

## Install on all RPi's

- Vim
  - iperf → dest node listens (server)
  - TDMA fixed script
    - ↳ usage: \$ python3 tdma\_send.py
- 
- A hand-drawn diagram illustrating a network setup. In the center, there is a box labeled "dest node". Three arrows point from the left, right, and bottom towards this central box, each originating from a small circle labeled "1", "2", and "3" respectively. This indicates that three separate nodes (labeled 1, 2, and 3) are connected to a single destination node.

# tdma - serv.py

- Cmd line args:
  - dest host
  - dest port
  - Mb to send
- implementation



while (not finished sending bytes):

if (is\_my\_turn()):

Send (1 frame)

How big?

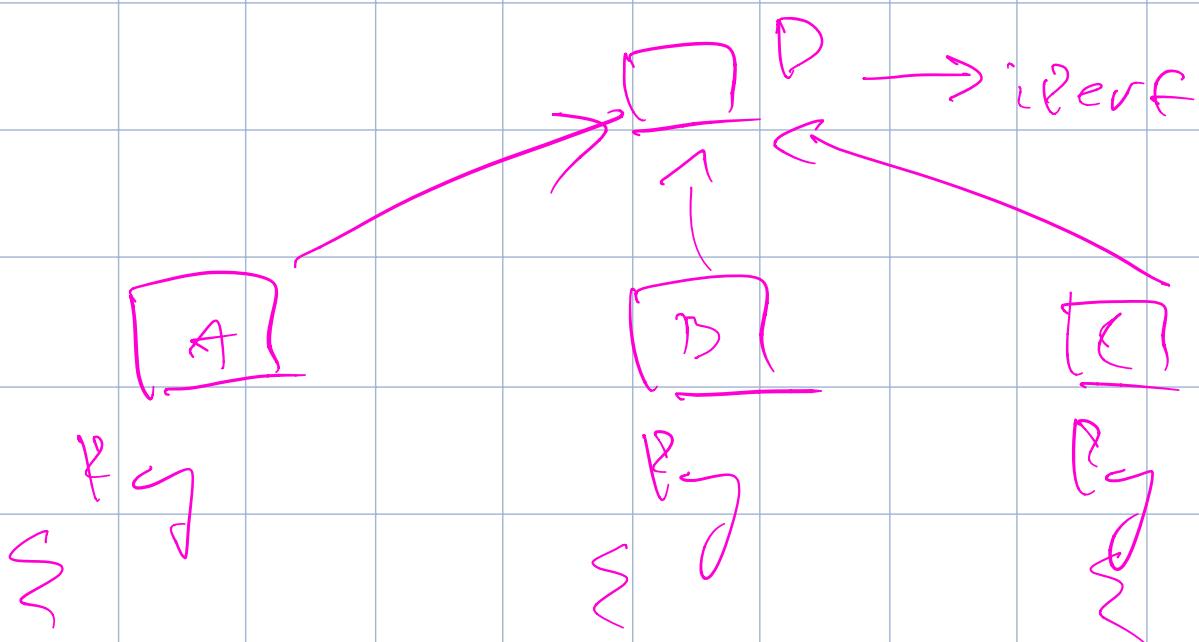
1 Mb?

24 bytes?

