

Optimal Routing Algorithm in Multilayer Octagon-Cell: A New Class of Octagon-Cell Interconnected Networks

¹Sanjukta Mohanty and ²Prafulla Ku. Behera

¹North Orissa University, Sriram Chandra Vihar, Takatpur, Baripada, Odisha, India

²Utkal University, Vani Vihar, Bhubaneswar, Odisha, India

sanjuktamohanty2005@rediffmail.com, p_behera@hotmail.com

ABSTRACT

In communication system of network routing algorithm acts an important role. The efficiency of a parallel system depends on the reliable and efficient routing algorithm which is used to route the messages between the fault-free nodes. In this paper a new class of interconnection network of Octagon-Cell is introduced, which is called Multilayer Octagon-Cell (MLO). The structure of Multilayer Octagon-cell is recursive in nature. It can be expanded, if we increase the depth of MLO. This paper introduces the node degree, diameter, number of links, bisection width of the MLO network and we have also developed the optimal routing algorithm of MLO.

Keywords: Multilayer Octagon-Cell; Octagon-Cell; Interconnection topology; Routing; Routing algorithm; Network services.

1 Introduction

In parallel system, the efficiency of communication system depends on the routing algorithm adopted in the system. Parallel processing system is used in many areas such as image processing and scientific computing and many problems can be solved in these areas using parallel processing elements [1-7]. The degree of a node in an interconnection network is related to the cost of hardware and message passing time is related to diameter of the network [16]. In an interconnection network, throughput is increased if the degree of network is decreased. Some features in the network of parallel machines are highly desirable such as minimal communication cost, efficient routing and the capability of topological structures [3]. The underlying communication network and the matching of the algorithm with the network structure strongly affect the performance of parallel machines. In this paper we have described a new topological structure called MLO with its optimal routing algorithm.

2 Description of Octagon-Cell Network

An interconnection network can be viewed as an undirected graph, in which vertices correspond to processors and edges correspond to the bidirectional communication links between processing elements [3]. An octagon-cell has eight nodes. It has d levels numbered from 1 to d with depth d . Level 1 represents one octagon-cell. Level 2 represents eight octagon-cells surrounding the octagon-cell at level 1. Level 3 represents 16 octagon-cells surrounding the 8 octagon-cells at level 2 and so on.

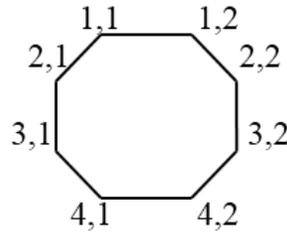


Figure 1: Addressing nodes in Octagon-Cell with level: 1
 (X,Y represents line no-X with node no-Y)

Each level i has N_i nodes, representing processing elements and interconnected in a ring structure. In an octagon-cell network, the number of nodes at level i is: $N_i = 8(4i-3)$

Now at level 1, $N_1 = 8$, since there is a single octagon-cell with 8 vertices. Level 2 introduces 8 octagon-cells. Therefore at level 2 the number of nodes

$$N_2 = 8(4*2-3) = 8*5 = 40, N_3 = 8(4*3-3) = 8*9 = 72$$

In octagon-cell the level $(i+1)$ has 32 nodes in addition to corresponding nodes to those at level i . Therefore

$$N_i = 8 + (i-1)*32 = 8+32*i-32 = 32*i-24 = 8(4*i-3)$$

The total number of nodes in an octagon-cell network is,

$$\begin{aligned} N &= \sum 8(4i - 3) = 32\sum i - \sum 24 = 32\sum_{i=1}^d i - 24\sum 1 \\ &= 32d(d+1)/2 - 24d = 16d^2 - 8d = 8d(2d-1) \end{aligned}$$

Or we can write $N = 8i(2i-1)$,

$$\text{Now } N = 16d^2 - 8d \text{ or } 16d^2 = N + 8d \text{ or } d^2 = N + 8d/16 \text{ or } d = 1/4 \sqrt{(N + 8d)}$$

Therefore the total no of nodes at level 1 is $N = 8(2*1-1) = 8$

At level 2, $N = 8(2*4-2) = 48$

At level 3, $N = 8(2*9-3) = 120$ and so on.

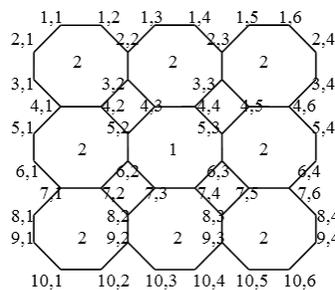


Figure 2: Addressing nodes in Octagon-Cell with level:2

3 Multilayer Octagon-Cell Interconnection Network

3.1 Number of Nodes

Multilayer Octagon-Cell (MLO) is a modular interconnection network which consists of layers of identical Octagon-Cell networks connected together in a hierarchical order. The MLO is represented as MLO (k, d), where k denotes the layer number and d denotes the depth of the identical Octagon-Cell. Each Layer k has N_k nodes, where $N_k = 8kd (2d-1)$.

3.2 Node Degree

The node degree on an interconnection network is defined as the maximum number of edges that a node can have in the network [3]. The node degree of MLO with depth 1 is 3. If $d > 1$, then node degree remains constant, that is 4. The network topology which secures constant node degree is highly desirable. Constant node degree facilitates modularity in building blocks for scalable systems [3,9-11]. Therefore the node degree of MLO is constant when $d > 1$.

3.3 Diameter

The diameter (D) of a network is defined as the maximum shortest path between any two nodes [3,9-11]. The path length is measured by the number of links traversed. The network diameter indicates the maximum number of distinct hops between any two nodes. The network diameter should be as small as possible.

The diameter is given by $4(2d-1) + k$.

3.4 Number of Nodes:

The total number of nodes in octagon-cell network is given by the formula: $8d (2d-1)$

So the total number of nodes in MLO is given by $8kd (2d-1)$.

3.5 Bisection width:

When we cut a network into two equal halves, the minimum number of edges or channels along that cut is called the channel bisection width [3,9-11]. Here we have denoted it as 'b'. The bisection width provides a good indicator of the maximum communication bandwidth along the bisection of a network. In octagon-cell network the bisection width is $2d$. Therefore in MLO it is $k (2d)$.

