Deterministic Routing on the Multibutterfly Network

Franz Diebold

23.09.2008

Contents

1 Introduction

Considering the standard butterfly network, one may ask, why we should then use the multibutterfly network (MBN) for parallel computing networks instead of the standard butterfly network?

Therefore the advantages of using the MBN should be made clear:

- The MBN is robust against `faults` (i.e. congestion, failure).
  - There are `several paths` connecting any input to any output.
- The MBN is a `high-bandwidth` network.
- The MBN is a `low-diameter` network.

In contrast to the standard butterfly network especially the first fact is important: By reason of the existence of several paths from any input to any output the MBN is highly fault-resistant. That means that the MBN could work quite well, if there was a failure of a node or congestion in routing packets.

2 The Multibutterfly-Graph

For understanding the structure of the multibutterfly graph, one should first know about Concentrators and Splitters.

2.1 Concentrators

Definition 1 (Concentrator)
A bipartite graph $G = (V \cup W, E)$ is called an (α, β, m, c)-`concentrator` if

1. $|V| = m$ *and* $|W| = \frac{m}{2}$,

2. *the nodes in V have degree at most c and the nodes in W have degree at most $2 \cdot c$, and*

3. *for all $U \subseteq V$, $|U| \leq \alpha \cdot |V|$: $|\Gamma(u)| \geq \beta \cdot |U|$ (`Expansion-property`)*

With

Declaration 1 *Given a graph $G = (V, E)$ and $U \subseteq V$:*

- $\Gamma(v) := \{u | \{u, v\} \in E\}$
- $\Gamma(U) := \bigcup_{v \in U} \Gamma(v)$

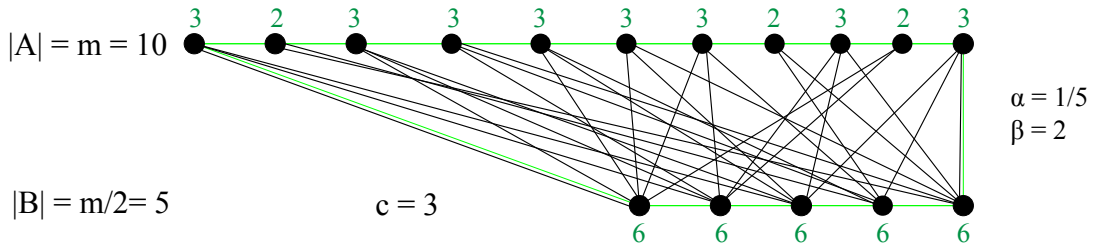

Figure 1: Design of a concentrator

The expansion-property means that every subset of nodes in V (up to a certain size) has many neighbours in W, i.e. V `expands`.

Let's now verify the existence of concentrators:

Lemma 1 (Existence of concentrators)
For $\alpha \leq \frac{1}{2\beta}(4\beta \cdot e^{1+\beta})^{-\frac{1}{c-\beta-1}}$ there exists an (α, β, m, c)-concentrator.

Proof
For an arbitrary permutation $\pi : A \to A$ let

$$E_\pi = \{(i,j) \in A \times B | \pi(i) \in \{j, j + \frac{m}{2}\}\}.$$

Consider $\mathcal{R}$ to be a class of `bipartite graphs`

$$G = (A \cup B, E), A = [m], B = \left[\frac{m}{2}\right].$$

Let $\mathcal{R} = \{G = (A \cup B, E) | E = E_{\pi_1} \cup \cdots \cup E_{\pi_c}$ for the permutations $\pi_1, \ldots, \pi_c : A \to A\}$.

Let G be an arbitratry graph in $\mathcal{R}$.

