Journal of Parallel and Distributed Computing 202 (2025) 105095

Contents lists available at ScienceDirect

Journal of Parallel and Distributed Computing

journal homepage: www.elsevier.com/locate/jpdc

# 2-edge-Hamilton-connected dragonfly network

Huimei Guo [a], Rong-Xia Hao [a,*], Jie Wu [b,c,*]

[a] School of Mathematics and Statistics, Beijing Jiaotong University, Beijing 100044, China
[b] China Telecom Cloud Computing Research Institute, Beijing 100088, China
[c] Department of Computer and Information Sciences, Temple University, USA

ARTICLE INFO

*Keywords:*
Dragonfly network
2-edge-Hamilton-connected
Paired disjoint cover
1-Hamilton-connected

ABSTRACT

The dragonfly networks are being used in the supercomputers of today. It is of interest to study the topological properties of dragonfly networks. Let $G = (V(G), E(G))$ be a graph. Let $X$ be a subset of $\{uv : u, v \in V(G)$ and $u \neq v\}$ such that every component induced by $X$ on $V(G)$ is a path. If, $|X| \leq k$ and after adding all edges in $X$ to $G$, the resulting graph contains a Hamiltonian cycle that includes all edges in $X$, then the graph $G$ is called $k$-edge-Hamilton-connected. This property can be used to design and optimize routing and forwarding algorithms. By finding such Hamiltonian cycle containing specific edges in the network, it can be ensured that every node can act as an intermediate node to forward packets through a specific channel, thus enabling efficient data transmission and routing. For $k = 2$, determining whether a graph is $k$-edge-Hamilton-connected is a challenging problem, as it is known to be NP-complete. 2-edge-Hamilton-connected is an extension of Hamilton-connected. In this paper, we prove that the relative arrangement dragonfly network, a type of dragonfly network constructed by the global connections based on relative arrangements, is 2-edge-Hamilton-connected, and this property shows that dragonfly networks have strong reliability. In addition, we determined that $D(n, h, g)$ is 1-Hamilton-connected and paired 2-disjoint path coverable with $n \geq 4$ and $h \geq 2$.

## 1. Introduction

One of the important interconnection networks is the dragonfly network, which was first described by Kim et al. [17,18]. The dragonfly network is a high-performance, low-latency interconnect network topology that is suitable for building large-scale distributed systems and data centers. Its design objective is to provide high bandwidth and low latency communication to support large-scale parallel computing and data transfer. The applications of dragonfly networks are primarily focused on the following domains:

- Data Centers: dragonfly networks serve as an interconnect network structure for data centers, connecting servers, storage devices, and network switches to achieve high-performance data center interconnection. Its low latency and high bandwidth characteristics make internal communication within the data center more efficient, facilitating tasks such as large-scale parallel computing, distributed storage, and data analytics.
- Information Technology: dragonfly networks can enhance inter connectivity between servers and improve overall system performance in cloud computing environments. In applications requiring large-scale data processing, such as machine learning and artificial intelligence, dragonfly networks can provide the necessary communication infrastructure for parallel computations.
- Data Transfer and Distribution: dragonfly network's high bandwidth capability makes it advantageous for data transfer and distribution. It can be used to construct large-scale Content Delivery Networks (CDNs) for efficient distribution of network content and media streams. Additionally, dragonfly networks can be applied to large-scale data transfer scenarios, such as transferring large data sets, backing up data, or performing remote data replication.

Hamiltonian-related properties of an interconnection network are crucial in studying parallel and distributed systems. Routing between nodes is one of the significant problems in networks, and the availability and stability of the paths for data broadcasting are essential. These can be achieved by using the Hamiltonian connectivity and fault tolerance properties in the corresponding network topology. In large-scale data-parallel training, a major drawback of general multi-card GPU training is that it requires one GPU at a time to collect the training

* Corresponding authors.
*E-mail addresses:* rxhao@bjtu.edu.cn (R.-X. Hao), jiewu@temple.edu (J. Wu).

https://doi.org/10.1016/j.jpdc.2025.105095
Received 2 September 2024; Received in revised form 15 February 2025; Accepted 17 April 2025
Available online 23 April 2025

gradients from other GPUs and then distribute the new model to the other GPUs. The communication cost grows linearly with the number of GPUs in the system. Recently, Baidu [7] proposed *ring-Allreduce algorithm*, which is an application of Hamiltonian connectivity. It is worth noting that the "ring" mentioned here refers to a Hamiltonian cycle. In contrast, the communication cost of the ring-Allreduce algorithm is constant, independent of the number of GPUs in the system, and is entirely determined by the slowest connection between GPUs in the system. In fact, ring-Allreduce is an optimal communication algorithm if only bandwidth is considered as a factor of communication cost and latency is ignored. Ring-Allreduce is a powerful technique for efficiently aggregating gradients in distributed deep learning systems. Leveraging a ring communication pattern minimizes communication overhead and can significantly improve the training speed and scalability of deep learning models across multiple devices or nodes. For more research on the ring-Allreduce, please refer to [44–46].

*Disjoint paths* are paths that do not share any vertices among them and can be used as parallel paths for specific routes or processing pipelines required for computation. To enhance the Hamiltonian connectivity concept, finding vertex-disjoint paths or embedding linear arrays can support multiple pairs of data transmission in a network, prevent congestion, and offer fault tolerance. There have been many research results in this area that have achieved fruitful progress.

A path that includes all vertices of a graph is known as a *Hamiltonian path*, while a cycle that consists of all vertices is called a *Hamiltonian cycle*. A graph G is considered as *Hamiltonian* if it contains a Hamiltonian cycle and *Hamilton-connected* if there exists a Hamiltonian path connecting any two vertices $x$ and $y$ in $G$. The term *k-Hamilton-connected* refers to a graph $G$ in which the removal of any set $F$ of faulty vertices, where $|F| \leq k$, still results in a Hamilton-connected graph denoted as $G - F$. A *factor* of a graph $G$ is a spanning subgraph of $G$, while a *k-path factor* is a factor composed of $k$ paths whose vertex sets form a partition of $V(G)$.

Kužel et al. [19] delved deeper into the concept of Hamilton-connectivity and provided the following definition.

**Definition 1.** (see [19]). Given a graph $G$, let $E^+(G) = \{uv : u, v \in V(G) \text{ and } u \neq v\}$ and $X \subset E^+(G)$. An element $e \in X$ is referred to as a *potential edge*, which can either be $e \in E(G)$ or $e \notin E(G)$. Define $G + X = (V(G), E(G) \cup X)$. A potential edge set $X$ is considered as a *path system* if every component induced by $X$ forms a path. A graph $G$ is said to be *k-edge-Hamilton-connected*($k$-EHC) if, for any potential edge set $X$ with $|X| \leq k$ such that $X$ determines a path system, the graph $G + X$ contains a Hamiltonian cycle that includes all edges in $X$.

It is worth noting that if a graph $G$ is $(k+1)$-EHC, then it is also $k$-EHC. Kužel et al. [19] observed that a graph $G$ possessing the 1-EHC property is equivalent to being Hamilton-connected. Furthermore, $G$ is 2-EHC if and only if it is both 1-Hamilton-connected and paired 2-disjoint path coverable. Additionally, any $k$-EHC graph must be $(k+2)$-connected. Subsequently, Ozeki et al. [31] demonstrated the NP-completeness of the problem of determining whether a graph is 2-EHC. They also determined the result that every planar graph with a 4-connectivity property is 2-EHC. Recently, Wang et al. [24] proved the 2-EHC of DCell data center network $D(k,n)$. Furthermore, they gave a sufficient condition of 2-EHC in terms of the diameter $m$ of the graph and a parameter $r$, which represents the fault-tolerant Hamiltonian connectivity capability, of $r$-HC. As applications, they obtained some results about 2-EHC of some networks such as 3-ary $n$-cube $Q_n^3$, alternating group graph $AQ_n$, augmented cube $AQ_n$, bcube data center network $BC_{n,k}$, crossed cube $CQ_n$, folded petersen cube $FPQ_{n,k}$, locally twisted cube $LTQ_n$, and varietal hypercube $VQ_n$.

The study of $k$-EHC of a graph is generally related to disjoint path cover. Next, we introduce definition of disjoint path cover of a graph and some of its results on networks.

**Definition 2.** For a given integer $k \geq 1$, let $S = \{s_1, s_2, \ldots, s_k\}$ and $T = \{t_1, t_2, \ldots, t_k\}$ be two arbitrary subsets of vertices in $G$, referred to as *sources* and *sinks*, respectively, with $S \cap T = \emptyset$. A *many-to-many k-disjoint path cover* ($k$-DPC) is a $k$-path factor that collectively connects $S$ and $T$. A graph $G$ is deemed *k-disjoint path coverable* if it possesses a $k$-DPC joining $S$ and $T$.

Research has focused on various applications of $k$-DPC networks, including *paired* and *unpaired* scenarios. In the former, each source $s_i$ is required to connect to a specific sink $t_i$ by a path, while the latter allows each path to connect to an arbitrary pair of source and sink. In a straightforward manner, the $k$-DPC property extends Hamiltonian connectivity from a single path to $k$ disjoint paths, thereby providing applications for multiple pairs of data transmission within a network. In communication networks, disjoint path covering can be used to provide fault tolerance and reliable data transmission. By constructing multiple disjoint paths to connect the source node and the destination node, even if one of the paths fails or becomes interrupted, the data can still be transmitted through alternative paths, ensuring the reliability and connectivity of the communication. Many scholars have conducted in-depth research on paired and unpaired disjoint path coverings. Park et al. obtained some results on paired or unpaired many to many disjoint path covers with faulty elements in references [35,36,39]. Moreover, they also studied paired disjoint path cover of interval graphs [37] and torus-like graphs [38]. Kim et al. [33,34] obtained some conclusions about many to many disjoint path covers in recursive circulant graph $G(2^m, 4)$. There have been some results on the study of disjoint path covers of $k$-ary $n$-cube [32,42], balanced hypercubes [40], hypercubes [41], burnt pancake graphs [43] and Dcell networks [27].

In this paper, we consider whether, given any two channels between switches in the relative arrangement dragonfly network $D(n,h,g)$, which may or may not exist in the original network, there exists a Hamiltonian cycle that must pass through the given two channels. In addition, we show that $D(n,h,g)$ is paired 2-disjoint path coverable and 1-Hamilton-connected. The rest of the paper is organized as follows. Section 2 presents the definition of the dragonfly networks. Section 3 summarizes some research results on dragonfly networks. Section 4 is an introduction to some notations and to the lemmas that will be used throughout the paper. The proof of 1-Hamilton-connected (Subsection 5.1) and paired 2-disjoint path cover (Subsection 5.2) of $D(n,h,g)$ are given in Section 5. The paper is concluded in Section 6.