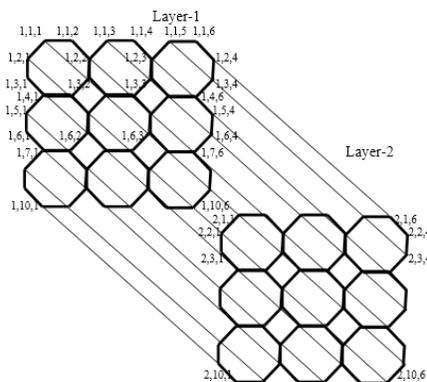


Figure 3: Multilayer Octagon-Cell network with two layers and two levels

4 Proposed Routing Algorithm in MLO

Information can be transmitted over a network from source to destination through a sequence of intermediate switching / buffering stations or nodes by a function called routing. Routing is necessary because in real systems not all nodes are directly connected. Routing algorithms can be classified as static or dynamic, and centralized or distributed.

In this paper we have developed optimal point-to-point routing algorithm for multi layer octagon-cell networks. An optimal routing algorithm is any network would route information or message from a source to a destination along a shortest path. The routing algorithm can be centralized or distributed. In centralized routing, a single processing element determines the shortest path from a given source node to a given destination node. Then the message is sent along that path. In distributed routing however, all intermediate nodes on the shortest path cooperate to find the shortest path using destination address. Therefore each intermediate node needs only the destination address to determine which neighbor falls on the shortest path to a given destination. In the routing algorithm each intermediate node determines in constant time, which of its neighbors will receive the message.

The routing of message can be viewed as a sequence of changes made on the source address label to become the destination address label. These changes are done at every intermediate node on the path [3]. In MLO network when the message is received by an intermediate node, it will consider itself as a new source. Here we have used the layer and line numbering scheme as follows. Each layer in the MLO is identified by (K, X, Y) , where k denotes the layer number in which the node exists, and Y denotes the location of the node in the as shown above in fig-3.

We have developed a recursive routing algorithm for MLO network. Due to its recursive structure routing can be done easily. In our algorithm we have used level numbering scheme. That is a node in MLO with address $(1,1,1)$ is the first node that exists at layer 1 and line number 1. The node $(1,1,2)$ refers to the second node that exists at layer 1 and line number 1 and so on.

4.1 Notations used in the algorithm

- X_s Represents the address of source node in the horizontal direction / Line number of source node in horizontal direction.
- Y_s Represents the address of source node in the vertical direction / Serial number of the source node in X_s line.
- K_s Represents the address of source layer.
- X_d Represents the address of destination node in the horizontal direction / Line number of destination node in horizontal direction.
- Y_d Represents the address of destination node in the vertical direction / Serial number of destination node in X_d line.
- K_d Represents the address of destination layer.
- $X_s \bmod 3$ is assigned by a value which is the remainder after dividing X_s by 3.
- Arrows in our figures represent the destination point.

- Darken circles in our figures represent the source point.
- m** Represents the line number of the octagon_cell network in horizontal direction.
- &&** Represents logical AND
- ||** Represents logical OR

Note: In the following algorithms, we have taken only integral parts of the quotients by neglecting the fractional parts where we have done division by using ‘/’ (division symbol)

NOTE: For each and every case the relation between K_s and K_d may be:

- i) $K_s = K_d$, This is the case of move in same layer
- ii) $K_s > K_d$, Here to reach at destination layer, the recursive function is “Move ($K_s-1, X_s, Y_s, K_d, X_d, Y_d$) “
- iii) $K_s < K_d$, Here to reach at destination layer, the recursive function is “Move ($K_s+1, X_s, Y_s, K_d, X_d, Y_d$) “

Before going to algorithm the reader should check the following conditions for source layer and destination layer K_s, K_d respectively

If ($k_s = k_d$)

Move ($k_s, x_s, y_s, k_d, x_d, y_d$)

Else if ($k_s > k_d$)

Move ($k_s-1, x_s, y_s, k_d, x_d, y_d$)

Else

Move ($k_s+1, x_s, y_s, k_d, x_d, y_d$)

Case-1(a) Optimal Routing Algorithm for Horizontal Move for lines m where $m \bmod 3 = 1$ && $X < [(d*5) + (d-2)]$ [Move from left to right, if ($X_s = X_d$ && $Y_s < Y_d$) and Move from right to left, if ($X_s = X_d$ && $Y_s > Y_d$)]

Move ($K_s, X_s, Y_s, K_d, X_d, Y_d$)

If ($y_s < y_d$)

If ($x_s = x_d$ && y_s is odd)

Move ($k_s, x_s, y_s + 1, k_d, x_d, y_d$)

Else If ($x_s = x_d$ && y_s is even)

Move ($k_s, x_s + 1, y_s/2 + 1, k_d, x_d, y_d$)

Else If ($x_s \neq x_d$)

Move ($k_s, x_s - 1, 2y_s - 1, k_d, x_d, y_d$)

Else

Destination reached

Else

If ($y_s > y_d$)

If ($x_s = x_d$ && y_s is odd)

Move ($k_s, x_s+1, y_s/2+1, k_d, x_d, y_d$)

Else If ($x_s = x_d$ && y_s is even)

Move ($k_s, x_s, y_s - 1, k_d, x_d, y_d$)

Else If ($x_s \neq x_d$)

Move ($k_s, x_s-1, 2y_s-2, k_d, x_d, y_d$)

Else

If ($y_s = y_d$)

Go to Vertical Move

Else

Destination reached

Case-1(b) Optimal Routing Algorithm for Horizontal Move for lines m where $m \bmod 3 = 1$ && $X = [(d*5) + (d-2)]$ [Move from left to right, if ($X_s = X_d$ && $Y_s < Y_d$) and Move from right to left, if ($X_s = X_d$ && $Y_s > Y_d$)]

Move ($K_s, X_s, Y_s, K_d, X_d, Y_d$)

If ($y_s < y_d$)

If ($x_s = x_d$ && y_s is odd)

Move ($k_s, x_s, y_s + 1, k_d, x_d, y_d$)

Else If ($x_s = x_d$ && y_s is even)

Move ($k_s, x_s - 1, y_s/2 + 1, k_d, x_d, y_d$)

Else If ($x_s \neq x_d$)

Move ($k_s, x_s + 1, 2y_s - 1, k_d, x_d, y_d$)

Else

Destination reached

Else

If ($y_s > y_d$)

If ($x_s = x_d$ && y_s is odd)

Move ($k_s, x_s - 1, y_s/2 + 1, k_d, x_d, y_d$)

Else If ($x_s = x_d$ && y_s is even)

Move ($k_s, x_s, y_s - 1, k_d, x_d, y_d$)

