Aalborg Universitet

AALBORG UNIVERSITY
DENMARK

Degree 3 Networks Topological Routing

Gutierrez Lopez, Jose Manuel; Riaz, M. Tahir; Pedersen, Jens Myrup; Cecilio, Sergio Labeage; Madsen, Ole Brun

Published in:
Image Processing & Communications

Publication date:
2009

Document Version
Early version, also known as pre-print

Degree 3 networks topological routing

J. GUTIERREZ, T. RIAZ, J. PEDERSEN, S. LABEAGA AND O. MADSEN
Department of Electronic Systems, NetSec, Aalborg University
Niels Jernes vej 12
9220 Aalborg
Emails: jgl@es.aau.dk, tahir@es.aau.dk, jens@es.aau.dk, sl@es.aau.dk. obm@es.aau.dk

Abstract Topological routing is a table free alternative to traditional routing methods. It is specially well suited for organized network interconnection schemes. Topological routing algorithms correspond to the type $O(1)$, constant complexity, being very attractive for large scale networks. It has been proposed for many topologies and this work compares the algorithms for three degree three topologies using a more analytical approach than previous studies.

Key words Topological Routing, Chordal Ring, N2R, Double Ring, Restoration

1 Introduction

Topological routing is an alternative to traditional routing methods, based on tables. It allows for very fast restoration, and it is particularly well suited for large-scale transmission systems where table updates can be time consuming and introduce significant overheads [1]. Topological routing is defined as follows:

At a given address scheme, from any node any packet can be routed given only knowledge of the addresses of the current node and the destination node, with no routing tables involved [2].

Related to this topological routing idea there are several studies about Small world networks (SWN) which demonstrate that with limited knowledge of the network, a greedy algorithm can construct short paths using only local information. Studies such as Kleinberg's Small-world models [3] or Watts and Strogatz Ring Model [4] are examples of the huge number of publications that prove the existence of topological routing algorithms.

The complexity of topological routing algorithms should be independent of the number of elements in the network, $O(1)$, and only local information should be required, mostly neighboring information. This kind of algorithms can be classified as "*forwarding algorithms*", their function is to determine the best outgoing link for any incoming packet at any node.

The goal of this study is to present three topological routing algorithms for three degree 3 topologies and compare them. This analysis has been traditionally performed by empirical results, execution time measurements [5]-[7], but this study includes a more theoretical approach. A simple runtime analysis is performed in order to first prove, first, that are algorithms $O(1)$ and compare them in terms of generic algorithm runtime based on their instructions. In this way the results can be extrapolated to any specific hardware and software instead of requiring empirical performance measurement for each specific equipment or language.

The rest of the document is as follows. Section 2 treats the background, definitions and proper notation to of the three topologies presented. Furthermore, complexity in routing and other important concepts are presented. Section 3 introduces the selected algorithms and their basic operations. In Section 4 the results are presented and compared. Section 5 extracts the conclusions from this paper. Section 6 summarizes the notation in he document.

2 Preliminary concepts

2.1 Topologies

The following paragraphs describe the three topologies under study: Chordal Ring, N2R and Double Ring.

2.1.1 Degree Three Chordal Rings (CR)

Let w be an even integer such that $w \geq 6$, and let q be an odd integer, such that $3 \leq q \leq w/2$. w and q then define *CR(w, q)* with $N = w$ nodes labeled N_0, . . . , N_{w-1}. For $0 \leq i \leq w - 1$ there exists a line between each of the following pairs of nodes [8]:

- $(N_i, N_{(i+1)(mod(w))})$
- $(N_i, N_{(i+q)(mod(w))})$ for i even. See Fig. 1(a).

In relation to Chordal Rings, CR for the rest of the document, theoretical routing approaches have demonstrated the possibility of optimal distance routing and effective failure supporting for certain configurations [9]. Authors use Triple-loop graphs to formulate a routing methodology and they prove that it is possible to calculate optimal paths for each of the optimal configuration for a given diameter. This type of routing is not the goal of this study since the complexity of the methods increases with the size of the network and it is only valid for certain configurations. However, it inspires the work for the use of honeycomb structures.

Topological routing for CR has been studied at [10] and different algorithms are provided. The selected algorithm to work with in this study is based on Honeycomb structures named "*Honeycomb Based Algorithm (HCBA)*" which provides optimal or near optimal path distances in all the possible CR configurations.

Link notation:

The link notation must be mentioned for the proper understanding of the studied algorithms. Each node is connected to its neighbors by three links. Ring links L (Left) connects N_i to $N_i + 1$ and R (Right) connects N_i to $N_i - 1$ and Chordal link C (Center) connects N_i to $N_i + q$ if N_i is even or N_i and $N_i - q$ if odd. Considering $mod(w)$ for all the nodes addresses.

2.1.2 $N2R(p, q)$

The number of nodes N in the N2R structure is any positive even integer larger or equal to 6. These of the rings each contain the same number of nodes (*p*). Links in the outer ring connect g node N_i and node $N_{(i+1)(mod(p))}$ and links in the inner ring are interconnecting node N_i and node $N_{(i+q)(mod(p))}$, being $1 \leq q \leq p/2$. To avoid forming two separated networks in the inner ring, *q* must fulfil *gcd(p, q)* = 1 (Greatest Common Divisor), [11]. See Fig. 1(b).

N2R topological routing has been developed by the authors in [12]-[7] and the main conclusion obtained is that it is not possible to implement a generic algorithm that provides optimal paths for all the configurations and with complexity $O(1)$. The problem is when the shortest path involves "several loops" to the network at the inner ring. To be able to optimize the path distances, the complexity of the algorithm increases with the number of elements. However, simpler algorithms provide near optimal solutions and constant complexity just by not considering several loops in the inner ring.

The proposed was named *"Balanced Algorithm (BA)"* which did not obtain the best results in path distances nor path completion time but those values had not a significant difference with the optimal [5].

