Mapping between Petersen Star Network and Hierarchical Petersen Network: Hierarchical Network Embedding

Jung-Hyun Seo [#1] and Hyeong Ok Lee [*2]
Dept. Electronic Commerce., National Univ. of Chonnam, Republic of Korea
Dept. Computer Edu., National Univ. of Sunchon, Republic of Korea
jhseo@scnu.ac.kr, oklee@scnu.ac.kr(Corresponding Author)

Article Info
Page Number: 11036 - 11049
Publication Issue:
Vol 71 No. 4 (2022)

Abstract
The demand for big data in data science and deep learning has been constantly increasing in recent years. In particular, deep learning using big data requires a high-performance computer. Interconnection networks for the design of high-performance computers are being consistently researched. The Petersen star network (PSN) and hierarchical Petersen network (HPN) are hierarchical interconnection networks (HINs) designed based on the Petersen graph and a small network cost. PSN(n) has 10n nodes and n+2 degrees, whereas HPN has 10n nodes and a degree of 5. When graph G is embedded in graph H, the algorithm developed in G can be simulated or reused in H. The evaluation measures for embedding include expansion, dilation, and congestion. First, previous studies on embedding are investigated and summarized. Second, the node and edge mapping functions between the two graphs are proposed. According to the proposed functions, it is shown that HPN(n) can be embedded into PSN(n) with an expansion of 1, dilation of n−1, congestion of 1, average dilation of 0.4n, and average congestion of 1. Furthermore, it is shown that PSN(n) can be embedded into HPN(n) with an expansion of 1, a dilation of n+4, a congestion of 0.5n, an average dilation of 0.5n, and an average congestion of 0.5n. The results of this study can be extended to prove the upper and lower limits of dilation and congestion with respect to the network characteristics.

Article History
Article Received: *15 September 2022*
Revised: *25 October 2022*
Accepted: *14 November 2022*
Publication: *21 December 2022*

Keywords: Dilation; Congestion; Embedding; Hierarchical Petersen Network; Petersen Star.

Introduction

A computer largely consists of software and hardware. Increase in functionalities of software and superior performance of hardware are expected over time. Hardware performance significantly depends on the processor speed and memory capacity. Parallel computing, which connects multiple processors and memories, is slowly being implemented and developed to improve the hardware performance. In parallel computing, a multiprocessor system refers to a connection of multiple processors, and multi-computing refers to the connection of multiple processors equipped with memory [1].

Multi-computing comprises many processors and communication links. A communication link exists between the processors. The performance of a multi-computing system is closely related to the connection structure of the processors. The connection structure of a processor is called an interconnection network and can be represented as a graph. A processor is represented as a node, whereas a communication link is represented as an edge. A graph consists of nodes and edges, where an edge connects the nodes [2].

Interconnection networks proposed in the initial phase include simple graphs such as the torus, hypercube, and star graphs. A graph can be transformed as follows. First, an edge can be added or removed from a simple graph. A torus [3] can be created by adding wrap-around edges to a mesh [4]. Second, two simple graphs can be combined, which is called a product network. A hyperstar [5] is created by combining a hypercube [6] and a star graph [7]. Third, a simple graph can be hierarchically expanded, which is commonly referred to as a hierarchical interconnection network (HIN). An HIN can include a hierarchical cubic network HCN(n,n) [8], a hierarchical star HS(n,n) [9], a hierarchical hypercube network HHN(n,n;h) [10], a hierarchical folded hypercube network HFN(n,n) [11], and a hierarchical hypercube n-HHC [12]. A recently proposed HIN includes the Petersen star network PSN(n) [13], 4-connectivity of hierarchical cubic k_m-HCN(n) [14], and a hierarchical Petersen network HPN(n) [15]. The networks that were proposed have been commercialized as MasPar, Intel Paragon, XP/S, Intel Touchstone Delta System, Mosaic C, Cray T3D, J-Machine of MIT, and Tera Computer [16].

Various algorithms need to be developed to implement an interconnection network to realize multi-computing. The development of various algorithms must be preceded by a topological analysis of the interconnection networks. The topological elements that need to be analyzed for the development of algorithms consist of the number of nodes, number of edges, degree, subgraph, Hamilton path, spanning tree, and parallel path. Routing and broadcast, in addition to process allocation, load balancing, task migration, and embedding, are the processes that require the development of a basic algorithm for message transmission [17]. Embedding allows the algorithms developed in an arbitrary interconnection network to be used in other interconnection networks.

