# Medium Access Control (MAC)

Elec 590: Networking fundemental

Presenter: Emad Shihab

# MAC

- Why MAC?
  - Wireless medium is open and shared
  - Any node can broadcast
  - Multiple nodes may access the wireless medium at the same time
  - This is why we MAC protocols
- MAC protocols?
  - Defined rules to force distributed nodes to access wireless medium in an orderly and efficient manner

# Goals of MAC

- To design an effective MAC, we first need to know what are the goals of MAC
  1. Fairness
  2. High medium utilization (can refer to throughput)
  3. Stability
  4. Limited delay
  5. Scalability
  6. Low power consumption (sometimes dependent on application)

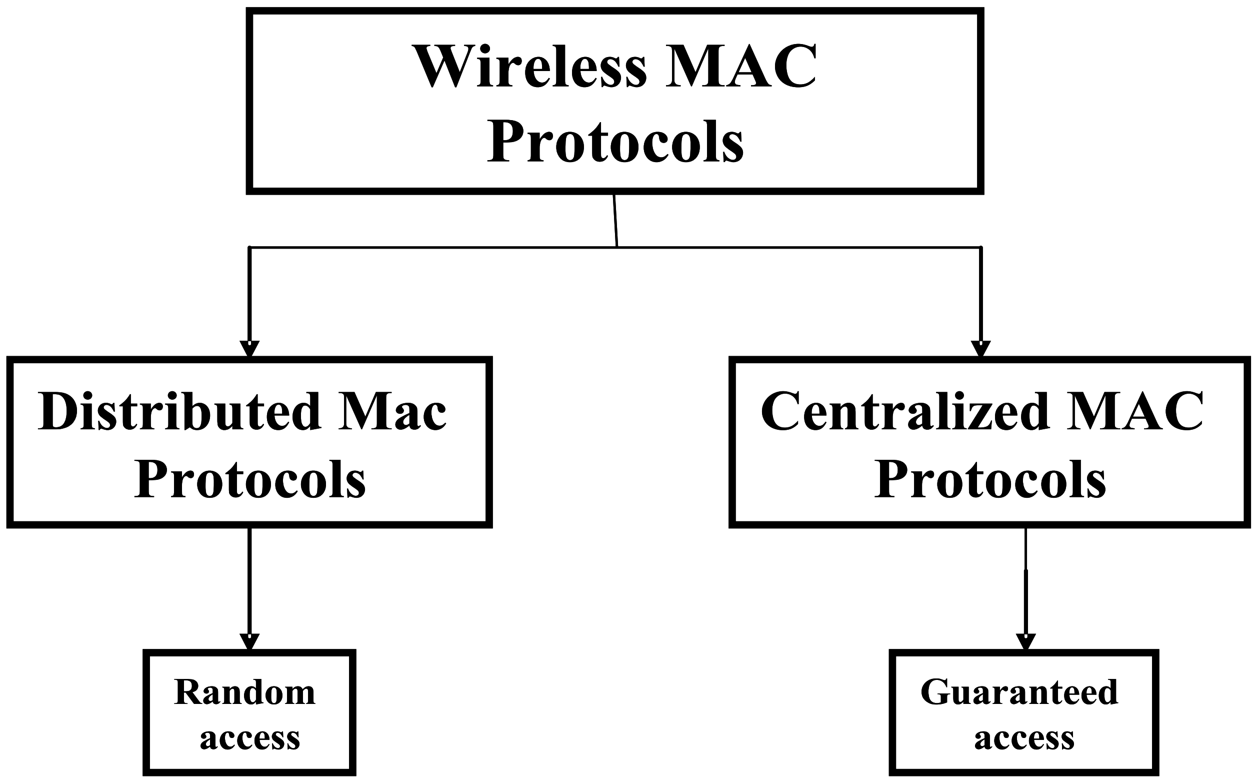
# Wireless MAC issues

- Half duplex operation
- Time varying wireless channel
- Location dependent carrier sensing
  - Hidden terminal problem
  - Exposed terminal problem
- A “collision” occurs when the receiver is in reception range of 2 senders and is unable to receive from either one

# Wireless MAC issues cont'd...

- A “capture” occurs when 2 nodes send to 1 receiver and only one of the signals is decoded because its signal strength is much higher
  - A and C send to B, simultaneously. B can only decode signal received from A, because it is a stronger signal. This results in unfair sharing of BW!
- “Interference” occurs when a node is in one sender's range and slightly out of another node's range.
  - This node is unable to receive a signal clearly because of the *interference* caused by the node slightly out of range

# Classification of wireless MAC protocols



# Distributed MAC protocols

- Nodes communicate directly
- Uses collision avoidance and carrier sensing mechanisms (can cause problems)
  - Collision avoidance with in-band control signals
    - Send control messages before transmitting
  - Collision avoidance with out-of-band signaling
    - Busy tone

# Centeralized MAC protocols

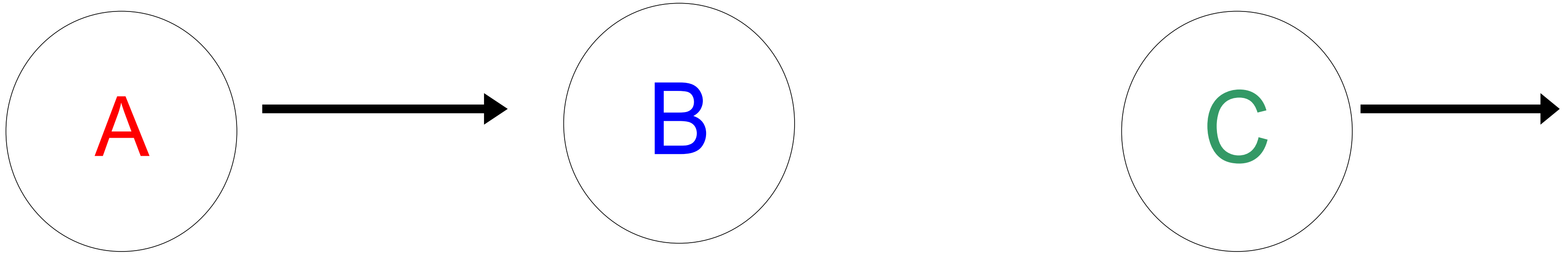
- Base station controls access to medium
- All nodes hear and talk to base station
- Complexity at the base station
- Note: no hidden and exposed terminal problem here!

