Chapter 6

Fault Tolerance

6.1 Introduction

In MANET nodes cooperate among themselves for transmission of packets. Nodes join and leave the network. The designs of these networks are based on following assumptions.

- Non adversarial environment: Here, nodes cooperate with each other and are well behaved i.e. send and receive packets as required.
- Adversarial environment: Here, nodes don't cooperate with each other either due to maliciousness or selfishness as discussed below.
    a. Maliciousness: It is defined as compromising the security of network or other nodes e.g. passing information to unwanted receivers (possibly an intruder).
    b. Selfishness: It is defined as maximizing owns benefit e.g. denying packet sending and receiving to preserve battery power.

When a node fails, if the remaining nodes are able to take up the load, then the network is said to be fault tolerant. Peer to peer systems (P2P) exhibits fault tolerance and the same is expected from MANET (being a P2P network).

Single path routing protocols learn routes and select the best route to destination, while multi path routing protocols learn routes and can select more than one path to destination. Hence they are more capable of load balancing, which eventually leads to fault tolerance.

Fault tolerance is of four types:

- Fault tolerance in Node Failures.
- Fault tolerance in Link failure and Network Failure.
- Fault tolerance in Transmission Power and Energy.
- Fault tolerance using check-pointing, message logging, reducing overload etc.

In this chapter, we have studied the level of fault tolerance offered by single path and multi path routing protocols. We have considered AODV [69], DSR [70] and E2FT [74] routing protocols for this purpose. Using NS2 simulator, we have observed the impact of changing the pause time and faultiness percentage of the node. To assess the performance packet delivery percentage has been plotted.

Organization of the rest of chapter is as follows. In section 6.2, state of the art is discussed followed by a description of the simulation setup in section 6.3. Results are given in section 6.4. Chapter is concluded in section 6.5.

6.2 State of the art

A routing algorithm was proposed by Xu et.al [36] that combines the power of position-based routing and fault-tolerant routing. By using combination of these two concepts, the authors achieved a simplified way of localizing routing overhead while at the same time improved the operational effectiveness of the position-based routing approaches by alleviating some of the drawbacks associated with them, such as routing deadlock occurrences, and therefore creating a robust and fault tolerant routing strategy. But, this algorithm suffers from the problem of deadlock.

Yuan Xue et.al [37] proposed a new routing service named best-effort fault-tolerant routing (BFTR). The design goal was to provide packet routing service with high delivery ratio and low overhead in presence of misbehaving nodes. It evaluates the routing feasibility of a path by its end-to-end performance. By continuously observing the routing performance, it dynamically routes packets via the most feasible path.

A protocol for reliable multicast within a group of mobile hosts that communicate with a wired infrastructure by means of wireless technology was proposed by Anastasi et.al [38]. It tolerates failures in the wired infrastructure. The wireless coverage may be incomplete and message losses could occur even within cells, due to physical obstructions or to the high error rate of the wireless technology. The authors concluded that the increase in the average latency experienced by messages is limited to few milliseconds.

Melamed et.al [39] proposed a position-based fault-tolerant protocol: Octopus. Here, the node location is updated using flexible state. Fault-tolerance is achieved by employing redundancy. It is able to achieve low overhead for updating location. But, it is applicable only for fixed node density and not others.

A Stochastic Learning Weak Estimator (SLWE), to estimate the parameters of a binomial distribution, where the convergence of the estimate is weak was introduced by Oommen et.al [40]. The estimation was based on the principles of stochastic learning. Authors concluded that SLWE is superior in pattern-recognition-based data compression, where the underlying data distribution is non-stationary..

Permission-based message efficient mutual exclusion (MUTEX) algorithm [41] was used to identify and resolve link or host failure using the time out based method. To reduce messages cost, the algorithm uses the “look-ahead” technique, which enforces MUTEX only among the hosts currently competing for the critical section. The proposed algorithm was able to tolerate link or host failures, using timeout-based mechanisms.

Qin et.al [42] proposed a cluster head based fault tolerant algorithm (FTCH). This algorithm provides guarantee for higher packet delivery fraction and lower routing overhead, in case of a fault. FTCH was compared with MMHH, AODV and DSR and was observed to perform better than others.

Riganelli et.al [43] investigated the transmission-power assignment problem for k-connected Mobile Ad hoc NETworks (MANETs), the problem of optimizing the lifetime of a MANET at a given degree k of connectivity by minimizing power consumption. The proposed algorithm was fully distributed and used predictive control to optimize transmission power and life time. .

