International Journal of Enhanced Research in Science, Technology & Engineering
ISSN: 2319-7463, Vol. 4 Issue 7, July-2015

Hardwired Implementation of Internet Protocol

Amit Kumar Dutta
Calcutta Institute of Technology, Uluberia, West Bengal, India

ABSTRACT

In this article, we attempt to solve a problem of sending information bits from a node to another node in a well connected network without any software control. We use a concept which is similar to the Dynamical Programming with depth of one node in a self directing manner and design an Internet protocol. We also propose a router circuit to implement this idea.

1.0 INTRODUCTION

In networking finding the shortest path dynamically is a complicated subject. Since middle of twentieth century, renowned mathematicians have attempted to solve that problem. Jaccobi 's quote in the context of finding the alternate route incase the shortest route is blocked in long distance telephone call in USA, in 1955 can be found in reference [1][2]. They understood the problem is similar to usage of freeway. To find the shortest path in a given graph with path characteristics, there are the methods of Bellman and Ford and Dijkstra's method. There are also variations of those methods. Orden found a solution of shortest path problem by linear programming. All these method needs calculation which varies in number with the number of nodes at passing through each node. Here, we suppose that the node address is in latitude, longitude and altitude of a geographical location of the node. Hence, we get the distance vectors with direction and choose the minimum vector in given direction of destination from source.

The organization of the paper is as follows. Section II introduces the dynamic programming. Section III introduces the protocol which is a modified dynamic programming. Section IV shows the network topology on which this protocol may work. Section V gives the hardwired implementation of the internet protocol. Section VI deals with the twisted pair coaxial cable, which allows better capacity. The last section concludes the paper.

2.0 DYNAMIC PROGRAMMING

The dynamic programming technique uses a very simple idea which is principle of optimality. In mathematical term it can be formulated as [3][4],

$$J_0(X_0) = E[g_K(X_K) + \sum_{n=0}^{K-1} g_n(X_n, \mu_n(X_n), X_{n+1})]$$

Where K is the number of stages and $g_k(X_k)$ is the terminal cost. It can be stated as follows,

1) Construct an optimal policy for the tail sub-problem involving only the last stage of the system.
2) Extend the optimal policy to the tail sub-problem involving the last two stages of the system.
3) Continue the procedure in this fashion to solve the entire stages.

3.0 MODIFIED DYNAMICAL PROGRAMMING AND INTERNET PROTOCOL

We perceive the internet as a interconnected network through which the data flows from a node to destination node. We have two problems: the problem of connectivity and problem of shortest path. Later we shall understand that both manifest into a single solution of choosing the alternate path given the lowest metric defined by their distance vector and other criteria. The problem of connectivity is solved by well connected network. This mean in a network a node is connected to any other node and not only that, in any direction network exist which ends in a closed boundary. The examples will explain the well connected network in the next section.

The problem of shortest path is explained as follows: the data has to travel from node 1 to node N through node 2 or 2' or 2'' , node 3 or 3' or 3'' and so on. Here we consider N as 4. We find the metrics as,

$$d1 = k_1|x_2 - x_1| + k_2|x_4 - x_2|$$
$$d1' = k_1|x_2' - x_1| + k_2|x_4 - x_2'|$$
$$d1'' = k_1|x_2'' - x_1| + k_2|x_4 - x_2''|$$

Where x is the locating vector by longitude, latitude and altitude of the location, k depends on the usage of the node and we choose the least metric. Assume that d1 is the least, then from node 2 we try to reach node 4 by finding minimum metric using nodes 3, 3' and 3''. The metrics are found as,

$$d2 = k_1|x_3 - x_2| + k_2|x_4 - x_3|$$
$$d2' = k_1|x_3' - x_2| + k_2|x_4 - x_3'|$$
$$d2'' = k_1|x_3'' - x_2| + k_2|x_4 - x_3''|$$

Assume that the least metric is d2', then the path the data traverse is node 1, node 2, node 3' and node 4. The total metric is $k_1|x_2 - x_1| + k_1|x_3' - x_2| + k_2|x_4 - x_3'|$ which is least. Here we assume that there exists a path from node 1 to node 4. Similarly, we can find the route dynamically by calculating the metric and choosing the least. Here, the optimal path is found for state of x_4 to x_1, which is same as x_1 to x_4 as the system is static in nature. So the data is divided into data blocks and each block has its header the destination address. Each step it calculates the metrics from that node to its connecting nodes and chooses the least metric. If the network is well connected the data will reach the destination. In case the network is not well connected we have to attach the node names of the traversed path so that it does not go into loop.

