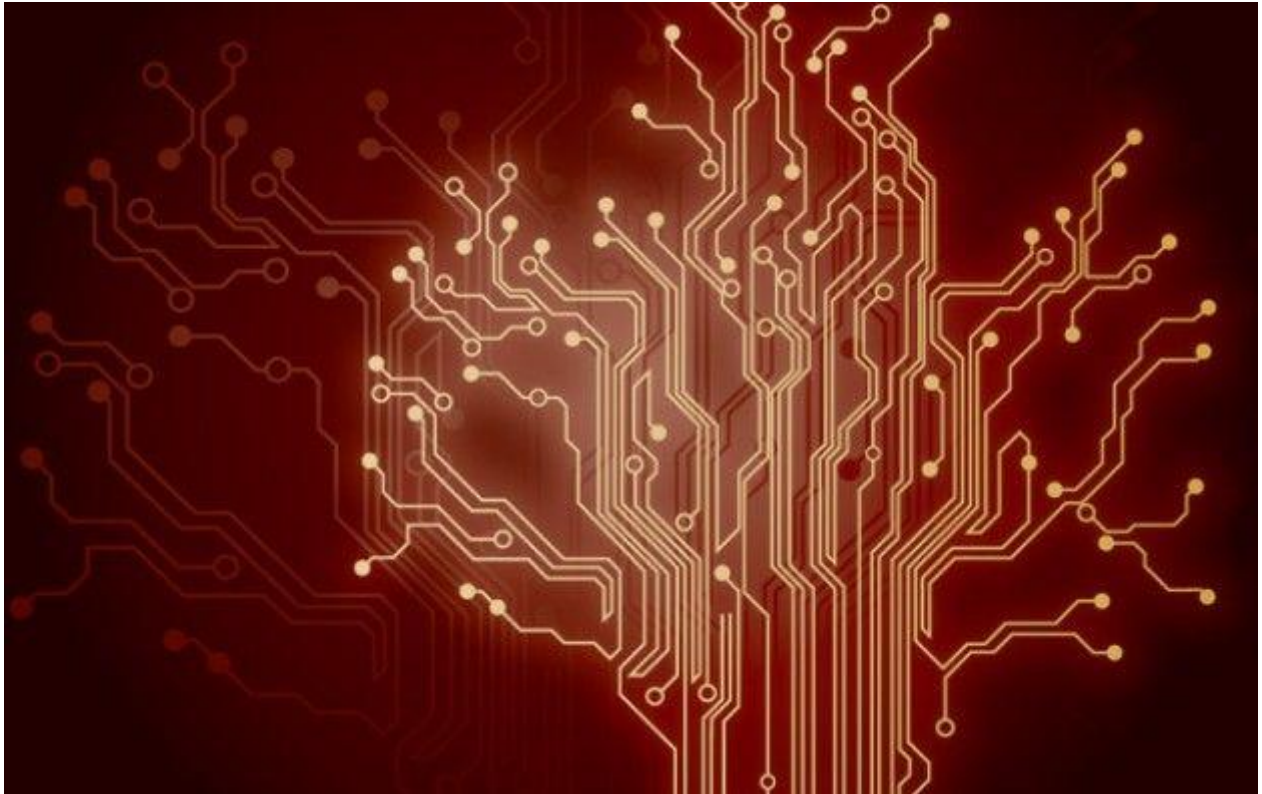


# NETWORK LAB REPORT

*CO3: Implement 1-persistent, non-persistent and p-persistent CSMA techniques.*



**Atanu Ghosh**

BCSE-III (2019-2023) 5th sem, Section: A-1,

Roll: 001910501005, Date: 20/09/2021

## ASSIGNMENT-2

**Implement 1-persistent, non-persistent and p-persistent CSMA techniques.**

### PROBLEM STATEMENT

In this assignment, you have to implement 1-persistent, non-persistent and p-persistent CSMA techniques. Measure the performance parameters like throughput (i.e., average amount of data bits successfully transmitted per unit time) and forwarding delay (i.e., average end-to-end delay, including the queuing delay and the transmission delay) experienced by the CSMA frames (IEEE 802.3). Plot the comparison graphs for throughput and forwarding delay by varying p. State your observations on the impact of performance of different CSMA techniques.

.

## DESIGN

### One-persistent CSMA:

This is the method that is used when the channel has time-slots and that time-slot duration is equal to or greater than the maximum propagation delay time. When the station is ready to send the frames, it will sense the channel. If the channel is found to be busy, the channel will wait for the next slot. If the channel is found to be idle, it transmits the frame with probability  $p$ , thus for the left probability i.e.  $q$  which is equal to  $1-p$  the station will wait for the beginning of the next time slot. In case, when the next slot is also found idle it will transmit or wait again with the probabilities  $p$  and  $q$ . This process is repeated until either the frame gets transmitted or another station has started transmitting.

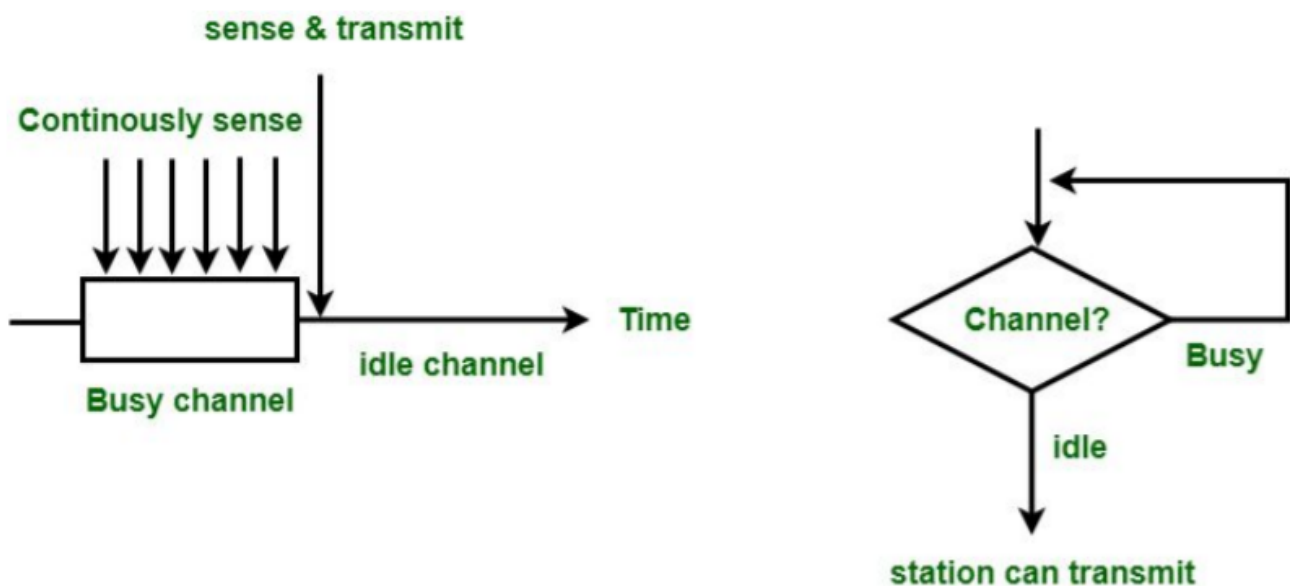


Fig : One persistent CSMA

FLOWCHART :