# CSMA

- Station senses the carrier before transmitting
- HOWEVER collisions occur at the *receiver* not sender
- CSMA provides information about potential collisions at the sender, not the receiver

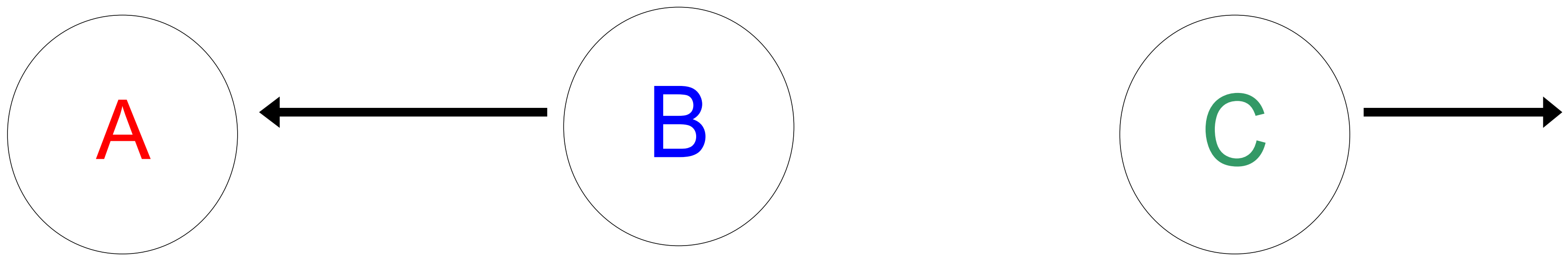
# Hidden terminal problem

- A sends to B
- B is *receiving* from A
- If C transmits
- Collision occurs at B!



# Exposed terminal problem

- B is sending to A
- C senses B's transmission and is not able to send
- C is "exposed to B"
- Loss in throughput



# Multiple Access with Collision Avoidance (MACA)

- Goal: overcome the exposed and hidden terminal problems (does really??)
- Idea:
  - Reserve the channel before sending/receiving packets
  - Control packets are much smaller than data packets (minimize cost of collisions)
- Contributions: A three-way handshake

# Assumptions

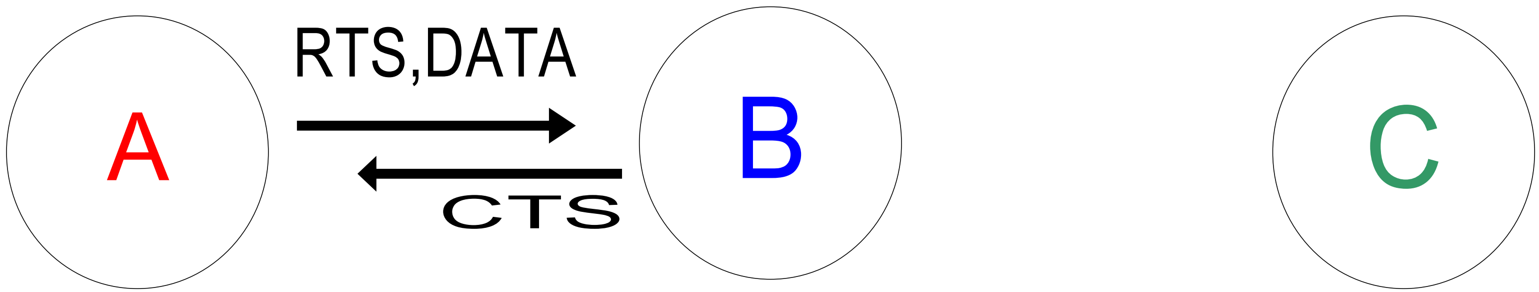
- For MACA and MACAW these assumptions hold:
  1. Symmetry

     If A can hear from B then B can hear from A
  2. Effect of capture is ignored
  3. Effect of channel fading is ignored
  4. Packet errors only due to collision
  5. In-band signaling

     Data and control packets are transmitted over the same channel

# MACA

- MACA uses 2 fixed size signaling packets:
  - Request-to-Send (RTS)
  - Clear-to-Send (CTS)
- In MACA, if A wants to send to B then...
  - A sends RTS
  - B sends CTS
  - A sends DATA



# MACA cont'd...

- Any station overhearing a RTS defers all transmission(Tx) until after the CTS would have been received.
- Any station overhearing a CTS would defer Tx until after the DATA Tx is complete
- Note: Any station that hears a RTS but not CTS is able to Tx after the CTS has been received
- This RTS-CTS mechanism is there to address the hidden terminal and exposed terminal problem
- Note: no carrier sense!