Link notation:

At a given address N_i of a outer node, to follow the link *L* means to reach the node $N_i + 1$, link *R* means $N_i - 1$ and link *C* means $N_i + p$. In the same way with the inner nodes, to follow the link *L* means to reach the node $N_i + q$, link *R* means $N_i - q$ and link *C* means *X-p* (considering *mod p* for all the neighbours addresses at the same ring).

2.1.3 Double Ring (DR)

It consists of two rings denoted inner and outer rings. These rings each contain the same number of nodes n; hence the total number of nodes is $N = 2 * n$, being $n \geq 2$. The rings are interconnected by links between each corresponding pair of nodes in the inner and outer ring [11]. Fig. 1(c).

Topological routing for DR has not been formally defined in any publication. However, a DR can be considered a $N2R(p, 1)$ and its routing algorithm is rather simple. DR algorithm will illustrate the simplicity of the idea when it is programmed for a specific case (same q). The difference on the complexity respect to the N2R algorithm measures to some extend the cost of the generalization of the algorithm for all configurations (all q).

Link notation:

For a given node N_i to follow the link *L* means to reach the node $N_i + 1$, link *R* means $N_i - 1$ and link *C* means $N_i \pm N/2$ depending o the ring N_i belongs to.

2.2 Other concepts

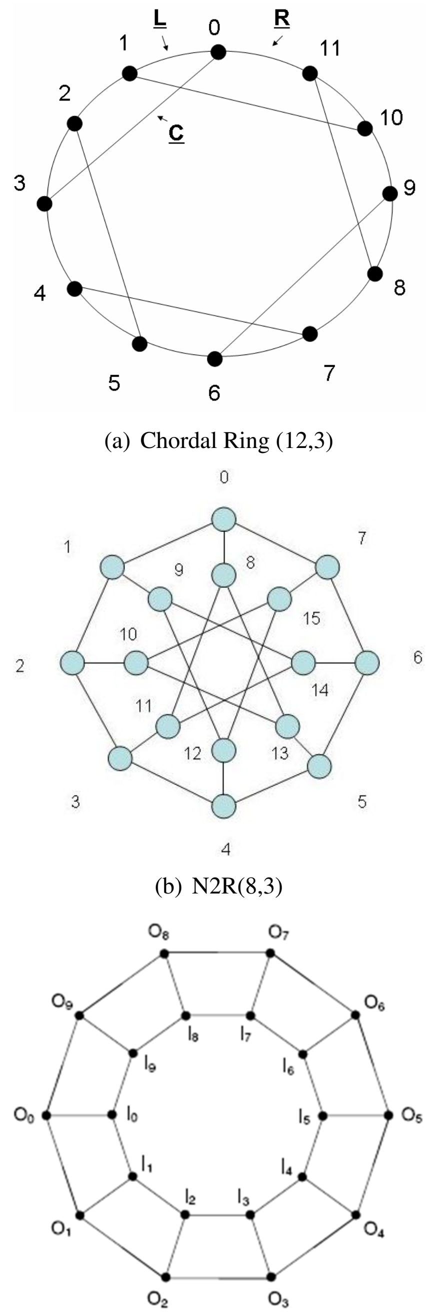
The following list covers some other important concepts of this work.

2.2.1 Shortest Optimal Chord, (SOC)

This term is used along the document and it is related to the "optimal q" for $N2R$ and CR configurations. For the same network size, same w for CR or p for N2R, there might be more than one configuration, several q, which gives the shortest average and maximum path distances. In real networks it is more likely to implement short chord lengths, therefore, SOC is defined as the shortest of the optimal q in terms of path distances.

2.2.2 Complexity in routing

For link state routing approaches, the **Dijkstra** algorithm is used [13]. The complexity of determining a path on a graph with N nodes and m edges is



(a) Chordal Ring (12,3)

(b) N2R(8,3)

(c) Double Ring

Fig. 1. Degree 3 topologies

$O(N^2)$. Dijkstra or its variances are used by real protocols such as Open Shortest Path First (OSPF) and Intermediate System to Intermediate System (IS-IS) for example.

Bellman Ford algorithm is used for distance vector routing approaches [13]. The complexity of determining a path on a graph with V nodes and E edges is $O(VE)$. Examples are Routing Information Protocol (RIP) and Interior Gateway Routing Protocol (IGRP).

$O(logN)$ algorithms such as **Minimum Spanning Tree** or **Sparse Routing Algorithm** give solutions to NP problems in polynomial time to determine the tree from a source to all destinations [13].

Topological routing is characterized for being $O(1)$ by avoiding path calculations. The information of the topology can be codified in simple math statements and the forwarding decisions do not require more than neighboring information [2].

2.2.3 Generic Runtime Measurement

There are five main function classes used on the proposed algorithms: *Variable Assignment* ($A = B$), *Algorithmic* (addition, multiplication), *Evaluation* ($<, >$), *Boolean Logic* (and, or) and *Conditional* (if, else). To generically measure the average runtime of an algorithm the following variables in Table 1 are defined for each class. It is assumed that these time factors include all the tasks required such as memory access or register write/read actions. In this way, if the specific times are known for a specific language or machine, the approximate numerical value can be calculated. This way of runtime measurement intends to keep the analysis fairly basic since it is not the intention of this work to get into the physics of algorithm processing. However, it is enough to move from empirical to analytical approaches and illustrate, more generically, the difference between algorithms.

Type	Time
Variable Assignment	T_1
Algorithmic	T_2
Evaluation	T_3
Boolean logic	T_4
Conditional	T_5

Tab. 1. Instructions types and generic runtime

3 Topological routing

This Section deals with the methodology of the algorithms and their mathematical representation presentation.