There exists a (α, β, m, c)-concentrator in $\mathcal{R}$ if

Prob(G is not a (α, β, m, c)-concentrator) < 1

Prob(G is not a (α, β, m, c)-concentrator)

$$\begin{aligned}
&\leq Prob(\exists X \subseteq A, |X| \leq \alpha m : \ |\Gamma(X)| < \beta \cdot |X|) \\
&\leq Prob(\exists \mu \leq \alpha m, \exists X \subseteq A, \exists Y \subseteq B : |X| = \mu \wedge |Y| = \lfloor \beta \cdot \mu \rfloor \wedge \Gamma(X) \subseteq Y) \\
&\leq \sum_{\mu=1}^{\lfloor \alpha \cdot m \rfloor} \sum_{\substack{X \subseteq A \\ |X| = \mu}} \sum_{\substack{Y \subseteq B \\ |Y| = \lfloor \beta \cdot \mu \rfloor}} Prob(\Gamma(X) \subseteq Y)
\end{aligned}$$

Estimation of the term:
$Prob(\Gamma(X) \subseteq Y)$

$$\begin{aligned}
&= Prob(\bigwedge_{i=1}^{c} \pi_i(X) \subseteq (Y \cup \underbrace{\{j + \frac{m}{2} \mid j \in Y\}}_{:=Y'}) \\
&= \prod_{i=1}^{c} Prob(\pi_i(X) \subseteq (Y \cup Y')) \\
&\leq \prod_{i=1}^{c} \frac{\mu! \cdot \binom{2\lfloor \beta\mu \rfloor}{\mu} \cdot (m-\mu)!}{m!} \\
&\leq \prod_{i=1}^{c} \left(\frac{2 \cdot \beta \cdot \mu}{m} \cdot \frac{2 \cdot \beta \cdot \mu - 1}{m-1} \cdot \frac{2 \cdot \beta \cdot \mu - 2}{m-2} \cdot \cdots \cdot \frac{2 \cdot \beta \cdot \mu - \mu + 1}{m - \mu + 1} \right) \\
&\leq \prod_{i=1}^{c} \left(\frac{2 \cdot \beta \cdot \mu}{m} \right)^{\mu} = \left(\frac{2 \cdot \beta \cdot \mu}{m} \right)^{c \cdot \mu}
\end{aligned}$$

Prob(G is not a (α, β, m, c)-concentrator)

$$\leq \sum_{\mu=1}^{\lfloor \alpha \cdot m \rfloor} \sum_{\substack{X \subseteq A \\ |X|=\mu}} \sum_{\substack{Y \subseteq B \\ |Y|=\lfloor \beta \cdot \mu \rfloor}} \left(\frac{2 \cdot \beta \cdot \mu}{m} \right)^{c \cdot \mu}$$

$$\leq \sum_{\mu=1}^{\lfloor \alpha \cdot m \rfloor} \binom{m}{\mu} \cdot \binom{\frac{m}{2}}{\lfloor \beta \mu \rfloor} \cdot \left(\frac{2 \cdot \beta \cdot \mu}{m} \right)^{c \cdot \mu}$$

Estimation of the binomial coefficient:
For k, l $\in N$, $0 \leq l \leq k$:

$$\binom{k}{l} \leq \left(\frac{k \cdot e}{l} \right)^{l}$$

Proof

$$\binom{k}{l} = \frac{k \cdot (k-1) \cdot \ldots \cdot (k-l+1)}{l!} \leq \frac{k^l}{l!} = \frac{k^l}{l^l} \cdot \frac{l^l}{l!} \leq \frac{k^l}{l^l} \cdot e^l$$

with $e^l = \sum_{i=0}^{\infty} \frac{l^i}{i!} \geq \frac{l^l}{l!}$

Prob(G is not a (α, β, m, c)-concentrator)

$$\leq \sum_{\mu=1}^{\lfloor \alpha \cdot m \rfloor} \binom{m}{\mu} \cdot \binom{\frac{m}{2}}{\lfloor \beta \mu \rfloor} \cdot \left(\frac{2 \cdot \beta \cdot \mu}{m} \right)^{c \cdot \mu}$$

$$\leq \sum_{\mu=1}^{\lfloor \alpha \cdot m \rfloor} \left(\frac{e \cdot m}{\mu} \right)^{\mu} \cdot \left(\frac{e \cdot \frac{m}{2}}{\beta \mu} \right)^{\beta \cdot \mu} \cdot \left(\frac{2 \cdot \beta \cdot \mu}{m} \right)^{c \cdot \mu}$$