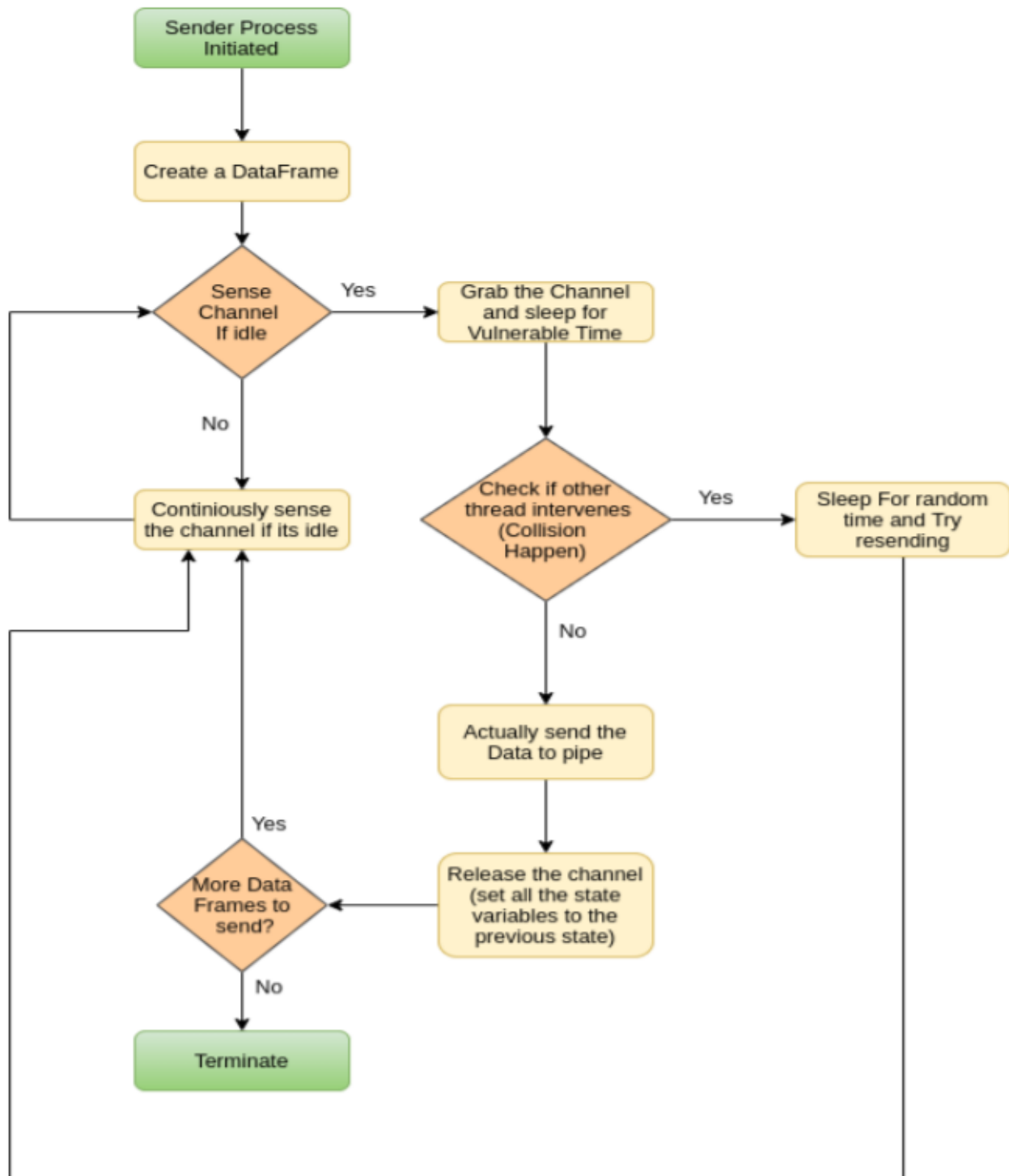


Fig : Implementation of One-persistent CSMA

## Non-persistent CSMA:

In this method, the station which has frames to send, only that station senses for the channel. In case of an idle channel, it will send a frame immediately to that channel. In case when the channel is found busy, it will wait for the random time and again sense for the state of the station whether idle or busy. In this method, the station does not immediately sense the channel for only the purpose of capturing it when it detects the end of the previous transmission. The main advantage of using this method is that it reduces the chances of collision. The problem with this is that it reduces the efficiency of the network.

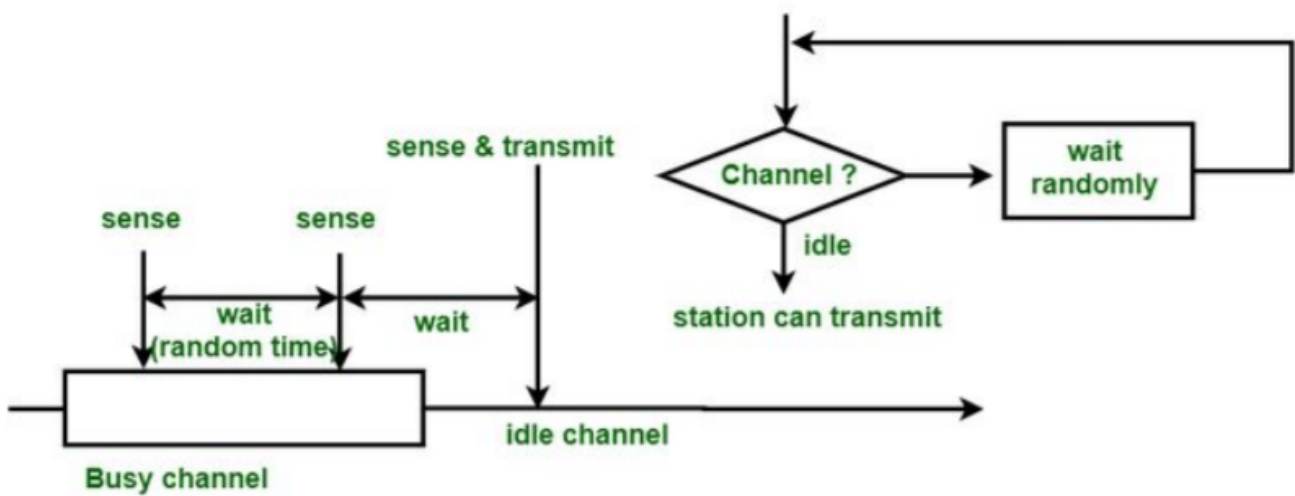


Fig : Non persistent CSMA

FLOWCHART :

