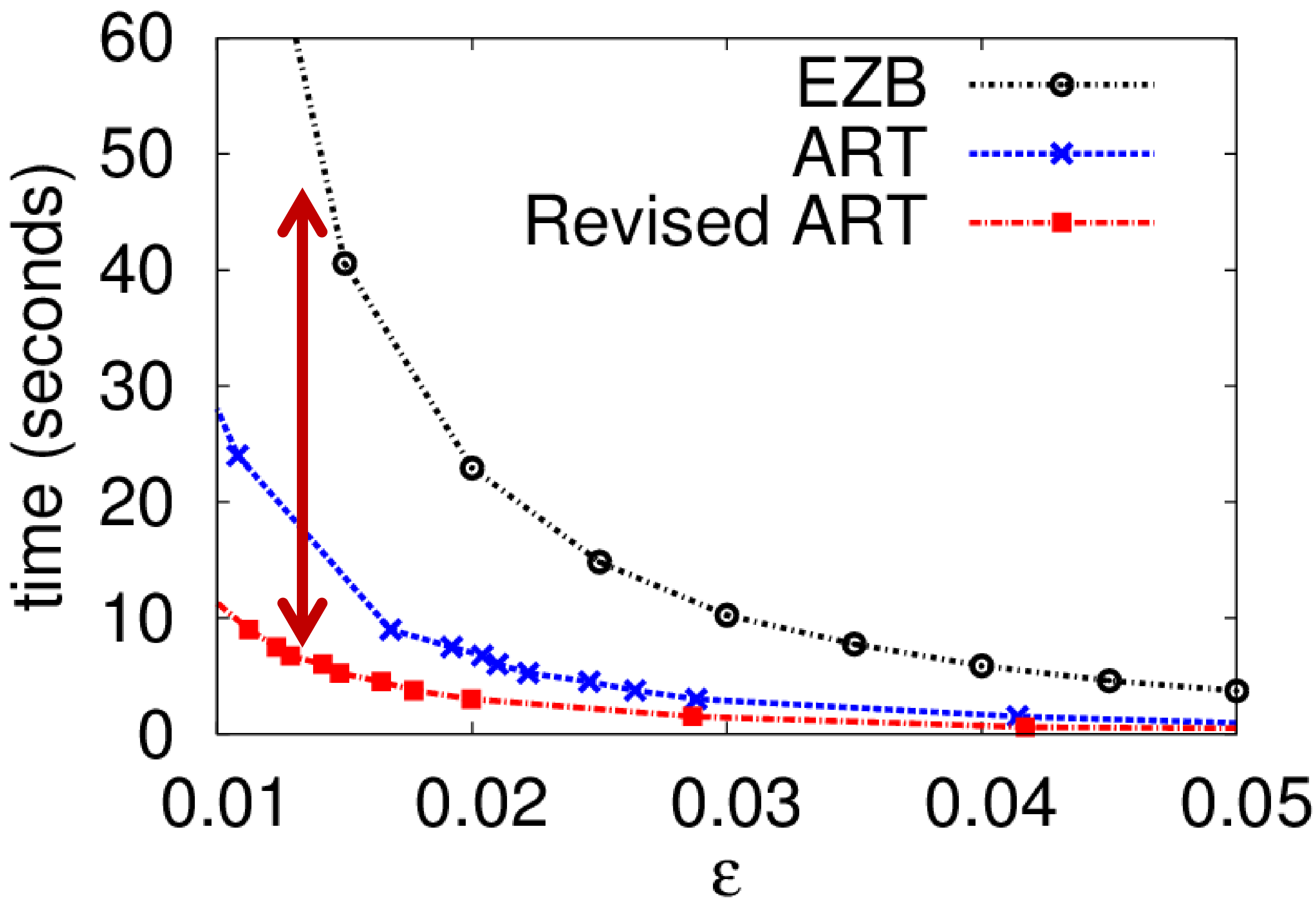
Let us examine ART's ('12) performance gain

(over the early EZB ('07) protocol)