$$\leq \sum_{\mu=1}^{\lfloor \alpha \cdot m \rfloor} \left[\left(\frac{m}{\mu} \right)^{1+\beta-c} \cdot e^{1+\beta} \cdot (2 \cdot \beta)^{c-\beta} \right]^{\mu}$$

$$\leq \sum_{\mu=1}^{\lfloor \alpha \cdot m \rfloor} \left[\left(\frac{m}{\mu} \right)^{1+\beta-c} \cdot e^{1+\beta} \cdot (2 \cdot \beta)^{c-\beta} \right]^{\mu}$$

$$< \sum_{\mu=1}^{\infty} \left(\underbrace{\alpha^{c-1-\beta} \cdot e^{1+\beta} \cdot (2\beta)^{c-\beta}}_{=:r} \right)^{\mu} \qquad (\mu \leq \alpha \cdot m)$$

$$\leq 1 \qquad (for\ r \leq \frac{1}{2})$$

Due to the `infinite geometric series` the necessary condition is:

$$\begin{aligned}
\alpha^{c-1-\beta} \cdot e^{1+\beta} \cdot (2\beta)^{c-\beta} &\leq \frac{1}{2} \\
\alpha^{c-1-\beta} &\leq \left(2 \cdot e^{1+\beta} \cdot (2\beta)^{c-\beta}\right)^{-1} \\
\alpha &\leq \left(2 \cdot e^{1+\beta} \cdot (2\beta)^{c-\beta}\right)^{-\frac{1}{c-1-\beta}} \\
\alpha &\leq \frac{1}{2\beta} \left(2e^{1+\beta} \cdot (2\beta)^{c-\beta} \cdot (2\beta)^{-(c-\beta-1)}\right)^{-\frac{1}{c-1-\beta}} \\
\alpha &\leq \frac{1}{2\beta} \cdot \left(4\beta \cdot e^{1+\beta}\right)^{-\frac{1}{c-1-\beta}}
\end{aligned}$$

□

Example
For $\beta = 2$ and $c = 4$ and $\alpha \leq \frac{1}{32e^3}$ there exists a $(\frac{1}{643}, 2, m, 4)$-concentrator.

2.2 Splitter

For the construction of the Multibuttefly-graph we need splitter, which consist of concentrators.

Definition 2 (Splitter)
A (α, β, m, c)-splitter is a bipartite graph $G = (V \cup (W_0 \cup W_1), E_0 \cup E_1)$ in which $(V \cup W_0, E_0)$ and $(V \cup W_1, E_1)$ represent (α, β, m, c)-concentrators.
Edges in E_0 are called 0-edges, and edges in E_1 are called 1-edges.

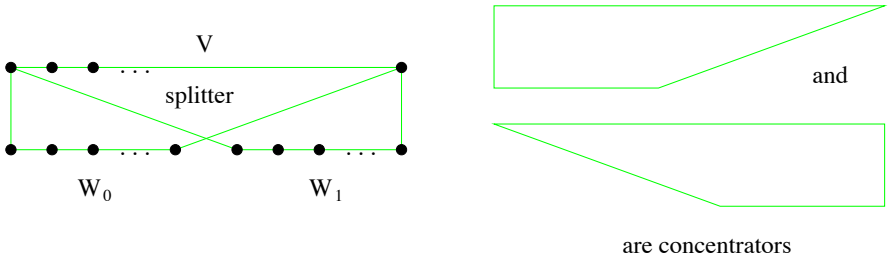

Figure 2: Design of a splitter

2.3 The Multibutterfly-Graph

Definition 3 (Multibutterfly-Graph)
The d-dimensional multibutterfly graph (d, α, β, c) *has* $N = \underbrace{(d+1)}_{number\ of\ levels} \cdot \underbrace{2^d}_{number\ of\ inputs/outputs}$ *nodes and degree at most* $4c$.