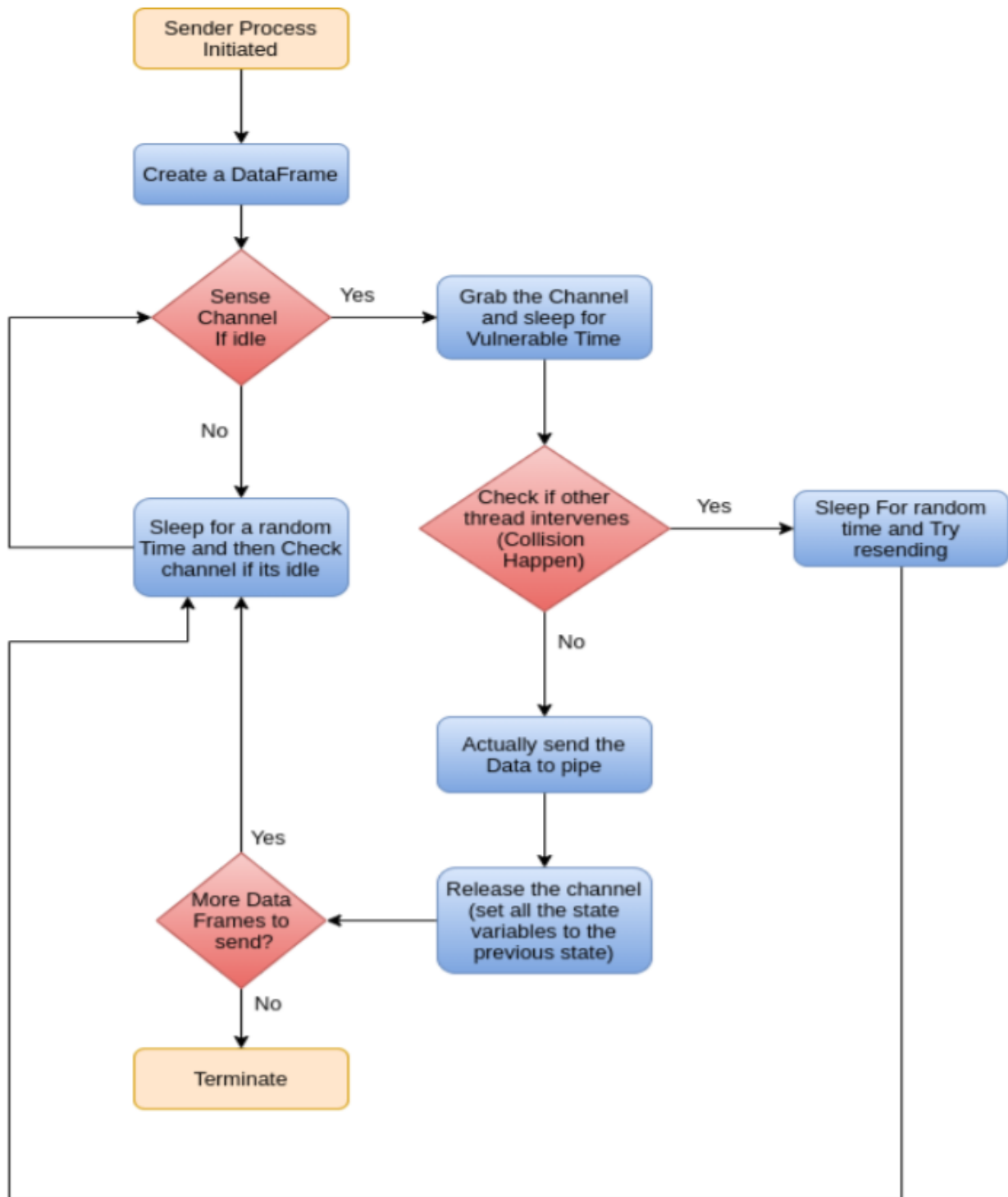


Fig : Implementation of Non-persistent CSMA

## P (1/n) -persistent CSMA:

This is the method that is used when the channel has time-slots and that time-slot duration is equal to or greater than the maximum propagation delay time. When the station is ready to send the frames, it will sense the channel. If the channel is found to be busy, the channel will wait for the next slot. If the channel is found to be idle, it transmits the frame with probability  $p$ , thus for the left probability i.e.  $q$  which is equal to  $1-p$  the station will wait for the beginning of the next time slot. In case, when the next slot is also found idle it will transmit or wait again with the probabilities  $p$  and  $q$ . This process is repeated until either the frame gets transmitted or another station has started transmitting.

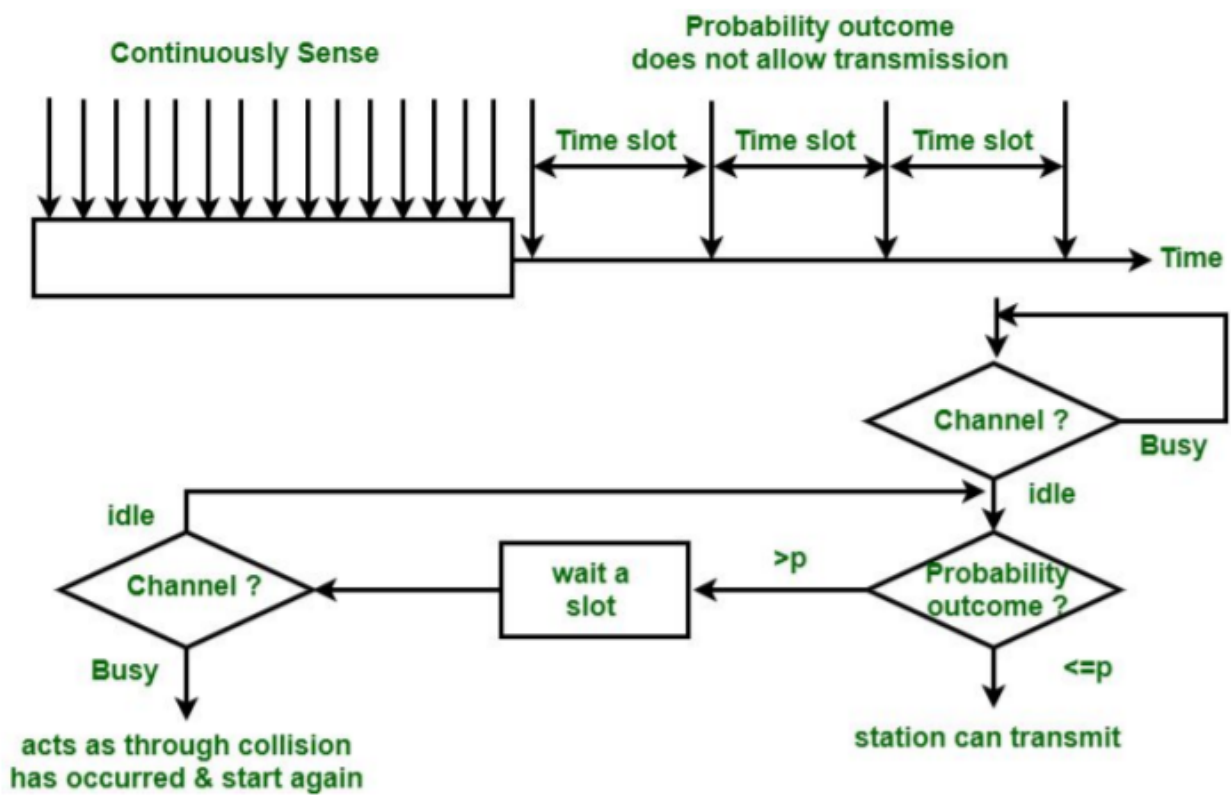


Fig : P persistent CSMA

FLOWCHART :