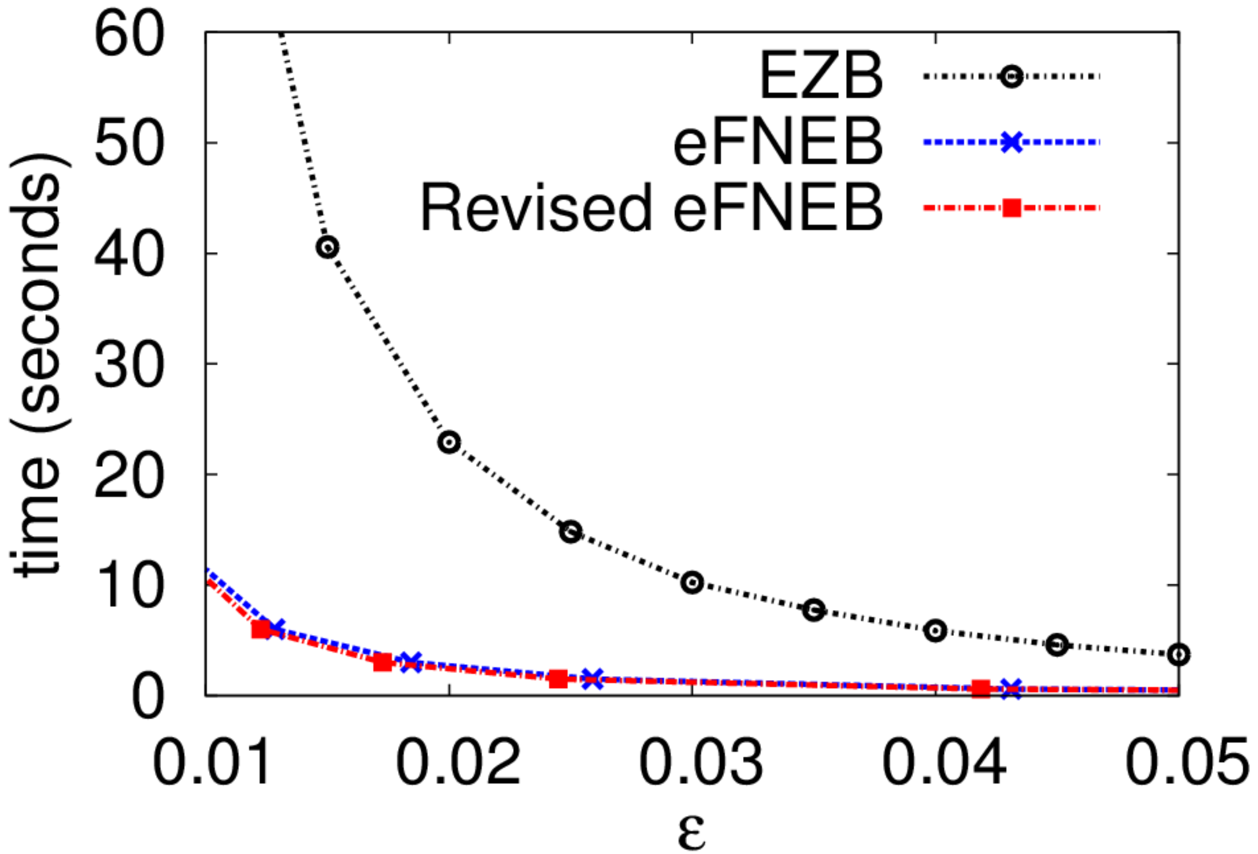
Replace ART's ('12) gauge by the old EZB's ('07) gauge

We keep everything else unmodified



Similarly …

- FNEB's gauge seems not help



- Neither does ZOE's

Validating our thesis

- Examine the importance of other techniques
- Apply our thesis to design better protocols

SRC_S: a Simple RFID Counting protocol *for single-set counting*

- The design of SRC_S is solely driven by our thesis:
  - It applies the 2-phase design
  - It uses simple & basic building blocks in all other aspects
    we claim no novelty for these building blocks

