Report 2

Controller

1. Send nearby node
information (10.0.0.2 ,
10.0.0.3)

4. send metrics
to controller

T1

2. Send ping and iperf
packets to nearby nodes

Layer 2

Layer 2

10.0.0.2

10.0.0.1

Information about nearby nodes to
other switches

T2

Layer 2

10.0.0.3

3. Delay and jitter
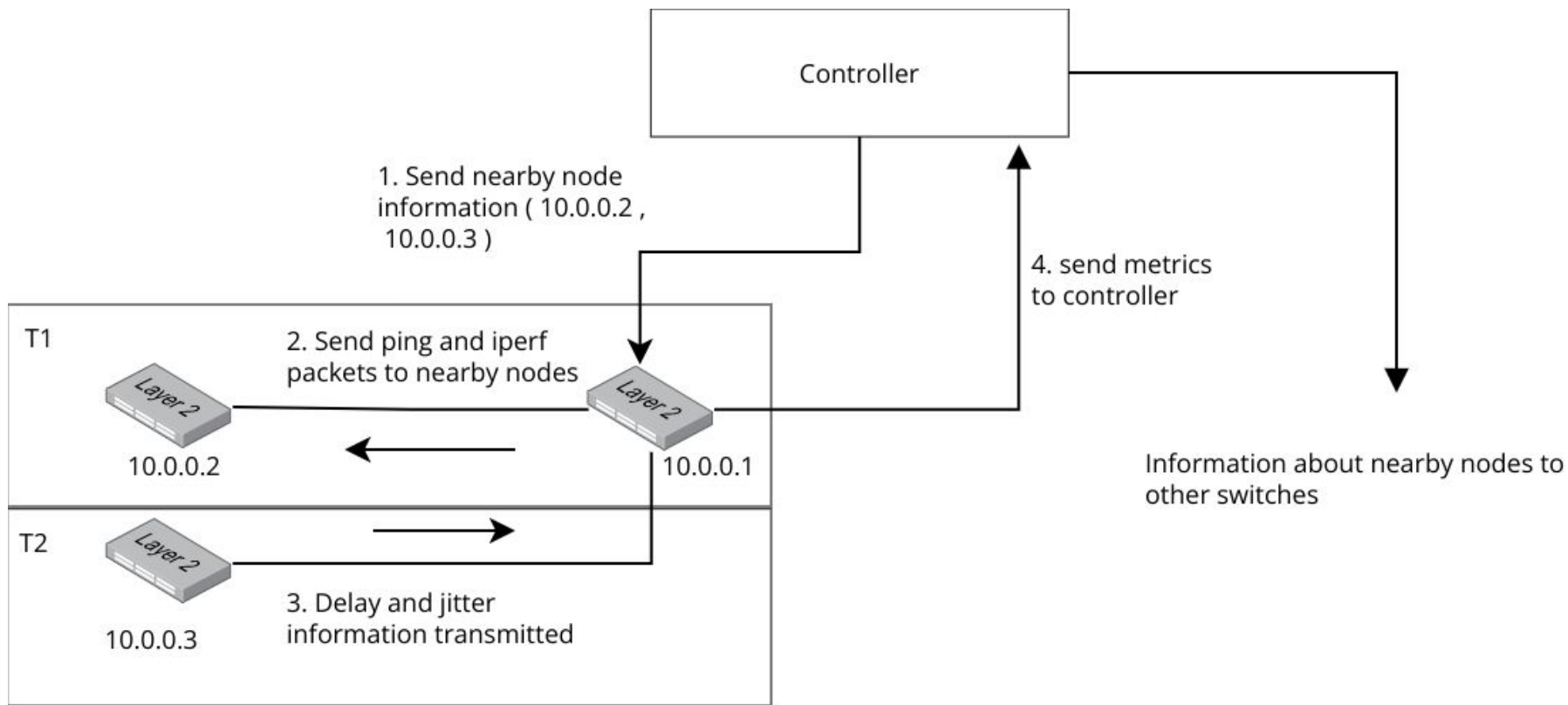information