3.1 Algorithm guideline

Firstly, the generic guideline to implement this kind of algorithms should be mentioned. The following list summarizes the main concepts:

- **No pre-calculated paths, no routing tables**: There is not path calculation to be performed and the resulting path after the first transmission is not stored.
- **Decide which link to use to forward the packet, "forwarding"**: The algorithm should be focused on the calculation of the best outgoing link to forward the packet to the next hop.
- **Same procedure at every node involved in the path**: A packet arrives to a node, the algorithm calculates the outgoing link and the packet is forwarded. No distinction between nodes.
- **One algorithm valid for all configurations**: To be able to evaluate the performance of an algorithm it should be defined for all the configurations of a certain topology. Specific algorithms for specific cases cannot be properly evaluated.
- **Address scheme**: The operations can be significantly simplified is the correct address or labeling scheme is used. It is a critical decision.
- **Decentralized**: Only information about the neighbors is available.

- **Constant Algorithmic complexity**, $O(1)$: Execution time independent of the number of elements in the network for the same topology.

3.2 Honeycomb Based Algorithm ($HCBA$) for CR

This algorithm can be found at [10]. In a nutshell, the basic idea of the algorithm is to compare potential distances that are calculated based on the graph properties of the topology. This specific algorithm deals with the comparison of two potential distances between any pair of nodes, N_c and N_d for source and destination nodes respectively.

Let D_{qccw} and D_{qcw} be the counterclockwise and clockwise distances path distances and K_{ccw} and K_{cw} the number of Chordal hops in them. The main challenge is to calculate these distances in a simple way since no trees or precalculated paths are used in topological routing. In order to identify the patterns followed by CR, graphs similar to Honeycombs simplify the process.

The following paragraphs are focused on explaining in depth the procedure to construct these graphs. Obviously, this whole procedure is part of the pattern analysis and only the results (distance formulas) are used for the topological routing algorithm implementation, but the resulting graphs are found interesting enough to be explained for further research on CR.

The graphs represent the "*view*" from any node of the rest of the network. There are two types of honeycomb shaped graphs from a node N_c, if it is even or odd. These two graphs can be divided in two subgraphs, clockwise and counterclockwise oriented paths. All nodes can be reached using both kinds of paths, therefore all nodes can be located at both subgraphs. The use of these graphs allows to directly identify formulas for D_{qccw} and D_{qcw} based on D_{occw} and D_{ocw} (counterclockwise and clockwise distances using the Ring, no chords) and the "K" values.

Honeycomb structures can be represented as "*brick format*", similarly to grids schemes, and can be characterized by the number of rows and columns, for more information it is highly recommended to read [14]. The use of this type of representation eases the analytical approach for CR.

To properly define these graphs it is necessary to define the "K" values, K_{ccw} and K_{cw}. They represent the number of Chordal hops in a path and, also, the row in which the destination node is located in the N_c honeycomb shaped graph. The formulas of these values are presented in Table 3.2 for N_c even and odd. Basically, these values can be calculated for any N_d and they are required to calculate D_{qccw} and D_{qcw} to make a forwarding decision. If both K values are calculated for all the nodes from the point of view of N_c, a Honeycomb shaped graph can be formed appearing each of the nodes twice in this graph. One representing a clockwise oriented path and another a counterclockwise path.

Fig. 2 presents the generic approach for this type of graphs for N_i even; for N_i odd it is similar, just to switch the positions in the graph of A - B and K_{ccw} - K_{cw}. A and B correspond to the values $A = \frac{w+1}{2}$ and $B = \frac{w-1}{2}$ and D_{occw} and D_{ocw} are easily calculated following Formula (1). Based on these graphs, the patterns for the distances values can be identified.

$$
\begin{aligned}
D_{occ} &= N_c - N_d(mod(w)) \\
D_{oc} &= N_d - N_c(mod(w))
\end{aligned}
\tag{1}
$$

Fig. 3 presents an example of the complete graph from node "0". It can be clearly identified how each node is present twice except for 0. This kind of representation will ease future work in CR such as failure supporting schemes.

The result of this graph analysis leads to the formulas implemented in the topological routing algorithm in order to make the forwarding decision. Formula (2) for counterclockwise and Formula (3) for clockwise oriented paths; the extra three variables a, b and c are defined in Formula (4).

The final step is to determine the link to forward the packet based on the shortest of D_{qccw} and D_{qcw}.

3.3 Balanced Algorithm (BA) for $N2R$

This algorithm can be found at [5]. The basic idea is similar to the $HCBA$ algorithm, to find the distance

N_c	$K_{ccw}(q>3)$	$K_{ccw}(q=3)$	$K_{cw} < \frac{q}{2}$	$K_{cw} > \frac{q}{2}$
Even	$\lfloor \frac{D_{occw}-A-1}{q} + 1 \rfloor$	$\lfloor \frac{D_{occw}+1}{q+1} \rfloor$	$\lfloor \frac{D_{ocw}-B-1}{q} + 1 \rfloor$	$\lfloor \frac{D_{ocw}+q-1}{q+1} \rfloor$
N_c	$K_{cw}(q>3)$	$K_{cw}(q=3)$	$K_{ccw} < \frac{q}{2}$	$K_{ccw} > \frac{q}{2}$
Odd	$\lfloor \frac{D_{ocw}-A-1}{q} + 1 \rfloor$	$\lfloor \frac{D_{ocw}+1}{q+1} \rfloor$	$\lfloor \frac{D_{occw}-B-1}{q} + 1 \rfloor$	$\lfloor \frac{D_{occw}+q-1}{q+1} \rfloor$