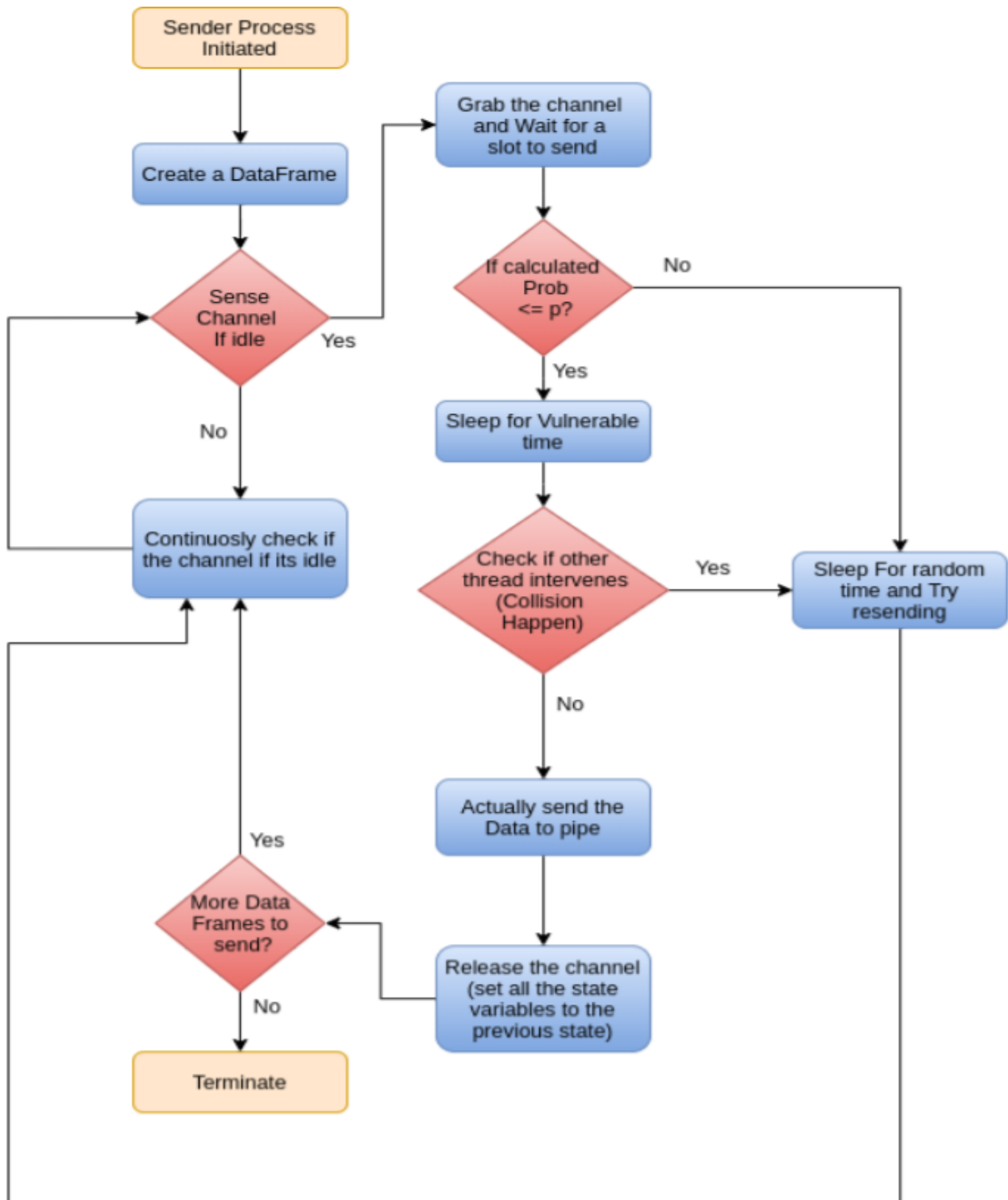


Fig : Implementation of P-persistent CSMA



## SCHEMATIC DIAGRAM (as a part of DESIGN)

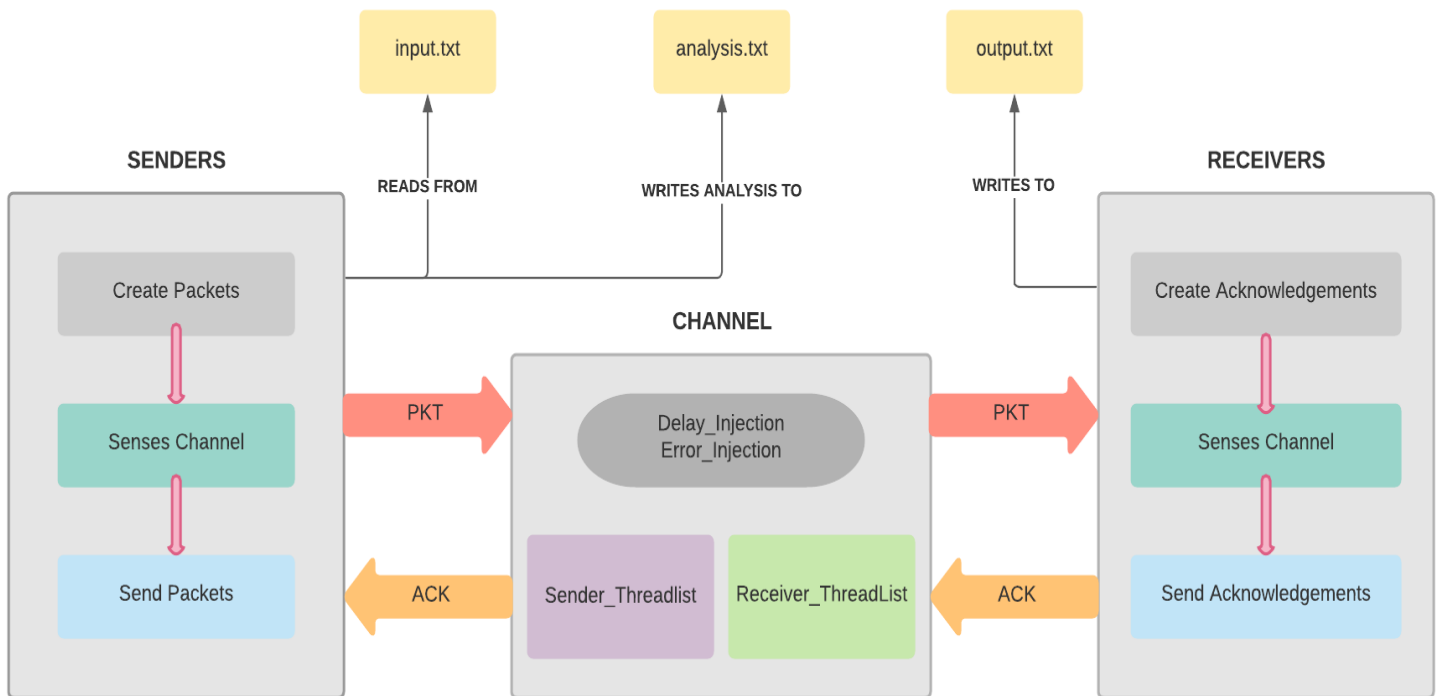


Fig : Schematic diagram to understand the flow of code for CSMA

## IMPLEMENTATION

### A. Packet Structure:

Here for better generalization and uniformity, **IEEE 802.3** frame format has been used throughout the communication. The fields are **Preamble + SFD + DEST\_MAC + SRC\_MAC + Node\_Info (custom addition) + DATA + CheckSum** (  $7 + 1 + 6 + 6 + 2 + 46 + 4 = 72$  Bytes ). Each data packet is encoded with **Checksum-32** for detecting transmission error.

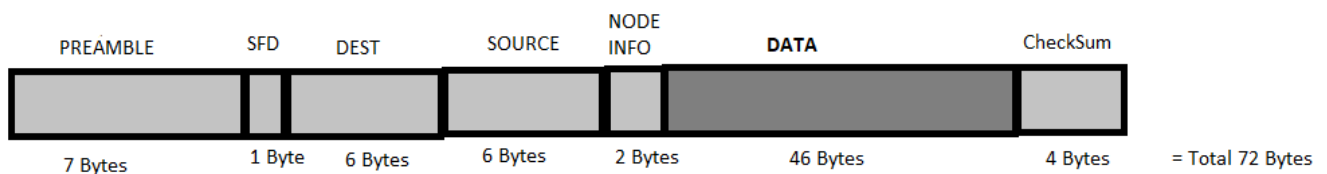


Fig : IEEE 802.3 Frame Format

### B. Algorithm for Sender:

```
select_receiver( )  
make_packets_to_send_by_one/non/P-persistent( )  
sense_signal_to_check_if_busy_or_not( )  
start_sending_packets( )  
receive_confirmation_from_reciver( )  
generate_report_for_performance_analysis( )
```

### C. Algorithm for Channel:

```
channelize_packets_from_sender_to_receiver( )  
channelize_response_from_receiver_to_sender( )  
start_the_threads_for_data_sending_and_response_receiving( )
```

### D. Algorithm for Receiver:

```
decode_which_sender_the_packet_is_coming_from( )  
receive_the_packet( )  
send_packet_reception_confirmation_to_specific_address( )
```

## SOURCE CODE STRUCTURE

The Source-Code folder (`src`) contains -

- `const.py`:

All the constants (number of threads, time-out, packet-size and etc) are defined here.

- `checker.py`:

Contains functions for CheckSum generation and ErrorChecking.

- `gen_packet.py`:

Packet class containing methods for generating packets maintaining IEEE 802.3 format, extracting the original data from the generated packets, decode the source and destination mac addresses, checking errors by CheckSum algorithm, etc.

- `sender.py`:

Sender Class containing methods for selecting a receiver, reading data from the input file(s), selecting the sending method (one/non/p-persistent), adding the packets into stream/channel after sensing whether the channel is busy or not, etc.

