CHAPTER V

NETWORKING BASED ON MODIFIED DE BRUIJN GRAPHS AND ITS PERFORMANCE MEASURES

5.1 Introduction to Modified De BruijnGraphs:

In the year 1999 B. Satyanarayana proposed a new graph know as Modified De Bruijn Graph based on De Bruijn Graph explained in chapter 3 [67]. A Modified De Bruijn Graph with respect to degree 'Δ' and diameter'd' is denoted by M (Δ,d) whose construction is explained as follows.

Construction of Modified De Bruijn Graphs:

The construction of Modified De Bruijn Graphs M (Δ, d) is mainly based on De Bruijn Graphs D (Δ, d). The network for Modified De Bruijn Graph is defined for total number of nodes (N),where N = Δ^d ($\Delta \geq 2$, d ≥ 2). **The basic difference between M(Δ,d) and D(Δ,d) is that presence of self loops in D(Δ,d) are removed because of the fact that the necessity does not arises to transmit an information/packet to the same node where it is generated.**

For example:

One can talk on phone to the other phone but not to thé same phone from where he is calling. Similarly, **messages are to be sent to one computer (Source) to another computer (Destination) but not to the same computer (Source) it self.** Thus self loops are removed in D(Δ,d) to obtain M(Δ,d). In M(Δ,d) self loops are removed and the nodes $(0_1,0_2,\ldots,0_d)$,$(1_1,1_2\ldots\ldots1_d)$ ….$((\Delta-1)_1,(\Delta-2)_2\ldots(\Delta-1)_d)$ are connected to one another in a cyclic manner.

Structural properties of M (Δ, d):

In M(Δ,d) 'Δ' self edges that are physically added to D(Δ,d) thus M(Δ,d) network distance between nodes for significant number of node pairs are reduced the reduction of distance between the nodes in M(Δ,d) makes the graph more efficient than D(Δ,d).

5.2 Algorithm for calculation of various parameters of Modified De Bruijn Graphs.

In order to develop the algorithm, schematic representation of connectivity of $M(\Delta,d)$ is given in the following figures for M(2,2) and M(2,3).

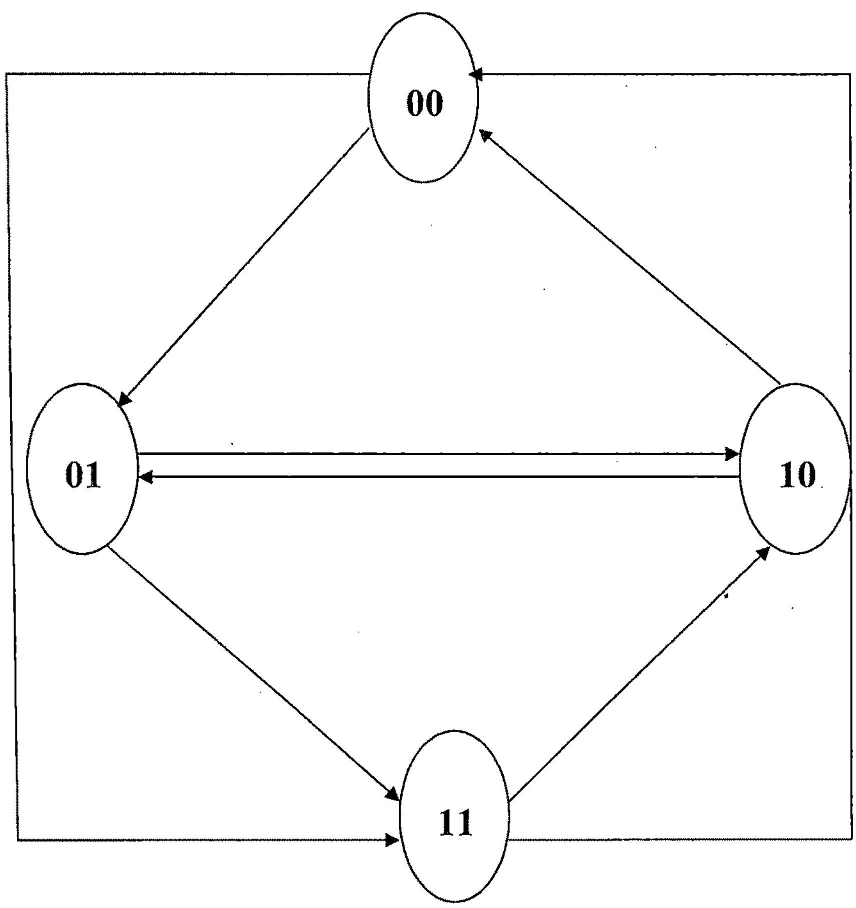


Figure (5.2.1) The Modified De Bruijn Graph M(2,2).