Embedding refers to the mapping of an arbitrary graph to a different graph. G(V, E) refers to the guest graph, whereas the host graph is H(V, E). If graph G can be embedded in graph H, the algorithms developed for graph G can also be used in graph H. Embedding involves node mapping and edge mapping. A node mapping function is called α, whereas an edge mapping function is called β. The embedding function is $f = (\alpha, \beta)$. Node mapping refers to mapping a node in graph G to a node in graph H, and the function is $\alpha: V(G) \rightarrow V(H)$. An arbitrary edge of graph G is given as $e = (u, v)$, where $\alpha(u) \rightarrow x$ and $\alpha(v) \rightarrow y$. Here, x and y are the nodes of graph H, and the path from x to y is called $p(x, y)$. Edge mapping refers to mapping an edge in graph G to a path in graph H, and the function is $\beta: (u, v) \rightarrow p(x, y)$. The precondition for embedding is that all V(G) and E(G) must participate in the mapping [18].

The evaluation measures for embedding include expansion, dilation, and congestion. The expansion of f is defined as $|V(H)| / |V(G)|$. One-to-one embedding refers to the mapping of one node in G to one node in H. In most studies, |V(G)| = |V(H)|, where the expansion is 1, is optimal. The path in graph H to which an edge (u, v) of graph G is mapped is called p(α(u), α(v)). Dilation of an edge (u, v) and that of f refers to the length of path p and the maximum dilation value of all edges, respectively. The congestion of an edge e′ and that of f in graph H refer to the number of paths ρ(x, y) containing e′ and the maximum congestion value of all edges in graph H, respectively. Dilation and congestion are greater than or equal to 1. The optimal value is 1; the average dilation and congestion values also correspond to significant measures [19].

The following studies were conducted for embedding. Bettayeb et al. embedded a star graph Sn into a hypercube graph Qn with a dilation of $\lceil \log_2 n! \rceil - 1$ [20]. Ranka et al. embedded a mesh M(n,n) into Sn with an expansion of 1 and dilation of 3 [21]. Saikia et al. embedded torus T(n,n) into Sn with a dilation of 4 and congestion of $\boldsymbol{O}(n)$ [22]. Kim et al. embedded a folded hypercube FQ_n into a folded hyperstar FHS($2n$,n) with a dilation of 2 and that of 1 conversely [23]. Bouabdallah et al. embedded a complete binary tree into S_n and pancake P_n with a dilation of 1 [24]. Seo et al. embedded a half pancake HP_n into S_n with a dilation of 1.5n−2 and a congestion of 6, and they also embedded a P_n into S_n with a dilation of 1.5n [25]. Embedding algorithms have been recently proposed [18, 19, 26-28].

Subsequent studies on embedding in hierarchical networks were conducted. Seo et al. embedded HPN(n) into a folded Petersen [29] FP_k with an expansion of 1, a dilation of $2k$, and a congestion of 4 [15]. Wei et al. embedded M(n,n) into HS(n,n) with a dilation of 3 and congestion of 4 [9]. Campbell et al. embedded M($2h$,$2h$) into hierarchical cliques $HiC_{(4,h)}$ with an expansion of 1, a dilation of $2^{(h-1)}$, and a congestion of $2h-1$ [30]. Chiang et al. embedded Q2n into HCN(n,n) with a dilation of 3 and an average dilation of $2-(1/2^n)$ [31]. Hamdi et al. embedded HHN(n,n;$n-1$) into Q_{2n} with an expansion of 1 and a dilation of 2 [32]. Kim et al. embedded HCN(n,n) into a hierarchical folded hypercube HFN(n,n) with a dilation of 3 and also embedded a Q_{2n} into HCN(n,n) and HFN(n,n) with a dilation of 3 and showed that an embedding in reverse requires a dilation of $\boldsymbol{O}(n)$ [33].

This study aims to prove that HPN and PSN can be embedded into each other. In Section II, HIN and embedding are briefly explained, and previous studies on embedding are summarized; PSN and HPN are also examined. In Section III, node and edge mapping functions between HPN and PSN are proposed; expansion, dilation, congestion, average dilation, and average congestion are derived. Finally, Section IV concludes the paper.

