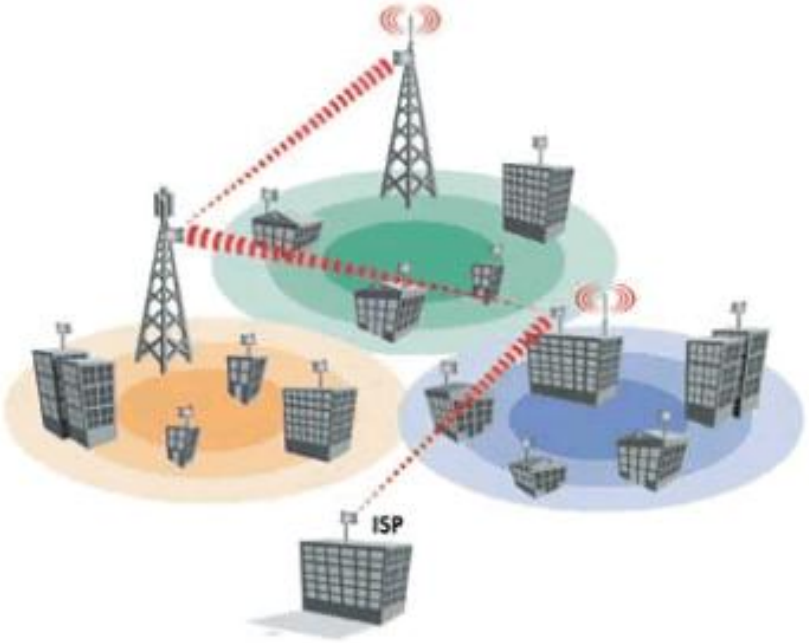


FAST: Realizing what your neighbors are doing

Lu WANG

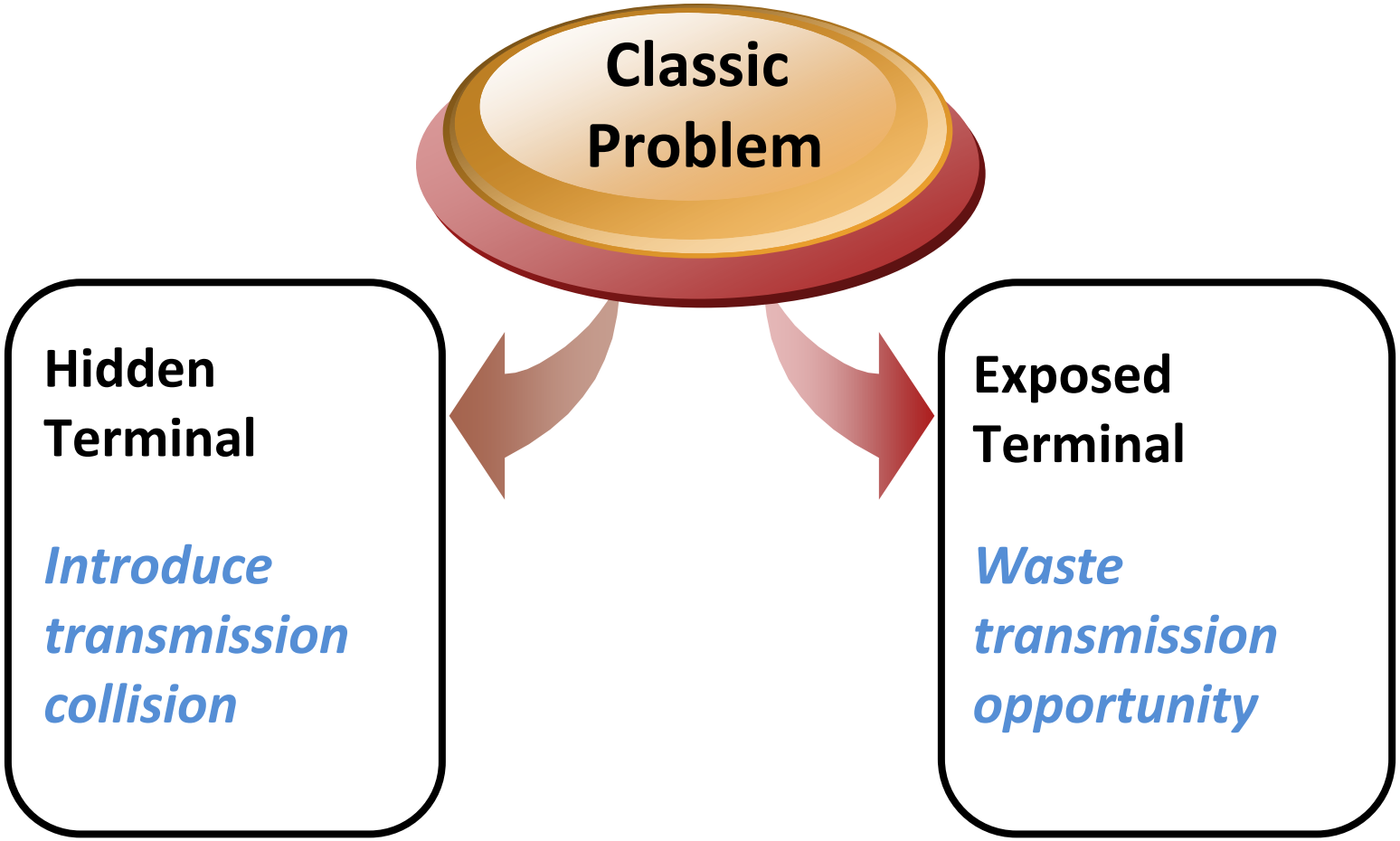
Computer Science and Engineering, HKUST

Jun 13, 2012

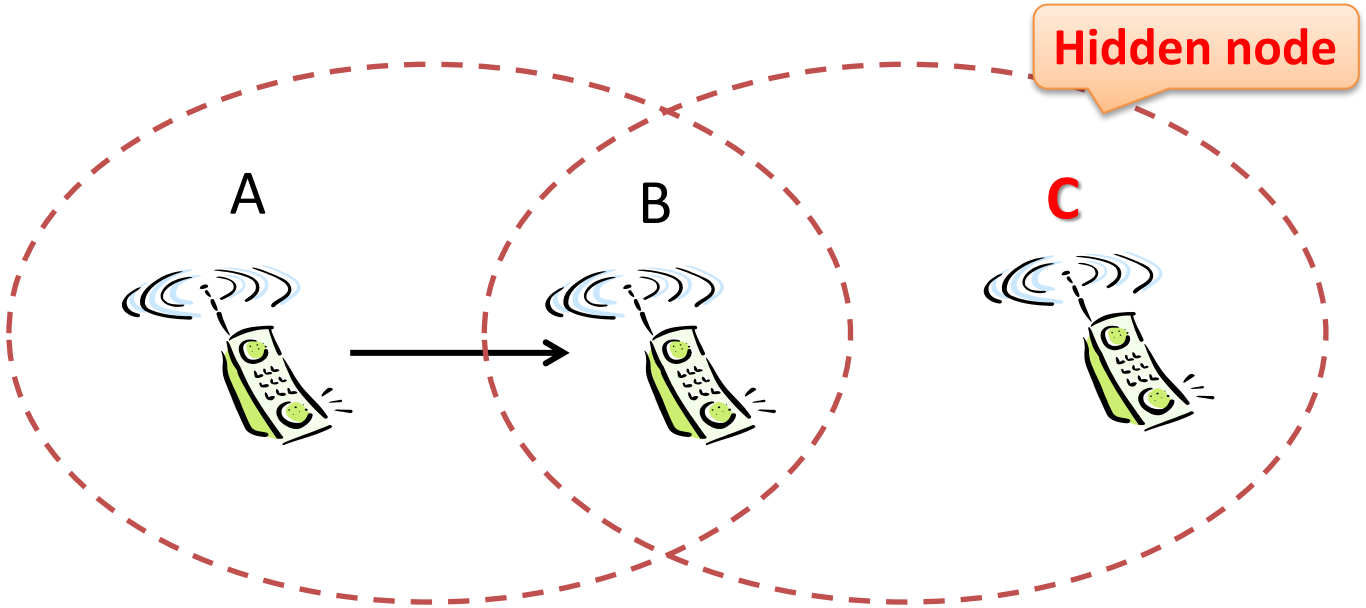
Roadmap

- **Introduction**
- Motivation
- FAST Design
- Performance Evaluation
- Conclusion