→ Research  
medical  
based  
Packet  
sizes

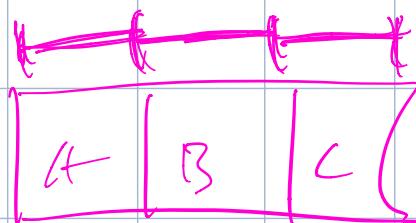
get -> convert view

do



① & chunk size

② Time slot while(



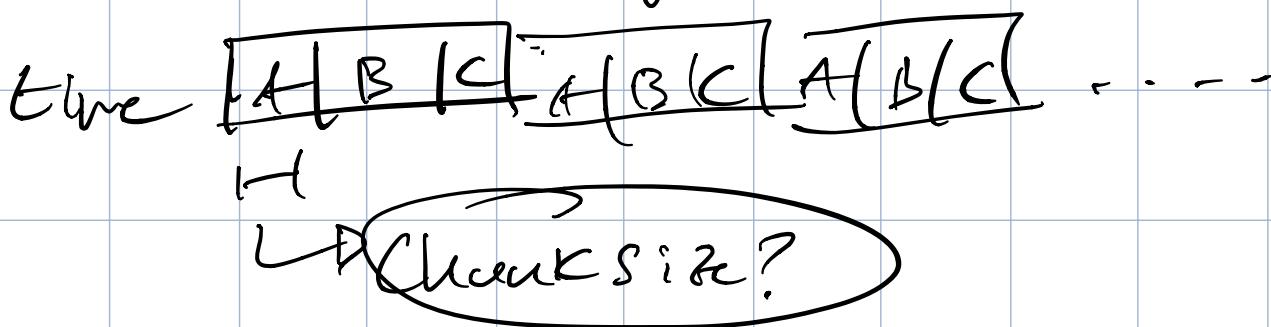
~~TDM~~ → MAC Layer

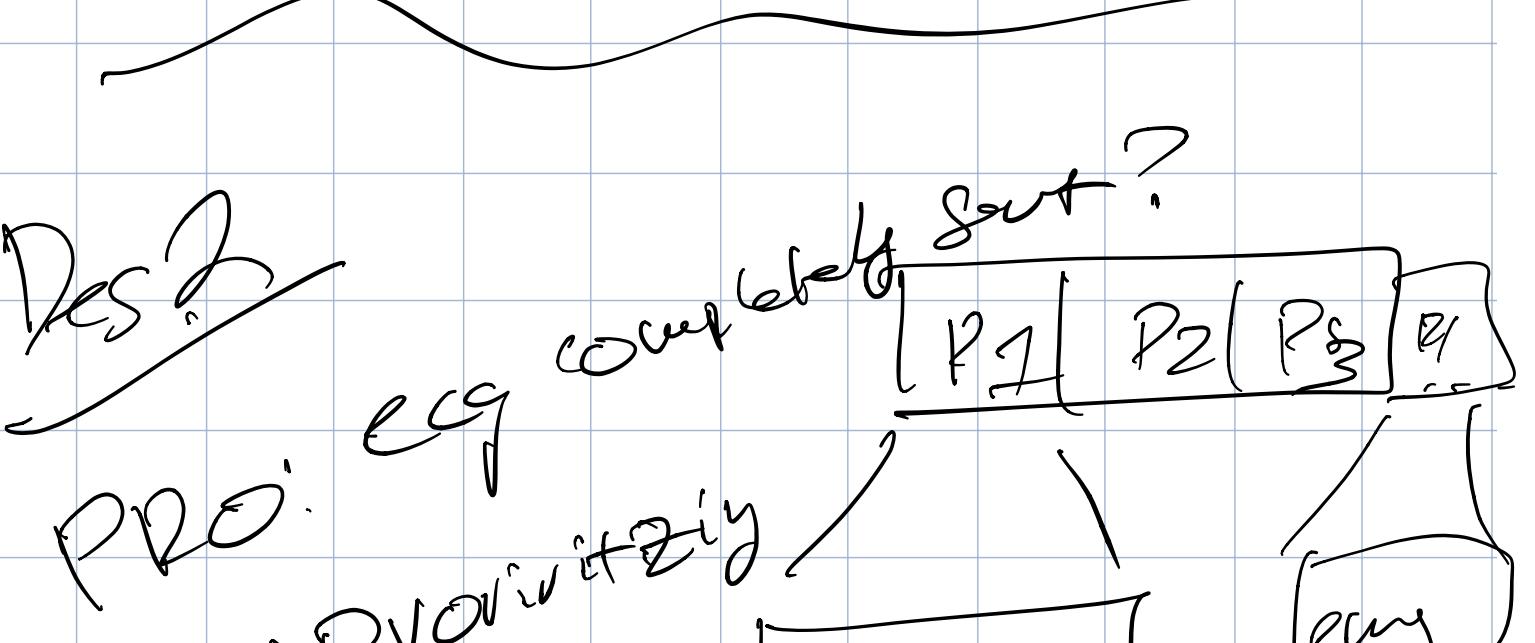
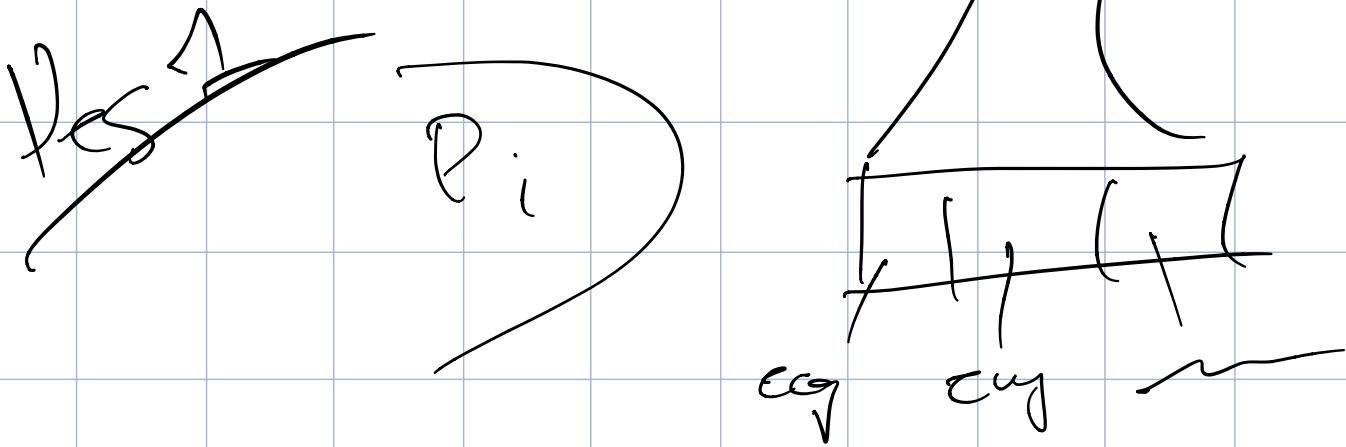
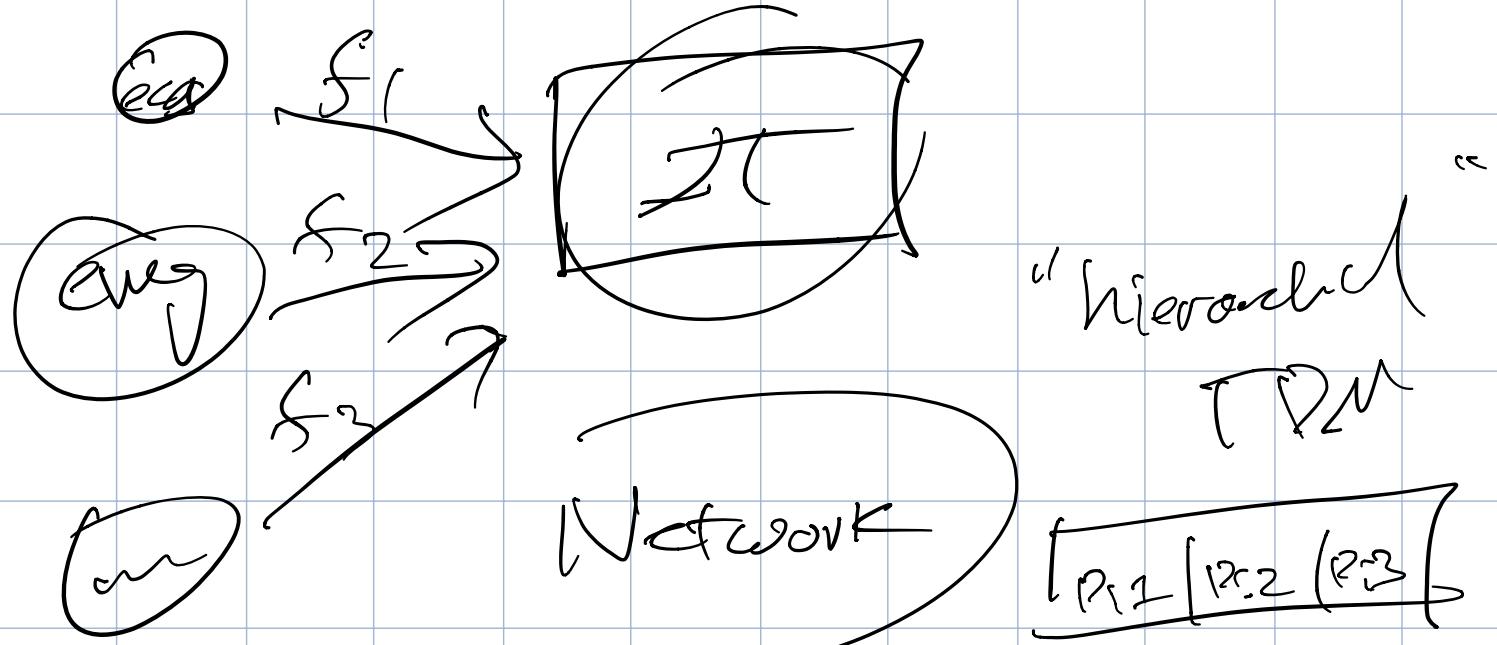
TDM → "Time Division Multiplexing"