transmitted
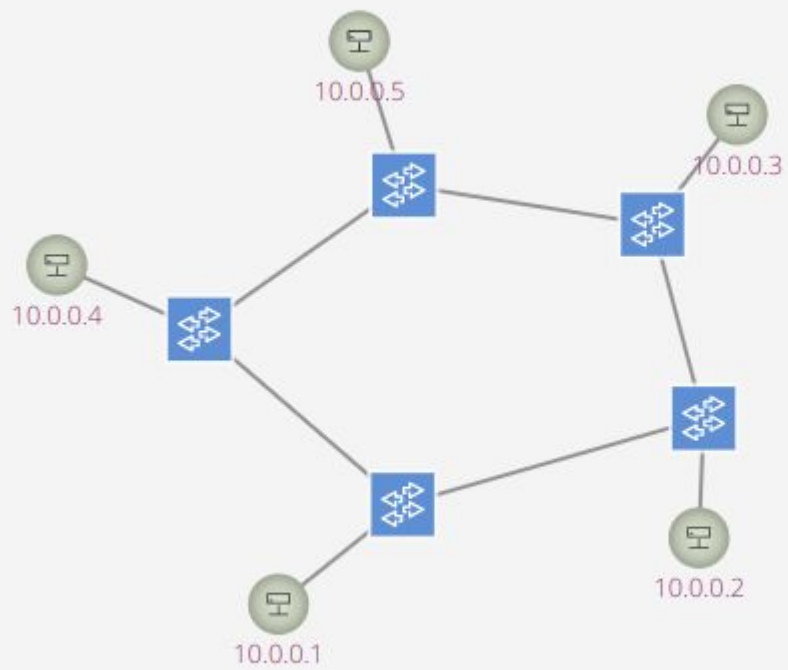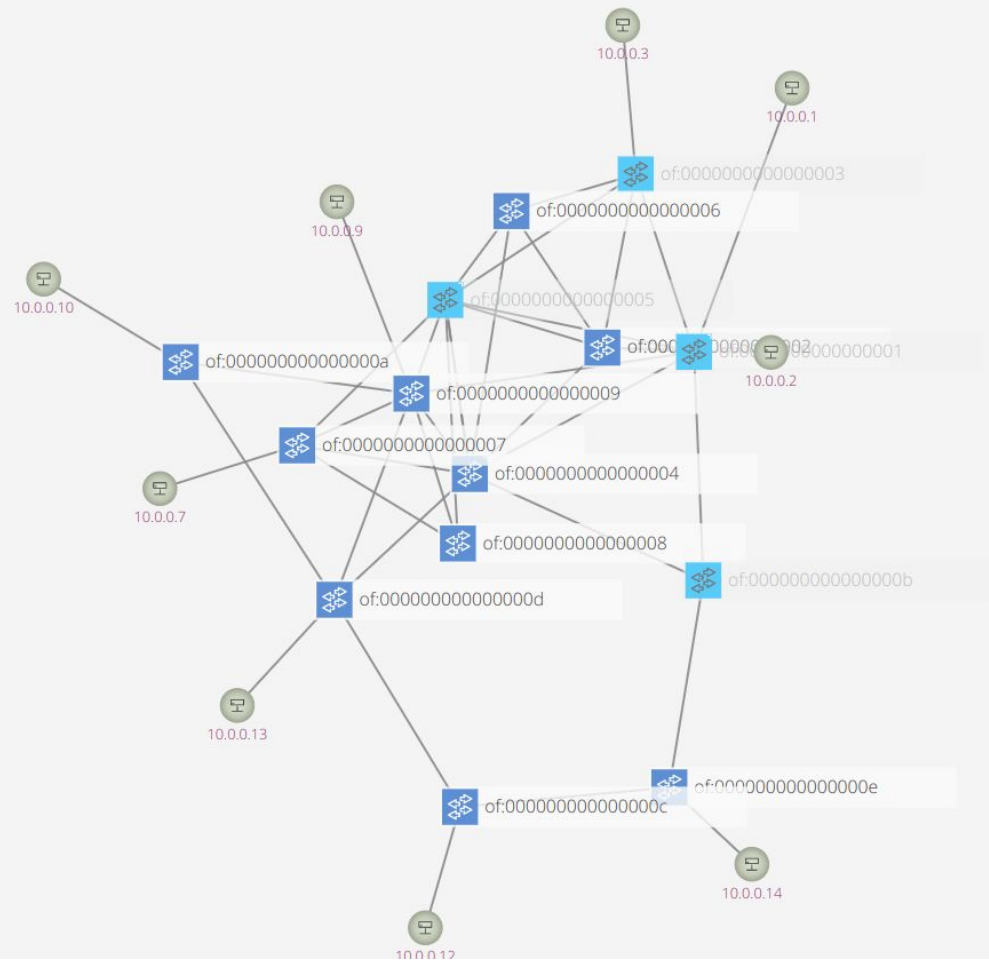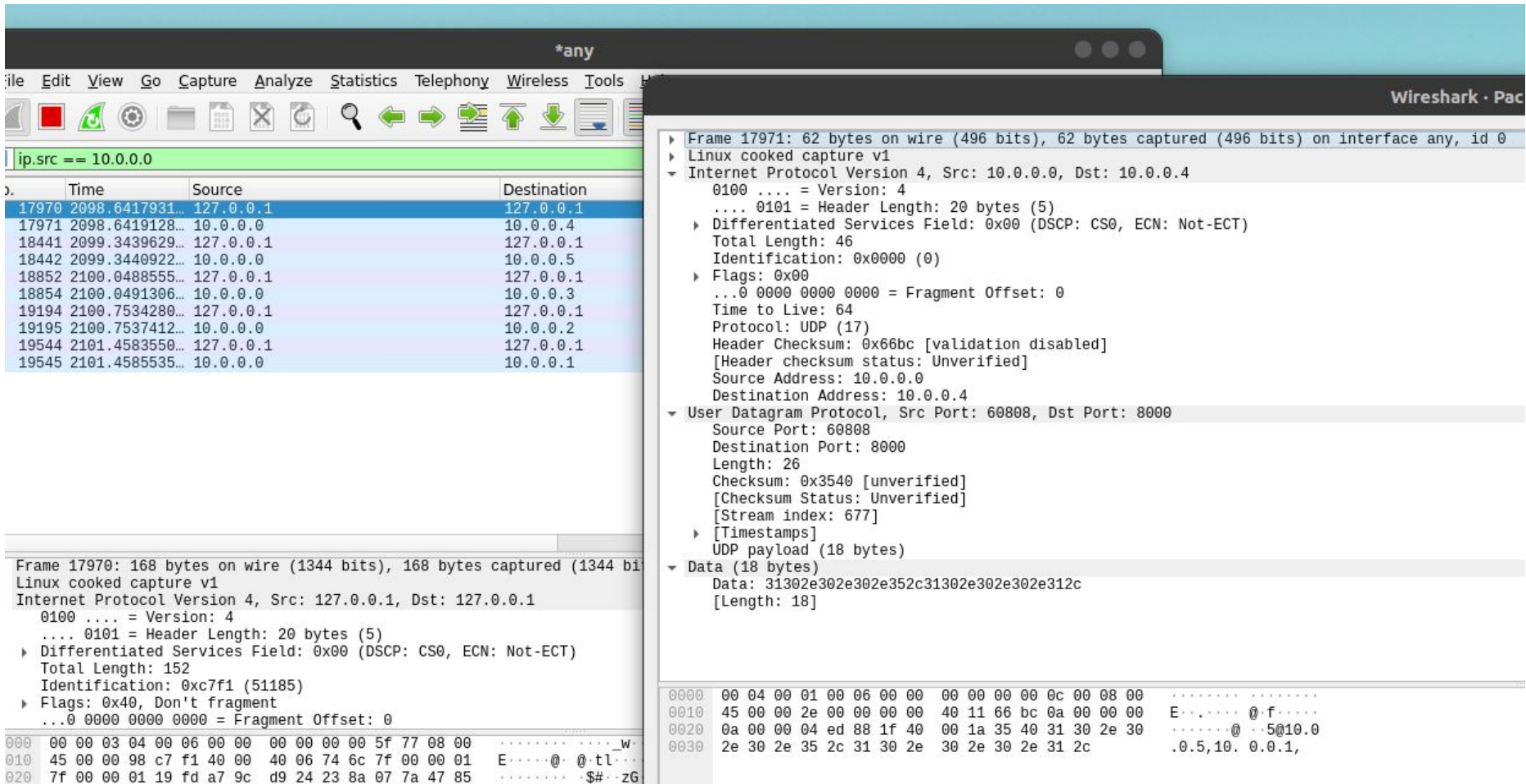
10.0.0.5