Else If ($x_s \neq x_d$)

Move ($k_s, x_s + 1, 2y_s - 2, k_d, x_d, y_d$)

Else

If ($y_s = y_d$)

Go to Vertical Move

Else

Destination reached

Example 1.1 Let $(K_s, X_s, Y_s) = (1, 1, 3)$ and $(K_d, X_d, Y_d) = (1, 1, 8)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

$1, 1, 3 \rightarrow 1, 1, 4 \rightarrow 1, 2, 3 \rightarrow 1, 1, 5 \rightarrow 1, 1, 6 \rightarrow 1, 2, 4 \rightarrow 1, 1, 7 \rightarrow 1, 1, 8$

The shortest path length is 7

Case-2 Optimal Routing Algorithm for Horizontal Move for lines m where $m \bmod 3 \neq 1$. [Move from left to right, If $(X_s = X_d \ \&\& \ Y_s < Y_d)$ and Move from right to left, if $(X_s = X_d \ \&\& \ Y_s > Y_d)$

Move ($K_s, X_s, Y_s, K_d, X_d, Y_d$)

If ($x_s = x_d \ \&\& \ y_s < y_d \ \&\& \ x_s \bmod 3 = 2$)

If ($x_s = x_d \ \&\& \ x_s \bmod 3 = 2$)

Move ($k_s, x_s - 1, 2y_s - 1, k_d, x_d, y_d$)

Else If ($x_s \neq x_d \ \&\& \ y_s$ is odd)

Move ($k_s, x_s, y_s + 1, k_d, x_d, y_d$)

Else If ($x_s \neq x_d \ \&\& \ y_s$ is even)

Move ($k_s, x_s + 1, y_s/2 + 1, k_d, x_d, y_d$)

Else

Destination reached

Else

If ($x_s = x_d \ \&\& \ y_s < y_d \ \&\& \ x_s = 3n$)

If ($x_s = x_d \ \&\& \ x_s = 3n$)

Move ($k_s, x_s + 1, 2y_s - 1, k_d, x_d, y_d$)

Else If ($x_s \neq x_d \ \&\& \ y_s$ is odd)

Move ($k_s, x_s, y_s + 1, k_d, x_d, y_d$)

Else If ($x_s \neq x_d \ \&\& \ y_s$ is even)

Move ($(k_s, x_s - 1, y_s/2 + 1, k_d, x_d, y_d)$)

Else

Destination reached

Else

If ($x_s = x_d$ && $y_s > y_d$ && $x_s \bmod 3 = 2$)

If ($x_s = x_d$ && $x_s \bmod 3 = 2$)

Move ($k_s, x_s-1, 2y_s-2, k_d, x_d, y_d$)

Else If ($x_s \neq x_d$ && y_s is even)

Move ($k_s, x_s, y_s - 1, k_d, x_d, y_d$)

Else If ($x_s \neq x_d$ && y_s is odd)

Move ($k_s, x_s + 1, y_s/2+ 1, k_d, x_d, y_d$)

Else

Destination reached

Else

If ($x_s = x_d$ && $y_s > y_d$ && $x_s = 3n$)

If ($x_s = x_d$ && $x_s = 3n$)

Move ($k_s, x_s+1, 2y_s-2, k_d, x_d, y_d$)

Else If ($x_s \neq x_d$ && y_s is even)

Move ($x_s, y_s - 1, x_d, y_d$)

Else If ($x_s \neq x_d$ && y_s is odd)

Move ($k_s, x_s - 1, y_s/2+ 1, k_d, x_d, y_d$)

Else

If ($y_s = y_d$)

Go to Vertical Move

Else

Destination reached

Example 2.1 Let $(K_s, X_s, Y_s) = (1, 2, 3)$ and $(K_d, X_d, Y_d) = (2, 2, 8)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

1,2,3 → 2,2,3 → 2,1,5 → 2,1,6 → 2,2,4 → 2,1,7 → 2,1,8 → 2,2,5 → 2,1,9 → 2,1,10 → 2,2,6 → 2,1,11 → 2,1,12 → 2,2,7 → 2,1,13 → 2,1,14 → 2,2,8

The shortest path length is 15

NOTE: In these algorithms first we check the main condition associated with If statement with the given source nodes and destination nodes. The calculated intermediate nodes x_s, y_s may violate the main condition. In example 2.1, given $y_s < y_d$. But one of the calculated intermediate node is 1,9 where $x_s = 1, y_s = 9$ and $y_s > y_d$. In this case the reader shouldn't jump to next group of algorithm for the condition $y_s > y_d$.

In case 2 we have four group of algorithms with four conditions. The reader should calculate the intermediate nodes by using the same group algorithm.

Case-3 Optimal Routing Algorithm for Vertical Move for lines $m = X_s$ where $m \bmod 3 = 1$. [Move from top to bottom, If $(X_s < X_d \ \&\& \ Y_s = Y_d)$]

Move ($K_s, X_s, Y_s, K_d, X_d, Y_d$)

If $(y_d = 1 \ || \ 2)$ // Logical OR

For $(i = x_s$ to x_d-1 in increasing order)

Move $(k_s, x_s+1, y_s, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 = 1 \ \&\& \ x_d \bmod 3 \neq 1 \ \&\& \ y_s$ is odd)

Move $(k_s, x_s, y_s+1, k_d, x_d, y_d)$

Else If $((x_s \bmod 3 = 1 \ \&\& \ x_d \bmod 3 \neq 1 \ \&\& \ y_s$ is even $\ \&\& \ x_s < x_d) \ ||$

$(x_s \bmod 3 = 1 \ \&\& \ x_d \bmod 3 = 1))$

Move $(k_s, x_s+1, y_s/2+1, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 = 1 \ \&\& \ x_d \bmod 3 \neq 1 \ \&\& \ y_s$ is even $\ \&\& \ x_s > x_d)$

Move $(k_s, x_s-1, y_s/2+1, k_d, x_d, y_d)$

Else If $((x_s \bmod 3 = 2 \ \&\& \ x_d \bmod 3 \neq 1 \ \&\& \ x_s \neq x_d) \ ||$

$(x_s \bmod 3 = 2 \ \&\& \ x_d \bmod 3 = 1))$

Move $(k_s, x_s+1, y_s, k_d, x_d, y_d)$

Else If $((x_s = 3n \ \&\& \ x_d \bmod 3 \neq 1) \ ||$

$(x_s = 3n \ \&\& \ y_d$ is odd $\ \&\& \ x_d \bmod 3 = 1))$

Move $(k_s, x_s+1, 2y_s-1, k_d, x_d, y_d)$

Else If $(x_s = 3n \ \&\& \ x_d \bmod 3 = 1 \ \&\& \ y_d$ is even)