---

### V1.0

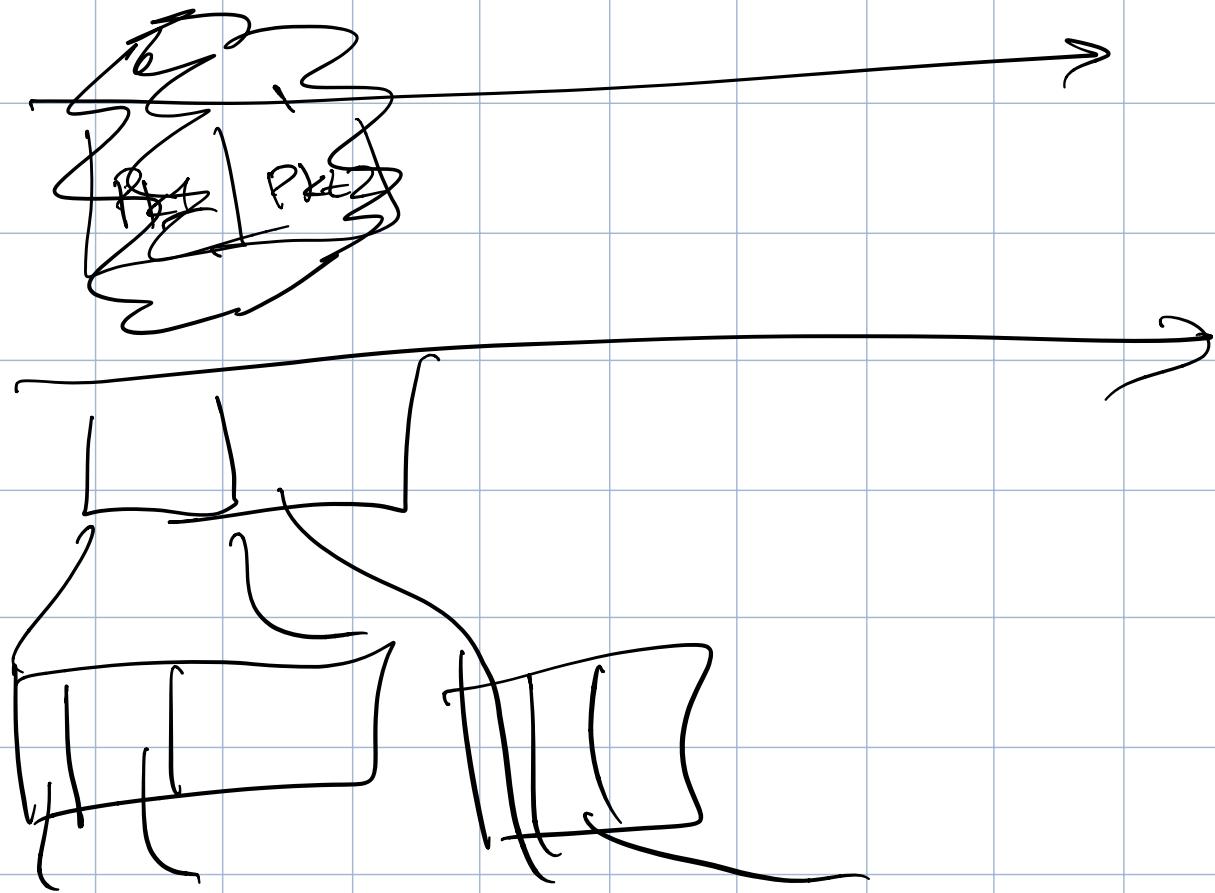
- 3 nodes, fixed five slots
- nodes have equal ~~slots~~ time to send  
(eg 1000 ms)





CV. &  
Save vital signs  
Over others.

susceptible to  
X work time

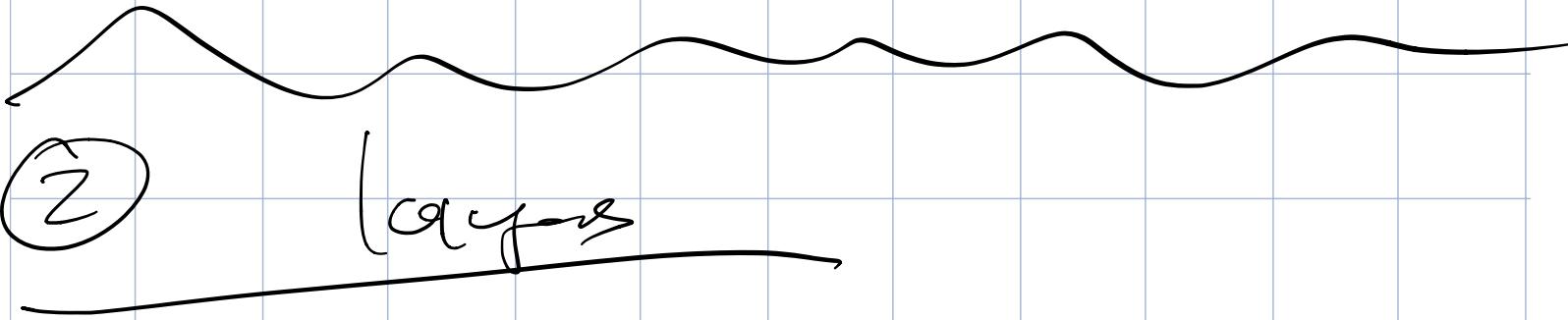


PK1.A PK2.A PK1.B PK2.B

2 ways to "TDM"