Design method of a hierarchical network

HIN recursively expands a base graph (hereafter referred to as a “factor”). When HIN expands a factor, fewer edges than the increasing number of nodes are added to maintain a short average distance between the nodes [34]. Fig. 1 depicts the HCN and HHN, which have Q_2 as a factor. As shown in Fig. 1(b), HCN(2,2) has Q_2 as a factor, and 2^2 factors are connected in the form of Q_2. The edges that were added after an expansion from (a) to (b) and then from (b) to (c) are represented as orange solid lines. As shown in Fig. 1(b) and (c), HHN(2,h) expands Q_2 to the h level. HHN(n,n) is the unfixed degree network (UDN), which increases the degree up to $2n$ as the network expands.

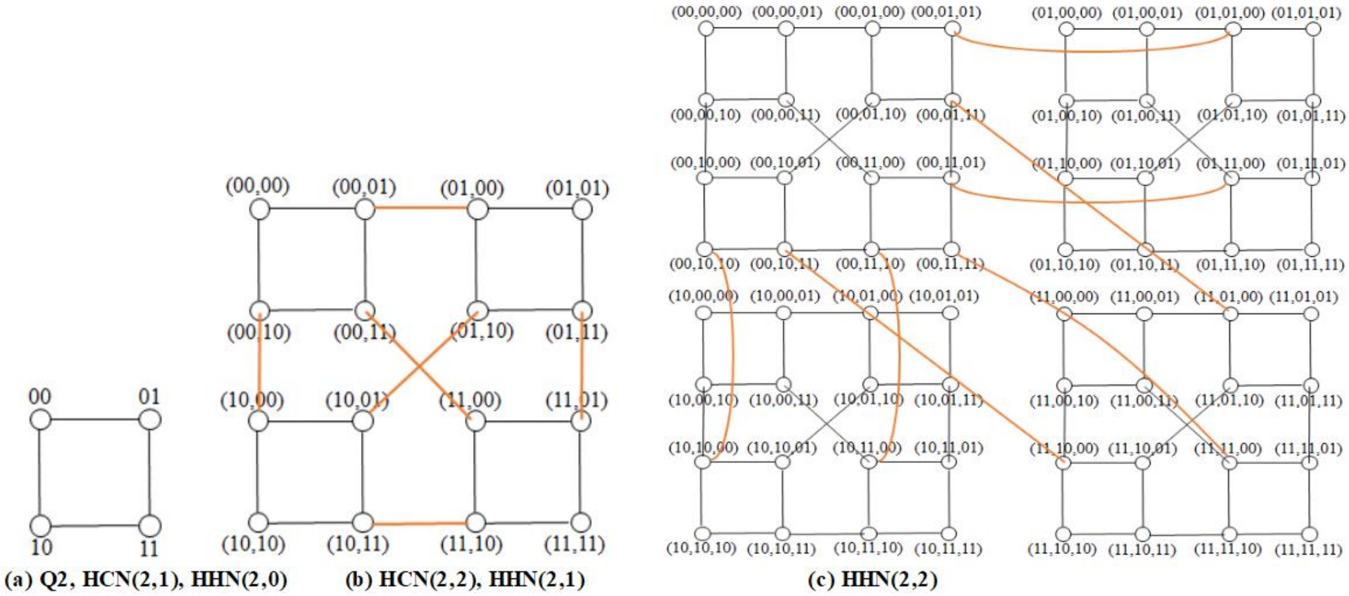

Figure 1. Hierarchical network

Introduction of embedding and investigation of previous research

Fig. 2 shows the embedding of a binary tree into Q3. The function that embeds a binary tree G into Q3 is called f. Each node of graph G is mapped to a node of graph H in which their node addresses are converted to binary numbers. The mapped pairs of nodes are as follows: (0, 000), (1,001), (2, 010), (3, 011), (4, 100), (5, 101), (6, 110). The expansion is given as |V(H)| / |V(G)| = 1.14. Note that f is a one-to-one embedding because all nodes are mapped one-to-one. The edges (0,1), (0,2), (1,3), and (1,4) in graph G are mapped to (0,1), (0,2), (1,3), and (1,4) in graph H, and thus, dilation of this edge is 1.

