Bisection Bandwidth of Product Networks with Application to Data Centers

Jordi Arjona Aroca
Institute IMDEA Networks and
Universidad Carlos III de Madrid
Madrid, Spain
Email: jorge.arjona@imdea.org

Antonio Fernández Anta
Institute IMDEA Networks
Madrid, Spain
Email: antonio.fernandez@imdea.org

***Abstract*—The bisection width of interconnection networks has always been important in parallel computing, since it bounds the amount of information that can be moved from one side of a network to another, i.e., the bisection bandwidth. Finding the exact value of the bisection width has proven to be challenging for some network families. For instance, the problem of finding the exact bisection width of the multidimensional torus was posed by Leighton [1, Problem 1.281] and has remained open for almost 20 years. In this paper we provide the exact value of the bisection width of the torus, as well as of several d-dimensional classical parallel topologies that can be obtained by the application of the Cartesian product of graphs. To do so, we first provide two general results that allow to obtain upper and lower bounds on the bisection width of a product graph as a function of some properties of its factor graphs. We also apply these results to obtain bounds for the bisection bandwidth of a d-dimensional BCube network, a recently proposed topology for data centers.**

***Keywords*-Torus, bisection width, product graphs, complete binary trees, extended trees, mesh-connected trees.**

I. Introduction

Bisection width and bisection bandwidth of interconnection networks have always been two important parameters in parallel computing. The first one reflects the smallest number of links which have to be removed to split a network in two equal parts, while the second one bounds the amount of information that can be moved from one side of a network to another. Finding the exact value of the bisection width has proven to be challenging for some specific networks. For instance, the problem of finding the exact bisection width of the multidimensional torus was posed by Leighton [1, Problem 1.281] and has remained open for almost 20 years. Beyond interconnection networks for parallel architectures, the bisection width is now also being used as a reference parameter on the analysis of the latest topologies that are being deployed in data centers. This can be seen in recent papers which propose new topologies like BCube[2] or DCell [3]. The bisection width is used to compare these new topologies with basic ones, like grids, tori, and hypercubes, or with currently used topologies, like trees and fat trees.

Basic interconnection topologies such as grids, tori, hypercubes, $\ldots$, have also generated interest among the researchers from the On-chip Interconnection Networks field lately [4][5][6][7]. These basic topologies are being revisited and the solution for these unsolved problems can also be important for the future of the On-Chip Networking.

In the construction of interconnection networks, it has proven to be useful to use the Cartesian product operation of graphs to build large networks from smaller factor networks. In particular, many popular interconnection networks are instances of graphs obtained by the application of the Cartesian product. Some examples are the grid, the torus and the hypercube. In this paper we derive a technique to bound the bisection width of several instances of product graphs. In particular we present exact values for graphs obtained as products of paths, rings, complete binary trees (CBT), and extended complete binary trees (XT) [8]. The same techniques are used for BCube.

A. Related Work

Youssef in [9], [10] presented the idea of working with product networks as a divide and conquer problem. Youssef exposes some important properties of product graphs in terms of the properties of their factor graphs.

The bisection width of arrays and tori was explored in [1], providing exact values for these networks with even number of nodes per dimension, but leaving open the case when there are odd number of nodes per dimension. Rolim et al. [11] gave some first exact results for the bisection width of 2 and 3-dimensional arrays and torus (we do not include these formulas for the sake of brevity) but left open the question for longer number of dimensions.

For the special case in which all the factors are isomorphic, Efe and Fernández [12] provided a lower bound for the bisection width of a product graph as $BW(PG_r(k)) = \frac{k^{r+1}}{2C}$, being $PG_r(k)$ the cartesian product of r graphs $G(k)$, and being $G(k)$ a k-nodes graph and C the maximal congestion of $G(k)$. (The maximal congestion is a new parameter of graphs defined by them.). Also Nakano [13] presented an interesting analysis for the bisection width of isomorphic d-dimensional arrays and cliques, providing exact results for both of them. Concretely, he proved that the bisection width of a isomorphic d-dimensional clique K_k^d, where k is the number of nodes of the complete graph used as factor and d is the number of dimensions, is k^{d+1}, when k is even, and $(k+1)\frac{(k^d-1)}{4}$, when k is odd. The result provided for the bisection width of a isomorphic d-dimensional array A_k^d is k^{d-1} when k is even, and $\frac{(k^d-1)}{(k-1)}$ when k is odd.

The exact bisection width of the d-dimensional grid with k nodes in each dimension, being k an odd number, was also found to be $\frac{k^d-1}{k-1}$ independently by Efe and Feng[14].

For the present paper it is very important the work by Azizoglu and Egecioglu. In [15] and [16] they studied the relations between the isoperimetric number and the bisection width of different product networks. In the former paper, they find the exact value of the bisection width of the generalized cylinders with an even number of nodes in its largest dimension. These networks are products of paths and rings. In the latter one, they find the exact bisection width of a d-dimensional array $A^{(d)} = P_{k_1} \times P_{k_2} \times \cdots \times P_{k_d}$, with k_i nodes along dimension i, and where $k_1 \leq k_2 \leq \ldots \leq k_d$. The value of the bisection width they found is $BW(A^{(d)}) = \sum_{i=\alpha}^{d} C_i$, α being the largest index for which k_α is even and $C_i = k_{i-1}k_{i-2}\ldots k_1$.

B. Contributions

In this paper we provide the exact value of the bisection width of several families of product graphs. To do so, we first provide two general results that allow to obtain upper and lower bounds on the bisection width of the product graph as a function of some properties of the factor graphs. The application of these results for several families yield the exact values.

One of the most interesting contribution of this paper is the exact value bisection width of the torus, since, as mentioned above, this problem has been open for almost 20 years. We find here that the exact value of the bisection width of a d-dimensional torus $T^{(d)} = R_{k_1} \times R_{k_2} \times \cdots \times R_{k_d}$, with k_i nodes along dimension i, and where $k_1 \geq k_2 \geq \ldots \geq k_d$, is $BW(T^{(d)}) = 2\sum_{i=1}^{\alpha} C_i$, α being the smallest index for which k_e is even and $C_i = k_{i+1}k_{i+2}\cdots k_d$.

In addition to this result, we provide the exact value for the bisection width of products of complete binary trees of any size (mesh connected trees), products of extended complete binary trees, products of CBTs and paths, and products of extended complete binary trees and rings as well as lower and upper bounds for BCube.

The rest of the paper is organized as follows. Section II presents some basic definitions used in the rest of sections. In Section III we provide some bounds on the bisection bandwidth of product graphs and the values of congestion and central cut of different basic graphs, which are used to find the bisection bandwidth of multidimensional networks. Section IV and Section V present our results for the bisection bandwidth of some d-dimensional classical parallel topologies, respectively for products of complete binary trees and paths and for products of rings and extended complete binary trees. Bounds for the bisection bandwidth of BCube are presented in Section VI. Finally, in Section VII we present our conclusions and some interesting open problems.

II. Definitions

A. Graphs and Bisections

In this section we present definitions and notation that will be used along the text. Given a graph G, we denote its sets of vertices and edges as $V(G)$ and $E(G)$, respectively. In some cases, when it is clear from the context, only V or E will be used, omitting the graph G. Unless otherwise stated, the graphs considered are undirected.

Given a graph G with n nodes, we use $S(G)$ to denote a subset of $V(G)$ such that $|S(G)| \leq \frac{n}{2}$. We also use $\partial^G S(G)$ to denote the set of edges connecting $S(G)$ and $V(G)\backslash S(G)$. Formally, $\partial^G S(G) = \{(u,v) \in E(G) : u \in S(G), v \in G \setminus S(G)\}$. The graph G may be omitted from this notation when it is clear from the context.