Move $(k_s, x_s+1, 2y_s-2, k_d, x_d, y_d)$

Else

If $(x_s = x_d)$

Go to Horizontal Move

Else

Destination reached

Example 3.1 Let $(K_s, X_s, Y_s) = (2, 1, 10)$ and $(K_d, X_d, Y_d) = (1, 9, 10)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

$2, 1, 10 \rightarrow 1, 1, 10 \rightarrow 1, 2, 6 \rightarrow 1, 3, 6 \rightarrow 1, 4, 11 \rightarrow 1, 4, 12 \rightarrow 1, 5, 7 \rightarrow 1, 6, 7 \rightarrow 1, 7, 13 \rightarrow 1, 7, 14 \rightarrow 1, 8, 8 \rightarrow 1, 9, 8 \rightarrow 1, 10, 15 \rightarrow 1, 10, 16 \rightarrow 1, 9, 9 \rightarrow 1, 10, 17 \rightarrow 1, 10, 18 \rightarrow 1, 9, 10$

The shortest path length is 16

Case-4 Optimal Routing Algorithm for Vertical Move for lines $m=X_s$, where $m \bmod 3 = 1$. [Move from bottom to top, If $(X_s > X_d \ \&\& \ Y_s = Y_d)$]

Move $(K_s, X_s, Y_s, \ K_d, X_d, Y_d)$

If $(y_d = 1 \ || \ 2)$ // Logical OR

For $(i = x_s$ to x_d-1 in decreasing order)

Move $(k_s, x_s-1, y_s, k_d, x_d, y_d)$

Else If $((x_s \bmod 3 = 1 \ \&\& \ x_d \bmod 3 = 1) \ || \$

$(x_s \bmod 3 = 1 \ \&\& \ x_d \bmod 3 \neq 1 \ \&\& \ y_s$ is even $\ \&\& \ x_s \neq x_d)$)

Move $(k_s, x_s-1, y_s/2+1, k_d, x_d, y_d)$

Else If $((x_s \bmod 3 = 1 \ \&\& \ x_d \bmod 3 \neq 1 \ \&\& \ y_s$ is odd $\ \&\& \ x_s \neq x_d)$

Move $(k_s, x_s, y_s+1, k_d, x_d, y_d)$

Else If $(x_s = 3n)$

Move $(k_s, x_s-1, y_s, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 = 2 \ \&\& \ x_d \bmod 3 = 1 \ \&\& \ y_d$ is even) $\ || \ (x_s \bmod 3 = 2 \ \&\& \ y_s = y_d)$

Move $(k_s, x_s-1, 2y_s-2, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 = 2 \ \&\& \ x_d \bmod 3 \neq 1 \ \&\& \ y_s \neq y_d \ \&\& \ x_s \neq x_d)$

Move $(k_s, x_s-1, 2y_s-1, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 \neq 1 \ \&\& \ x_d \bmod 3 \neq 1 \ \&\& \ y_s = y_d)$

Go to Vertical Move Bottom to Top for $X_s \bmod 3 \neq 1$

Else If $(x_s \bmod 3 = 2 \ \&\& \ x_d \bmod 3 = 1 \ \&\& \ y_d$ is odd)

Move $(k_s, x_s-1, 2y_s-1, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 = 1 \ \&\& \ x_d \bmod 3 \neq 1 \ \&\& \ x_s = x_d)$

Go to Horizontal Move

Else

Destination reached

Example 4.1 Let $(K_s, X_s, Y_s) = (2, 10, 5)$ and $(K_d, X_d, Y_d) = (1, 3, 5)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

$2, 10, 5 \rightarrow 1, 10, 5 \rightarrow 1, 10, 6 \rightarrow 1, 9, 4 \rightarrow 1, 8, 4 \rightarrow 1, 7, 7 \rightarrow 1, 7, 8 \rightarrow 1, 6, 5 \rightarrow 1, 5, 5 \rightarrow 1, 4, 8 \rightarrow 1, 3, 5$

The shortest path length is 9

Case-5 Optimal Routing Algorithm for Vertical Move for lines $m=X_s$ where $X_s \bmod 3 \neq 1$. [Move from top to bottom, If $(X_s < X_d \ \&\& \ Y_s = Y_d)$]

Move $(K_s, X_s, Y_s, K_d, X_d, Y_d)$

If $(y_d = 1 \ || \ 2)$ // Logical OR

For $(i = x_s$ to x_d-1 in increasing order)

Move $(k_s, x_s+1, y_s, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 = 2)$

Move $(k_s, x_s+1, y_s, k_d, x_d, y_d)$

Else If $(x_s = 3n)$

Move $(k_s, x_s+1, 2y_s-2, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 = 1 \ \&\& \ x_d \bmod 3 = 1 \ \&\& \ y_s$ is even $\ \&\& \ y_s \neq y_d \ \&\& \ x_s \neq x_d)$

Move $(k_s, x_s, y_s-1, k_d, x_d, y_d)$

Else If $((x_s \bmod 3 = 1 \ \&\& \ x_d \bmod 3 = 1 \ \&\& \ y_s$ is odd $\ \&\& \ y_s \neq y_d \ \&\& \ x_s \neq x_d) \ ||$

$(x_s \bmod 3 = 1 \ \&\& \ x_d \bmod 3 \neq 1)$

Move $(k_s, x_s+1, y_s/2+1, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 = 1 \ \&\& \ x_d \bmod 3 = 1 \ \&\& \ x_s = x_d)$

Go to Horizontal Move

Else If $(x_s \bmod 3 = 1 \ \&\& \ x_d \bmod 3 = 1 \ \&\& \ y_s = y_d)$

Go to Vertical Move Top to Bottom for $X_s \bmod 3 = 1$

Else

Destination reached

Example 5.1 Let $(K_s, X_s, Y_s) = (2, 5, 10)$ and $(K_d, X_d, Y_d) = (2, 11, 10)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

$2, 5, 10 \rightarrow 2, 6, 10 \rightarrow 2, 7, 18 \rightarrow 2, 8, 10 \rightarrow 2, 9, 10 \rightarrow 2, 10, 18 \rightarrow 2, 11, 10$