① entire pkt in 1 time slot

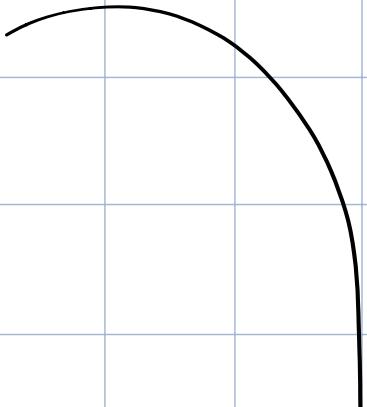
② break pkt up over mult  
time slots



② layers

① Network

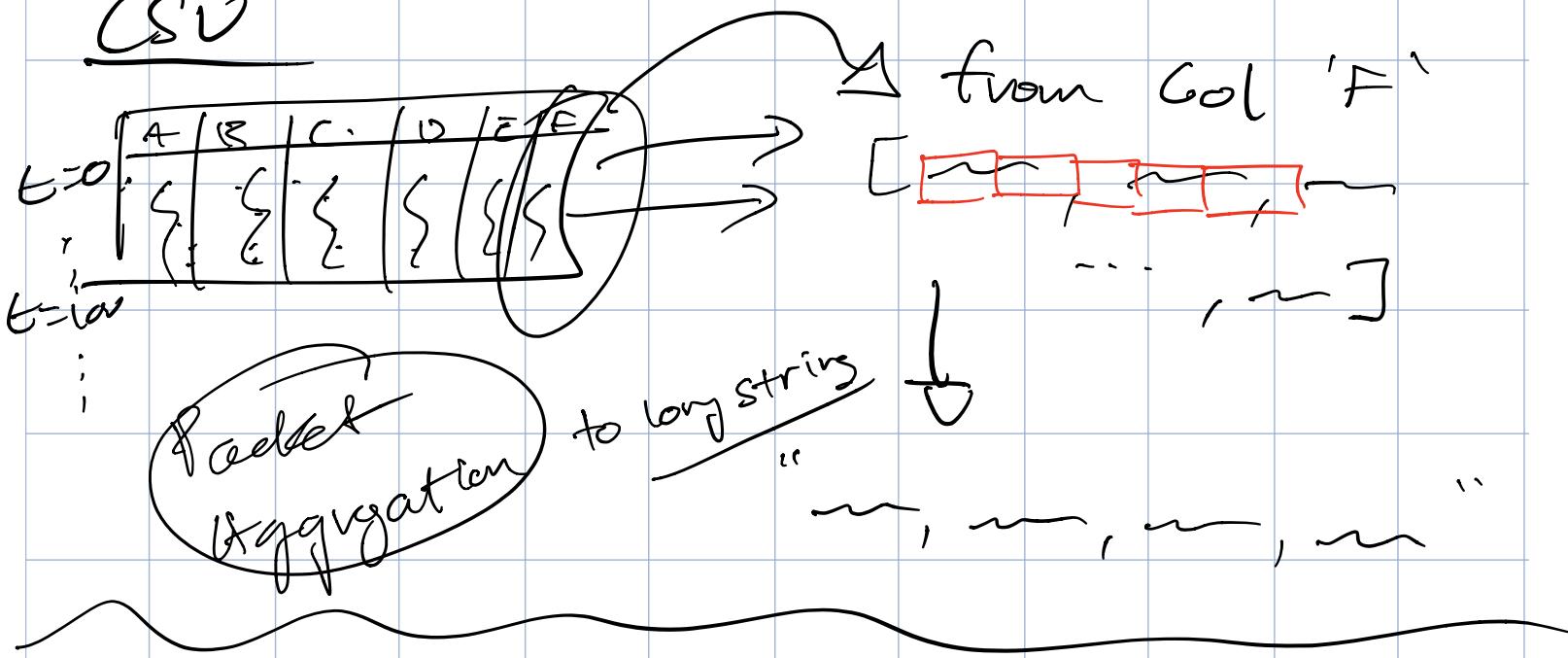
+ "pkt" =



② Pi layer (coord. node)

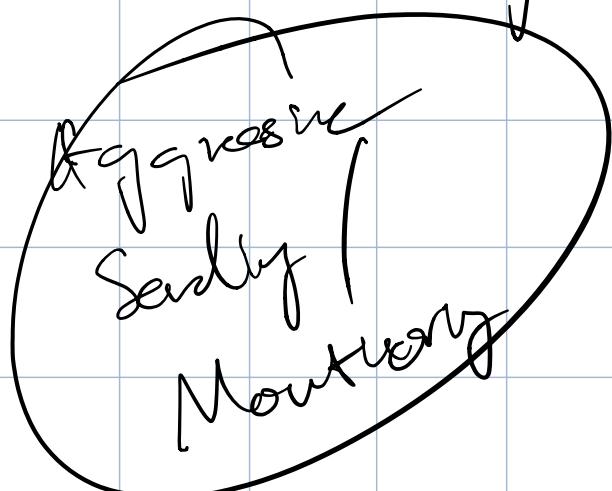
1 "pkt" = ecg ✓  
OR  
=   
ecg ~~gt~~ ekg asc ~~gt~~

CSV



while True:

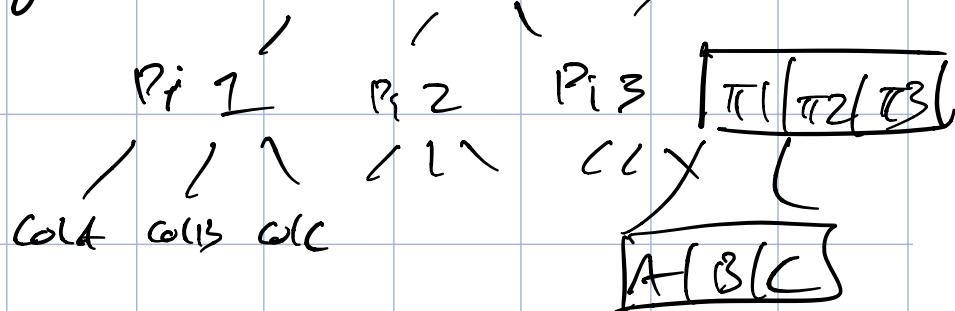
data = gen\_random\_data(~)



# Experiment Variables

- TDM Packet splitting (whole vs split)
- # nodes ( $3 \rightarrow 5 \rightarrow 7$ )  
"Scaling"
  - +2
  - Sam
  - Rai
  - Justin
  - Laura
- Chunk size .. ( $3 \rightarrow 6 \rightarrow 9$ )

② TDM hierarchy (network TRM) -



- Aggressive vs Packet Aggregation  
(eg send multiple flows at once or one flow at a time ...)
- ② Also, # of sensor nodes (cols)

# PLAN

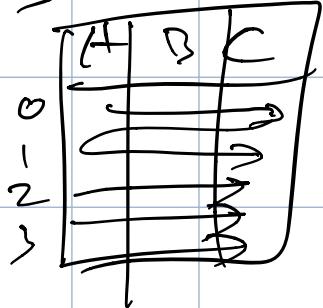
## ① Proof-of-concept

- use iPerf → on Dest Node
- Control data: send cont. stream  
3 clients simult.
- exp. data: fixed TDM scheme

## ② Control

### ③ Main

- No packet aggr
- TDM



## ② Smart

## ③ Scale: 6, 9 nodes

## Implementation

• how TDM in python?