# Backoff in MACA

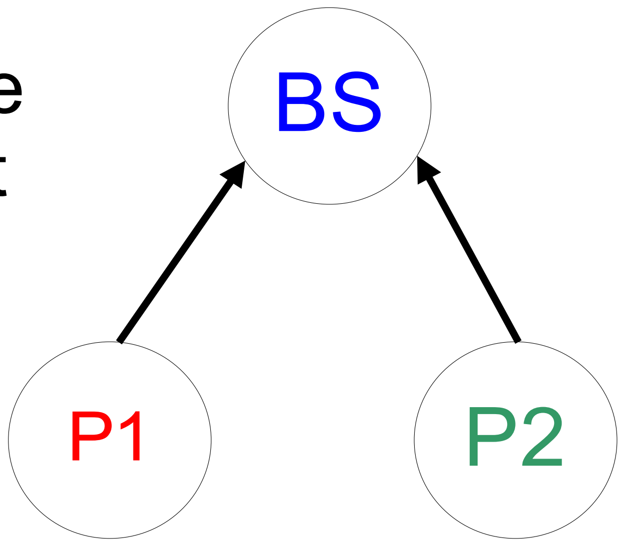
- MACA uses a binary backoff algorithm (BEB)
- If CTS is received after RTS, then $BO=BO_{min}$
- If CTS not received after RTS, then $BO=min[2BO_{old}, BO_{max}]$
- [BHARGHAVAN94] uses $BO_{min}=2$ and $BO_{max}=64$ for MACA

# MACAW

- Goals: is an extension and improvement based on MACA
- Idea: share information to achieve fairness
- Main contributions:
  - Four-way handshake (RTS-CTS-DATA-ACK)
  - Five-way handshake (RTS-CTS-DS-DATA-ACK)
  - Request-for-RTS (RRTS)
  - Modified the backoff algorithm (MILD)
  - Synchronize backoff counter using piggy back messages (copy)
  - Multiple stream model

# MACAW backoff algrothim

- BEB
  - Consider the figure below, if P1 gets the access to the channel and P2 backs off then P1 will continue to have $BO_1=BO_{min}$ and $BO_2$ will eventually reach $BO_2=BO_{max}$
- BEB copy
  - Includes a copy of the current value of the backoff counter in the packet header
  - Whenever a station hears a packet it copies the backoff counter value



# MACAW backoff algrothim cont'd..

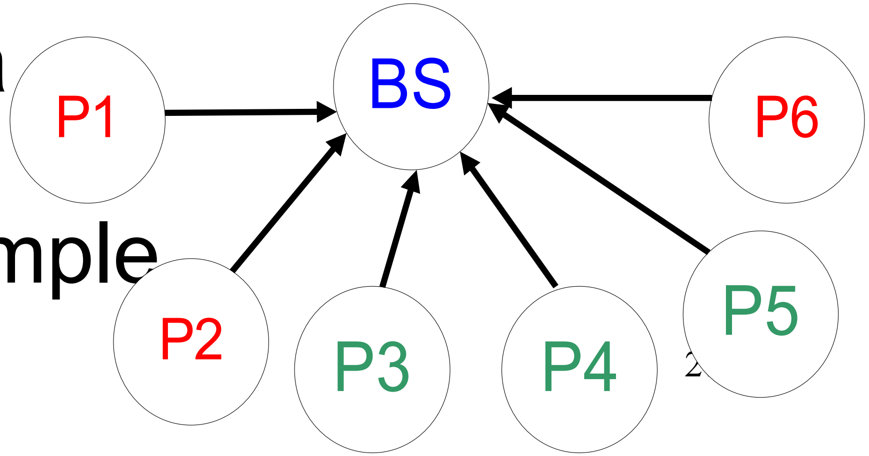
- BEB copy cont'd...
  - This is good when you have multiple pads competing for access (i.e the figure)
  - This is bad in a multi-cell environment, because congestion in one cell can cause other cells to operate at the level of the congested cell (example later on in the presentation)

# MACAW backoff algrothim cont'd..

- Multiplicative Increase, Linear Decrease (MILD)
  - To prevent wild oscillations, MACAW proposes to:
    - **If collision occurs BO=min[1.5$BO_{old}$, $Bo_{max}$]**
    - **If successful Tx BO=max[$BO_{old}$-1, $Bo_{min}$]**

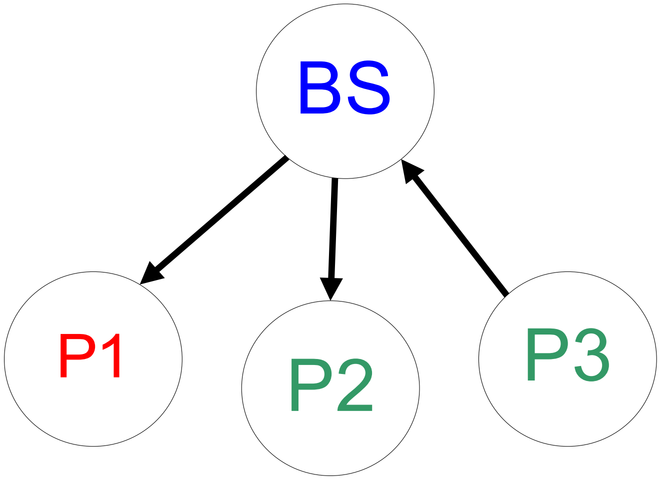
# A MILD example..