The shortest path length is 6

Case-6 Optimal Routing Algorithm for Vertical move for lines $m = X_s$ where $m \bmod 3 \neq 1$. [Move from bottom to top, If $(X_s > X_d \ \&\& \ Y_s = Y_d)$]

Move $(K_s, X_s, Y_s, K_d, X_d, Y_d)$

If $(y_d = 1 \ || \ 2)$ // Logical OR

For $(i = x_s$ to x_d-1 in decreasing order)

Move $(k_s, x_s-1, y_s, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 = 2)$

```

    Move ( $k_s, x_s-1, 2y_s-2, k_d, x_d, y_d$ )

    Else If ( $x_s = 3n$ )

        Move ( $k_s, x_s-1, y_s, k_d, x_d, y_d$ )

    Else If ( $x_s \bmod 3 = 1 \ \&\& \ x_d \bmod 3 = 1 \ \&\& \ y_s$  is even  $\ \&\& \ y_s \neq y_d \ \&\& \ x_s \neq x_d$ )

        Move ( $k_s, x_s, y_s-1, k_d, x_d, y_d$ )

    Else If ( $(x_s \bmod 3 = 1 \ \&\& \ x_d \bmod 3 = 1 \ \&\& \ y_s$  is odd  $\ \&\& \ y_s \neq y_d \ \&\& \ x_s \neq x_d) \ ||$ 
            ( $x_s \bmod 3 = 1 \ \&\& \ x_d \bmod 3 \neq 1 \ \&\& \ x_s \neq x_d$ ))

        Move ( $k_s, x_s-1, y_s/2+1, k_d, x_d, y_d$ )

    Else If ( $x_s \bmod 3 = 1 \ \&\& \ x_d \bmod 3 = 1 \ \&\& \ y_s = y_d$ )

        Go to Vertical Move Bottom to Top for  $X_s \bmod 3 = 1$ 

Else

    If ( $x_s = x_d$ )

        Go to Horizontal Move

Else

    Destination reached

```

Example 6.1 Let $(K_s, X_s, Y_s) = (1, 11, 5)$ and $(K_d, X_d, Y_d) = (2, 4, 5)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

$1, 11, 5 \rightarrow 2, 11, 5 \rightarrow 2, 10, 8 \rightarrow 2, 10, 7 \rightarrow 2, 9, 4 \rightarrow 2, 8, 4 \rightarrow 2, 7, 6 \rightarrow 2, 7, 5 \rightarrow 2, 6, 3 \rightarrow 2, 5, 3 \rightarrow 2, 4, 5$

The shortest path length is 9

Case-7 Optimal Routing Algorithm for $X_s \neq X_d \ \&\& \ Y_s \neq Y_d \ \&\& \ X_s < X_d \ \&\& \ Y_s < Y_d$

- SUB CASE -7.1 ($X_s \bmod 3 = 1 \ \&\& \ X_d \bmod 3 = 1$)

Move ($K_s, X_s, Y_s, K_d, X_d, Y_d$)

If ($x_s \neq x_d \ \&\& \ y_s \neq y_d$)

If ($x_s \bmod 3 = 1 \ \&\& \ y_s$ is odd)

Move ($k_s, x_s, y_s+1, k_d, x_d, y_d$)

Else If ($x_s \bmod 3 = 1 \ \&\& \ y_s$ is even)

Move ($k_s, x_s+1, y_s/2+1, k_d, x_d, y_d$)

Else If ($x_s \bmod 3 = 2$)

Move ($k_s, x_s+1, y_s, k_d, x_d, y_d$)

Else If ($x_s = 3n$)

Move ($k_s, x_s+1, 2y_s-1, k_d, x_d, y_d$)

Else If ($x_s = x_d$)

Go to Horizontal Move

Else If ($y_s = y_d$)

Go to Vertical Move Top to Bottom

Else

Destination reached.

Example 7.1.1 Let $(K_s, X_s, Y_s) = (1, 1, 11)$ and $(K_d, X_d, Y_d) = (2, 7, 12)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

$1, 1, 11 \rightarrow 2, 1, 11 \rightarrow 2, 1, 12 \rightarrow 2, 2, 7 \rightarrow 2, 3, 7 \rightarrow 2, 4, 12 \rightarrow 2, 5, 7 \rightarrow 2, 6, 7 \rightarrow 2, 7, 12$

The shortest path length is 7

• **SUBCASE -7.2 ($X_s \bmod 3 = 1$ && $X_d \bmod 3 \neq 1$)**

Go to subcase-7.1

Example 7.2.1 Let $(K_s, X_s, Y_s) = (2, 7, 3)$ and $(K_d, X_d, Y_d) = (1, 12, 7)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

$2, 7, 3 \rightarrow 1, 7, 3 \rightarrow 1, 7, 4 \rightarrow 1, 8, 3 \rightarrow 1, 9, 3 \rightarrow 1, 10, 5 \rightarrow 1, 10, 6 \rightarrow 1, 11, 4 \rightarrow 1, 12, 4 \rightarrow 1, 13, 7 \rightarrow 1, 13, 8 \rightarrow 1, 12, 5 \rightarrow 1, 13, 9 \rightarrow 1, 13, 10 \rightarrow 1, 12, 6 \rightarrow 1, 13, 11 \rightarrow 1, 13, 12 \rightarrow 1, 12, 7$

The shortest path length is 16

• **SUBCASE -7.3 ($X_s \bmod 3 \neq 1$ && $X_d \bmod 3 = 1$)**

If ($2y_s - 2 \geq y_d$ (Move to left))

Move ($K_s, X_s, Y_s, K_d, X_d, Y_d$)

If ($x_s \neq x_d$ && $y_s \neq y_d$)

If ($x_s \bmod 3 = 1$ && y_s is odd)

Move ($k_s, x_s+1, y_s/2+1, k_d, x_d, y_d$)

Else If ($x_s \bmod 3 = 1$ && y_s is even)

Move ($k_s, x_s, y_s-1, k_d, x_d, y_d$)

Else If ($x_s \bmod 3 = 2$)

Move ($k_s, x_s+1, y_s, k_d, x_d, y_d$)

Else If ($x_s = 3n$)

Move ($k_s, x_s+1, 2y_s-2, k_d, x_d, y_d$)

Else If ($x_s = x_d$)

Go to Horizontal Move

Else If ($y_s = y_d$)

Go to Vertical Move Top to Bottom

Else

Destination reached.