• 3 nodes, IDs: {0, 1, 2}  
↓      ↓      ↓  
0      10      20

time(μs)

0 % (ID \* time\_slot\_length)

1

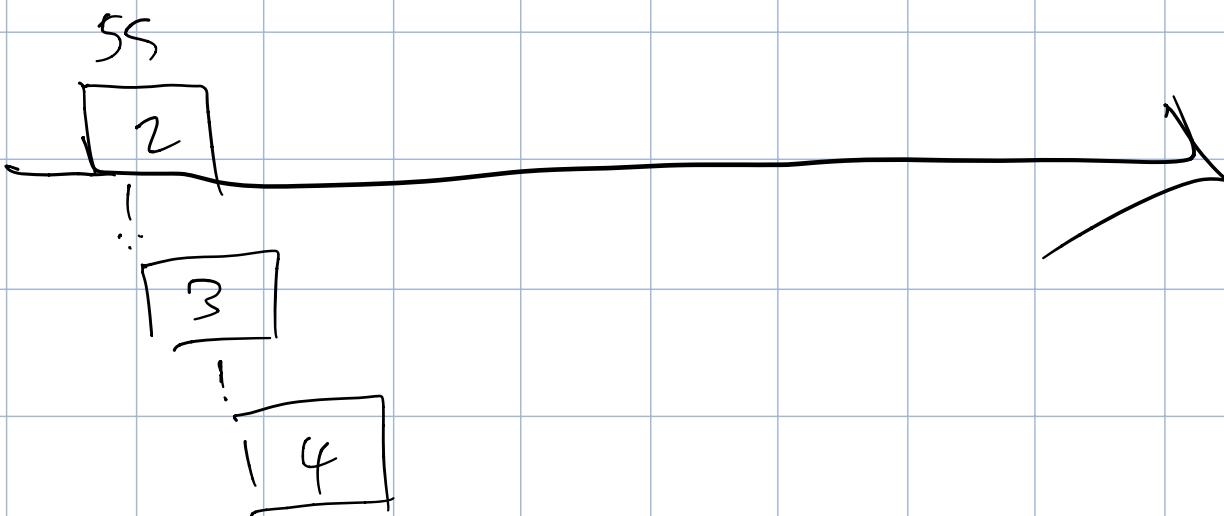
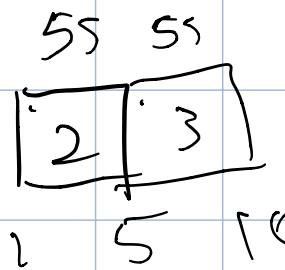
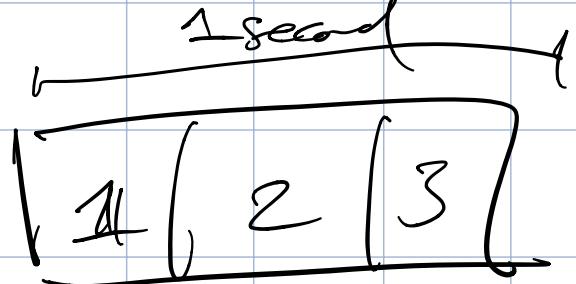
2

:

l00

l01

l02



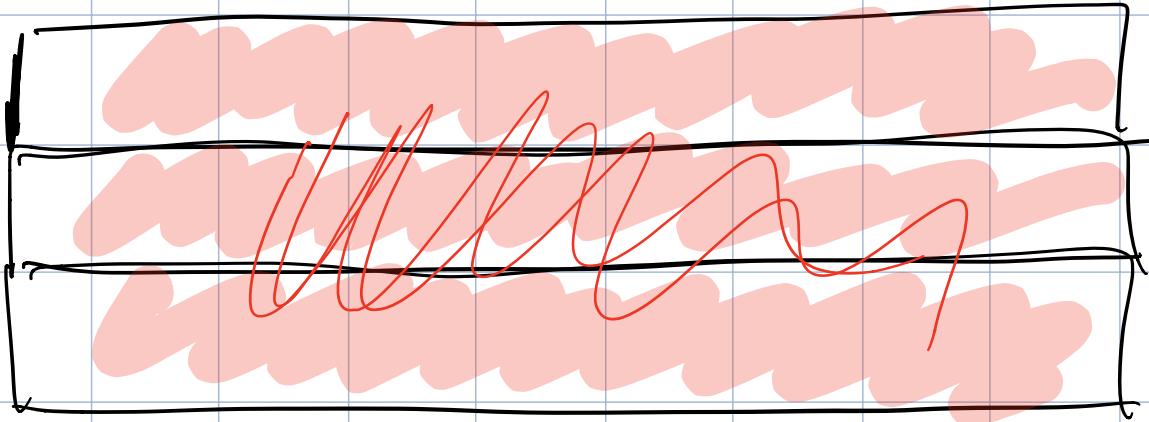
## oo - Stream



2

3

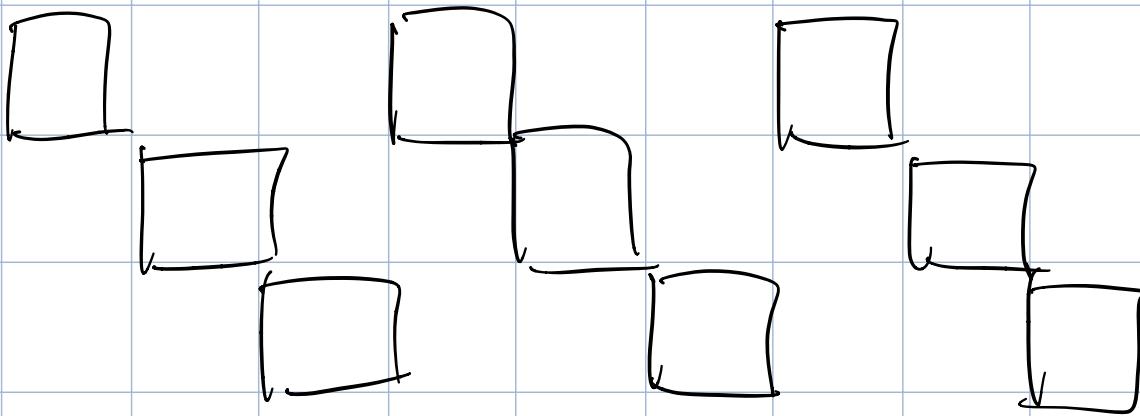
u



2

3

u

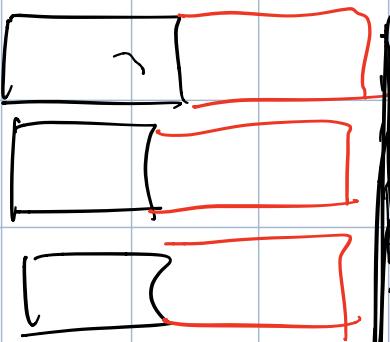


## small-stream (our mistake.)

2

3

u



2

3

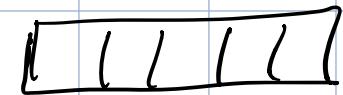
u

pretty much  
the same

# ① Hand-coded scheduling

while True:

