Bisection (Band)Width of Product Networks with Application to Data Centers

Jordi Arjona Aroca and Antonio Fernández Anta, *Senior Member*, *IEEE*

Abstract—The bisection width of interconnection networks has always been important in parallel computing, since it bounds the speed at which information can be moved from one side of a network to another, i.e., the bisection bandwidth. Finding its exact value 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. We provide two general results that allow us to obtain upper and lower bounds on the bisection width of any product graph as a function of some properties of its factor graphs. The power of these results is shown by deriving 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. We also apply these results to data centers, by obtaining bounds for the bisection bandwidth of the d-dimensional BCube network, a recently proposed topology for data centers.

Index Terms—Bisection bandwidth, bisection width, torus, BCube, product graphs, complete binary trees, extended trees, mesh-connected trees

1 Introduction

The bisection width and the bisection bandwidth of interconnection networks have always been two important parameters of a network. The first one reflects the smallest number of links which have to be removed to split the network into two equal parts, while the second one bounds the speed at which data can be moved between these parts. In general, both values are derivable from one another, which is the reason why most previous work has been devoted to only one of them (in particular, the bisection width).

The bisection width has been a typical goodness parameter to evaluate and compare interconnection networks for parallel architectures [1], [2], [3]. This interest has been transferred to the network-on-chip topologies, as the natural successors of the parallel architectures of the 1990s [4], [5], [6], [7]. The bisection (band)width is also nowadays being used as a reference parameter on the analysis of the latest topologies that are being deployed in data centers. The bisection bandwidth can be used to compare the potential throughput between any two halves of the network in different topologies. Similarly, the bisection width also gives some insights on their fault tolerance, showing the maximum number of critical link errors a network can suffer before being split in two halves. This can be seen in recent papers which propose new topologies, like BCube [8] or DCell [9]. The bisection (band)width is one of the parameters used to compare these new topologies with classical topologies, like grids, tori, and hypercubes, or with other data center topologies, like trees and fat trees.

Finding the exact value of the bisection width is hard in general. Computing it has proven to be challenging even for very simple families of graphs. 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 20 years. One general family of interconnection networks, of which the torus is a subfamily, is the family of product networks. The topology of these networks is obtained by combining factor graphs with the Cartesian product operator. This technique allows us to build large networks from the smaller factor networks. Many popular interconnection networks are instances of product networks, like the grid and the hypercube. In this paper, we derive techniques to bound the bisection width of product networks and apply these techniques to obtain the bisection width of some product network families.

1.1 Related Work

To our knowledge, Youssef [10], [11] was among the first to explore the properties of product networks as a family. He presented the idea of working with product networks as a divide-and-conquer problem, obtaining important properties of a product network in terms of the properties of its factor graphs.

The bisection width of arrays and tori was explored by Dally [12] and Leighton [1] in the early 1990s, presenting exact results for these networks when the number of nodes per dimension was even. The case when there are odd number of nodes per dimension was left open. Rolim et al. [13] gave the exact values for the bisection width of two- and three-dimensional grids and tori, but left open the question for higher number of dimensions.

- *J. Arjona Aroca is with the Institute IMDEA Networks and Universidad Carlos III de Madrid, Madrid, Spain. E-mail: jorge.arjona@imdea.org.*
- *A. Fernández Anta is with the Institute IMDEA Networks, Madrid, Spain. E-mail: antonio.fernandez@imdea.org.*

Manuscript received 18 June 2012; revised 20 Feb. 2013; accepted 20 Mar. 2013; published online 28 Mar. 2013.
Recommended for acceptance by D. Wang.
For information on obtaining reprints of this article, please send e-mail to: tpds@computer.org, and reference IEEECS Log Number TPDS-2012-06-0577.
Digital Object Identifier no. 10.1109/TPDS.2013.95.

TABLE 1

Notation	
$(B)BW(G)$	Bisection (Band)Witdh of graph G
n	Number of nodes in graph G
k_i	Number of nodes in dimension i
d	Dimension index
$G^{(d)}_{k_1,\ldots,k_d}$	Graph G with d dimensions of sizes $k_1, k_2, \ldots, k_d$
α	Lowest index of an even k_i
$\Psi(\alpha)$	Bisection Width of a d-dimensional array
$\partial^g S(G)$	Edges connecting $S(G)$ and $V(G) \setminus S(G)$
$B_g(S)$	Dimension normalized boundary of $S(G)$
$CC(G)$	Central Cut of graph G
$\beta_r(G)$	Normalized congestion of graph G of multiplicity r
$m_r(G)$	Congestion of graph G with multiplicity r
$\mathcal{E}$	Set of all possible embeddings M_r of rK_n onto G
T	Links capacity
s	Switching capacity

For the special case in which all the factors are isomorphic, Efe and Fernández [14] provided a lower bound on the bisection width of a product graph as a function of a new parameter of a factor network they defined, the maximal congestion. Nakano [15] presented the exact value of the bisection width for the Cartesian product of isomorphic paths and cliques (i.e., square grids and Hamming graphs). If the factor graphs have k nodes, he proved that the d-dimensional square grid has bisection width k^{d-1} when k is even, and $\frac{(k^d-1)}{(k-1)}$ when k is odd. Similarly, the square Hamming graph has bisection width k^{d+1} when k is even, and $(k+1)\frac{(k^d-1)}{4}$ when k is odd. The exact bisection width of the d-dimensional square grid was found independently by Efe and Feng [16]. For this and all the basic notation used throughout the paper, we refer the reader to Table 1.