- Consider the topology below, here BEB copy only achieves half of the throughput that can be achieved by MILD copy.
- This is because in MILD copy, the backoff after a collision is linear and this optimistic number is copied to all pads. Therefore, the BO for each of the pads remains fairly large, but equal.
- This is ideal when there is a high level of contention, in one cell such as in this example



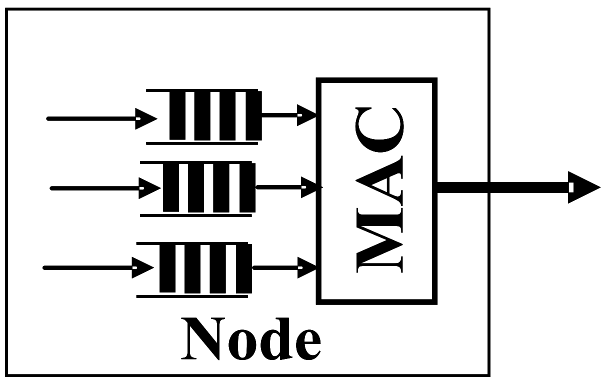
# MACAW: Multiple Stream Model

- We can see from the figure below, if we allocate BW based on station, then the P3-BS pair gets half of the BW and the P1-BS and P2-BS pairs share the other half. This is unfair!
- Want to allocate BW to the streams not stations
- A stream is a flow of data packets between a source-destination pair

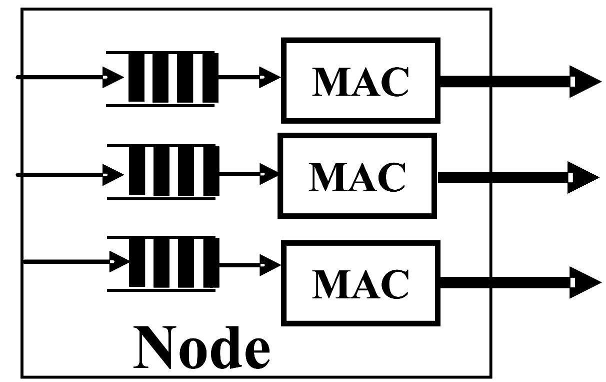


# MACAW: Multiple Stream Model cont'd..

- What does this mean?




**Single Stream MAC**



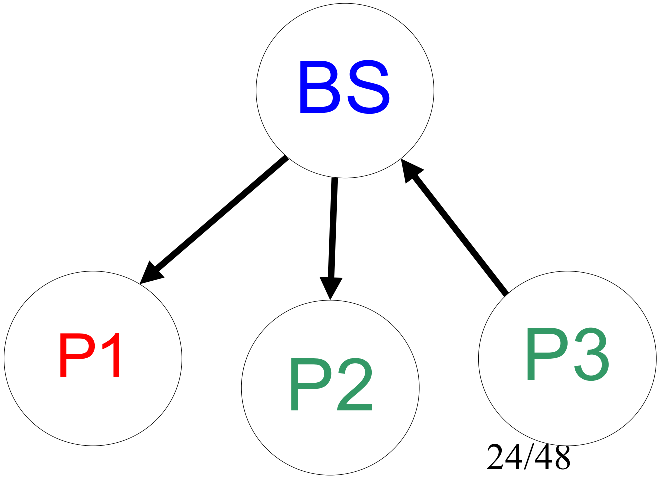

**Multiple Stream MAC**

- Keep a separate queue for each stream in each station, and run backoff algorithm for each queue independently

# MACAW: Multiple Stream Model cont'd..

- When a station is allowed to send and finds that it has N destinations, it flips a coin to determine, based on the backoff counter, how long to wait before sending RTS.
- Then picks the one with the shortest time.

| | Single Stream | Multiple Stream |
|---|---|---|
| B-P1 | 11.42 | 15.07 |
| B-P2 | 12.34 | 15.82 |
| P3-B | 22.74 | 15.64 |



# MACAW: ACK

- To better support reliable delivery of data, MACAW proposes to use ACK after receiving data packet
- This means that instead of RTS-CTS-DATA, we use RTS-CTS-DATA-ACK
- If the ACK is not sent, then sender retransmit
- If the ACK is not received, then sender sends ACK in response to RTS for retransmission
- If no ACK or CTS, then increase BO at sender

# MACAW: ACK cont'd...

- If RTS-CTS successful, but no ACK arrives then BO doesn't change
- The additional overhead of ACK is about 9% in the no noise case. BUT there is significant improvement when errors are introduced.

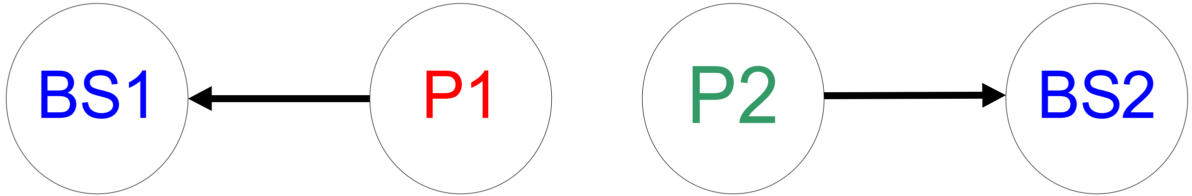
| Error Rate | RTS-CTS-DATA | RTS-CTS-DATA-ACK |
|---|---|---|
| 0 | 40.41 | 36.76 |
| 0.001 | 36.58 | 36.67 |
| 0.01 | 16.65 | 35.52 |
| 0.1 | 2.48 | 9.93 |