10.0.0.3

10.0.0.4

10.0.0.2

10.0.0.1

▼ Frame 178113: 80 bytes on wire (640 bits), 80 bytes captured (640 bits) on interface any, id 0
   ▶ Interface id: 0 (any)
   Encapsulation type: Linux cooked-mode capture v1 (25)
   Arrival Time: Oct 26, 2022 12:55:35.111454752 IST
   [Time shift for this packet: 0.000000000 seconds]
   Epoch Time: 1666769135.111454752 seconds
   [Time delta from previous captured frame: 0.006075838 seconds]
   [Time delta from previous displayed frame: 0.006095177 seconds]
   [Time since reference or first frame: 11.874450665 seconds]
   Frame Number: 178113
   Frame Length: 80 bytes (640 bits)
   Capture Length: 80 bytes (640 bits)
   [Frame is marked: False]
   [Frame is ignored: False]
   [Protocols in frame: sll:ethertype:ip:udp:data]
   [Coloring Rule Name: UDP]
   [Coloring Rule String: udp]
▶ Linux cooked capture v1
▶ Internet Protocol Version 4, Src: 10.0.0.1, Dst: 10.0.0.1
▶ User Datagram Protocol, Src Port: 8000, Dst Port: 8000
▼ Data (36 bytes)
   Data: 31302e302e302e343a31303033302c302c0a31302e302e302e323a31303132302c302c0a
   [Length: 36]

```
0000   00 03 00 01 00 06 ce 13   d3 cd 7c 20 51 5f 08 00    ········ ··| Q_··
0010   45 00 00 40 00 01 00 00   40 11 66 ab 0a 00 00 01    E··@···· @·f·····
0020   0a 00 00 01 1f 40 1f 40   00 2c ad 14 31 30 2e 30    ·····@·@ ·,··10.0
0030   2e 30 2e 34 3a 31 30 30   33 30 2c 30 2c 0a 31 30    .0.4:100 30,0,·10
0040   2e 30 2e 30 2e 32 3a 31   30 31 32 30 2c 30 2c 0a    .0.0.2:1 0120,0,·
```

File Edit View Go

ip.dst == 10.0.0.1

No.	Time
1780…	11.86809305
1781…	11.86834477
1781…	11.86835207
1781…	11.86835294
1781…	11.86835548
1781…	11.87445066
1781…	11.87471594
1781…	11.89221397
1781…	11.89221917
1781…	11.89222006
1781…	11.89222284
1781…	12.08411147
1781…	12.08411780
1781…	12.08411860
1781…	12.08412094
1781…	12.18729810
1781…	12.18730969
1781…	12.18731118
1781…	12.18731514
1782	12.27891500

▶ Frame 178113: 80
▶ Linux cooked capt
▼ Internet Protocol
   0100 .... = Ve
   .... 0101 = Hea
  ▶ Differentiated
   Total Length:
   Identification

```
String getDeviceFromIp(String s){ ⋯

void parsePacketData(String sender , String dt){ ⋯

void getBandwidth(){ ⋯

public void installInitRule(){ ⋯

public String getNearbyNodes(Host hs){ ⋯

public void sendInitPacket(){ ⋯
```

```go
func main() {
    // go iperfServ()
    // get list of host whose details are required
    ar := udpServer()
    fmt.Println(ar)
    for {
        // store ip address : metric arrayS
        mp = make(map[string][]string)

        var wg sync.WaitGroup

        for _, hs := range ar {
            wg.Add(1)
            go func(hs string) {
                pingHost(hs)
                wg.Done()
            }(hs)
        }

        for _, hs := range ar {
            ipefClient(hs)
        }

        wg.Wait()
        udpClient()
        fmt.Println(mp)

        time.Sleep(5 * time.Second)
    }
}
```

