

Networked Wireless Systems Assignment 2:

Akilesh B, CS13B1042

PART A: PDR vs Error Rate for Rate Error Model by setting ErrorUnit [Bit] (npackets = 100)

Changes in the code:

```
Ptr<RateErrorModel> em = CreateObject<RateErrorModel> ();

Ptr<UniformRandomVariable> v = CreateObject<UniformRandomVariable> ();
v->SetAttribute ("Min", DoubleValue (0.0));
v->SetAttribute ("Max", DoubleValue (1.0));

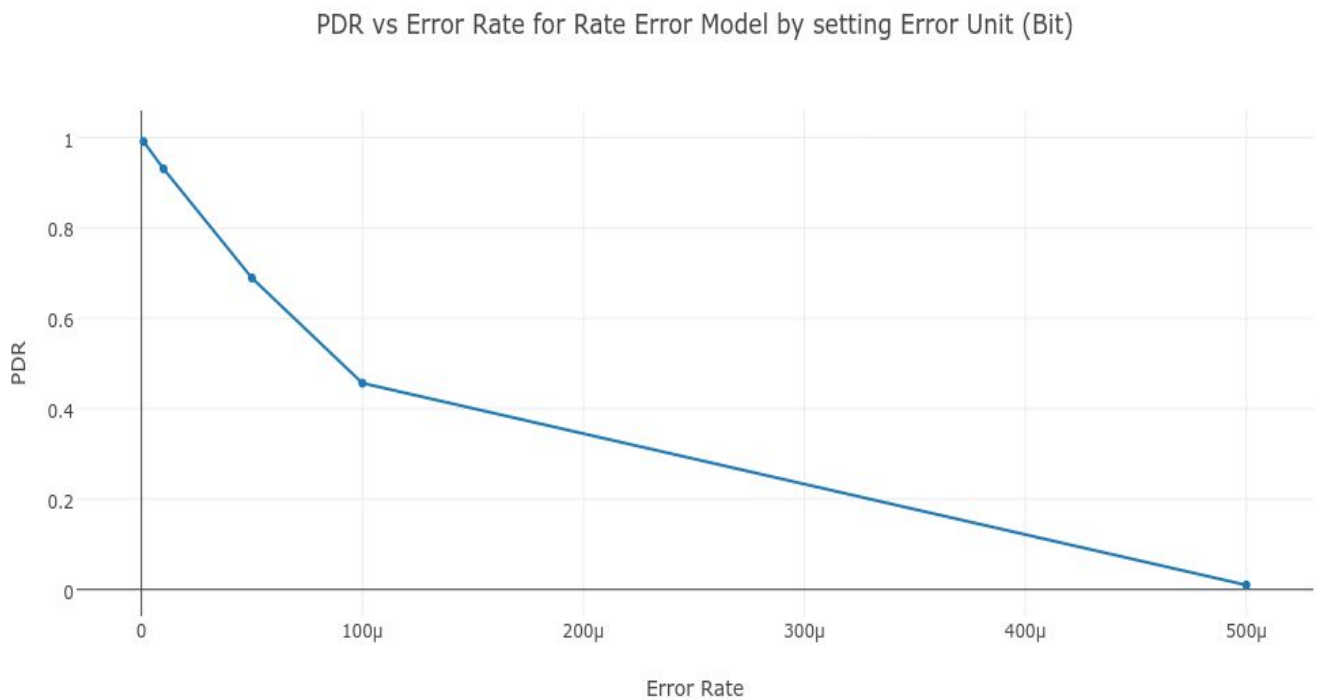
em->SetRate(0.1);           //set error rate
em->SetUnit((RateErrorModel::ErrorUnit)0);           //enumeration - set error unit
em->SetRandomVariable(v);           //set random variable
```

Readings:

RngRun/Error Rate	ER1=0.000001	ER2=0.00001	ER3=0.00005	ER4=0.0001	ER5=0.0005
RngRun (42) =	194/198	173/190	117/167	79/153	0/100
RngRun (43) =	193/198	174/191	122/169	72/144	2/102
RngRun (44) =	199/200	187/196	122/173	68/147	2/102
RngRun (45) =	194/198	178/192	109/167	72/147	0/100
RngRun (46) =	200/200	183/195	110/166	65/146	0/100
RngRun (47) =	200/200	185/196	128/174	65/143	2/102
RngRun (48) =	200/200	185/194	130/177	76/150	2/102

RngRun (49) =	199/200	183/195	115/172	60/144	0/100
RngRun (50) =	196/199	175/191	110/168	43/132	1/101
RngRun (51) =	200/200	174/190	107/163	65/145	1/101
Avg PDR	0.909268057	0.930911476 2	0.689294355 3	0.45659	0.009823335 3

Graph



Conclusion:

- 1) PDR decreases exponentially with increase in error rate. The loss depends on the distribution of the random variable v .