Tab. 2. K values

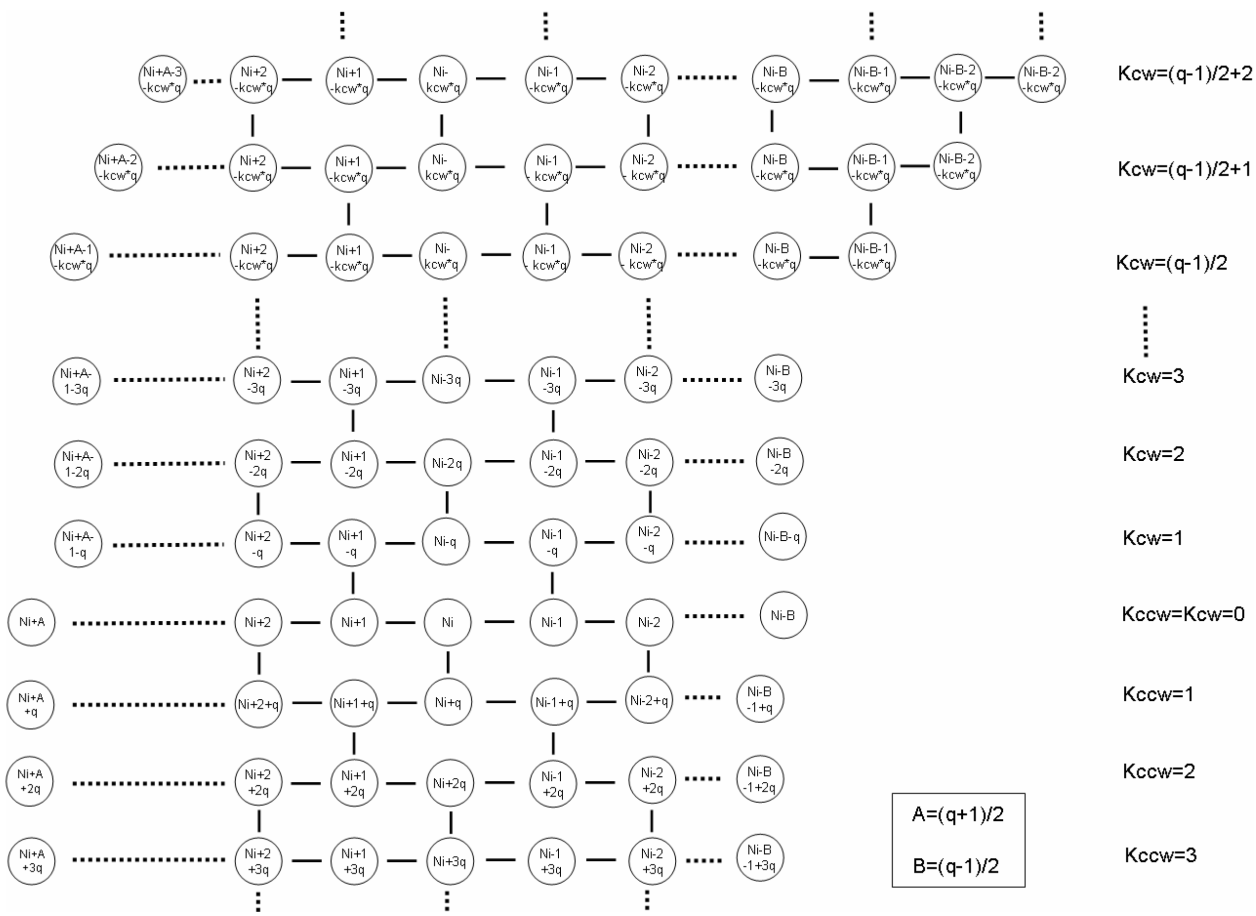


Fig. 2. Graph for a node N_i (even)

$$D_{qccw} = \begin{cases} K_{ccw}(w-1) - D_{occw} + 2K_{ccw} & \text{if } K_{ccw}(w-1) - a > D_{occw} \ \& \ q > 3 \\ D_{occw} - K_{ccw}(w+1) + 2a + 2K_{ccw} & \text{if } K_{ccw}(w+1) + a < D_{occw} \ \& \ q > 3 \\ D_{occw} - K_{ccw}(w-1) & \text{if } q = 3 \ \& \ N_c \text{ even} \\ 2K_{ccw} - c & \text{rest} \end{cases} \tag{2}$$

$$D_{qcw} = \begin{cases} K_{cw}(w-1) - D_{ocw} + 2K_{cw} & \text{if } K_{cw}(w-1) - b > D_{ocw} \ \&\ q > 3 \\ D_{ocw} - K_{cw}(w+1) + 2b + 2K_{cw} & \text{if } K_{cw}(w+1) + b < D_{ocw} \ \&\ q > 3 \\ D_{ocw} - K_{cw}(w-1) & \text{if } q = 3 \ \&\ N_c \text{ odd} \\ 2K_{cw} + c & \text{rest} \end{cases} \quad (3)$$

$$a = \begin{cases} 1 & \text{if } N_c \text{ odd} \\ 0 & \text{if } N_c \text{ even} \end{cases} b = \begin{cases} 0 & \text{if } N_c \text{ odd} \\ 1 & \text{if } N_c \text{ even} \end{cases} c = \begin{cases} 1 & \text{if } N_c \text{ even} \ \&\ N_d \text{ odd} \\ -1 & \text{if } N_c \text{ odd} \ \&\ N_d \text{ even} \\ 0 & \text{rest} \end{cases} \quad (4)$$

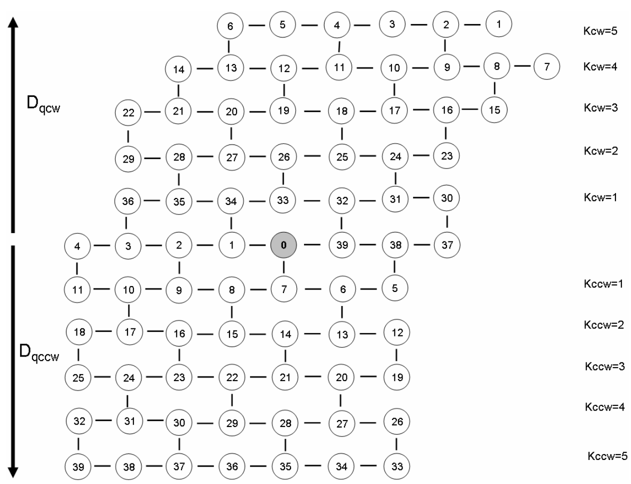


Fig. 3. Example of the graph from node 0 in CR(40,7)

values of the different ways of establishing transmissions. In this case there are three options: using the outer ring, the inner ring clockwise and counter-clockwise.