The main object of these work is to calculate the *bisection width* and *bisection bandwidth* of different product networks. These bisections can be defined as follows.

Definition 1: The *bisection width* of an n-node graph G, denoted $BW(G)$, is the smallest number of edges which have to be removed from G to partition it in two halves. Formally, $BW(G) = \min_{S:|S|=\lfloor \frac{n}{2} \rfloor} |\partial^G S|$.

Definition 2: The bisection bandwidth of a network/graph G, denoted $BBW(G)$, is the minimal amount of traffic which can be transferred between any two halves of the network when its links are transmitting at full speed.

In general, for regular graphs, we can assume that the relation between the bisection bandwidth and the bisection width is $BBW(G) = BW(G) \cdot T$, where T is the capacity of the links involved in the bisection bandwidth. We are assuming, for simplicity, that all the links in the graph will have the same capacity T.

B. Factor and Product Graphs

We define first the Cartesian product of graphs.

Definition 3: The *d-dimensional Cartesian product* of graphs $G_1, G_2, ..., G_d$, denoted $G_1 \times G_2 \times \cdots \times G_d$, is the graph with vertex set $V(G_1) \times V(G_2) \times \cdots \times V(G_d)$, in which vertices $(u_1, ..., u_i, ..., u_d)$ and $(v_1, ..., v_i, ..., v_d)$ are adjacent if and only if $(u_i, v_i) \in E(G_i)$ and $u_j = v_j$ for all $j \neq i$.

The graphs $G_1, G_2, ..., G_d$ are called the *factors* of $G_1 \times G_2 \times \cdots \times G_d$. Observe that $G_1 \times G_2 \times \cdots \times G_d$ contains $\prod_{j \neq i} |V(G_j)|$ disjoint copies of G_i, which form dimension i. We define now some of the basic factor graphs that will be used in this and in the following sections.

Definition 4: The *path* of k vertices, denoted P_k, is a graph such that $V(P_k) = \{0, 1, \ldots, k-1\}$ and where $E(P_k) = \{(i, i+1) : i \in [0, k-2]\}$.

Definition 5: The *complete graph* (a.k.a. the clique) of k vertices, denoted K_k, is a graph such that $V(K_k) =$

$\{0, 1, \ldots, k-1\}$ and where $E(K_k) = \{(i,j) : (j \neq i) \wedge (i,j \in V(K_k))\}$.

Definition 6: The *r-complete graph* of k vertices denoted rK_k, is a graph such that $V(rK_k) = \{0, 1, \ldots, k-1\}$ and where $E(rK_k)$ is a multiset such that each pair of vertices $i, j \in V(rK_k)$ are connected with r parallel edges. (i.e., each $e \in E(rK_k)$ has multiplicity r).

Using these and other graphs as factors, we will define, across the text, different d-dimensional Cartesian product graphs. For convenience, for these graphs we will use the general notation $G^{(d)}_{k_1,k_2,\ldots,k_d}$, where G is the name of the graph, the superscript (d) means that it is a d-dimensional graph, and $k_1, k_2, \ldots k_d$ are the number of vertices in each dimension. (Superscript and subscripts may be omitted when clear from the context.) It will always hold that $k_1 \geq k_2 \geq \ldots \geq k_d$, i.e., the factor graphs are sorted by decreasing number of vertices. We will often use n to denote the number of nodes of the graph $G^{(d)}_{k_1,k_2,\ldots,k_d}$, i.e., $n = k_1 k_2 \cdots k_d$ and we will always use α to denote the index of the lowest dimension with an even number of vertices (if there is no such dimension, $\alpha = d$, where d is the index of the lowest dimension). According to this notation we will present different d-dimensional product graphs as follows.

Definition 7: The *d-dimensional array*, denoted $A^{(d)}_{k_1,k_2,\ldots,k_d}$, is the Cartesian product of d paths of $k_1, k_2, \ldots, k_d$ vertices, respectively. I.e., $A^{(d)}_{k_1,k_2,\ldots,k_d} = P_{k_1} \times P_{k_2} \times \cdots \times P_{k_d}$.

Definition 8: The *d-dimensional r-Hamming graph*, denoted $rH^{(d)}_{k_1,k_2,\ldots,k_d}$, is the Cartesian product of d r-complete graphs of $k_1, k_2, \ldots, k_d$ nodes, respectively. I.e., $rH^{(d)}_{k_1,k_2,\ldots,k_d} = rK_{k_1} \times rK_{k_2} \times \cdots \times rK_{k_d}$.

Observe that the *Hamming graph* [17] is a particular case of the r-Hamming graph, with $r = 1$. For brevity, we use $H^{(d)}_{k_1,k_2,\ldots,k_d}$ instead of $1H^{(d)}_{k_1,k_2,\ldots,k_d}$, to denote the Hamming graph.

C. Boundaries and Partitions

We define now the dimension-normalized boundary [16].

Definition 9: Let $G^{(d)}_{k_1,k_2,\ldots,k_d}$ be a d-dimensional product graph and $S(G)$ a subset of $V(G)$. Then, the *dimension-normalized boundary* of $S(G)$, denoted $B_G(S)$, is defined as

$$B_G(S) = \frac{|\partial^G_1 S|}{\sigma_1} + \frac{|\partial^G_2 S|}{\sigma_2} + \ldots + \frac{|\partial^G_d S|}{\sigma_d}, \qquad (1)$$

where, for each $i \in [1, d]$, ∂^G_i is ∂^G applied to the dimension i of G and

$$\sigma_i = \begin{cases} k_i^2 & \text{if } k_i \text{ is even} \\ k_i^2 - 1 & \text{if } k_i \text{ is odd.} \end{cases} \qquad (2)$$

Observation 1: For $rH^{(d)}_{k_1,k_2,\ldots,k_d}$, any subset S of nodes, and any dimension i, it holds that $|\partial^{rH}_i S| = r \cdot |\partial^H_i S|$. Hence,

$$\begin{aligned} B_{rH}(S) &= \frac{|\partial^{rH}_1 S|}{\sigma_1} + \cdots + \frac{|\partial^{rH}_d S|}{\sigma_d} \\ &= r\left(\frac{|\partial^H_1 S|}{\sigma_1} + \cdots + \frac{|\partial^H_d S|}{\sigma_d}\right) \\ &= rB_H(S). \end{aligned}$$

Let us define the lexicographic-order. Consider graph $H^{(d)}_{k_1,k_2,\ldots,k_d}$, we say that vertex $x = (x_1, x_2, \ldots, x_d)$ precedes vertex $y = (y_1, y_2, \ldots, y_d)$ in *lexicographic-order* if there exists an index i such that $x_i < y_i$ and $x_j = y_j$ for all $j < i$. Azizoglu and Egecioglu [17] proved the following result.

Theorem 1 ([17]): Given a d-dimensional Hamming graph $H^{(d)}_{k_1,k_2,\ldots,k_d}$, with $k_1 \geq k_2 \geq \cdots \geq k_d$. Let S be any subset of $V(H)$ and $\bar{S}$ the set of first $|S|$ vertices of H in lexicographic-order[1]. Then $B_H(\bar{S}) \leq B_H(S)$.

We finally define the function $\Psi(x)$ as

$$\Psi(x) = \sum_{i=1}^{x} C_i = \prod_{j=i+1}^{d} k_j, \qquad (3)$$

which corresponds to the result of the bisection width of $A^{(d)}_{k_1,k_2,\ldots,k_d}$, provided by [16], when x is equal to α. This function will be used in following sections.

