This paper was presented as part of the Mini-Conference at IEEE INFOCOM 2011

BCN: Expansible Network Structures for Data Centers Using Hierarchical Compound Graphs

Deke Guo*†, Tao Chen†, Dan Li‡, Yunhao Liu§, Xue Liu¶, Guihai Chen‖
*School of Computer Science and Technology, Huazhong University of Science and Technology, China
†School of Information System and Management, National University of Defense Technology, China
‡Tsinghua University, China, §Hong Kong University of Science and Technology, Hong Kong
¶University of Nebraska-Lincoln, USA, ‖Shanghai Jiaotong University, China

***Abstract*—A fundamental challenge in data centers is how to design networking structures for efficiently interconnecting a large number of servers. Several server-centric structures have been proposed, but are not truly expansible and suffer low degree of regularity and symmetry. To address this issue, we propose two novel structures called HCN and BCN, which utilize hierarchical compound graphs to interconnect large population of servers each with two ports only. They own two topological advantages, i.e., the expansibility and equal degree. In addition, HCN offers high degree of regularity, scalability and symmetry, which well conform to the modular design of data centers. Moreover, a BCN of level one in each dimension involves more servers than FiConn with server degree 2 and diameter 7, and is large enough for a single data center. Mathematical analysis and comprehensive simulations show that BCN possesses excellent topology properties and is a viable network structure for data centers.**

I. Introduction

Mega data centers have emerged as infrastructures for building online applications, such as web search, email and on-line gaming, as well as infrastructural services, such as GFS [1], HDFS [2], and BigTable [3]. Inside a data center, large number of servers are interconnected using a specific data center networking (DCN) structure.

The tree-based structures are increasingly difficult to meet the design goals of data centers [4], [5], [6]. Consequently, a number of novel DCN structures are proposed recently. These structures can be roughly divided into two categories. One is switch-centric, which organizes switches into structures other than tree and puts interconnection intelligence on switches. Fat-Tree [4], VL2 [7] fall into this category. The other is server-centric, which puts interconnection intelligence on servers and uses switches only as cross-bars. DCell [5], BCube [8] and FiConn [6] fall into the second category. Among others, server-centric structures have the following advantages. In current practice, servers are more programmable than switches, so the deployment of new DCN structure is more feasible.

For DCell and BCube, many nice topological properties and efficient algorithms have been derived at the following cost. They use more than 2 ports, typically 4, and large number of links. If they use servers with only 2 ports, the network order are very limited and cannot be enlarged since they are at most two layers. When network structures are expanded to one higher level, DCell as well as BCube add one NIC and link for each existing server, and BCube is appended large number of additional switches. Hence, a major drawback of these topologies is that they are not truly expansible. A network is expansible if no changes with respect to node configuration and link connections are necessary when it is expanded. This might cause negative influence on applications running on all existing servers during the process of topology expansion.

A. Motivation and Contributions

With the considerations of easy implementation and low costs of hardware and wiring, an expansible data center should be equipped with servers with constant number of NIC ports, most desirable of 2, which exists in most commodity servers in current data centers. FiConn is one of such kind of topologies, however, suffers low degree of regularity and symmetry, which are desirable to the modular design of distributed systems involving a large number of computing elements.

We first propose a hierarchical irregular compound network, denoted as HCN, which can be expanded independent of the server degree by only adding one link to a few number of servers. Moreover, HCN offers high degree of regularity, scalability and symmetry, which very well conform to the modular design of data centers. Inspired by the smaller network order of HCN than FiConn, we propose BCN, Bidimensional Compound Networks for data centers, which inherits the advantages of HCN and has higher network order than FiConn. A BCN of level one in each dimension is the largest known DCN with server degree 2 and network diameter 7. For example, if 48-port switches are used, BCN offers 787,968 servers, while a level-2 FiConn only supports 361,200 servers.

B. Related work

Definition 1: Given two regular graphs G and G_1, a level-1 regular compound graph $G(G_1)$ is obtained by replacing each node of G by a copy of G_1 and replacing each link of G by a link which connects corresponding two copies of G_1.

A level-1 regular compound graph $G(G_1)$ employs G_1 as a unit cluster and connects many such clusters by means of a regular graph G. In the resultant graph, the topology of G is preserved and only one link is inserted to connect two copies of G_1. An additional remote link is associated to each node in a cluster. For each node in the resultant network, the degree is identical. A *constraint* must be satisfied for the two graphs to constitute a regular compound graph. That is, the node degree

978-1-4244-9921-2/11/$26.00 ©2011 IEEE

of G must be equal to the number of nodes in G_1. Otherwise, we obtain an *irregular compound graph*.