The first step is to determine the relative position of source N_s and destination N_d and some conversions are required to properly work with the addresses. The following operations, Formulas (5) and (6), are required to obtain a result in the necessary range[1].

$$N'_d = \begin{cases} N_d & \text{if } N_s < p \ \&\ N_d < p \text{ (O - O)} \\ N_d & \text{if } N_s \geq p \ \&\ N_p \geq p \text{ (I - I)} \\ N_d - p & \text{if } N_s < p \ \&\ N_d \geq p \text{ (O - I)} \\ N_d + p & \text{if } N_s \geq p \ \&\ N_d < p \text{ (I - O)} \end{cases} \quad (5)$$

$$N''_d = \begin{cases} N'_d & \text{if } N'_d > N_s \\ N'_d + p & \text{if } N'_d < N_s \end{cases} \quad (6)$$

The next step is to calculate the distance using the Outer Ring DT_{out}, Formula (10). D'_{out} corresponds to the hops taken at the Outer Ring, Formulas (7) and (8). D_{rout} are the hops in between rings in case that are necessary, Formula (9).

$$D_{out} = N''_d - N_s \quad (7)$$

$$D'_{out} = \begin{cases} p - D_{out} & \text{if } D_{out} > p/2 \\ D_{out} & \text{if } D_{out} \leq p/2 \end{cases} \quad (8)$$

$$D_{rout} = \begin{cases} 0 & \text{if } N_S < p \ \&\ N_D < p \text{ (O - O)} \\ 1 & \text{if } N_S \geq p \ \&\ N_D < p \text{ (I - O)} \\ 1 & \text{if } N_S < p \ \&\ N_D \geq p \text{ (O - I)} \\ 2 & \text{if } N_S \geq p \ \&\ N_D \geq p \text{ (I - I)} \end{cases} \quad (9)$$

[1]O=Outer Ring, I=Inner Ring

$$DT_{out} = D'_{out} + D_{rout} \quad (10)$$

The next step is to calculate DT_{in1} and DT_{in2}, distances using the inner ring (or both). The following paragraphs describe how to determine DT_{in1} being D_{in1} the hops at the inner ring to establish a connection, Formula (11).

$$D_{in1} = [D'_{out}/q] \quad (11)$$

In addition, the path can be formed by hops between rings (D_{rin1}) Formula (12) and hops at the outer ring (Dq_{out1}).

$$D_{rin1} \begin{cases} 2 & \text{if } N_s < p \;\&\; N_d < p \text{ (O - O)} \\ 1 & \text{if } N_s \geq p \;\&\; N_d < p \text{ (I - O)} \\ 1 & \text{if } N_s < p \;\&\; N_d \geq p \text{ (O - I)} \\ 0 & \text{if } N_s \geq p \;\&\; N_d \geq p \;\&\; \frac{D'_{out}}{q} \in \mathbb{I} \text{ (I - I)} \\ 2 & \text{if } N_s \geq p \;\&\; N_d \geq p \;\&\; \frac{D'_{out}}{q} \notin \mathbb{I} \text{ (I - I)} \end{cases} \quad (12)$$

Dq_{out1} is the distance from the node where the transmission jumps from the inner ring to the outer ring (N'_{q1}), to the destination node. This value is calculated as D_{out} by Formula (7) and implying the same necessary conversions. The value of N_{q1} is calculated by Formula (13), "+" for left oriented and "-" for right oriented:

$$N_{q1} = N_s \pm D_{in1} * q \quad (13)$$

Similarly to Formula (7) the values must be given in the proper range. Therefore, assuming that $0 \leq N_s < p$ (value converted if necessary) there are two conversions depending on the orientation, see Formula (14):

$$N'_{q1} = \begin{cases} N_{q1} - p & \text{if } N_{q1} \geq p \text{ (Left orientated)} \\ N_{q1} + p & \text{if } N_{q1} < 0 \text{ (Right orientated)} \end{cases} \quad (14)$$

In this way the values of N'_{q1} are in the proper range to work with.

Finally, there are two possibilities for (DT_{in1}), using the inner ring or both, it is explained at Formula (15):

- If D'_{out}/q is an integer, the transmission will only need to use the inner ring.
- If D'_{out}/q is not an integer, the transmission will use both of the rings.

$$DT_{in1} = \begin{cases} D_{in1} + D_{rin1} & \text{if } \frac{D'_{out}}{q} \in \mathbb{I} \\ D_{in1} + D_{rin1} + Dq_{out1} & \text{if } \frac{D'_{out}}{q} \notin \mathbb{I} \end{cases} \quad (15)$$

For DT_{in2} the procedure is similar just considering Formula (16). The rest of the operations are omitted to avoid repetition.

$$D_{in2} = [(p - D'_{out})/q] \quad (16)$$

Finally, the comparison between the three distances, DT_{out}, DT_{in1} and DT_{in2}, gives the outgoing link to forward the packet by choosing the lowest option.

3.4 DR algorithm

The best way to illustrate the algorithm is with the following pseudo code. Let N_s be any source node and N_d any destination node. N is the number of nodes.

if ($N_s < \frac{N}{2}$ & $N_d < \frac{N}{2}$ or $N_s \geq \frac{N}{2}$ & $N_d \geq \frac{N}{2}$) **then**
  if ($abs(N_s - N_d) \leq \frac{N}{4}$ & $N_s > N_d$ or $abs(N_s - N_d) > \frac{N}{4}$ & $N_s < N_d$) **then**
    $nexthop = R$
  else
    $nexthop = L$
else
  $nexthop = C$

4 Results

This Section presents the results for the path distance and instruction analysis. Briefly, the first result to illustrate is the average distance and diameter comparison for the three topologies. Fig. 4 presents the path distance results, for CR and $N2R$ the values correspond to the SOC option. These results are not new, detailed information about the CR algorithm performance can be found in [10] and for the $N2R$ at [5]. For the experiments, transmissions for all the combinations of pairs of nodes and all the possible configurations from 6 to 100 nodes are performed. In transmission networks, shortest path distances are desirable since they keep the overall traffic of the network as low as possible by minimizing the routing traffic at the nodes, requiring lower capacity to support the demands.