## 2. Topology descriptions of dragonfly networks

The following symbols are used to introduce the topology of the dragonfly networks.

- $N$: Number of network terminals
- $p$: Number of terminals connected to each router
- $n$: Number of switches in each group
- $h$: Number of channels within each switch used to connect to other groups
- $g$: Number of groups in the system
- $k$: Degree of the switches
- $k'$: Effective degree of the group

As seen in Fig. 1, the dragonfly network is a hierarchical network with three levels: *switch*, *group*, and *system*. Each switch's connections at the bottom level are to *p terminals*, $n-1$ *local channels*, which connect to other switches in the same group, and *h global channels*, which connect to switches in different groups. As a result, the degree of each switch is $k = p + n + h - 1$. An *intra-group interconnection network* made of local channels connects the switches in a group (Fig. 1 (I)). An *inter-group interconnection network* made of global channels connects the groups in a system (Fig. 1 (II)). The switches in a group function as a single *virtual switch* with degree $k' = n(p + h)$, and each group has $np$ connections

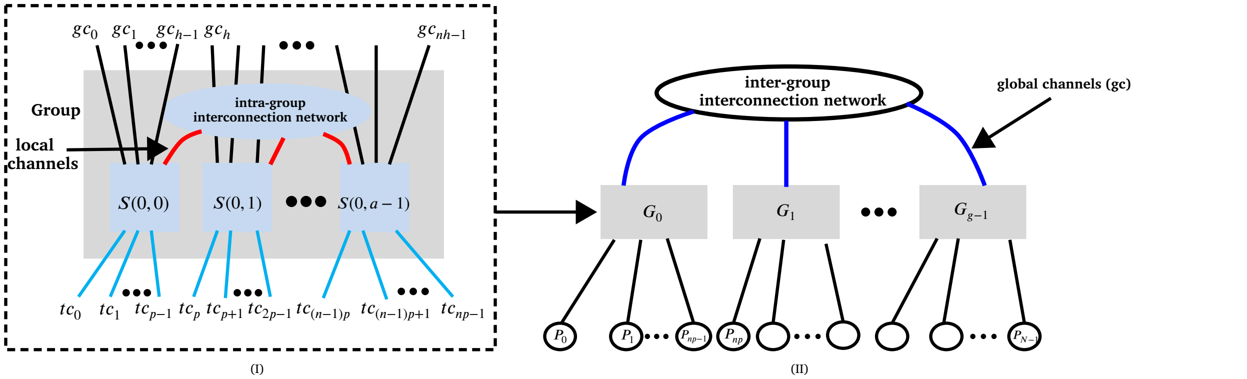


**Fig. 1.** (I) Block diagram of a group (virtual switch) and (II) high-level block diagram showing a topology of dragonfly with several groups. For $q \in [N-1]$, $P_q$ represents to $q$-th terminal; for $i \in \{0,1,\dots,np-1\}$, $tc_i$ represents to channel connected to the terminal; for $j \in \{0,1,\dots,nh-1\}$, $gc_j$ represents to global channel for inter-group connections; for $k \in [n]$ and $\ell \in [g]$, $G_\ell$ represents $\ell$-th group and $S(\ell,k)$ represents the $k$-th switch in $\ell$-th group.

to terminals and $nh$ connections to global channels. With a very low global diameter—the maximum number of costly global channels on the shortest path between any two groups—the system level network (Fig. 1 (II)) can be accomplished thanks to this extremely high degree, $k' \gg k$. A global diameter of one allows for the connection of up to $g = nh+1$ groups ($N = np(nh+1)$ terminals). The intra-group and inter-group networks shown in Fig. 1 can be implemented using arbitrary networks. In this paper, we employ a completely connected topology for both networks. Fig. 2 depicts an example of a dragonfly network with $p = h = 2$, $n = 3$, and a scaling to $N = 42$ with $k = 6$ switches.

In order to study the structure of the dragonfly networks more intuitively, we ignore the terminal links in the logic diagram. A dragonfly network can be viewed as an undirected graph $G$. The switches in its group and the channels between the switches are considered as the vertices and edges of $G$, respectively. An electrical local channel that connects each pair of switches in a group can be considered as *local edge* of $G$. Similarly, an electrical global channel which connects each pair of switches in different groups can be considered as *global edge* of $G$. Let $V(G)$ be vertex set of $G$, $E_1(G)$ and $E_2(G)$ be the two types edge sets of $G$ where $E_1(G)$ is local edge set and $E_2(G)$ is global edge set. Every group has a global edge over every other group thanks to the way these are arranged. The size of a dragonfly network is determined by the number $n$ of switches of per group, the number $h$ of global edges of per switch, and the number $g$ of groups. In the following, we introduce the definition of the dragonfly network, denoted by $DG(n,h,g)$.

**Definition 3.** (see [25]). For any integers $n$, $h$ and $g$ with $n \ge 1$, $1 \le h \le g-1$ and $g = nh+1$, the dragonfly network $G = DG(n,h,g)$ is connected $(n-1+h)$-regular that satisfies the following conditions.

(1) Let $V(G) = V(G_0) \cup V(G_1) \cup \dots \cup V(G_{g-1})$ for $i \in [g]$, where $G_i \cong K_n$ with $V(G_i) = \{(i,k) : k \in [n]\}\}$ is the $i$-th group of $G$ and $(i,k)$ represents switch $k$ in $G_i$. For any two distinct integers $i, j$ with $0 \le i, j \le g-1$, $V(G_i) \cap V(G_j) = \emptyset$.

(2) There is only one global edge between any two groups.

(3)For any vertex $(i,k) \in V(G)$ with $0 \le i \le g-1$ and $1 \le k \le n-1$, there are $h$ vertices which are located in different groups.

According to Definition 3, the number of groups of $DG(n,h,g)$ is $g$. For $0 \le i \le g-1$, let $u$ be a vertex in $G_i$. The set of neighbors of $u$ that belong to different groups than $u$ is called the *external neighbor set* of $u$, denoted by ex$(u)$. For $0 \le j \le g-1$, let $v \in V(G_j)$. Denote ex$(u) \cap$ ex$(v)$ as the *common external neighbor set* of $u$ and $v$, which contains no vertices in $G_i \cup G_j$. We refer to the set of neighbors of $u$ that are within the same group as $u$ as the *internal neighbor set* of $u$. Every group of $DG(n,h,g)$ can be abstracted as a vertex, with two vertices being adjacent if and only if their corresponding groups are connected by a global edge. The abstract graph can be mapped to the complete graph $K_g$. For $0 \le i \le g-1$, let $x_i$ be the vertex abstracted by group $i$. The original concept of the dragonfly

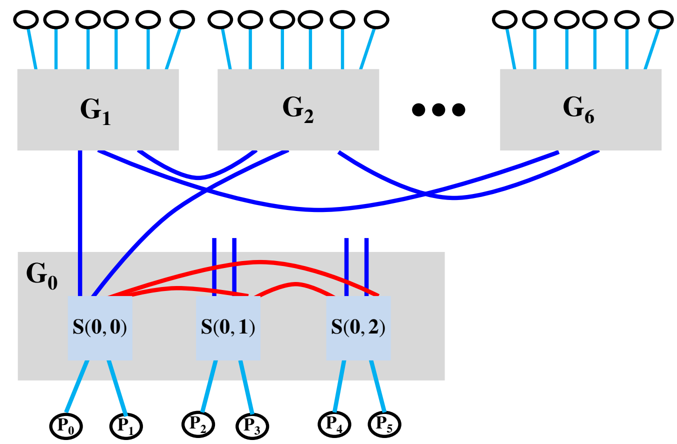


**Fig. 2.** An example block diagram of a dragonfly topology with $p = 2$, $n = 3$ and $h = 2$.

networks did not specifically define the exact connection of the global edges.

Camarero [8] et al. define three arrangements, which are absolute, relative, and circulant-based. The relative arrangement exhibits notable symmetries, including the rotational symmetry, which can be utilized to streamline routing and demonstrate the equivalence of groups [25].

**Definition 4.** (see [22,25]). For any integers $n$, $h$ and $g$ with $n \ge 2$, $h \ge 1$ and $g = nh+1$, the relative arrangement dragonfly network $G = D(n,h,g)$ is defined as follows:

(1) $V(D(n,h,g)) = \{(x,y) : 0 \le x \le g-1, 0 \le y \le n-1\}$ where $x$ represents group $x$ and $y$ represents switch $y$.

(2) For any two vertices $u = (x_1, y_1)$ and $v = (x_2, y_2)$ in $D(n,h,g)$, $(u,v) \in E(D(n,h,g))$ if and only if both of the following conditions are holds:

- $y_2 = n - 1 - y_1$;
- $x_2 = (hy_1 + x_1 + k) \bmod g$ for $k = 1, 2, \dots, h$.

Wu et al. [26]demonstrated the existence of a Hamiltonian cycle in the Dragonfly network $D(n,h,g)$ and propose an $O(g)$ algorithm for constructing a Hamiltonian cycle in $D(n,h,g)$ when $n \ge 2$ and $h \ge 1$. Subsequently, they established that $D(n,h,g)$ is Hamilton-connected for $n \ge 4$ and $h \ge 2$. In reference [25], they also provided a proof that the connectivity of $D(n,h,g)$ is $n-1+h$. Furthermore, they proposed an $O(n)$ algorithm to compute the disjoint path between any two distinct vertices in $D(n,h,g)$ and analyzed the maximum length of these disjoint paths, which does not exceed 7. The Fig. 3 shows the relative arrangement dragonfly network $D(3,2,7)$.

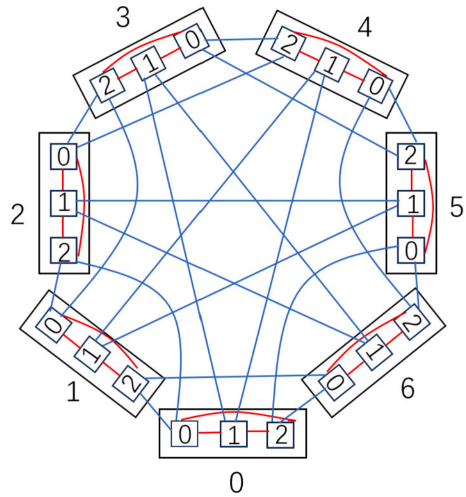


**Fig. 3.** Logical structure of the relative arrangement dragonfly network with $n = 3$, $h = 2$ and $g = 7$.

## 3. Related work

The topology structure of the dragonfly network has the following characteristics: low latency, high bandwidth, path diversity, scalability, flexibility, and so on. Dragonfly networks have been implemented in the current generation of supercomputers, including the Frontier supercomputer [10], Cray XC [1,9], and PERCS [2].