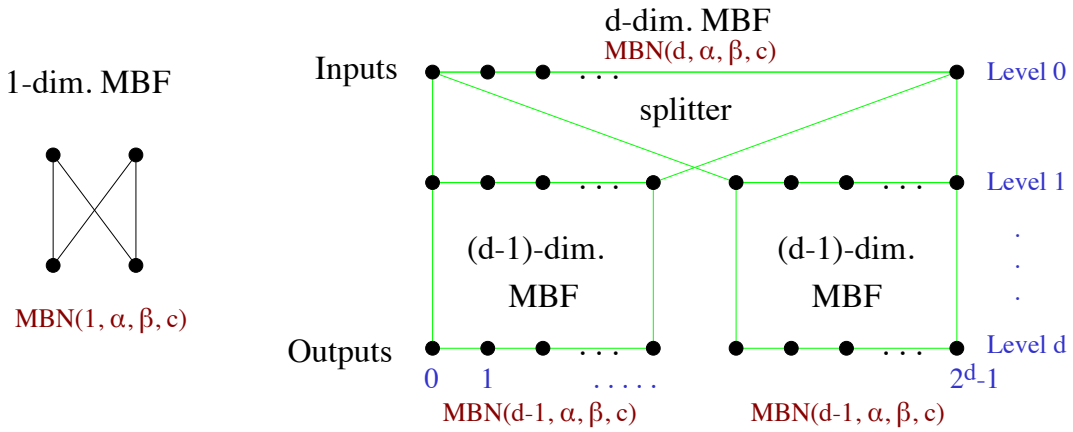

Figure 3: Recursive construction of a multibutterfly graph

As the concentrators and the splitters, the Multibutterfly-graph also has the `expansion-property`.

3 Routing on the Multibutterfly-network

The expansion-property of the Multibutterfly-graph will help us to route efficiently in the network.

3.1 The algorithm

Preparations

Consider α, β and c as `constant`. As we saw before, the `number of inputs` of the MBN is $n = 2^d$ - that means that there are n packets, one per input.

We define a variable

$$L = \left\lceil \frac{1}{2\alpha} \right\rceil .$$

Now we can partition the `packets` into `waves` A_i, whereas the destinations j in each packet are congruent modulo L:

$$j \; mod \; L \; = \; i$$

Hence the waves $A_0, \ldots, A_{L-1}$ consist of approximately $\frac{n}{L}$ packets. (n packets in L waves)

The actual routing proceeds in `stages` consisting of an

- `even` phase (sending from even to odd levels), and an
- `odd` phase (sending from odd to even levels).

Each of these phases consists of `2c steps`.

The edges connecting levels are `colored` in 2c colors so that each node is incident to one edge of each color (`matching`).

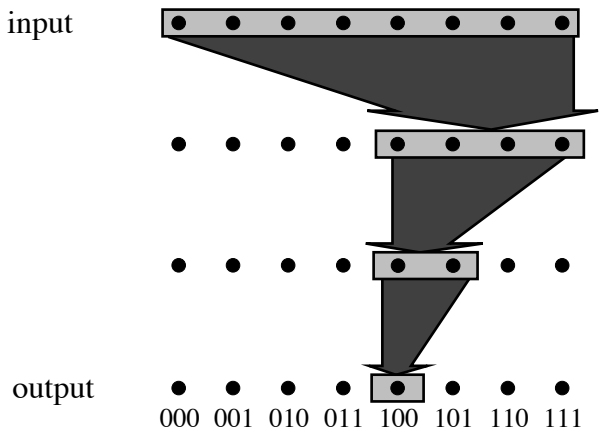

Figure 4: The unique logical path between an input and an output.

The actual algorithm

- `Input: packets` p_0`, ...,` p_n `with destinations j (`$n = 2^d$`)`
- $L = \lceil \frac{1}{2\alpha} \rceil$
- `Partitioning the packets into waves` $A_0, \ldots, A_{L-1}$ `by destination j so that j mod L = i (for` A_i`)`
- `Foreach wave` $A_i \in A_0, \ldots, A_{L-1}$ `do  // the waves`

- While $\exists p_k \in A_i$ do // packets not by the output
  - Foreach node $n_{e/o} \in \{even\ level\}$, $n_{o/e} \in \{odd\ level\}$
    - For j := 1 to $2c$ do // colors of the edges
    - e := 0-/1-edge with color j and incident to $n_{e/o}$
    - if there is no packet at $e.head$
    - send packet p_k over e

The choice of the 0- or 1-edge in the current level depends on the current bit. The routing follows the destination-bit-sequence from the left-most to the right-most bit.