# MACAW: Data-Sending (DS)

- Take for example the figure below
- P1 sends RTS to BS1
- P2 does not know if the RTS-CTS between P1-B1 was successful
- P2 has to sense before sending RTS

or

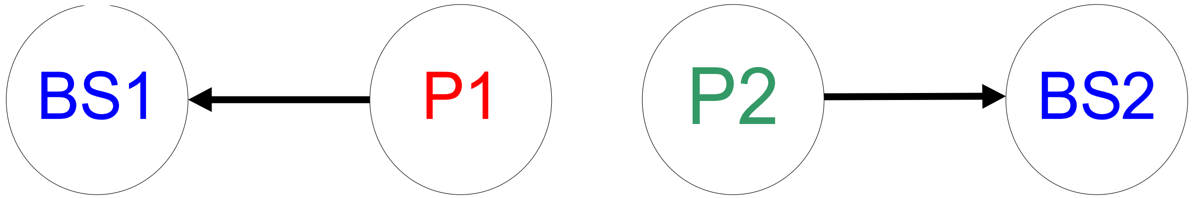
- P1 sends DS before DATA to let everyone know that RTS-CTS was successful
- P2 defers Tx until after ACK received



# MACAW: Data-Sending (DS) cont'd...

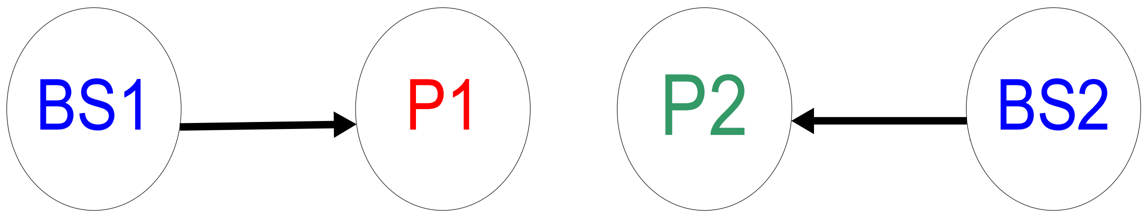
- What happens here is that (without DS) P1 wins the first contention period and continues to get access of the medium
- This is mainly because P2 is not synchronized and cannot contend for the medium fairly
- When using DS, P2 knows when P1 will finish, hence, it will know when the next contention period is and can contend fairly

| | RTS-CTS-DATA-ACK | RTS-CTS-DS-DATA-ACK |
|---|---|---|
| P1-B1 | 46.72 | 23.35 |
| P2-B2 | 0 | 22.63 |



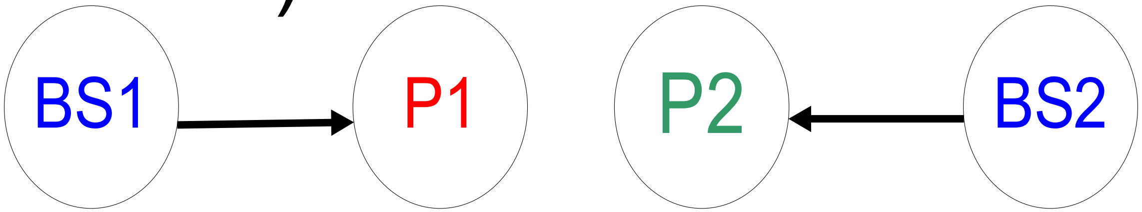
# MACAW: Problem with DS

- If we consider the figure below, then DS does not solve the synchronization issue
- Here BS1 does not know when the contention period is, therefore its only chance to contend for the medium is between a completed data Tx and P2's CTS, otherwise, BS1's RTS will not be responded to with a CTS. never!
- At the same time BS1's BO keeps increasing because it never hears from P1



# MACAW: Resquest-for-RTS(RRTS)

- This problem can be solved by having P1 contend for BS1
- Whenever a station gets a RTS which is not responded to due to a deferral, it contends for the medium during the next contention period and sends a RRTS to BS1
- The recipient of the RRTS responds immediately with a RTS
- Stations that hear a RRTS defer Tx for 2 time slots (enough for RTS-CTS)

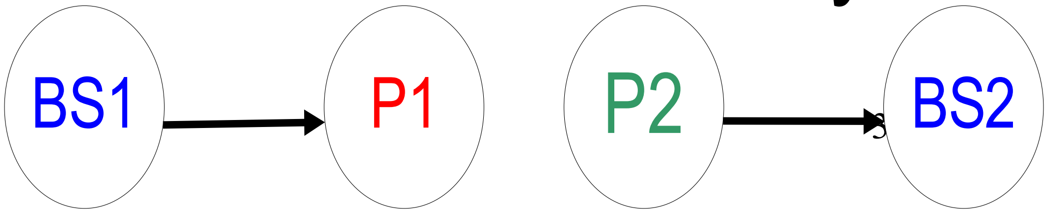


# MACAW: Resquest-for-RTS(RRTS)

- Why RRTS? Why not send CTS when contention period is about to begin?
- RRTS packet size << than CTS packet size
- When nodes overhear the CTS they need to defer long enough for DATA packet
- When nodes overhear the RRTS, only need to defer long enough to overhear the expected CTS
- Using CTS will cause us to wait a longer time which is unnecessary!

# MACAW: Unsloved problem

- Consider the example below, here P1 will never hear a RTS from BS1
- This is due to P2's Tx
- The only time B1 can successfully initiate a transfer is when its RTS arrives during those very short gaps in between a complete data Tx and P2's RTS
- Here BS1 has no way of knowing when the contention period is
- RRTS is irrelevant because P1 can't hear any incoming RTS




Unsolved!