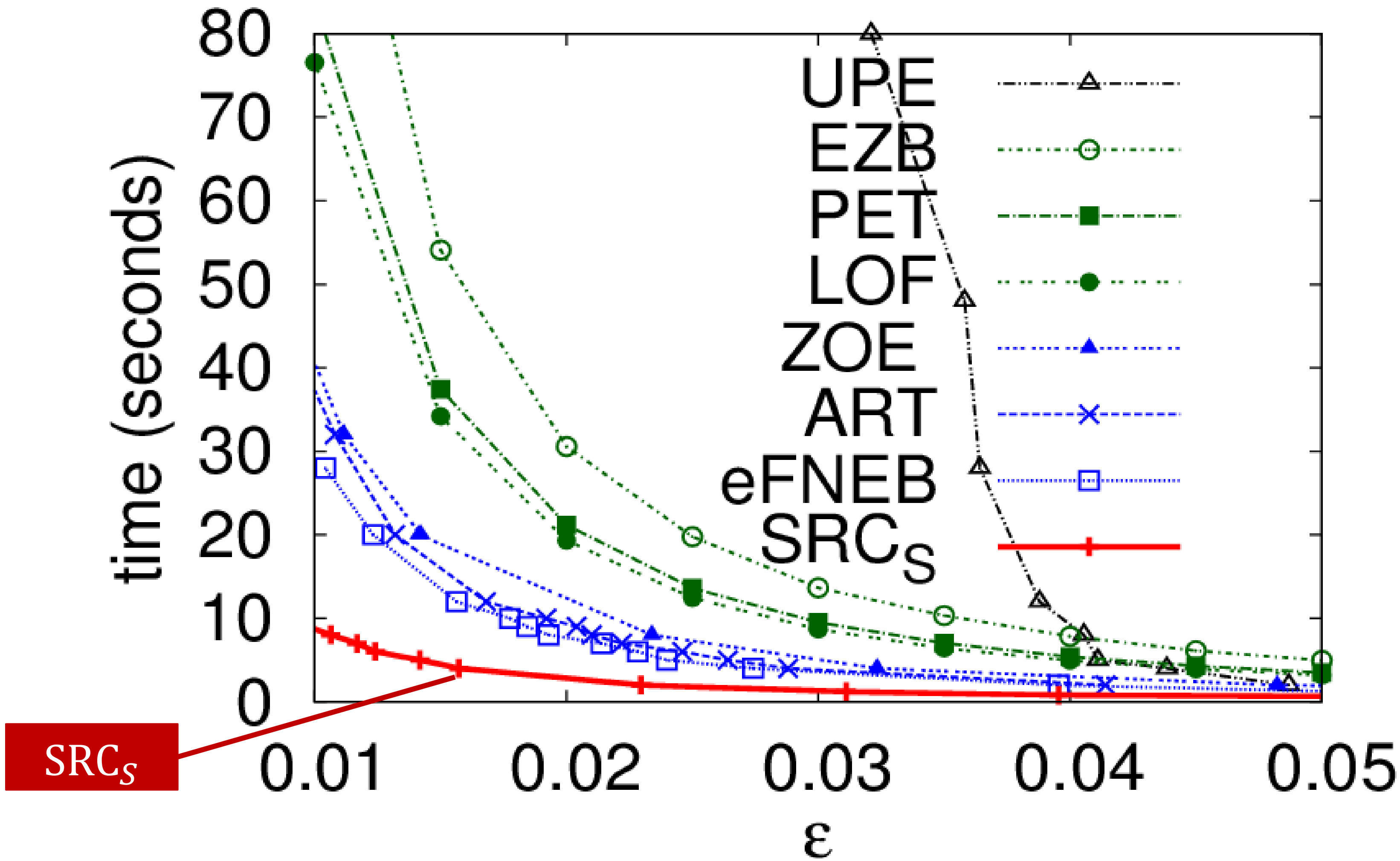
SRC_S pseudo-code:

1: Invoke a simple early protocol (LOF '08) to get a rough estimate $\tilde{n}$;

2.1: calculate tag-responding probability according to $\tilde{n}$;

2.2: Use a simple early gauge (EZB '07) to obtain the final estimate;