According to Hasting et al. [15], the global link arrangements frequently result in bisection bandwidths that differ by 10 s of percent, and the specific separation varies depending on the relative bandwidths of local and global links. Moreover, the regularity of task mappings for nearest neighbor stencil communication patterns can be significantly impacted by the choice of global link arrangement. Belka et al. [6] presented two novel configurations for global links, called nautilus arrangement and helix arrangement, each of which is based on the idea of optimizing bisection bandwidth when global links have a higher bandwidth than local links. In spite of this, the new arrangements perform better across the board for all bandwidth relationships than previously known arrangements.

Jiang et al. [16] proposed indirect global adaptive routing (IAR), in which adaptive routing decisions use information that cannot be directly obtained by the source router. They introduced four IAR routing methods: credit round-trip (CRT), progressive adaptive routing (PAR), piggyback routing(PB), and reservation routing (RES). Yébenes et al. [30] provided hierarchical two-level queuing, a queuing technique specifically created to minimize HoL blocking in minimal-path routing fully-connected dragonfly networks. Prisacari et al. [20] shown how simple changes to either the routing strategy or the process to node assignment can bring performance back close to ideal levels. Bhatele [4] et al. explored the intelligent topological perception mapping of different communication modes and physical topology to determine the situation of minimizing link utilization. They also analyzed the trade-offs between direct and indirect routes using different maps. Moreover, to gain a better understanding of inter-job interference, Bhatele et al. [5] investigated the impacts of job location, parallel workloads, and network topologies on network health. To simulate the network behavior of Cray Cascade, they created a functional network simulator called Damselfly, and a visual analytics tool called DragonView.

In reference [11], Faizian et al. suggested improving UGAL by adding an in-group communication adaptation mechanism based on traffic mode to Dragonfly. Rahman et al. [21] made studies on improving the performance of traditional UGAL. Chaulagain et al. [23] had made in-depth researches on improving the accuracy of UGAL delay estimation. García et al. in references [12–14] presented a unique flow-control/routing strategy that separates the deadlock avoidance mechanisms from the routing. Their concept allows for on-the-fly (in-transit) adaptive packet routing since it does not impose any dependencies across virtual channels. They use an injection-restricted deadlock-free escape subnetwork to avoid deadlock. Compared to other approaches that have been offered, this model reduces latency, boosts throughput, and adjusts to transient traffic more quickly.

Xiang et al. [28] put forth two distinct broadcast schemes, namely the group-first and router-first. It is demonstrated that unicast-based broadcast schemes are essential to circumvent deadlocks at consumption channels. The group-first broadcast scheme expedites the delivery of a message to all groups, whereas the router-first scheme minimizes the number of unicast steps traversing global links. The method in reference [28] shows the inaugural collective communication work for dragonfly networks. Moreover, Xiang et al. [29] proposed a new deadlock-free adaptive fault-tolerant routing algorithm, which maps the router into a group based on the new double-layer security information model and maps the group of the dragonfly network into two independent hypercubes. The new fault-tolerant routing algorithm can tolerate both static and dynamic faults. Full simulation results show that the proposed fault-tolerant routing algorithm is even better than the previous fault-free network minimum routing algorithm in many cases.

## 4. Preliminary

For a positive integer $n$, let $[n] = \{0, 1, \ldots, n-1\}$. Let $G = (V(G), E(G))$ be an undirected graph with vertex set $V(G)$ and edge set $E(G)$. An interconnected network can be considered as an undirected graph such that its processors are viewed as vertices of the graph and links between processors are viewed as edges of the graph. If there is only one edge connecting any two vertices in graph $G$ and there is not a loop at any vertex, the graph is considered simple. The graphs studied in this paper are simple graphs. For $u, v \in V(G)$, $(u, v)$ represents the edge between vertices $u$ and $v$. Two graphs $G_1$ and $G_2$ are called isomorphic, denoted as $G_1 \cong G_2$, if there exist a bisection $\psi : V(G_1) \to V(G_2)$ such that $(x, y) \in E(G_1)$ if and only if $(\psi(x), \psi(y)) \in E(G_2)$. A complete graph, say $K_n$, is the graph with order $n$ in which every pair of different vertices is adjacent. The union graph of $G_1$ and $G_2$, denoted by $G_1 \cup G_2$, is a subgraph of $G$, with vertices set $V(G_1) \cup V(G_2)$ and edges set $E(G_1) \cup E(G_2)$. The set of vertices that are adjacent to vertex $v$ is called the neighborhood of $v$, denoted by $N_G(v)$, i.e. $N_G(v) = \{u : (u, v) \in E(G)\}$. The degree of $v$ is the number of edges incident with $v$, denoted by $\deg_G(v)$. Let $F$ be a nonempty proper subset of $V(G)$. Denote $G - F$ as a graph obtained from $G$ by removing all of the vertices and incident edges in $F$. Let $G[F]$ be the subgraph of $G$ induced by $F$. Also, sometimes we use $[F]$ instead of $G[F]$ if the graph is clear from the context. We call $H$ to be a subgraph of $G$, say $H \subseteq G$, if $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$. We use $G - H$ to denote $G[V(G) \setminus V(H)]$. For flexible use of notations, $G - H$ and $G - V(H)$ represent the same meaning and can be used interchangeably.

A path joining two vertices $u$ and $v$ in $G$ is the union of edges, that is

$$P(u, v) = \langle (u_1, u_2), (u_2, u_3), \ldots, (u_{\ell-1}, u_\ell) \rangle,$$

where $u = u_1$, $v = u_\ell$ and $(u_i, u_{i+1}) \in E(G)$ with $1 \le i \le \ell - 1$. Note that all vertices in $P(u, v)$ are distinct. Also, sometimes $P(u, v)$ and $P(v, u)$ represent the same meaning in this paper. We call $u$ and $v$ the end vertices of path $P(u, v)$. For simplicity of notation, we use $(u, v)$-path to represent a path $P(u, v)$. If a path with end vertices $u$ and $v$ covers all vertices of $G$, then we denote HP$(u, v)$ as the path. For $t < \ell$, if $P_1(u_1, u_t) = \langle (u_1, u_2), (u_2, u_3), \ldots, (u_{t-1}, u_t) \rangle$ and $P_2(u_t, u_\ell) = \langle (u_t, u_{t+1}), (u_{t+1}, u_{t+2}), \ldots, (u_{\ell-1}, u_\ell) \rangle$ are two paths in $G$, then we denote $P(u, v) = P_1(u_1, u_t) + P_2(u_t, u_\ell)$. A cycle $C$ is a closed path, that is $C = P(u, v) + \langle (v, u) \rangle$. For $1 \le i \le \ell - 1$, let $(u_i, u_{i+1})$ be an edge in $P(u, v)$. After removing $(u_i, u_{i+1})$ from $P(u, v)$, the result subgraphs can be denoted by $P(u, v) - (u_i, u_{i+1})$. A graph $G$ is *covered* by a path $P(u, v)$, which means that $P(u,v)$ is a Hamiltonian path of $G$. For notations not explicitly explained here, we refer to [3].

**Lemma 4.1.** (see [26]). *For $n \geq 4$ and $h \geq 2$, $D(n,h,g)$ is Hamilton-connected.*

**Lemma 4.2.** (see [19]). *Let $G$ be a graph. Then $G$ is 2-edge-Hamilton-connected if and only if*

(1) *$G$ is 1-Hamilton-connected, and*

(2) *for any two distinct vertex set $S=\{s_1,s_2\}, T=\{t_1,t_2\}$ with $S,T \subseteq V(G)$, $G$ has a path factor consisting of two paths $P_1, P_2$ such that both $P_1$ and $P_2$ have one end vertex in $S$ and the other one end vertex in $T$.*

Statement (2) of Lemma 4.2 can be converted to that $G$ is paired 2-disjoint path coverable.

**Lemma 4.3.** *For $n \geq 4$ and $h \geq 2$, any two vertices in $D(n,h,g)$ have at most $h-1$ common external neighbors.*

**Proof.** For $a_1,a_2 \in [g]$ and $b_1,b_2 \in [n]$, let $u=(a_1,b_1)$ and $v=(a_2,b_2)$ be two vertices which are located in group $G_{a_1}$ and group $G_{a_2}$, respectively. If $a_1=a_2$, by Definition 4, then $\mathrm{ex}(u) \cap \mathrm{ex}(v)=\emptyset$. The lemma holds.

If $a_1 \neq a_2$, without loss of generality, let $a_1<a_2$. For an integer $t \in [g]\setminus\{0\}$, let $a_2=a_1+t$. Suppose that $u$ and $v$ have common external neighbors. By Definition 4, one has that $b_1=b_2$ and $(hb_1+a_1+k_1) \equiv (hb_2+a_2+k_2) \pmod g$ with $1 \leq k_1,k_2 \leq h$. Let $r_j=hb_j+a_j+k_j$ with $j \in \{1,2\}$. We consider the following three situations to prove that $|\mathrm{ex}(u) \cap \mathrm{ex}(v)| \leq h-1$.

**Case 1.** $r_1,r_2<g$.

As $r_1 \equiv r_2 \pmod g$, we have $r_1=r_2$, which implies that

$$hb_1+a_1+k_1=hb_2+a_2+k_2.$$

Combining $b_1=b_2$ and $a_2=a_1+t$, one has that $k_1=t+k_2$. Since $1 \leq k_1,k_2 \leq h$ and $1 \leq t \leq g-1$, we can obtain that $1 \leq k_2=k_1-t \leq h-t \leq h-1$. Thus, if $u$ and $v$ have common external neighbors, then $1 \leq |\mathrm{ex}(u) \cap \mathrm{ex}(v)| \leq h-1$.

**Case 2.** $r_1<g<r_2$.

For an integer $\ell \geq 1$, since $r_1 \equiv r_2 \pmod g$, we have $r_2=\ell g+r_1$, which implies that

$$hb_1+a_2+k_2=\ell g+hb_1+a_1+k_1.$$

Combining $a_2=a_1+t$ and $b_1=b_2$, one has that $t+k_2=\ell g+k_1$. If $\ell \geq 2$, then

$$k_2=\ell g+k_1-t \geq 2g+k_1-t \geq 2g+1-(g-1)=g+2,$$

which contradicts with $k_2 \leq h \leq g-1$. Thus $\ell \leq 1$. Since $\ell \geq 1$, we have $\ell=1$. Hence, $t+k_2=g+k_1$. By $1 \leq t \leq g-1$ and $1 \leq k_1,k_2 \leq h \leq g-1$, one has that $2 \leq k_2 \leq h$. Thus, if $u$ and $v$ have common external neighbors, then $1 \leq |\mathrm{ex}(u) \cap \mathrm{ex}(v)| \leq h-1$.