# MACAW: Multicast

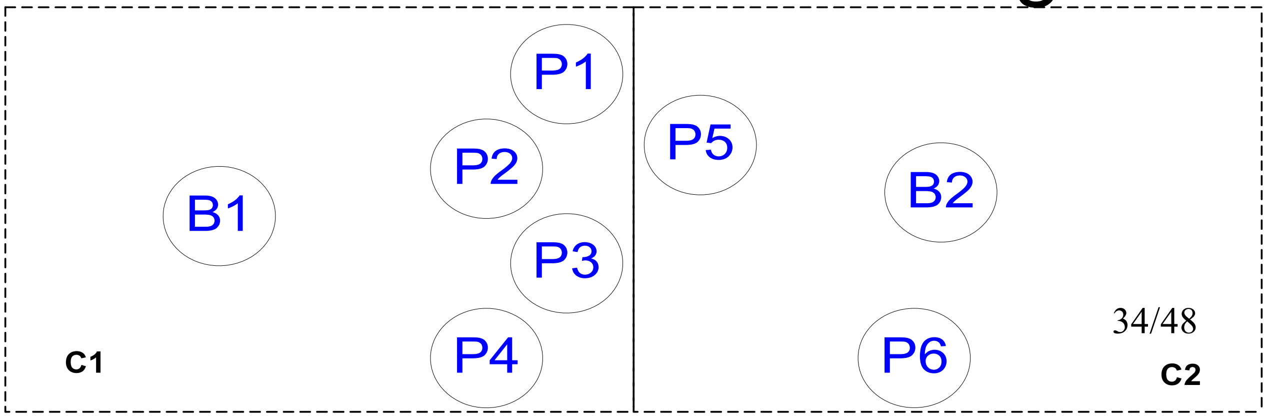
- If one sender and multiple receivers, then if all receivers reply to RTS with CTS collisions will occur
- MACAW addresses this problem by using a RTS-DATA scheme.
- Stations that overhear the RTS know that it is for multicast and therefore defer till after DATA Tx
- Problem: Only stations near sender will defer. Same problem as in CSMA.



Unsloved!

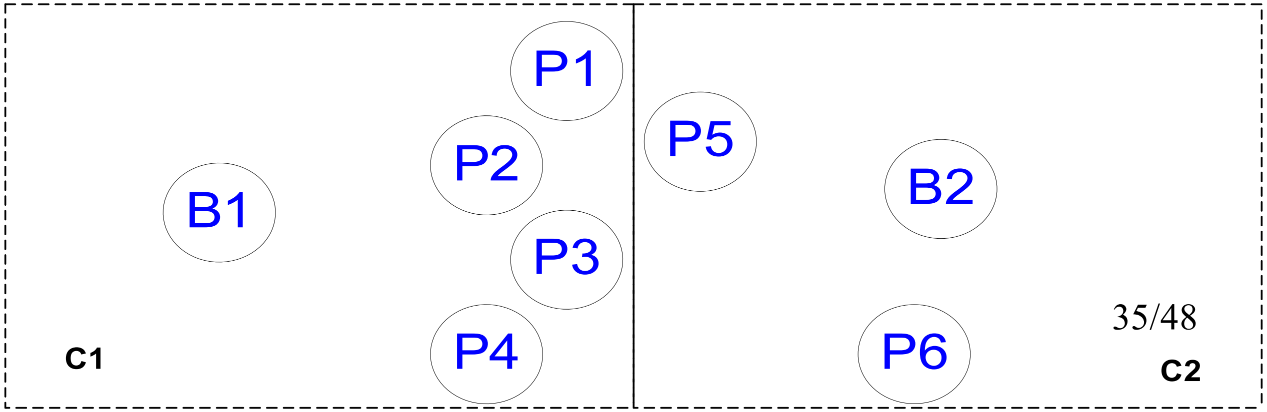
# MACAW: Backoff Algrotithm revisited

- Goal: to achieve high overall throughput and “fair” allocation to streams
- We can see here that the contention level would be high in C1 and relatively low in C2
- Border pads can hear one another, this leads to “leakage” of the backoff values between 2 cells
- P5 copies from P1 to B2 to P6 . Leads to high BO in C2
- Waste of resources!



# MACAW: Backoff Algrotithm revisited cont'd..

- This use of a single number to model congestion in combination with the copying algorithm creates problems.
- The congestion level in one cell can be copied to another cell, although the congestion levels are different

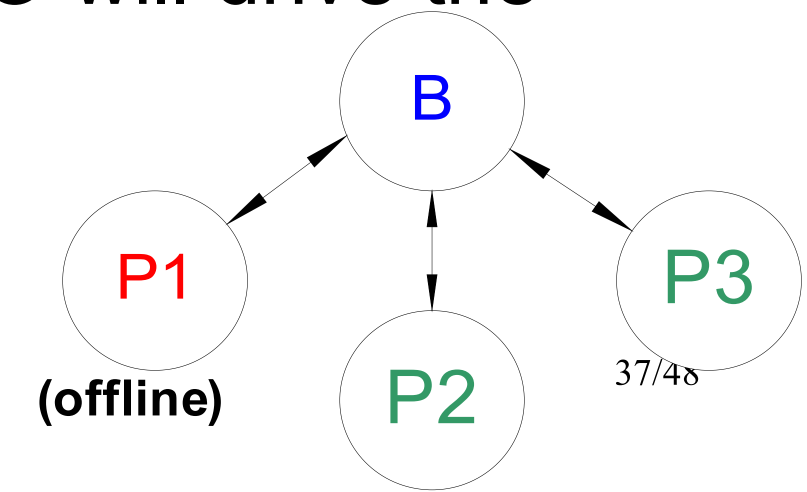


# MACAW: Backoff Algrotithm revisited cont'd..

- So far, MACAW assumed that a RTS failure is due to a collision and collisions reflect congestion
- However, other reasons could cause this (such as noise at either sender or receiver or if receiver is turned off)
- The copying algorithm makes this worse