In [44], the authors introduce a matrix multiplication algorithm based on the checksum of the result to provide fault tolerance. They showed that fail-stop process failures in ScaLAPACK matrix-matrix multiplication kernel can be tolerated without check pointing or message logging. It maintains the checksum information in the middle of the computation. From this information the fail-stop process is computed.

FTRA (Fault Tolerant Routing Algorithm) based on location of nodes was proposed by Zhou et.al [45]. Here the network is divided into grids according to geographical location information. In case of a fault, unused alternate route is used based on the location of neighbouring grid(s).

Gong et.al [46] investigated the communication failure in grid based, distributed genetic algorithms with various topologies. The authors evaluated the performance behavior of distributed GAs under varying levels of persistent communication failures, using the sorting network problem as a benchmark. To resolve the issue of link failure, retry and reroute approach was used.

In [47] Khazaei et.al proposed creation of backup path between source and destination to increase the data transfer and fault tolerance. The backup path was created during route reply, route maintenance and local recovery. The protocol performance was demonstrated by using GloMoSim. The experimental results show that this protocol can decrease the packet loss ratio rather than DSR and SMR and it is useful for the applications that need a high level of reliability.

An algorithm for topology control and to automatically reduce the number of link was introduced by Moraes et.al [48]. It reduces the power consumption. The problem was optimized to four variants namely

a. Symmetric topology for input graph.
b. Asymmetric topology for input graph.
c. Unidirectional result graph.
d. Bidirectional result graph.

Fault-tolerant distributed topology control algorithm [49] maintains a k-regular and k-node connected network, with energy efficient multi-hop communication. It also builds a hierarchy of clusters that reflects the node density in the network, with guaranteed and localized fault-tolerant communication between any pair of cluster members. Here a hierarchy of clusters is built, which is based on node density.

Shaji et.al [50] proposed a Self-eliminating Fault-tolerant based Un-interrupted reliable Service switching mobile Protocol (SFUSP). Here clustering and self-elimination is used to create a reliable route and early identify the link break.

In [51], Tuli et.al proposed a minimum process checkpoint scheme which is based on Cluster Based Routing Protocol (CBRP). Minimum number of nodes in a cluster is used as a checkpoint. It produces a consistent set of checkpoints and the algorithm makes sure that only minimum number of nodes in the cluster is required to take checkpoints.

A Modified Cluster-based QOS routing algorithm was proposed by Llewellyn et.al [52]. It evaluated node failure based on failure recovery time, throughput, dropped packet(s) and bandwidth.

Chandrasekaran et.al [53] proposed a Trusted Fault Tolerant model (TFT) to be used in Location Aided Routing (LAR) protocol. The model assumes that the node is either a selfish or a misbehaving one. Based on the recovery of lost packet, location awareness and node trust level is improved.

Singh et.al [54] proposed a novel algorithm using NxN matrix to represent the cost between the participating nodes, and uses K-FT topology to tackle the fault tolerant problem of Mobile Adhoc Networks. The topology handles the fault identification between the participating nodes based on cost. It achieves optimal resource utilization and fairness.

DSR protocol was extended for supporting fault tolerance by Rana et.al [55]. The proposed protocol tries to find two routing paths (if they exist) from the source to the destination node. During the route discovery process, the protocol identifies several new paths that are not able to be detected by the application of the basic DSR. The proposed protocol offers low overhead over the basic DSR, in terms of the number and sizes of control messages sizes.

One span fault tolerant graph was introduced by Rouzi et.al [56], which was used to protect minimum energy paths. This paper proved that the topology constructed by *k*-FT1S is a *k*-fault-tolerant 1-spanner that can tolerate up to *k* node failures, such that the remaining network after node failures preserves all the minimum energy paths of the remaining network gained from the initial network by removing the same failed nodes.

A Spiral Millipede-inspired Routing Algorithm (SMiRA) was proposed by Adeluyi et.al [57]. To decrease the overhead and improve fault tolerance, a bio-inspired approach was followed. It was compared to AODV. The results showed an improvement in the monitored metrics, with a limited penalty.

Although, a considerable number of fault tolerant routing protocols have been proposed, we have considered the load sharing abilities of this multi path routing protocol as compared to two single path routing protocols namely AODV and DSR.

6.3 Simulations

Simulations were carried on NS-2, which is a discrete event driven simulation. The simulated environment that is considered consists of a flat square of length 500 meters. Performance metric considered was Percentage of packets delivered.

Percentage of packets delivered: It represents the rate at which packets are successfully delivered to the destination.