III. Bounds on the bisection width of product graphs

In this section we present general bounds on the bisection width of product graphs as well as presenting two important parameters, the normalized congestion and the central cut, which are used to obtain them. These bounds will be used in the upcoming sections to find the bisection width of several instances of product graphs.

A. Lower bound

We start by defining the normalized congestion of a graph. Let G be a graph with n nodes. Then, an *embedding* of graph rK_n onto G is a mapping of the edges of rK_n into paths in G. We define the *congestion of G with multiplicity r*, denoted $m_r(G)$, as the minimum over all such embeddings of the maximum number of embedded paths that contain an edge from G. To formally define this concept, we first define the congestion of an edge $e \in E(G)$ under the embedding M_r of rK_n onto G, denoted $c_{M_r}(e)$, as

$$c_{M_r}(e) = |\{e' \in E(rK_n) : e \in M_r(e')\}|. \qquad (4)$$

(Observe that $M_r(e') \subseteq E(G)$ is a path in G.) Then, the congestion $m_r(G)$ is

$$m_r(G) = \min_{M_r \in \mathcal{E}} \max_{e \in E(G)} \{c_{M_r}(e)\}, \qquad (5)$$

[1]Observe that, in Theorem 1, we have reversed the ordering of dimensions with respect to the original theorem from Azizoglu and Egecioglu.

where $\mathcal{E}$ is the set of all possible embeddings of rK_n onto G.

Then, using Eqs. 5 and 2, we define the normalized congestion of G as

$$\beta_r(G) = \frac{m_r(G)}{\sigma}. \quad (6)$$

Having defined the normalized congestion, we proceed to extend Theorem 1 to r-Hamming graphs.

Theorem 2: Given a d-dimensional r-Hamming graph $rH^{(d)}$, let S be any vertex subset of $V(rH^{(d)})$ and $\bar{S}$ the set of first $|S|$ vertices of $rH^{(d)}$ in lexicographic-order. Then $B_{rH}(\bar{S}) \leq B_{rH}(S)$.

Proof: We prove the theorem by contradiction. Assume that there is a set of vertices $S \neq \bar{S}$ such that $|S| = |\bar{S}|$ and $B_{rH}(\bar{X}) > B_{rH}(X)$. Then, applying Observation 1 to both S and $\bar{S}$, we obtain that

$$B_H(\bar{S}) = \frac{B_{rH}(\bar{S})}{r} > \frac{B_{rH}(S)}{r} = B_H(S), \quad (7)$$

which contradicts Theorem 1 and proves the theorem. ■

We now present the following lemma.

Lemma 1: Let $\bar{S}$ be a subset of the vertices of graph $rH^{(d)}_{k_1,k_2,\ldots,k_d}$, such that $\bar{S}$ are the first $\lfloor \frac{n}{2} \rfloor$ vertices of rH in lexicographical order, and n is the number of vertices of rH. Then, the dimension-normalized boundary of $\bar{S}$ is

$$B_{rH}(\bar{S}) = \frac{r}{4}\Psi(\alpha).$$

Proof: We will derive first the value of $B_H(\bar{S})$, and then use Observation 1 to prove the claim. It was shown in [16], that $\partial_i^H \bar{S} = \emptyset$ for all $i > \alpha$.[2] The number of edges in each dimension $i \in [1, \alpha]$ on the boundary of $\bar{S}$ in H is

$$|\partial_i^H \bar{S}| = \begin{cases} \frac{k_i}{2}(\prod_{j=i+1}^{d} k_j)\frac{k_i}{2} & \text{if } k_i \text{ is even} \\ \frac{k_i-1}{2}(\prod_{j=i+1}^{d} k_j)\frac{k_i+1}{2} & \text{if } k_i \text{ is odd.} \end{cases} \quad (8)$$

Then, from the definition of $B_H(\bar{S})$, we obtain that

$$\begin{aligned} B_H(\bar{S}) &= \frac{\frac{k_1-1}{2}(\prod_{j=2}^{d} k_j)\frac{k_1+1}{2}}{k_1^2-1} + \frac{\frac{k_2-1}{2}(\prod_{j=3}^{d} k_j)\frac{k_2+1}{2}}{k_2^2-1} \\ &\quad + \cdots + \frac{\frac{k_\alpha}{2}(\prod_{j=\alpha+1}^{d} k_j)\frac{k_\alpha}{2}}{k_\alpha^2} \\ &= \frac{\prod_{j=2}^{d} k_j}{4} + \frac{\prod_{j=3}^{d} k_j}{4} + \cdots + \frac{\prod_{j=\alpha+1}^{d} k_j}{4} \\ &= \frac{\sum_{i=1}^{\alpha} C_i}{4} = \frac{\Psi(\alpha)}{4}. \end{aligned}$$

Finally, from Observation 1, we derive

$$B_{rH}(\bar{S}) = rB_H(\bar{S}) = \frac{r}{4}\Psi(\alpha).$$

■

[2] Observe that they use reverse lexicographic order and sort dimensions in the opposite order we do.

Using Definition 3, Lemma 1, and Eq. (5), we obtain the following theorem.

Theorem 3: Let $G = G_1 \times \ldots \times G_d$, where $|V(G_i)| = k_i$ and $k_1 \geq k_2 \geq \ldots \geq k_d$. Let $\beta_r(G_i)$ be the normalized congestion of G_i with multiplicity r, for $i \in [1, d]$. Consider any subset $S \subset V(G)$ and the subset $\bar{S}$ which contains the first $|S|$ vertices of G, in lexicographic order. Then,

$$B_{rH}(\bar{S}) \leq \sum_{i=1}^{d} \beta_r(G_i)|\partial_i^G S|$$

Proof: First, observe that, for any $S_i \subset V(G_i)$,

$$|\partial^{rK_{k_i}} S_i| \leq m_r(G_i) \cdot |\partial^{G_i} S_i|. \quad (9)$$

Then, for $S \subset V(G)$ as defined,

$$|\partial_i^{rH} S| \leq m_r(G_i) \cdot |\partial_i^G S|.$$

Finally, using Theorem 2, we can state that

$$\begin{aligned} B_{rH}(\bar{S}) &\leq B_{rH}(S) \\ &\leq m_r(G_1)\frac{|\partial_1^G S|}{c_1} + \cdots + m_r(G_d)\frac{|\partial_d^G S|}{c_d} \\ &= \beta_r(G_1)|\partial_1^G S| + \cdots + \beta_r(G_i)|\partial_d^G S|. \end{aligned}$$

■

From this theorem, we derive a corollary for the case of $|S| = \lfloor \frac{n}{2} \rfloor$:

Corollary 1: Let $G = G_1 \times \ldots \times G_d$, where $|V(G_i)| = k_i$ and $k_1 \geq k_2 \geq \ldots \geq k_d$. Let $\beta_r(G_i)$ be the normalized congestion of G_i for any r, for $i \in [1, d]$. Consider any subset $S \subset V(G)$ such that $|S| = \lfloor \frac{|V(G)|}{2} \rfloor$. Then

$$\frac{r}{4}\Psi(\alpha) \leq \sum_{i=1}^{d} \beta_r(G_i)|\partial_i^G S|.$$

B. Upper bound

Having proved the lower bound of the bisection width, we follow with the upper bound. We define first the central cut of a graph G.

Consider a graph G with n nodes, and a partition of $V(G)$ into three sets S^-, S^+, and S, such that $|S^-| = |S^+| = \lfloor \frac{n}{2} \rfloor$ (observe that if n is even then $S = \emptyset$, otherwise $|S| = 1$). Then, the *central cut of* G, denoted $CC(G)$, is

$$\min_{\{S^-, S^+, S\}} \max\{|\partial^G S^-|, |\partial^G S^+|\}.$$

Observe that, for even n, the central cut is the bisection width.