For the present paper, the work of Azizoğlu and Eğecioğlu is very relevant. In [17] and [18], they studied the relationship 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 cylinders (products of paths and rings) with even number of nodes in its largest dimension. In the latter reference, they found the exact bisection width of the grid $A^{(d)}_{k_1,k_2,\ldots,k_d}$, with k_i nodes along dimension i, and where $k_1 \geq k_2 \geq \cdots \geq k_d$. The value of this bisection width is $BW(A^{(d)}_{k_1,k_2,\ldots,k_d}) = \sum_{i=1}^{\alpha} C_i$, where α is the smallest index for which k_i is even ($\alpha = d$ if no index is even), and $C_i = \prod_{j=i+1}^{d} k_j$. Since the value of the bisection width of the array will appear frequently, we will use the notation $\Psi(\alpha) = \sum_{i=1}^{\alpha} C_i = \sum_{i=1}^{\alpha} \prod_{j=i+1}^{d} k_j$ throughout the rest of the paper.

1.2 Contributions

The main contributions of this paper are two theorems that allow us to derive lower and upper bounds on the bisection width of any product network as a function of some simple parameters of its factor graphs. We prove the power of these theorems by obtaining the exact value of the bisection width for several families of product networks. The families presented are of some interest as well, because they have been proposed as interconnection networks for parallel architectures, but their exact bisection width has never been derived.

One of the most interesting derived result of this paper is the exact value of the bisection width of the torus of any dimension, since, as mentioned before, 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 $T^{(d)}_{k_1,k_2,\ldots,k_d}$, that has k_i nodes along dimension i, and where $k_1 \geq k_2 \geq \cdots \geq k_d$, is exactly twice the bisection width of the grid of similar dimensions $A^{(d)}_{k_1,k_2,\ldots,k_d}$, i.e., $BW(T^{(d)}_{k_1,k_2,\ldots,k_d}) = 2\Psi(\alpha)$, where α is the smallest index for which k_i is even ($\alpha = d$ if no index is even). In addition to the result for the torus, we provide the exact value for the bisection width of products of complete binary trees (CBT) of any size (mesh-connected trees [19]), products of extended CBT (which are CBT with the leaves connected with a path [19]), products of CBT and paths, and products of extended CBT and rings. To obtain the bisection *bandwidth* of these networks, we assume that every edge removed by the bisection width is in fact a duplex link with bandwidth of T in each direction. This directly implies that for any of these networks G, the bisection bandwidth is computed as $BBW(G) = 2T \cdot BW(G)$.

The general upper and lower bound results are also used to derive bounds on the bisection bandwidth of a topology proposed for data centers, the BCube. A BCube is the Cartesian product of factors networks formed by k nodes connected via a k-port switch (where the switch is not considered to be a node). An essential difference of this topology from the previous one is that edges do not connect nodes directly, and the direct relation between bisection width and bisection bandwidth does not hold anymore. In networks with switches like this one, the switching capacity of the switch comes into play as well. Since the bisection bandwidth is the parameter of interest in data centers, we derive bounds on its value for two cases: when the bottleneck for the bisection bandwidth is at the links (Model A), and when it is at the switches (Model B).

Table 2 summarizes the results derived for the bisection bandwidth obtained for the different parallel topologies and for BCube. As can be seen, for the former, the values obtained are exact, while for the latter the upper and lower bounds do not match exactly. However, they differ by less than a factor of two.

The rest of the paper is organized as follows: Section 2 presents some basic definitions used in the rest of sections. In Section 3, we provide the general results to derive bounds on the bisection bandwidth of product networks. Sections 4 and 5 present our results for the bisection bandwidth of some classical parallel topologies. Bounds on the bisection bandwidth of the BCube network are presented in Section 6. Finally, in Section 7, we present our conclusions and some open problems.

2 Definitions

2.1 Graphs and Bisections

In this section, we present definitions and notation that will be used along the text. Given a graph[1] G, we denote its sets of vertices and edges as $V(G)$ and $E(G)$, respectively.

1. Unless otherwise stated, we will use the terms graph and network indistinctly.

TABLE 2
Bisection Bandwidth of Different Product Networks ($\Psi(\alpha) = \sum_{i=1}^{\alpha} C_i = \sum_{i=1}^{\alpha} \prod_{j=i+1}^{d} k_j$)

Product graph	Factor graphs		$\beta(G)$	$CC(G)$	Bisection bandwidth
Torus	Ring		$1/8$	2	$4T \cdot \Psi(\alpha)$
Product of extended CBT	XTs		$1/8$	2	$4T \cdot \Psi(\alpha)$
Product of extended CBT & rings	Rings & XTs		$1/8$	2	$4T \cdot \Psi(\alpha)$
Mesh connected trees	CBT		$1/4$	1	$2T \cdot \Psi(\alpha)$
Product of CBT and paths	Paths & CBTs		$1/4$	1	$2T \cdot \Psi(\alpha)$
BCube	Model A	even	$\frac{k-1}{k^2}$	$\frac{k}{2}$	$2T\frac{k^{d+1}}{4(k-1)} \leq BBW(BCA_k^{(d)}) \leq 2T\frac{k^d}{2}$
		odd	$\frac{1}{k+1}$	$\frac{k-1}{2}$	$2T\frac{k+1}{4}\frac{k^d-1}{k-1} \leq BBW(BCA_k^{(d)}) \leq 2T\frac{k^d-1}{2}$
	Model B	even	$\frac{k-1}{2k}$	1	$s\frac{k^d}{2(k-1)} \leq BBW(BCB_k^{(d)}) \leq s\frac{k^d-1}{k-1}$
		odd	$\frac{k}{2(k+1)}$	1	$s\frac{k+1}{2k}\frac{k^d-1}{k-1} \leq BBW(BCB_k^{(d)}) \leq s\frac{k^d-1}{k-1}$

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 $V(G) \setminus S(G)$. 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 this 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 that 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 G, denoted $BBW(G)$, is the minimal amount of data per time unit that can be transferred between any two halves of the network when its links are transmitting at full speed.*

As mentioned above, unless otherwise stated, we assume that all the links in a network G are duplex and have the same capacity T in each direction. Then, we can generally assume that the relation between the bisection bandwidth and the bisection width is $BBW(G) = 2T \cdot BW(G)$.