- `channel.py`:

Channel Class for channelizing packets from sender, channelizing response from the receiver, starting the sender and receiving threads one by one in queued manner.

- `receiver.py`:

Receiver Class for extracting the data from the receiver packet, writing the confirmation into the output file, etc.

- `main.py`:

Contains support for adding the sender and receiver connections to separate lists one by one, adding the sender and receiver threads to separate lists one by one, and start the sending of packets based on the chosen option and generating a performance report.

- `/textfiles` :

Folder containing input and output files, and the report file (written number of packets sent, received and total delay and calculated throughput).

## OUTPUT & SCREENSHOTS

```
-----  
##### CHOSEN CSMA TECHNIQUE IS : ONE PERSISTENT METHOD #####  
  
20/09/2021 20:08:55 CHANNEL has been initialised  
  
20/09/2021 20:08:55 SENDER-1 || PACKET 1 SENT TO CHANNEL  
20/09/2021 20:08:55 SENDER-2 || PACKET 1 SENT TO CHANNEL  
  
20/09/2021 20:08:56 RECEIVER-1 || PACKET RECEIVED  
  
20/09/2021 20:08:56 SENDER-1 || FOUND CHANNEL BUSY  
20/09/2021 20:08:56 SENDER-2 || FOUND CHANNEL BUSY  
20/09/2021 20:08:57 SENDER-2 || FOUND CHANNEL BUSY  
20/09/2021 20:08:57 SENDER-1 || FOUND CHANNEL BUSY  
  
20/09/2021 20:08:57 RECEIVER-2 || PACKET RECEIVED  
  
20/09/2021 20:08:57 SENDER-2 || PACKET 2 SENT TO CHANNEL  
20/09/2021 20:08:57 SENDER-1 || COLLISION  
20/09/2021 20:08:58 SENDER-1 || PACKET 2 SENT TO CHANNEL  
  
20/09/2021 20:08:58 RECEIVER-2 || PACKET RECEIVED  
  
20/09/2021 20:08:59 SENDER-2 || FOUND CHANNEL BUSY  
20/09/2021 20:08:59 SENDER-1 || FOUND CHANNEL BUSY  
20/09/2021 20:08:59 SENDER-2 || FOUND CHANNEL BUSY  
  
20/09/2021 20:08:59 RECEIVER-1 || PACKET RECEIVED  
  
20/09/2021 20:08:59 SENDER-1 || FOUND CHANNEL BUSY  
20/09/2021 20:09:00 SENDER-2 || PACKET 3 SENT TO CHANNEL  
20/09/2021 20:09:00 SENDER-1 || PACKET 3 SENT TO CHANNEL  
  
20/09/2021 20:09:00 RECEIVER-2 || PACKET RECEIVED  
  
20/09/2021 20:09:01 SENDER-2 || FOUND CHANNEL BUSY  
20/09/2021 20:09:01 SENDER-1 || FOUND CHANNEL BUSY  
20/09/2021 20:09:01 SENDER-2 || FOUND CHANNEL BUSY  
  
20/09/2021 20:09:01 RECEIVER-1 || PACKET RECEIVED  
  
20/09/2021 20:09:01 SENDER-1 || FOUND CHANNEL BUSY  
20/09/2021 20:09:02 SENDER-2 || PACKET 4 SENT TO CHANNEL  
20/09/2021 20:09:02 SENDER-1 || PACKET 4 SENT TO CHANNEL  
  
20/09/2021 20:09:03 RECEIVER-2 || PACKET RECEIVED  
  
20/09/2021 20:09:03 SENDER-2 || FOUND CHANNEL BUSY  
20/09/2021 20:09:03 SENDER-1 || FOUND CHANNEL BUSY  
20/09/2021 20:09:03 SENDER-2 || FOUND CHANNEL BUSY
```

```

53 20/09/2021 20:09:03 RECEIVER-1 || PACKET RECEIVED
54
55 20/09/2021 20:09:03 SENDER-1 || FOUND CHANNEL BUSY
56 20/09/2021 20:09:04 SENDER-2 || PACKET 5 SENT TO CHANNEL
57 20/09/2021 20:09:04 SENDER-1 || PACKET 5 SENT TO CHANNEL
58
59 20/09/2021 20:09:05 RECEIVER-2 || PACKET RECEIVED
60
61 20/09/2021 20:09:05 SENDER-2 || FOUND CHANNEL BUSY
62 20/09/2021 20:09:05 SENDER-1 || FOUND CHANNEL BUSY
63 20/09/2021 20:09:05 SENDER-2 || FOUND CHANNEL BUSY
64
65 20/09/2021 20:09:06 RECEIVER-1 || PACKET RECEIVED
66
67 20/09/2021 20:09:06 SENDER-1 || FOUND CHANNEL BUSY
68 20/09/2021 20:09:06 SENDER-2 || PACKET 6 SENT TO CHANNEL
69 20/09/2021 20:09:06 SENDER-1 || PACKET 6 SENT TO CHANNEL
70
71 20/09/2021 20:09:07 RECEIVER-2 || PACKET RECEIVED
72
73 20/09/2021 20:09:07 SENDER-2 || FOUND CHANNEL BUSY
74 20/09/2021 20:09:07 SENDER-1 || FOUND CHANNEL BUSY
75 20/09/2021 20:09:08 SENDER-2 || FOUND CHANNEL BUSY
76
77 20/09/2021 20:09:08 RECEIVER-1 || PACKET RECEIVED
78
79 20/09/2021 20:09:08 SENDER-1 || FOUND CHANNEL BUSY
80 20/09/2021 20:09:08 SENDER-2 || PACKET 7 SENT TO CHANNEL
81 20/09/2021 20:09:08 SENDER-1 || PACKET 7 SENT TO CHANNEL
82
83 20/09/2021 20:09:09 RECEIVER-2 || PACKET RECEIVED
84
85 20/09/2021 20:09:09 SENDER-2 || FOUND CHANNEL BUSY
86 20/09/2021 20:09:09 SENDER-1 || FOUND CHANNEL BUSY
87 20/09/2021 20:09:10 SENDER-2 || FOUND CHANNEL BUSY
88
89 20/09/2021 20:09:10 RECEIVER-1 || PACKET RECEIVED
90
91 20/09/2021 20:09:10 SENDER-1 || FOUND CHANNEL BUSY
92 20/09/2021 20:09:10 SENDER-2 || PACKET 8 SENT TO CHANNEL
93 20/09/2021 20:09:10 SENDER-1 || PACKET 8 SENT TO CHANNEL
94
95 20/09/2021 20:09:11 RECEIVER-2 || PACKET RECEIVED
96
97
98
99 ***** 20/09/2021 20:09:11 SENDER-2 STATS *****
100 * Total packets: 8
101 * Total Delay: 16.11 secs
102 * Total collisions: 0
103 * Throughput: 1.0
104 *****