Else If $(2y_s - 2) \leq y_d$ (Move to right)

Go to subcase-7.1

Example 7.3.1 Let $(K_s, X_s, Y_s) = (2, 3, 7)$ and $(K_d, X_d, Y_d) = (2, 13, 11)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

$2, 3, 7 \rightarrow 2, 4, 12 \rightarrow 2, 4, 11 \rightarrow 2, 5, 6 \rightarrow 2, 6, 6 \rightarrow 2, 7, 11 \rightarrow 2, 8, 6 \rightarrow 2, 9, 6 \rightarrow 2, 10, 11 \rightarrow 2, 11, 6 \rightarrow 2, 12, 6 \rightarrow 2, 13, 11$

The shortest path length is 11

• **SUBCASE -7.4 $(X_s \bmod 3 \neq 1 \ \&\& \ X_d \bmod 3 \neq 1)$**

Go to subcase-7.1

Example 7.4.1 Let $(K_s, X_s, Y_s) = (1, 2, 4)$ and $(K_d, X_d, Y_d) = (2, 8, 8)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

$1, 2, 4 \rightarrow 2, 2, 4 \rightarrow 2, 3, 4 \rightarrow 2, 4, 7 \rightarrow 2, 4, 8 \rightarrow 2, 5, 5 \rightarrow 2, 6, 5 \rightarrow 2, 7, 9 \rightarrow 2, 7, 10 \rightarrow 2, 8, 6 \rightarrow 2, 7, 11 \rightarrow 2, 7, 12 \rightarrow 2, 8, 7 \rightarrow 2, 7, 13 \rightarrow 2, 7, 14 \rightarrow 2, 8, 8$

The shortest path length is 14

Case-8 Optimal Routing Algorithm for $X_s \neq X_d \ \&\& \ Y_s \neq Y_d \ \&\& \ X_s < X_d \ \&\& \ Y_s > Y_d$

• **SUB CASE-8.1 $(X_s \bmod 3 = 1 \ \&\& \ X_d \bmod 3 = 1)$**

Move $(K_s, X_s, Y_s, K_d, X_d, Y_d)$

If $(x_s \neq x_d \ \&\& \ y_s \neq y_d)$

If $(x_s \bmod 3 = 1 \ \&\& \ y_s$ is odd)

Move $(k_s, x_s+1, y_s/2+1, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 = 1 \ \&\& \ y_s$ is even)

Move $(k_s, x_s, y_s-1, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 = 2)$

Move $(k_s, x_s+1, y_s, k_d, x_d, y_d)$

Else If $(x_s = 3n)$

Move $(k_s, x_s+1, 2y_s-2, k_d, x_d, y_d)$

Else If $(x_s = x_d)$

Go to Horizontal Move

Else If $(y_s = y_d)$

Go to Vertical Move Top to Bottom

Else

Destination reached

Example 8.1.1 Let $(K_s, X_s, Y_s) = (1, 1, 9)$ and $(K_d, X_d, Y_d) = (1, 7, 3)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

$1, 1, 9 \rightarrow 1, 2, 5 \rightarrow 1, 3, 5 \rightarrow 1, 4, 8 \rightarrow 1, 4, 7 \rightarrow 1, 5, 4 \rightarrow 1, 6, 4 \rightarrow 1, 7, 6 \rightarrow 1, 7, 5 \rightarrow 1, 8, 3 \rightarrow 1, 7, 4 \rightarrow 1, 7, 3$

The shortest path length is 11

• **SUB CASE-8.2 ($X_s \bmod 3 = 1$ && $X_d \bmod 3 \neq 1$)**

If $(2y_d - 1) > y_s$ (Move left to right)

Move $(K_s, X_s, Y_s, K_d, X_d, Y_d)$

If $(x_s \neq x_d$ && $y_s \neq y_d)$

If $(x_s \bmod 3 = 1$ && y_s is even)

Move $(k_s, x_s+1, y_s/2+1, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 = 1$ && y_s is odd)

Move $(k_s, x_s, y_s+1, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 = 2)$

Move $(k_s, x_s+1, y_s, k_d, x_d, y_d)$

Else If $(x_s = 3n)$

Move $(k_s, x_s+1, 2y_s-1, k_d, x_d, y_d)$

Else If $(x_s = x_d)$

Go to Horizontal Move

Else If $(y_s = y_d)$

Go to Vertical Move Top to Bottom

Else

Destination reached

Else If $(2y_d - 1) \leq y_s$ (Move right to left)

Go to subcase-8.1

Example 8.2.1 Let $(K_s, X_s, Y_s) = (2, 4, 10)$ and $(K_d, X_d, Y_d) = (1, 8, 7)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

$2, 4, 10 \rightarrow 1, 4, 10 \rightarrow 1, 5, 6 \rightarrow 1, 6, 6 \rightarrow 1, 7, 11 \rightarrow 1, 7, 12 \rightarrow 1, 8, 7$

The shortest path length is 5

• **SUB CASE-8.3 ($X_s \bmod 3 \neq 1$ && $X_d \bmod 3 = 1$)**

Go to subcase-8.1

Example 8.3.1 Let $(K_s, X_s, Y_s) = (1, 2, 10)$ and $(K_d, X_d, Y_d) = (2, 10, 8)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

1,2,10 → 2,2,10 → 2,3,10 → 2,4,18 → 2,4,17 → 2,5,9 → 2,6,9 → 2,7,16 → 2,7,15 → 2,8,8 → 2,9,8 → 2,10,14 → 2,10,13 → 2,11,7 → 2,10,12 → 2,10,11 → 2,11,6 → 2,10,10 → 2,10,9 → 2,11,5 → 2,10,8

The shortest path length is 19

• **SUB CASE-8.4 ($X_s \bmod 3 \neq 1$ && $X_d \bmod 3 \neq 1$)**

Go to subcase-8.1

Example 8.4.1 Let $(K_s, X_s, Y_s) = (2, 2, 8)$ and $(K_d, X_d, Y_d) = (2, 6, 7)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

2,2,8 → 2,3,8 → 2,4,14 → 2,4,13 → 2,5,7 → 2,6,7

The shortest path length is 5

Case-9 Optimal Routing Algorithm for $X_s \neq X_d$ && $Y_s \neq Y_d$ && $X_s > X_d$ && $Y_s < Y_d$

• **SUB CASE-9.1 ($X_s \bmod 3 = 1$ && $X_d \bmod 3 = 1$)**

Move $(K_s, X_s, Y_s, K_d, X_d, Y_d)$

If $(x_s \neq x_d$ && $y_s \neq y_d)$

If $(x_s \bmod 3 = 1$ && y_s is odd)