Percentage delivered packets = Total number of delivered packets/ Total number of sent packets

The variation was done in pause time 1, 25, 50 and 100s. The faultiness of nodes was varied as 25, 50, 75 and 95 %. Along with this, the node density was varied as 10, 25, 50 and 100. The nodes have a data delivery probability which decreases as they move away from the center of the region, and increases as they move closer to it. This may happen due to the fact that signal strength diminishes when the nodes move away from each other. The four algorithms that were simulated are DSR, AODV and E2FT. The simulation parameters are listed in table 7.1.

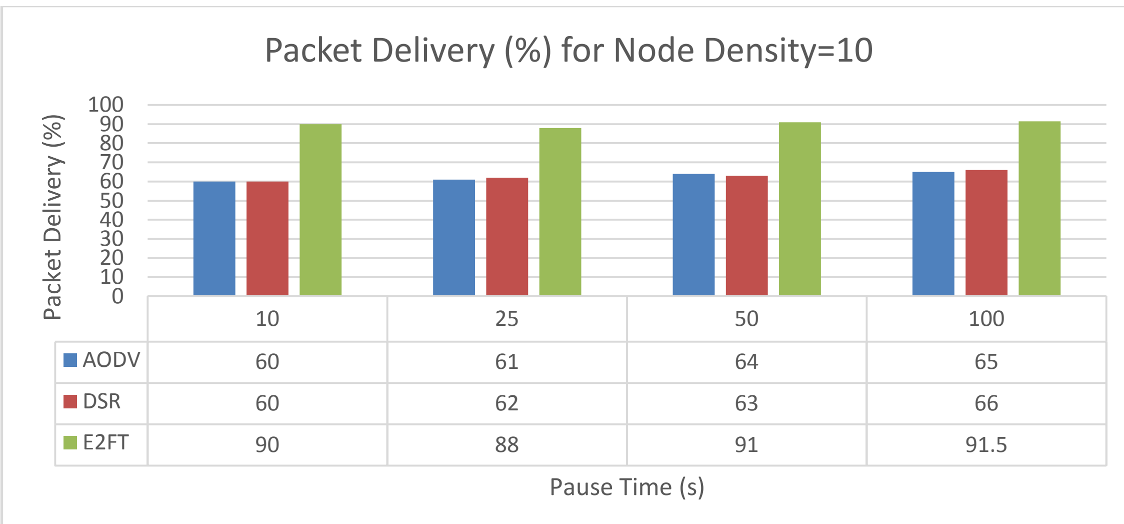
Parameter	Value
Area	1500m x 1500 m
No. of nodes	10, 25, 50, 100
Protocols	AODV, DSR, E2FT
Data transfer rate	2 MBPS
Mobility Model	Random Waypoint
Packet size	512 Bytes
Data Transmission Speed	4 Packets/Second