2.2 Factor and Product Graphs

We first define the Cartesian product of graphs.

Definition 3. *The d-dimensional* Cartesian product *of graphs $G_1, G_2, \ldots, G_d$, denoted by $G_1 \times G_2 \times \cdots \times G_d$, is the graph with vertex set $V(G_1) \times V(G_2) \times \cdots \times V(G_d)$, in which vertices $(u_1, \ldots, u_i, \ldots, u_d)$ and $(v_1, \ldots, v_i, \ldots, v_d)$ 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 $G_1, G_2, \ldots, G_d$ 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 now define some of the basic factor graphs that will be considered.

Definition 4. *The* path *of k vertices, denoted by 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 by 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 by 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 $i, j \in V(rK_k)$ is connected with r parallel edges.*

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 $G_{k_1,\ldots,k_d}^{(d)}$, where G is the name of the graph, the superscript (d) means that it is a d-dimensional graph, and $k_1, \ldots, k_d$ are the number of vertices in each dimension. (Superscript and subscripts may be omitted when clear from the context.) It will always hold that $k_1 \geq k_2 \geq \cdots \geq k_d$, i.e., the factor graphs are sorted by decreasing number of vertices. We will often use n to denote the number of nodes in the graph $G_{k_1,\ldots,k_d}^{(d)}$, i.e., $n = k_1 k_2 \ldots k_d$, 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 by $A_{k_1,\ldots,k_d}^{(d)}$, is the Cartesian product of d paths of $k_1, \ldots, k_d$ vertices, respectively, i.e., $A_{k_1,\ldots,k_d}^{(d)} = P_{k_1} \times P_{k_2} \times \cdots \times P_{k_d}$.*

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

Observe that the *Hamming graph* [20] is the particular case of the r-Hamming graph, with $r = 1$. For brevity, we use $H_{k_1,\ldots,k_d}^{(d)}$, instead of $1H_{k_1,\ldots,k_d}^{(d)}$, to denote the Hamming graph.

2.3 Boundaries and Partitions

We define now the dimension-normalized boundary [18].

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

$$B_G(S) = \frac{|\partial_1^G S|}{\sigma_1} + \frac{|\partial_2^G S|}{\sigma_2} + \cdots + \frac{|\partial_d^G S|}{\sigma_d}, \quad (1)$$

where, for each $i \in [1, d]$, ∂_i^G is the subset of the edges in the boundary ∂^G that belong to dimension i, and σ_i is defined as

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

Observation 1. For $rH^{(d)}_{k_1,\dots,k_d}$, any subset S of nodes, and any dimension i, it holds that $|\partial_i^{rH} S| = r \cdot |\partial_i^H S|$. Hence,

$$\begin{aligned} B_{rH}(S) &= \frac{|\partial_1^{rH} S|}{\sigma_1} + \cdots + \frac{|\partial_d^{rH} S|}{\sigma_d} \\ &= r\left(\frac{|\partial_1^H S|}{\sigma_1} + \cdots + \frac{|\partial_d^H S|}{\sigma_d}\right) \\ &= rB_H(S). \end{aligned}$$

Let us define the lexicographic order. Consider graph $H^{(d)}_{k_1,\dots,k_d}$, we say that vertex $x = (x_1, x_2, \dots, x_d)$ precedes vertex $y = (y_1, y_2, \dots, y_d)$ in *lexicographic order* if there exists an index $i \in [1, d]$ such that $x_i < y_i$ and $x_j = y_j$ for all $j < i$. Azizoğlu and Eğecioğlu [20] proved the following result.

Theorem 1 [20]. *Consider a d-dimensional Hamming graph $H^{(d)}_{k_1,\dots,k_d}$, 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[2]; then, $B_H(\bar{S}) \leq B_H(S)$.*

3 Bounds on the BW of Product Graphs

In this section, we present general bounds on the bisection width of product graphs as well as 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.

3.1 Lower Bound

We start by defining the normalized congestion of a graph. Let G be a graph with n nodes. Then, an *embedding* of the 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 by $m_r(G)$, 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 an embedding M_r of rK_n onto G as $c_{M_r}(e) = |\{e' \in E(rK_n) : e \in M_r(e')\}|$. (Observe that $M_r(e') \subseteq E(G)$ is a path in G.) Then, the congestion $m_r(G)$ is

$$m_r(G) = \min_{M_r \in \mathcal{E}} \max_{e \in E(G)} \{c_{M_r}(e)\}, \quad (3)$$

where $\mathcal{E}$ is the set of all possible embeddings of rK_n onto G. Then, using (3) and (2), we define the *normalized congestion* with multiplicity r of G as $\beta_r(G) = m_r(G)/\sigma_n$. Having defined the normalized congestion, we proceed to extend Theorem 1 to r-Hamming graphs.

Theorem 2. *Consider 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 $X \neq \bar{S}$ such that $|X| = |\bar{S}|$ and $B_{rH}(\bar{S}) > B_{rH}(X)$. Then, applying Observation 1 to both X and $\bar{S}$, we obtain that $B_H(\bar{S}) = B_{rH}(\bar{S})/r > B_{rH}(X)/r = B_H(X)$, 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 $rH^{(d)}_{k_1,k_2,\dots,k_d}$, such that $\bar{S}$ are the first $\lfloor \frac{n}{2} \rfloor$ vertices of rH in lexicographic order, and n is the number of vertices of rH. Then, the dimension-normalized boundary of $\bar{S}$ is $B_{rH}(\bar{S}) = r\Psi(\alpha)/4$.*