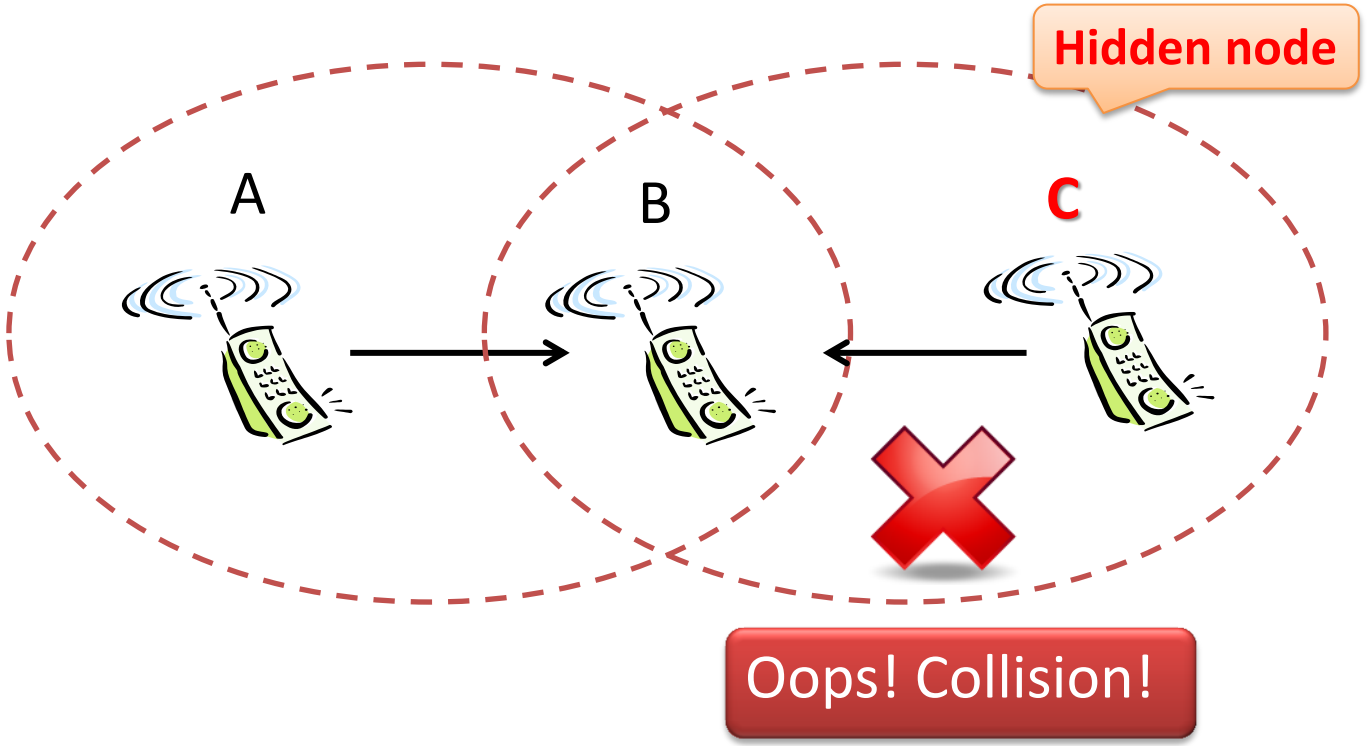
Introduction



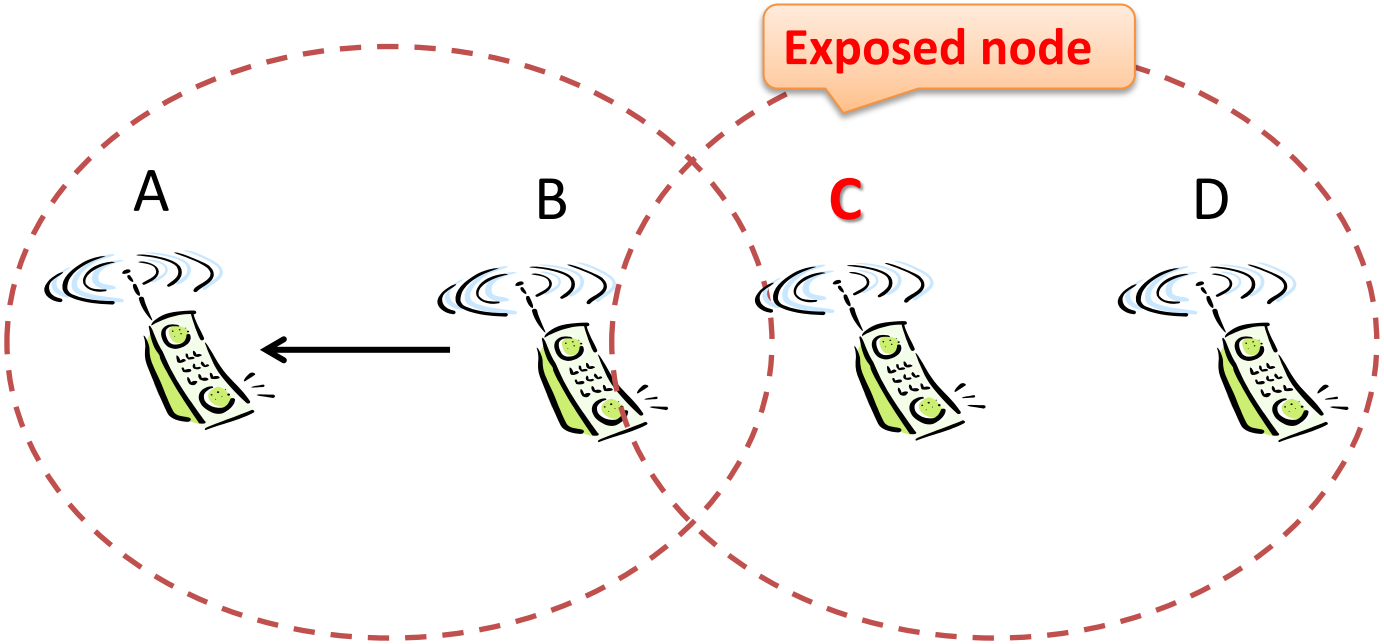
Hidden Terminal Problem



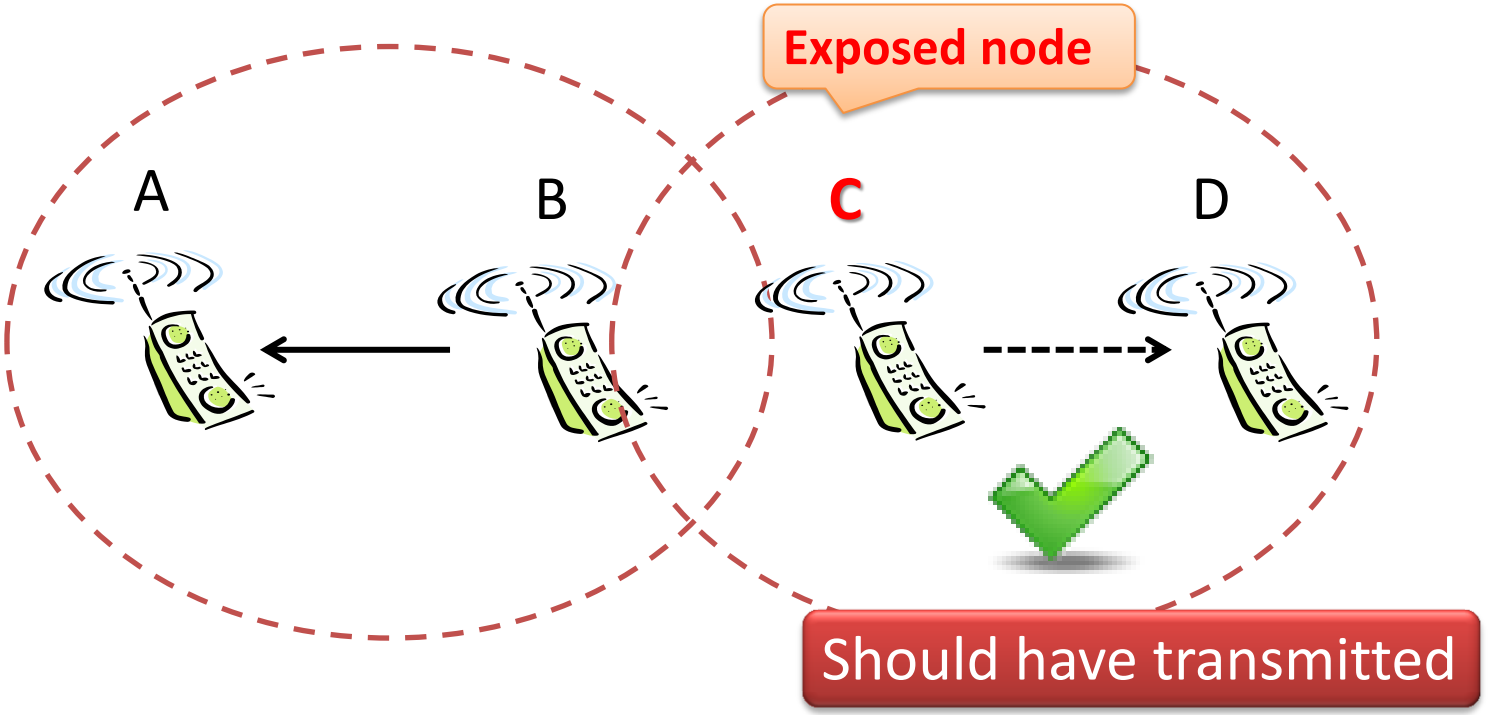
Hidden Terminal Problem



Exposed Terminal Problem



Exposed Terminal Problem



State-of-art Solution

CSMA/CA

Carrier Sense Multiple Access with Collision Avoidance

RTS/CTS handshake

RTS: Request to send

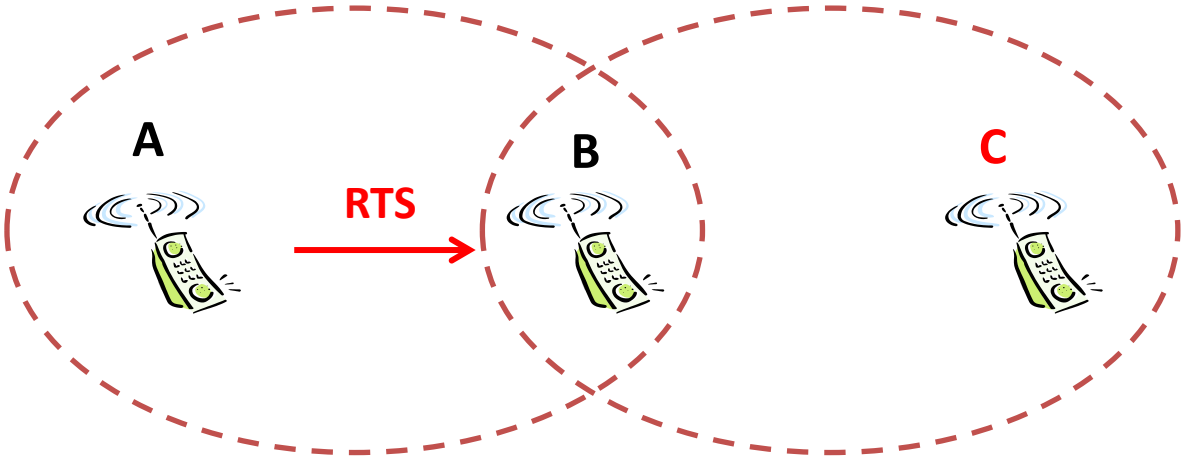
CTS: Clear to send

Provide Channel Usage Information

State-of-art Solution

CSMA/CA

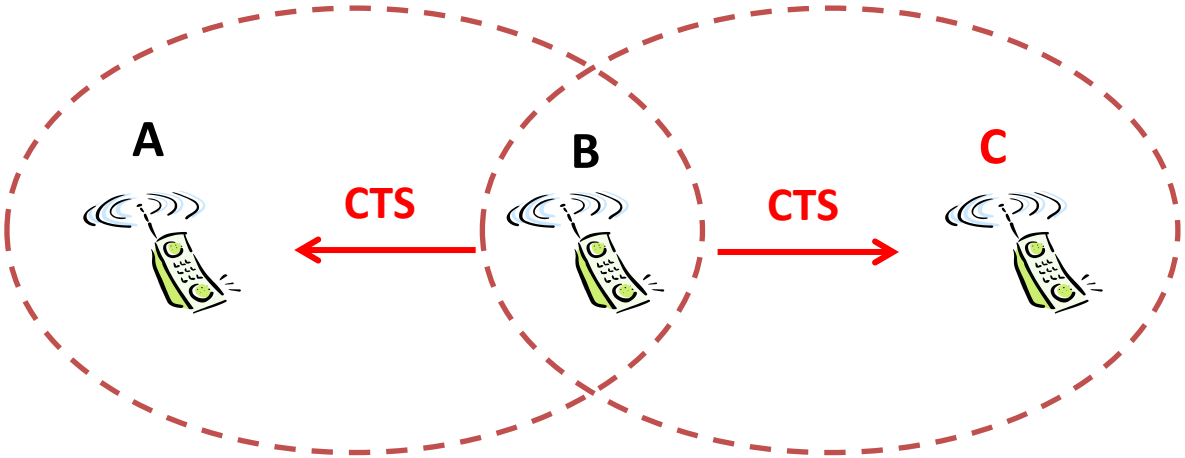
Carrier Sense Multiple Access with Collision Avoidance



State-of-art Solution

CSMA/CA

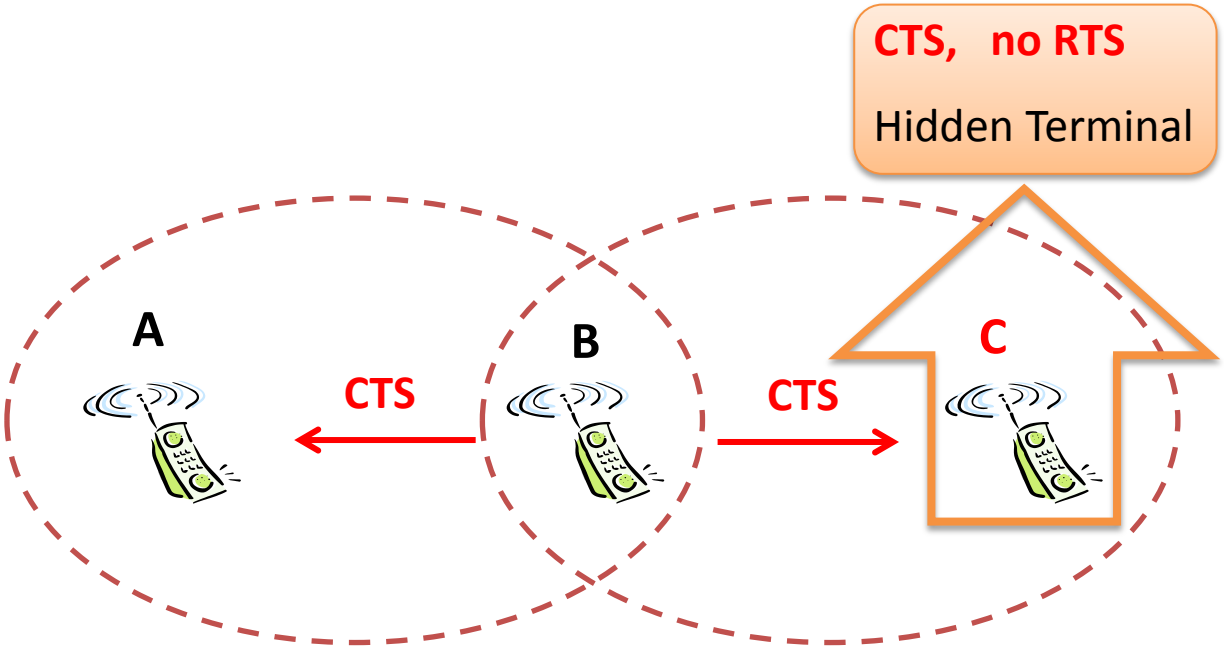
Carrier Sense Multiple Access with Collision Avoidance