The novel contribution for this Section is the results on complexity analysis based on the generic instructions runtime. The following concepts must be introduced in order to properly understand the representation of the algorithms:

- Instructions: As a reminder Table 1 in Section 2 presents the types and runtime for reach basic instruction.

- There are some operations in the algorithms that are not strictly part of the five defined instruction types and need some explanation of how they are measured. These are the following:

  - **Check if N_i is even**: It is assumed as one Evaluation operation T_3, check if last bit is equal to 0

  - **Integer value of X**: It is considered as one Algorithmic an one Variable assignment, $T1 + T2$.

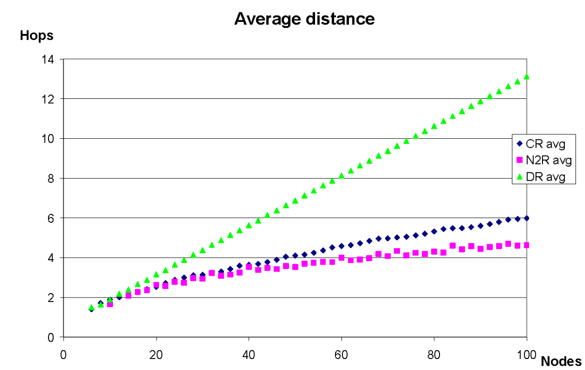
  - **Absolute value**: It is treated as the combination in Formula (17). The total runtime value is $T_1 + T_2 + T_3 + T_5$.

$$\begin{aligned} & A = abs(b-c) \\ & if\, b > c \;\; A = b - c \\ & else \;\; A = c - b \end{aligned} \tag{17}$$

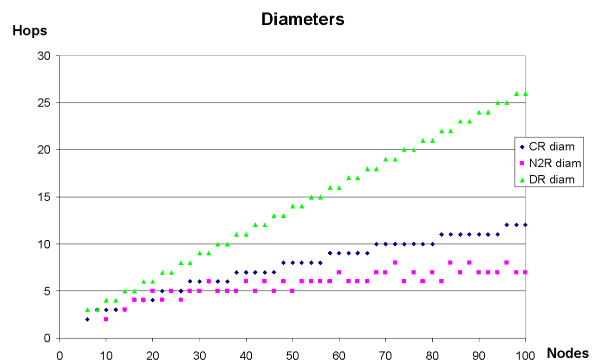
- The algorithms are represented from the point when the destination address is passed to the algorithm up to when the outgoing link value is set.

- Flow charts: There are two kinds of flow charts, generic and time presentation.

- The generic type has the three types of boxes. “Code” boxes simply refer to some code and instructions to execute, “Statement” boxes are



(a)



(b)

Fig. 4. Average and Maximum distance for the three algorithms

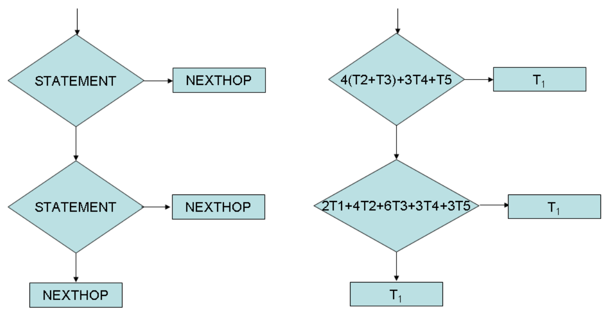


Fig. 5. DR algorithm flow and time distribution

conditional operations (if) and "Nexthop" is the value assignment of the outgoing link. Any brach from any topological routing algorithm has to end with this type box.

Fig. 6 illustrates the generic and time presentation flow charts for the Double Ring algorithm. The diagram is kept very schematic since it is meant to be used to calculate the average theoretical runtime and not the exact function of each block. The definition of the algorithms has been deeply treated in Section 3.

Figs. 6 and 7 present the flow charts for CR and $N2R$. For the $N2R$ case, Fig. 7, a generic "Block A" is introduced to simplify the presentation. This block basically deals with format conversion of variables.

A a result from the analysis, Table 4 presents the runtime from the flow charts analysis. The time is given as a strictly mathematical average following Formula 18, where T_i is the time of each of the flow branches and B is the number of branches. For specific number of nodes or sources or destination some branches could be used more than others.

$$T = \frac{1}{B} \sum T_i \tag{18}$$

The main conclusion extracted from Table 4 is the proof of the constant complexity of the algorithms. The runtime of the three algorithms is not a function of the number of nodes N.