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 [18], that $\partial_i^H \bar{S} = \emptyset$ for all $i > \alpha$.[3] The number of edges in each dimension $i \in [1, \alpha]$ on the boundary of $\bar{S}$ in H is

$$|\partial_i^H \bar{S}| = \begin{cases} \frac{k_i}{2}\left(\prod_{j=i+1}^{d} k_j\right)\frac{k_i}{2}, & \text{if } k_i \text{ is even}, \\ \frac{k_i - 1}{2}\left(\prod_{j=i+1}^{d} k_j\right)\frac{k_i + 1}{2}, & \text{if } k_i \text{ is odd}. \end{cases} \quad (4)$$

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

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

Finally, from Observation 1, we derive $B_{rH}(\bar{S}) = rB_H(\bar{S}) = r\Psi(\alpha)/4$. □

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

Theorem 3. *Let $G = G_1 \times \cdots \times G_d$, where $|V(G_i)| = k_i$ and $k_1 \geq k_2 \geq \cdots \geq k_d$. Let $\beta_r(G_i)$ be the normalized congestion with multiplicity r of G_i (for any r), for all $i \in [1, d]$. 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 $S_i \subset V(G_i)$, $|\partial^{rK_{k_i}} S_i| \leq m_r(G_i) \cdot |\partial^{G_i} S_i|$. 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

2. Observe that we have reversed the ordering of dimensions with respect to the original theorem of [20].

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

$$\begin{aligned} B_{rH}(\bar{S}) &\leq B_{rH}(S) \\ &\leq m_r(G_1)\frac{|\partial_1^G S|}{\sigma_1} + \cdots + m_r(G_d)\frac{|\partial_d^G S|}{\sigma_d} \\ &= \beta_r(G_1)|\partial_1^G S| + \cdots + \beta_r(G_i)|\partial_d^G S|. \end{aligned}$$

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

Corollary 1. *Let $G = G_1 \times \cdots \times G_d$, where $|V(G_i)| = k_i$ and $k_1 \geq k_2 \geq \cdots \geq k_d$. Let $\beta_r(G_i)$ be the normalized congestion with multiplicity r 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, $r\Psi(\alpha)/4 \leq \sum_{i=1}^{d} \beta_r(G_i)|\partial_i^G S|$.*

Corollary 2. *Let $G = G_1 \times \cdots \times G_d$, where $|V(G_i)| = k_i$ and $k_1 \geq k_2 \geq \cdots \geq k_d$. Let $\beta_r(G_i) = \beta$ be the normalized congestion with multiplicity r 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, $r\Psi(\alpha)/4\beta \leq BW(G)$.*

3.2 Upper Bound

Having proved the lower bound on 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 by $CC(G)$, 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 $G = G_1 \times \cdots \times G_d$, where $|V(G_i)| = k_i$ and $k_1 \geq k_2 \geq \cdots \geq k_d$. Then, $BW(G) \leq \max_i\{CC(G_i)\} \cdot \Psi(\alpha)$.*

Proof. It was shown in [18] 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 path 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 $|\partial_i^G S|/CC(G_i) \leq |\partial_i^{A^{(d)}} S|$, which also holds if the paths in dimension i are not cut. Applying this to all dimensions, we obtain

$$\frac{|\partial_1^G S|}{CC(G_1)} + \cdots + \frac{|\partial_d^G S|}{CC(G_d)} \leq BW(A^{(d)}) = \Psi(\alpha). \quad (5)$$

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.

4 BW 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 complete binary trees. 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.

4.1 Factor and Product Graphs

In this section, we will work with paths, which were defined in Section 2, and CBTs, which are defined now.

Definition 10. *The* complete binary tree *of k vertices, denoted by 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 by $MCTP^{(d)}_{k_1,k_2,\ldots,k_d}$, 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* [19], denoted by $MCT^{(d)}_{k_1,k_2,\ldots,k_d}$ 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.)

4.2 Congestion and Central Cut of Paths and CBTs

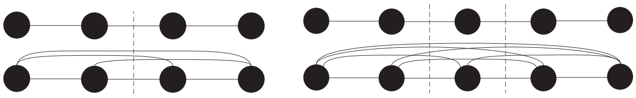
The bisection widths of the aforementioned product graphs can be calculated using the bounds defined in Section 3. To do so, we need to compute first the values of the normalized congestion and central cut of their factor graphs, it is, of a path and of a CBT.

We 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:*

$$m_r(P_k) = \begin{cases} r\dfrac{k^2}{4}, & \text{if } k \text{ is even,} \\ r\dfrac{k^2 - 1}{4}, & \text{if } k \text{ is odd.} \end{cases} \quad (6)$$

Proof. This proof is illustrated in Fig. 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,



(a) The 4-vertex path and clique. (b) The 5-vertex path and clique.

Fig. 1. Paths and possible cuts.

$$m_r(P_k) = \min_{M \in \mathcal{E}} \max_{e \in E(P_k)} \{c_M(e)\} = \max_{e \in P_k}\{c_{M_r}(e)\}. \quad (7)$$

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

$$c_{M_r}(e) = r(i+1)(k-i-1). \quad (8)$$

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, $i = k/2 - 1$ when k is even or $i = (k-1)/2 - 1$ when k is odd. Using these values in (8) leads to the final result

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

□

Corollary 3. *The normalized congestion of a path is* $\beta_r(P_k) = \frac{r}{4}$.

The value of the congestion of a CBT is exactly the same as the congestion of a path with an odd number of nodes. CBT shares with paths the property of having only one possible routing between two nodes. As can be seen in Figs. 1 and 2, 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 *is* $m_r(CBT_k) = r(k^2-1)/4$.

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 $c_{M_r}(e) = r(2^l-1)(k-2^l+1)$. The value of l which maximizes the equation is $l = j - 1$, which is 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 $m_r(CBT_k) = r(k^2-1)/4$. □

Corollary 4. *The normalized congestion of a CBT is* $\beta_r(CBT_k) = r/4$.

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

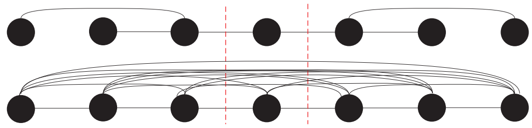


Fig. 2. The seven-vertex complete binary tree and the seven-vertex clique, with their possible cuts.