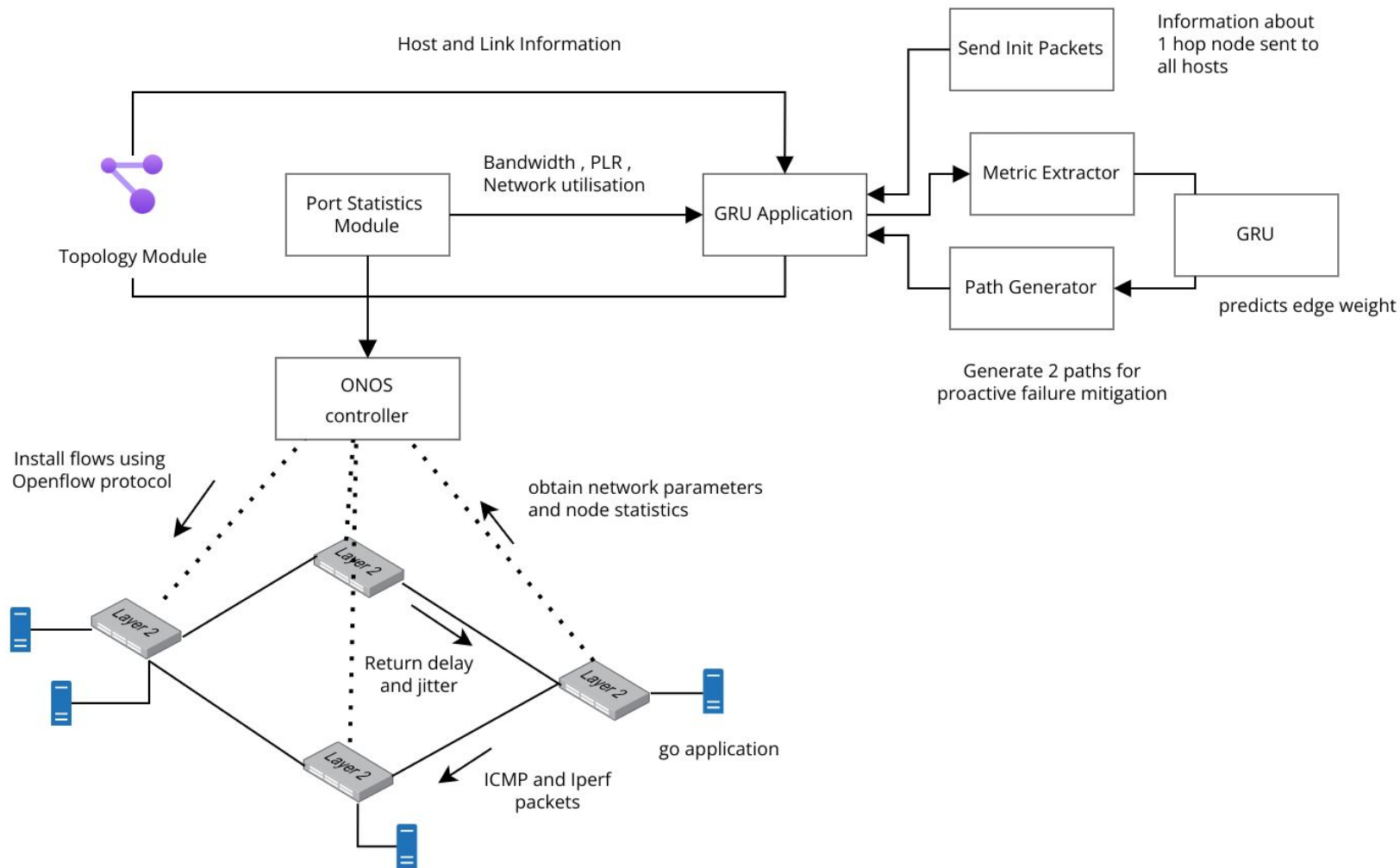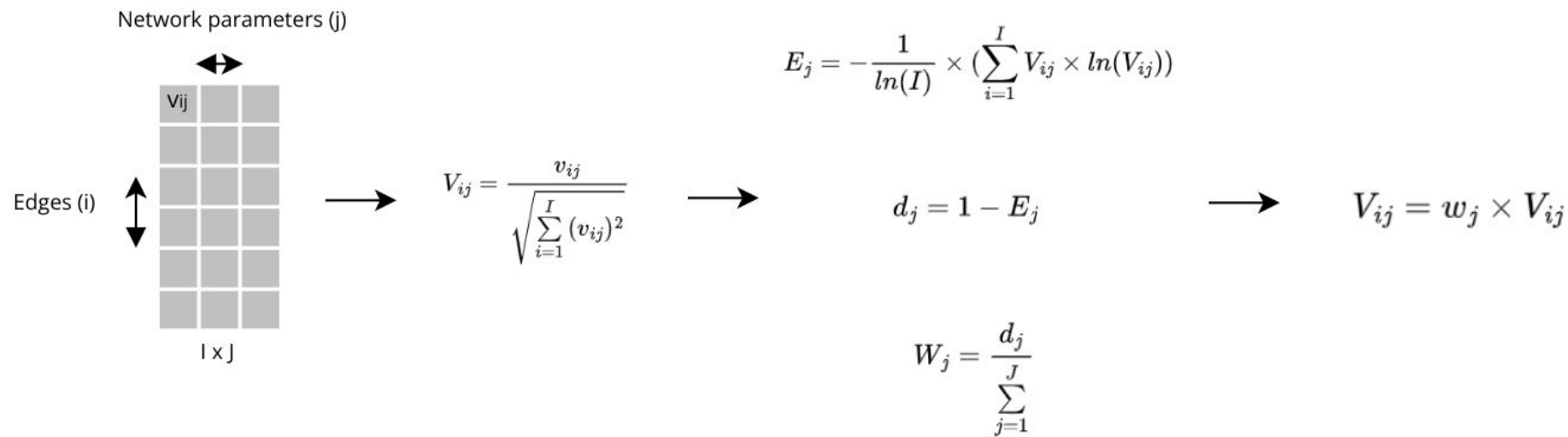Wait for packet from controller