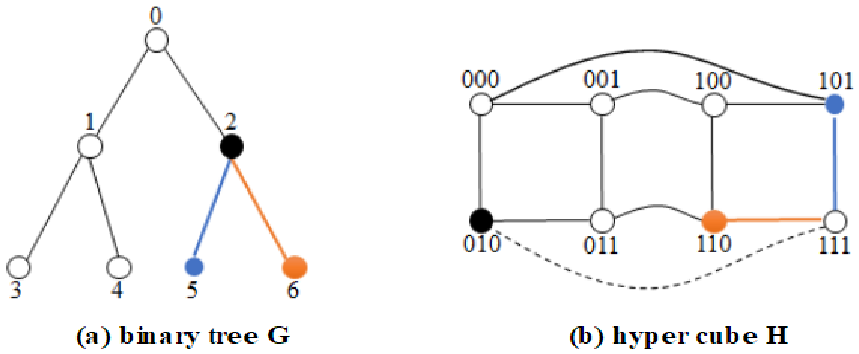

Figure 2. Embedding binary tree into hypercube Q3

Edge (2, 5) marked in blue in Fig. 2 is mapped to path 010 → 111 → 101 in graph H. Edge (2, 6) marked with orange is mapped to path 010 → 111 → 110 in graph H. The dilation of edges (2, 5) and (2, 6) is 2; thus, the dilation of f is 2. The paths to which the edges excluding (2, 5) and (2, 6) are mapped are edge-disjoint in H and thus have a congestion of 1. Edge (010, 111) has a congestion of 2 because the mapped paths of edges (2, 5) and (2, 6) contain (010, 111); thus, the congestion of f is 2. The expansion indicates the number of nodes in which the processes executed in graph G are simulated in graph H, and the dilation indicates the extent to which the routing path length of graph G is extended in graph H. The congestion indicates how much the routing path congestion of graph G has worsened in graph H. While simulating graph G in graph H, the message delay time increases for an increase in the dilation and congestion.

The characteristics of the two networks involved in embedding contribute to the end results. An interconnection network can be divided into a UDN, where the degree increases, and a fixed degree network (FDN), where the degree is fixed as the network expands. A mesh or torus is an FDN, whereas a hypercube, a star, and an HIN are UDNs. The expansion was measured to be 1 in almost all studies, and the dilation was thoroughly examined in this study. Tables 1 and 2 present the results of the investigated embeddings that have been reported thus far. In Table 1, the graphs involved in embedding are distinguished into a UDN and FDN, where the torus, hypercube, and star graphs and their transformations are included. The dilation is found to be O(1) for the worst case when the FDN is embedded into an FDN or UDN and is found to be O(n) otherwise. The network names in Tables 1 and 2 are abbreviated if they have been defined in the introduction, and full names are provided otherwise.

Table 2 presents the results of the investigation of embeddings including the HIN. The dilation is found to be O(1) in the worst case when a UDN is embedded into HIN and O(n) otherwise. The dilation of O(n) may result in the worst case because this study deals with embeddings among HINs.

Tables 1 and 2 show that embedding a UDN (including HIN) into an FDN is not easy. Organizing Tables 1 and 2 by including the HIN in the UDN yields the results presented in Table 3. The dilation between FDN and UDN in Table 3 is greater than or equal to O(1) and smaller than or equal to O(n).

Table 1. Dilation of embedding between FDN and UDN

Guest	Host	Guest	Host	dilation	Ref.
FDN	FDN	$M(n,n)$	$T(n,n)$	1	[38]
		n level $T(n,n)$	n level $M(n,n)$	2	
		$T(3n,2n)$	hexagonal honeycomb torus n-HHT	3	[39]
	UDN	$M(n,n)$	S_n	3	[21]
		$M(2^n,3^n)$	hexcube HC_n	2	[36]
		$M(2^{n/2},2^{n/2})$	Q_n	1	[38]
		$T(n,n)$	S_n	4	[22]
		Binary Tree	S_n	1	[24]
		Binary Tree	P_n	1	[24]
		$n!$ ring	P_n	1	[37]
		$M(n,(n-1)!)$	P_n	7	
UDN	FDN	Q_n	$T(2^{n/2},2^{n/2})$	$n/2$	[38]
	UDN	S_n	Q_n	$\lceil \log_2 n! \rceil - 1$	[20]
		FQ_n	$FHS(2n,n)$	2	[23]
		FQ_n	Q_n	2	[35]
		augmented cube AQ_n	Q_n	2	
		AQ_n	FQ_n	2	

		crossed cube CQ_n	Q_n	2	
		FHS(2*n*,*n*)	FQ_n	1	[23]
		Q_n	HC_n	3	[36]
		Q_n	P_n	6	[37]
		HP_n	S_n	1.5*n*-2	[25]
		P_n	S_n	1.5*n*	[25]