A level-1 regular compound graph can be extended to level-i ($i\geq 2$) recursively. A level-i ($i>0$) regular graph $G^i(G_1)$ adopts a level-$(i-1)$ regular graph $G^{i-1}(G_1)$ as a unit cluster and connects many such clusters by a regular graph G.

II. The BCN Network Structure

We start with two expansible network structures, i.e., HCN and BCN, which build scalable and low-cost data centers using dual port servers. We then propose the routing algorithms.

A. Hierarchical Irregular Compound Networks

For any given $h\geq 0$, we denote a level-h irregular compound network as $HCN(n,h)$. HCN is a recursively defined structure. A high-level $HCN(n,h)$ employs a low level $HCN(n,h-1)$ as a unit cluster and connects many such clusters by means of a complete graph. $HCN(n,0)$ is the smallest module (basic construction unit), which consists of n servers each with dual-ports and a n-port mini-switch. For each server, its first port is used to connect with the mini-switch while its second port is employed to interconnect with another server in different smallest modules for constituting larger networks. A server is *available* if its second port has not been connected.

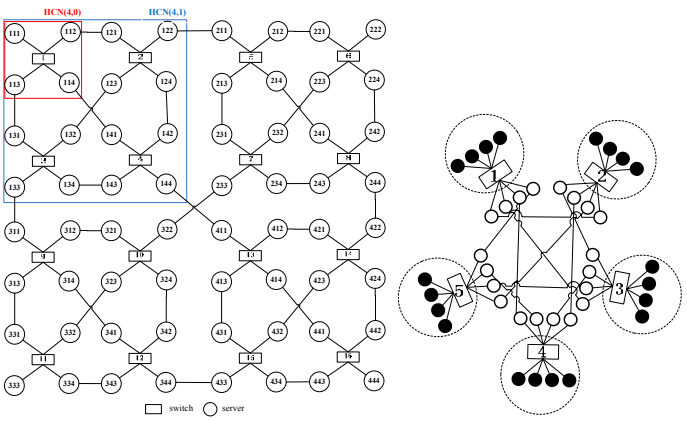
$HCN(n,1)$ is constructed using n basic modules $HCN(n,0)$. In $HCN(n,1)$, there is only one link between any two basic modules by connecting two *available* servers that belong to different basic modules. Consequently, for each $HCN(n,0)$ of $HCN(n,1)$ all its servers are associated with a level-1 link except one server which is reserved for the construction of $HCN(n,2)$. Thus, there are n available servers in $HCN(n,1)$ for further expansion at a higher level. Similarly, $HCN(n,2)$ is formed by n level-1 $HCN(n,1)$, and has n available servers for interconnection at a higher level.

In general, $HCN(n,i)$ for $i\geq 0$ is formed by n $HCN(n,i-1)$, and has n available servers each in a $HCN(n,i-1)$ for further expansion of network. According to Definition 1, $H(n,i)$ acts as G_1 and a complete graph acts as G. Here, $G(G_1)$ produces an irregular compound graph since the number of available servers in $H(n,i)$ is n while the node degree of G is $n-1$. To facilitate the construction of any level-h HCN, we define Definition 2 as follows.

Definition 2: Each server in $HCN(n,h)$ is assigned a label $x_h\cdots x_1x_0$, where $1\leq x_i\leq n$ for $0\leq i\leq h$. Two servers $x_h\cdots x_1x_0$ and $x_h\cdots x_{j+1}x_{j-1}x_j^j$ are connected only if $x_j\neq x_{j-1}$, $x_{j-1}=x_{j-2}=\cdots=x_1=x_0$ for some $1\leq j\leq h$, where $1\leq x_0\leq\alpha$ and x_j^j represents j consecutive x_js. Here, n servers are reserved for further expansion only if $x_h=x_{h-1}=\cdots=x_0$ for any $1\leq x_0\leq n$.

In any level-h HCN, each server achieves a unique label produced by Definition 2 and is appended a link to its second port. Fig.1(a) plots an example of $HCN(4,2)$ according to Definition 2. $HCN(4,2)$ consists of four $H(4,1)$s, while each $H(4,1)$ has four $H(4,0)$s. The second port of four servers 111, 222, 333, and 444 are reserved for further expansion.

In a level-h HCN, each server recursively belongs to level-0, level-1,level-2,...,level-h HCNs, respectively. Similarly, any lower level HCN belongs to many higher level



(a) An example of $HCN(n,h)$, where n=4 and h=2. (b) An example of $G(BCN(4,4,0))$.