Ping hosts

Iperf client

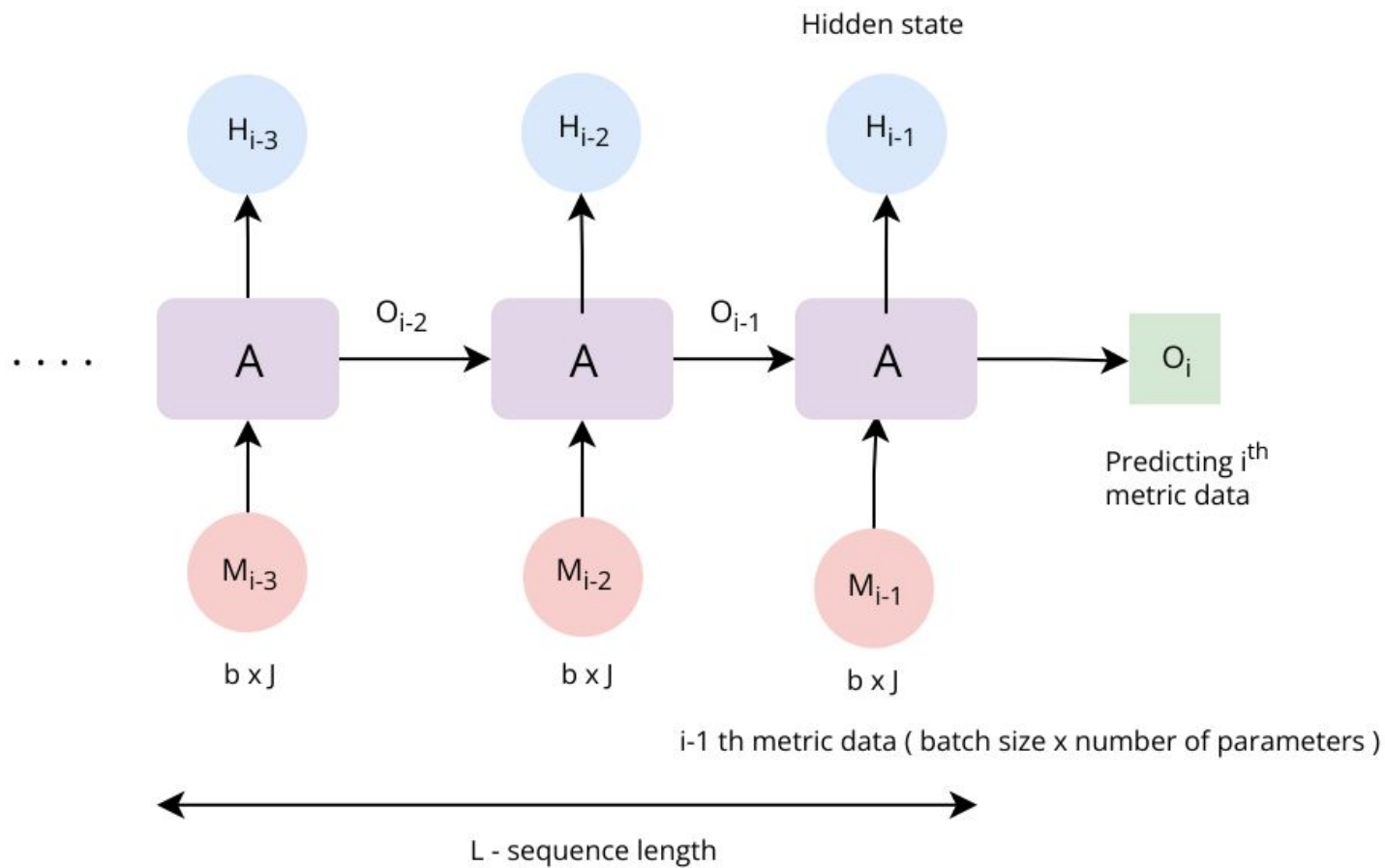Send packet With metrics to controller

```
16 [0.1169060351882137, 0.0, 0.109258559543121]
1 [0.06299599823613589, 0.0, 0.10930140603705947]
2 [0.15203841881990487, 0.0, 0.09910394047970152]
19 [0.10297422788599134, 0.0, 0.10960133149462883]
5 [0.13386649625178876, 0.0, 0.10930140603705947]
22 [0.1169060351882137, 0.0, 0.09918963346757848]
23 [0.09267680509739222, 0.0, 0.10960133149462883]
8 [0.13992380377449412, 0.0, 0.109258559543121]
10 [2.4076232811648334E-9, 0.0, 0.09910394047970152]
14 [1.7427517963662142E-8, 0.0, 0.09918963346757848]
10:19:57.426 INFO  [MetricUpdate] [323.77622909460774, 0.0, 2546.8721993849
10:19:57.427 INFO  [MetricUpdate] [-4.9289138539893514E-32, -0.0, -4.76965
10:19:57.427 INFO  [MetricUpdate] Network Metrics :
16 [1.2035682139286071E-4, 0.0, 0.12564431393629216]
1 [0.13074873587394018, 0.0, 0.15312900760985607]
2 [0.17192944008620478, 0.0, 0.12734574735417944]
19 [1.0601378050148352E-4, 0.0, 0.007198372152600072]
5 [0.20384448585070988, 0.0, 0.15312900760985607]
22 [1.2035682139286071E-4, 0.0, 0.007198372152600072]
23 [9.541240245133518E-5, 0.0, 0.007198372152600072]
8 [0.15133908798007248, 0.0, 0.12564431393629216]
10 [2.4786905549052757E-12, 0.0, 0.12734574735417944]
14 [1.7941936560387475E-11, 0.0, 0.007198372152600072]
10:20:10.754 INFO  [MetricUpdate] [368.4440255996144, 0.0, 2746.6343040164(
10:20:10.755 INFO  [MetricUpdate] [-3.477785295422988E-49, -0.0, -2.846912(
10:20:10.755 INFO  [MetricUpdate] Network Metrics :
16 [1.0888746694078516E-7, 0.0, 0.1248801358803733]
```