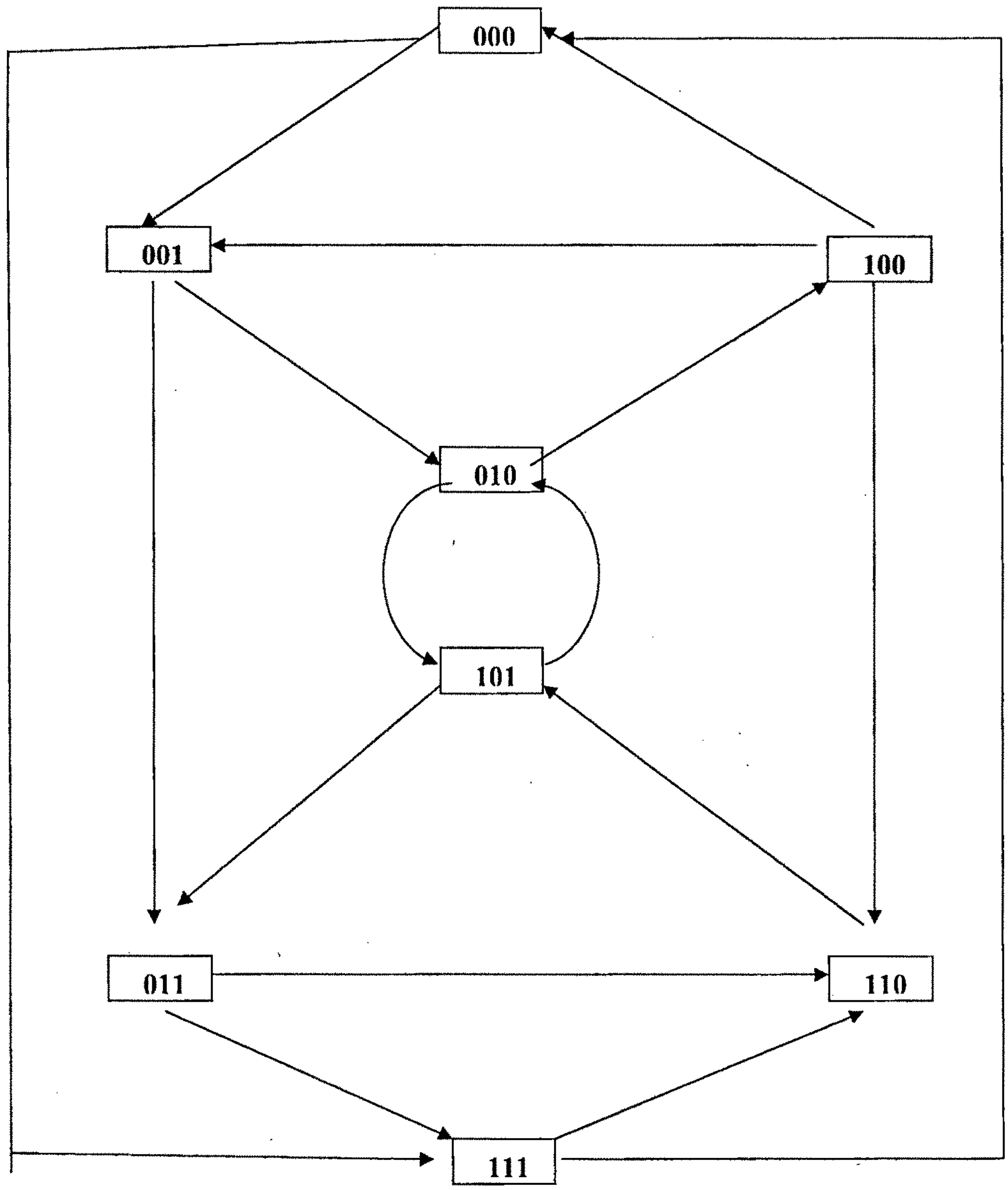


Figure (5.2.2) The Modified De Bruijn Graph M(2,3).