**Case 3.** $g<r_1<r_2$. Let $\ell_1$, $\ell_2$ and $s$ be three integers with $s \leq g-1$. Since $r_1,r_2>g$ and $r_1 \equiv r_2 \pmod g$, let $r_1=\ell_1 g+s$ and $r_2=\ell_2 g+s$. We have $s=hb_1+a_1+k_1-\ell_1 g$. Combining $a_2=a_1+t$, one has that

$$t+k_2=(\ell_2-\ell_1)g+k_1.$$

If $\ell_2-\ell_1 \geq 2$, then $k_2 \geq 2g+k_1-t \geq g+1$ since $1 \leq t \leq g-1$ and $1 \leq k_1 \leq h$, which contradicts with $1 \leq k_2 \leq h \leq g-1$. Thus $0 \leq \ell_2-\ell_1 \leq 1$. If $\ell_2-\ell_1=0$, one has that $1 \leq k_2=k_1-t \leq h-1$. If $\ell_2-\ell_1=1$, one has that $2 \leq k_2=g+k_1-t \leq h$. Thus, if $u$ and $v$ have common external neighbors, then $1 \leq |\mathrm{ex}(u) \cap \mathrm{ex}(v)| \leq h-1$. □

## 5. Main results

This section proves that the relative arrangement dragonfly network $D(n,h,g)$ is 2-edge-Hamilton-connected. That is Theorem 1.

**Theorem 1.** *For $n \geq 4$ and $h \geq 2$, the relative arrangement dragonfly network $D(n,h,g)$ is 2-edge-Hamilton-connected.*

According to Lemma 4.2, proving Theorem 1 requires only demonstrating that $D(n,h,g)$ is 1-Hamilton-connected and has a paired 2-disjoint path cover. Theorems 2 and 3 are proved in the following two parts, respectively. Combining Lemma 4.2, Theorems 2 and 3, Theorem 1 is gotten directly.

### 5.1. 1-Hamilton-connected of $D(n,h,g)$

In this part, we consider 1-Hamilton-connected of $D(n,h,g)$. For each $i \in [g]$ and $k \in [n]$, let $G_i$ be the $i$-th group of $G=D(n,h,g)$ and $(i,k)$ represents switch $k$ in $G_i$. Then $V(G)=V(G_0)\cup V(G_1)\cup \cdots \cup V(G_{g-1})$ with $V(G_i)=\{(i,k): k \in \{0,1,\ldots,n-1\}\}$. Let $x_i$ be the vertex obtained by contracting group $G_i$ to a vertex and let $G_c$ be the graph obtained by contracting each group in $G$ to a vertex. Obviously, $G_c \cong K_g$ and $V(G_c)=\{x_i : 0 \leq i \leq g-1\}$.

**Theorem 2.** *For $n \geq 4$ and $h \geq 2$, $D(n,h,g)$ is 1-Hamilton-connected.*

**Proof.** We will prove that the graph $G=D(n,h,g)$ remains Hamilton-connected after arbitrarily removing a vertex. Without loss of generality, let the faulty vertex $w=(0,0) \in V(G_0)$. By Definition 4, one has that each of $G_1, G_2, \cdots,$ and $G_h$ contains only one external neighbor of $w$. Let $u$ and $v$ be any two distinct vertices in $G-w$. We classify the cases according to the location of $u$ and $v$.

**Case 1.** $u,v \in V(G_0)$.

Since $G_0 \cong K_n$ with $n \geq 4$, one has that $G_0-w$ is Hamilton-connected. Let $P_0(u,v)$ be a Hamiltonian path of $G_0-w$ connecting $u$ and $v$. Let $z_1,z_2 \in V(P_0(u,v))$ and $(z_1,z_2) \in E(P_0(u,v))$. Let $P_0(u,z_1)$ and $P_0(z_2,v)$ be two paths obtained by removing $(z_1,z_2)$ from $P_0(u,v)$, that is $P_0(u,v)-(z_1,z_2)=P_0(u,z_1)+P_0(z_2,v)$. Let $a$ and $b$ be some external neighbor of $z_1$ and $z_2$, respectively, where $a \in V(G_i)$ and $b \in V(G_j)$ with $i \neq j$ and $i,j \in [g]\setminus[h+1]$. Recalling that $G_c$ is the graph obtained by contracting each group in $G$ to a vertex, $G_c \cong K_g$ and $V(G_c)=\{x_i : 0 \leq i \leq g-1\}$. Then $G_c-x_0$ is Hamilton-connected. One has that there exists a Hamiltonian path, say $P(x_i,x_j)$, between $x_i$ and $x_j$ in $G_c-x_0$. Combining the fact that $G_i$ is a complete graph for each $i \in [g]$, we can expand $P(x_i,x_j)$ into a Hamiltonian path of $G-G_0$, denoted by $P(a,b)$, between $a$ and $b$. Thus

$$P_0(u,z_1)+(z_1,a)+P(a,b)+(b,z_2)+P_0(z_2,v)$$

is a Hamiltonian path of $G-w$ connecting $u$ and $v$. Fig. 4 (a) shows the specific situation of Case 1.

**Case 2.** $|\{u,v\} \cap V(G_0)|=1$.

Without loss of generality, let $u \in V(G_0)$ and $v \in V(G_i)$ for $i \in [g]\setminus\{0\}$. Let $z$ be a vertex in $G_0$ such that external neighbors of $z$ are not located in $G_i$. Let $c$ be a vertex in $V(G_j)\cap \mathrm{ex}(z)$ with $j \neq i$ and $j \in [g]\setminus[h+1]$. By similar analysis of Case 1, $G-G_0$ has a Hamiltonian path, say $P(c,v)$, connecting $c$ and $v$, and $G_0-w$ has a Hamiltonian path, say $P_0(u,z)$, connecting $u$ and $z$. Thus

$$P_0(u,z)+(z,c)+P(c,v)$$

is a Hamiltonian path of $G-w$ connecting $u$ and $v$. Fig. 4 (b) shows the specific situation of Case 2.

**Case 3.** $u,v \in V(G_i)$ for $i \in [g]\setminus\{0\}$.

Since $u$ and $v$ are located in $G_i$, one has that $G_i$ contains a Hamiltonian path, say $P_i(u,v)$, connecting $u$ and $v$. For $n \geq 4$, let $c$ and $d$ be two vertices in $V(P_i(u,v))$ such that $(u,c),(c,d) \in E(P_i(u,v))$. By Definition 4, one has that $|(\mathrm{ex}(c)\cup \mathrm{ex}(d))\cap V(G_0)| \leq 1$. Without loss of generality, let $\mathrm{ex}(d)\cap V(G_0)=\emptyset$. Note that $P_i(u,v)-(c,d)=P_i(u,c)+P_i(d,v)$. For $j,k \in [g]\setminus\{0,i\}$, let $c'(\in V(G_j))$ and $d'(\in V(G_k))$ be an external neighbor of $c$ and $d$, respectively. Let $a$ and $b$ be two vertices in $G_0-w$ such that $(\mathrm{ex}(a)\cup \mathrm{ex}(b))\cap V(G_i)=\emptyset$. As $G_0-w$ is a graph isomorphic to

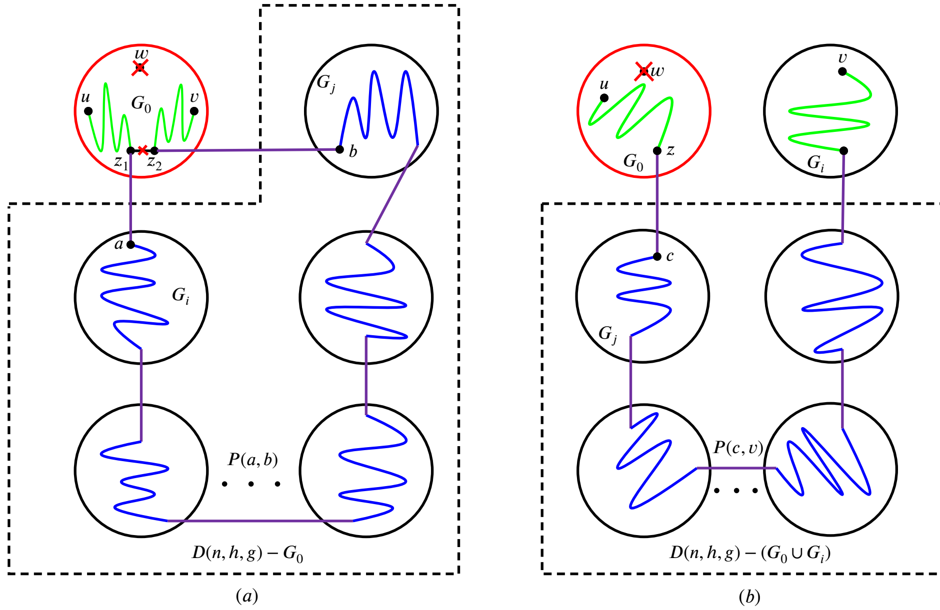


**Fig. 4.** An illustration of Theorem 2: (a) Case 1; (b) Case 2.

$K_{n-1}$, then $G_0 - w$ contains a Hamiltonian path, say $P_0(a,b)$, connecting $a$ and $b$. Next, we consider the cases of the location of the external neighbors of $a$ and $b$ to prove that $G$ contains a Hamiltonian path connecting $u$ and $v$.

**Case 3.1.** $|\text{ex}(a) \cap V(G_j \cup G_k)| = 2$ or $|\text{ex}(b) \cap V(G_j \cup G_k)| = 2$.

Without loss of generality, let $|\text{ex}(a) \cap V(G_j \cup G_k)| = 2$. Let $a'$ and $a''$ be the external neighbor of $a$ in $G_j$ and $G_k$, respectively. By Definition 4 and $h \geq 2$, one has that $\text{ex}(b) \cap V(G_j \cup G_k) = \emptyset$ and $d$ has an external neighbor, say $d''$, in a group $G_\ell$ with $\ell \in [g] \setminus \{0, i, j, k\}$. Note that $|\text{ex}(b) \cap V(G_\ell)| \leq 1$. Combining $h \geq 2$ and $\text{ex}(b) \cap V(G_i \cup G_j \cup G_k) = \emptyset$, one has that $b$ has an external neighbor, say $b'$, in group $G_m$ with $m \in [g] \setminus \{0, i, j, k, \ell\}$. Recalling that every group of $G$ and the graph obtained from $G$ by contracting each group of $G$ into a vertex are complete graphs. Then we can find a path, say $P(c', a'')$, connecting $c'$ and $a''$ and put the groups covered by this path into the subgraph set $H_1$ with $H_1 \subset \{G_t : t \in [g] \setminus \{0, i, \ell, m\}\}$. Similarly, we also can find a path, say $P(b', d'')$, connecting $b'$ and $d''$ and put the groups covered by this path into the subgraph set $H_2$ with $H_2 = G - H_1 - \{G_0, G_i\}$. Thus, $G - w$ has a Hamiltonian path connecting $u$ and $v$ as follows:

$$P_i(u,c) + (c,c') + P(c',a'') + (a'',a) + P_0(a,b) + (b,b') + P(b',d'') + (d'',d) + P_i(d,v).$$

Fig. 5 (c) shows the specific situation of Case 3.1.