State-of-art Solution

CSMA/CA

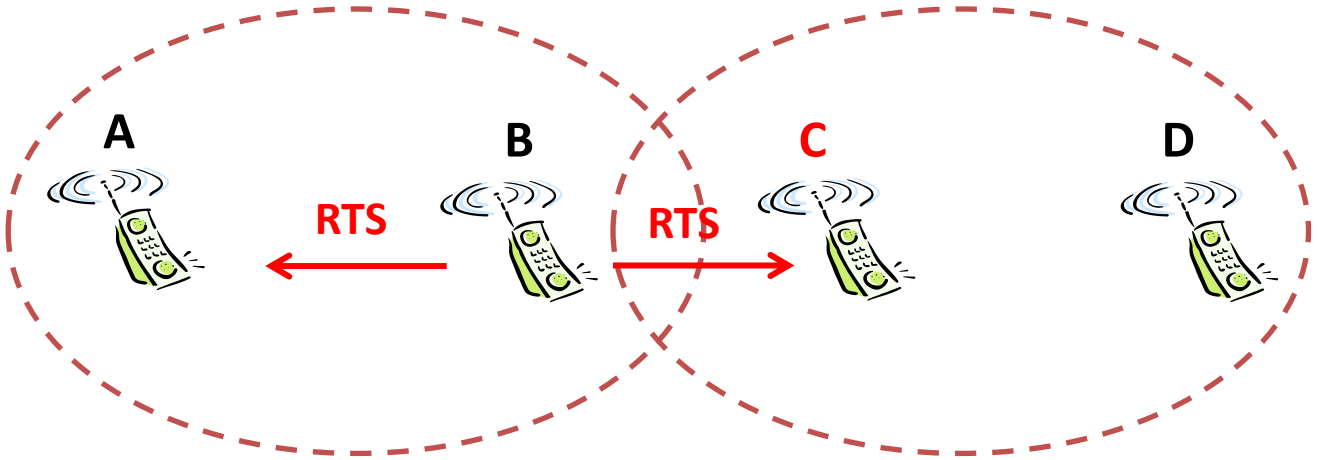
Carrier Sense Multiple Access with Collision Avoidance



State-of-art Solution

CSMA/CA

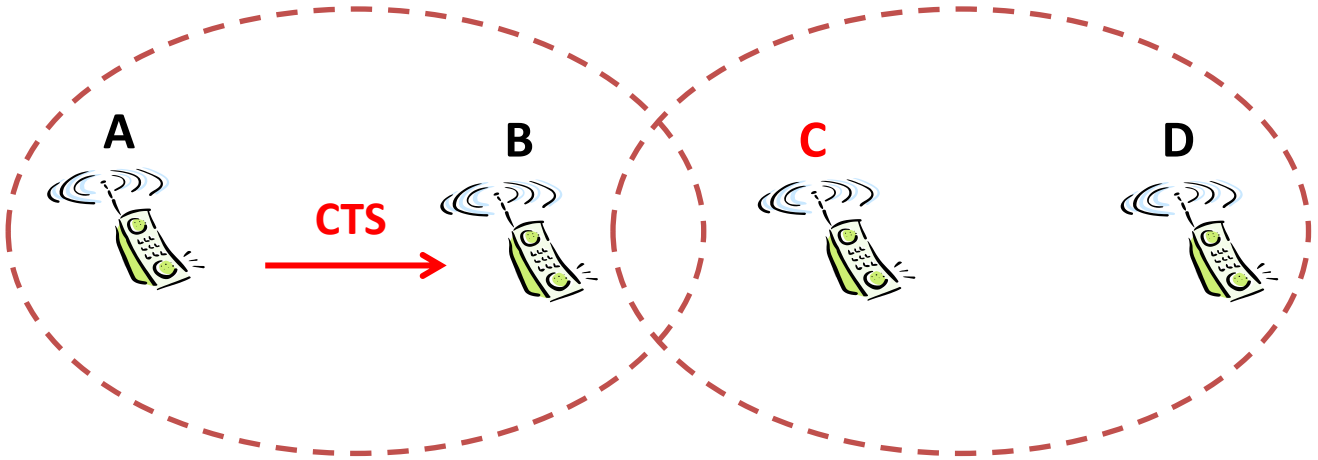
Carrier Sense Multiple Access with Collision Avoidance



State-of-art Solution

CSMA/CA

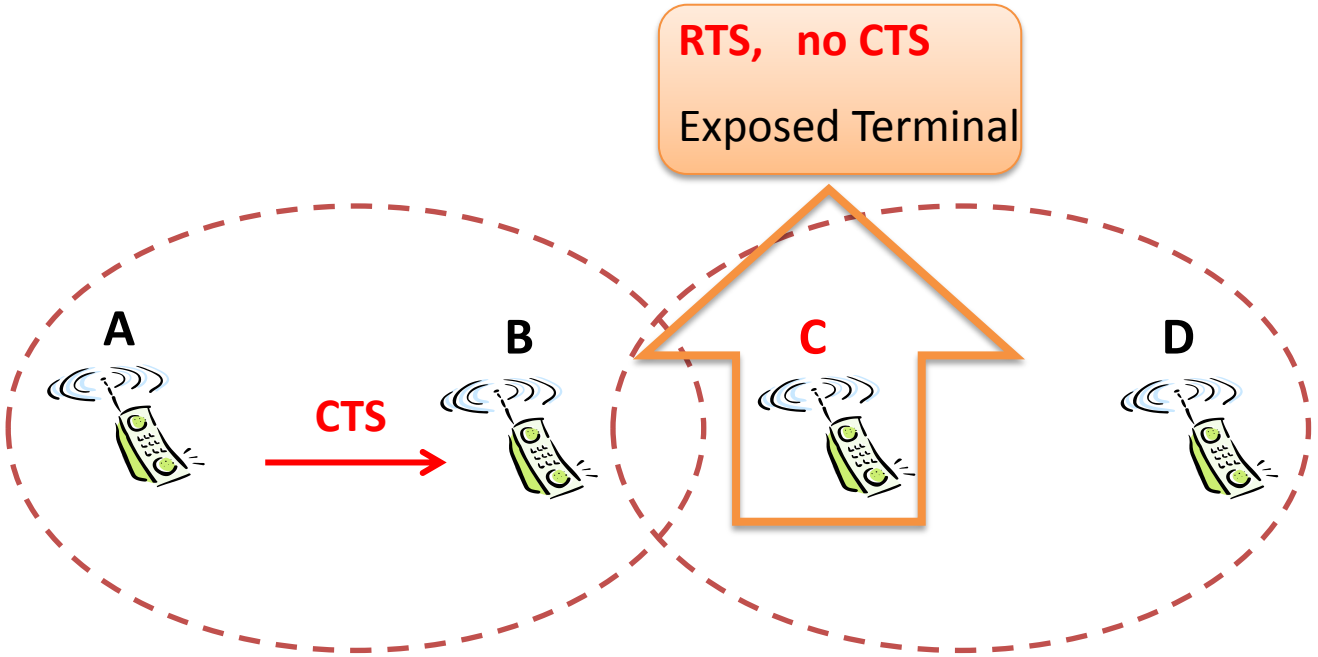
Carrier Sense Multiple Access with Collision Avoidance



State-of-art Solution

CSMA/CA

Carrier Sense Multiple Access with Collision Avoidance



Key Problem

CSMA/CA

Carrier Sense Multiple Access with Collision Avoidance

Costly Channel Usage Information (CUI)→

Reduce the effective throughput of data traffic!

Key Problem

CSMA/CA

Carrier Sense Multiple Access with Collision Avoidance

Costly ***(CUI)→***

Reduce ***traffic!***

RTS/CTS is disabled by default in 802.11

Key Problem

CSMA/CA

Carrier Sense Multiple Access with Collision Avoidance

***Cost-effective Channel Usage Information*→**

Avoid collision & Fully-utilize **channel capacity**

Key Problem

CSMA/CA

Carrier Sense Multiple Access with Collision Avoidance

How?

***Cost-effective Channel Usage Information*→**

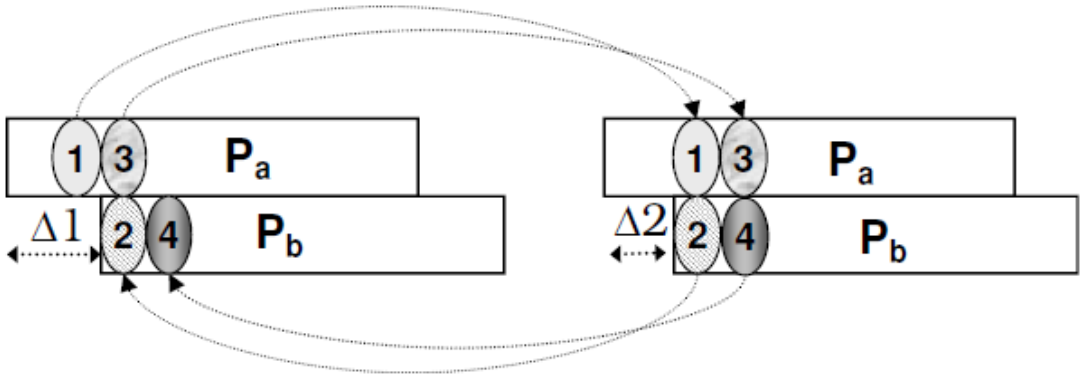
***Avoid collision & Fully-utilize* channel capacity**

Roadmap

- ~~Introduction~~
- **Motivation**
- FAST Design
- Performance Evaluation
- Conclusion

Observation 1/2

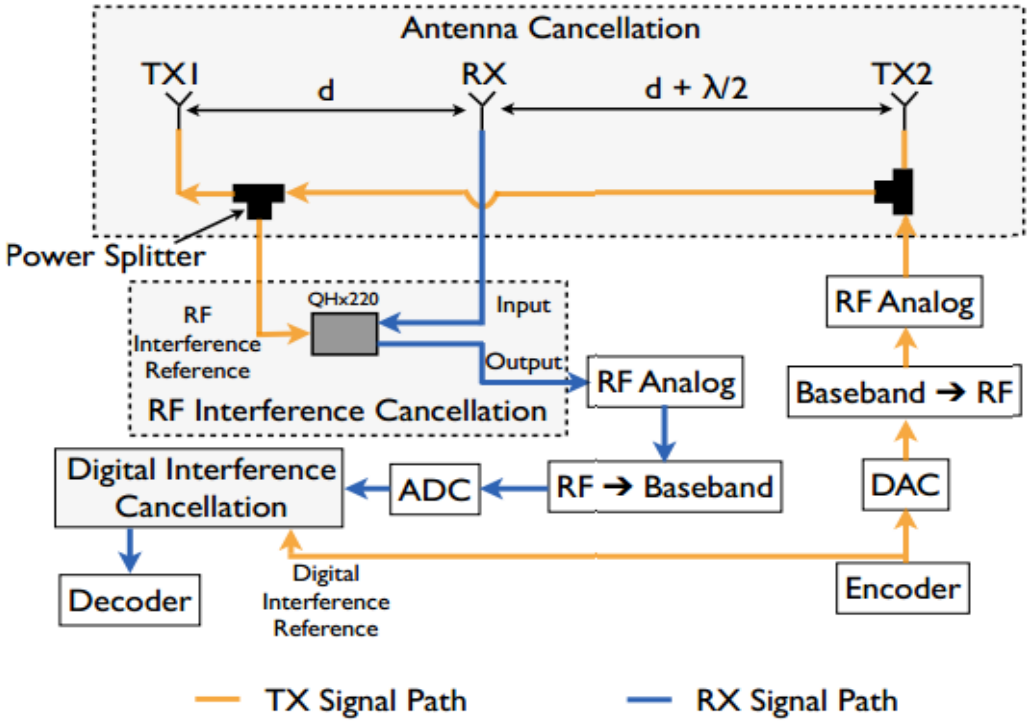
- Interference cancellation
  - ZigZag Decoding [*SIGCOMM'08*]
  - Using *interference-free* chunk of packet to *decode the collided chunk* in an iterative way



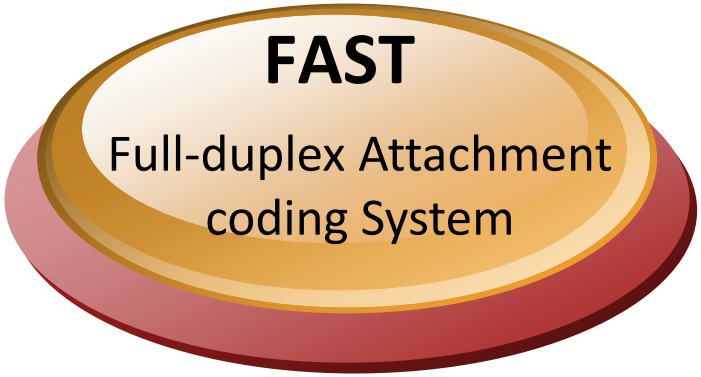
$\Delta 1 \neq \Delta 2$

Observation 2/2

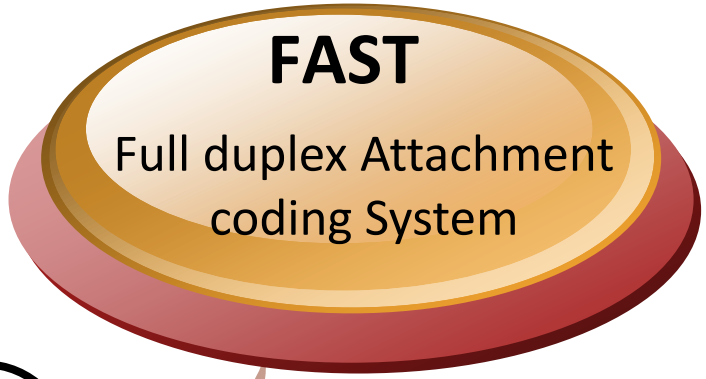
- Full–duplex wireless transceiver
  - Practical Real-Time Full Duplex Wireless [MOBICOM'11]
  - Utilizing *self-cancellation* to cancel out the *TX signal* at *RX side*