In order to calculate various performance measurers for M(Δ,d) formulae are developed and are given as follows:

Total Number of nodes	N	$= \Delta^d$	----------(5.2.1)
Average Hop Length	$\overline{H}$	= Hsum/N(N-1)	---------(5.2.2)
Average Edge Loading	$\overline{L}$	= Hsum/(N*Δ)	----------(5.2.3)
LMAX (Maximum Edge load)			----------(5.2.4)
Average Network Utilization	U_{AVG}	= $\overline{L}$ / LMAX	----------(5.2.5)
Throughput	λ	= 1/LMAX	--------- (5.2.6)

Hsum: Sum of the shortest paths of all possible source and destination nodes.

An algorithm is developed for calculation of various performance measures of Modified De Bruijn Graph is given as follows:

Modified De Bruijn Graphs have a self-routing property, namely, packets can be routed from a source(src) to a destination (dest) Using distributed routing algorithm in which the packets are forwarded to intermediate node, basing on a routing decision given only on the label of the destination node of the packet. The basic difference in De Bruijn and Modified De Bruijn Graph is, in De Bruijn Graphs self loops are present where as in Modified De Bruijn Graphs self loops are removed wide Fig(5.2.1) and Fig(5.2.2).

In this section, we explain the notations and definitions used in the development of shortest path algorithm.

A Shortest path routing algorithm for Modified De Bruijn Graphs (networks) from a source node Src = ($a_1,a_2,\ldots\ldots a_d$) to a destination node Dest = ($b_1,b_2,\ldots b_d$).

Δ: Degree of the network

d: Diameter of the network(Which denotes the maximum length between any nodes)

N : Total number of nodes in the Network.

The nodes of the Modified De Bruijn Graphs(network) are classified into two types.

(i) Self-nodes, in which all Symbols in the label are same.

(ii) Non-self nodes, in which all symbols in its label are not same.

The source- destination pairs are categorized into four types

(i) Self-node to Self-node

(ii) Non-Self node to Non-Self node.

(iii) Self-node to Non-Self node

(iv) Non-Self node to Self-node.

Is Self(Node) -> Finds whether it is a Self-node or not.

Both Self -> Finds the Shortest distance in number of hops between a Self-node(source) to a Self-node(destination) using self-nodes alone.

PreselftoDest-> refers to the preceding self-node to the destination node.

DistPreselfToDest-> Finds the preceding self-node to a Non-self node(destination) and also returns the shortest distance between the PeselftoDest(self-node) and destination.

AdjselftoSrc -> refers to the adjacent self-node to the source node.

DistofAdjselfToSrc-> Finds the adjacent self node to source(Non-self node)

And returns the shortest distance between the source and AdjselftoSrc.

DistSrcToAdjself-> finds the distance between source node and the immediate adjacent self node.

Shiftmatch(i,Src,Dest) , $0 \leq i \leq d$ → An operation on the two strings

Src and Dest to be TRUE iff src = $(b_1,b_2,\dots b_{d-i}) = (a_{i+1},a_{i+2},\dots a_d)$

And false otherwise.

Merge (i ,Src,Dest), $0 \leq i \leq d$ → is a Sting(or Sequence) of length d+i

Given by $(a_1,a_2,\dots\dots a_d,b_{d-i+1},\dots\dots b_d)$.

Find NormalLength Find the distance between source node and the destination node using theDe Bruijn routing.

Transmit Next node refers to the adjacent node from source node to which the packet has to be transmitted.