**Case 3.2.** $|\text{ex}(a) \cap V(G_j \cup G_k)| = 1$ and $|\text{ex}(b) \cap V(G_j \cup G_k)| = 1$.

Without loss of generality, let $|\text{ex}(a) \cap V(G_j)| = 1$ and $|\text{ex}(b) \cap V(G_k)| = 1$. By Definition 4 and $h \geq 2$, $a$ contains an external neighbor, say $a'$, in group $G_\ell$ with $\ell \in [g] \setminus \{0, i, j, k\}$ and $b$ contains an external neighbor, say $b'$, in group $G_m$ with $m \in [g] \setminus \{0, i, j, k, \ell\}$. We can find a path, say $P(c', a')$, connecting $c'$ and $a'$ and put the groups covered by this path into the subgraph set $H_1$ with $H_1 \subset \{G_t : t \in [g] \setminus \{0, i, \ell, m\}\}$. Similarly, we also can find a path, say $P(b', d')$, connecting $b'$ and $d'$ and put the groups covered by this path into the subgraph set $H_2$ with $H_2 = G - H_1 - \{G_0, G_i\}$. Thus, $G - w$ has a Hamiltonian path connecting $u$ and $v$ as follows:

$$P_i(u,c) + (c,c') + P(c',a') + (a',a) + P_0(a,b) + (b,b') + P(b',d') + (d',d) + P_i(d,v).$$

Fig. 5 (d) shows the specific situation of Case 3.2.

**Case 3.3.** $|\text{ex}(a) \cap V(G_j \cup G_k)| = 1$ and $|\text{ex}(b) \cap V(G_j \cup G_k)| = 0$ (resp. $|\text{ex}(a) \cap V(G_j \cup G_k)| = 0$ and $|\text{ex}(b) \cap V(G_j \cup G_k)| = 1$).

We consider $|\text{ex}(a) \cap V(G_j \cup G_k)| = 1$ and $|\text{ex}(b) \cap V(G_j \cup G_k)| = 0$. Without loss of generality, let $|\text{ex}(a) \cap V(G_j)| = 1$. For $h \geq 2$ and Definition 4, one has that $a$ contains an external neighbor, say $a'$, in group $G_\ell$ with $\ell \in [g] \setminus \{0, i, j, k\}$. As $|\text{ex}(b) \cap V(G_j \cup G_k)| = 0$, then $b$ has an external neighbor, say $b'$, in group $G_m$ with $m \in [g] \setminus \{0, i, j, k, \ell\}$. Similar to Case 3.2, we can find a Hamiltonian path of $G - w$ connecting $u$ and $v$ as follows:

$$P_i(u,c) + (c,c') + P(c',a') + (a',a) + P_0(a,b) + (b,b') + P(b',d') + (d',d) + P_i(d,v).$$

**Case 3.4.** $|\text{ex}(a) \cap V(G_j \cup G_k)| = 0$ and $|\text{ex}(b) \cap V(G_j \cup G_k)| = 0$.

By Definition 4, $a$ contains an external neighbor, say $a'$, in group $G_\ell$ with $\ell \in [g] \setminus \{0, i, j, k\}$ and $b$ contains an external neighbor, say $b'$, in group $G_m$ with $m \in [g] \setminus \{0, i, j, k, \ell\}$. Similar to Case 3.2, we can find a Hamiltonian path of $G - w$ connecting $u$ and $v$ as follows:

$$P_i(u,c) + (c,c') + P(c',a') + (a',a) + P_0(a,b) + (b,b') + P(b',d') + (d',d) + P_i(d,v).$$

**Case 4.** $u \in V(G_i)$ and $v \in V(G_j)$ for $i \neq j$ and $i, j \in [g] \setminus \{0\}$.

For $n \geq 4$, let $a$ be a vertex in $G_0 - w$ such that $\text{ex}(a) \cap V(G_i \cup G_j) = \emptyset$. Let $a'$ be an external neighbor of $a$ in group $G_k$ with $k \in [g] \setminus \{0, i, j\}$. Let $b$ be a vertex in $G - \{w, a\}$ such that $\text{ex}(b) \cap V(G_i) = \emptyset$. For $h \geq 2$, $b$ has an external neighbor, say $b'$, in group $G_\ell$ with $\ell \in [g] \setminus \{0, i, j, k\}$. Recalling that every group of $G$ and the graph obtained from $G$ by contracting each group of $G$ into a vertex are complete graphs. Then we can find a path, say $P(u, a')$, connecting $u$ and $a'$ and put the groups covered by this path into the subgraph set $H_1$ with $H_1 \subset \{G_t : t \in [g] \setminus \{0, \ell, j\}\}$. Similarly, we also can find a path, say $P(b', v)$, connecting $b'$ and $v$ and put the groups covered by this path into the subgraph set $H_2$ with $H_2 = G - H_1 - \{G_0\}$. Thus, $G - w$ has a Hamiltonian path connecting $u$ and $v$ as follows:

$$P(u,a') + (a',a) + P_0(a,b) + (b,b') + P(b',v).$$

Fig. 5 (e) shows the specific situation of Case 4.

In conclusion, if $n \geq 4$ and $h \geq 2$, the relative arrangement dragonfly network $D(n,h,g)$ is Hamilton-connected after removing an arbitrary vertex. □

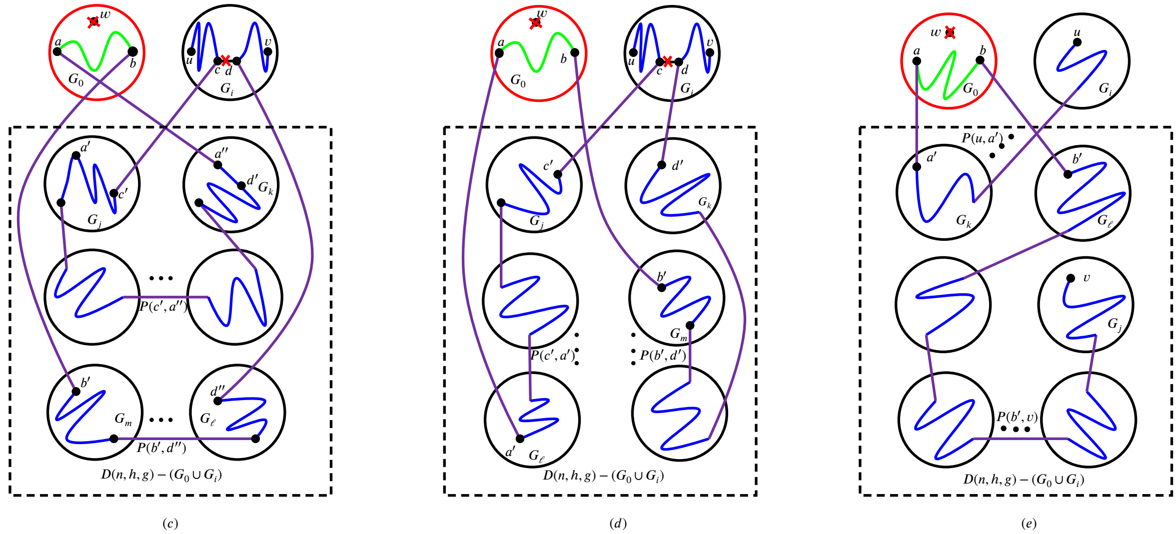


**Fig. 5.** An illustration of Theorem 2: (c) Case 3.1; (d) Case 3.2; (e) Case 4.

### 5.2. Paired 2-DPC of $D(n,h,q)$

**Theorem 3.** *For $n \geq 4$, $h \geq 2$ and any two disjoint vertices set $S$ and $T$ with $S = \{s_1, s_2\}$ and $T = \{t_1, t_2\}$, $D(n,h,g)$ has a paired 2-DPC joining $S$ and $T$.*

**Proof.** Let $G = D(n,h,g)$. Recalling that $V(G) = V(G_0) \cup V(G_1) \cup \cdots \cup V(G_{g-1})$, where $G_i$ with $G_i \cong K_n$ and $V(G_i) = \{(i,k) : k \in \{0,1,\ldots,n-1\}\}$ is the $i$-th group of $G = D(n,h,g)$ and $(i,k)$ represents the switch $k$ in $G_i$. The graph $G_c$, obtained from $G$ by contracting each group $G_i$ to a vertex $x_i$, is isomorphic to $K_g$ with vertex set $\{x_i : 0 \leq i \leq g-1\}$.

Since the 2-DPC is paired, the following two types will be considered according to the end vertex assignments. For the first type, we will find a 2-DPC, say $\mathcal{P}_1 = \{P_1(s_1,t_1), P_2(s_2,t_2)\}$, where $P_i(s_i,t_i)$ is a path with end vertices $s_i$ and $t_i$ for $i \in \{1,2\}$. For the second type, we will find a 2-DPC, say $\mathcal{P}_2 = \{P_1(s_1,t_2), P_2(s_2,t_1)\}$, where $P_i(s_i,t_j)$ is a path with end vertices $s_i$ and $t_j$ for $i \neq j$ and $i,j \in \{1,2\}$. Since the discussions for these two types are similar, we only consider the first type here.

**Case 1.** $S$ and $T$ are located in a group.

For $i \in [g]$, let $(S \cup T) \subseteq V(G_i)$. Since $G_i$ is a complete graph, $G_i$ has a 2-DPC joining $S$ and $T$, say

$$P_1(s_1,t_1) = \langle (s_1,r_1),(r_1,r_2),\ldots,(r_a,t_1)\rangle$$

and

$$P_2(s_2,t_2) = \langle (s_2,z_1),(z_1,z_2),\ldots,(z_b,t_2)\rangle,$$

where $r_q, z_m \in V(G_i)$ with $0 \leq q \leq a+1$, $0 \leq m \leq b+1$ and $a+b = n-4$, especially, $s_1 = r_0$, $t_1 = r_{a+1}$, $s_2 = z_0$, and $t_2 = z_{b+1}$. For $0 \leq q \leq a$, we choose an edge $(r_q, r_{q+1})$ from $P(s_1,t_1)$. Let $r'_q \in V(G_j)$ be an external neighbor of $r_q$ and $r'_{q+1} \in V(G_k)$ be an external neighbor of $r_{q+1}$ with distinct $j,k \in [g] \setminus \{i\}$. Since every group of $G$ and the graph obtained by contracting each group of $G$ into a vertex are complete graphs, we can find a path, say $P(r'_q, r'_{q+1})$, connecting $r'_q$ and $r'_{q+1}$. We then put the groups covered by this path into the subgraph set $H_1$ with $H_1 = G - G_i$. Thus, $G$ has a paired 2-DPC joining $S$ and $T$ as follows:

$$P_1(s_1,t_1) = P(s_1,t_1) - (r_q,r_{q+1}) + (r_q,r'_q) + P(r'_q,r'_{q+1}) + (r'_{q+1},r_{q+1})$$

and

$$P_2(s_2,t_2) = \langle (s_2,z_1),(z_1,z_2),\ldots,(z_b,t_2)\rangle.$$

Fig. 6 (a) shows the specific situation of Case 1.