Motivation



Motivation

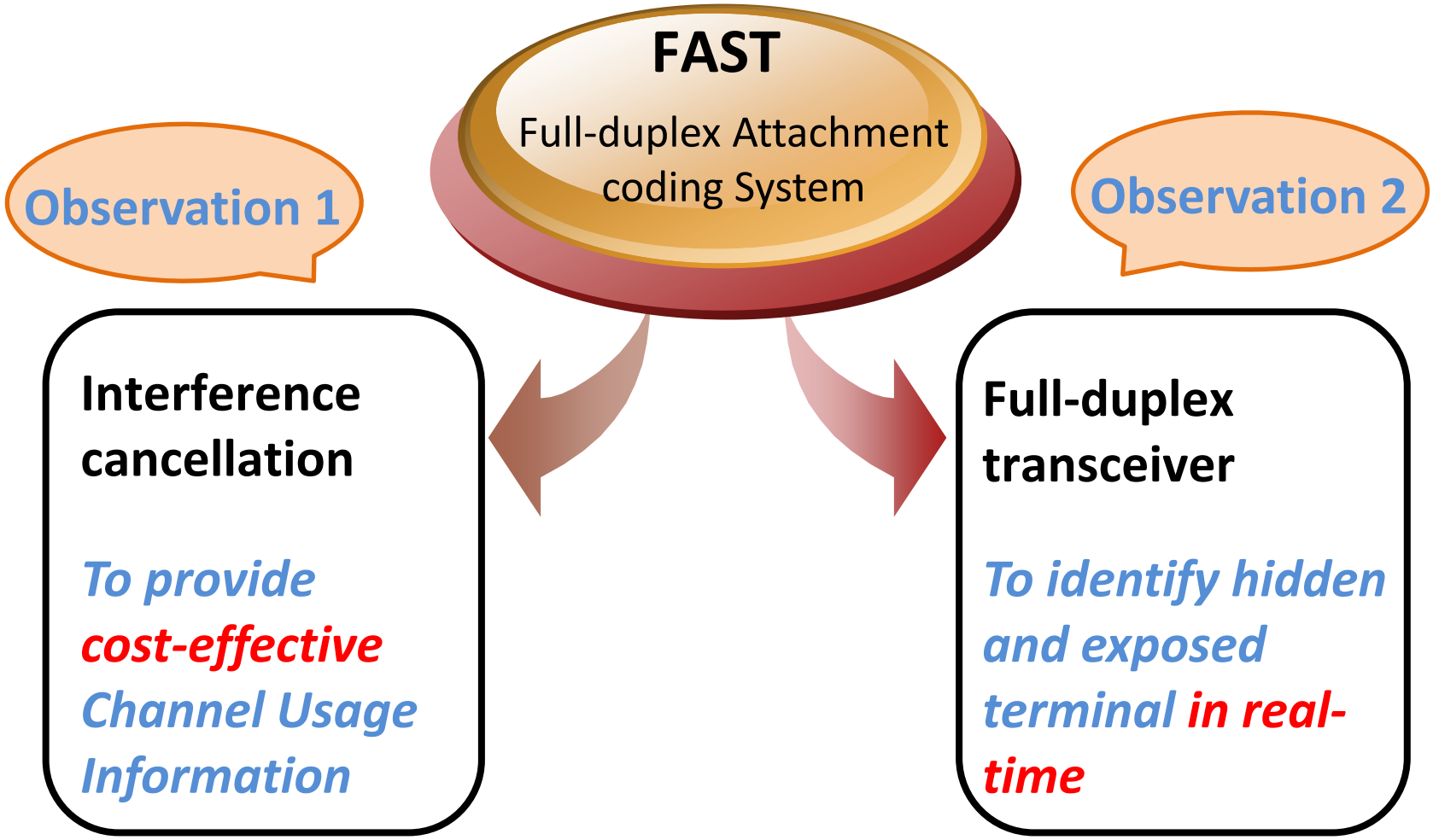


Observation 1

Interference cancellation

To provide cost-effective Channel Usage Information

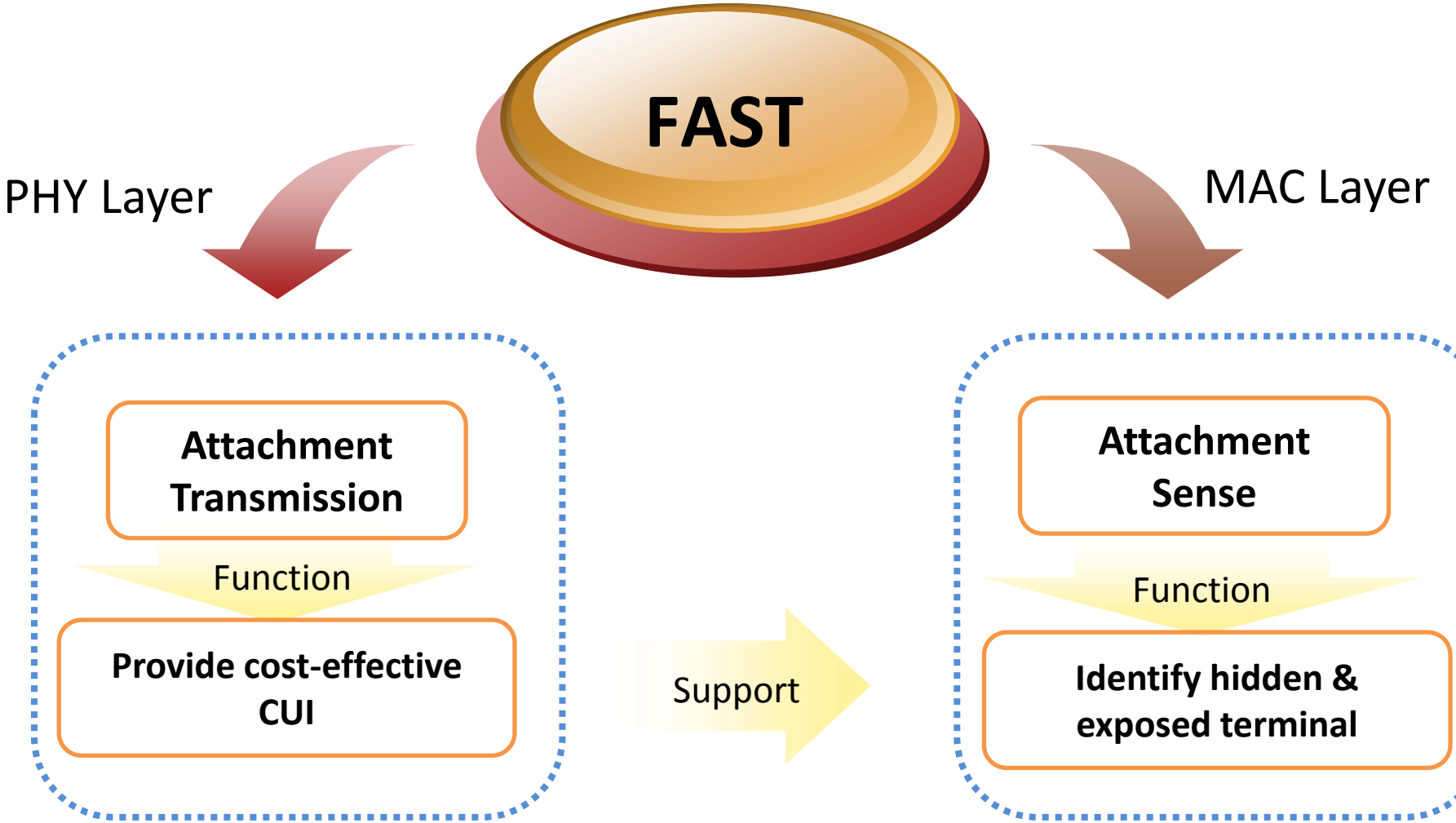
Motivation


FAST
Full-duplex Attachment coding System
Observation 1
Observation 2
Interference cancellation
To provide cost-effective Channel Usage Information
Full-duplex transceiver
To identify hidden and exposed terminal in real-time

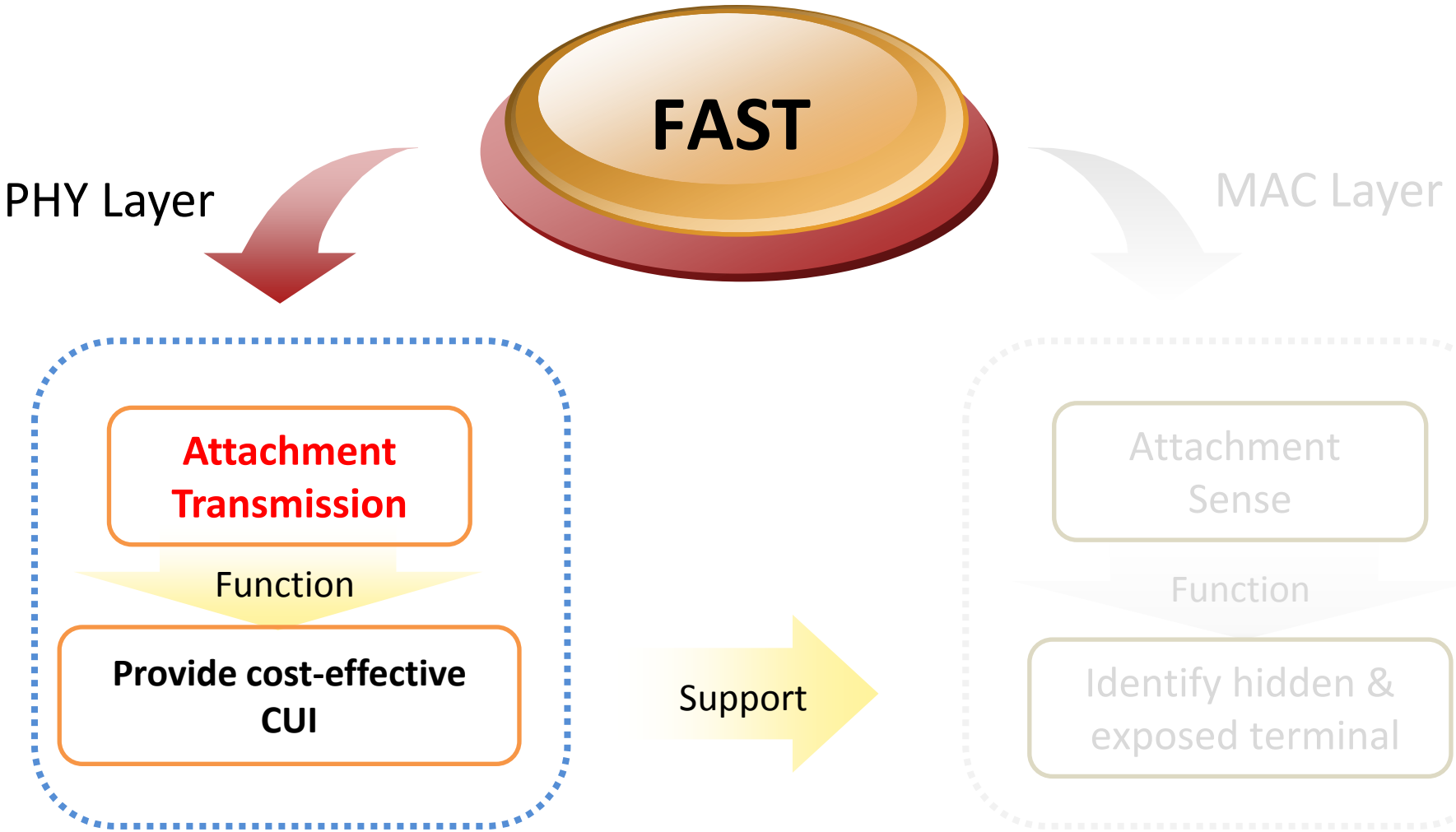
Roadmap

- ~~Introduction~~
- ~~Motivation~~
- **FAST Design**
- Performance Evaluation
- Conclusion