5 Conclusion

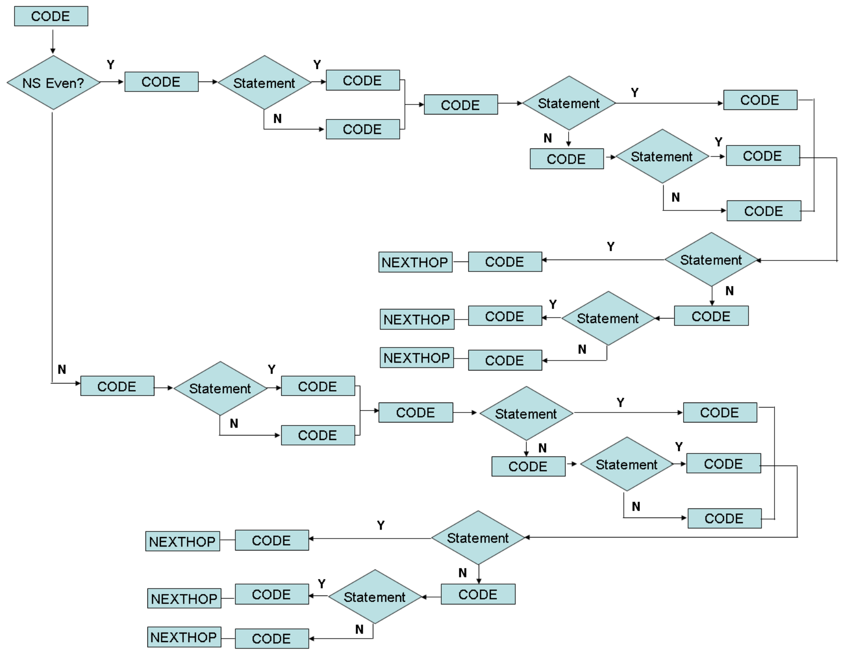
Topological routing has been presented for three very well known degree 3 topologies. The selected algorithms are not optimal in terms of path distances but they provide reasonable pseudo optimal solutions under a constant complexity $O(1)$.

Furthermore, a novel procedure has been presented in order to study the runtime of the algorithms. This analytical approach gives more generic conclusions than the previous empirical results analysis. In this way, the comparison of algorithms or the performance prediction can be calculated for any specific hardware or software.

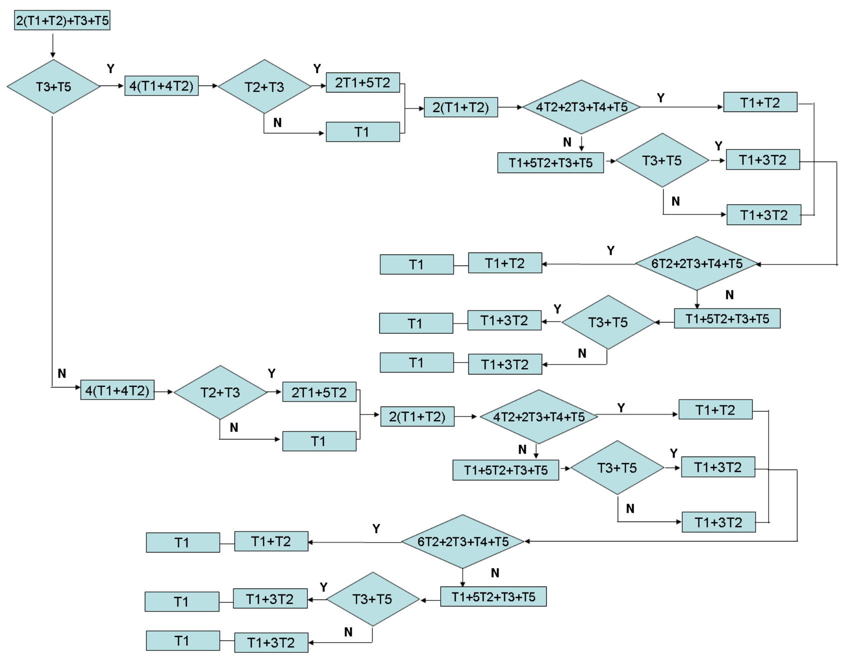
As an addition, the basic guidelines for topological routing implementation have been formally stated.

6 Appendix Notation

- Topologies: CR Chordal Ring, DR Double Ring and $N2R$.
- Nodes: N number of nodes, N_i generic node address, N_c source node, N_d destination node.
- Links: R Right, L Left, C Center.
- Runtime variables:Variable Assignment T_1, Algorithmic T_2, Evaluation T_3, Boolean Logic T_4 and Conditional T_5.
- Algorithms: $HCBA$ Honeycomb Based algorithm, BA Balanced Algorithm.
- Variables from $HCBA$: D_{qccw} and D_{qcw} counterclockwise and clockwise distances; K_{qccw} and K_{qccw} chordal hops for counterclockwise and clockwise paths.
- Variables from BA: O Outer Ring and I Inner Ring. DT_{out} distance using O, DT_{in1} and DT_{in2} distances using I. Auxiliary variables: D_{rout}, D_{out}, D'_{out}, D_{rin1}, D_{rin2}, D_{in1}, D_{in2}, Dq_{out1}, Dq_{out1}, N_{q1}, N'_{q1}, N'_d and N''_d.

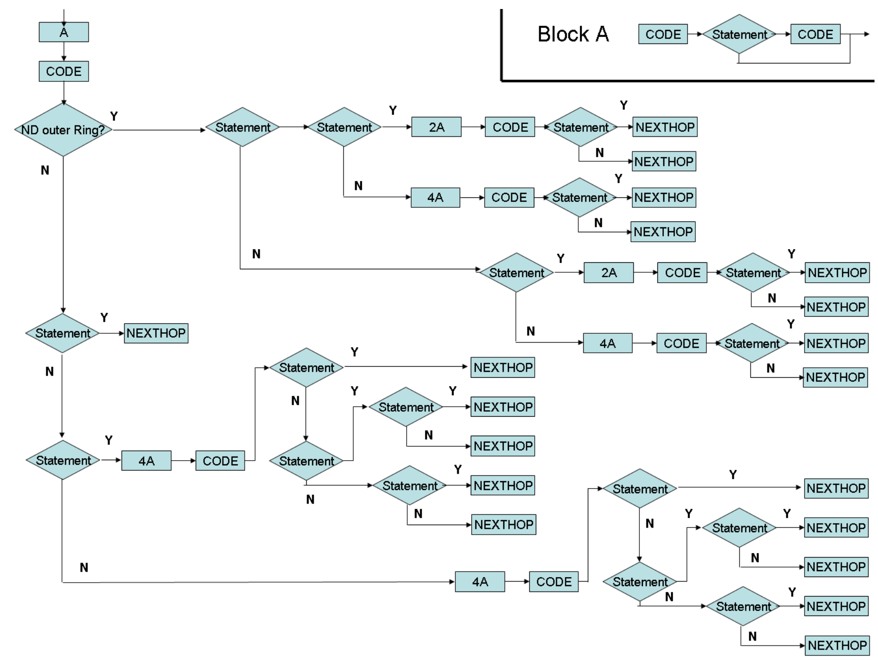


(a)