**Case 2.** $S$ and $T$ are located in two different groups.

**Case 2.1.** For distinct $i,j \in [g]$, $S \subseteq V(G_i)$ and $T \subseteq V(G_j)$.

Let $a_1$ and $b_1$ be any two vertices in $V(G_i)$. Since $G_i \cong K_n$, we can find a paired 2-DPC of $G_i$ joining $\{s_1,t_1\}$ and $\{a_1,b_1\}$. Denote $P(s_1,a_1)$ and $P(t_1,b_1)$ as the paired 2-DPC of $G_i$. Let $a_2$ and $b_2$ be any two vertices in $V(G_j)$ such that $(a_2,a_1) \notin E(G)$ and $(b_2,b_1) \notin E(G)$. Since $G_j \cong K_n$, we can find a paired 2-DPC of $G_j$ joining $\{s_2,t_2\}$ and $\{a_2,b_2\}$. Denote $P(s_2,a_2)$ and $P(t_2,b_2)$ as the paired 2-DPC of $G_j$. By Lemma 4.3 and $h \geq 2$, $a_1$ and $a_2$ have at most $h-1$ external common neighbors, which implies that $a_1$ has an external neighbor, say $a'_1$, and $a_2$ has an external neighbor, say $a'_2$, such that $a'_1 \neq a'_2$. Since every group of $G$ and the graph obtained by contracting each group of $G$ into a vertex are complete graphs, we can find a path, say $P(a'_1,a'_2)$, connecting $a'_1$ and $a'_2$. Furthermore, we put the groups covered by this path into the subgraph set $H_1$ with $H_1 \subset \{G_t : t \in [g] \setminus \{i,j\}\}$ and $H_1 \neq G - G_i \cup G_j$. Similarly, by Lemma 4.3 and $h \geq 2$, $b_1$ and $b_2$ have at most $h-1$ external common neighbors, which implies that $b_1$ has an external neighbor, say $b'_1$, and $b_2$ has an external neighbor, say $b'_2$, such that $b'_1 \neq b'_2$. We can find a path, say $P(b'_1,b'_2)$, connecting $b'_1$ and $b'_2$ and put the groups covered by this path into the subgraph set $H_2$ with $H_2 = G - H_1 - G_i \cup G_j$. Thus, $G$ has a paired 2-DPC joining $\{s_1,s_2\}$ and $\{t_1,t_2\}$ as follows:

$$P_1(s_1,t_1) = P(s_1,a_1) + (a_1,a'_1) + P(a'_1,a'_2) + (a'_2,a_2) + P(a_2,t_1)$$

and

$$P_2(s_2,t_2) = P(s_2,b_1) + (b_1,b'_1) + P(b'_1,b'_2) + (b'_2,b_2) + P(b_2,t_2).$$

Fig. 6 (b) shows the specific situation of Case 2.1.

**Case 2.2.** For distinct $i,j \in [g]$, $|S \cap V(G_i)| = |S \cap V(G_j)| = 1$ and $|T \cap V(G_i)| = |T \cap V(G_j)| = 1$.

In this case, without loss of generality, let $s_1,t_1 \in V(G_i)$ and $s_2,t_2 \in V(G_j)$. Since $G_i, G_j \cong K_n$, one has that $G_i$ has a Hamiltonian path, say $P(s_1,t_1)$, connecting $s_1$ and $t_1$, and $G_j$ has a Hamiltonian path, say $P(s_2,t_2)$, connecting $s_2$ and $t_2$. We can choose an edge, say $(z_1,z_2)$, in $P(s_1,t_1)$. Let $z'_1 \in V(G_k)$ be an external neighbor of $z_1$ with $k \neq j$ and $z'_2 (\in V(G_\ell))$ be an external neighbor of $z_2$ with $\ell \neq j$. Then we can find a path, denoted by $P(z'_1,z'_2)$, connecting $z'_1$ and $z'_2$ and put the groups covered by this path into the subgraph set $H$ with $H = G - G_i \cup G_j$. Thus, $G$ has a paired 2-DPC joining $\{s_1,s_2\}$ and $\{t_1,t_2\}$ as follows:

$$P_1(s_1,t_1) = P(s_1,t_1) - (z_1,z_2) + (z_1,z'_1) + P(z'_1,z'_2) + (z'_2,z_2)$$

and

$$P_2(s_2,t_2) = P(s_2,t_2).$$

Fig. 7 (c) shows the specific situation of Case 2.2.

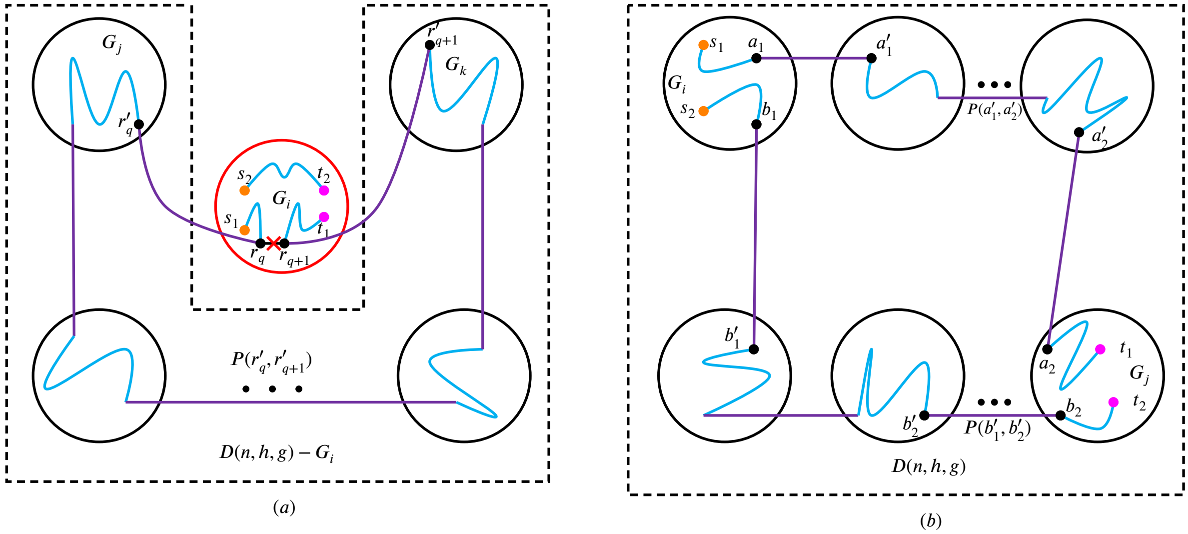


**Fig. 6.** An illustration of Theorem 3: (a) Case 1; (b) Case 2.1.

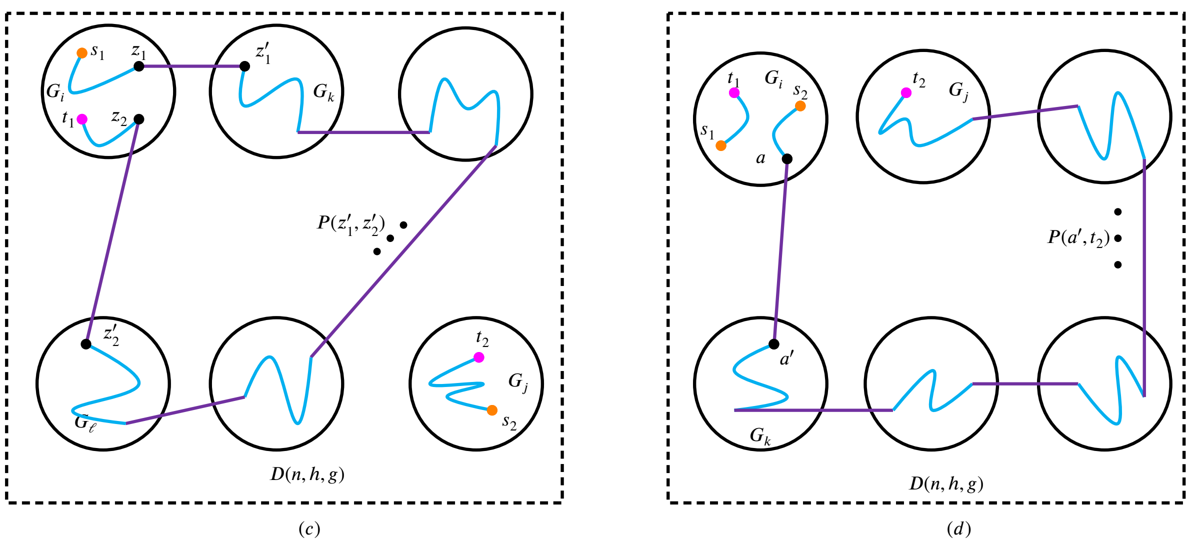


**Fig. 7.** An illustration of Theorem 3: (c) Case 2.2; (d) Case 2.3.

**Case 2.3.** For distinct $i, j \in [g]$, $|(S \cup T) \cap V(G_i)| = 3$ and $|(S \cup T) \cap V(G_j)| = 1$.

In this case, without loss of generality, let $s_1, s_2, t_1 \in V(G_i)$ and $t_2 \in V(G_j)$. Let $a$ be a vertex in $G_i$ and $a \notin \{s_1, t_1, s_2\}$. Since $G_i \cong K_n$, there exists a paired 2-DPC, denoted by $P(s_1, t_1)$ and $P(s_2, a)$, of $G_i$ joining $\{s_1, s_2\}$ and $\{t_1, a\}$. Let $a'$ be an external neighbor of $a$ and $a' \in V(G_k)$ with $k \neq j$. Then we can find a path, denoted by $P(a', t_2)$, connecting $a'$ and $t_2$ and put the groups covered by this path into the subgraph set $H$ with $H = G - G_i$. Thus, $G$ has a paired 2-DPC joining $\{s_1, s_2\}$ and $\{t_1, t_2\}$ as follows:

$$P_1(s_1, t_1) = P(s_1, t_1)$$

and

$$P_2(s_2, t_2) = P(s_2, a) + (a, a') + P(a', t_2).$$

Fig. 7 (d) shows the specific situation of Case 2.3.