Fig. 1. Illustrative examples of $HCN(n,h)$ and $G(BCN(4,4,0))$.

HCNs. To characterize this property, let x_i indicate the order of a $HCN(n,i-1)$, containing a server $x_h\cdots x_1x_0$, among all level-$(i-1)$ HCNs of $HCN(n,i)$ for $1\leq i\leq h$. We further use $x_hx_{h-1}\cdots x_i$ ($1\leq i\leq h$) as a prefix to indicate the $HCN(n,i-1)$ that contains this server in $HCN(n,h)$. We use server 423 as an example. x_1=2 indicates the 2^{th} $HCN(4,0)$ in a $HCN(4,1)$ this server is located at. This $HCN(4,0)$ contains the servers 421, 422, 423, and 444. x_2=4 indicates the 4^{th} level-1 HCN in a level-2 HCN that contains this server. Thus, x_2x_1=42 indicates the level-0 HCN that contains the server 423 in a level-2 HCN.

We have emphasized two topological advantages, i.e., expansibility and equal degree, with the consideration of easy implementation and low cost. HCN owns the two properties, and offers high degree of regularity, scalability and symmetry, which very well conform to the modular design of data centers. Inspired by the fact that the order of HCN is less than that of FiConn under the same configurations, we further propose the structure of BCN on the basis of HCN.

B. BCN Physical Structure

BCN is a multi-level irregular compound graph recursively defined in the first dimension, and a level one regular compound graph in the second dimension. In each dimension, a high-level BCN employs a one low level BCN as a unit cluster and connects many clusters by means of a complete graph.

Let $BCN(\alpha,\beta,0)$ denote the basic building block, where $\alpha+\beta=n$. It has n servers and a n-port mini-switch. All servers are connected to the mini-switch using their first ports, and are partitioned into two disjoint groups, referred to as the *master* and *slave* servers. Let α and β be the number of master servers and slave servers, respectively. As discussed later, the second port of master servers and slave servers are used to constitute larger BCNs in the first and second dimensions, respectively.

1) Hierarchical BCN in the first dimension: For any given $h\geq 0$, we use $BCN(\alpha,\beta,h)$ to denote a level-h BCN formed by all *master* servers in the first dimension. For any $h>1$, $BCN(\alpha,\beta,h)$ is an irregular compound graph, where G is a complete graph with α nodes while G_1 is $BCN(\alpha,\beta,h-1)$ with α available master servers. It is worth noticing that, for any $h\geq 0$, $BCN(\alpha,\beta,h)$ still has α available master servers

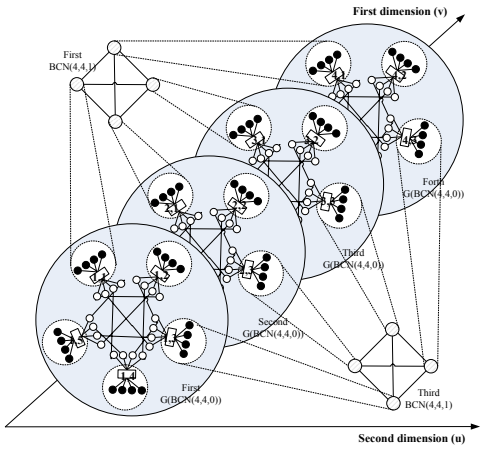


Fig. 2. An illustrative example of $BCN(4,4,1,0)$.

for further expansion, and is equivalent to $HCN(\alpha,h)$. The only difference is that each mini-switch also connects β slave servers besides α master servers in $BCN(\alpha,\beta,h)$.

2) Hierarchical BCN in the second dimension: There are β available slave servers in the smallest module $BCN(\alpha,\beta,0)$. In general, there are $s_h{=}\alpha^h{\cdot}\beta$ available slave servers in any given $BCN(\alpha,\beta,h)$ for $h{\geq}0$. We study how to utilize those available slave servers to expand $BCN(\alpha,\beta,h)$ from the second dimension. A level-1 regular compound graph $G(BCN(\alpha,\beta,h))$ is a natural way to realize this goal. It uses $BCN(\alpha,\beta,h)$ as a unit cluster and connects $s_h{+}1$ copies of $BCN(\alpha,\beta,h)$ by means of a complete graph using the second port of all available slave servers. The resulting $G(BCN(\alpha,\beta,h))$ cannot be further expanded in the second dimension since it has no available slave servers. It, however, still can be expanded in the first dimension without destroying the existing network. Fig.1(b) plots an example of $G(BCN(4,4,0))$.