    Start = time.time()



    time.sleep(id \* time\_slot)

    while True:

        if (time.time() - Start) > time\_slot: break

        send(frame\_size \* 'A')

Sleep((num\_nodes - 1) \* time\_slot)

2 constants:

FRAME\_SIZE

TIME\_SLOT\_MS

conflicts all at the same time

↑ vs ↓

TDM [ sleep time 1 ]  
⋮

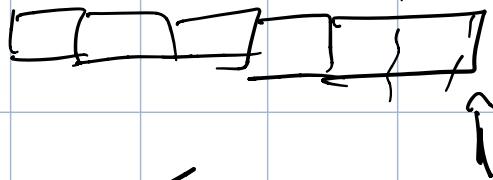
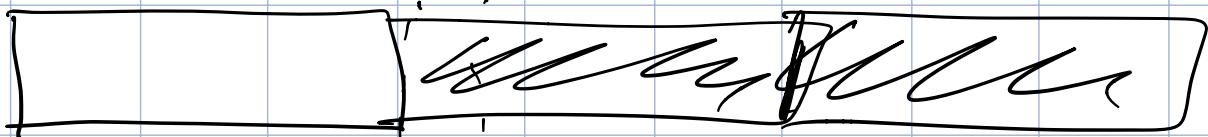
0.5



15

Save  
Changes

1



# Basic

2	3	4	5	6	7
---	---	---	---	---	---

$$f_{\text{total}} = 6$$

O.I.S

2 | 2 | 3 | a | 5 | 6 | ?

Total 26 7

22234567

00000

12/5/18 Update

- Realization WiFi CSMA/CD is very good...
  - ↳ Last night's experiments showed that any time not sending = wasted opportunity...
  - ↳ Can we still achieve anything w/ TDM?
    - Prioritization or nodes > others ✓