Table 6.1: Simulation Parameters

6.4 Results

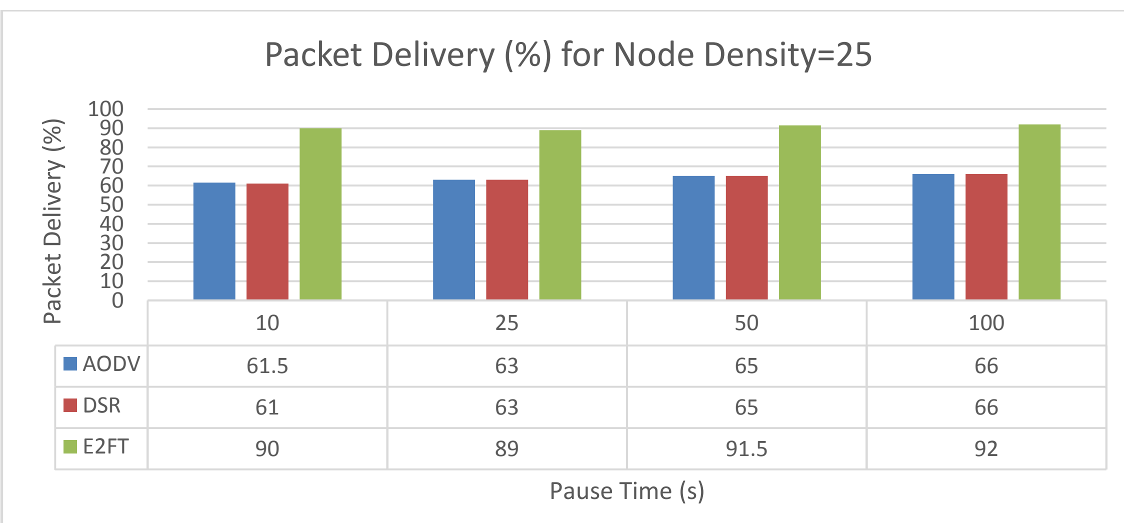
6.4.1 Variation in Pause Time

Figure 6.1 to 6.4 shows the packet delivery (%) on variation of pause time (s) between 10, 25, 50 and 100 for node density 10, 25, 50 and 100 respectively. E2FT clearly shows better performance than others. The reason is, it uses a stochastic learning-based weak estimation procedure to enhance a route estimation phase. The reactive protocols AODV and DSR are single path routing algorithms, hence the lower values in the graph. While E2FT is multipath algorithm, hence perform better.