3) Bidimensional hierarchical BCN: After separately designing $BCN(\alpha,\beta,h)$ and $G(BCN(\alpha,\beta,h))$, we design a scalable Bidimensional BCN formed by both master and slave servers. Let $BCN(\alpha,\beta,h,\gamma)$ denote a Bidimensional BCN, where h denote the level of BCN in the first dimension, and γ denotes the level of a BCN which is selected as the unit cluster in the second dimension.

To increase servers in data centers on-demand, it is required to expand an initial lower-level $BCN(\alpha,\beta,h)$ from the first or second dimension without destroying an existing structure. A Bidimensional BCN is always $BCN(\alpha,\beta,h)$ as h increases when $h{<}\gamma$. In this scenario, the unit cluster for expansion in the second dimension has not been formed. When h increases to γ, we achieve $BCN(\alpha,\beta,\gamma)$ in the first dimension and then expand it from the second dimension using the construction method of $G(BCN(\alpha,\beta,\gamma))$ in Section II-B2. In the resulting $BCN(\alpha,\beta,\gamma,\gamma)$, there are $\alpha^{\gamma}{\cdot}\beta{+}1$ copies of $BCN(\alpha,\beta,\gamma)$ and α available master servers in each $BCN(\alpha,\beta,\gamma)$. A sequential number u is employed to identify a $BCN(\alpha,\beta,\gamma)$ among $\alpha^{\gamma}{\cdot}\beta{+}1$ ones in the second dimension, where u ranges from 1 to $\alpha^{\gamma}{\cdot}\beta{+}1$. The example in Fig.1(b) is also a $BCN(4,4,0,0)$ consisting of five $BCN(4,4,0)$, where $h{=}r{=}0$.

We further consider the case that h exceeds γ. That is, each $BCN(\alpha,\beta,\gamma)$ in $BCN(\alpha,\beta,\gamma,\gamma)$ becomes $BCN(\alpha,\beta,h)$ in the first dimension once h exceeds γ. There are $\alpha^{h-\gamma}$ homogeneous $BCN(\alpha,\beta,\gamma)$ inside each $BCN(\alpha,\beta,h)$. Thus, we use a sequential number v to identify a $BCN(\alpha,\beta,\gamma)$ inside each $BCN(\alpha,\beta,h)$ in the first dimension, where v ranges from 1 to $\alpha^{h-\gamma}$. Thus, the coordinate of each $BCN(\alpha,\beta,\gamma)$ in the resulting structure is denoted by a pair of v and u.

It is worth noticing that only those $BCN(\alpha,\beta,\gamma)$ with $v{=}1$ in the resulting structure are connected by a complete graph in the second dimension, and form the first $G(BCN(\alpha,\beta,\gamma))$. Consequently, messages between any two servers in different $BCN(\alpha,\beta,\gamma)$ with the same value of v except $v{=}1$ must be relayed by related $BCN(\alpha,\beta,\gamma)$ in the first $G(BCN(\alpha,\beta,\gamma))$. Thus, the first $G(BCN(\alpha,\beta,\gamma))$ becomes a bottleneck of the resulting structure. To address this issue, all $BCN(\alpha,\beta,\gamma)$ with $v{=}i$ are also connected by means of a completed graph and produce the i^{th} $G(BCN(\alpha,\beta,\gamma))$, for other values of v besides 1. By now, we achieve $BCN(\alpha,\beta,h,\gamma)$ in which each $G(BCN(\alpha,\beta,\gamma))$ is a regular compound graph mentioned in Section I-B, where G is a complete graph with $\alpha^{\gamma}{\cdot}\beta$ nodes and G_1 is a $BCN(\alpha,\beta,\gamma)$ with $\alpha^{\gamma}{\cdot}\beta$ available slave servers.

Fig.2 plots a $BCN(4,4,1,0)$ formed by all master and slave servers from the first and second dimensions. Note that only the first and third $BCN(4,4,1)$ are plotted, while other three $BCN(4,4,1)$ are not shown due to page limitations. We can see that $BCN(4,4,1,0)$ has five homogeneous $BCN(4,4,1)$ in the second dimension and four homogeneous $G(BCN(4,4,0))$ in the first dimension.

C. The Construction Methodology of BCN

1) In the case of $h{<}\gamma$: In this case, $BCN(\alpha,\beta,h)$ can be achieved by the construction methodology of $HCN(\alpha,h)$ as mentioned in Section II-A.

