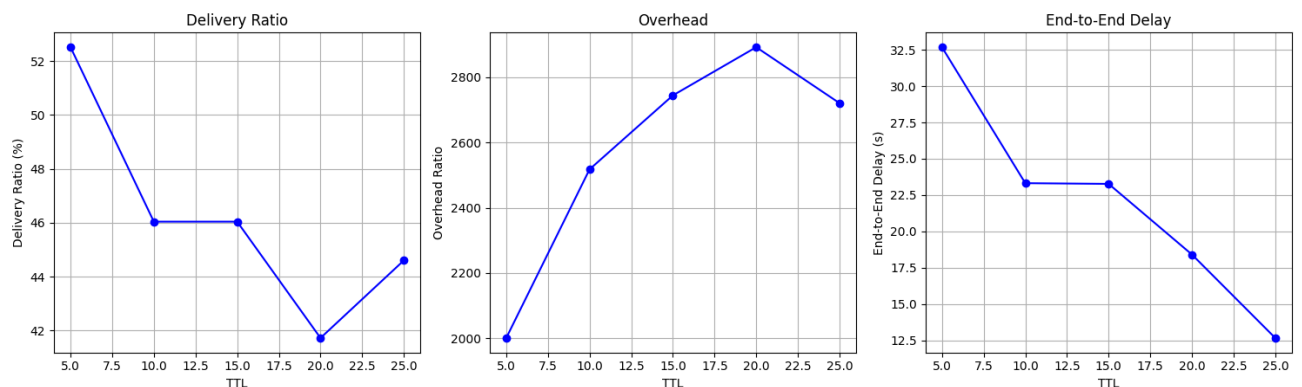


## 1. TTL

data\ttl	5	10	15	20	25
Packets sent	139	139	139	139	139
Packets delivered	73	64	64	58	62
Delivery percentage	52.518%	46.0432%	46.0432%	41.7266%	44.6043%
Total BytesSent	1.49662e+08	1.65065e+08	1.79908e+08	1.71809e+08	1.72773e+08
Total BytesReceived	5.28681e+07	5.64234e+07	5.64316e+07	5.74853e+07	5.84458e+07
Total PacketsSent	146154	161196	175691	167782	168724
Total PacketsReceived	408896	449510	468936	445517	465696
Average End-to-End Delay	32.683	23.3195	23.2717	18.406	12.66

TTL Variation Analysis



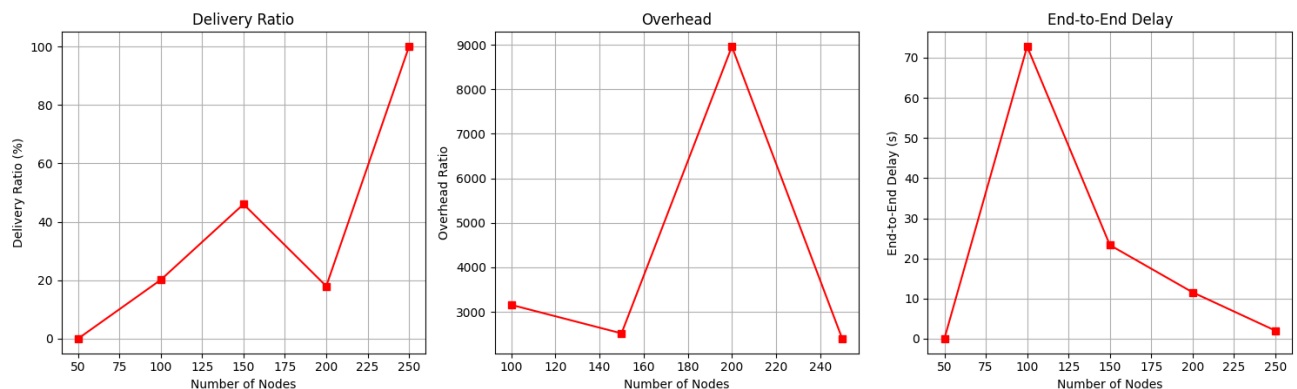
TTL(封包存活時間)的影響：

- 傳遞率(Delivery Ratio):
  - TTL從5增加到25時，傳遞率反而下降，從52.5%降至44.6%
  - 這表示較長的TTL並不一定能提高傳遞成功率
  - 最佳TTL值似乎在5左右，過長反而會降低效率
- 網路負載(Overhead):
  - 隨著TTL增加，總封包傳送量(Total PacketsSent)也隨之增加
  - TTL=5時傳送146154個封包，TTL=25時增加到168724個
  - 較長的TTL會導致更多冗餘傳輸，增加網路負載
- 延遲時間(End-to-End Delay):
  - TTL越長，延遲時間反而越短
  - 從TTL=5時的32.68秒降到TTL=25時的12.66秒
  - 這可能是因為較長的TTL允許封包尋找更多可能的路徑

## 2. Number of Nodes

dataTtl	50	100	150	200	250
Packets sent	139	139	139	139	139
Packets delivered	0	28	64	25	139
Delivery percentage	52.518%	20.1439%	46.0432%	17.9856%	100%
Total BytesSent	5.58418e+07	9.05902e+07	1.65065e+08	2.29394e+08	3.41448e+08
Total BytesReceived	646144	1.10008e+07	5.64234e+07	1.12602e+08	1.79798e+08
Total PacketsSent	54533	88467	161196	224018	333445
Total PacketsReceived	59490	211902	449510	768085	1183385
Average End-to-End Delay	-nan	72.7906	23.3195	11.5502	2.00868

Number of Nodes Variation Analysis



Number of Nodes(節點數量)的影響：

- 傳遞率：
  - 節點數從50增加到250時，傳遞率大幅提升
  - 50個節點時傳遞率為0%，250個節點時達到100%
  - 這說明較高的節點密度有助於提高傳遞成功率
- 網路負載：
  - 節點數增加會導致總封包量顯著增加
  - 從50節點的54533個封包增加到250節點的333445個
  - 更多節點會產生更多轉發流量
- 延遲時間：
  - 節點數增加時，延遲時間反而減少
  - 從100節點的72.79秒降到250節點的2.01秒
  - 這可能是因為較多節點提供了更多可能的傳輸路徑

總結/問題與討論：

TTL需要適中，過長或過短都不理想。節點密度越高，網路效能越好，但同時也會增加網路負載。在實際應用中需要根據具體場景權衡這些參數。

另外可以使用 `sudo chmod 777 -R <資料夾路徑>` 來避免權限問題