is very low, as it needs a creation and transfer of a learning packet for every data packet forwarded.

Dual reinforcement Q-Routing (DRQR) [4] is a modified version of the Q-Routing algorithm, where learning occurs in both ways. Since, the learning process occurs in both ways the learning performance of the Q-Routing algorithm doubles. However, it adds more overheads to the network.

This paper investigates the performance of all three algorithms by simulating them on 14 nodes NFSNET. Their throughput and packet delay analysis will be discussed in section 5.

II. SHORTEST PATH ROUTING ALGORITHM

Shortest path algorithm also known as the Dijkstra's algorithm, is one of the most widely deployed routing algorithms in today's internet works, OSPF [5]. Shortest path routing (SPR) algorithm computes the routes based on the cost metric, which is expressed in different metrics by the network administrators as a function of delay, data rate, financial cost etc.

Every node in the network computes its routes to every other node in the network. The shortest path algorithm can be best explained by considering the network as a directed graph, where, every node is treated as an edge in the digraph and every link as the vertex connecting any two edges [6]. The cost of these vertices should always be non-negative.

Let V be a set of all the edges of the digraph and $c[s,w]$ be the array of the cost metrics of the vertices and $D[w]$ be the array of the least costs (of the shortest path) to any node to be computed. Initially every edge (node), before computing its routes, sets the cost of the shortest path to any other edge as infinity *i.e.* $D[w] = \infty$. If there are n edges in the digraph, every edge s starts computing the costs $D[w]$ of the shortest paths to other nodes by choosing its neighbour $w \in (V\text{-}s)$ with minimum $c[s,w]$ value. Then, it computes the routes for every edge $v \in (V\text{-}s\text{-}w)$, which is a neighbour of w, such that $D[v] = \min(D[v], D[w]+c[w,v])$, where $c[w,v]$ are the costs of the vertices connected to w. This process is continuous until all the edges of the digraph are visited. When the algorithm is stopped all the nodes in the network should have discovered the shortest path to the every other node in the network [1].

III. Q-ROUTING ALGORITHM

Q-Learning is a model free algorithm that learns from the delayed reinforcements and it is one of the easiest approaches to implement in reinforcement learning, thus one of the most popular [7]. Q-learning is applied to the routing problem in Q-Routing algorithm, where routing table in the distance vector algorithm [7] is replaced by the table of estimations (Q values) based on the link delay. In the Q-Routing same routing policies are used as in the distance vector routing algorithm. Q-function is used to update routing table entries in Q-Routing [3]. However, it has been suggested that neural networks can be used for approximating the Q-function by incorporating diverse parameters of the network such as time delays, queue lengths and etc [3].

In Q-Routing, when x sends a packet to the node d via its neighbour y it receives *y's* estimated remaining trip times to the destination *d.* Then it selects the neighbour with the smallest time (equation 1). Q-Routing uses forward exploration to update the Q values in the routing table that represents the delivery time estimation to the given destination (equation 2).

$$Q_y(z',d) = \min_{z \in N(y)} Q_y(z,d) \quad (1)$$

$$\Delta Q_x(y,d) = \eta_f (Q_y(z',d) + q_y - Q_x(y,d)) \quad (2)$$

$\Delta Q_x(y,d)$ is the new estimation value for node x to destination d via the neighbour node y (fig. 1). This new estimation is calculated by subtracting old estimation value $Q_x(y,d)$ from the sum of the estimation time for packet travelling from node y to destination d via neighbour z ($Q_y(\acute{z},d)$) and current queue delay for the packet in node x (q_x). η_f[2] is the learning rate parameter defined by the programmer. In fig. 1, x updates its $Q_x(y,d)$ value pertaining to the remaining path of packet p via node y.

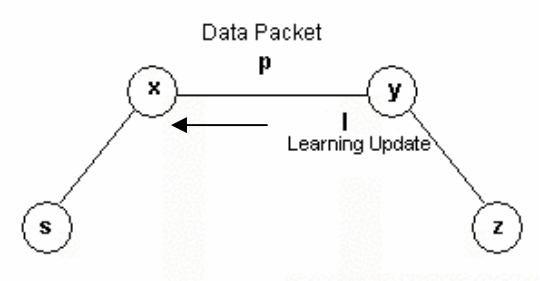

Figure 1: Forward exploration

[2] η_f is set to 0.7 in original Q-routing algorithm.