Algorithm for Shortest path routing in ModifiedDe Bruijn graph.

```
Procecure Modified _de Bruijn route ( Src,Dest,Δ,d)
Begin
    If(Src = Dest) consume packet;
Return;
If ((Isself(Src)= FALSE) and (Isself(Dest) = FALSE) then
    Len = both-nonself(Src, Dest);
Else if ((Is Self(Src =TRUE)and (IsSelf(Dest) = TRUE))
   Len = bothself(Src,Dest);
Else if ((IsSelf(Src=TRUE)and Isself(Dest=FALSE)) then
  Len = Source_self(Src,Dest);
Else if (Isself(Src = FALSE)and IsSelf(Dest = TRUE))
  Len = dest_self(Src,Dest);
  Transmitnext_node;
Return len;
end.
Procedure findNormalLength(Src,Dest, Δ,d )
```

```
Begin
For(i= 0to d)
Begin
If(Shift_match(i,Src,Dest)= FALSE)
Break;
End;
Merge(Src,Dest,i,spath);
Return i    ; ( i is the pathlength)
End;
Procedure both_self(Src,Dest, Δ,d)
Begin
Length= findNormalLength(Src,Dest, Δ,d);
If (Length < d) return Length;
Count  = DistBothSelf(Src,Dest);
If(Length<count) return Length;
Return count;
End;
Procedure both_nonself(Src,Dest, Δ,d)
Begin
Length = findNormalLength(Src,Dest, Δ,d)
If Length <d return Length;
Length 1 = DistOfSrcToAdjself(Src,AdjselfToSrc)+
             DistOfPreSelfToSrc(PreselfToDest,Dest) +
            DistBothself(AdjselfToSrc,PreSelfToDest);
If Length > Length 1
```

```
Return Length;

End;

Procedure SourceSelf(Src,Dest, Δ,d)

Begin

Length = FindNormalLength(Src,Dest, Δ,d);

If(Length<d) return length;

Length1 = DistBothSelf(Src,PreSelfToDist) +

          DistPreselfToDest(PreSelfToDest,Dest);

If(Length<Length1) return Length;

Return Length1;

End;

Procedure DestSelf(Src,Dest, Δ,d)

Begin

Length= FindNormalLength(Src,Dest, Δ,d);

If(Length<d) return Length;

Length= DistOfAdjselfToSrc(Src,AdjSelftoSrc)+

        DistBothself(AdjSelfToSrc,Dest);

If(Length <Length1) return Length;
Return Length1;
End;
```

For Example:

Δ=2 and d=3 and A =(001) and B(101)

Here A and B are recognized as Nonself to Nonself nodes respectively and it pertains to bothnonself routing procedure.

First, Find normallength(i,A,B) which yields = 2 hops

Which is less than diameter 'd' which is 3 in this case

Hence 2 is the shortest path length for A and B.

5.3 Calculation of various parameters of M(Δ,d):

Using the above algorithm a program is written in c–language. Various parameters of the network under consideration are calculated and are given in the following table.

Table (5.3.1) Calculated values of Performance measures of Modified De Bruijn Graphs

Δ	d	N	$\overline{H}$	$\overline{L}$	LMAX	U_{AVG}	λ
2	2	4	1.333333	2	3	0.666667	0.333333
2	3	8	1.964286	6.875	8	0.859375	0.125
2	4	16	2.733333	20.5	26	0.788462	0.038462
2	5	32	3.584677	55.5625	80	0.694531	0.0125
3	2	9	1.625	4.33333	7	0.619048	0.142857
3	3	27	2.448718	21.22222	31	0.684588	0.032258
3	4	81	3.370833	89.88889	138	0.651369	0.007246
3	5	243	4.33716	349.8642	535	0.653952	0.001869
4	2	16	1.733333	6.5	9	0.722222	0.111111
4	3	64	2.630952	41.4375	57	0.726974	0.017544
4	4	256	3.595037	229.1836	313	0.732216	0.003195
4	5	1024	4.583257	1172.168	1589	0.737677	0.000629
5	2	25	1.791667	8.6	11	0.781818	0.090909
5	3	125	2.724194	67.56	86	0.785581	0.011628
5	4	625	3.704795	462.3584	586	0.789008	0.001707

Some interesting and relevant statistics are calculated parameter wise [57] and are given in the following table using the data given in Table (5.3.1)

Table (5.3.2) S ome interesting and relevant statistics for various parameters of Modified De Bruijn Graphs