Table 2. Dilation of embedding between HIN, UDN and FDN

Guest	**Host**	**Guest**	**Host**	**dilation**	**Ref.**
HIN	**FDN**	not found			
	UDN	HPN(*n*)	FP_n	2*n*	[15]
		HHN(*n*,*n*;*n*-1)	Q_{2n}	2	[32]
		HCN(*n*,*n*)	Q_{2n}	***O***(*n*)	[33]
		HFN(*n*,*n*)	Q_{2n}	***O***(*n*)	[33]
HIN	**HIN**	HCN(*n*,*n*)	HFN(*n*,*n*)	3	[33]
		HFN(*n*,*n*)	HCN(*n*,*n*)	***O***(*n*)	[33]
FDN	**HIN**	M(*n*,*n*)	HS(*n*,*n*)	3	[9]
		$M(2^n,2^n)$	$HiC_{(4,n)}$	2(*n*-1)	[30]
UDN		Q_{2n}	HCN(*n*,*n*)	3	[31]
		Q_{2n}	HFN(*n*,*n*)	3	[33]

Table 3. Bestand worst cased of embedding dilation

Guest	**Host**	**dilation**	
		best case	**worst case**
FDN	**FDN**	***O***(1)	***O***(*n*)
FDN	**UDN**	***O***(1)	***O***(*n*)
UDN	**FDN**	***O***(*n*)	***O***(*n*)
UDN	**UDN**	***O***(1)	***O***(*n*)

Petersen graph

Petersen graph is used in both PSN and HPN. It is regular and has node (edge) symmetry. It has degree 3 and diameter 2. In this section, an address consisting of permutations of double digits is used; in the other section, for convenience of indicating the node address, an address consisting of single digits in parentheses is used. Petersen graph is shown in Fig. 3. The number on the left in the node address is smaller than the number on the right. In Petersen graph, P = (Vp, Ep); Node Vp = xy; Edge Ep = (xy, x'y'); x, y∈{1,2,3,4,5}; and x', y'∈ {{1,2,3,4,5} − {x, y}}. It is assumed that in Petersen graph, u=u1u2 is the start node and v=v1v2 the destination node. Node w is composed of {1,2,3,4,5} – {{u1, u2} ∪ {v1, v2}}. The routing algorithm from u to v is given by Algorithm 1 [15].

Algorithm 1. *Routing* (*u*, *v*) {

if $(u \cap v) = \emptyset$ then $u \rightarrow v$;

else $u \rightarrow w \rightarrow v$;

3:}

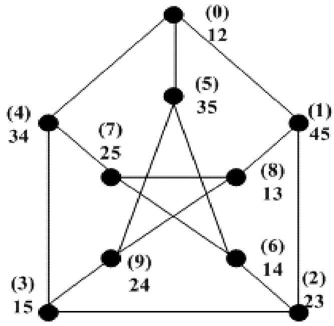

Figure 3. Petersen graph

PSN and HPN

Modified PSN [13] and HPN [15] are proposed in this section. An external edge was modified to simplify it without adjusting the degree or diameter. PSN(0) and HPN(0) are both Petersen graphs. The edges of HPN(n) and PSN(n) are divided into internal and external edges. Edges connecting nodes belonging to the same Petersen graph (hereafter referred to as a "factor") are called internal edges; the edges of the Petersen graph are used as they are. Edges connecting nodes in different factors are called external edges [13, 15]. $u=u_1u_2u_3 \cdots u_{i-1}u_iu_{i+1} \cdots u_{n-1}u_n$, $p=p_1p_2p_3 \cdots p_{i-1}p_ip_{i+1} \cdots p_{n-1}p_n$. The operation of an external edge is as follows:

Definition 1. *exchange* $EEi(p) = p_ip_2p_3 \cdots p_{i-1}p_1p_{i+1} \cdots p_{n-1}p_n$.

Definition 2. *left rotate* $LR(u) = u_2u_3 \cdots u_{i-1}u_iu_{i+1} \cdots u_{n-1}u_nu_1$.

Definition 3. *right rotate* $RR(u) = u_nu_1u_2u_3 \cdots u_{i-1}u_iu_{i+1} \cdots u_{n-1}$.