FAST Architecture



FAST Architecture

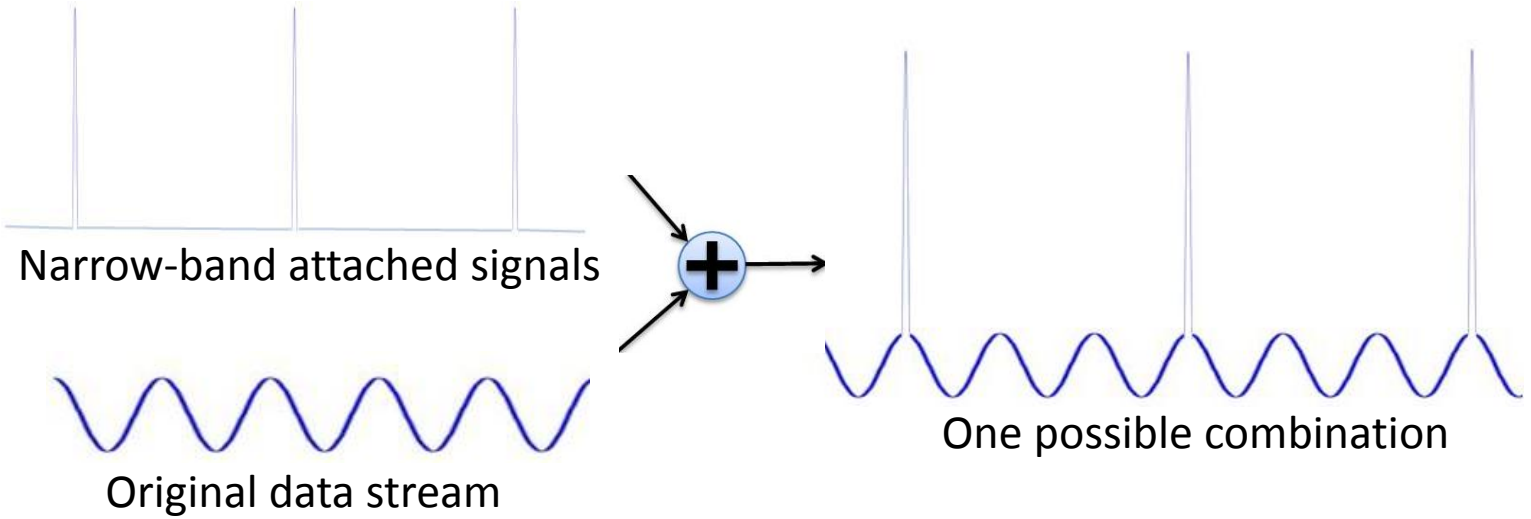


Roadmap

- ~~Introduction~~
- ~~Motivation~~
- **FAST Design**
  - Attachment Transmission
  - Attachment Sense
- Performance Evaluation
- Conclusion

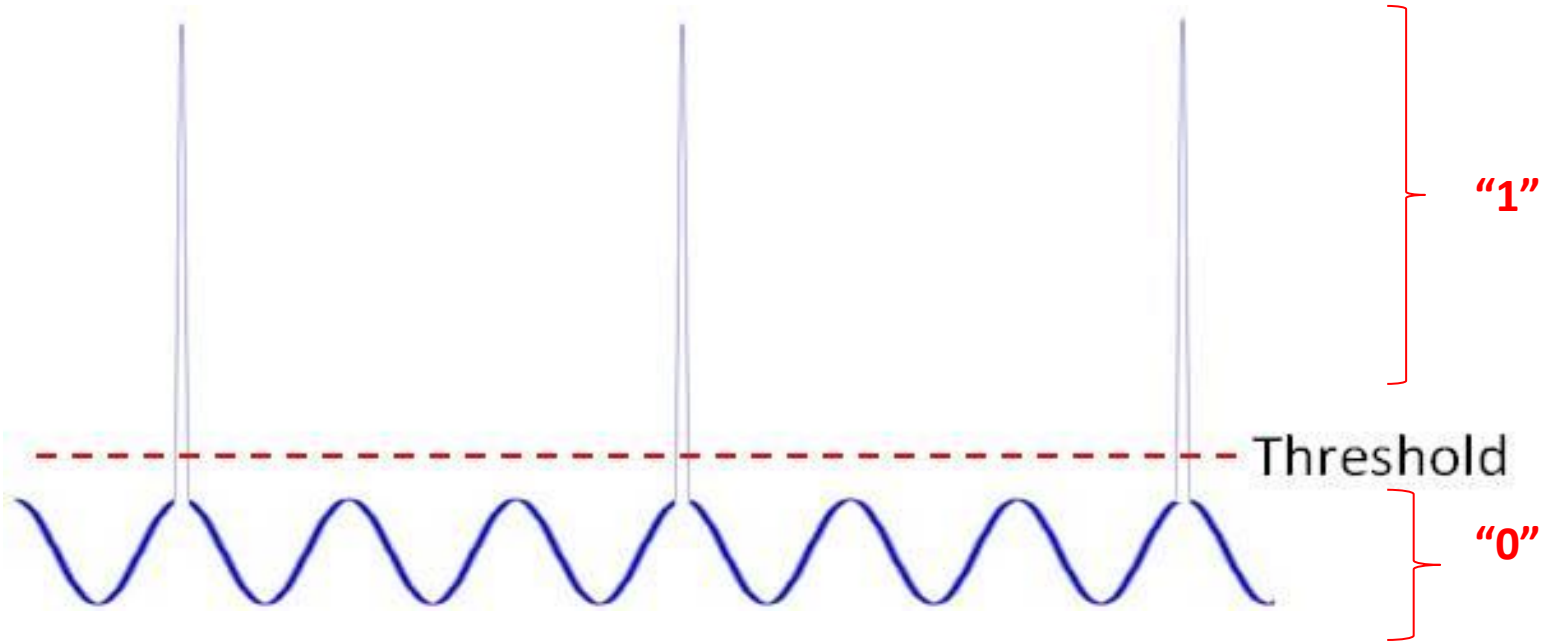
Attachment Transmission

- Narrower the channel width of ***attached signal.***
- Transmit the attached signal on ***one particular subcarrier*** with **higher energy.**



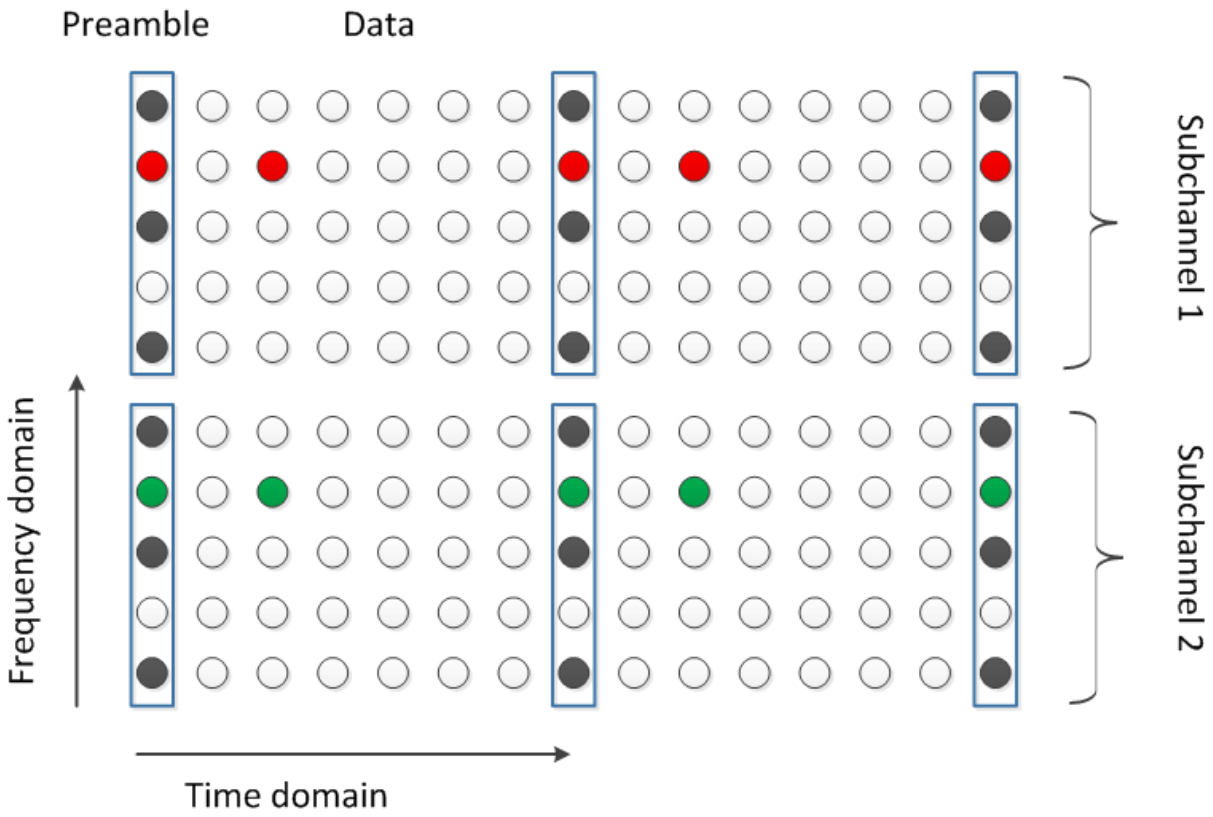
Attachment Detection

- ***Detect*** and ***record*** the energy which exceeds a certain threshold



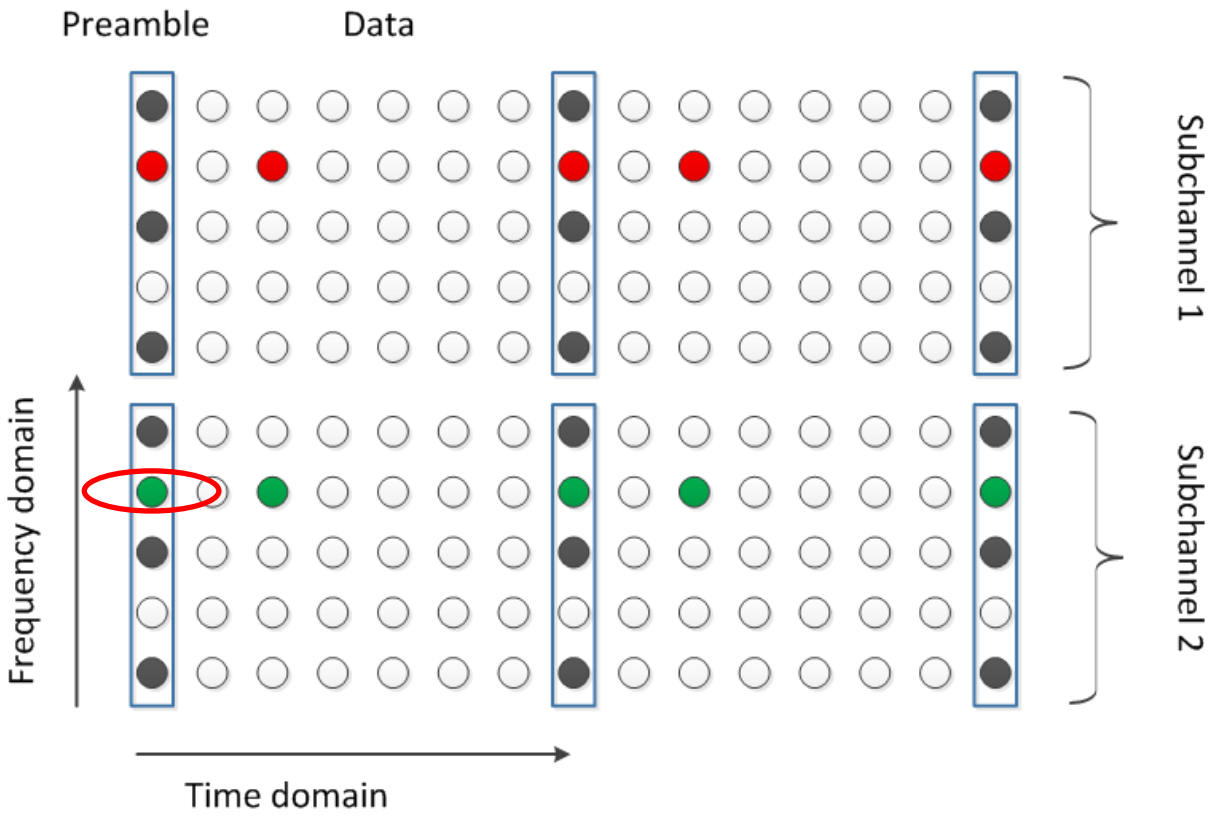
Attachment Cancellation

- ***Record*** the attached signal on ‘’clean’’ preamble
- ***Cancel out*** the attached signal in subsequent data