4.3 Bounds on the BW of Products of CBTs and Paths

Having presented both the congestion and the central cut of the possible factor graphs, we can compute now the lower and upper bound on the bisection width of a product of CBTs and paths. We will start by the lower bound on 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 we can see in Corollaries 3 and 4, the normalized congestion of both factor graphs is the same value $r/4$. Then, we can apply Corollary 2, so $r\Psi(\alpha)(4r/4) \leq BW(MCTP^{(d)})$, which yields, $BW(MCTP^{(d)}) \geq \Psi(\alpha)$.

We follow now by presenting an upper bound on 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, $BW(MCTP^{(d)}) \leq \Psi(\alpha)$. □

From the results obtained from Lemmas 4 and 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 $MCT^{(d)}_{k_1,k_2,\ldots,k_d}$.

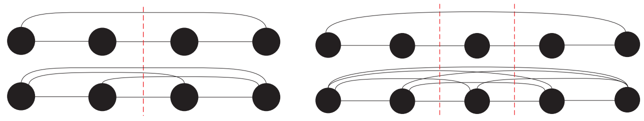
Corollary 5. *The bisection width of the d-dimensional mesh-connected trees* $MCT^{(d)}_{k_1,k_2,\ldots,k_d}$ *is* $BW(MCT^{(d)}) = \Psi(d)$.

5 Products of Rings and Extended Trees

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.

5.1 Factor and Product Graphs

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



(a) The 4-vertex ring and clique. (b) The 5-vertex ring and clique.

Fig. 3. Rings and possible cuts.

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

Definition 13. *The* extended complete binary tree *(a.k.a. XT) of k vertices, denoted by 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 following product graphs:

Definition 14. *A d-dimensional* mesh-connected extended trees and rings, *denoted by $MCXR^{(d)}_{k_1,k_2,\ldots,k_d}$, is the Cartesian product of d graphs of $k_1, k_2, \ldots, k_d$ vertices, respectively, where each factor graph is an 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 by $T^{(d)}_{k_1,k_2,\ldots,k_d}$, 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 4 with $MCT^{(d)}$, we also define the *d-dimensional mesh-connected extended trees*, denoted by $MCX^{(d)}_{k_1,k_2,\ldots,k_d}$, a special case of $MCXR^{(d)}_{k_1,k_2,\ldots,k_d}$ in which all factor graphs are XTs. (The torus is the special case of $MCXR^{(d)}_{k_1,k_2,\ldots,k_d}$ in which all factor graphs are rings.)

5.2 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 3. We present the following lemma for the congestion of a ring.

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

$$m_2(R_k) \le \begin{cases} \dfrac{k^2}{4}, & \text{if } k \text{ is even,} \\ \dfrac{k^2-1}{4}, & \text{if } k \text{ is odd.} \end{cases} \quad (9)$$

Proof. While a path had only one possible routing, for R_k we have two possible routes connecting each pair of nodes. When we embed rK_k, for $r = 2$, into R_k, we embed the parallel edges connecting two nodes through the shortest path, when this is unique. Otherwise, when two nodes are equally distant along each of the two available routes (note that this happens only if k is even), each parallel edge is embedded following a different route. □

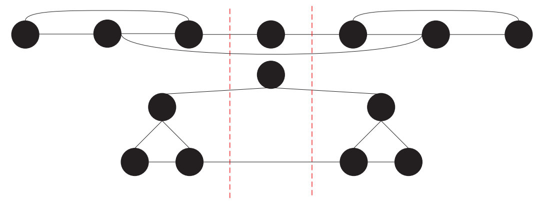


Fig. 4. Central cut on an extended complete binary tree.

A counting argument yields the congestion on any edge under this routing. Let us consider without loss of generality the edge $e = (0, 1)$. Any two nodes that are at distance at most $\lfloor \frac{k-1}{2} \rfloor$ by a shortest path that crosses e, have the two parallel edges connecting them embedded in this path. Then, there are $\lfloor \frac{k-1}{2} \rfloor$ shortest paths that cross e and end at node 1, $\lfloor \frac{k-1}{2} \rfloor - 1$ shortest paths that cross e and end at node 2, and so on. Hence, the congestion in e due to the embedding of these edges is

$$2 \sum_{i=1}^{\lfloor \frac{k-1}{2} \rfloor} \left\lfloor \frac{k-1}{2} \right\rfloor - i + 1 = \left\lfloor \frac{k-1}{2} \right\rfloor \left(\left\lfloor \frac{k-1}{2} \right\rfloor + 1 \right).$$

When k is odd, this is the congestion of the edge e, which becomes

$$\left(\frac{k-1}{2}\right)\left(\left(\frac{k-1}{2}\right) + 1\right) = \frac{k-1}{2}\frac{k+1}{2} = \frac{k^2-1}{4}.$$

When k is even, edge e is also crossed by one of the parallel embedded edges connecting nodes at distance $k/2$. This increases the congestion in edge e by $k/2$. Hence, given that $\lfloor (k-1)/2 \rfloor = (k-2)/2$, the congestion for k even is

$$\left(\frac{k-2}{2}\right)\left(\left(\frac{k-2}{2}\right) + 1\right) + \frac{k}{2} = \frac{k^2}{4}.$$

Corollary 6. *The normalized congestion with multiplicity $r = 2$ of a ring is $\beta_2(R_k) = 1/4$.*

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 [19], 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.

Corollary 7. *The normalized congestion with multiplicity $r = 2$ of an XT is $\beta_2(X_k) = 1/4$.*

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

5.3 Bounds on the BW of Products of XTs and Rings

With the normalized congestion and central cut of the different factor graphs, we can calculate the lower and upper bounds on the bisection width of products of XTs and rings. We will start by the lower bound on the bisection width presenting the following lemma.