2) In the case of $h{=}\gamma$: As mentioned in Section II-B2, all slave servers in $BCN(\alpha,\beta,\gamma)$ are utilized for expansion in the second dimension. Each slave server in $BCN(\alpha,\beta,\gamma)$ is identified by a unique label $x{=}x_{\gamma}\cdots x_1x_0$ where $1{\leq}x_i{\leq}\alpha$ for $1{\leq}i{\leq}\gamma$ and $\alpha{+}1{\leq}x_0{\leq}n$. Besides the unique label, each slave server can be equivalently identified by a unique $id(x)$ which denotes its order among all slave servers in $BCN(\alpha,\beta,\gamma)$ and ranges from 1 to s_{γ}. For each slave server, the mapping between a unique id and its label is bijection defined in Theorem 1. Meanwhile, the label can be derived from its unique id in a reversed way.

Theorem 1: For any slave server $x{=}x_{\gamma}\cdots x_1x_0$, its unique id is given by

$$id(x_{\gamma}\cdots x_1x_0)=\sum_{i=1}^{\gamma}(x_i{-}1){\cdot}\alpha^{i-1}{\cdot}\beta+(x_0{-}\alpha). \qquad (1)$$

As mentioned in Section II-B2, the resultant BCN network when $h{=}\gamma$ is $G(BCN(\alpha,\beta,\gamma))$ consisting of $s_{\gamma}{+}1$ copies of a unit cluster $BCN(\alpha,\beta,\gamma)$. In this case, $BCN_u(\alpha,\beta,\gamma)$ denotes the u^{th} unit cluster in the second dimension. In $BCN_u(\alpha,\beta,\gamma)$, each server is assigned a unique label $x{=}x_{\gamma}\cdots x_1x_0$ and a 3-tuples $[v(x){=}1,u,x]$, where $v(x)$ is defined in Theorem 2. In $BCN_u(\alpha,\beta,\gamma)$, all master servers are interconnected according to the rules in Definition 2 for $1{\leq}u{\leq}s_{\gamma}{+}1$.

Many ways can interconnect all slave servers in $s_\gamma+1$ homogeneous $BCN(\alpha,\beta,\gamma)$ to constitute a $G(BCN(\alpha,\beta,\gamma))$. For any two slave servers $[1,u_s,x_s]$ and $[1,u_d,x_d]$, as mentioned in literatures [9] they are interconnected only if

$$\begin{aligned} u_d &= (u_s+id(x_s)) \bmod (s_\gamma+2) \\ id(x_d) &= s_\gamma+1-id(x_s), \end{aligned} \tag{2}$$

where $id(x_s)$ and $id(x_d)$ are calculated by Formula 1. In literature [5], the two slave servers are connected only if

$$\begin{aligned} u_s &> id(x_s) \\ u_d &= id(x_s) \\ id(x_d) &= (u_s-1) \bmod s_\gamma. \end{aligned} \tag{3}$$

This paper does not design new interconnection methods of all slave servers since the above two and other permutation methods are all suitable to constitute $G(BCN(\alpha,\beta,\gamma))$. Please refer literatures [5], [9] for more information.

3) In the case of $h>\gamma$: After achieving $BCN(\alpha,\beta,\gamma,\gamma)$, the resulting network can be incrementally expanded in the first dimension without destroying the existing structure. As discussed in Section II-B3, $BCN(\alpha,\beta,h,\gamma)$ ($h>\gamma$) consits of $s_\gamma+1$ copies of a unit cluster $BCN(\alpha,\beta,h)$ in the second dimension. Each server in $BCN_u(\alpha,\beta,h)$, the u^{th} unit cluster of $BCN(\alpha,\beta,h,\gamma)$, is assigned a unique label $x=x_h\cdots x_1x_0$ for $1\le u\le s_\gamma+1$. In addition, $BCN_u(\alpha,\beta,h)$ has $\alpha^{h-\gamma}$ $BCN(\alpha,\beta,\gamma)$ in the first dimension. Recall that a sequential number v is employed to rank those $BCN(\alpha,\beta,\gamma)$ in $BCN_u(\alpha,\beta,h)$.

In $BCN_u(\alpha,\beta,h)$, each server $x=x_h\cdots x_1x_0$ is assigned a 3-tuples $[v(x),u,x]$, where $v(x)$ is defined in Theorem 2. A pair of u and $v(x)$ is sufficient to identify the unit cluster $BCN(\alpha,\beta,\gamma)$ that contains the server x in $BCN(\alpha,\beta,h,\gamma)$. For a slave server x, we further assign a unique $id(x_\gamma\cdots x_1x_0)$ to indicates the order of x among all slave servers in the same $BCN(\alpha,\beta,\gamma)$.