Network parameters (j)

↔

Vij

Edges (i) ↕

I x J

→

$$V_{ij} = \frac{v_{ij}}{\sqrt{\sum\limits_{i=1}^{I} (v_{ij})^2}}$$

→

$$E_j = -\frac{1}{ln(I)} \times \left(\sum_{i=1}^{I} V_{ij} \times ln(V_{ij})\right)$$

$$d_j = 1 - E_j$$

$$W_j = \frac{d_j}{\sum\limits_{j=1}^{J}}$$

→

$$V_{ij} = w_j \times V_{ij}$$

Hidden state

H_{i-3} H_{i-2} H_{i-1}

O_{i-2} O_{i-1}

A   A   A O_i

. . . .

Predicting i^{th} metric data

M_{i-3} M_{i-2} M_{i-1}

b x J   b x J   b x J

i-1 th metric data (batch size x number of parameters)

L - sequence length

```python
class RNNModel(pl.LightningModule):
    def __init__(self ):
        super().__init__()


        num_features = 3
        self.rnn = nn.GRU(num_features, cfg['rnnh'] , cfg['rnnl'] , ba
tch_first=True)
        self.dropout = nn.Dropout(cfg['drp'])
        self.fc = nn.Linear(cfg['rnnh'], num_features )


        self.lr = cfg['lr']
        self.ls = nn.MSELoss()


    def loss( self, x , y ):
        return self.ls(x, y)


    def forward(self, x ):
        out,x = self.rnn(x)
        out = out[:,-1,:]
        out = self.fc( out )
        return out


    def predict_step(self, batch, batch_idx: int , dataloader_idx: int
```