Now we use the definition of central cut in the following theorem.

Theorem 4: Let $G = G_1 \times \ldots \times G_d$, where $|V(G_i)| = k_i$ and $k_1 \geq k_2 \geq \ldots \geq k_d$. Then,

$$BW(G) \leq \max_i \{CC(G_i)\} \cdot \Psi(\alpha).$$

Proof: It was shown in [16] how to bisect $A^{(d)}$ by cutting exactly $BW(A^{(d)}) = \Psi(\alpha)$ links. Furthermore, this bisection satisfies that, if the paths P_{k_i} in dimension i are cut, each of them can be partitioned into subpaths P^+ and P^- of size $\lfloor \frac{k_i}{2} \rfloor$ (connected by a link if k_i is even or by a node with links to both if k_i is odd) so that the cut separates P^+ or P^- from the rest of the path. Each path is then cut by removing one link. We map the sets S^+ and S^+ of the partition that gives the central cut of G_i to P^+ and P^-, respectively. Then, any cut of a paths P_{k_i} in dimension i becomes a cut of G_i with at most $CC(G_i)$ links removed.

Then, if S is the subset of $V(G)$ that ends at one side of the bisection described above, we have that

$$\frac{|\partial_i^G S|}{CC(G_i)} \leq |\partial_i^{A^{(d)}} S|, \quad (10)$$

which also holds if the paths in dimension i are not cut. Applying this to all dimensions, we obtain

$$\frac{|\partial_1^G S|}{CC(G_1)} + \cdots + \frac{|\partial_d^G S|}{CC(G_d)} \leq BW(A^{(d)}) = \Psi(\alpha). \quad (11)$$

This yields,

$$BW(G) \leq |\partial_1^G S| + \cdots + |\partial_d^G S| \leq \max_i \{CC(G_i)\} \cdot \Psi(\alpha),$$

proving Theorem 4. ■

IV. Bisection width of products of CBTs and paths

In this section we will obtain the bisection bandwidth of product graphs which result from the Cartesian product of paths and CBTs. We will present, first, the different factor graphs we are using and the product graphs we are bisecting, then, we will compute the congestion and central cut of these factor graphs and, finally, calculate the bisection width of these product graphs.

A. Factor and product graphs

In this section we will work with paths, which were defined in Section II, and CBTs, which we define now.

Definition 10: The *complete binary tree* of k vertices, denoted CBT_k, is a graph such that $V(CBT_k) = \{1, 2, \ldots, k\}$, with $k = 2^j - 1$ (j is the number of levels of the tree), and where $E(CBT_k) = \{(i, j) : ((j = 2i) \vee (j = 2i + 1)) \wedge (i \in [1, 2^{j-1} - 1])\}$.

Combining these factor graphs through the Cartesian product, we obtain the product networks that we define below.

Definition 11: A *d-dimensional mesh-connected trees and paths*, denoted $MCTP^{(d)}_{k_1,k_2,\ldots,k_d}$, is the Cartesian product of d graphs of $k_1, k_2, \ldots, k_d$ vertices, respectively, where each factor graph is a complete binary tree or a path. I.e., $MCTP^{(d)}_{k_1,k_2,\ldots,k_d} = G_{k_1} \times G_{k_2} \times \cdots \times G_{k_d}$, where either $G_{k_i} = CBT_{k_i}$ or $G_{k_i} = P_{k_i}$.

We also define the *d-dimensional mesh-connected trees* [8], denoted $MCT^{(d)}_{k_1,k_2,\ldots,k_d}$ as the graph $MCTP^{(d)}_{k_1,k_2,\ldots,k_d}$ in which all the factor graphs are complete binary trees. (Observe that the array is also the special case of $MCTP^{(d)}_{k_1,k_2,\ldots,k_d}$ in which all the factor graphs are paths.)

B. Congestion and central cut of paths and CBTs

The bisection widths of the aforementioned product graphs can be calculated using the bounds defined in Section III. To do so, we need to compute first the values of the normalized congestion and central cut of their factor graphs, it is, of a path and of a CBT.

We will start by computing the congestion of a path and of a CBT and, then, their central cuts. We present the following lemma.

Lemma 2: The congestion of P_k with multiplicity r, denoted $m_r(P_k)$, has two possible values, depending on whether the number of vertices k is even or odd, as follows,

$$m_r(P_k) = \begin{cases} r\frac{k^2}{4} & \text{if } k \text{ is even} \\ r\frac{k^2-1}{4} & \text{if } k \text{ is odd} \end{cases} \quad (12)$$

Proof: This proof is illustrated in Figure 1 where it can be seen that there are two possible cases, depending on whether k is even or odd. The congestion $m_r(P_k)$ is defined as the minimum congestion over all embeddings of rK_k onto P_k. As there is only one possible path between every pair of vertices, the congestion of an edge will always be the same for any embedding M_r of rK_k into P_k. Let M_r be an embedding of rK_k onto P_k. Then,

$$m_r(P_k) = \min_{M \in \mathcal{E}} \max_{e \in E(P_k)} \{c_M(e)\} = \max_{e \in P_k} \{c_{M_r}(e)\}. \quad (13)$$

If we fix $e = (i, i+1) \in E(P_k)$, $i \in [0, k-1]$, the congestion of e follows the equation:

$$c_{M_r}(e) = r(i+1)(k-i-1). \quad (14)$$

The value of i that maximizes $c_{M_r}(e)$ is $i = \frac{k}{2} - 1$. As k is an integer, depending on whether k is even or odd, $\frac{k}{2}$ will be exact or not. Hence, we consider two possible cases,

$$i = \begin{cases} \frac{k}{2} - 1 & \text{if } k \text{ is even} \\ \frac{k-1}{2} - 1 & \text{if } k \text{ is odd.} \end{cases} \quad (15)$$

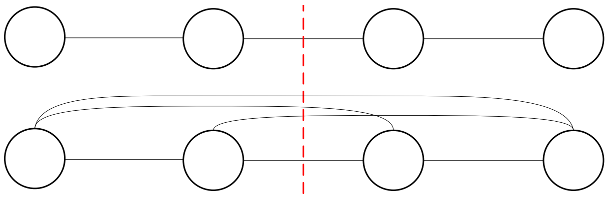
Using these values in Eq. (14) leads to the final result

$$m_r(P_k) = \begin{cases} r\frac{k^2}{4} & \text{if } k \text{ is even} \\ r\frac{k^2-1}{4} & \text{if } k \text{ is odd} \end{cases}$$

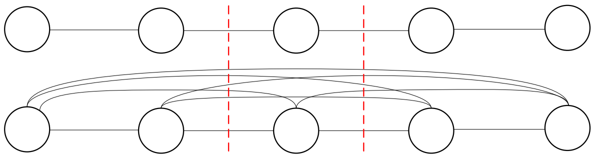
■

Corollary 2: The normalized congestion of a path is $\beta_r(P_k) = r\frac{1}{4}$.

The value of the congestion of a CBT will be exactly the same obtained for a path with an odd number of nodes. CBTs share the property of the path of having only one possible routing between two nodes. As can be seen in Figure 2,



(a) The 4-vertex path and the 4-vertex clique



(b) The 5-vertex path and the 5-vertex clique

Figure 1. Paths and possible cuts

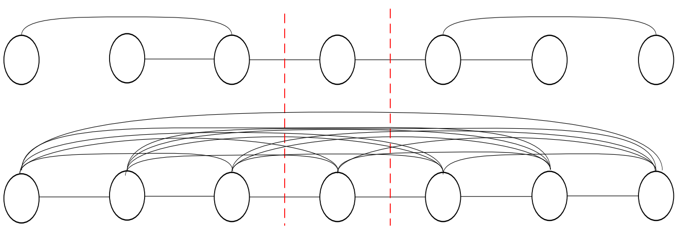


Figure 2. The 7-vertex complete binary tree and the 7-vertex clique, with their possible cuts

the possible cuts are similar. We present Lemma 3 for the congestion of a CBT.