Theorem 2: For any server labeled $x=x_h\cdots x_1x_0$ for $h\ge\gamma$, the rank of the module $BCN(\alpha,\beta,\gamma)$ in $BCN(\alpha,\beta,h)$ this server resides in is given by

$$v(x) = \begin{cases} 1, \text{ if } h=\gamma \\ x_{\gamma+1}, \text{ if } h=\gamma+1 \\ \sum_{i=\gamma+2}^{h}(x_i-1)\cdot\alpha^{i-\gamma-1}+x_{\gamma+1}, \text{ if } h>\gamma+1 \end{cases} \tag{4}$$

After assigning a 3-tuples to all master and slave servers, we propose a general procedure to constitute a $BCN(\alpha,\beta,h,\gamma)$ ($h>\gamma$)The entire produce includes three parts. The first part groups all servers into the smallest modules $BCN(\alpha,\beta,0)$ for further expansion. The second part constructs $s_\gamma+1$ homogeneous $BCN(\alpha,\beta,h)$ by connecting the second port of those master servers which have the same u and satisfy the constraints mentioned in Definition 2. Furthermore, the third part connects the second port of those slave servers that have the same v and satisfy the constraints defined by Formula 2. Consequently, the construction produce results in $BCN(\alpha,\beta,h,\gamma)$ consisting of $\alpha^{h-\gamma}$ homogeneous $G(BCN(\alpha,\beta,\gamma))$. Note that it is not necessary that the connection rule of all slave servers must be Formula 2. It also can be that defined by Formula 3.

D. Routing for one-to-one traffic in BCN

1) In the case of $h<\gamma$: For any $BCN(\alpha,\beta,h)$ ($1\le h$) in the first dimension, we propose an efficient routing scheme, denoted as *FdimRouting*, to find a single-path between any pair of servers in a distributed manner. Let *src* and *dst* denote the source and destination servers in the same $BCN(\alpha,\beta,h)$ but different $BCN(\alpha,\beta,h-1)$. The source and destination can be of master server or slave server. The routing scheme first determines the link $(dst1,src1)$ that interconnects the two $BCN(\alpha,\beta,h-1)$ that *src* and *dst* are located at. It then derives two sub-paths from *src* to $dst1$ and from $src1$ to *dst*. The path from *src* to *dst* is the combination of the two sub-paths and $(dst1,src1)$. Each of the two sub-paths can be obtained by invoking *FdimRouting* recursively.

From *FdimRouting*, we obtain the following theorem. Note that the length of the path between two servers connecting to the same switch is one.

Theorem 3: The longest shortest path length among all the server pairs of $BCN(\alpha,\beta,h)$ is at most $2^{h+1}-1$ for $h\ge 0$.

2) In the case of $h\ge\gamma$: Consider the routing scheme in any $BCN(\alpha,\beta,h,\gamma)$ consisting of $\alpha^\gamma\cdot\beta+1$ copies of $BCN(\alpha,\beta,h)$ for $h\ge\gamma$. The *FdimRouting* scheme can discover a path only if the two servers are located at the same $BCN(\alpha,\beta,\gamma)$. In other cases, *FdimRouting* alone cannot guarantee to find a path between any pair of servers. To handle this issue, we propose *BdimRouting* scheme for the cases that $h\ge\gamma$.

For any pair of servers *src* and *dst* in $BCN(\alpha,\beta,h,\gamma)$ ($h\ge\gamma$), *BdimRouting* invokes *FdimRouting* to discover the path between the two servers only if they are in the same $BCN(\alpha,\beta,h)$. Otherwise, it first identifies the link $(dst1,src1)$ that interconnects the $v(src)^{th}$ $BCN(\alpha,\beta,\gamma)$ of $BCN_{u_s}(\alpha,\beta,h)$ and $BCN_{u_d}(\alpha,\beta,h)$. Note that the link that connects the $v(dst)^{th}$ instead of the $v(src)^{th}$ $BCN(\alpha,\beta,\gamma)$ of $BCN_{u_s}(\alpha,\beta,h)$ and $BCN_{u_d}(\alpha,\beta,h)$ is an alternative link. *BdimRouting* then derives a sub-path from *src* to $dst1$ that are in the $v(src)^{th}$ $BCN(\alpha,\beta,\gamma)$ inside $BCN_{u_s}(\alpha,\beta,h)$ and another sub-path from $src1$ to *dst* that are in $BCN_{u_d}(\alpha,\beta,h)$ by invoking *FdimRouting*. Consequently, the path from *src* to *dst* is the combination of the two sub-paths and $(dst1,src1)$.