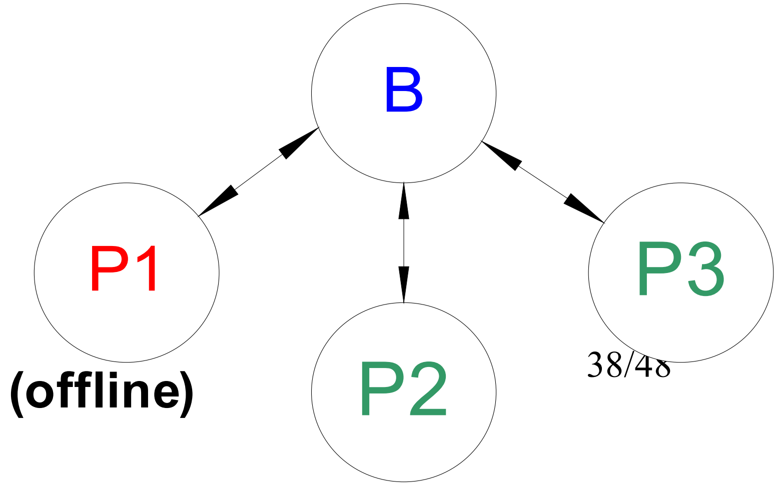
# MACAW: Backoff Algrotithm revisited cont'd..

- Consider the figure below, if P1 is turned off, each time B tries to communicate with P1, it fails and BO will increase
- Each time a successful Tx to P2, BO only decreases by 1
- Due to the fact that MILD increases much faster than it decreases, this high BO will drive the overall throughput to be low



# MACAW: Backoff Algrotithm revisited cont'd..

- This high backoff is copied to the pads that are sending to B
- The problem is exacerbated by the copying algorithm
- Result = low throughput!



# MACAW: Backoff Algrotithm revisited cont'd..

- Authors conclude that we must differentiate between congestion at the sender and at the receiver
- Backoff should reflect congestion at sender and destination
- If RTS is received but CTS is not, we know there is congestion at the sender
- If RTS not receiver, we know there is congestion at receiver, but not sure if there is congestion at the sender as well

# MACAW: Backoff Algrotithm revisited cont'd..

- To achieve fairness all stations attempting to communicate with the same receiving station should use the same backoff value
- This is done by inserting the backoff values of both ends into each packet header
- This per destination backoff copying allows backoff information to be specific for each station, yet copied to all stations that are sending to it

# MACAW: Preliminary Evaluation

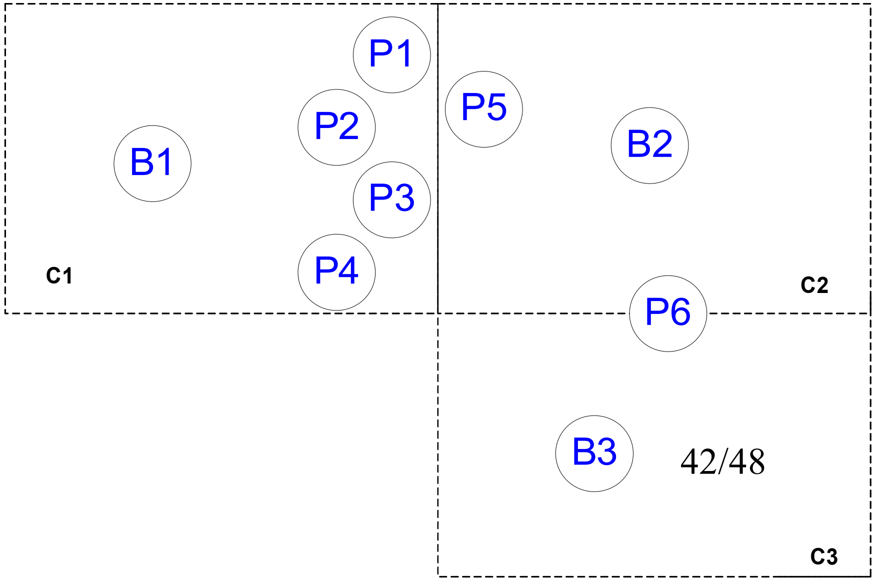
- The addition of the DS and ACK introduce a decrease in throughput of approximately 8%

| | | |
|---|---|---|
| MACA | RTS-CTS-DATA | 53.07 |
| MACAW | RTS-CTS-DS-DATA-ACK | 49.07 |

# MACAW: Preliminary Evaluation cont'd...

- In the scenario below, MACAW yielded an improvement of 37% in throughput
- MACAW also yielded a “fairer” division of the throughput