Move $(k_s, x_s, y_s+1, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 = 1$ && y_s is even)

Move $(k_s, x_s-1, y_s/2+1, k_d, x_d, y_d)$

Else If $(x_s = 3n)$

Move $(k_s, x_s-1, y_s, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 = 2)$

Move $(k_s, x_s-1, 2y_s-1, k_d, x_d, y_d)$

Else If $(x_s = x_d)$

Go to Horizontal Move

Else If $(y_s = y_d)$

Go to Vertical Move Bottom to Top

Else

Destination reached

Example 9.1.1 Let $(K_s, X_s, Y_s) = (2, 13, 10)$ and $(K_d, X_d, Y_d) = (1, 1, 16)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

2,13,10 → 1,13,10 → 1,12,6 → 1,11,6 → 1,10,11 → 1,10,12 → 1,9,7 → 1,8,7 → 1,7,13 → 1,7,14 → 1,6,8 → 1,5,8 → 1,4,15 → 1,4,16 → 1,3,9 → 1,2,9 → 1,1,16

The shortest path length is 15

• **SUB CASE-9.2 ($X_s \bmod 3 = 1$ && $X_d \bmod 3 \neq 1$)**

Go to sub case - 9.1

Example 9.2.1 Let $(K_s, X_s, Y_s) = (1, 7, 3)$ and $(K_d, X_d, Y_d) = (2, 2, 8)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

$1, 7, 3 \rightarrow 2, 7, 3 \rightarrow 2, 7, 4 \rightarrow 2, 6, 3 \rightarrow 2, 5, 3 \rightarrow 2, 4, 5 \rightarrow 2, 4, 6 \rightarrow 2, 3, 4 \rightarrow 2, 2, 4 \rightarrow 2, 1, 7 \rightarrow 2, 1, 8 \rightarrow 2, 2, 5 \rightarrow 2, 1, 9 \rightarrow 2, 1, 10 \rightarrow 2, 2, 6 \rightarrow 2, 1, 11 \rightarrow 2, 1, 12 \rightarrow 2, 2, 7 \rightarrow 2, 1, 13 \rightarrow 2, 1, 14 \rightarrow 2, 2, 8$

The shortest path length is 19

• **SUB CASE-9.3 ($X_s \bmod 3 \neq 1$ && $X_d \bmod 3 = 1$)**

If $(2y_s - 1) \geq y_d$ (Move right to left)

Move $(K_s, X_s, Y_s, K_d, X_d, Y_d)$

If $(x_s \neq x_d$ && $y_s \neq y_d)$

If $(x_s \bmod 3 = 1$ && y_s is even)

Move $(k_s, x_s, y_s - 1, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 = 1$ && y_s is odd)

Move $(k_s, x_s - 1, y_s / 2 + 1, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 = 2)$

Move $(k_s, x_s - 1, 2y_s - 2, k_d, x_d, y_d)$

Else If $(x_s = 3n)$

Move $(k_s, x_s - 1, y_s, k_d, x_d, y_d)$

Else If $(x_s = x_d)$

Go to Horizontal Move

Else If $(y_s = y_d)$

Go to Vertical Move

Else

Destination reached

Else If $(2y_s - 1) \leq y_d$ (Move left to right)

Go to subcase-9.1

Example 9.3.1 Let $(K_s, X_s, Y_s) = (2, 14, 10)$ and $(K_d, X_d, Y_d) = (1, 4, 13)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

$2, 14, 10 \rightarrow 1, 14, 10 \rightarrow 1, 13, 18 \rightarrow 1, 13, 17 \rightarrow 1, 12, 9 \rightarrow 1, 11, 9 \rightarrow 1, 10, 16 \rightarrow 1, 10, 15 \rightarrow 1, 9, 8 \rightarrow 1, 8, 8 \rightarrow 1, 7, 14 \rightarrow 1, 7, 13 \rightarrow 1, 6, 7 \rightarrow 1, 5, 7 \rightarrow 1, 4, 13$

The shortest path length is 13

- **SUB CASE-9.4 ($X_s \bmod 3 \neq 1$ && $X_d \bmod 3 \neq 1$)**

Go to sub case - 9.1

Example 9.4.1 Let $(K_s, X_s, Y_s) = (2, 11, 5)$ and $(K_d, X_d, Y_d) = (1, 3, 6)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

$2, 11, 5 \rightarrow 1, 11, 5 \rightarrow 1, 10, 9 \rightarrow 1, 10, 10 \rightarrow 1, 9, 6 \rightarrow 1, 8, 6 \rightarrow 1, 7, 10 \rightarrow 1, 6, 6 \rightarrow 1, 5, 6 \rightarrow 1, 4, 10 \rightarrow 1, 3, 6$

The shortest path length is 9

Case-10 Optimal Routing Algorithm for $X_s \neq X_d$ && $Y_s \neq Y_d$ && $X_s > X_d$ && $Y_s > Y_d$

- **SUB CASE-10.1 ($X_s \bmod 3 = 1$ && $X_d \bmod 3 = 1$)**

Go to sub case 9.3 [Move right to left]

Example 10.1.1 Let $(K_s, X_s, Y_s) = (2, 10, 16)$ and $(K_d, X_d, Y_d) = (2, 1, 8)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

$2, 10, 16 \rightarrow 2, 10, 15 \rightarrow 2, 9, 8 \rightarrow 2, 8, 8 \rightarrow 2, 7, 14 \rightarrow 2, 7, 13 \rightarrow 2, 6, 7 \rightarrow 2, 5, 7 \rightarrow 2, 4, 12 \rightarrow 2, 4, 11 \rightarrow 2, 3, 6 \rightarrow 2, 2, 6 \rightarrow 2, 1, 10 \rightarrow 2, 1, 9 \rightarrow 2, 2, 5 \rightarrow 2, 1, 8$

The shortest path length is 15

- **SUB CASE-10.2 ($X_s \bmod 3 = 1$ && $X_d \bmod 3 \neq 1$)**

If $(2y_d - 1) > y_s$ (Move to right)

Move $(K_s, X_s, Y_s, K_d, X_d, Y_d)$

If $(x_s \neq x_d$ && $y_s \neq y_d)$

If $(x_s \bmod 3 = 1$ && y_s is odd)

Move $(k_s, x_s, y_s+1, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 = 1$ && y_s is even)