```

```

108
109 ***** 20/09/2021 20:09:11 SENDER-1 STATS *****
110 *   Total packets: 8
111 *   Total Delay: 16.25 secs
112 *   Total collisions: 1
113 *   Throughput: 0.889
114 *****
115
116
117
118 20/09/2021 20:09:12 RECEIVER-1 || PACKET RECEIVED
119
120
121 -----
122 | ##### CHOSEN CSMA TECHNIQUE IS : NON PERSISTENT METHOD #####
123 |
124 20/09/2021 20:09:40 CHANNEL has been initialised
125
126 20/09/2021 20:09:40 SENDER-1   || PACKET 1 SENT TO CHANNEL
127 20/09/2021 20:09:40 SENDER-2   || PACKET 1 SENT TO CHANNEL
128
129 20/09/2021 20:09:41 RECEIVER-1 || PACKET RECEIVED
130
131 20/09/2021 20:09:41 SENDER-2   || FOUND CHANNEL BUSY
132 20/09/2021 20:09:41 SENDER-1   || FOUND CHANNEL BUSY
133
134 20/09/2021 20:09:42 RECEIVER-2 || PACKET RECEIVED
135
136 20/09/2021 20:09:42 SENDER-1   || PACKET 2 SENT TO CHANNEL
137 20/09/2021 20:09:42 SENDER-2   || PACKET 2 SENT TO CHANNEL
138
139 20/09/2021 20:09:43 RECEIVER-1 || PACKET RECEIVED
140
141 20/09/2021 20:09:43 SENDER-1   || FOUND CHANNEL BUSY
142 20/09/2021 20:09:43 SENDER-2   || FOUND CHANNEL BUSY
143
144 20/09/2021 20:09:44 RECEIVER-2 || PACKET RECEIVED
145
146 20/09/2021 20:09:44 SENDER-1   || PACKET 3 SENT TO CHANNEL
147 20/09/2021 20:09:44 SENDER-2   || PACKET 3 SENT TO CHANNEL
148
149 20/09/2021 20:09:45 RECEIVER-1 || PACKET RECEIVED
150
151 20/09/2021 20:09:45 SENDER-1   || FOUND CHANNEL BUSY
152 20/09/2021 20:09:45 SENDER-2   || FOUND CHANNEL BUSY
153
154 20/09/2021 20:09:46 RECEIVER-2 || PACKET RECEIVED
155
156 20/09/2021 20:09:46 SENDER-1   || PACKET 4 SENT TO CHANNEL
157 20/09/2021 20:09:46 SENDER-2   || PACKET 4 SENT TO CHANNEL
158

```

```

159 20/09/2021 20:09:47 RECEIVER-1 || PACKET RECEIVED
160
161 20/09/2021 20:09:48 SENDER-1 || FOUND CHANNEL BUSY
162 20/09/2021 20:09:48 SENDER-2 || FOUND CHANNEL BUSY
163
164 20/09/2021 20:09:48 RECEIVER-2 || PACKET RECEIVED
165
166 20/09/2021 20:09:49 SENDER-2 || PACKET 5 SENT TO CHANNEL
167 20/09/2021 20:09:49 SENDER-1 || PACKET 5 SENT TO CHANNEL
168
169 20/09/2021 20:09:49 RECEIVER-2 || PACKET RECEIVED
170
171 20/09/2021 20:09:50 SENDER-1 || FOUND CHANNEL BUSY
172 20/09/2021 20:09:50 SENDER-2 || FOUND CHANNEL BUSY
173
174 20/09/2021 20:09:50 RECEIVER-1 || PACKET RECEIVED
175
176 20/09/2021 20:09:51 SENDER-2 || PACKET 6 SENT TO CHANNEL
177 20/09/2021 20:09:51 SENDER-1 || PACKET 6 SENT TO CHANNEL
178
179 20/09/2021 20:09:52 RECEIVER-2 || PACKET RECEIVED
180
181 20/09/2021 20:09:52 SENDER-1 || FOUND CHANNEL BUSY
182 20/09/2021 20:09:52 SENDER-2 || FOUND CHANNEL BUSY
183
184 20/09/2021 20:09:52 RECEIVER-1 || PACKET RECEIVED
185
186 20/09/2021 20:09:53 SENDER-2 || PACKET 7 SENT TO CHANNEL
187 20/09/2021 20:09:53 SENDER-1 || PACKET 7 SENT TO CHANNEL
188
189 20/09/2021 20:09:54 RECEIVER-1 || PACKET RECEIVED
190
191 20/09/2021 20:09:54 SENDER-1 || FOUND CHANNEL BUSY
192 20/09/2021 20:09:54 SENDER-2 || FOUND CHANNEL BUSY
193
194 20/09/2021 20:09:55 RECEIVER-2 || PACKET RECEIVED
195
196 20/09/2021 20:09:55 SENDER-1 || PACKET 8 SENT TO CHANNEL
197 20/09/2021 20:09:55 SENDER-2 || PACKET 8 SENT TO CHANNEL
198
199 20/09/2021 20:09:56 RECEIVER-1 || PACKET RECEIVED
200
201
202
203 ***** 20/09/2021 20:09:56 SENDER-1 STATS *****
204 * Total packets: 8
205 * Total Delay: 16.06 secs
206 * Total collisions: 0
207 * Throughput: 1.0
208 *****
209

```

```

213 ***** 20/09/2021 20:09:56 SENDER-2 STATS *****
214 *   Total packets: 8
215 *   Total Delay: 16.07 secs
216 *   Total collisions: 0
217 *   Throughput: 1.0
218 *****
219
220
221
222 20/09/2021 20:09:57 RECEIVER-2 || PACKET RECEIVED
223
224
225 -----
226 | ##### CHOSEN CSMA TECHNIQUE IS : P PERSISTENT METHOD #####
227
228 20/09/2021 20:10:12 CHANNEL has been initialised
229
230 20/09/2021 20:10:12 SENDER-2 || PACKET 1 SENT TO CHANNEL
231 20/09/2021 20:10:12 SENDER-1 || PACKET 1 SENT TO CHANNEL
232
233 20/09/2021 20:10:13 RECEIVER-1 || PACKET RECEIVED
234
235 20/09/2021 20:10:13 SENDER-1 || FOUND CHANNEL BUSY
236 20/09/2021 20:10:13 SENDER-2 || FOUND CHANNEL BUSY
237 20/09/2021 20:10:13 SENDER-1 || FOUND CHANNEL BUSY
238
239 20/09/2021 20:10:13 RECEIVER-2 || PACKET RECEIVED
240
241 20/09/2021 20:10:14 SENDER-2 || WAITING
242 20/09/2021 20:10:14 SENDER-1 || WAITING
243 20/09/2021 20:10:14 SENDER-2 || PACKET 2 SENT TO CHANNEL
244 20/09/2021 20:10:14 SENDER-1 || COLLISION OCCURED
245 20/09/2021 20:10:14 SENDER-1 || PACKET 2 SENT TO CHANNEL
246
247 20/09/2021 20:10:15 RECEIVER-2 || PACKET RECEIVED
248
249 20/09/2021 20:10:15 SENDER-2 || FOUND CHANNEL BUSY
250 20/09/2021 20:10:15 SENDER-1 || FOUND CHANNEL BUSY
251 20/09/2021 20:10:16 SENDER-2 || FOUND CHANNEL BUSY
252
253 20/09/2021 20:10:16 RECEIVER-1 || PACKET RECEIVED
254
255 20/09/2021 20:10:16 SENDER-1 || FOUND CHANNEL BUSY
256 20/09/2021 20:10:16 SENDER-2 || PACKET 3 SENT TO CHANNEL
257 20/09/2021 20:10:16 SENDER-1 || WAITING
258 20/09/2021 20:10:17 SENDER-1 || FOUND CHANNEL BUSY
259 20/09/2021 20:10:17 SENDER-1 || FOUND CHANNEL BUSY
260
261 20/09/2021 20:10:17 RECEIVER-2 || PACKET RECEIVED
262
263 20/09/2021 20:10:17 SENDER-2 || PACKET 4 SENT TO CHANNEL
264 20/09/2021 20:10:18 SENDER-1 || FOUND CHANNEL BUSY
265 20/09/2021 20:10:18 SENDER-1 || FOUND CHANNEL BUSY