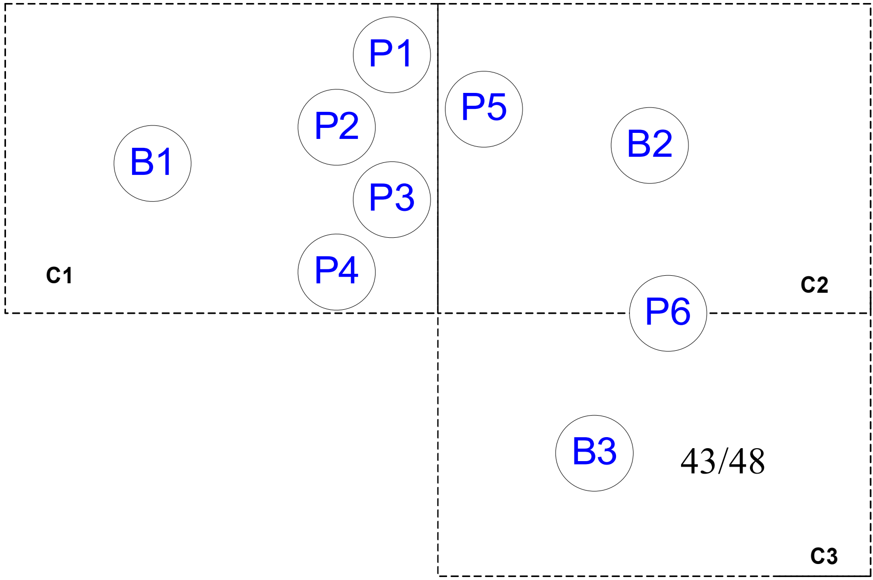
| | MACA | MACAW |
|---|---|---|
| P1-B1 | 9.61 | 3.45 |
| P2-B1 | 2.45 | 3.84 |
| P3-B1 | 3.70 | 3.27 |
| P4-B1 | 0.46 | 3.80 |
| B1-P1 | 0.12 | 3.83 |
| B1-P2 | 0.01 | 3.72 |
| B1-P3 | 0.20 | 3.72 |
| B1-P4 | 0.66 | 3.59 |
| P5-B2 | 2.24 | 7.82 |
| B2-P5 | 3.21 | 7.80 |
| P6-B3 | 28.40 | 25.16 |



# MACAW: Preliminary Evaluation cont'd...

- C1 has a much higher contention level than C3 but they both achieve the same media utilization

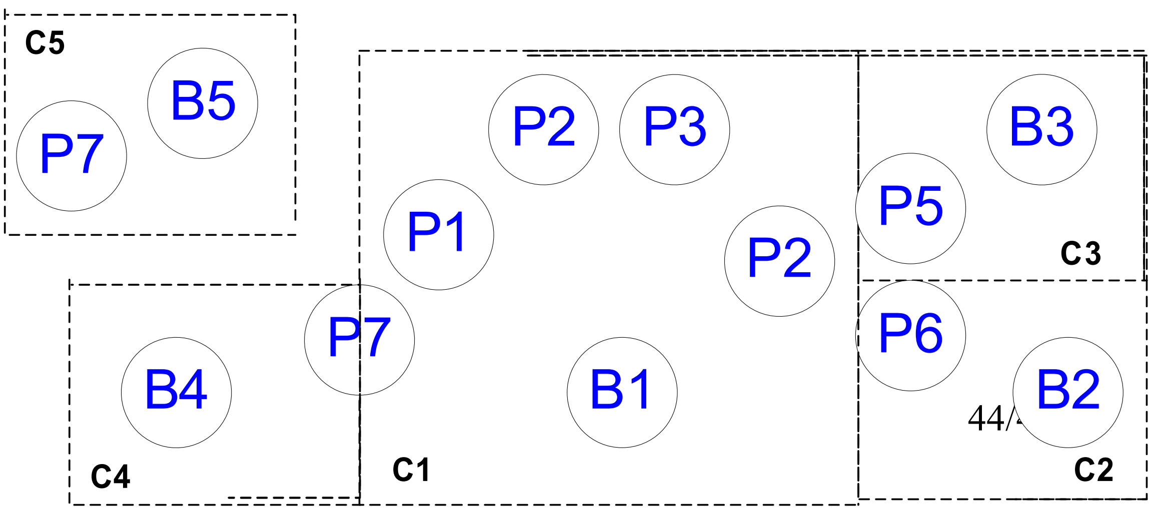
| | MACA | MACAW |
|---|---|---|
| P1-B1 | 9.61 | 3.45 |
| P2-B1 | 2.45 | 3.84 |
| P3-B1 | 3.70 | 3.27 |
| P4-B1 | 0.46 | 3.80 |
| B1-P1 | 0.12 | 3.83 |
| B1-P2 | 0.01 | 3.72 |
| B1-P3 | 0.20 | 3.72 |
| B1-P4 | 0.66 | 3.59 |
| P5-B2 | 2.24 | 7.82 |
| B2-P5 | 3.21 | 7.80 |
| P6-B3 | 28.40 | 25.16 |



# MACAW: Preliminary Evaluation cont'd...

- A noise source is present in C1 (PER 0.01)
- TCP connections used
- P7 can hear P1 and B1

| | MACA | MACAW |
|---|---|---|
| P1-B1 | 0.78 | 2.39 |
| P2-B1 | 1.10 | 2.72 |
| P3-B1 | 0.22 | 2.54 |
| P4-B1 | 0.06 | 2.87 |
| P5-B3 | 18.17 | 14.45 |
| P6-B2 | 6.94 | 14.00 |
| P7-B4 | 23.82 | 19.18 |



# MACAW: Future Design issues

- Use piggy back ACK in subsequent CTS
- Use NACK when data not received correctly
- Use carrier-sense to inhibit RTS-RTS collisions

# MACAW: Summary

- Realize that relevant congestion is at the receiver
- Congestion is not a homogenous phenomenon, it depends on the location of the intended receiver
- MACAW introduced backoff based on stream and further identified which end of the stream is congested
- Learning about congestion levels should be a collective enterprise

# MACAW: Summary cont'd...

- This is why “copying” was introduced.
  - We saw that this can be misleading sometimes
- MAC should propagate synchronization information about contention periods so all devices can contend effectively
  - DS is one example of providing synchronization information
- RRTS is introduced so the receiver can contend when for BW when it is in the presence of congestion

# Questions?