Lemma 3: The congestion of CBT_k with multiplicity r, denoted $m_r(CBT_k)$ is

$$m_r(CBT_k) = r\frac{k^2-1}{4} \quad (16)$$

Proof: Let CBT_{2^j-1} be a complete binary tree of j levels with $k = 2^j - 1$ nodes. Whichever edge we cut results on two parts, one of them being another complete binary tree, let us call it A and assume it has $l < j$ levels; and the other being the rest of the previous complete binary tree, let us call it B. The number of nodes in A will be $2^l - 1$ while the number of nodes in B will be $k - 2^l + 1$. For any embedding M of rK_k into CBT_k, the congestion of any edge e follows the equation

$$c_{M_r}(e) = r(2^l - 1)(k - 2^l + 1). \quad (17)$$

The value of l which maximizes the equation is $l = j-1$, which equivalent to cut one of the links of the root. This divides the tree into subgraphs of sizes $\frac{k+1}{2}$ and $\frac{k-1}{2}$. Then, the final value for congestion will be

$$m_r(CBT_k) = r\frac{k^2-1}{4}$$

■

Corollary 3: The normalized congestion of a CBT is $\beta_r(CBT_k) = r\frac{1}{4}$.

The central cut of both the path and CBT can be easily deduced from Figures 1(a),1(b) and 2, being $CC(P_k) = CC(CBT_k) = 1$.

C. Bounds for the bisection width of products of CBTs and paths

Having computed both the congestion and the central cut of the possible factor graphs, we can calculate now the lower and upper bound for the bisection width of a product of CBTs and paths. We will start by the lower bound of the bisection width.

Lemma 4: The bisection width of a d-dimensional mesh-connected trees and paths, $MCTP^{(d)}$, is lower bounded by $\Psi(\alpha)$.

Proof: As this graph is the result of a Cartesian product of d CBTs and paths, we can apply Corollary 1, so

$$\frac{r}{4}\Psi(\alpha) \le \sum_{i=1}^{d} \beta_r(G_{k_i})|\partial_i^{MCTP} S| \quad (18)$$

when $|S| = \lfloor \frac{n}{2} \rfloor$ and G_{k_i} is either a path or a CBT.

As we can see in Corollaries 2 and 3, the normalized congestion of both factor graphs is the same, so it follows that

$$\frac{r}{4}\Psi(\alpha) \le \sum_{i=1}^{d} \frac{r}{4}|\partial_i^{MCTP} S|, \quad (19)$$

which yields,

$$BW(MCTP^{(d)}) = \sum_{i=1}^{d} |\partial_i^{MCTP} S| \ge \Psi(\alpha). \quad (20)$$

■

We follow now by presenting an upper bound for the bisection width of d-dimensional mesh-connected trees and paths.

Lemma 5: The bisection width of a d-dimensional mesh-connected trees and paths, $MCTP^{(d)}$, is upper bounded by $\Psi(\alpha)$.

Proof: Obviously, as this graph can also be embedded into a d-dimensional array, we can use Theorem 4. We know that the central cut of both CBTs and paths is 1 independently of their sizes or number of levels, and hence also $\max_i\{CC(G_{k_i})\} = 1$ (where G_{k_i} is either a CBT or a path). Then,

$$BW(MCTP^{(d)}) \le \Psi(\alpha). \quad (21)$$

■

From the results obtained from Lemma 4 and Lemma 5 the proof of Theorem 5 follows.

Theorem 5: The bisection width of a d-dimensional mesh-connected trees and paths $MCTP^{(d)}_{k_1,k_2,\ldots,k_d}$ is $\Psi(\alpha)$.

We can also present the following corollary for the particular case of the d-dimensional mesh-connected trees $MCT^{(d)}_{k_1,k_2,\ldots,k_d}$.

Corollary 4: The bisection width of the d-dimensional mesh-connected trees $MCT^{(d)}_{k_1,k_2,\ldots,k_d}$ is $BW(MCT^{(d)}) = \Psi(d)$.

V. Products of rings and extended trees

Similarly to what was done in Section IV, in this section we will obtain a result for the bisection bandwidth of the product graphs which result from the Cartesian product of rings and extended complete binary trees, a.k.a. XTs.

A. Factor and product graphs

The factor graphs which are going to be used in this section are rings and XTs. We define them below.

Definition 12: The *ring* of k vertices, denoted R_k, is a graph such that $V(R_k) = \{0, 1, \ldots, k-1\}$ and where $E(R_k) = \{(i, (i+1) \mod k) : i \in V(R_k)\}$.

Definition 13: The *extended complete binary tree* (a.k.a. XT) of k vertices, denoted X_k, is a complete binary tree in which the leaves are connected as a path. More formally, $V(X_k) = V(CBT_k)$ and $E(X_k) = E(CBT_k) \cup \{(i, i+1) : i \in [2^{j-1}, 2^j - 2]\}$.

Combining these graphs as factor graphs in a Cartesian product, we can obtain the three following different kinds of product graphs:

Definition 14: A *d-dimensional mesh-connected extended trees and rings*, denoted $MCXR^{(d)}_{k_1,k_2,\ldots,k_d}$, is the Cartesian product of d graphs of $k_1, k_2, \ldots, k_d$ vertices, respectively, where each factor graph is a extended complete binary tree or a ring. I.e., $MCXR^{(d)}_{k_1,k_2,\ldots,k_d} = G_{k_1} \times G_{k_2} \times \cdots \times G_{k_d}$, where either $G_{k_i} = X_{k_i}$ or $G_{k_i} = R_{k_i}$.

Definition 15: The *d-dimensional torus*, denoted $T^{(d)}_{k_1,k_2,\ldots,k_d}$, is the Cartesian product of d rings of $k_1, k_2, \ldots, k_d$ vertices, respectively. I.e., $T^{(d)}_{k_1,k_2,\ldots,k_d} = R_{k_1} \times R_{k_2} \times \cdots \times R_{k_d}$.

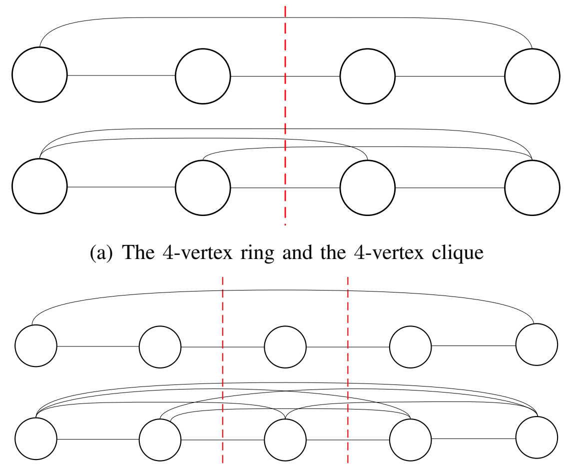
And, as happened in Section IV with $MCT^{(d)}$, we also define the *d-dimensional mesh-connected extended trees*, denoted denoted $MCX^{(d)}_{k_1,k_2,\ldots,k_d}$, a special case of $MCXR^{(d)}_{k_1,k_2,\ldots,k_d}$ in which all factor graphs are extended complete binary trees. (The torus is the special case of $MCXR^{(d)}_{k_1,k_2,\ldots,k_d}$ in which all factor graphs are rings.)