```



```

263 20/09/2021 20:10:17 SENDER-2 || PACKET 4 SENT TO CHANNEL
264 20/09/2021 20:10:18 SENDER-1 || FOUND CHANNEL BUSY
265 20/09/2021 20:10:18 SENDER-1 || FOUND CHANNEL BUSY
266
267 20/09/2021 20:10:18 RECEIVER-2 || PACKET RECEIVED
268
269 20/09/2021 20:10:18 SENDER-2 || PACKET 5 SENT TO CHANNEL
270 20/09/2021 20:10:19 SENDER-1 || WAITING
271 20/09/2021 20:10:19 SENDER-1 || FOUND CHANNEL BUSY
272
273 20/09/2021 20:10:19 RECEIVER-2 || PACKET RECEIVED
274
275 20/09/2021 20:10:19 SENDER-1 || FOUND CHANNEL BUSY
276 20/09/2021 20:10:20 SENDER-2 || WAITING
277 20/09/2021 20:10:20 SENDER-2 || WAITING
278 20/09/2021 20:10:20 SENDER-1 || PACKET 3 SENT TO CHANNEL
279 20/09/2021 20:10:20 SENDER-2 || FOUND CHANNEL BUSY
280 20/09/2021 20:10:21 SENDER-2 || FOUND CHANNEL BUSY
281
282 20/09/2021 20:10:21 RECEIVER-1 || PACKET RECEIVED
283
284 20/09/2021 20:10:21 SENDER-1 || WAITING
285 20/09/2021 20:10:21 SENDER-2 || PACKET 6 SENT TO CHANNEL
286 20/09/2021 20:10:21 SENDER-1 || PACKET 4 SENT TO CHANNEL
287
288 20/09/2021 20:10:22 RECEIVER-2 || PACKET RECEIVED
289
290 20/09/2021 20:10:22 SENDER-2 || FOUND CHANNEL BUSY
291 20/09/2021 20:10:22 SENDER-1 || FOUND CHANNEL BUSY
292 20/09/2021 20:10:23 SENDER-2 || FOUND CHANNEL BUSY
293
294 20/09/2021 20:10:23 RECEIVER-1 || PACKET RECEIVED
295
296 20/09/2021 20:10:23 SENDER-1 || PACKET 5 SENT TO CHANNEL
297 20/09/2021 20:10:23 SENDER-2 || FOUND CHANNEL BUSY
298 20/09/2021 20:10:24 SENDER-2 || FOUND CHANNEL BUSY
299
300 20/09/2021 20:10:24 RECEIVER-1 || PACKET RECEIVED
301
302 20/09/2021 20:10:24 SENDER-1 || PACKET 6 SENT TO CHANNEL
303 20/09/2021 20:10:24 SENDER-2 || FOUND CHANNEL BUSY
304 20/09/2021 20:10:25 SENDER-2 || FOUND CHANNEL BUSY
305
306 20/09/2021 20:10:25 RECEIVER-1 || PACKET RECEIVED
307
308 20/09/2021 20:10:25 SENDER-1 || WAITING
309 20/09/2021 20:10:25 SENDER-2 || PACKET 7 SENT TO CHANNEL
310 20/09/2021 20:10:25 SENDER-1 || WAITING
311 20/09/2021 20:10:26 SENDER-1 || FOUND CHANNEL BUSY
312 20/09/2021 20:10:26 SENDER-1 || FOUND CHANNEL BUSY
313
314 20/09/2021 20:10:26 RECEIVER-2 || PACKET RECEIVED

```

```

299
300 20/09/2021 20:10:24 RECEIVER-1 || PACKET RECEIVED
301
302 20/09/2021 20:10:24 SENDER-1 || PACKET 6 SENT TO CHANNEL
303 20/09/2021 20:10:24 SENDER-2 || FOUND CHANNEL BUSY
304 20/09/2021 20:10:25 SENDER-2 || FOUND CHANNEL BUSY
305
306 20/09/2021 20:10:25 RECEIVER-1 || PACKET RECEIVED
307
308 20/09/2021 20:10:25 SENDER-1 || WAITING
309 20/09/2021 20:10:25 SENDER-2 || PACKET 7 SENT TO CHANNEL
310 20/09/2021 20:10:25 SENDER-1 || WAITING
311 20/09/2021 20:10:26 SENDER-1 || FOUND CHANNEL BUSY
312 20/09/2021 20:10:26 SENDER-1 || FOUND CHANNEL BUSY
313
314 20/09/2021 20:10:26 RECEIVER-2 || PACKET RECEIVED
315
316 20/09/2021 20:10:26 SENDER-2 || PACKET 8 SENT TO CHANNEL
317 20/09/2021 20:10:27 SENDER-1 || FOUND CHANNEL BUSY
318 20/09/2021 20:10:27 SENDER-1 || FOUND CHANNEL BUSY
319
320 20/09/2021 20:10:27 RECEIVER-2 || PACKET RECEIVED
321
322
323
324 ***** 20/09/2021 20:10:28 SENDER-2 STATS *****
325 * Total packets: 8
326 * Total Delay: 15.94 secs
327 * Total collisions: 0
328 * Throughput: 1.0
329 *****
330
331
332 20/09/2021 20:10:28 SENDER-1 || PACKET 7 SENT TO CHANNEL
333
334 20/09/2021 20:10:29 RECEIVER-1 || PACKET RECEIVED
335
336 20/09/2021 20:10:29 SENDER-1 || PACKET 8 SENT TO CHANNEL
337
338 20/09/2021 20:10:30 RECEIVER-1 || PACKET RECEIVED
339
340
341
342 ***** 20/09/2021 20:10:30 SENDER-1 STATS *****
343 * Total packets: 8
344 * Total Delay: 18.34 secs
345 * Total collisions: 1
346 * Throughput: 0.889
347 *****
348

```

## RESULTS and ANALYSIS

To calculate and get an idea about the avg bandwidth, I turn off all the delay elements in the channel and set the Channel to transmit data with a frame size of 72 bytes. It came out to be approximately 500 kbps.

- BandWidth Calculated = 500 Kbps

- Avg packets send from each sender = 13

### Number of Senders = n = 2

CSMA Technique	Avg. Delay Per Sender	Avg. Collision Per Sender	Avg. Throughput
One Persistent	23.795	2.50	0.8471
None Persistent	50.73	0.00	1.0000
1/n Persistent	26.86	0.00	1.0000

### Number of Senders = n = 4

CSMA Technique	Avg. Delay Per Sender	Avg. Collision Per Sender	Avg. Throughput
One Persistent	44.807	2.00	0.8825
None Persistent	69.97	1.75	0.8825
1/n Persistent	42.27	2.50	0.8432

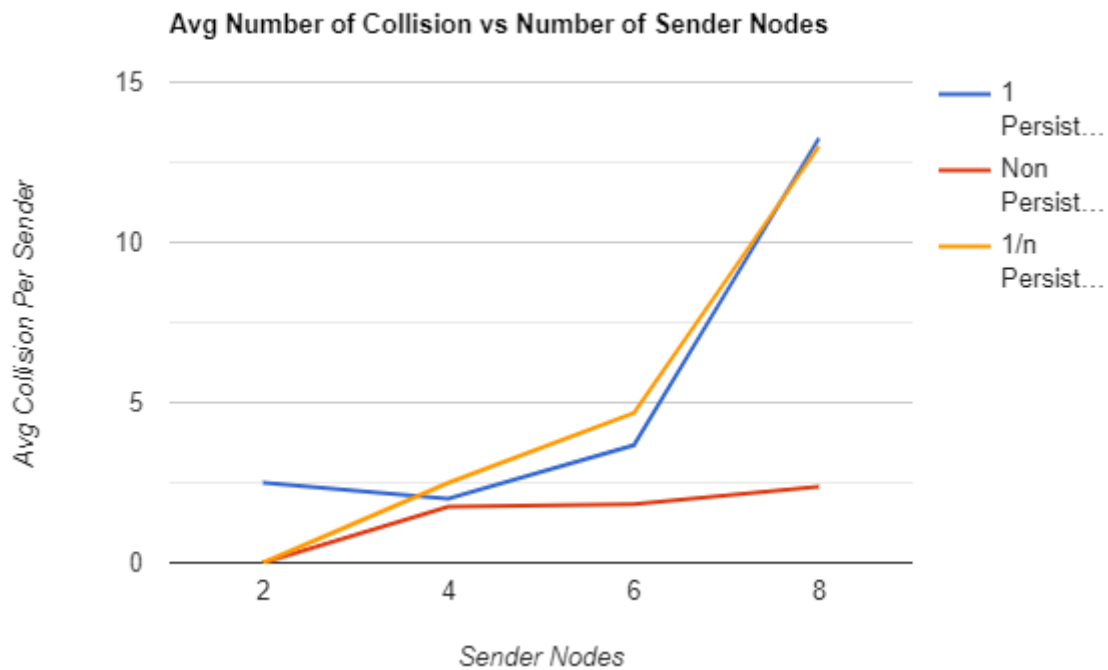
### Number of Senders = n = 6

CSMA Technique	Avg. Delay Per Sender	Avg. Collision Per Sender	Avg. Throughput
One Persistent	61.30	3.67	0.7963
None Persistent	84.06	1.83	0.8800
1/n Persistent	66.93	4.67	0.749