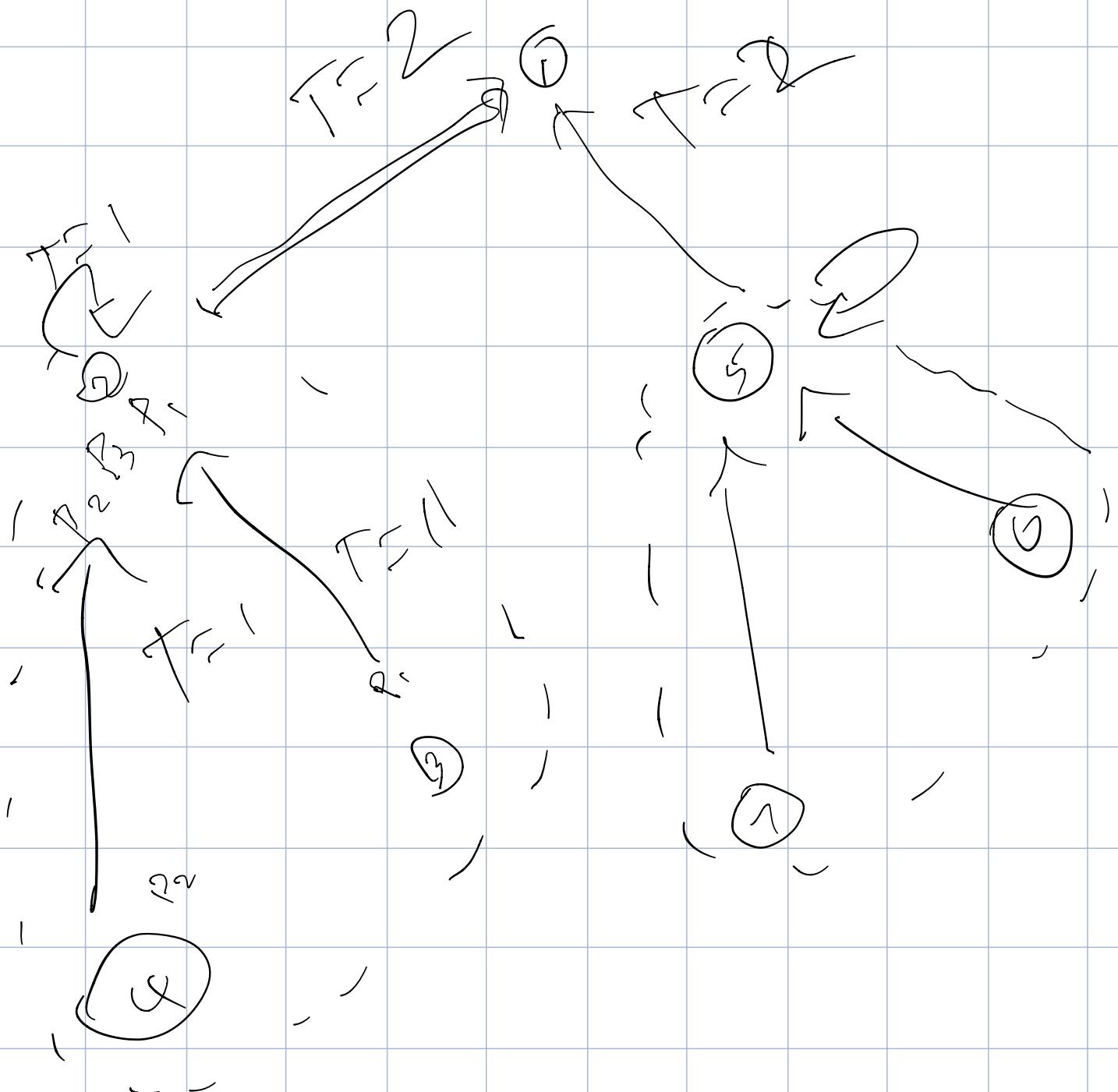


## Report Items

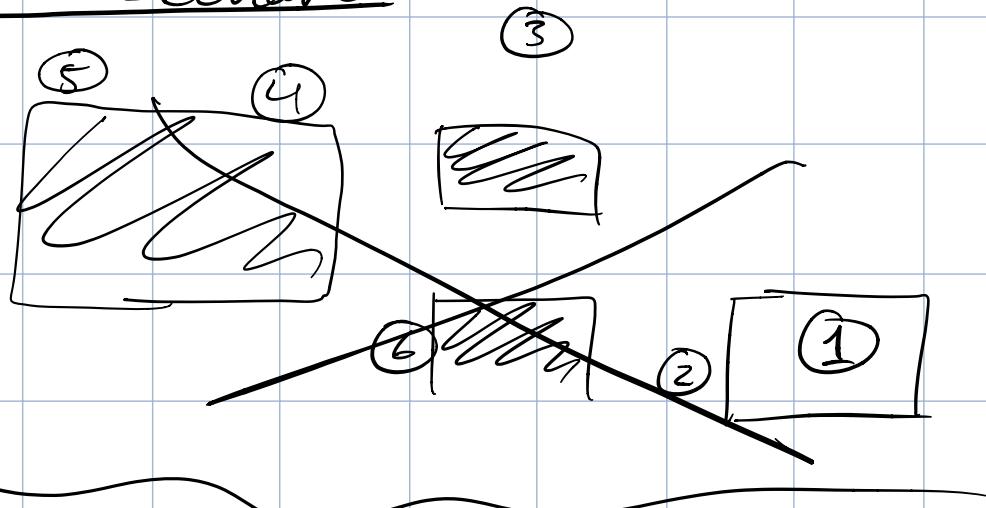
- TDM Introduction
  - Implementation (iperf, Python script)
  - Prioritization ✓
  - On long distances? ↗ TO-DO
  - { - Limitations ↗ WiFi goal  
    → the sync
  - { - Forces of long? Static ↗ WiFi goal  
    → the sync
- Ad-Hoc Real World
  - Implementation (locctrl, locobj)
  - Hopping different floors
  - Moving sender? ↗ Motivation?
    - 3rd world
    - less \$ for infra.
  - Balancer block

## TO → DO

- One more TDM protocol (group)
- Stream vs TDMat
  - fast vs spread aport
- BP Hall exper.
  - long, rise
  - of student traffic

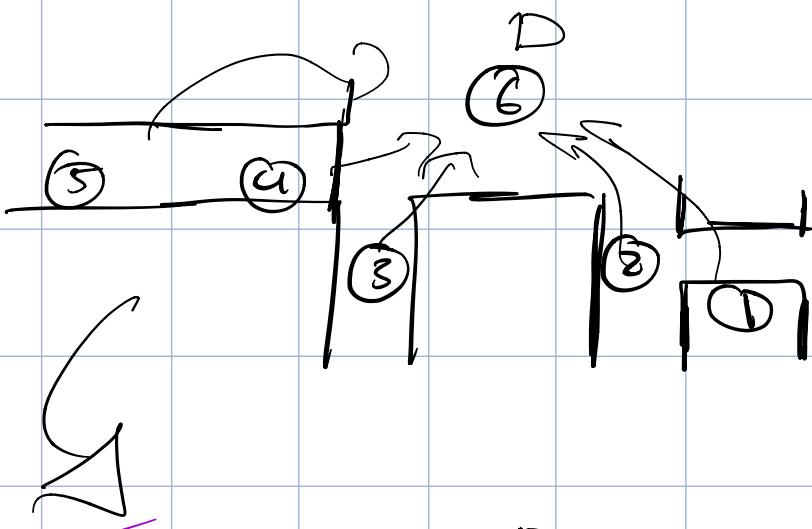


## Current Scenario

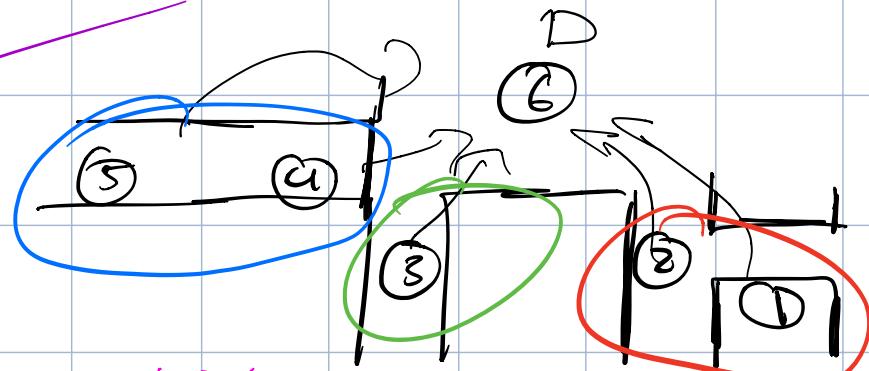


Sail 84

Fish



Group



Group → chain

wait for

885

# Experimental Limitations

- Sam & Chels
- Network conditions change v. quickly even few tests hard to get a baseline measurement
- CSMA/CD on WiFi is great
  - nodes close = No collision occurred
  - ~~so CSJ CD~~
- Node 2 is least