- 2) This is mainly because as the error rate increases, the loss in the number of packets which the server receives from the client and number of ACKs which the server sends back increases drastically.

**PART B: PDR vs Error Rate for Rate Error Model by setting ErrorUnit [Byte]
(npackets = 100)**

Changes in code:

em->SetUnit((RateErrorModel::ErrorUnit)1); //Just change this and keep rest same

Readings:

RngRun/Err or Rate	ER1=0.0000 5	ER2=0.0001	ER3=0.0003	ER4=0.0005	ER5=0.001
RngRun (42) =	179/192	168/190	136/175	96/158	61/145
RngRun (43) =	179/194	170/188	140/177	100/162	62/142
RngRun (44) =	189/196	181/193	133/175	106/165	57/142
RngRun (45) =	187/196	175/192	127/175	91/156	58/139
RngRun (46) =	192/197	179/193	122/169	104/165	52/139
RngRun (47) =	189/196	184/196	139/176	105/164	52/138
RngRun (48) =	186/194	183/194	144/182	117/173	54/140
RngRun (49) =	187/195	178/193	128/174	99/165	45/132
RngRun (50) =	185/195	162/183	128/173	87/156	30/124
RngRun (51)	187/194	169/188	134/174	94/156	54/140

=					
Avg PDR	0.954261715	0.915374549 7	0.760232412 4	0.615774038 4	0.378116947 1

Graph:



Conclusion:

- 1) Here also the decrease in PDR with increase in error rate is exponential. This is because error unit in byte is similar to that in bit (as 1 byte = 8bits).
- 2) The loss in the number of packets which the client sends to the server and the ACKs sent back by the server is high. The loss depends on the distribution of random variable v .

**PART C: PDR vs Error Rate for Rate Error Model by setting ErrorUnit [Packet]
(npackets = 100)**

Changes in the code:

em->SetUnit((RateErrorModel::ErrorUnit)2); //Just change this and keep rest same

Readings

RngRun/Err or Rate	ER1 = 0.2	ER2 = 0.3	ER3 = 0.4	ER4 = 0.6	ER5 = 0.8
RngRun (42) =	144/178	130/173	100/160	72/153	32/129
RngRun (43) =	151/181	134/175	103/164	66/142	33/127
RngRun (44) =	151/182	130/175	109/165	66/147	33/127
RngRun (45) =	147/182	121/173	97/160	64/142	25/121
RngRun (46) =	135/173	119/169	106/165	61/143	20/117
RngRun (47) =	156/184	134/176	106/164	62/143	23/121
RngRun (48) =	163/188	137/179	122/174	62/142	28/123
RngRun (49) =	154/185	122/172	102/165	54/140	23/119
RngRun (50) =	137/176	124/171	90/158	38/130	21/115
RngRun (51) =	145/178	121/170	95/156	59/141	27/123
Avg PDR	0.820124799 4	0.733652048 8	0.630659640 3	0.422828058	0.215842191 9

PART D: PDR vs Error Rate for Burst Error Model (npackets = 100)

Changes in the code:

```
Ptr<UniformRandomVariable> v1 = CreateObject<UniformRandomVariable> ();  
v1->SetAttribute ("Min", DoubleValue (1.0));  
v1->SetAttribute ("Max", DoubleValue (5.0));
```

```
Ptr<BurstErrorModel> em = CreateObject<BurstErrorModel> ();  
em->SetRandomBurstSize(v1);  
em->SetRandomVariable(v);  
em->SetBurstRate(0.4);
```