The coloring of the edges prevents congestion in nodes due to receiving two packets in one phase.

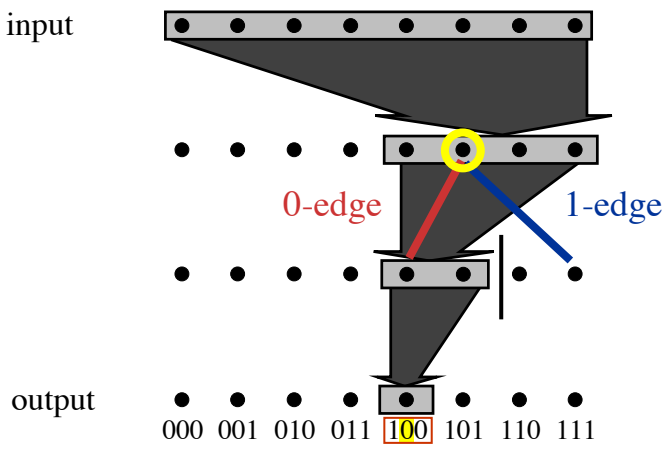

Figure 5: Example of the algorithm

3.2 The analysis

Each phase (even or odd) takes $O(c) = O(1)$, i.e. constant time, because c is a constant.

But how many phases are necessary for one wave?

We have to count the number of phases until all packets of one wave are at the outputs.

The number of phases: Approach

Suppose t $\geq$ 0 the number of phases.

We will analyze the distribution of packets in the network by taking snapshots of the network after t phases.

Let K_i be the number of packets after t phases in level i and K_i' the number of packets after $t+1$ phases in level i.

Assume b_i to be the number of `blocked packets` after $t+1$ phases in level i and a_i the number of `active packets` (recently sent) after $t+1$ phases in level i.

$$\rightarrow \quad K_i^{'} = a_i + b_i$$

Lemma 2 (Number of phases in one wave)

a) $a_0 = 0$ *(no packets are sent to level 0)*

b) $a_i = K_{i-1} - b_{i-1}$ *for $i > 0$: (all packets are sent (active) to level i except for the blocked packets of level $i-1$)*

c) $b_i \leq \frac{1}{\beta+1} \cdot (K_{i+1} + K_i)$ *for $i \leq d$ - 2*

d) $b_{d-1} = 0$*: No packets are blocked in the level second to last.*

Proof of c)

Regarding a submultibutterfly-network with m inputs:
Each wave consists of approximately $\frac{m}{L}$ packets, so

$$\frac{1}{2} \cdot \frac{m}{L} \approx \frac{1}{2} \cdot \frac{m}{\frac{1}{2\alpha}} = \alpha \cdot m$$

packets take the 0- or 1-edge in the (α, β, m, c)-splitter.

Let z be the number of blocked packets in round $t+1$, so z is at most $\alpha \cdot m$.

Due to the expansion-property of the (α, β, m, c)-concentrator these $\alpha \cdot z$ are blocked by at least $\beta \cdot z$ packets. In round $t+1$ there are at most $K_{i+1} + a_{i+1}$ blocking packets on level $i+1$:

$$\begin{aligned} \beta \cdot b_i &\leq K_{i+1} + a_{i+1} =^{b)} K_{i+1} + K_i - b_i \\ \Rightarrow b_i \cdot (1+\beta) &\leq K_{i+1} + K_i \\ \Rightarrow b_i &\leq \frac{1}{1+\beta} \cdot (K_{i+1} + K_i) \end{aligned}$$

□

We will analyze the running time by means of a `potential function argument`:
Suppose $\omega \in (0, 1)$:
The potential function after t phases:

$$\phi_t = \sum_{i=0}^{d-1} (K_i \cdot \omega^i)$$

$\rightarrow$ The packets are `weighted` depending on their distance to the outputs.