Statistics \ Parameters	N	$\overline{H}$	$\overline{L}$	LMAX	U_{AVG}	λ
Mean	170.3333	2.810705	169.2036	231.9333	0.726232	0.062017
Standard Error	74.25559	0.26259048	80.23005	108.7519	0.017035	0.023223
Median	32	2.724194	41.4375	57	0.726974	0.017544
Coefficient of Variation	168.8399	36.1833974	183.6425	181.6015	9.084764	145.028
Standard Deviation	287.5906	1.01700856	310.7297	421.1943	0.065976	0.089941
Sample Variance	82708.38	1.03430641	96552.92	177404.6	0.004353	0.008089
Kurtosis	5.52536	-1.0498533	8.319058	8.220688	-0.43998	5.57213
Skewness	2.376975	0.24301218	2.765519	2.741879	0.254546	2.206653
Range	1020	3.249924	1170.168	1586	0.240327	0.332704
Minimum	4	1.333333	2	3	0.619048	0.000629
Maximum	1024	4.583257	1172.168	1589	0.859375	0.333333
Sum	2555	42.160575	2538.054	3479	10.89349	0.930248
Count	15	15	15	15	15	15

5.4 : Results and Discussions:

Critically comparing the calculated values of various parameters Table (5.3.1) and the corresponding statistics of the parameter given in Table (5.3.2) ,One can draw the following conclusions:

- Modified De Bruijn Graphs also exhibits similar type of relation as exhibited by Shuffle Net (or)De Bruijn Graphs(or)Kautz Graphs with respect to all parameter for different values of degree ' Δ 'and diameter 'd'.
- With respect to Coefficient of Variation the parameter (U_{AVG}) is showing more consistency than other parameters.
- Maximum Variation is observed in Modified De Bruijn Graph with respect to the parameter $(\overline{L})$
- More Consistency is observed with respect to U_{AVG} in Modified De Bruijn Graph.

On similar lines of previous chapters parameter N and H are represented diagrammatically for various values of degree ' Δ ' and diameter'd'.

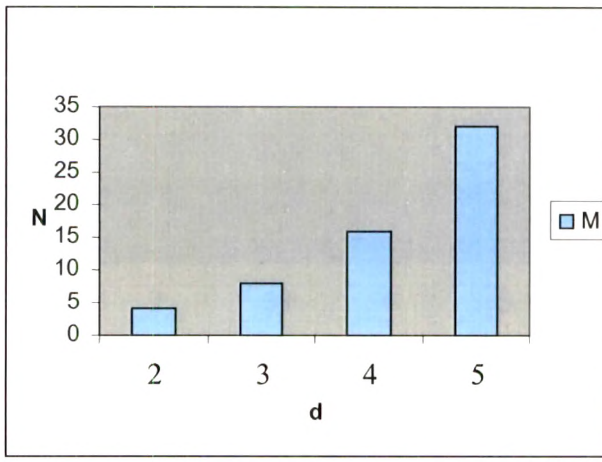
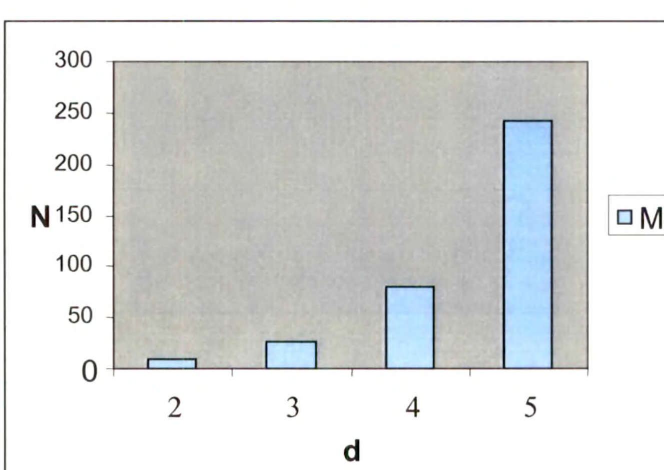


Figure (5.4.1) Total Number of nodes(N) for Modified De Bruijn Graph for various values of'd' when ' Δ=2'



Figure(5.4.2) Total Number of nodes(N) for Modified De Bruijn Graph for various values of 'd' when ' Δ=3'