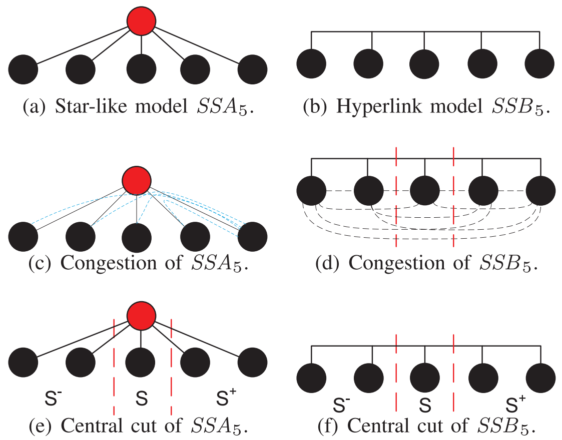


Fig. 5. Proposed models for a five-node switched star, SS_5, their congestion and central cut.

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

Proof. The normalized congestion of both factor graphs is $\beta_2(R_k) = \beta_2(X_k) = 1/4$. Then, applying Corollary 2 with $r = 2$, we get $2\Psi(\alpha)/(4(1/4)) \leq BW(MCXR^{(d)})$. This yields, $BW(MCXR^{(d)}) \geq 2\Psi(\alpha)$. □

We calculate now the upper bound on 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, $BW(MCXR^{(d)}) \leq 2\Psi(\alpha)$. From Lemmas 7 and 8, Theorem 6 follows. □

Theorem 6. *The bisection width of a d-dimensional mesh-connected extended trees and rings* $MCXR^{(d)}_{k_1,k_2,\ldots,k_d}$ *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 8. *The bisection width of the d-dimensional torus* $T^{(d)}_{k_1,k_2,\ldots,k_d}$ *is* $BW(T^{(d)}) = 2\Psi(\alpha)$.

Corollary 9. *The bisection width of the d-dimensional mesh-connected extended trees* $MCX^{(d)}_{k_1,k_2,\ldots,k_d}$ *is* $BW(MCX^{(d)}) = 2\Psi(d)$.

6 BCube

We devote this section to obtain bounds on the bisection width of a d-dimensional BCube [8]. 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 a 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.

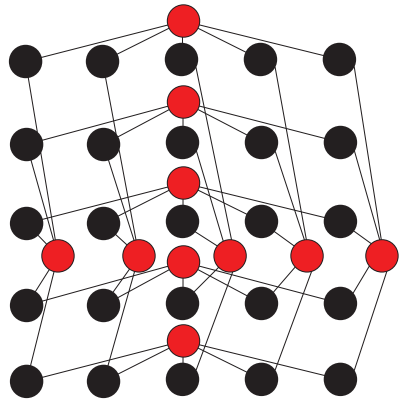


Fig. 6. Cartesian product of two SSA_5 networks.

6.1 Factor and Product Graphs

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

Definition 16. *A* Switched Star network *of k nodes, denoted SS_k, is composed of 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.*

An example of SS_5 is presented in Fig. 5a. Combining d copies of SS_k as factor networks of 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* SS_k *(the switches are not considered nodes for the Cartesian product), i.e.,* $BC_k^{(d)} = SS_k \times SS_k \times \cdots \times SS_k$.

The topology of $BC_5^{(2)}$ (which results from combining 2 copies of SS_5) is shown in Fig. 6. $BC_k^{(d)}$ 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 $(u_1, \ldots, u_i, \ldots, u_d)$ and $(v_1, \ldots, v_i, \ldots, v_d)$ are connected to the same switch if and only if $(u_i \neq v_i)$ and $u_j = v_j$ for all $j \neq i$.

Strictly speaking, the bisection width of $BC_k^{(d)}$ is the smallest number of links (connecting nodes to switches) that have to be removed to bisect it (extending the definition to networks with switches). However, 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 typical graph, the bisection width can have different forms depending on where the communication bottleneck is located in a BCube network.

We present two possible models for SS_k. The first one, Model A or *star-like model*, denoted by SSA_k, consists of k nodes connected one-to-one to a virtual node which represents the switch. This model corresponds with the actual physical topology of BCube. The second one, Model B or *hyperlink model*, denoted by SSB_k, consists of k nodes connected by a hyperlink.[4] While the two presented models are logically equivalent to a complete graph, they have a different behavior from the traffic point of view. We show this with two simple examples.

Let us consider that we have an SS_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 another situation now, where the BCube switch still supports 1 Gbps of internal traffic but the links also transmit at 1 Gbps. In this case, the switches are the bottleneck of the network and the bisection bandwidth is only 1 Gbps, although the links would be able to support up to 2 Gbps.

The first example illustrates a scenario where we would bisect the network by removing the links that connect the servers to the switches, which corresponds to Model A. On the other hand, what we find in the second example is a typical scenario for Model B, where we would do better by removing entire switches when bisecting the network. In particular, 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 B when $s \leq 2T$. (Note that this does not cover the whole spectrum of possible values of s, T, and k.)

6.2 Congestion and Central Cut of BCube

We will compute now the congestion and central cut of both models to be able to calculate the respective lower and upper bounds. We start by the congestion and central cut of Model A. If we set $r = 1$, the congestion of every link of the star is easily found[5] to be $m_r(SSA_k) = k - 1$ as shown in Fig. 5c. The central cut, which is also trivial, can be found in Fig. 5e. Both will depend on whether the number of nodes k is even or odd.

Corollary 10. *The normalized congestion of SSA_k is $\beta_r(SSA_k) = \frac{k-1}{k^2-b}$, and the central cut is $CC(SSA_k) = \frac{k-b}{2}$, where $b = k \bmod 2$.*

Having computed the congestion and the central cut for Model A, we will compute them now for Model B. 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., $m_r(SSB_k) = \frac{k(k-1)}{2}$. The central cut is also easily computed, as there is only one hyperlink. Both $m_r(SSB_k)$ and $CC(SSB_k)$ are shown in Fig. 5.

