Tapestry Algorithm Failure Model

Implementation of Tapestry failure model

5. Fault tolerance using failure models

In a dynamic node failure scenario, fault tolerant systems do not have graceful node exits. A node may enter and exit any number of times in a short span. While implementing a fault tolerant tapestry network, it is observed that node failures interrupt message passing at three stages. Therefore, a failure model has been built to handle all the scenarios.

5.1 Involuntary node deletion during message passing

When node failure happens during message passing in the tapestry network, the node gets removed from the network and no other node can communicate with it. This is implemented in the actor-model by removing the node's PID value so that no other node can access it. The deleted node could be playing one of the below roles in a request: source node, destination node, intermediate node. Failure model implemented by the actor-model is designed to handle all the three scenarios.

5.1.1 Failed node is the source node

This scenario is encountered in case where the input number of requests is greater than 1. Every node is required to send one request per second to its peers based on the numRequests parameter. So, when a node fails, it cannot send any requests to its peers.

5.1.2 Failed node is the destination node

When the destination node for a request fails, then surrogate routing is done, and the message is delivered to the surrogate node. This new node acts as a surrogate to the deleted node and maximum number of hops attained for this request is always 4. Hence, when a destination node fails for a request it increases the maximum number of hops traversed by peers in a tapestry network.

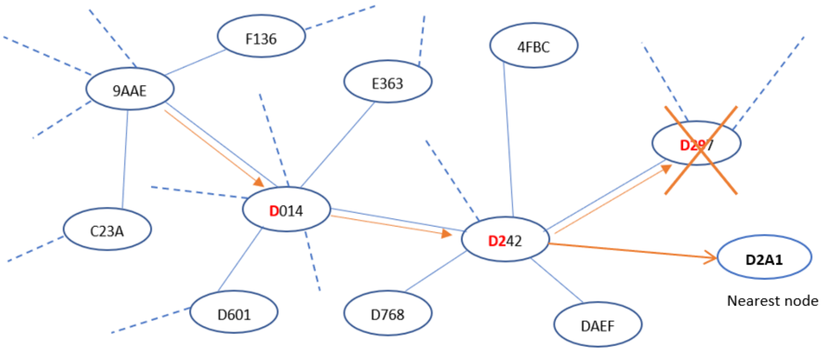
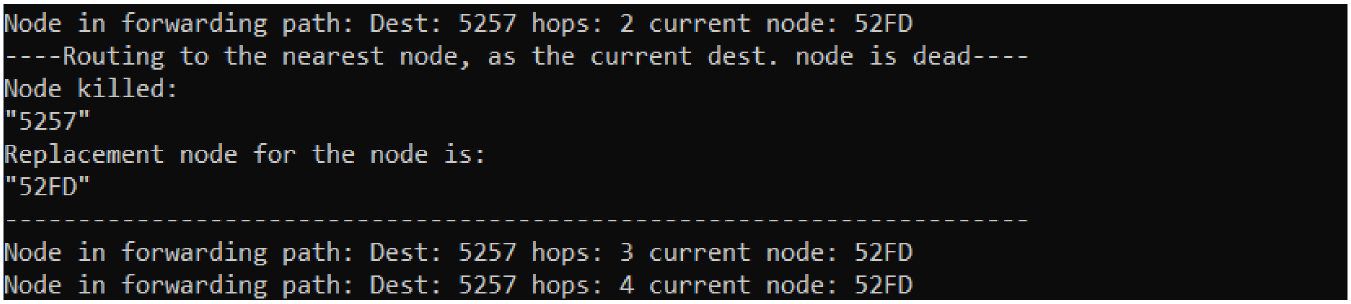


Figure 1: Surrogate routing to the nearest node when destination node fails



```
Node in forwarding path: Dest: 5257 hops: 2 current node: 52FD
----Routing to the nearest node, as the current dest. node is dead----
Node killed:
"5257"
Replacement node for the node is:
"52FD"
----------------------------------------------------------------------
Node in forwarding path: Dest: 5257 hops: 3 current node: 52FD
Node in forwarding path: Dest: 5257 hops: 4 current node: 52FD
```

Example 1: depicts when destination node **5257** fails, the message is routed to its surrogate node **52FD**

5.1.3 Failed node is an intermediate node

If an intermediate node fails during message passing request, in order to maintain availability and redundancy, the local node iterates over failed node's backpointer node IDs and gets the best suitable nearest forwarding pointer (root node with maximum matching prefix) alive for the intermediate failed node. The backup forwarding pointer computed acts as a replacement for the failed node and message request is forwarded to this replacement so that the message passing does not halt and reaches the destination.