From *BdimRouting*, we obtain the following theorem.

Theorem 4: The longest shortest path length among all the server pairs of $BCN(\alpha,\beta,h,\gamma)$ ($h>\gamma$) is at most $2^{h+1}+2^{\gamma+1}-1$.

III. Evaluation

In this section, we analyze several basic topology properties of BCN, and then conduct simulations to evaluate the average path length and the robustness of routing algorithms.

A. Large Network Order

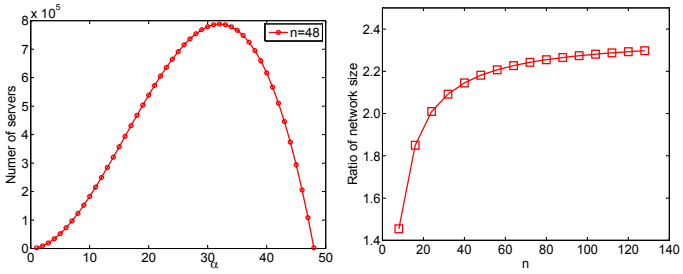
Lemma 1: The total number of servers in $BCN(\alpha,\beta,h)$ is $\alpha^h\cdot(\alpha+\beta)$, including α^{h+1} master and $\alpha^h\cdot\beta$ slave servers.

Lemma 2: The number of servers in $G(BCN(\alpha,\beta,h))$ is $\alpha^h\cdot(\alpha+\beta)\cdot(\alpha^h\cdot\beta+1)$, including $\alpha^{h+1}\cdot(\alpha^h\cdot\beta+1)$ and $\alpha^h\cdot\beta\cdot(\alpha^h\cdot\beta+1)$ master and slave servers, respectively.

Theorem 5: The number of servers in $BCN(\alpha,\beta,h,\gamma)$ is

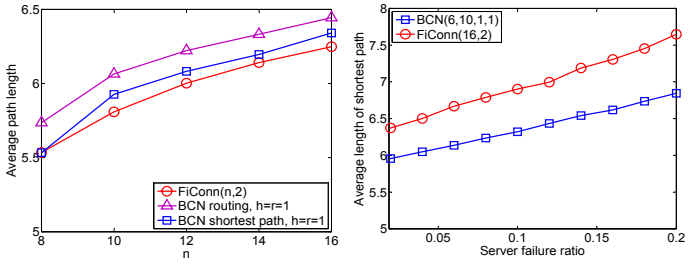
$$\begin{cases} \alpha^h\cdot(\alpha+\beta), \text{ if } h<\gamma \\ \alpha^{h-\gamma}\cdot\left(\alpha^\gamma\cdot(\alpha+\beta)\cdot(\alpha^\gamma\cdot\beta+1)\right), \text{ if } h\ge\gamma \end{cases} \tag{5}$$

Theorem 6: For any given $n=\alpha+\beta$, the optimal α that maximizes the total number of servers in $BCN(\alpha,\beta,\gamma,\gamma)$ is given by $\alpha\approx(2\cdot\gamma\cdot n)/(2\cdot\gamma+1)$.



(a) The network order of $BCN(\alpha,\beta,1,1)$. (b) The ratio of network order of $BCN(\alpha,\beta,1,1)$ to that of FiConn(n,2).

Fig. 3. The network order vs α and the ratio of network order vs. n.



(a) The average length of shortest path and routing path vs. the value of n. (b) The average length of routing paths for BCN and FiConn vs. the server failure ratio.

Fig. 4. The path length of *BCN* and *FiConn* under different scenarios.

Fig.3(a) plots the number of servers in $BCN(\alpha,\beta,1,1)$ when n=48. The network order goes up and then goes down after it reaches the peak point as α increases in the both cases. The largest network order of $BCN(\alpha,\beta,1,1)$ is 787,968 for n=48, and can be achieved only if α=32. This matches well with Theorem 6. Fig.3(b) depicts the changing trend of the ratio of network order of BCN$(\alpha,\beta,1,1)$ to that of FiConn$(n,2)$ as the number of ports in each mini-switch increases, where α is assigned the optimal value $\alpha \approx (2\cdot\gamma\cdot n)/(2\cdot\gamma+1)$. The results show that the number of servers of BCN is significantly larger than that of FiConn$(n,2)$ with the same server degree 2 and network diameter 7, irrespective the value of n.