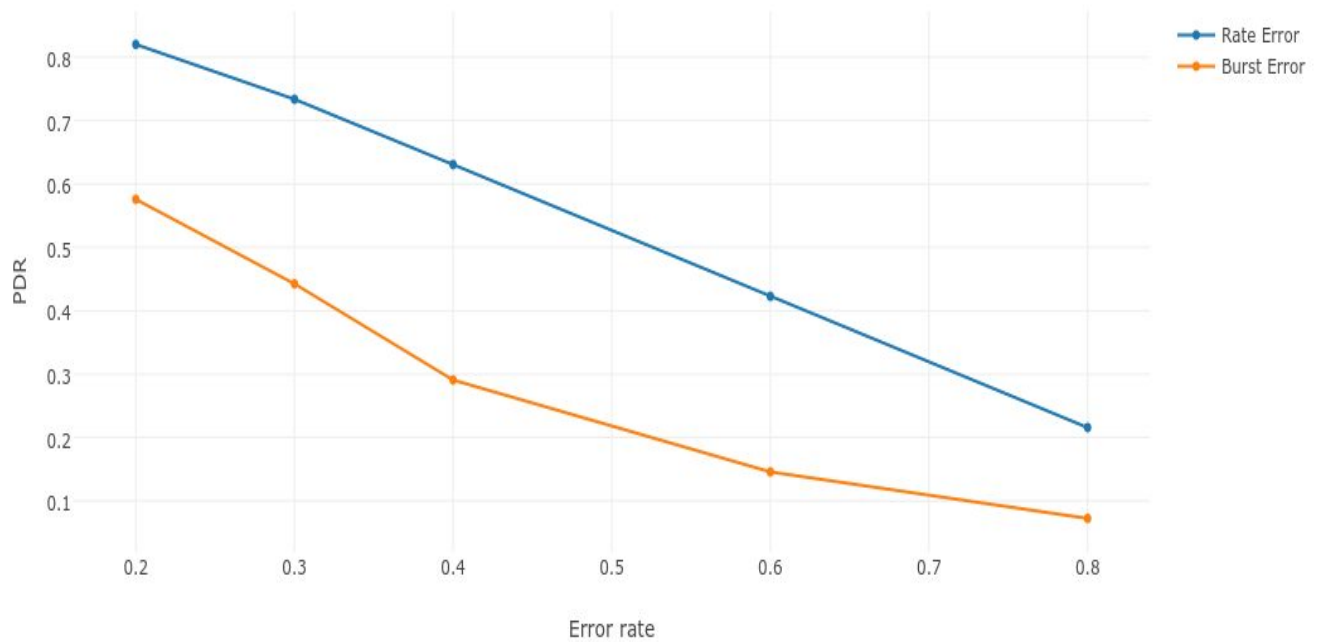
Readings:

RngRun/Err or Rate	ER1 = 0.2	ER2 = 0.3	ER3 = 0.4	ER4 = 0.6	ER5 = 0.8
RngRun (42) =	87/150	69/139	47/126	27/119	13/117
RngRun (43) =	108/159	83/146	26/116	20/114	12/111
RngRun (44) =	80/145	62/135	41/127	6/105	6/104
RngRun (45) =	77/142	44/126	30/119	23/114	3/102
RngRun (46) =	87/148	40/125	29/120	13/112	3/103
RngRun (47) =	88/149	62/135	25/113	10/108	9/108
RngRun (48) =	91/150	68/138	48/130	21/114	10/108
RngRun (49) =	75/143	51/130	39/124	14/110	9/105
RngRun (50)	70/142	61/135	36/122	13/110	6/105

=					
RngRun (51) =	88/146	58/133	36/122	18/113	7/107
Avg PDR	0.575850447	0.442562428 9	0.290890572 7	0.145884772 4	0.725659141

Graph

PDR vs ErrorRate for Rate Error Model by setting ErrorUnit (Packet) and that for Burst Error Model



Conclusion:

- 1) The decrease in PDR with increase in error rate when error unit is set in packets is linear. This is because in this case when we set random variable v , it refers to the number of packets which are dropped.
- 2) Burst Error Model is similar to Rate Error model except that a contiguous sequence of packets are lost (packets are lost in a burst). Since in this case, more number of packets are lost, there is a relative dip in the average PDR value when compared to the Rate Error Model. Burst size is a random variable from 1 to 5.
- 3) In both cases the decrease is almost linear.

OPTIONAL

- 1) When the data rate and the UDP Echo Client packet interval are changed, there is a dip in the average PDR value (although the difference is not that significant). This is because we are decreasing the packet interval from 1s to 0.1s, more number of packets are sent into the network. This leads to more congestion and number of packets received decreases.
- 2) When we disable error model, there is no packet loss, the PDR is 1.

ANNEXURE

- 1) I have included the modified first.cc and the script I used to obtain the readings in this folder.