Corollary 11. *The normalized congestion of SSB_k is $\beta_r(SSB_k) = \frac{k-1}{2(k^2-b)}$, where $b = k \bmod 2$, and the central cut is $CC(SSB_k) = 1$.*

6.3 Bounds on the BBW of BCube

Having computed the congestion and central cut of both models, we can calculate the lower and upper bounds on the bisection width of each one of them. We will start by the lower and upper bounds on 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 on the bisection width of a Model A BCube.

Lemma 9. *The bisection width of a Model A d-dimensional BCube, $BCA_k^{(d)}$, 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 normalized congestion of a Model A BCube in Corollary 2, it follows that

$$BW\big(BCA_k^{(d)}\big) \geq \begin{cases} \dfrac{1}{4}\dfrac{k^2}{k-1}\Psi(\alpha) = \dfrac{k^{d+1}}{4(k-1)}, & \text{if } k \text{ is even,} \\ \dfrac{k+1}{4}\Psi(\alpha) = \dfrac{k+1}{4}\dfrac{k^d-1}{k-1}, & \text{if } k \text{ is odd.} \end{cases}$$

□

After presenting the lower bound on 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, $BCA_k^{(d)}$, 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 use the values of the central cut of Model A in Theorem 4, it is immediate to compute the following upper bound:

$$BW\big(BCA_k^{(d)}\big) \leq \begin{cases} \dfrac{k^d}{2}, & \text{if } k \text{ is even,} \\ \dfrac{k^d-1}{2}, & \text{if } k \text{ is odd.} \end{cases}$$

□

Now, from the combination of Lemmas 9 and 10, we can state the following theorem.

Theorem 7. *The value of the bisection bandwidth of a Model A d-dimensional BCube, $BCA_k^{(d)}$, is in the interval $[2T \cdot \frac{k^{d+1}}{4(k-1)}, 2T \cdot \frac{k^d}{2}]$ if k is even, and in the interval $[2T \cdot \frac{k+1}{4}\frac{k^d-1}{k-1}, 2T \cdot \frac{k^d-1}{2}]$ if k is odd.*

From the observation that the topology of BCube is the same as that of Model A, we derive the following corollary.

Corollary 12. *The bisection width of a d-dimensional BCube $BC_k^{(d)}$ satisfies*

$$BBW\big(BCA_k^{(d)}\big) \in \begin{cases} \left[\dfrac{k^{d+1}}{4(k-1)}, \dfrac{k^d}{2}\right], & \text{if } k \text{ is even,} \\ \left[\frac{k+1}{4}\frac{k^d-1}{k-1}, \frac{k^d-1}{2}\right], & \text{if } k \text{ is odd.} \end{cases}$$

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.

4. This model is quite similar to the one proposed by Pan in [21].
5. Note that in the computation of the congestion, the switch is not considered a node of the graph.

TABLE 3
Gap between Bounds (Results in Gbps)

		Model A			Model B		
k	d	BBW	LB	UB	BBW	LB	UB
3	2	0.8	0.8	0.8	4	2.667	4
4	2	1.6	1.067	1.6	5	3.125	5
7	2	4.8	3.2	4.8	8	4.571	8
8	2	6.4	3.657	6.4	8	4.571	9
3	3	2.6	2.6	2.6	13	8.667	13
4	3	6.4	4.267	6.4	16	10.667	21
7	3	34.2	22.8	34.2	57	32.571	57
8	3	51.2	29.257	51.2	64	36.571	73

Lemma 11. *The bisection width of a Model B d-dimensional BCube,* $BCB_k^{(d)}$*, 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 normalized congestion of Model B in Corollary 2. Since all the dimensions have the same size k, it follows that

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

□

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

Lemma 12. *The bisection width of a Model B d-dimensional BCube,* $BCB_k^{(d)}$*, 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 on the bisection width, $BW(BCB_k^{(d)}) \leq 1 \cdot \Psi(\alpha) = \frac{k^d-1}{k-1}$. □

Combining the previous lemmas, we can state the following theorem.

Theorem 8. *The value of the bisection bandwidth of a Model B d-dimensional BCube,* $BCB_k^{(d)}$*, is in the interval* $[s \cdot \frac{k^d}{2(k-1)}, s \cdot \frac{1-k^d}{1-k}]$ *if k is even, and in the interval* $[s \cdot \frac{k+1}{2k}\frac{k^d-1}{k-1}, s \cdot \frac{k^d-1}{k-1}]$ *if k is odd.*

In Table 3, we present some results for some concrete BCube networks. We assume the same two scenarios we used in the previous example: links of $T = 100$ Mbps and switches with $s = 1$ Gbps for Model A ($s \geq \lfloor \frac{k}{2} \rfloor \cdot 2T$); and links of $T = 1$ Gbps and switches with $s = 1$ Gbps for Model B ($s \leq 2T$).

7 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 on the bisection width of a d-dimensional BCube are also provided. Some of the product networks studied had factor graphs of the same class, like the d-dimensional torus, mesh-connected trees, or mesh-connected extended trees, while some other combined different factor graphs, like the mesh-connected trees and paths or mesh-connected extended trees and rings. See Table 2 for a summary of the results obtained.

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.

At a more general level, we believe that it would be interesting to explore the properties of product networks as topologies for data centers (currently, to our knowledge, only BCube fits in this category). The Cartesian product operation has been used in the past for building parallel processing topologies. We believe that it can be also used to build efficient topologies for data centers.

Acknowledgments

This work was supported in part by the Comunidad de Madrid grant S2009TIC-1692, Spanish MICINN grant TEC2011-29688-C02-01, and National Natural Science Foundation of China grant 61020106002.