	10	25	50	100
AODV	60	61	64	65
DSR	60	62	63	66
E2FT	90	88	91	91.5

Figure 6.1: Packet Delivery (%) for Node Density = 10



	10	25	50	100
AODV	61.5	63	65	66
DSR	61	63	65	66
E2FT	90	89	91.5	92

Figure 6.2: Packet Delivery (%) for Node Density = 25

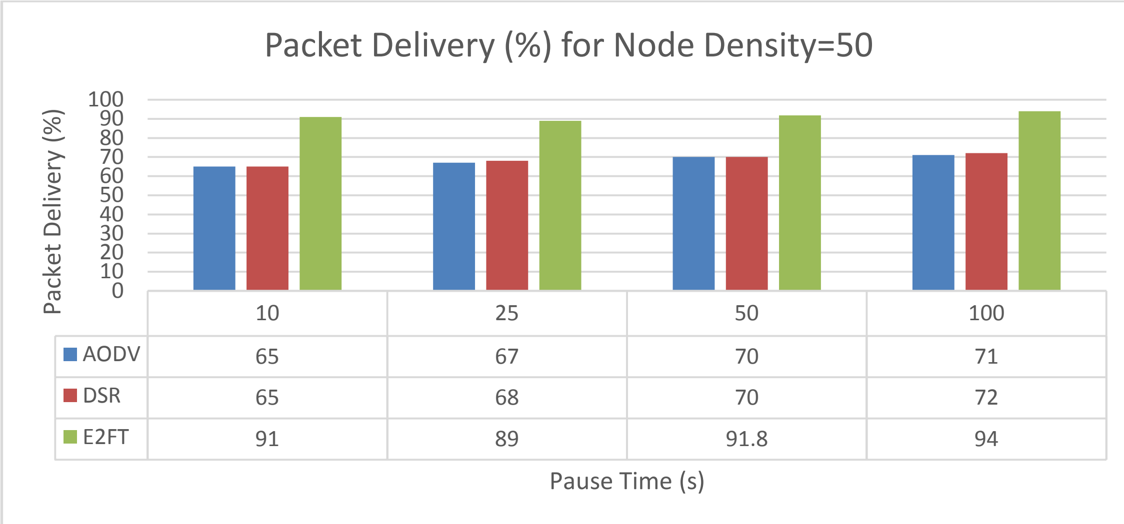


Figure 6.3: Packet Delivery (%) for Node Density = 50

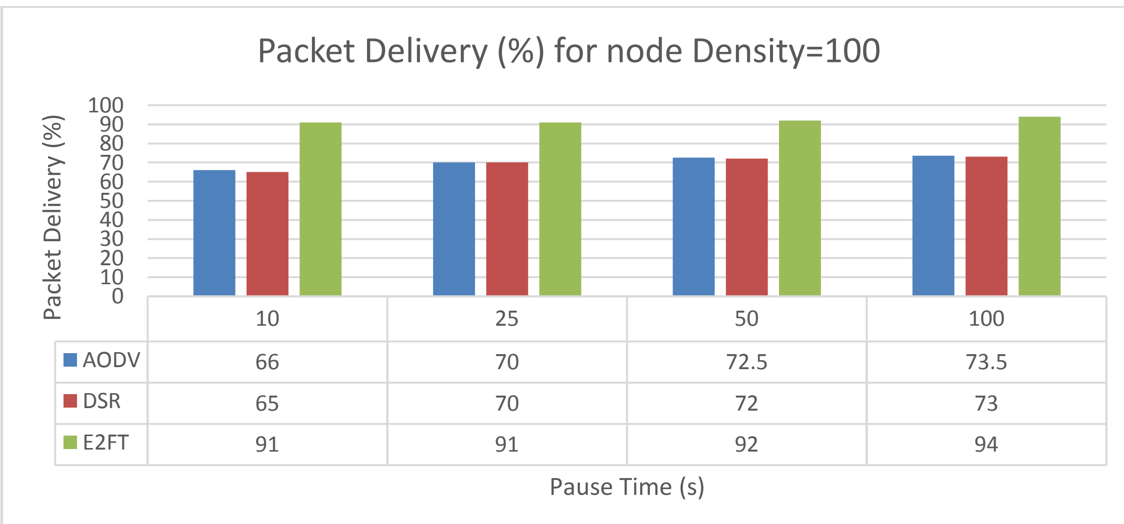


Figure 6.4: Packet Delivery (%) for Node Density = 100

6.4.2 Variation in Faultiness

Figure 6.5 to 6.8 shows the packet delivery (%) on variation of faultiness (%) between 25, 50, 75 and 95 for node density 10, 25, 50 and 100 respectively. E2FT has an edge over others, when faultiness parameter is considered. The reactive protocols AODV and DSR are single path routing algorithms, hence the lower values in the graph. While E2FT is a multipath algorithm, hence perform better.