4.0 NETWORK TOPOLOGY

The network should be well connected for dynamical programming with depth one node to work. It means that in any direction network should continue (with bifurcation) till it reaches the closed boundary. It should not stop and start again. It is explained in Figure 1.

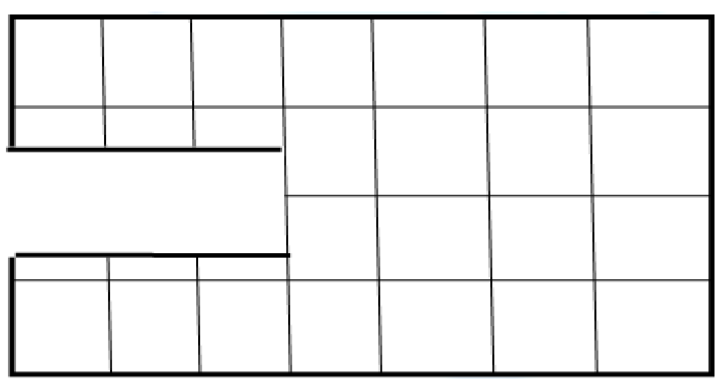

Figure 1. Not well connected network.

A well connected network topology is explained in Figure 2. It is a grid topology. A better but more complicated network is the star network shown in Figure 3.

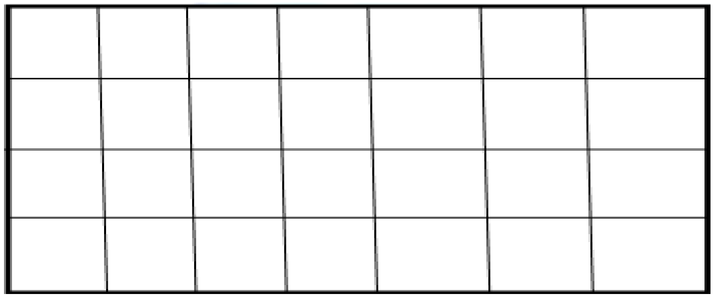

Figure 2: Grid connected network topology with 2 x 2 router at cross over.

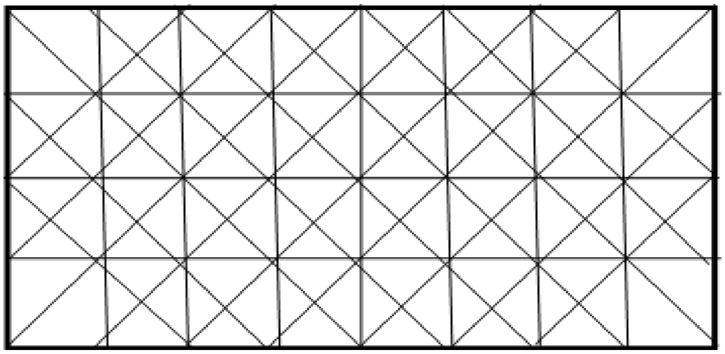

Figure 3: Star network topology with 4 x 4 router.

The difference between Figure 1 and Figure 2 or Figure 3 is that, in any direction network is there in a closed boundary. Because $|x_n-x_{n-1}|$has direction towards the destination. If we put the direction in the opposite direction, it may go into loop. Hence, topology as shown in Figure 1 is not wanted.

5.0 HARDWIRED IMPLEMENTATION OF THE PROTOCOL

We can implement the router easily using Q-Dot technology. The Figure 4 explains a 2-input-2-output router. Two 2 x 2 routers are used to form a router for 4 input-4 output used in star grid.