Attachment Cancellation

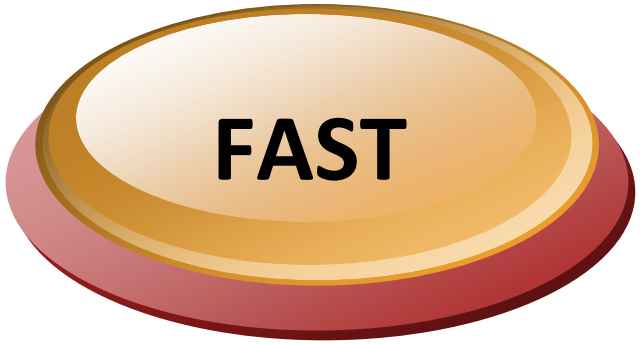
- ***Record*** the attached signal on ‘’clean’’ preamble
- ***Cancel out*** the attached signal in subsequent data



Attachment Cancellation

- ***Record*** the attached signal on ‘’clean’’ preamble
- ***Cancel out*** the attached signal in subsequent data

FAST Architecture



PHY Layer

MAC Layer

Attachment transmission

Function

Provide accurate CUI

Support

Attachment Sense

Function

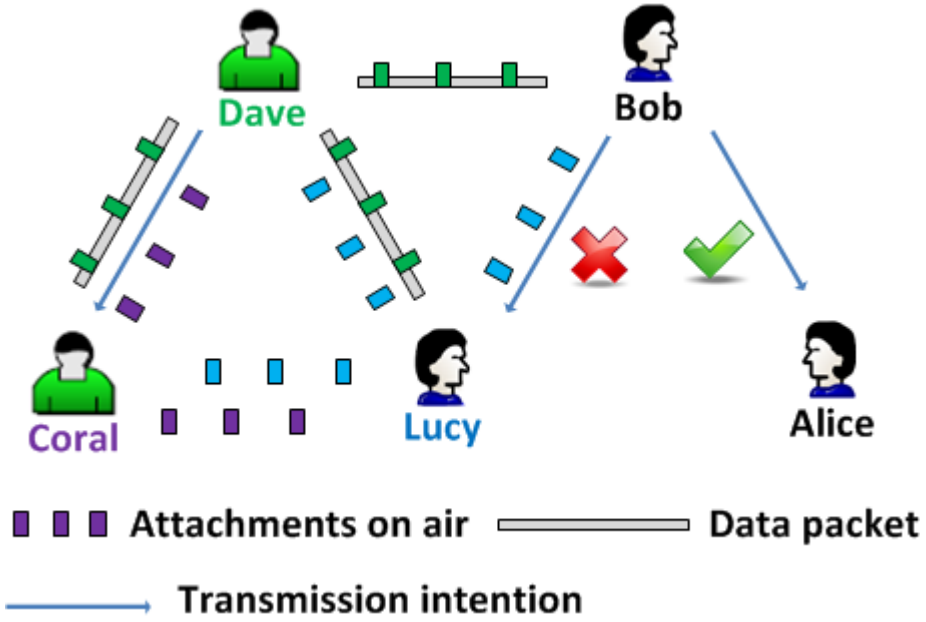
Identify hidden & exposed terminal

Roadmap

- ~~Introduction~~
- ~~Motivation~~
- **AT-Learning Design**
  - ~~Attachment Transmission~~
  - Attachment Sense
- Performance Evaluation
- Conclusion

MAC Layer design

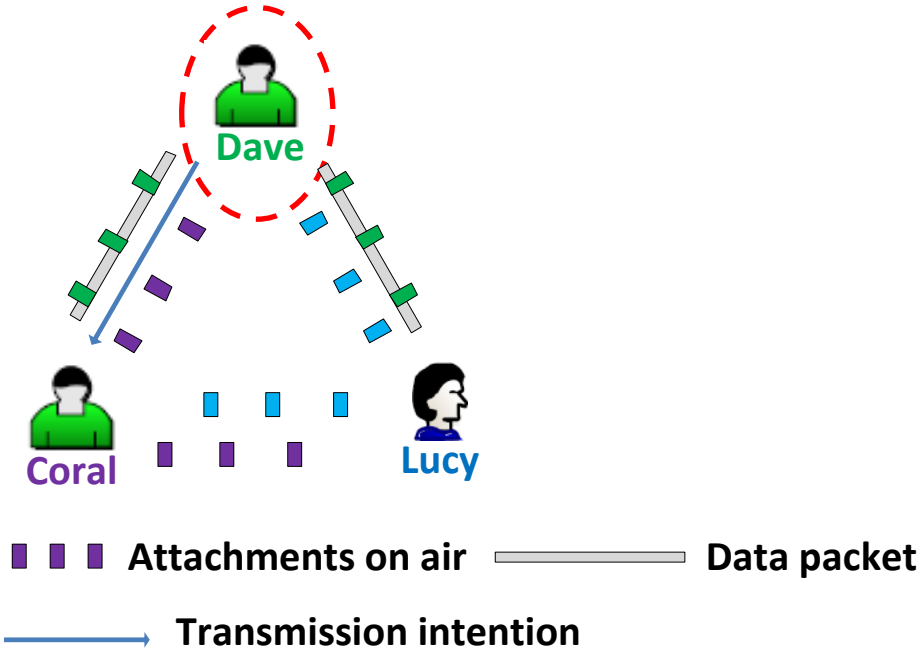
- *Sender*, *receiver* and *victim encode* *Channel Usage Information* into *Attachment*



To ***identify*** hidden & exposed terminal through *Attachment*

MAC Layer design

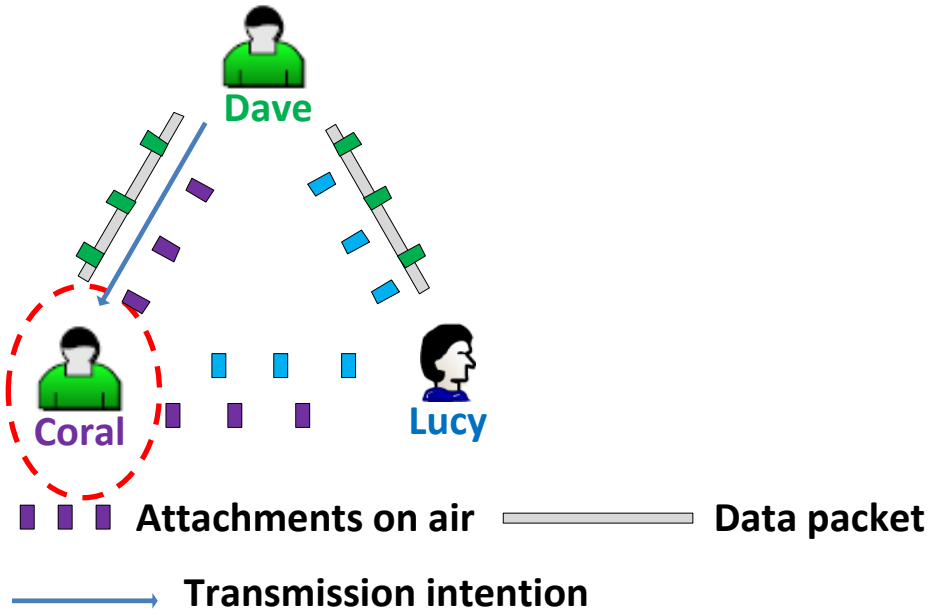
- *Sender*, *receiver* and *victim encode Channel Usage Information* into *Attachment*



To ***identify*** hidden & exposed terminal through *Attachment*

MAC Layer design

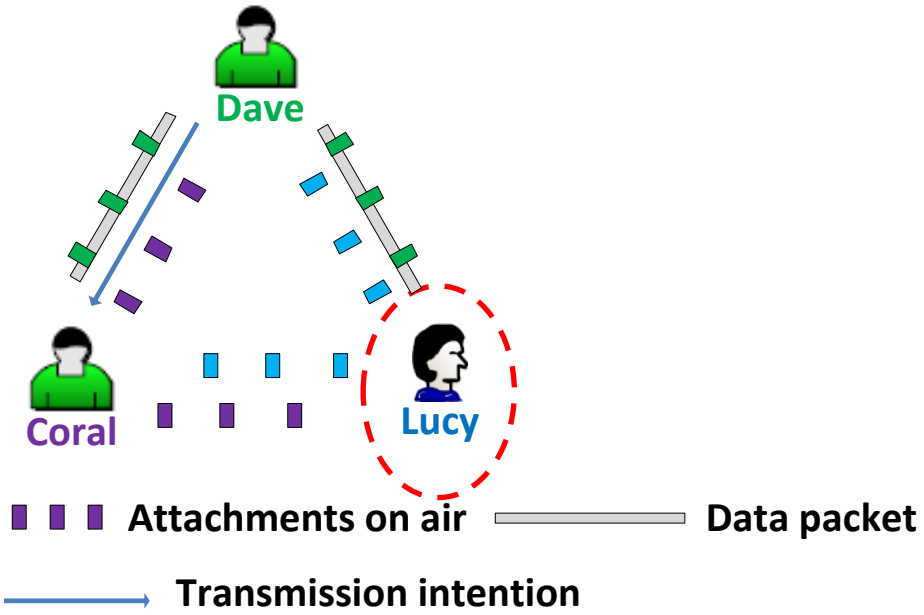
- *Sender*, *receiver* and *victim encode* *Channel Usage Information* into *Attachment*



To ***identify*** hidden & exposed terminal through *Attachment*

MAC Layer design

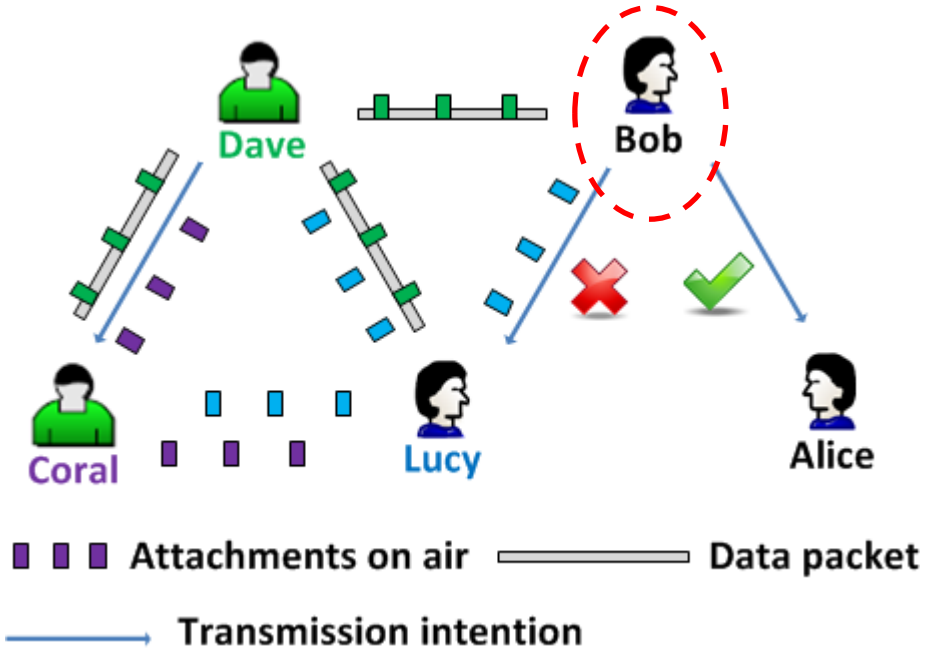
- *Sender*, *receiver* and *victim* encode *Channel Usage Information* into *Attachment*



To ***identify*** hidden & exposed terminal through *Attachment*

MAC Layer design

- Attachment Sense: *intended sender* listens to *Attachments* on air, obtains the current *Channel Usage Information*