**Case 3.** $S$ and $T$ are located in three different groups.

**Case 3.1.** Vertices in $S$ are located in a group and vertices in $T$ are located in other two different groups.

In this case, for distinct $i, j, k \in [g]$, without loss of generality, let $S \subseteq V(G_i)$, $t_1 \in V(G_j)$ and $t_2 \in V(G_k)$. For $h \geq 2$ and $n \geq 4$, let $a_1$ be a vertex in $G_i$ such that $a_1$ contains no external neighbors in $G_j$ and $a_1 \neq s_2$, especially, $a_1$ may be $s_1$. Let $a_2$ be a vertex in $G_i$ such that $a_2$ contains no external neighbors in $G_k$ and $a_2 \notin \{a_1, s_1\}$, especially, $a_2$ may be $s_2$. Since $G_i \cong K_n$, $G_i$ has a paired 2-DPC, denoted as $P(s_1, a_1)$ and $P(s_2, a_2)$, joining $\{s_1, s_2\}$ and $\{a_1, a_2\}$. Let $a'_1$ be an external neighbor of $a_1$ such that $a'_1 \in V(G_\ell)$ with $\ell \neq i, j, k$. Let $a'_2$ be an external neighbor of $a_2$ such that $a'_2 \in V(G_m)$ with $m \neq i, j, k, \ell$. By Lemma 4.3, one has that $a'_1$ and $t_1$ have at least one different external neighbor. By Definitions 3 and 4, we can find a path, denoted by $P(a'_1, t_1)$, connecting $a'_1$ and $t_1$ and put the groups covered by this path into the subgraph set $H_1$ with $H_1 \subset \{G_t : t \in [g] \setminus \{i, k, m\}\}$. Similarly, we can find a path, denoted by $P(a'_2, t_2)$, connecting $a'_2$ and $t_2$ and put the groups covered by this path into the subgraph set $H_2$ with $H_2 = G - H_1 - G_i$. Thus, $G$ has a paired 2-DPC joining $\{s_1, s_2\}$ and $\{t_1, t_2\}$ as follows:

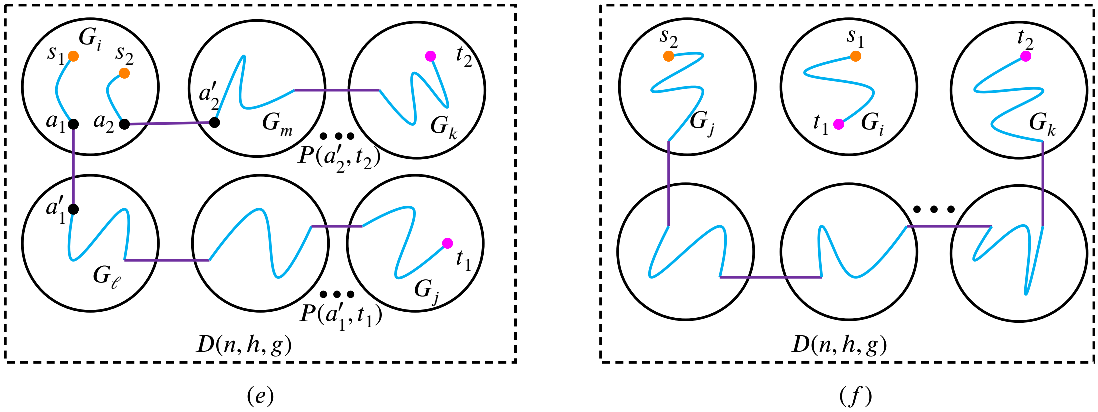


**Fig. 8.** An illustration of Theorem 3: (e) Case 3.1; (f) Case 3.2.

$P_1(s_1, t_1) = P(s_1, a_1) + (a_1, a'_1) + P(a'_1, t_1)$

and

$P_2(s_2, t_2) = P(s_2, a_2) + (a_2, a'_2) + P(a'_2, t_2).$

Fig. 8 (e) shows the specific situation of Case 3.1.

**Case 3.2.** One of vertices in $S$ and one of vertices in $T$ are located in a group, and the other vertices in $S \cup T$ are located in other two groups.

Without loss of generality, let $s_1, t_1 \in V(G_i)$, $s_2 \in V(G_j)$ and $t_2 \in V(G_k)$ with distinct $i, j, k \in [g]$. Since $G_i \cong K_n$, one has that $G_i$ has a hamiltonian path, say $P(s_1, t_1)$, connecting $s_1$ and $t_1$. By Definitions 3 and 4, one has that the graph obtained by contracting every group of $G - G_i$ is isomorphic to $K_{g-1}$. Thus, there exists a Hamiltonian path, say $P(s_2, t_2)$, of $G - G_i$ connecting $s_2$ and $t_2$. Thus, $G$ has a paired 2-DPC joining $\{s_1, s_2\}$ and $\{t_1, t_2\}$ as follows:

$P_1(s_1, t_1) = P(s_1, t_1)$

and

$P_2(s_2, t_2) = P(s_2, t_2).$

Fig. 8 (f) shows the specific situation of Case 3.2.

**Case 4.** Vertices of $S \cup T$ are located in four different groups.

For four distinct integers $i, j, k, \ell \in [g]$, without loss of generality, let $s_1 \in V(G_i)$, $s_2 \in V(G_j)$, $t_1 \in V(G_k)$ and $t_2 \in V(G_\ell)$. For each $m \in [g]$, since the graph $G_c$, obtained by contracting each group $G_m$ in $G$ to a vertex $x_m$, is isomorphic to $K_g$ and $G_m \cong K_n$, one has that there exists a path, denoted by $P(s_1, t_1)$, connecting $s_1$ and $t_1$. We then put the groups covered by this path into the subgraph set $H_1$ with $H_1 \subset \{G_t : t \in [g] \setminus \{j, \ell\}\}$. Similarly, we can find a path, denoted by $P(s_2, t_2)$, connecting $s_2$ and $t_2$ and put the groups covered by this path into the subgraph set $H_2$ with $H_2 = G - H_1$. Thus, $G$ has a paired 2-DPC joining $\{s_1, s_2\}$ and $\{t_1, t_2\}$ as follows:

$P_1(s_1, t_1) = P(s_1, t_1)$

and

$P_2(s_2, t_2) = P(s_2, t_2).$

Fig. 9 (g) shows the specific situation of Case 4.

Through the above analysis, for $n \geq 4$ and $h \geq 2$, $G = D(n, h, g)$ has a paired 2-DPC joining $S$ and $T$ where $S = \{s_1, s_2\}$ and $T = \{t_1, t_2\}$. □

## 6. Conclusion

The concept of 2-edge Hamilton-connected (2-EHC, for short) is an extension of Hamilton-connected and plays a crucial role in guaranteeing fault-tolerant network routing. However, it should be noted that 2-EHC is not universally applicable to all types of networks. Research has

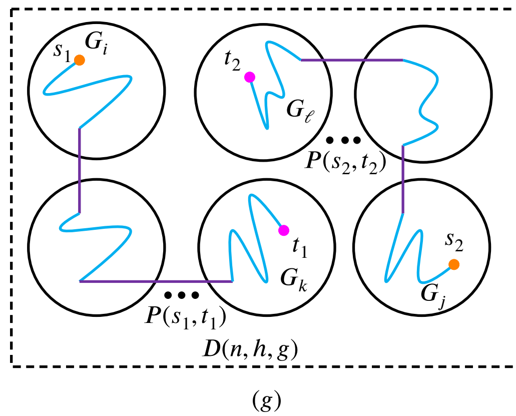


**Fig. 9.** An illustration of Theorem 3: (g) Case 4.

demonstrated that these networks retain 2-edge-Hamilton-connected (2-EHC) by applying proofs that rely on the properties of being 1-Hamilton-connected and having paired 2-disjoint path covers. This paper proves that the relative arrangement dragonfly network $D(n, h, g)$, which is a topology structure used in supercomputers, is 2-EHC. Moreover, we get that $D(n, h, g)$ is 1-Hamilton-connected and paired 2-disjoint path coverable. In the future, one can discuss the $k$-edge-Hamilton-connected of dragonfly network $D(n, h, g)$ with $k \geq 3$.

### CRediT authorship contribution statement

**Huimei Guo:** Writing – original draft, Validation, Software, Methodology, Formal analysis. **Rong-Xia Hao:** Writing – review & editing, Supervision, Investigation, Funding acquisition. **Jie Wu:** Writing – review & editing, Validation, Supervision, Conceptualization.

### Declaration of competing interest

The authors declare that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

### Acknowledgments

This work was supported by the National Natural Science Foundation of China (Nos. 12471321, 12331013 and 12161141005). This work was completed when Guo was an interim at China Telecom Cloud Computing Research Institute.

### Data availability

No data was used for the research described in the article.

## References


[1] B. Alverson, E. Froese, L. Kaplan, D. Roweth, Cray XC series network, white paper, Cray, 2012.

[2] B. Arimilli, R. Arimilli, V. Chung, S. Clark, W. Denzel, B. Drerup, T. Hoefler, J. Joyner, J. Lewis, J. Li, N. Ni, R. Rajamony, The PERCS high-performance interconnect, in: Proceedings of the 2010 18th IEEE Symposium on High Performance Interconnects, 2010, pp. 75–82.

[3] J.A. Bondy, U.S.R. Murty, Graph Theory, Springer, 2008.

[4] A. Bhatele, W.D. Gropp, N. Jain, L.V. Kale, Avoiding hotspots on two-level direct networks, in: Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis, 2011, pp. 1–11.

[5] A. Bhatele, N. Jain, Y. Livnat, V. Pascucci, P.-T. Bremer, Analyzing network health and congestion in dragonfly-based supercomputers, in: 2016 IEEE International Parallel and Distributed Processing Symposium, 2016, pp. 93–102.

[6] M. Belka, M. Doubet, S. Meyers, R.L. Momoh, D. Rincon-Cruz, D.P. Bunde, New link arrangements for dragonfly networks, in: 2017 IEEE 3rd International Workshop on High-Performance Interconnection Networks in the Exascale and Big-Data Era, 2017, pp. 17–24.

[7] Baidu, Baidu ring all-reduce, Available: https://github.com/baidu-research/baidu-allreduce, 2017.

[8] C. Camarero, E. Vallejo, R. Beivide, Topological characterization of Hamming and dragonfly networks and its implications on routing, ACM Trans. Archit. Code Optim. 11 (4) (2015) 25–39.