### Number of Senders = n = 8

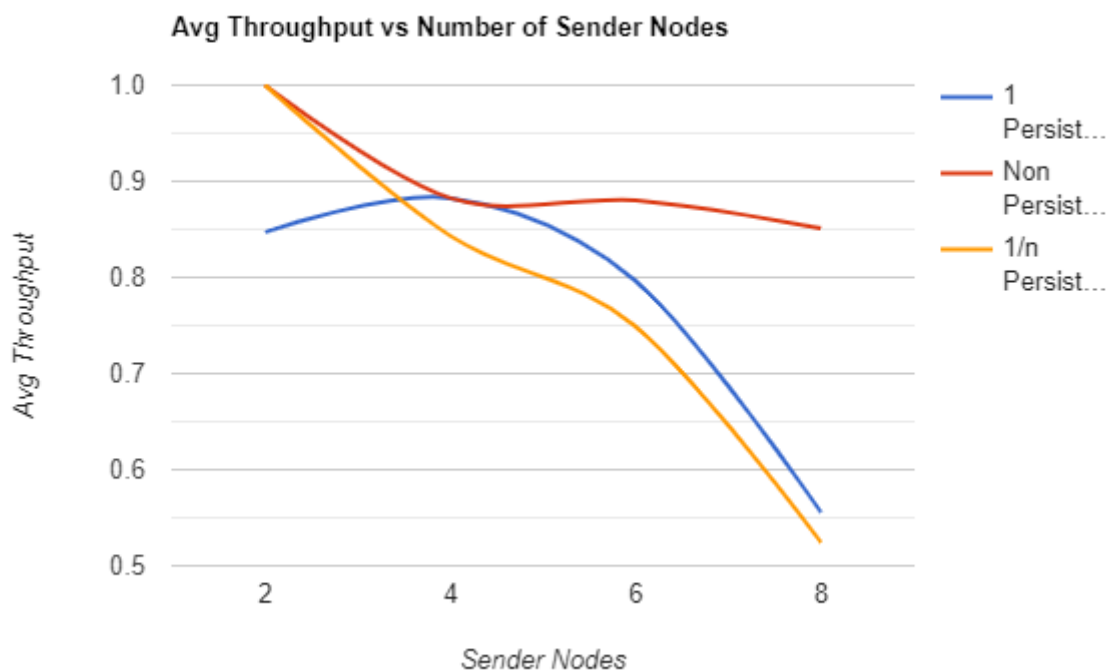
CSMA Technique	Avg. Delay Per Sender	Avg. Collision Per Sender	Avg. Throughput
One Persistent	79.74	13.25	0.5557
None Persistent	115.74	2.37	0.8511
1/n Persistent	81.31	13.00	0.5245

### Comparison of Avg Number of Collision:



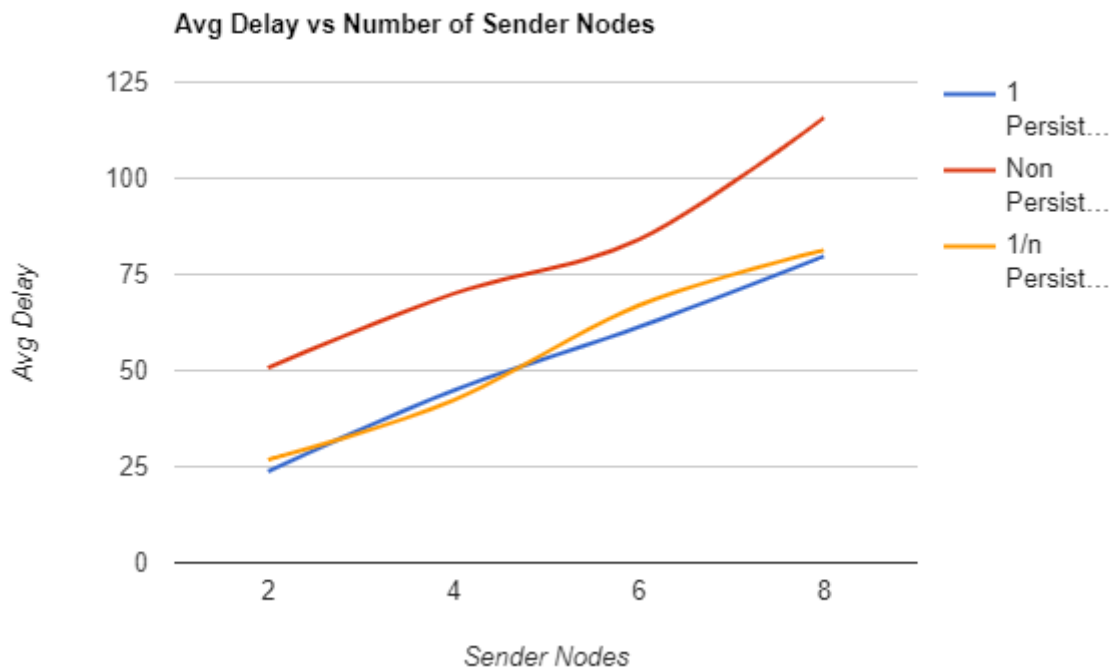
**1/n persistent** method shows the significant number of collisions and with a greater number of senders it just abruptly increases and almost shows **exponential growth**. The **One-persistent** method actually performs best at the medium number of senders and increases with the increasing number of senders but the **1/n persistent** method outperformed it gradually in the long run. The **Non-persistent** graph actually shows greater consistency throughout and managed to end up having the least number of collisions.

### Comparison of Throughputs:



The throughput of a network system is defined as the number of successful transmissions per frame time. The **one persistent** method shows the worst possible throughput and quickly dies to almost zero as the number of senders increases. Interestingly enough **non-persistent** method works pretty well at higher numbers of senders. **1/n persistent** method shows similar performance to **non-persistent** in less number of senders but performs much better than the rest two with the increasing number of senders.

### Comparison of Avg Delay:



The **Non-persistent** method shows the greatest delay of the bunch. For lower numbers of nodes, **One persistent** method is the best. In the lower region **One-persistent** method actually able to beat the **1/n persistent** method, however with increasing sender number it slightly underperforms but in the longer run, the **1/n persistent** method shows the best result and actually is the best of the bunch.

## CONCLUSION

In the Collision comparison and Throughput comparison, The **Non-persistent** method actually outperforms both of the rest two methods but in the case of Delay comparison, it abruptly shows huge delay compared to the other two. When we compare the **One-persistent** and **1/n persistent** methods, we can see that the first one always outperforms the latter one in all of the above scenarios.

## Scopes of Improvement

- In a relatively small network, the number of nodes falls somewhere between 10-30. So I should've considered more data points at a higher number of nodes.
- A more real-life approach would have been taken into consideration, the receiver nodes are also a part of this sensing algorithms. It would've provided more data points.
- I should've considered more data points regarding **non-persistent** CSMA varying the range of random sleep.

## COMMENTS

This assignment has helped me in understanding the different CSMA protocols immensely, by researching and implementing them. It has also helped in understanding the demerits of a protocol, and how such demerits are overcome by other protocols.