To ***identify*** hidden & exposed terminal through *Attachment*

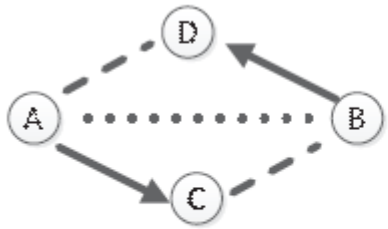
Roadmap

- ~~Introduction~~
- ~~Motivation~~
- ~~FAST Design~~
- Performance Evaluation
- Conclusion

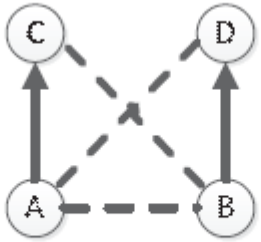
Simulation

- NS3 simulator

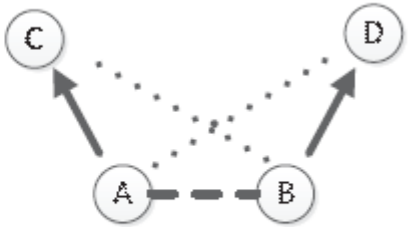
Baseline topology



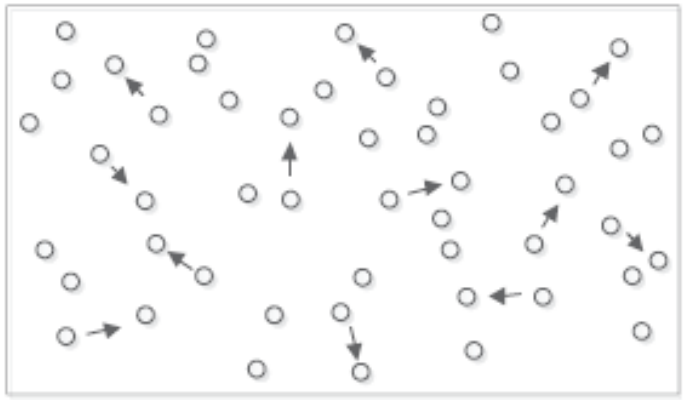
(a) Hidden node



(b) Interfering node



(c) Exposed node



(d) Ad-hoc network

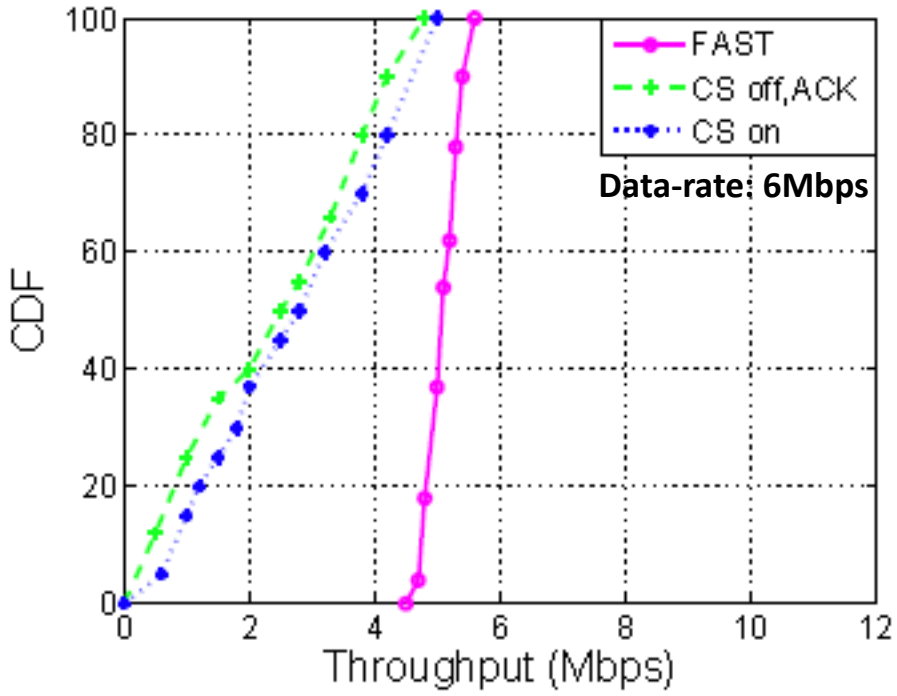
Transmission link:
PRR > 0.9, Signal > 90%-ile

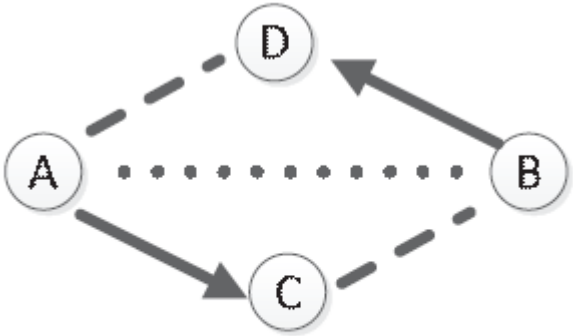
In range: PRR > 0.2

Out of range: PRR < 0.2

Practical network

Baseline Topology 1/3



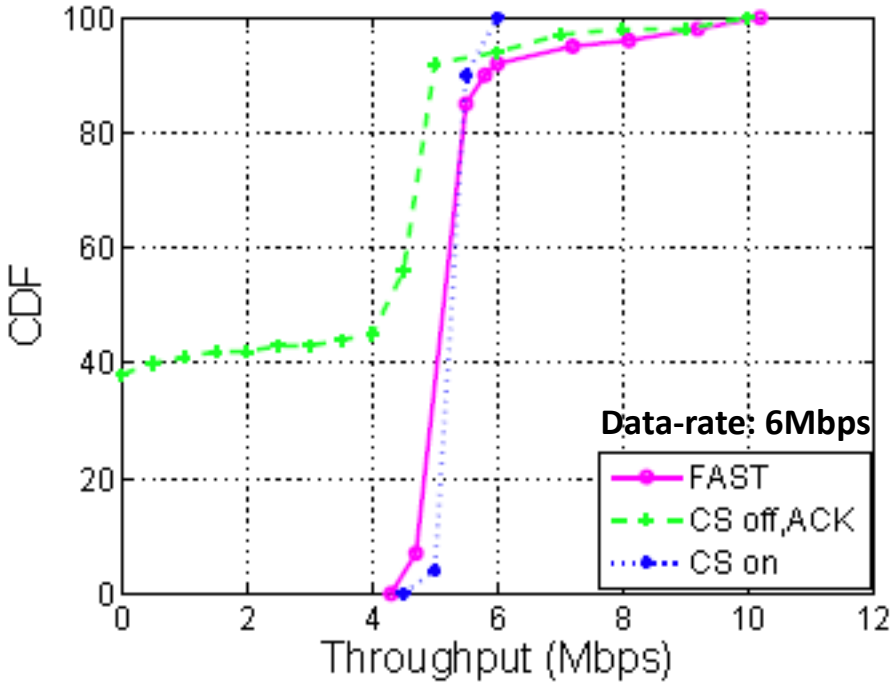


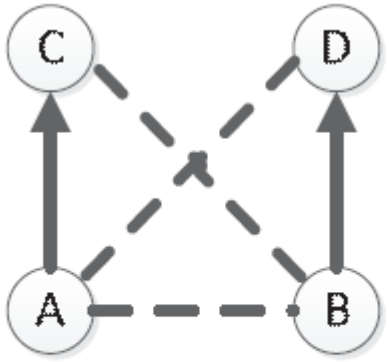
(a) Hidden node

Successfully identify receivers nearby

Avoid collisions due to hidden terminal

Baseline Topology 2/3

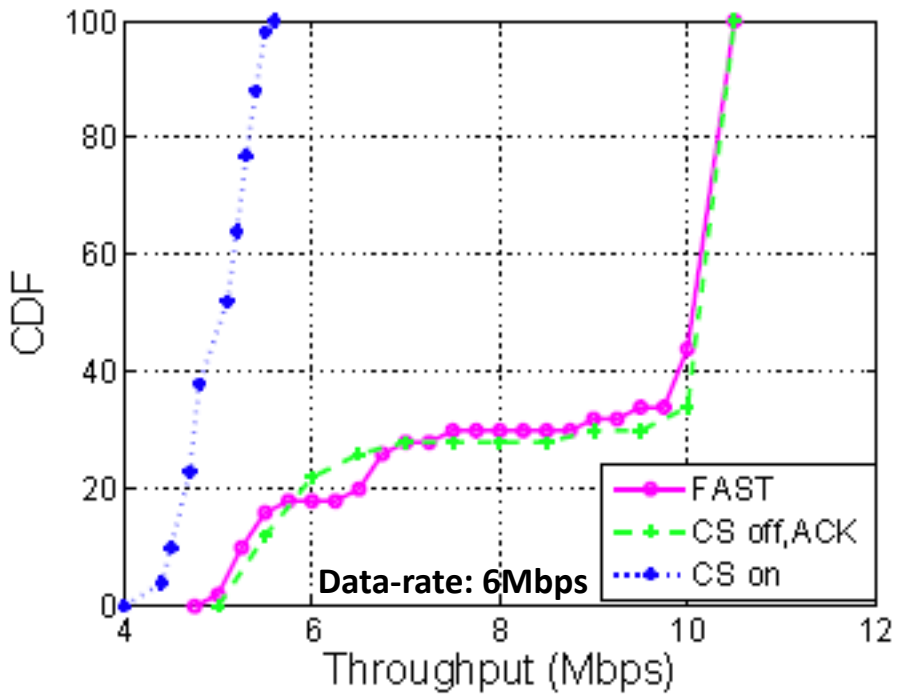


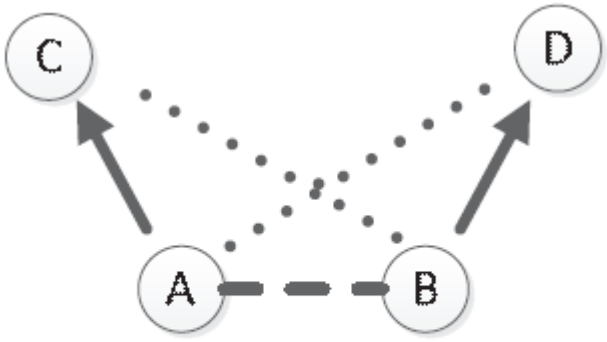


(b) Interfering node

Figure out interfering transmissions
Guide senders take turns to transmit

Baseline Topology 3/3

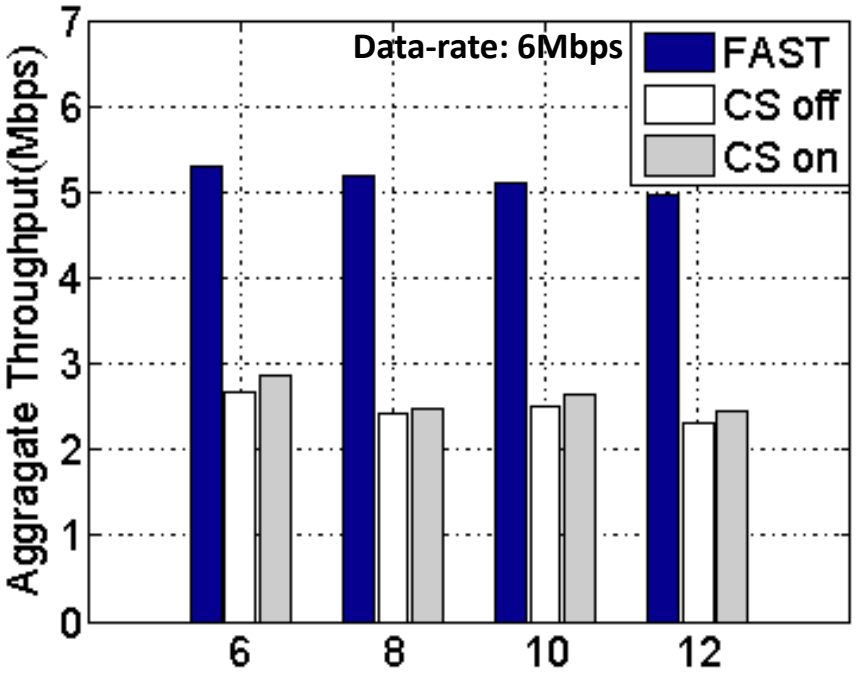




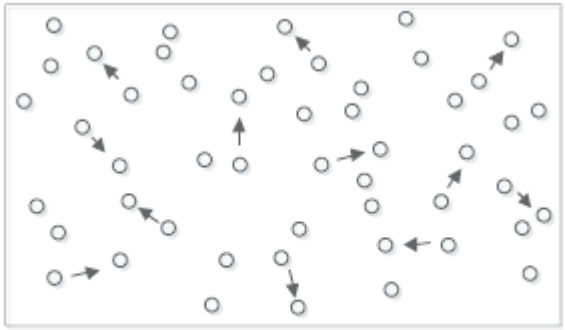
(c) Exposed node

Successfully identify exposed nodes
Fully utilize exposed nodes for transmission

Practical Networks



Per-sender throughput in ad-hoc networks



(d) Ad-hoc network

FAST can achieve up to 200% throughput gain over CSMA

Roadmap

- ~~Introduction~~
- ~~Motivation~~
- ~~FAST Design~~
- ~~Performance Evaluation~~
- **Conclusion**

Conclusion

- In wireless networks, *hidden and exposed terminal problem* *are two key problems.*
- We propose a cross-layer *FAST* to solve hidden and exposed terminal problems.
  - *Attachment Transmission* to *cost-effectively* transmit Channel Usage Information.
  - *Attachment Sense* to *identify hidden and exposed terminal*, and guide node to make the right access decision *fast and accurate*.