B. Low Diameter and Server Degree

According to Theorems 3 and 4, we obtain that the diameters of $BCN(\alpha,\beta,h)$ and $BCN(\alpha,\beta,h,\gamma)$ ($h<\gamma$) are $2^{h+1}-1$ and $2^{\gamma+1}+2^{h+1}-1$, respectively. In practice, h and γ are small integers. Therefore, BCN is a low-diameter network.

After measuring the network order and diameter of BCN, we study the node degree distribution in $BCN(\alpha,\beta,h,\gamma)$. If $h<\gamma$, the node degree of master servers are 2 except the α available master servers for further expansion. The α master servers and all slave servers are of degree 1. Otherwise, there are $\alpha\cdot(\alpha^{\gamma}\cdot\beta+1)$ available master servers that are of degree 1. Other master servers and all slave servers are of degree 2.

C. Evaluation of Path Length

We run simulations on $BCN(\alpha,\beta,1,1)$ and FiConn$(n,2)$ in which $n\in\{8,10,12,14,16\}$ and α is assigned its optimal value. The ratio of network order of BCN to that of FiConn varies between 1.4545 and 1.849. For the all to all traffic, Fig.4(a) shows the average length of the shortest path of FiConn, the shortest path of BCN, and the routing path of BCN. For any BCN, the routing path length is a little bit larger than the shortest path length since the current routing protocols do not entirely realize the shortest path routing. Although the network order of BCN is a lot larger than that of FiConn, the average shortest path length is a little bit larger than that of BCN.

Then we evaluate the fault-tolerance ability of the topology and routing algorithm of BCN$(6,10,1,1)$ and FiConn$(16,2)$. The network sizes of BCN$(6,10,1,1)$ and FiConn$(16,2)$ are 5856 and 5327, respectively. As shown in Fig.4(b), the average routing path length of BCN is a lot shorter than that of FiConn, when the server failure ratio is not zero. These results demonstrate that the topology and routing algorithm of BCN possess better fault-tolerant ability.

IV. Conclusion

We present HCN and BCN, two structures for data centers involving servers with two-ports only. They own the advantages of expansibility and equal degree. Moreover, HCN offers high degree of regularity, scalability and symmetry which very well conform to the modular design and implementation of data centers. A BCN of level one in each dimension is the largest known DCN with server degree 2 and diameter 7. Analysis and simulations show that HCN and BCN are viable structures for data centers. Note that the proofs of all theorems and lemmas are presented in our technical report [10].

V. Acknowledgement

This work is supported in part by the Research Foundation of NUDT, the NSF China under Grants Nos. 60825205, 60903206, 61070216, and 61073152, the National Basic Research Program of China under Grant No. A0420070009, and the China Postdoctoral Science Foundation under Grant No. 20100480898.

References

[1] S. Ghemawat, H. Gobioff, and S.-T. Leung, "The google file system," in *Proc. SOSP*, Bolton Landing, NY, USA, 2003, pp. 29–43.
[2] D. Borthakur. The hadoop distributed file system: Architecture and design. [Online]. Available: http://hadoop.apache.org
[3] F. Chang, J. Dean, S. Ghemawat, W. C. Hsieh, and etc., "Bigtable: A distributed storage system for structured data," *ACM Transactions on Comput Systems*, vol. 26, no. 2, 2008.
[4] M. A. Fares, A. Loukissas, and A. Vahdat, "A scalable, commodity data center network architecture," in *Proc. SIGCOMM*, Seattle, Washington, USA, 2008.
[5] 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 *Proc. SIGCOMM*, Seattle, Washington, USA, 2008.
[6] D. Li, C. Guo, H. Wu, Y. Zhang, and S. Lu, "Ficonn: Using backup port for server interconnection in data centers," in *Proc. IEEE INFOCOM*, Brazil, 2009.
[7] A. Greenberg, N. Jain, S. Kandula, C. Kim, P. Lahiri, D. A. Maltz, and P. P. and, "VL2: A scalable and flexible data center network," in *Proc. SIGCOMM*, Barcelona, Spain, 2009.
[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," in *Proc. SIGCOMM*, Barcelona, Spain, 2009.
[9] P. T. Breznay and M. A. Lopez, "A class of static and dynamic hierarchical interconnection networks," in *Proc. IEEE ICPP*, vol. 1, 1994, pp. 59–62.
[10] D. Guo, T. Chen, D. Li, and Y. Liu, "Expansible network structures for data centers using hierarchical compound graphs," National University of Defense Technology, China, Tech. Rep., Dec. 2010.