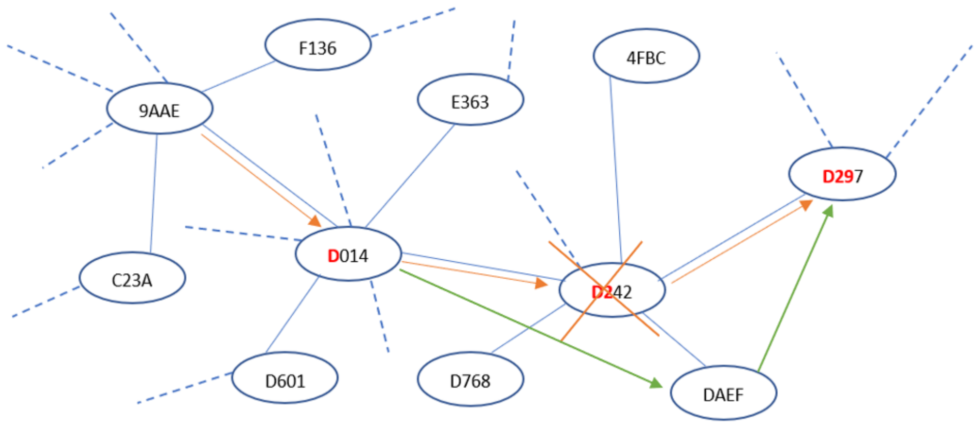
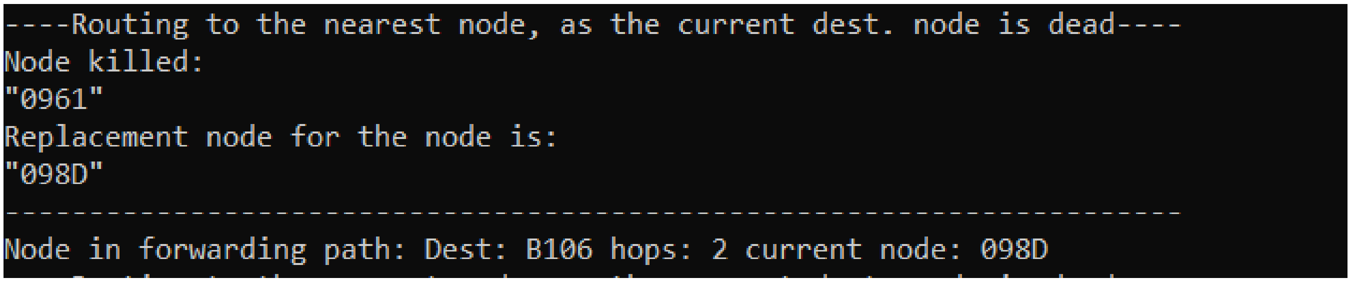


Figure 2: Message forwarded to root node when intermediate node fails



```
----Routing to the nearest node, as the current dest. node is dead----
Node killed:
"0961"
Replacement node for the node is:
"098D"
----------------------------------------------------------------------
Node in forwarding path: Dest: B106 hops: 2 current node: 098D
```

Example 2: depicts when intermediate node **0961** fails, the message is forwarded to **098D** which is the nearest replacement node sharing maximum prefix value

Observations for message passing in Tapestry Algorithm Failure Model

1. It has been observed that the value of "numRequests" influences the maximum hop count. When number of requests from peer to peer increases, probability of running into scenarios where the destination node ID (nodes that are far without matching prefix) takes 4 hops also increases.
2. When nodes in a network are increasing, the hop count also increases.
3. The above observation (2) is made in 2 scenarios, first one when the destination ID and source ID do not share common neighboring nodes in their respective routing tables and second one is when the slots in a node's routing table are not able to accommodate all the nodes sharing similar prefix values which will in turn result in an increase of hop count.
4. The probability of a message reaching a surrogate node is proportional to node failures. The number of hops taken in case of surrogate routing is 4, therefore increase in percentage of node failures always results in a maximum hop count 4.
5. Intermediate node failures are handled better in a larger network because when number of nodes increases, there is a higher range of nodes with similar prefix. Hence, when one node fails, it can easily find a suitable match (root node with maximum prefix match) to forward the message.
6. During the above scenario (5), when the percentage of node failures is less for a larger network, the root node can achieve message passing to destination ID within 4 hops, but when the node failure percentage is more, the hop count increases accordingly.

Number of Nodes	Failed nodes %				
	10	20	30	40	50
10	2	2	1	2	1
15	2	2	4	4	4
20	4	2	4	4	4
30	4	4	4	4	4
50	4	4	4	4	4
100	4	4	4	4	4
200	4	4	4	4	4
300	4	4	4	4	4
500	4	4	4	4	4
1000	4	4	4	4	4
2000	4	4	4	4	4

Table 1: Maximum number of hops obtained when **NumRequests=1**

Number of Nodes	Failed nodes %				
	10	20	30	40	50
10	2	2	4	4	4
15	4	4	4	4	4
20	4	4	4	4	4
30	4	4	4	4	4
50	4	4	4	4	4
100	4	4	4	4	4
200	4	4	4	4	4
300	4	4	4	4	4
500	4	4	4	4	4
1000	4	4	4	4	4
2000	4	4	4	4	4

Table 2: Maximum number of hops obtained when **NumRequests=5**

Number of Nodes	Failed nodes %				
	10	20	30	40	50
10	4	4	4	4	4
15	4	4	4	4	4
20	4	4	4	4	4
30	4	4	4	4	4
50	4	4	4	4	4
100	4	4	4	4	4
200	4	4	4	4	4
300	4	4	4	4	4
500	4	4	4	4	4
1000	4	4	4	4	4
2000	4	4	4	4	4

Table 3: Maximum number of hops obtained when **NumRequests=10**