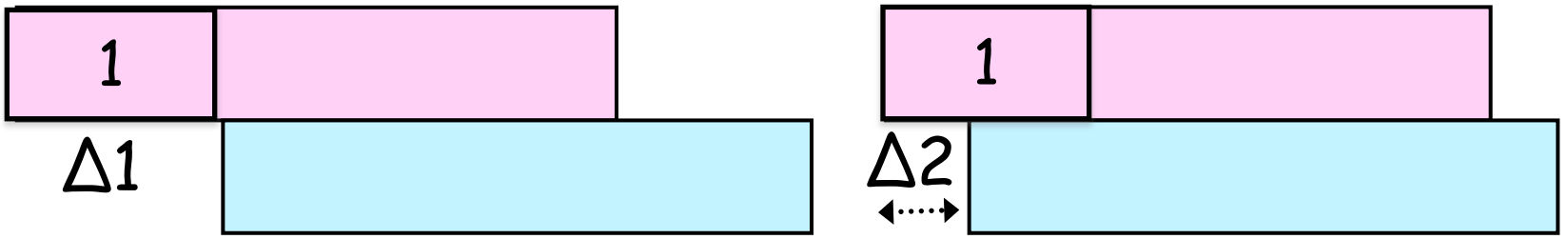
Thank You !

Q&A

HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY
wanglu@cse.ust.hk

Backup

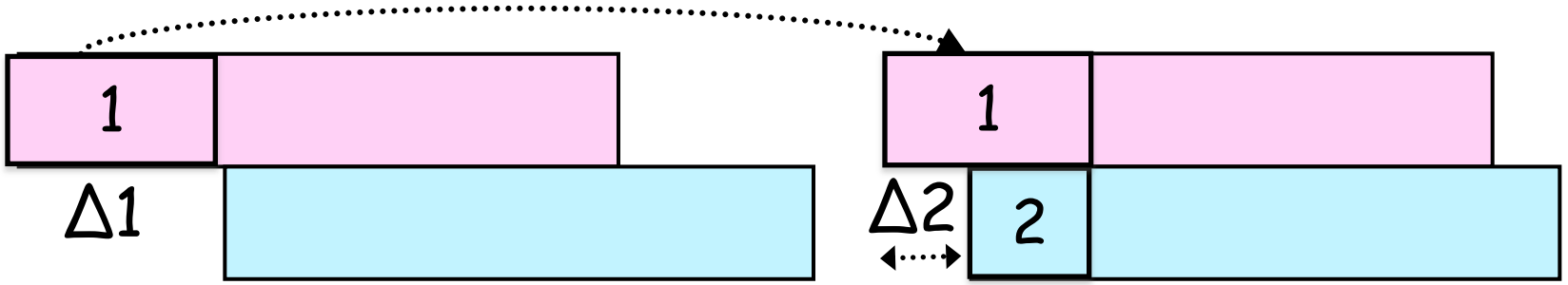
- ZigZag



$\Delta 1 \neq \Delta 2$

Backup

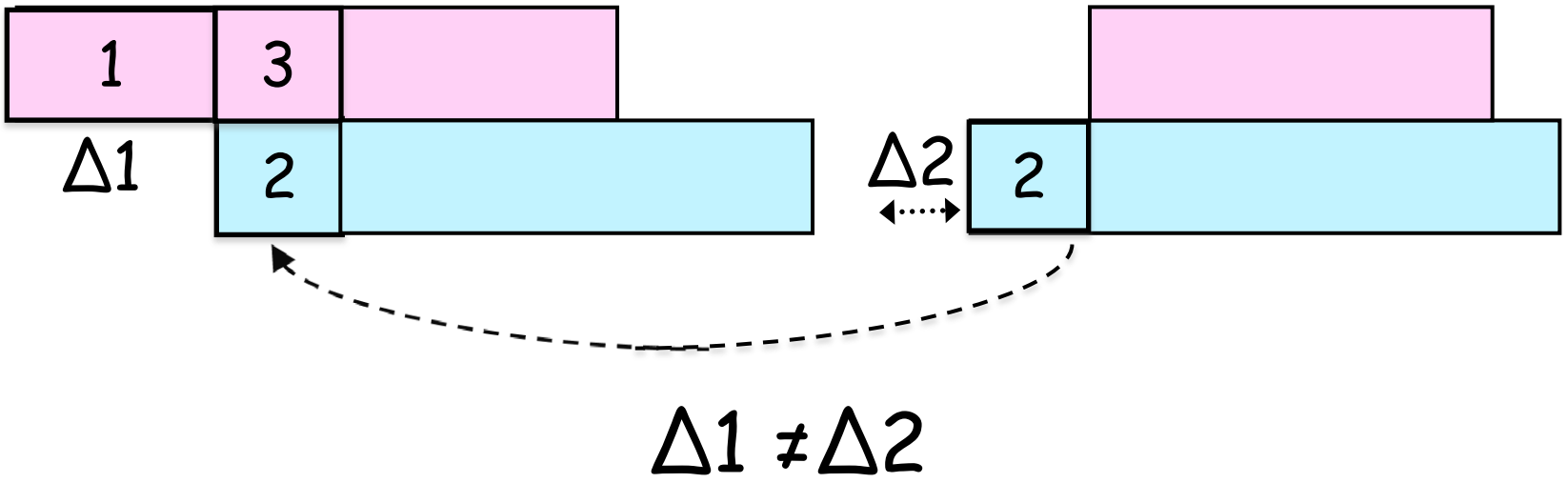
- ZigZag



$\Delta 1 \neq \Delta 2$

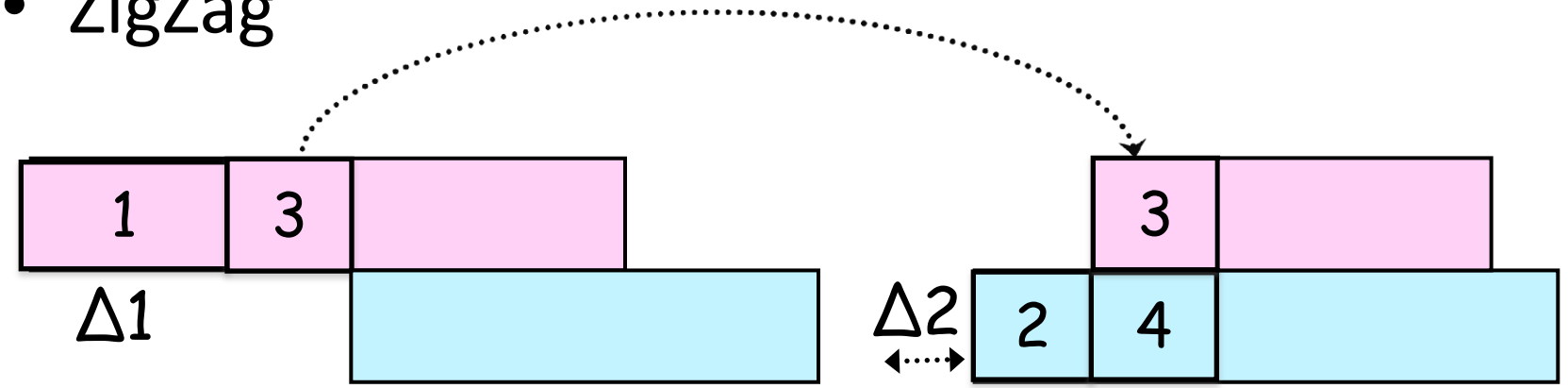
Backup

- ZigZag



Backup

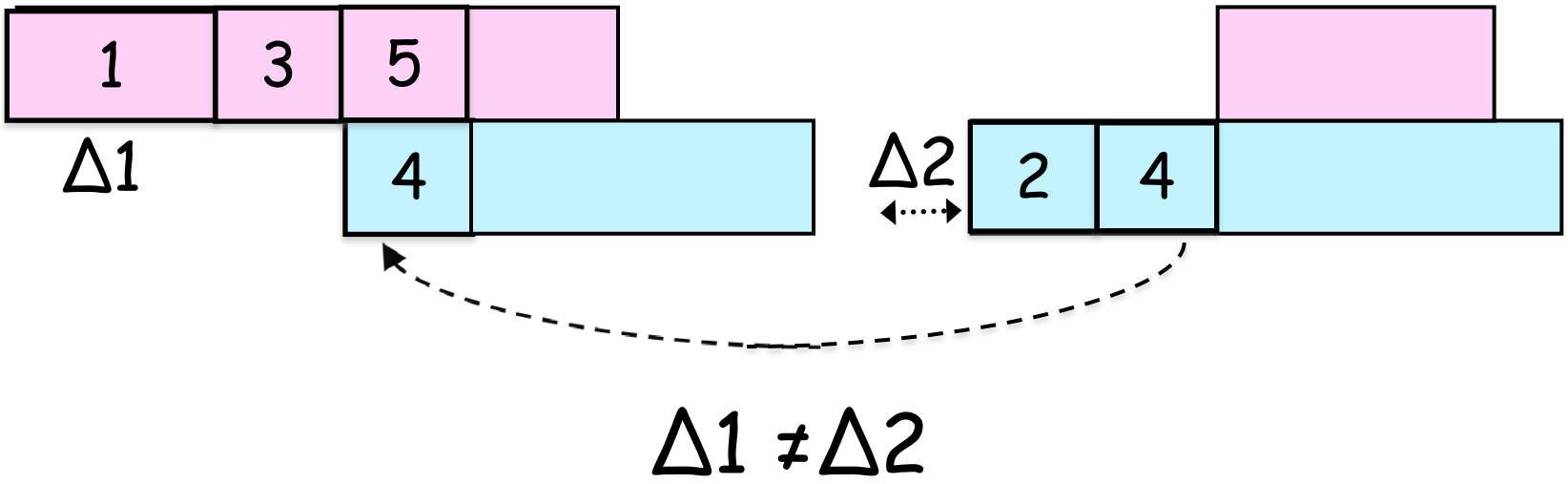
- ZigZag



$\Delta 1 \neq \Delta 2$

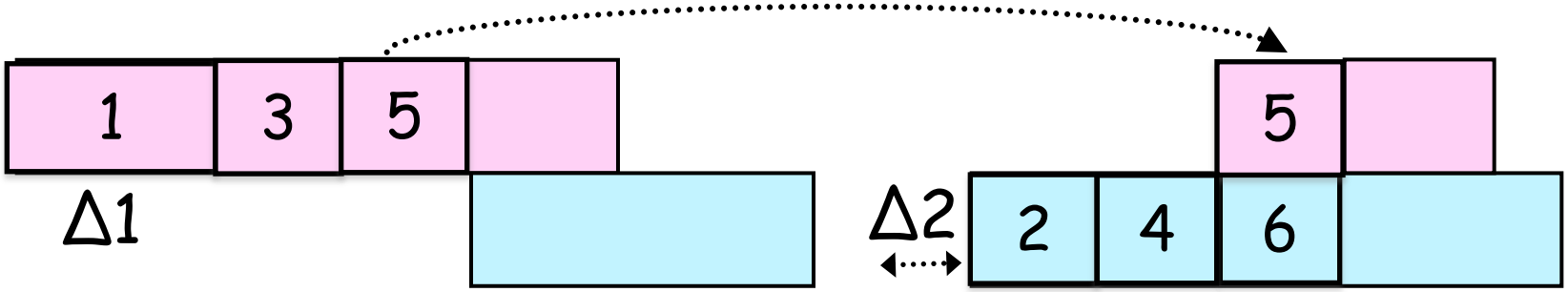
Backup

- ZigZag



Backup

- ZigZag



$\Delta 1 \neq \Delta 2$

Backup

- ZigZag