B. Congestion and central cut of rings and XTs

The congestion and central cut of both a ring and an XT are needed to calculate the bounds obtained in Section III. We present the following lemma for the congestion of a ring.

Lemma 6: The congestion of R_k with multiplicity $r = 2$, denoted $m_r(R_k)$, has two possible upper bounds depending on whether the number of vertices k is even or odd, as follows,

$$m_2(R_k) \leq \begin{cases} 1\frac{k^2}{4} & \text{if } k \text{ is even} \\ 1\frac{k^2-1}{4} & \text{if } k \text{ is odd} \end{cases} \quad (22)$$



(a) The 4-vertex ring and the 4-vertex clique

(b) The 5-vertex ring and the 5-vertex clique

Figure 3. Rings and possible cuts

Proof: While a path had only one possible routing, for R_k we have two possible routes connecting each pair of nodes, what implies that the congestion of an edge depends on the mapping of rK_k into R_k.

If we embed rK_k, for $r = 2$, into R_k, we can route each of the parallel edges connecting two nodes through each of the possible routings. This yields,

$$m_2(R_k) \leq \begin{cases} 2\frac{\frac{k}{2}\frac{k}{2}}{2} = \frac{k^2}{4} & \text{if } k \text{ is even} \\ 2\frac{\frac{k-1}{2}\frac{k+1}{2}}{2} = \frac{k^2-1}{4} & \text{if } k \text{ is odd.} \end{cases}$$

■

Corollary 5: The normalized congestion of a ring is $\beta_2(R_k) = \frac{1}{4}$.

Similarly to what happened with paths and CBTs, the congestion of rings and XTs is the same. The extended complete binary tree X_k has a Hamiltonian cycle [8], so we can find a ring R_k contained onto it. Consequently, the congestion of an XT and a ring with the same number of nodes will be the same. Then, the normalized congestion of both factor graphs will also be the same.

Corollary 6: The normalized congestion of an XT is $\beta_2(X_k) = \frac{1}{4}$.

Due to these similarities, central cuts of both graphs are also going to be the same. As can be easily deduced from Figures 3(a),3(b) and 4, $CC(R_k) = CC(X_k) = 2$.

C. Bounds for the bisection width of products of XTs and rings

As we did in Section IV, once we have computed the results for the normalized congestion and central cut of the different factor graphs, we can calculate the lower and upper bound of the bisection width of products of XTs and rings. We will start by the lower bound of the bisection width presenting the following lemma.

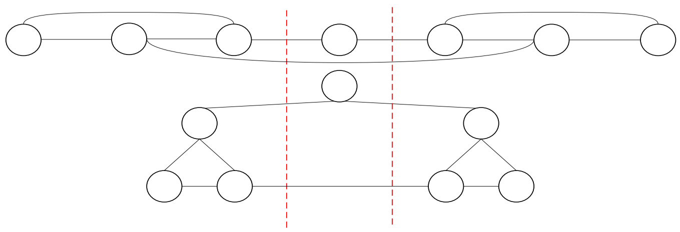


Figure 4. Central cut on a extended complete binary tree

Lemma 7: The bisection width of a d-dimensional mesh-connected XTs and rings, $MCXR^{(d)}$, is lower bounded by $2\Psi(\alpha)$.

Proof: As the d-dimensional mesh-connected XTs and rings is a Cartesian product of d rings and XTs, from Corollary 1, we can state that

$$\frac{r}{4}\Psi(\alpha) \leq \sum_{i=1}^{d} \beta_r(G_{k_i})|\partial_i^{MCXR} S| \tag{23}$$

when $|S| = \lfloor \frac{n}{2} \rfloor$ and where $r = 2$ and G_{k_i} is either a ring or an XT.

As we already know that the normalized congestion of both factor graphs is $\beta_2(R_k) = \beta_2(X_k) = \frac{1}{4}$, we have that

$$BW(MCXR^{(d)}) \geq 2\Psi(\alpha). \tag{24}$$

■

We calculate now the upper bound for the bisection width of a d-dimensional mesh-connected rings and XTs.

Lemma 8: The bisection width of a d-dimensional, $MCXR^{(d)}$, is upper bounded by $2\Psi(\alpha)$.

Proof: The d-dimensional mesh-connected XTs and rings graph can also be embedded into a d-dimensional array, so then, we can use Theorem 4. As happened with the congestion, the value of the central cut of both XTs and rings is the same, concretely, $CC(R_k) = CC(X_k) = 2$, independently of their sizes or number of levels. Hence, $\max_i\{CC(G_{k_i})\} = 2$ (where G_{k_i} is either a ring or an XT). Then,

$$BW(MCXR^{(d)}) \leq 2\Psi(\alpha). \tag{25}$$

■

From Lemma 7 and Lemma 8, Theorem 6 follows.

Theorem 6: The bisection width of a d-dimensional mesh-connected XTs and rings $MCXR^{(d)}_{k_1,k_2,...,k_d}$ is $2\Psi(\alpha)$.

From the bisection width of the d-dimensional mesh-connected XTs and rings, we can derive the following corollaries for the particular cases where all the factor graphs are rings, Torus $T^{(d)}$, or XTs, mesh-connected extended trees $MCX^{(d)}$.

Corollary 7: The bisection width of the d-dimensional torus $T^{(d)}_{k_1,k_2,...,k_d}$ is $BW(T^{(d)}) = 2\Psi(\alpha)$.

Corollary 8: The bisection width of the d-dimensional mesh-connected extended trees $MCX^{(d)}_{k_1,k_2,...,k_d}$ is $BW(MCX^{(d)}) = 2\Psi(d)$.

VI. BCube

We devote this section to obtain bounds for the bisection width of a d-dimensional BCube[2]. BCube is different from the topologies considered in the previous sections because it is obtained as the combination of basic networks formed by a collection of k nodes (servers) connected by a switch. These factor networks are combined into multidimensional networks in the same way product graphs are obtained from their factor graphs. This allows us to study the BCube as an special instance of a product network. The d-dimensional BCube can be obtained as the d dimensional product of one-dimensional BCube networks, each one of k nodes.

A. Factor and product graphs

We first define a BCube and how a d-dimensional BCube network is built from it.

Definition 16: A *BCube* of k nodes, denoted BC_k, is composed by k nodes connected to a k-ports switch. It can be seen as a complete graph K_k where all the edges have been replaced by a switch.

Combining this graph d times as a factor graph in the Cartesian product, we obtain a d-dimensional BCube.

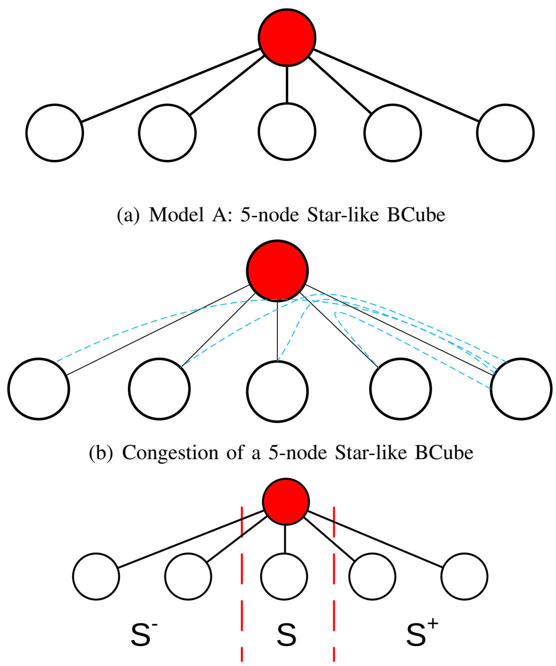
Definition 17: A *d-dimensional BCube*, denoted by $BC_k^{(d)}$, is the Cartesian product of d BC_k (the switches are not considered nodes for the Cartesian product). I.e., $BC_k^{(d)} = BC_k \times BC_k \times \cdots \times BC_k$.