# TDM

## ① Static Scheduling

→ Close together 

→ Spread out  GC hidden nodes

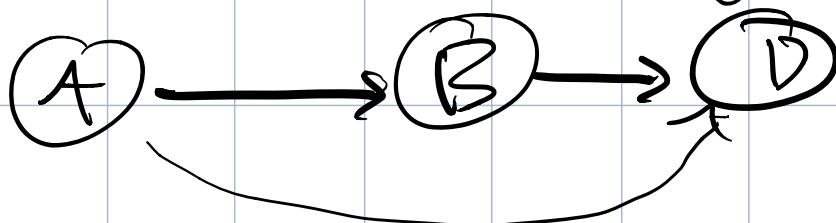
→ Scalloped Fish 'selected' data

## ② Prioritization

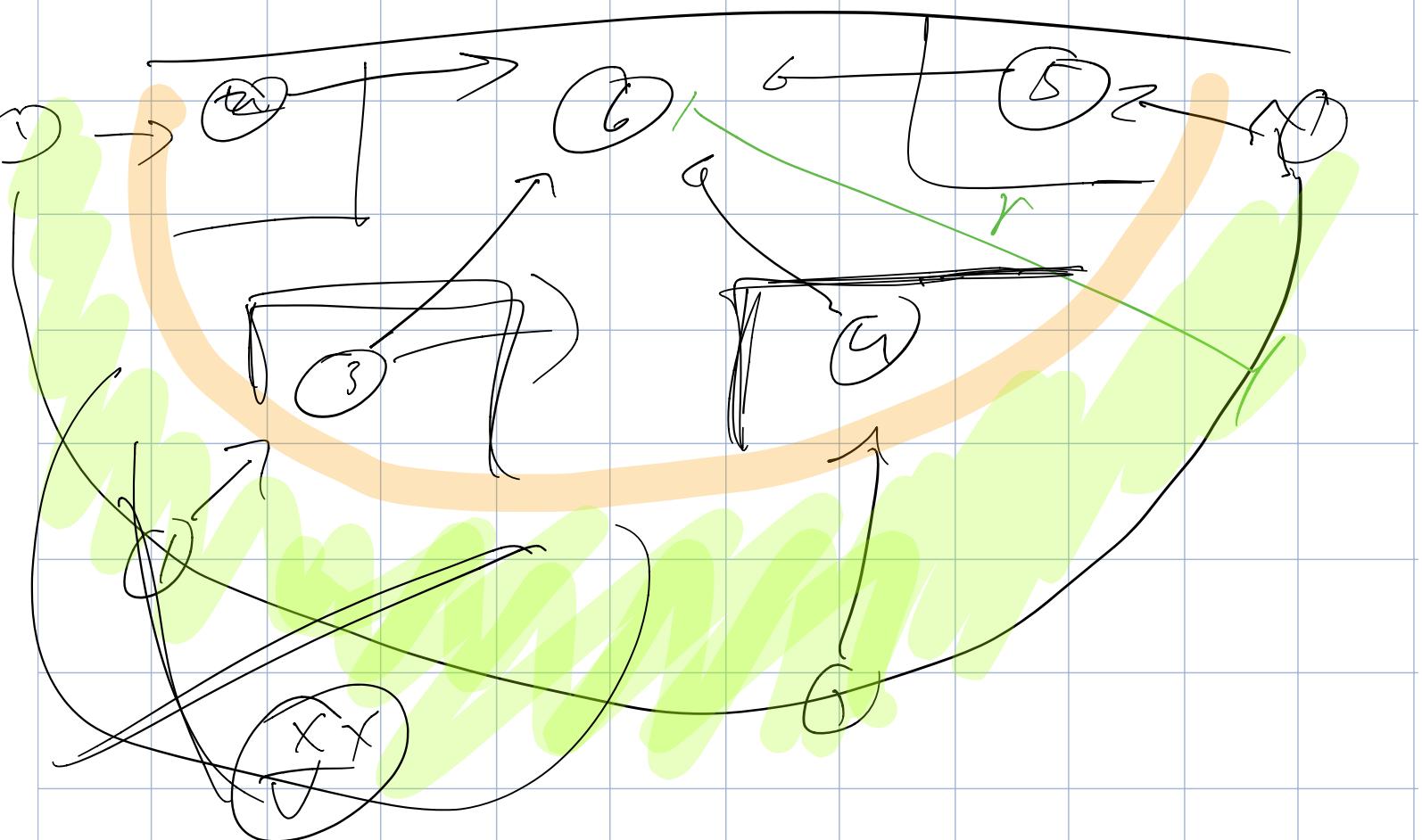
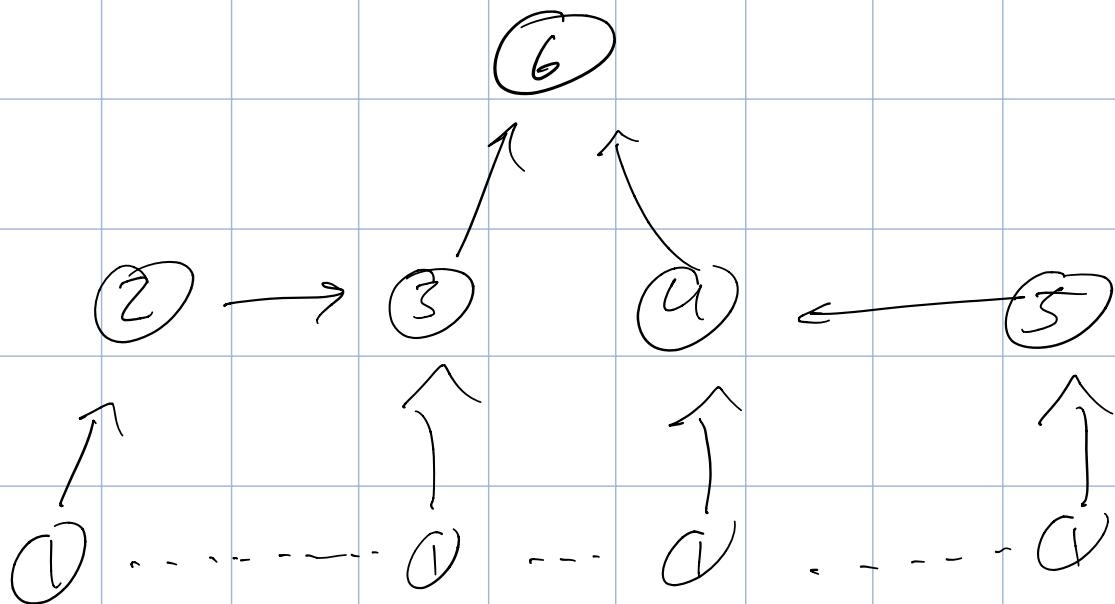
→ IT curves

## ③ Group and

→ weaker nodes 'piggy-back'  
off stronger nodes



# Real-World - Movements



## Motivation

- Try Put ~~Petbat~~ dat  
Sensor network on ~~hat~~  
different board(y & ~~quartz~~)  
to avoid congestion on  
2.4/5 Gblz ✓
- We think is can  
build good system  
for Petbat monitoring

## Implementation/Mobile

- Ad-hoc
  - Point-to-point of blz
  - batman details ?  
*First my no  
be  
carious*

# TDM

- Ghetto TDM
  - together = bad results <sup>be</sup> compared
  - spread out workers vs = good
- Grouped TDM <sup>"daisy-chain"</sup>
  - far <sup>near</sup> workers <sup>"piggy back"</sup>

$5 \rightarrow 4 \rightarrow 6 \rightarrow 2 \rightarrow 1$



- Prioritization