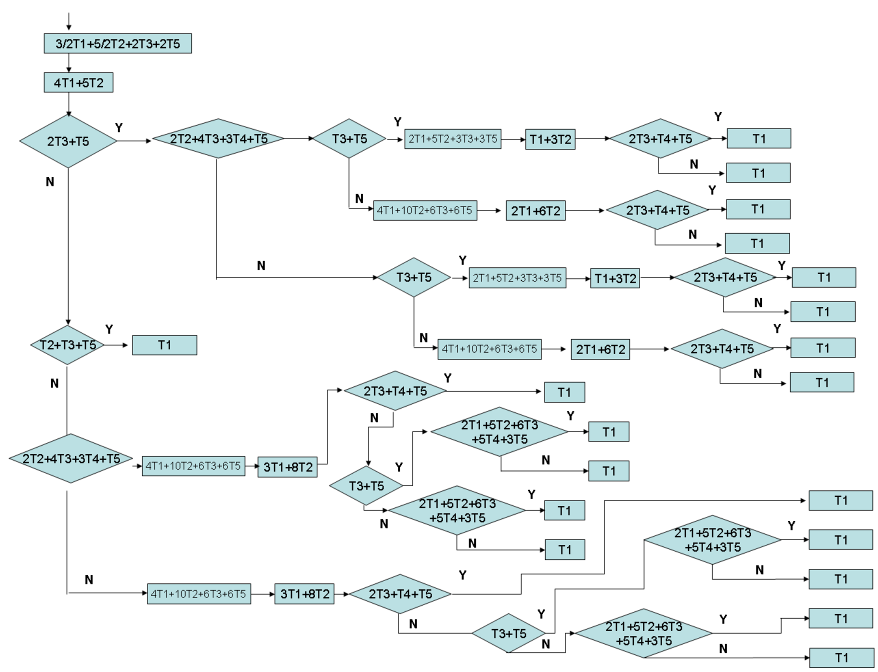


(b)

Fig. 6. CR algorithm flow charts and time distribution



(a)



(b)

Fig. 7. N2R algorithm flow chart and time distribution

Algorithm	Average Time
DR	$1,33T_1 + 2,66T_2 + 3,33T_3 + 2T_4 + 1,33T_5$
BA	$9,25T_1 + 20,125T_2 + 14,125T_3 + 5,875T_4 + 9,25T_5$
HCBA	$13,5T_1 + 42,5T_2 + 9T_3 + 2T_4 + 7T_5$

Tab. 3. Algorithms' Time Presentation

References

[1] W. D. Grover, *Mesh-Based Survivable Networks, Options and Strategies for Optical, MPLS, SONET and ATM Network*, Prentice Hall PTR, 2003.

[2] Jens Myrup Pedersen, Thomas Phillip Knudsen, Ole Brun Madsen. *Topological Routing in Large-Scale Networks*. Proceedings of IEEE/ICACT 2004.

[3] Chip Martel and Van Nguyen.*Analyzing Kleinberg's (and other) small-world Models*, Annual ACM Symposium on Principles of Distributed Computing archive Proceedings of the twenty-third annual ACM symposium on Principles of distributed computing. St. John's, Newfoundland, Canada, 2004. Pages: 179 - 188 ISBN:1-58113-802-4

[4] D. Watts and S. Strogatz. *Collective dynamics of small-world networks*. Nature, 393:440442, 1998.

[5] Jose M. Gutierrez, Ruben Cuevas, Jens M. Pedersen and Ole B. Madsen *Improving Topological Routing in N2R Networks*. CAAN07, August 2007, Canada.

[6] Jose M. Gutierrez, Jens M. Pedersen , Ruben Cuevas and Ole B. Madsen *Protection mechanism for the N2R Topological Routing Algorithm* IEEE HPSR 08, China

[7] Jose M. Gutierrez, Jens M. Pedersen , Ruben Cuevas and Ole B. Madsen *Restoration mechanism for the N2R Topological Routing Algorithm*ICN 2008 Mexico

[8] S. Bujnowski, B. Dubalski, A. Zabludowski *The evaluation of transmission ability of 3rd degree chordal rings with the use of adjacent matrix*. 7th INFORMS 2004

[9] L. Barrierre, J. Fabrega, E. Simo, M. Zaragoza. *Fault-Tolerant Routings in chordal Ring Networks*. Networks 2000, vol. 36, pp. 180-190. ISSN 0028-3045.

[10] J. Gutierrez, T. Riaz, J. Pedersen, S. Labeaga and O. Madsen. *On topological routing on degree 3 chordal rings* IPC 2009, Poland.

[11] T. Jorgensen, L.Pedersen and J.M. Pedersen.*Reliability in single, double and N2R ring network structures* The International Conference on Communications in Computing (CIC'05) Las Vegas, Nevada, United States, June 2005 .

[12] Jens M. Pedersen, Muhammad Tahir Riaz, Ole B. Madsen *A Simple, Efficient Routing Scheme for N2R Network Structures*, IT&T Annual Conference 2005 - Cork, Ireland .

[13] James F. Kurose and Keith W. Ross. *Computer networking, a top-dowm approach*, ISBN: 0-13-607967-9.

[14] Ivan Stojmenavie, *Honeycomb Netwoks: Topological Properties and Communication Algorithm*, IEEE Transaction on Parallel and Distributed System, Vol8, NO. IO, October 1997.