Definition 4. Petersen Edge $PE(xyu_2u_3 \cdots u_{i-1}u_iu_{i+1} \cdots u_{n-1}u_n) = (x'y'u_2u_3 \cdots u_{i-1}u_iu_{i+1} \cdots u_{n-1}u_n)$ or $PE(xyp_2p_3 \cdots p_{i-1}p_1p_{i+1} \cdots p_{n-1}p_n) = (x'y'p_2p_3 \cdots p_{i-1}p_1p_{i+1} \cdots p_{n-1}p_n)$.

$$PSN(n) = (Vp,Ep) \tag{1}$$

The node and edge are defined below.

$$Vp = \{p_1p_2p_3 \cdots p_{i-1}p_ip_{i+1} \cdots p_{n-1}p_n \mid 0 \le pi \le 9, 1 \le i \le n, 2 \le n\} \tag{2}$$

$$\text{external edge EEi} = (p, \text{EE}(p)) \tag{3}$$

$$\text{internal edge PE} = (p, \text{PE}(p)) \tag{4}$$

Fig. 4 shows a two-level PSN. Black solid lines represent the internal edges, while the orange solid lines represent the external edges. To avoid the complexity of Fig. 4, the external edges are shown only for the factor x0(0≤x≤9). The PSN(n) has 10n nodes and a degree of n+2.

HPN(n) = (V_{hp}, E_{hp}). The node and edge are defined below.

$$Vhp = \{u_1u_2u_3 \cdots u_{i-1}u_iu_{i+1} \cdots u_{n-1}u_n \mid 0 \le ui \le 9, 1 \le i \le n, 3 \le n\} \tag{5}$$

$$\text{external edge LR} = (u, \text{LR}(u)) \tag{6}$$

$$\text{external edge RR} = (u, \text{RR}(u)) \tag{7}$$

$$\text{internal edge PE} = (u, \text{PE}(u)) \tag{8}$$

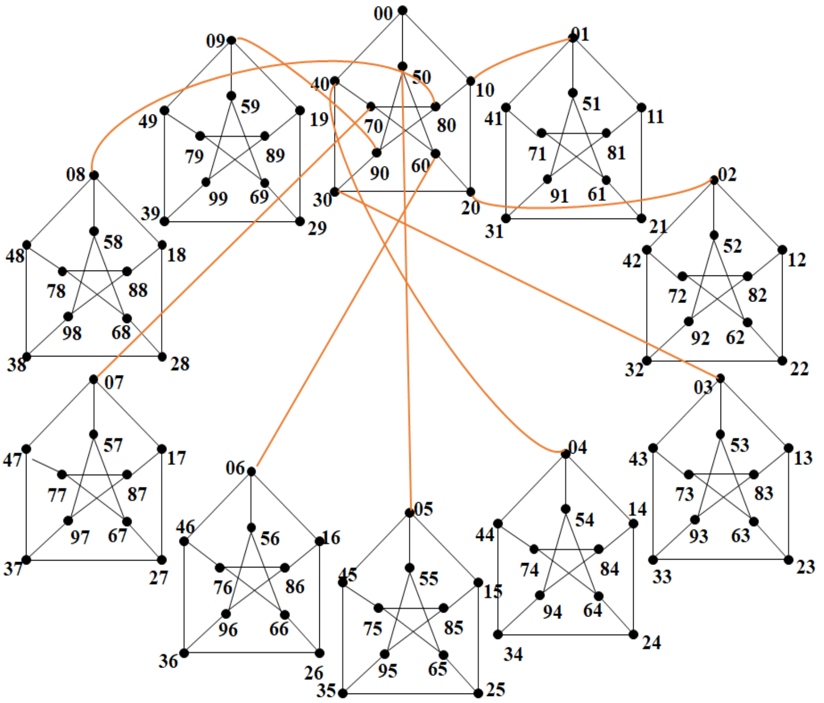

Figure 4. Two-level Petersen star network PSN(2)

Fig. 5 shows the structure of HPN(3) [15]; (a) and (b) do not belong to HPN. HPN(n) has 10n nodes and a degree of 5.

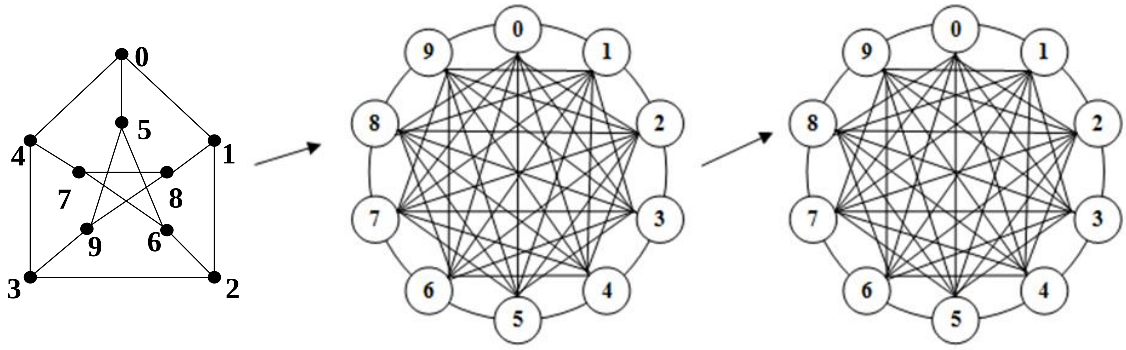

Figure 5. Hierarchical petersen network HPN(3)