[9] G. Faanes, A. Bataineh, D. Roweth, T. Court, E. Froese, B. Alverson, T. Johnson, J. Kopnick, M. Higgins, J. Reinhard, Cray cascade: a scalable HPC system based on a dragonfly network, in: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis, vol. 103, 2012, pp. 1–9.

[10] Oak Ridge national laboratory, Frontier, https://www.olcf.ornl.gov/frontier/, 2022. (Accessed 30 September 2023).

[11] E. Faizian, M.S. Rahman, M.A. Mollah, X. Yuan, S. Pakin, M. Lang, Traffic pattern-based adaptive routing for intra-group communication in dragonfly networks, in: 2016 IEEE 24th Annual Symposium on High-Performance Interconnects (HOTI), 2016, pp. 19–26.

[12] M. García, E. Vallejo, R. Beivide, M. Odriozola, C. Camarero, G. Rodrguez, J. Labarta, C. Minkenberg, On-the-fly adaptive routing in high-radix hierarchical networks, in: 2012 41st International Conference on Parallel Processing, 2012, pp. 280–288.

[13] M. García, E. Vallejo, R. Beivide, M. Odriozola, M. Valero, Efficient routing mechanisms for dragonfly networks, in: 2013 42nd International Conference on Parallel Processing, 2013, pp. 582–592.

[14] M. García, E. Vallejo, R. Beivide, C. Camarero, M. Valero, G. Rodríguez, C. Minkenberg, On-the-fly adaptive routing for dragonfly interconnection networks, J. Supercomput. 71 (2015) 1116–1142.

[15] E. Hastings, D. Rincon-Cruz, M. Spehlmann, S. Meyers, A. Xu, D. Bunde, V. Leung, Comparing global link arrangements for dragonfly networks, in: Proceedings of the 2015 IEEE International Conference on Cluster Computing, 2015, pp. 361–370.

[16] N. Jiang, J. Kim, W.J. Dally, Indirect adaptive routing on large scale interconnection networks, ACM SIGARCH Comput. Archit. News 37 (3) (2009) 220–231.

[17] J. Kim, W. Dally, S. Scott, D. Abts, Technology-driven, highly-scalable dragonfly topology, ACM Sigarch Comput. Archit. News 36 (3) (2008) 77–88.

[18] J. Kim, W. Dally, S. Scott, D. Abts, Cost-efficient dragonfly topology for large-scale systems, IEEE MICRO 29 (1) (2009) 33–40.

[19] R. Kužel, Z. Ryjáček, P. Vrána, Thomassen's conjecture implies polynomiality of 1-Hamilton-connectedness in line graphs, J. Graph Theory 69 (3) (2012) 241–250.

[20] B. Prisacari, G. Rodrguez, C. Minkenberg, M. Garcia, E. Vallejo, R. Beivide, Performance optimization of load imbalanced workloads in large scale dragonfly systems, in: 2015 IEEE 16th International Conference on High Performance Switching and Routing, 2015, pp. 1–6.

[21] M.S. Rahman, S. Bhowmik, Y. Ryasnianskiy, X. Yuan, M. Lang, Topology-custon UGAL routing on dragonfly, Proc. Int. Conf. High Perform. Comput. 17 (2019) 1–15.

[22] Z. Salamah, A. Alzaid, B. Saptarshi, Y. Xin, L. Michael, Global link arrangement for practical dragonfly, in: Proceedings of the 34th ACM International Conference on Supercomputing, Association for Computing Machinery, vol. 21, 2020, pp. 1–11.

[23] R.S. Chaulagain, X. Yuan, Enhanced UGAL routing schemes for dragonfly networks, in: Proceedings of the 38th ACM International Conference on Supercomputing, 2024, pp. 449–459.

[24] M.-L. Wang, R.-X. Hao, J.-M. Chang, S. Bang, 2-edge Hamiltonian connectedness: characterization and results in data center networks, Appl. Math. Comput. 490 (2025) 129197.

[25] S. Wu, J. Fan, B. Cheng, J. Yu, Y. Wang, Connectivity and constructive algorithms of disjoint paths in dragonfly networks, Theor. Comput. Sci. 922 (2022) 257–270.

[26] S. Wu, B. Cheng, Y. Wang, Y. Han, J. Fan, Hamiltonian properties of the dragonfly network, New Trends Comput. Technol. Appl., Commun. Comput. Inf. Sci. 1723 (2022) 167–179.

[27] X. Wang, J. Fan, C.-K. Lin, X. Jia, Vertex-disjoint paths in DCell networks, J. Parallel Distrib. Comput. 96 (2016) 38–44.

[28] D. Xiang, X. Liu, Deadlock-free broadcast routing in dragonfly networks without virtual channels, IEEE Trans. Parallel Distrib. Syst. 27 (9) (2016) 2520–2532.

[29] D. Xiang, B. Li, Y. Fu, Fault-tolerant adaptive routing in dragonfly networks, IEEE Trans. Dependable Secure Comput. 16 (2) (2019) 259–271.

[30] P. Yébenes, J. Escudero-Sahuquillo, P. García, F. Quiles, Efficient queuing schemes for hol-blocking reduction in dragonfly topologies with minimal-path routing, in: 2015 IEEE International Conference on Cluster Computing, 2015, pp. 817–824.

[31] K. Ozeki, P. Vrána, 2-edge-Hamiltonian-connectedness of 4-connected plane graphs, Eur. J. Comb. 35 (2014) 432–448.

[32] Y.-K. Shih, S.-S. Kao, One-to-one disjoint path covers on $k$-ary $n$-cubes, Theor. Comput. Sci. 412 (35) (2011) 4513–4530.

[33] S.-Y. Kim, J.-H. Lee, J.-H. Park, Disjoint path covers in recursive circulants $G(2^m, 4)$ with faulty element, Theor. Comput. Sci. 412 (35) (2011) 4636–4649.

[34] S.-Y. Kim, J.-H. Park, Paired many-to-many disjoint path covers in recursive circulants $G(2^m, 4)$, IEEE Trans. Comput. 62 (12) (2013) 2468–2478.

[35] J.-H. Park, H.-C. Kim, H.-S. Lim, Many-to-many disjoint path covers in hypercube-like interconnection networks with faulty elements, IEEE Trans. Parallel Distrib. Syst. 17 (3) (2006) 227–240.

[36] J.-H. Park, H.-C. Kim, H.-S. Lim, Many-to-many disjoint path covers in the presence of faulty elements, IEEE Trans. Comput. 58 (4) (2009) 528–540.

[37] J.-H. Park, J.-H. Kim, H.-S. Lim, Disjoint path covers joining prescribed source and sink sets in interval graphs, Theor. Comput. Sci. 776 (2019) 125–137.

[38] J.-H. Park, Torus-like graphs and their paired many-to-many disjoint path covers, Discrete Appl. Math. 289 (2021) 64–77.

[39] J.-H. Park, Unpaired many-to-many disjoint path covers in nonbipartite torus-like graphs with faulty elements, IEEE Access 10 (2022) 127589–127600.

[40] H. Lü, Paired many-to-many two-disjoint path cover of balanced hypercubes with faulty edges, J. Supercomput. 75 (2019) 400–424.

[41] S. Jo, J.-H. Park, K.-Y. Chwa, Paired many-to-many disjoint path covers in faulty hypercubes, Theor. Comput. Sci. 513 (2013) 1–24.

[42] S. Zhang, S. Wang, Many-to-many disjoint path covers in $k$-ary $n$-cubes, Theor. Comput. Sci. 491 (2013) 103–118.

[43] T. Dvořák, M.-M. Gu, Paired 2-disjoint path covers of burnt pancake graphs with faulty elements, arXiv:2310.18831v1.

[44] Y. Fan, Y. Li, B. Zhang, L. Chen, Y. Wang, J. Guo, W. Wang, Y. Zhao, J. Zhang, GPU-efficient deployment of ring all-reduce-based distributed model training in tidal computing power network, in: 2023 International Photonics and Optoelectronics Meetings (ACP/POEM), 2023, pp. 1–4.

[45] Y. Jiang, H. Gu, Y. Lu, X. Yu, 2D-HRA: two-dimensional hierarchical ring-based all-reduce algorithm in large-scale distributed machine learning, IEEE Access 8 (2020) 183488–183494.

[46] M. Yu, Y. Tian, B. Ji, C. Wu, H. Rajan, J. Liu, GADGET: online resource optimization for scheduling ring-all-reduce learning jobs, in: IEEE INFOCOM 2022-IEEE Conference on Computer Communications, 2022, pp. 1569–1578.


**Huimei Guo** received her B.S. and M.S. degrees in Mathematics from Hanshan Normal University in 2019 and Xinjiang University in 2022, respectively. She is currently a PhD student at Beijing Jiaotong University. Her research interests including interconnection networks, reliability analysis of networks, graph theory.

**Rong-Xia Hao** received the Ph.D. degree from Beijing Jiaotong University, China in 2002. From 1998 to 2006, she was an associate professor. Since 2006, she was a professor at Department of Mathematic, Beijing Jiaotong University. She received Beijing Jiaotong University Zhi Jin Foundation Outstanding Youth Teaching Award in 2007 and the First Prize of 2008 Excellent Paper Awarded by Beijing Operations Research Society. Her research interests include graph theory, interconnection network and fault-tolerant computing.

**Jie Wu** is the Director of the Center for Networked Computing and Laura H. Carnell professor at Temple University. He also serves as the Director of International Affairs at College of Science and Technology. He served as Chair of Department of Computer and Information Sciences from the summer of 2009 to the summer of 2016 and Associate Vice Provost for International Affairs from the fall of 2015 to the summer of 2017. Prior to joining Temple University, he was a program director at the National Science Foundation and was a distinguished professor at Florida Atlantic University. His current research interests include mobile computing and wireless networks, routing protocols, cloud and green computing, network trust and security, and social network applications. Dr. Wu regularly publishes in scholarly journals, conference proceedings, and books. He serves on several editorial boards, including IEEE Transactions on Mobile Computing, IEEE Transactions on Service Computing, Journal of Parallel and Distributed Computing, and Journal of Computer Science and Technology. Dr. Wu was general co-chair for IEEE MASS 2006, IEEE IPDPS 2008, IEEE ICDCS 2013, ACM MobiHoc 2014, ICPP 2016, and IEEE CNS 2016, as well as program co-chair for IEEE INFOCOM 2011 and CCF CNCC 2013. He was an IEEE Computer Society Distinguished Visitor, ACM Distinguished Speaker, and chair for the IEEE Technical Committee on Distributed Processing (TCDP). Dr. Wu is a CCF Distinguished Speaker and a Fellow of the IEEE. He is the recipient of the 2011 China Computer Federation (CCF) Overseas Outstanding Achievement Award.