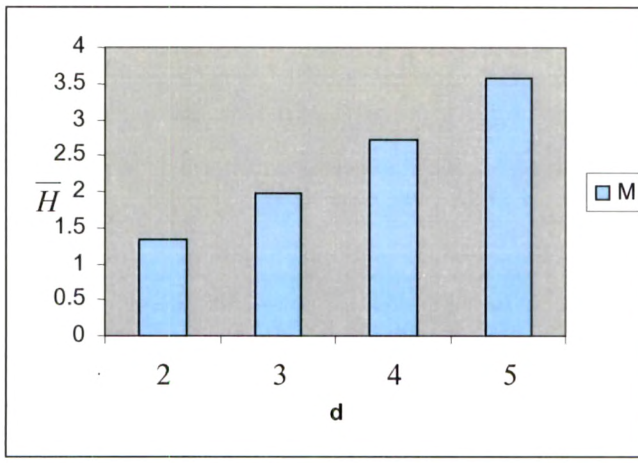


Figure (5.4.5) Average Hop Length($\overline{H}$) for Modified De Bruijn Graphs for various values of'd' when 'Δ=2'

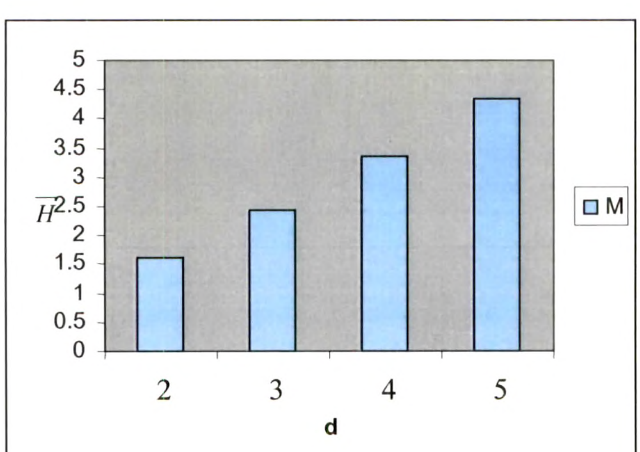


Figure (5.4.6) Average Hop Length($\overline{H}$) for Modified De Bruijn Graphs for various values of 'd' when 'Δ=3'

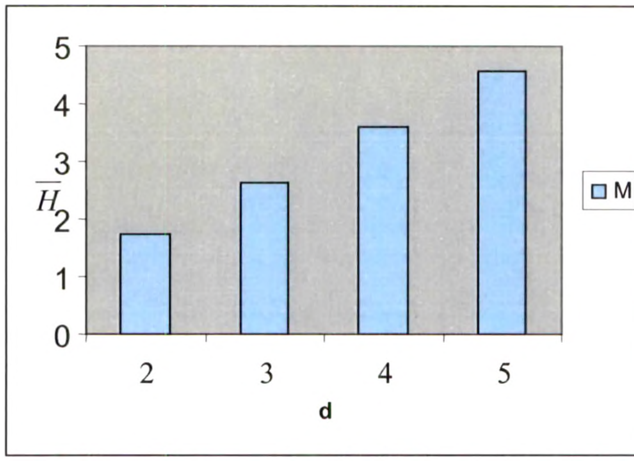


Figure (5.4.7) Average Hop Length ($\overline{H}$) for Modified De Bruijn Graphs for various values of 'd' when ' Δ=4'

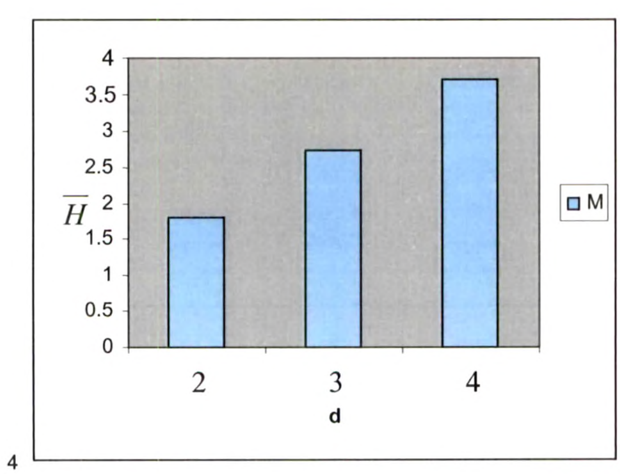


Figure (5.4.8) Average Hop Length($\overline{H}$) for Modified De Bruijn Graphs for various values of'd' when ' Δ=5'

Remarks :

Results of this Chapter was communicated for publication in the research Journal – (ANIJMIT) by Sarma, K.L.A.P., Satyanaraya, B and Praveen Kumar, P.T.V, entitled "Shortest Path routing algorithm based on Modified De Bruijn Graphs".