Emvedding algorithm

Two embeddings are proposed in this section. First, a node mapping function commonly used in both embeddings is proposed. Next, a function for mapping the edge of HPN(n) into the path of PSN(n) is proposed, and the dilation, congestion, and average dilation are analyzed. Finally, a function for mapping the edge of PSN(n) into the path of HPN(n) is proposed, and the dilation and congestion are analyzed. The mapping graph is G, the graph being mapped is H, and the embedding function is f = (α, β). Function 1 is the node mapping that is commonly used in both embeddings.

Two embeddings are proposed in this section. First, a node mapping function commonly used in both embeddings is proposed. Next, a function for mapping the edge of HPN(n) into the path of PSN(n) is proposed, and the dilation, congestion, and average dilation are analyzed. Finally, a function for mapping the edge of PSN(n) into the path of HPN(n) is proposed, and the dilation and congestion are analyzed. The mapping graph is G, the graph being mapped is H, and the embedding function is f = (α, β). Function 1 is the node mapping that is commonly used in both embeddings.

Function 1. *Node mapping α:*

u and v are the arbitrary nodes of graphs G and H, respectively; α:u→v and α:v→u. Here, the node addresses of u and v are exactly the same.

Theorem 1. *The embedding between HPN(n) and PSN(n) is one-to-one, and the expansion is 1.*

Proof of Theorem 1. The number of nodes of HPN(n) and PSN(n) are both 10^n, while the length of node addresses for both is n. Moreover, the domain of symbols constituting the node addresses of both is a natural number ranging from 0 to 9. A node in graph G is mapped to a node in graph H through function 1. Thus, f is a one-to-one embedding. In addition, |G|/|H| = |H|/|G| = 1; therefore, the expansion of f is 1. □

The nodes of the PSN(n) are $p = p_1p_2p_3 \cdots p_{i-1}p_ip_{i+1} \cdots p_{n-1}p_n$. The nodes of HPN($n$) are $u=u_1u_2u_3 \cdots u_{i-1}u_iu_{i+1} \cdots u_{n-1}u_n$.

Notation 1. *The symbol ⇒ refers to the path through the internal edges according to Algorithm 1, while → refers to the path through the external edges. For example, paths 1234, 2134, 6134, and 3164 in PSN(4) are expressed as follows:*

1234→2134⇒6134→3164 or 1234→EE_2→2134⇒PE⇒6134→EE_3→3164.

Mapping the edges of HPN(n) into the path of PSN(n)

HPN(n) is referred to as G, while PSN(n) is referred to as H in this section. The edge (u, v) of G is mapped to the path p(α(u), α(v)) of H by function 1, where the function β that creates path p is proposed. Dilation and congestion are then analyzed. For a better understanding, the case of n = 4 is examined first, and then the function is generalized. Fig. 6 depicts the factor x234, which is a portion of HPN(4) and PSN(4). Function 1 maps all nodes in Fig. 6(a) exactly to the nodes having the same address in Fig. 6(b).