Lemma 3 (Potential function properties)

a) $\phi_0 = \frac{n}{L}$

b) *If $\phi_t < \omega^{d-1}$, all packets are by the outputs, the wave completed.*

c) $\phi_{t+1} \leq \phi_t \cdot \left(\frac{\beta}{\beta+1} \omega + \frac{1}{(\beta+1) \cdot \omega} \right)$.

Proof of Lemma a): $\phi_0 = \frac{n}{L}$
At the beginning there are $\frac{n}{L}$ packets in Level 0, i.e. $K_0 = \frac{n}{L}, K_i = 0$ for $i = 1, \ldots, d-1$.
$\rightarrow \phi_0 = \frac{n}{L} \cdot \omega^0 = \frac{n}{L}$

Proof of Lemma b): $\phi_t < \omega^{d-1}$
If $\phi_t < \omega^{d-1}$ then $\phi_t = 0$ and all $K_i = 0$, i.e. all packets are on level d at the outputs.

Proof of Lemma c): $\phi_{t+1} \leq \phi_t \cdot \left(\frac{\beta}{\beta+1} \omega + \frac{1}{(\beta+1) \cdot \omega} \right)$

$$
\begin{aligned}
\phi_{t+1} &= \sum_{i=0}^{d-1} K_i' \omega^i = \sum_{i=0}^{d-1} (b_i + a_i)\omega^i \\
&= b_0 + \underbrace{a_0}_{=0} + \sum_{i=1}^{d-1} (b_i + K_{i-1} - b_{i-1})\omega^i \\
&= \sum_{i=1}^{d-1} K_{i-1}\omega^i + \sum_{i=0}^{d-2} b_i (\underbrace{\omega^i - \omega^{i+1}}_{>0}) + \underbrace{b_{d-1}\omega^{d-1}}_{=0} \\
&\leq \sum_{i=1}^{d-1} K_{i-1}\omega^i + \sum_{i=0}^{d-2} \frac{1}{\beta+1}(K_{i+1} + K_i)(\omega^i - \omega^{i+1}) \\
&= K_0 \cdot \left(\omega^1 + \frac{1}{\beta+1}(\omega^0 - \omega^1)\right) \\
&\quad + K_1 \cdot \left(\omega^2 + \frac{1}{\beta+1}(\omega^0 - \omega^1 + \omega^1 - \omega^2)\right) \\
&\quad .. \\
&\quad .. \\
&\quad + K_i \cdot \left(\omega^{i+1} + \frac{1}{\beta+1}(\omega^{i-1} - \omega^i + \omega^i - \omega^{i+1})\right) \\
&\quad .. \\
&\quad .. \\
&\quad + K_{d-2} \cdot \left(\omega^{d-1} + \frac{1}{\beta+1}(\omega^{d-3} - \omega^{d-2} + \omega^{d-2} - \omega^{d-1})\right) \\
&\quad + K_{d-1} \cdot \left(\frac{1}{\beta+1}(\omega^{d-2} - \omega^{d-1})\right)
\end{aligned}
$$

$$
\begin{aligned}
&= K_0 \cdot \omega^0 \left(\omega + \frac{1}{\beta+1}(1-\omega)\right) \\
&\quad + K_1 \cdot \omega^1 \left(\omega + \frac{1}{\beta+1}(\frac{1}{\omega}-\omega)\right) \\
&\quad .. \\
&\quad .. \\
&\quad + K_i \cdot \omega^i \left(\omega + \frac{1}{\beta+1}(\frac{1}{\omega}-\omega)\right) \\
&\quad .. \\
&\quad .. \\
&\quad + K_{d-2} \cdot \omega^{d-2} \left(\omega + \frac{1}{\beta+1}(\frac{1}{\omega}-\omega)\right) \\
&\quad + K_{d-1} \cdot \omega^{d-1} \left(\frac{1}{\beta+1}(\frac{1}{\omega}-\omega)\right) \\
&\leq \sum_{i=0}^{d-1} K_i \omega^i \left(\omega + \frac{1}{\beta+1}\left(\frac{1}{\omega}-\omega\right)\right) \quad (\frac{1}{\omega} > 1 \; and \; \omega > 0) \\
&= \phi_t \cdot \left(\left(1 - \frac{1}{\beta+1}\right)\omega + \frac{1}{(\beta+1)\omega}\right) \\
&= \phi_t \cdot \left(\frac{\beta}{\beta+1}\omega + \frac{1}{(\beta+1)\omega}\right)
\end{aligned}
$$