$BC_k^{(d)}$ can also be seen as a d-dimensional homogeneous array where all the edges in each path have been removed and replaced by a switch where two nodes $(u_1, ..., u_i, ..., u_d)$ and $(v_1, ..., v_i, ..., v_d)$ are connected to the same switch if and only if $(u_i \neq v_i)$ and $u_j = v_j$ for all $j \neq i$.

Although the logical interconnection model of a BC_k is a complete graph, we can not use it to calculate the bisection bandwidth of the network as we need to know the number of physical links that we will remove, not the logical ones. We present two possible models for BC_k.

The first one, Model-A or star-like model, denoted by $BC(A)_k$, consists in k nodes connected one-to-one to a virtual node which represents the switch. The second one, Model-B or hyperlink model, denoted by $BC(B)_k$, consists in k nodes connected by a hyperlink.

The main reason for obtaining the bisection width of a d-dimensional BCube is to be able to bound its bisection bandwidth. However, as the d-dimensional BCube is not a regular graph, the bisection width can have different forms depending on where the communication bottleneck occurs in a BCube graph. While the two presented models are logically equivalent to a complete graph, they have a different behavior from the traffic point of view.



(a) Model A: 5-node Star-like BCube

(b) Congestion of a 5-node Star-like BCube

(c) Central cut of a 5-node Star-like BCube

Figure 5. Model A of a 5-node BCube and its congestion and central cut

Let us consider that we have a BC_3 where the links have a speed of 100 Mbps while the switch can switch at 1 Gbps. Under these conditions, the links become the bottleneck of the network and, even when the switches would be able to provide a bisection bandwidth of 1 Gbps, the effective bisection bandwidth is only of 200 Mbps in both directions.

Consider the opposite situation now, where the BCube switch only supports 500 Mbps of internal traffic while the links transmit at 1 Gbps. In this case, the switches are the bottleneck of the network and the bisection bandwidth is only 500 Mbps, although the links would be able to support up to 1 Gbps.

The first example illustrates an scenario where we would bisect the network by removing the links that connect the servers to the switches, corresponding it to Model A. On the other hand, what we find in example 2 is a perfect scenario for Model B, where we would do better by removing entire switches when bisecting the network. Concretely, being s the switching capacity of a switch, and T the traffic supported by a link, we will choose Model A when $s \geq \lfloor \frac{k}{2} \rfloor \cdot 2T$ and model $s \leq 2T$

B. Congestion and central cut of BCube

We will compute now the congestion and central cut of both models in order to be able to calculate the respective lower and upper bounds. We start by the congestion and central cut of Model-A.

Model-A is also called star-like model. The name of star-like comes from the fact that the factor graph can be seen as a star with the switch in the center. If we set $r = 1$, the congestion of every link of the star is easily found to be $m_r(BC(A)_k) = k - 1$[3] as shown in Figure 5(b).

Corollary 9: The normalized congestion of $BC(A)_k$ is

$$\beta_r(BC(A)_k) = \begin{cases} \frac{k-1}{k^2} & \text{if } k \text{ is even} \\ \frac{1}{k+1} & \text{if } k \text{ is odd.} \end{cases}$$

The central cut, which is also trivial and can be found in Figure 5(c), will depend on whether the number of nodes k is even or odd,

$$CC(BC(A)_k) = \begin{cases} \frac{k}{2} & \text{if } k \text{ is even} \\ \frac{k-1}{2} & \text{if } k \text{ is odd.} \end{cases}$$

Having computed the congestion and the central cut for Model-A, we will compute them now for Model-B. We also call Model-B hyperlink model[4] due to the fact that all the servers from the BCube are connected by a hyperlink so no switch is needed.

Calculating the congestion of a Model-B BCube will be easy then. If we set $r = 1$ there will be only one edge to be removed, the congestion of the graph will be total amount of edges of its equivalent K_k, i. e., $m_r(BC(B)_k) = \frac{k(k-1)}{2}$.

Corollary 10: The normalized congestion of $BC(B)_k$ is

$$\beta_r(BC(B)_k) = \begin{cases} \frac{k-1}{k} & \text{if } k \text{ is even} \\ \frac{k}{k+1} & \text{if } k \text{ is odd.} \end{cases}$$

As for Model-A, the central cut is easily computed. As there is only one hyperlink, its central cut will be $CC(BC(B)_k) = 1$.

C. Bounds for the bisection width of BCube

Having computed the congestion and central cut of both models, we can calculate the lower and upper bounds of the bisection width of each one of them.

We will start by the lower and upper bound of the bisection width of model A and, then, we will calculate both bounds for model B.

We first present the following lemma for the lower bound of the bisection width of a Model-A BCube.

Lemma 9: The bisection width of a Model-A d-dimensional Bcube, $BC(A)_k^{(d)}$, is lower bounded by $\frac{k^{d+1}}{4(k-1)}$ if k is even, and by $\frac{k+1}{4}\frac{k^d-1}{k-1}$ if k is odd.

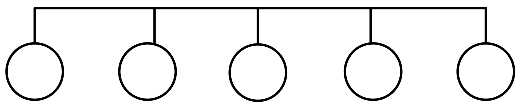
Proof: Using the value of the congestion of a Model-A BCube in Corollary 1, it follows that

$$BW(BC(A)_k^{(d)}) \geq \begin{cases} \frac{1}{4}\frac{k^2}{k-1}\Psi(\alpha) = \frac{k^{d+1}}{4(k-1)} & \text{if } k \text{ is even} \\ \frac{1}{4}\frac{k^2-1}{k-1}\Psi(\alpha) = \frac{k+1}{4}\frac{k^d-1}{k-1} & \text{if } k \text{ is odd} \end{cases}$$

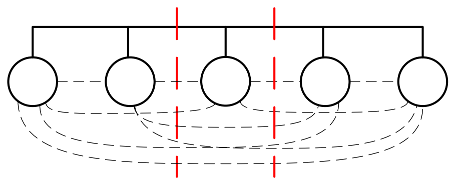
■

[3]Note that in the computation of the congestion, the switch is not considered a node of the graph.

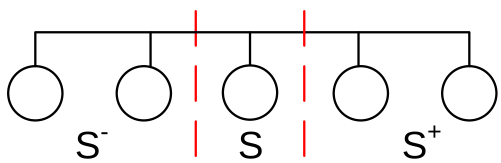
[4]This model is quite similar to the one proposed by Pan in [18].



(a) Model B: 5-node hyperlink BCube



(b) Congestion of a 5-node hyperlink BCube



(c) Central cut of a 5-node hyperlink BCube

Figure 6. Model B of a 5-node BCube and its congestion and central cut

After proving the lower bound for the bisection width of a Model-A d-dimensional BCube, we follow with the upper bound.

Lemma 10: The bisection width of a Model-A d-dimensional Bcube, $BC(A)_k^{(d)}$, is upper bounded by $\frac{k^d}{2}$ if k is even, and by $\frac{k^d-1}{2}$ if k is odd.