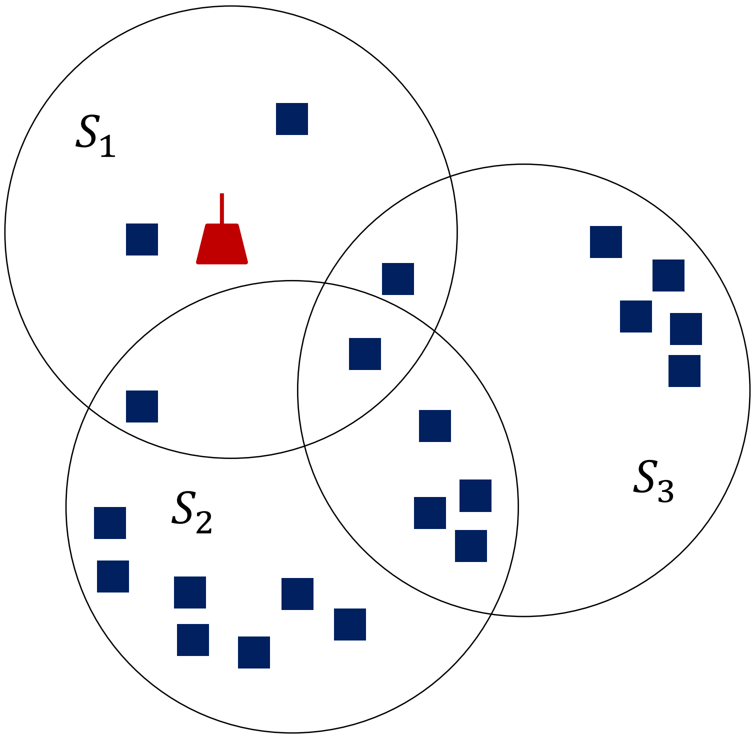
SRC_S is ≥ 100% faster



Note: We have done extensive experiments under different settings
Please see our paper for more details

How about multiple-set RFID counting?

- Consider a reader sequentially visits multiple locations to count # of tags in a large space
  - Here $n = |S_1 \cup S_2 \cup S_3|$: the sets can overlap



Apply our thesis

- Unlike single-set case, no one happens to use 2 phase
  - All existing protocols incur multiplicative overhead
  - Our thesis hints that big improvement might be possible

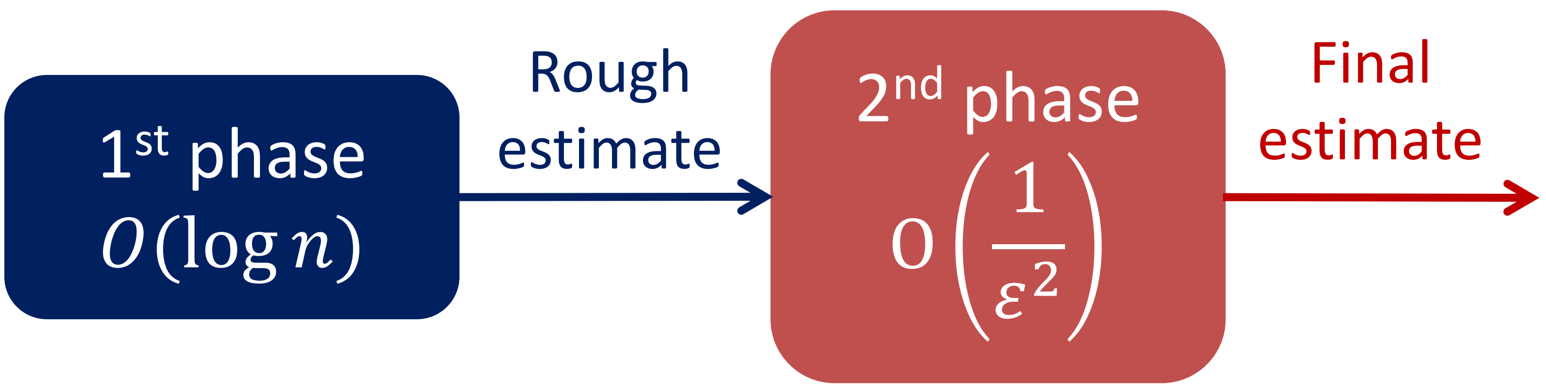
- Applying our thesis needs to overcome a challenge
  - The reader has no rough estimate of n until the last location

- Our SRC_M protocol uses some interesting techniques to overcome the challenge
  - It achieves additive overhead, and is $\geq 500\%$ faster

- Knowing the thesis is critical
  - It guides us to identify & focus on the key challenge

Summary

- Inspired by our RFID counting lower bound results, we find the overlooked key is a 2-phase design



- All other techniques are less important
- Our thesis leads to better protocols