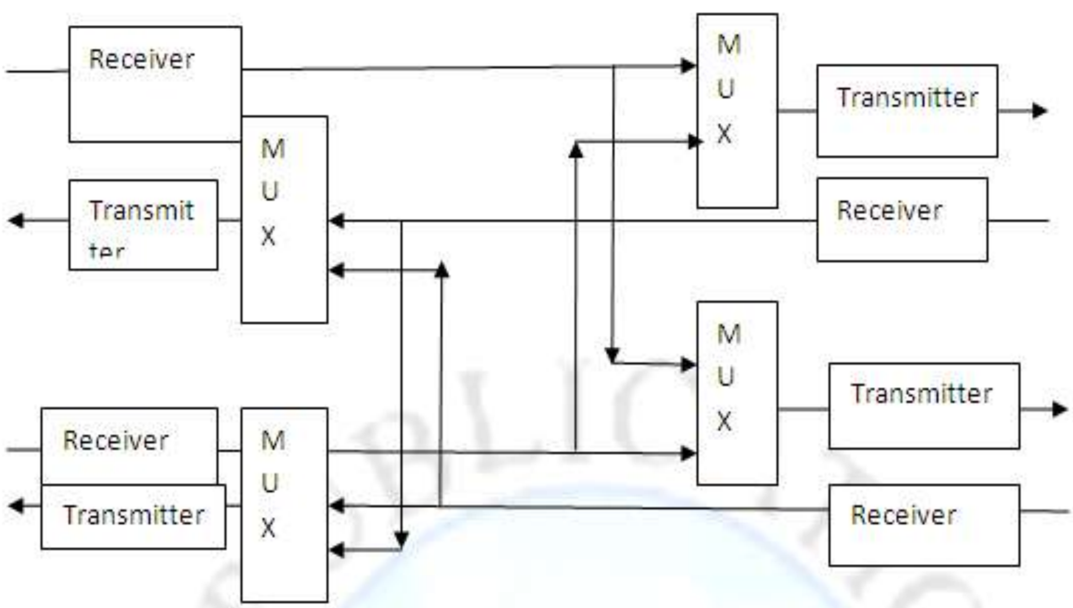

Figure 4: 2 x 2 router.

We can use the available bandwidth of the cable to increase the throughput to 100 Mbps. If it is a coaxial cable, we can divide the bandwidth for outgoing and incoming data stream. We have a cross coupler using multiplexor whose select bit is chosen by the metric. We decode the data at the receiver and look for data block header. Then we calculate the metric and change the data flow using multiplexor. The receiver structure is as shown in [5]. The modulation is QAM and FDMA multiple access. Figure 5 and Figure 6 explain the cross bar of router without regeneration.

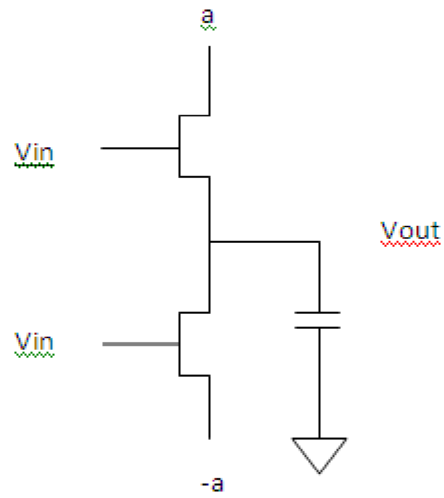

Figure 5: Variable gain with high bandwidth.

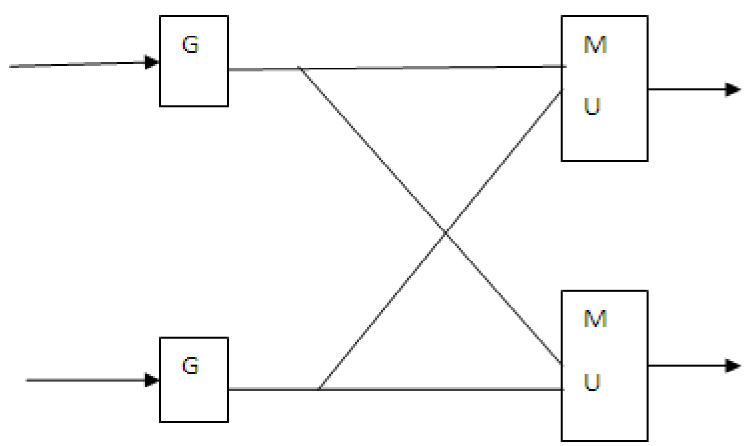

Figure 6: Crossbar in Router without regeneration. G is the variable gain.

6.0 CABLE

As the incoming and outgoing are in same cable wire, it is difficult to implement. Also, twisted pair cable with or without sheath is available. We can use the incoming and the outgoing in different line. As the signals are voltages there will be electrostatic voltage induced on receive signal from the transmitted signal. This is proportional to the transmitted signal and we cancel them using simple hardwire. This trick cannot be used if both the wires carry data in the same direction. Thus we can double the throughput.

CONCLUSION

Here we discussed a self directing router which implements dynamic programming optimization. This protocol is implemented in hardwire and is most suitable for high speed routing. It does not require storage memory at the nodes and any routing table.

REFERENCES

[1]. Schrijver A., “On the history of shortest path problem,” Documenta Matyhematica, ISMP 2012.
[2]. Forouzan BA, “Data Communications and Networking,” Tata McGraw Hill, 4th Edn..
[3]. Haykin S., “Neural Networks and Learning Machines,” PHI, 3rd Edition.
[4]. Bellman R., “Dynamic Programming,” Princeton, NJ: Princeton University Press.
[5]. Dutta A. K., “Receiver for QAM Modulation,” IJERSTE, ISSN 2319-7463 Vol. 4 Issue 4, April-2015.