Proof: The Cartesian product of Model-A star-like factor graphs can be embedded into a d-dimensional array, so Theorem 4 will be extremely useful again. If we apply the values of the central cut of Model-A in Theorem 4, is immediate to compute the following upper bound

$$BW(BC(A)_k^{(d)}) \leq \begin{cases} \frac{k^d}{2} & \text{if } k \text{ is even} \\ \frac{k^d-1}{2} & \text{if } k \text{ is odd.} \end{cases}$$

■

Now, from the combination of Lemma 9 and Lemma 10 we can state Theorem 7:

Theorem 7: The value of the bisection width of a Model-A d-dimensional Bcube, $BC(A)_k^{(d)}$, is in the interval $[\frac{k^{d+1}}{4(k-1)}, \frac{k^d}{2}]$ if k is even, and in the interval $[\frac{k+1}{4}\frac{k^d-1}{k-1}, \frac{k^d-1}{2}]$ if k is odd.

Let us calculate now the bounds of a Model-B d-dimensional BCube. As we did with Model A, we will first prove the lower bound and then the upper one. For the lower bound we present the following lemma.

Lemma 11: The bisection width of a Model-B d-dimensional Bcube, $BC(B)_k^{(d)}$, is lower bounded by $\frac{k^d}{2(k-1)}$ if k is even, and by $\frac{k+1}{2k}\frac{k^d-1}{k-1}$ if k is odd.

Proof: Like in the case of Model A, we use the value of the congestion of Model B in Corollary 1. Since all the dimensions have the same size k, it follows that

$$BW(BC(B)_k^{(d)}) \geq \begin{cases} \frac{k^d}{2(k-1)} & \text{if } k \text{ is even} \\ \frac{k+1}{2k}\frac{k^d-1}{k-1} & \text{if } k \text{ is odd} \end{cases}$$

■

We present now Lemma 12 for the upper bound of the bisection width of a Model-B d-dimensional Bcube.

Lemma 12: The bisection width of a Model-B d-dimensional Bcube, $BC(B)_k^{(d)}$, is upper bounded by $\frac{k^d-1}{k-1}$.

Proof: As for model A, the d-dimensional BCube resulting from the Cartesian product of Model-B graphs can be embedded into a d-dimensional array. Thanks to this fact, we can use the computed value of its central cut in Theorem 4 to obtain the upper bound of the bisection width,

$$BW(BC(B)_k^{(d)}) \leq 1 \cdot \Psi(\alpha) = \frac{k^d-1}{k-1}.$$

■

Combining both Lemmas, 9 and 10, we can state Theorem 7.

Theorem 8: The value of the bisection width of a Model-B d-dimensional Bcube, $BC(B)_k^{(d)}$, is in the interval $[\frac{k^d}{2(k-1)}, \frac{1-k^d}{1-k}]$ if k is even, and in the interval $[\frac{k+1}{2k}\frac{k^d-1}{k-1}, \frac{k^d-1}{k-1}]$ if k is odd.

VII. Conclusions

Exact results for the bisection bandwidth of various d-dimensional classical parallel topologies have been provided in this paper. These results consider any number of dimensions and any size, odd or even, for the factor graphs. These multidimensional graphs are based on factor graphs such as paths, rings, complete binary trees or extended complete binary trees. Upper and lower bounds for the bisection width of a d-dimensional BCube are also provided.

Some of the products used the same factor graph multiple times, like the d-dimensional torus, mesh-connected trees or mesh-connected extended trees and some other combined different factor graphs, like the mesh connected trees and paths or mesh-connected extended trees and rings.

An interesting open problem is how to obtain the exact value of the bisection width of graph obtained by combining paths and rings (cylinders) and other combinations not considered in this paper. Similarly, obtaining an exact result for the bisection bandwidth of the d-dimensional BCube remains as an open problem.

References

[1] F. T. Leighton, *Introduction to parallel algorithms and architectures: array, trees, hypercubes*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1992.

[2] C. Guo, G. Lu, D. Li, H. Wu, X. Zhang, Y. Shi, C. Tian, Y. Zhang, and S. Lu, "Bcube: a high performance, server-centric network architecture for modular data centers," in *SIGCOMM*, P. Rodriguez, E. W. Biersack, K. Papagiannaki, and L. Rizzo, Eds. ACM, 2009, pp. 63–74.

[3] C. Guo, H. Wu, K. Tan, L. Shi, Y. Zhang, and S. Lu, "Dcell: a scalable and fault-tolerant network structure for data centers," in *SIGCOMM*, V. Bahl, D. Wetherall, S. Savage, and I. Stoica, Eds. ACM, 2008, pp. 75–86.

[4] D. N. Jayasimha, B. Zafar, and Y. Hoskote, "On chip interconnection networks why they are different and how to compare them," *Intel*, 2006.

[5] M. Mirza-Aghatabar, S. Koohi, S. Hessabi, and M. Pedram, "An empirical investigation of mesh and torus noc topologies under different routing algorithms and traffic models," in *Proceedings of the 10th Euromicro Conference on Digital System Design Architectures, Methods and Tools*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 19–26. [Online]. Available: http://dl.acm.org/citation.cfm?id=1302494.1302781

[6] D. Zydek and H. Selvaraj, "Fast and efficient processor allocation algorithm for torus-based chip multiprocessors," *Comput. Electr. Eng.*, vol. 37, pp. 91–105, January 2011. [Online]. Available: http://dx.doi.org/10.1016/j.compeleceng.2010.10.001

[7] E. Salminen, A. Kulmala, and T. D. H, "Survey of network-on-chip proposals," *Simulation*, no. March, pp. 1–13, 2008.

[8] K. Efe and A. Fernández, "Mesh-connected trees: A bridge between grids and meshes of trees," *IEEE Trans. Parallel Distrib. Syst.*, vol. 7, no. 12, pp. 1281–1291, 1996.

[9] A. Youssef, "Cartesian product networks," in *ICPP (1)*, 1991, pp. 684–685.

[10] ——, "Design and analysis of product networks," in *Proceedings of the Fifth Symposium on the Frontiers of Massively Parallel Computation (Frontiers'95)*. Washington, DC, USA: IEEE Computer Society, 1995, pp. 521–.

[11] J. D. P. Rolim, O. Sýkora, and I. Vrto, "Optimal cutwidths and bisection widths of 2- and 3-dimensional meshes," in *WG*, ser. Lecture Notes in Computer Science, M. Nagl, Ed., vol. 1017. Springer, 1995, pp. 252–264.

[12] K. Efe and A. Fernández, "Products of networks with logarithmic diameter and fixed degree," *IEEE Trans. Parallel Distrib. Syst.*, vol. 6, no. 9, pp. 963–975, 1995.

[13] K. Nakano, "Linear layout of generalized hypercubes," *Int. J. Found. Comput. Sci.*, vol. 14, no. 1, pp. 137–156, 2003.

[14] K. Efe and G.-L. Feng, "A proof for bisection width of grids," *World Academy of Science, Engineering and Technology*, vol. 27, no. 31, pp. 172 – 177, 2007.

[15] M. C. Azizoğlu and Ö. Eğecioğlu, "The isoperimetric number and the bisection width of generalized cylinders," *Electronic Notes in Discrete Mathematics*, vol. 11, pp. 53–62, 2002.

[16] ——, "The bisection width and the isoperimetric number of arrays," *Discrete Applied Mathematics*, vol. 138, no. 1-2, pp. 3–12, 2004.

[17] ——, "Extremal sets minimizing dimension-normalized boundary in hamming graphs," *SIAM J. Discrete Math.*, vol. 17, no. 2, pp. 219–236, 2003.

[18] Y. Pan, S. Q. Zheng, K. Li, and H. Shen, "An improved generalization of mesh-connected computers with multiple buses," *IEEE Trans. Parallel Distrib. Syst.*, vol. 12, pp. 293–305, March 2001. [Online]. Available: http://dx.doi.org/10.1109/71.914773