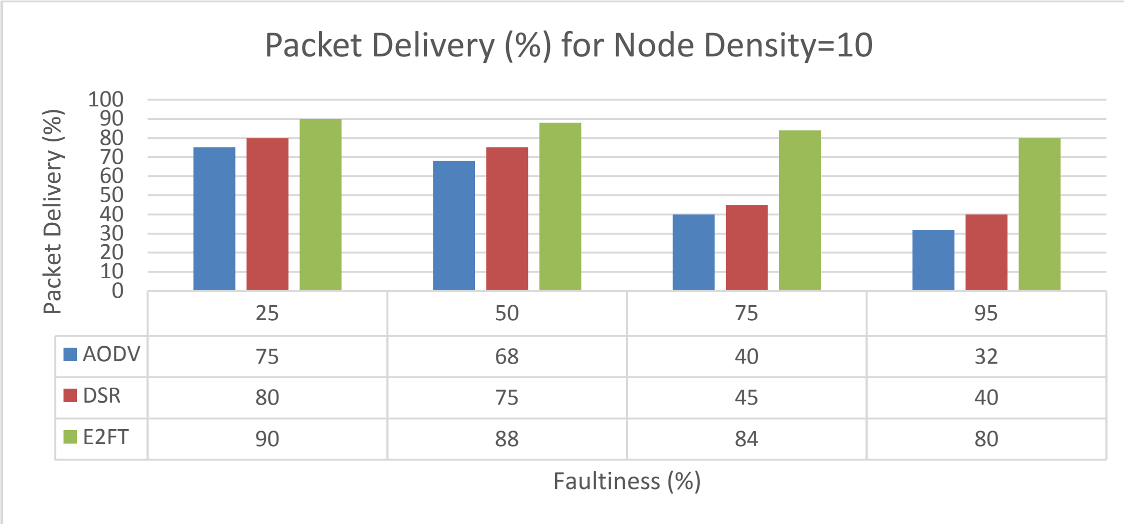


Figure 6.5: Packet Delivery (%) for Node Density = 10

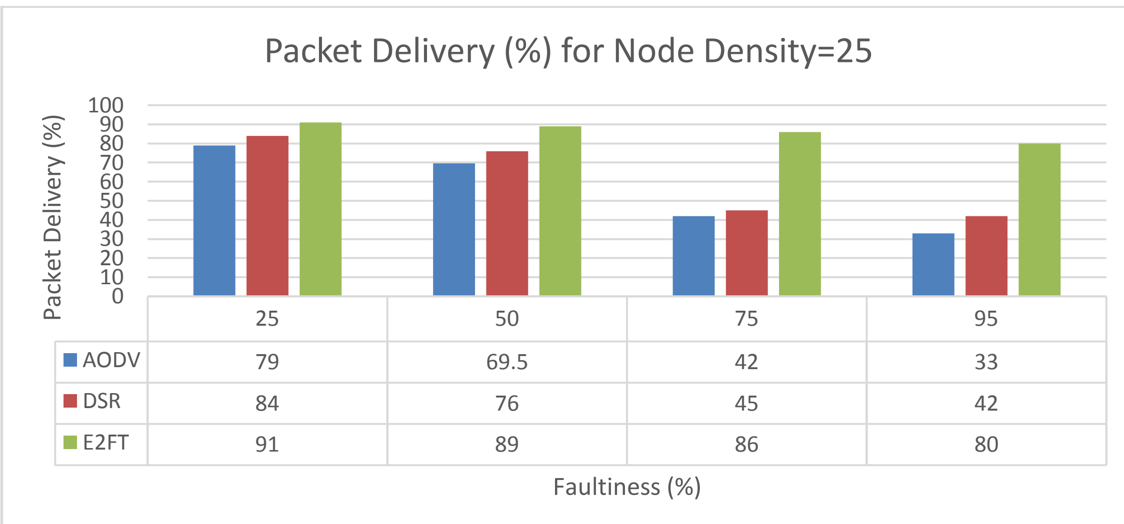


Figure 6.6: Packet Delivery (%) for Node Density = 25

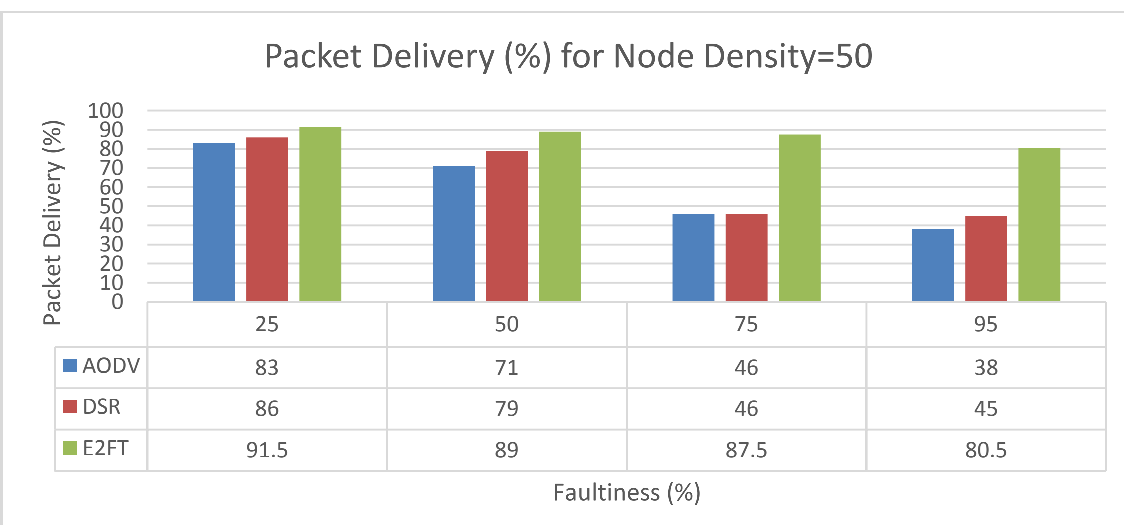
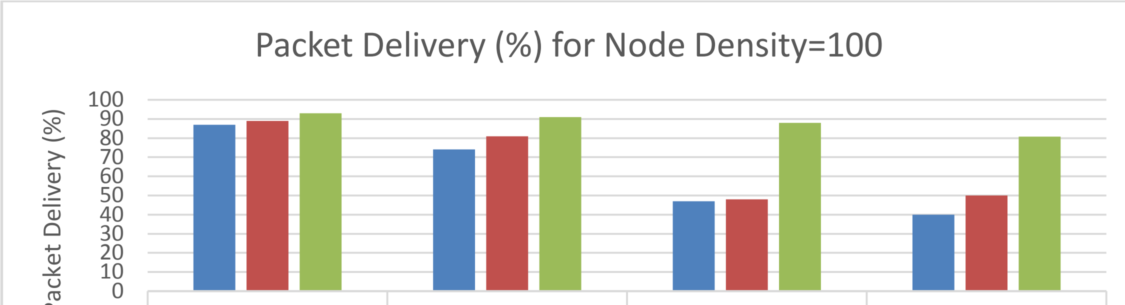


Figure 6.7: Packet Delivery (%) for Node Density = 50



	25	50	75	95
AODV	87	74	47	40
DSR	89	81	48	50
E2FT	93	91	88	80.8

Figure 6.8: Packet Delivery (%) for Node Density = 100

6.5 Conclusion

We studied the effect of variation in pause time and faultiness. Multipath algorithms offer greater fault tolerance than single path algorithms. The performance of single path algorithms, degraded when 50% or more nodes failed. From our performance evaluation, we demonstrate thatE2FT is the most efficient of all considered protocols in terms of the packet delivery percentage for pause time (s) and faultiness (%).