□

$\Longrightarrow$ If $\left(\frac{\beta}{\beta+1}\omega + \frac{1}{(\beta+1)\omega}\right) < 1$, ϕ_t *converges to* 0

Theorem 1 (The running time)
Suppose an arbitrary $\beta > 1$, let α and c be chosen, so that there exists a MBN(d, α, β, c) for any d.
So, the MBN(d, α, β, c) can route arbitrary permutations of $n = 2^d$ packets in

$$O(\log n).$$

Proof of Theorem
Assume $\delta := \frac{\beta}{\beta+1}\omega + \frac{1}{(\beta+1)\omega}$
Now we have to choose $\omega \in (0,1) and \; \delta \in (0,1)$:

$$\frac{\beta}{\beta+1}\omega + \frac{1}{(\beta+1)\omega} = \delta$$
$$\Leftrightarrow \omega^2 - \delta \cdot \frac{\beta+1}{\beta} \cdot \omega + \frac{1}{\beta} = 0$$
$$\Leftrightarrow \omega_{1/2} = \frac{1}{2}\frac{\delta(\beta+1)}{\beta} \pm \sqrt{\left(\frac{\delta(\beta+1)}{2\beta}\right)^2 - \frac{1}{\beta}}$$

We choose $\delta = \frac{2\sqrt{\beta}}{\beta+1}$ so that the square-root will be 0 and there will be just one unique solution of the equation.

So, $\omega = \frac{1}{2}\frac{\beta+1}{\beta}\left(\frac{2\sqrt{\beta}}{\beta+1}\right) = \frac{1}{\sqrt{\beta}}.\ (\beta > 1)$
With $\beta > 1$, ω will be less than 1.

At the beginning $\phi_0 = \frac{n}{L}$.
In every phase, ϕ_T decreases, so $\phi_T \leq \frac{n}{L} \cdot \delta^T$.
We have to find the first T with $\phi_T < \omega^{d-1}$: Then T phases are `sufficient` for one wave. (Lemma))

We have to specify $min\{T \mid \phi_T < \omega^{d-1}\}$.

$$\frac{n}{L} \cdot \delta^T < \omega^{d-1}$$
$$\Leftrightarrow \delta^T < \omega^{d-1} \cdot \frac{L}{n}$$
$$\Leftrightarrow \left(\frac{1}{\delta}\right)^T > \left(\frac{1}{\omega}\right)^{d-1} \cdot \frac{n}{L}$$
$$\Leftrightarrow T \cdot \log\left(\frac{1}{\delta}\right) > (d\ -\ 1) \cdot \log\left(\frac{1}{\omega}\right) + \log\left(\frac{n}{L}\right)$$

So,

$$T = \left\lceil \frac{(d\ -\ 1) \cdot \log\left(\frac{1}{\omega}\right) + \log\left(\frac{n}{L}\right)}{log\left(\frac{1}{\delta}\right)} \right\rceil = \ O\left(d + \log\left(\frac{n}{L}\right)\right) = \ O(\log n)$$

phases are sufficient for one wave.

Since there are L waves, the total running-time is

$$O(L \cdot \log\ n) = O\left(\frac{\log\ n}{\alpha}\right) = O(\log\ n).$$

□

3.3 Improvements

Here are some techniques to improve the running-time of the routing:

Eliminating the waves
The waves just `simplified` the analysis of the algorithm. By eliminating the waves the routing could work faster, but the analysis would be much harder.

Queueing
Queueing could be realized by using a `buffer size greater than one`, so each node can store more than one packet. Depending on the destination of the packets the node could choose which packet to send to the next level.

4 Conclusion

As we saw, the Multibutterfly-network can route n packets in $O(\log n)$.
Furthermore the MBN is a higly fault-resistant network, which comes from the expansion-property.

Even `simulations` of the MBN show, that the routing is very fast, but one can say that the `structure` of the MBN is `very complex` compared to other network-structures.