References

[1] F.T. Leighton, *Introduction to Parallel Algorithms and Architectures: Array, Trees, Hypercubes.* Morgan Kaufmann, 1992.

[2] J. Duato, S. Yalamanchili, and N. Lionel, *Interconnection Networks: An Engineering Approach.* Morgan Kaufmann, 2002.

[3] W. Dally and B. Towles, *Principles and Practices of Interconnection Networks.* Morgan Kaufmann, 2003.

[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," *Proc. 10th Euromicro Conf. Digital System Design Architectures, Methods and Tools,* pp. 19-26, 2007.

[6] D. Zydek and H. Selvaraj, "Fast and Efficient Processor Allocation Algorithm for Torus-Based Chip Multiprocessors," *Computers and Electrical Eng.,* vol. 37, pp. 91-105, Jan. 2011.

[7] E. Salminen, A. Kulmala, and T.D. Hamalainen, "Survey of Network-on-Chip Proposals," white paper, OCP-IP, pp. 1-13, 2008.

[8] 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," *Proc. SIGCOMM,* pp. 63-74, 2009.

[9] C. Guo, H. Wu, K. Tan, L. Shi, Y. Zhang, and S. Lu, "Dcell: A Scalable and Fault-Tolerant Network Structure for Data Centers," *Proc. SIGCOMM,* pp. 75-86, 2008.

[10] A. Youssef, "Cartesian Product Networks," *Proc. Int'l Conf. Parallel Processing,* vol. 1, pp. 684-685, 1991.

[11] A. Youssef, "Design and Analysis of Product Networks," *Proc. Fifth Symp. Frontiers of Massively Parallel Computation (Frontiers '95),* pp. 521-528, 1995.

[12] W.J. Dally, "Performance Analysis of K-Ary N-Cube Interconnection Networks," *IEEE Trans. Computers,* vol. 39, no. 6, pp. 775-785, June 1990.
[13] J.D.P. Rolim, O. Sýkora, and I. Vrto, "Optimal Cutwidths and Bisection Widths of 2- and 3-Dimensional Meshes," *Proc. 21st Int'l Workshop Graph-Theoretic Concepts in Computer Science (WG),* pp. 252-264, 1995.
[14] K. Efe and A. Fernández, "Products of Networks with Logarithmic Diameter and Fixed Degree," *IEEE Trans. Parallel and Distributed Systems,* vol. 6, no. 9, pp. 963-975, Sept. 1995.
[15] K. Nakano, "Linear Layout of Generalized Hypercubes," *Int'l J. Foundations Computer Science,* vol. 14, no. 1, pp. 137-156, 2003.
[16] K. Efe and G.-L. Feng, "A Proof for Bisection Width of Grids," *Proc. World Academy of Science, Eng. and Technology,* vol. 27, no. 31, pp. 172-177, 2007.
[17] M.C. Azizoğlu and Ö. Eğecioğlu, "The Isoperimetric Number and the Bisection Width of Generalized Cylinders," *Electronic Notes in Discrete Math.,* vol. 11, pp. 53-62, 2002.
[18] M.C. Azizoğlu and Ö. Eğecioğlu, "The Bisection Width and the Isoperimetric Number of Arrays," *Discrete Applied Math.,* vol. 138, nos. 1/2, pp. 3-12, 2004.
[19] K. Efe and A. Fernández, "Mesh-Connected Trees: A Bridge between Grids and Meshes of Trees," *IEEE Trans. Parallel and Distributed Systems,* vol. 7, no. 12, pp. 1281-1291, Dec. 1996.
[20] M.C. Azizoğlu and Ö. Eğecioğlu, "Extremal Sets Minimizing Dimension-Normalized Boundary in Hamming Graphs," *SIAM J. Discrete Math.,* vol. 17, no. 2, pp. 219-236, 2003.
[21] Y. Pan, S.Q. Zheng, K. Li, and H. Shen, "An Improved Generalization of Mesh-Connected Computers with Multiple Buses," *IEEE Trans. Parallel and Distributed Systems,* vol. 12, no. 3, pp. 293-305, Mar. 2001.

Jordi Arjona Aroca received the undergraduate studies (Ingeniería Superior de Telecomunicaciones) from the Universidad Politécnica de Valencia and the master's degree in telemática from the Universidad Carlos III de Madrid, Spain. He is currently working toward the PhD degree at the Institute IMDEA Networks and the Universidad Carlos III de Madrid, where he is involved in the field of energy optimization in data centers and data center networking. He has published his work in some international conferences such as the 39th Annual IEEE/IFIP International Conference on Dependable Systems and Networks and the Ninth Annual Conference on Theory and Applications of Models of Computation. He has also collaborated as a reviewer for some venues like the International Colloquium on Automata, Languages and Programming, IEEE Consumer Communications Networking Conference, and *IEEE Communication Letters*.

Antonio Fernández Anta (M'98-SM'02) received the undergraduate studies (Licenciado and Diplomado en Informática) from the Universidad Politécnica de Madrid, Spain, in 1988 and 1991, respectively, having received awards at the university and national level for his academic performance. He received the MSc and PhD degrees in computer science from the University of Louisiana in 1992 and 1994, respectively. He is a senior researcher at Institute IMDEA Networks. Previously, he was a full professor at the Universidad Rey Juan Carlos in Madrid, where he has been on the faculty since 1998. He was a faculty of the Universidad Politécnica de Madrid, where he received an award for his research productivity. He was a postdoc at the Massachusetts Institute of Technology from 1995 to 1997. He has almost 20 years of research experience, with a steady productivity of more than five papers per year on average. He has published in top conferences and journals like INFOCOM, STOC, FOCS, PODC, DISC, *Journal of the ACM*, *SIAM Journal on Computing*, and *IEEE Transactions on Computers*. He is a vice chair of the Steering Committee of DISC and has served in the TPC of numerous conferences and workshops. He has been a senior member of the IEEE since 2002 and of the ACM since 2007.