Move $(k_s, x_s-1, y_s/2+1, k_d, x_d, y_d)$

Else If $(x_s \bmod 3 = 2)$

Move $(k_s, x_s-1, 2y_s-1, k_d, x_d, y_d)$

Else If $(x_s = 3n)$

Move $(k_s, x_s-1, y_s, k_d, x_d, y_d)$

Else If $(x_s = x_d)$

Go to Horizontal Move

Else If $(y_s = y_d)$

Go to Vertical Move

Else

Destination reached

If $(2y_d-1) \leq y_s$ [Move right to left]

Go to sub case 9.3

Example 10.2.1 Let $(K_s, X_s, Y_s) = (1, 10, 16)$ and $(K_d, X_d, Y_d) = (1, 2, 10)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

$1, 10, 16 \rightarrow 1, 9, 9 \rightarrow 1, 8, 9 \rightarrow 1, 7, 17 \rightarrow 1, 7, 18 \rightarrow 1, 6, 10 \rightarrow 1, 5, 10 \rightarrow 1, 4, 18 \rightarrow 1, 3, 10 \rightarrow 1, 2, 10$

The shortest path length is 9

• **SUB CASE-10.3** ($X_s \bmod 3 \neq 1$ && $X_d \bmod 3 = 1$)

Go to sub case 9.3 [Move right to left]

Example 10.3.1 Let $(K_s, X_s, Y_s) = (2, 12, 7)$ and $(K_d, X_d, Y_d) = (1, 7, 5)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

$2, 12, 7 \rightarrow 1, 12, 7 \rightarrow 1, 11, 7 \rightarrow 1, 10, 12 \rightarrow 1, 10, 11 \rightarrow 1, 9, 6 \rightarrow 1, 8, 6 \rightarrow 1, 7, 10 \rightarrow 1, 7, 9 \rightarrow 1, 8, 5 \rightarrow 1, 7, 8 \rightarrow 1, 7, 7 \rightarrow 1, 8, 4 \rightarrow 1, 7, 6 \rightarrow 1, 7, 5$

The shortest path length is 13

• **SUB CASE-10.4** ($X_s \bmod 3 \neq 1$ && $X_d \bmod 3 \neq 1$)

Go to sub case -9.3 [Move right to left]

Example 10.4.1 Let $(K_s, X_s, Y_s) = (1, 8, 7)$ and $(K_d, X_d, Y_d) = (2, 3, 4)$ be the source and destination nodes respectively. To reach the destination the intermediate nodes for optimal paths are:

$1, 8, 7 \rightarrow 2, 8, 7 \rightarrow 2, 7, 12 \rightarrow 2, 7, 11 \rightarrow 2, 6, 6 \rightarrow 2, 5, 6 \rightarrow 2, 4, 10 \rightarrow 2, 4, 9 \rightarrow 2, 3, 5 \rightarrow 2, 4, 8 \rightarrow 2, 4, 7 \rightarrow 2, 3, 4$

The shortest path length is 10

5 Conclusion

In an interconnection network data routing is the most fundamental function. Information can be moved across an interconnection network from a source to a destination by the act of data routing. In this paper a new topological structure called multi layer octagon-cell network has been developed and also we have derived a simpler and an efficient optimal point to point routing algorithm with its properties for MLO.

REFERENCES

- [1]. Sanjukta Mohanty and Prafulla Ku. Behera, *Optimal Routing Algorithm in a Octagon-Cell Network*. International Journal of Advanced Research in Computer Science, 2011. Vol.2, No.5: p.625-637
- [2]. Boxer, L. and Miller, R., *Dynamic Computational Geometry on Meshes and Hypercubes*. Proceedings of the International Conference on Parallel Processing, st. Charles, Illinois, 1998. p. 323-330.
- [3]. Ahmad Sharieh, Mohammad Qatawneh, Wesam Almobaideen, Azzam Sleit, *Hex-Cell: Modeling, Topological Properties and Routing Algorithm*. European Journal of Scientific Research, 2008. Vol.22 No.2: p. 457-468.
- [4]. Catherine Decayeux and David Seme, *3D Hexagonal Network: Modeling, Topological properties, Addressing scheme, and Optimal Routing Algorithm*. IEEE Transaction on parallel and distributed Systems, 2005. Vol. 16, No. 9: p. 875-884.
- [5]. Parhami, *Computer Architecture: From Microprocessors to Supercomputers*, Oxford. 2005.

- [6]. Qatawneh Mohammed, *Adaptive Fault Tolerant Routing algorithm for Tree-Hypercube Multicomputer*. Journal of Computer Science, 2006. Vol.2, No.2: p. 124-126.
- [7]. Y Saad and M H Schultz, *Topological Properties of Hypercubes*, IEEE Transaction Computers. 1998. Vol. 37: p. 867.
- [8]. Chan, T. F. and Saad, Y., *Multigrid Algorithms on the Multiprocessor*. IEEE Transactions on Computers, November, 2002. Vol. C-35, No. 11: p. 969-977.
- [9]. Della Vecchia, G. and C. Sanges, *A Recursively Scalable Network for VLSI Implementation*, Future Generation Computer Systems, 1995. p.235-243.
- [10]. Kai Hwang, *Advanced Computer Architecture: Parallelism, Scalability, Programmability*. McGraw-Hill Book Co. International Edition, 1993.
- [11]. Parhami, B., *Introduction to parallel Processing: Algorithms and Architectures*, Plenum, 1999.
- [12]. Ghose, K. and K.R. Desai, *Hierarchical Cubic Network*. IEEE Transaction Parallel and Distributed Systems, 1995. Vol. 6, No. 4: p. 427-435.
- [13]. J. Bosh, D. Maltz, D. Johnson, Y. Hu, and J. Jetcheva, *A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols*. Proceedings of 4th Annual AAM/IEEE International Conference On Mobile Computing and Networking , 1998.
- [14]. R. Schoonderwoerd, O. Holland, J. Bruten, and L. Rothkrantz, *Ant-based Load Balancing in telecommunications Networks*. Adaptive Behavior, 1997. Vol. 5: p. 169-207, 1997.
- [15]. C. K. Toh, *Maximum Battery Life Routing to Support Ubiquitous Mobile Computing in Wireless D Hoc Networks*. IEEE Communications Magazine, 2001. p. 138-147.
- [16]. Mohammad Qatawneh, *Multilayer Hex-Cells: A New Class of Hex-Cell Interconnection Networks for Massively Parallel Systems*. Int.J. Communications, Network and System Sciences, 2011. Vol-4: p.704-708.