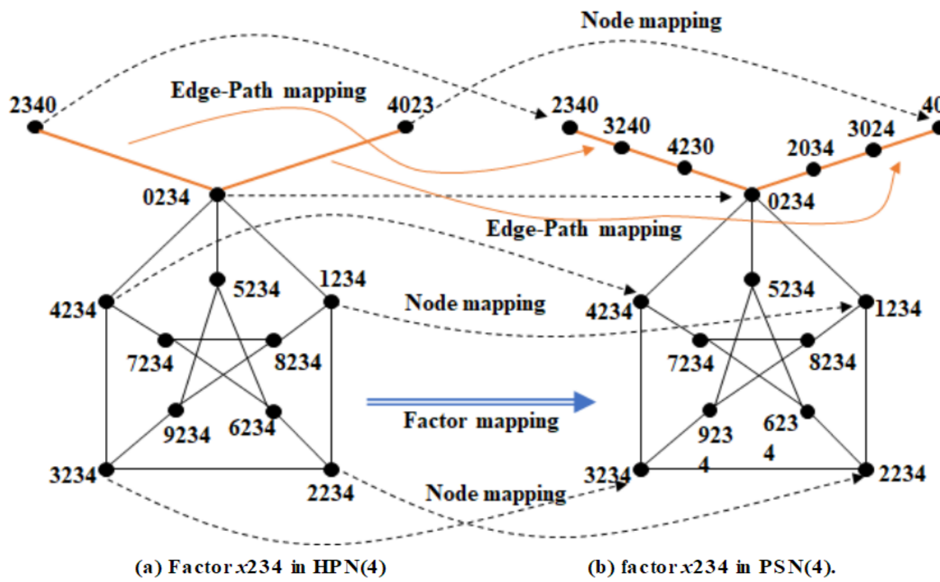

Figure 6. Embedding HPN(4) into PSN(4)

Factor x234 of G is mapped to the factor x234 of H; thus, the mapping of the internal edges is omitted. The external edges of G are mapped to the path of H. Edge (0234, 2340) of G is mapped to path p1 (0234, 2340) of H, whereas the edge (0234, 4023) is mapped to path p2 (0234, 4023). Using notation 1, path p1 becomes $0234 \rightarrow EE_4 \rightarrow 4230 \rightarrow EE_3 \rightarrow 3240 \rightarrow EE2 \rightarrow 2340$, while path p2 becomes $0234 \rightarrow EE_2 \rightarrow 2034 \rightarrow EE_3 \rightarrow 3024 \rightarrow EE_4 \rightarrow 4023$. The generalized function β is shown in function 2.

Function 2. Edge mapping β:

(a) $\beta((u, LR(u)))$: $p1 = u \rightarrow EE_n \rightarrow EE_{n-1} \rightarrow \cdots \rightarrow EE_3 \rightarrow EE_2 \rightarrow LR(u)$.

(b) $\beta((u, RR(u)))$: $p2 = u \rightarrow EE_2 \rightarrow EE_3 \rightarrow \cdots \rightarrow EE_{n-1} \rightarrow EE_n \rightarrow LR(u)$.

(c) $\beta(u, PE(u))$: $u \Rightarrow PE(u)$.

Theorem 2. *HPN(n) is embedded into PSN(n) with a dilation of n−1 and a congestion of 1.*

Proof of Theorem 2. The ten nodes in which all the symbols have the same value in G are called loop nodes. In HPN(4), the nodes 0000, 1111, 2222, …, 8888, and 9999 are loop nodes. All nodes in G have three internal and two external edges. The loop nodes do not have any external edges. Functions 1 and 2(c) map all the factors of G to all the factors of H having the same address; thus, dilation and congestion of internal edges are 1. The external edges are divided into LR and RR. According to functions 2(a) and 2(b), the lengths of paths p1 and p2 are n−1; thus, the dilation of f is n−1. Next, the congestion is examined. If LR(u) = RR(u), the number of external edges is 1; thus, the congestion is 1. If LR(u) ≠ RR(u), β((u, LR(u))) and β((u, RR(u))) are edge disjoint, and the congestion of f is 1. The reasons for this are as follows. If the two paths β((u, LR(u))) and β((u, RR(u))) are to overlap, $u = EE2(u) = EEn(u)$ or $u_1 = u_2 = u_n$, the congestion is 1 because a new edge is not added to paths p1 and p2. If $u_1 \neq u_2 = u_n$ or $u_1 = u_2 \neq u_n$, β((u, LR(u))) and β((u, RR(u))) are edge disjoint. □

Corollary 3. *HPN(n) is embedded into PSN(n) with an average dilation of 0.4n+0.2 and an average congestion of 1.*

Proof of Corollary 3. The average dilation of G is equal to the average dilation of all the edges. Similarly, the average congestion refers to the average of all the congestion values. All nodes have three internal edges and two external edges; therefore, the average congestion and dilation are examined from the edges connected to one node. G has 10 loop nodes, without any external edges. The number of loop nodes among all 1000 nodes in HPN(3) is 10, or 1%, and 0.1% in HPN(4). Loop nodes were disregarded while calculating the average dilation. It was proven in Theorem 1 that the dilation and congestion of the internal edges are 1, while those of the external edges are n−1 and 1, respectively. Therefore, the average dilation of f is $(3+2(n-1)) / 5 = (2n+1)/5 = 0.4n+0.2$. The congestion of the external edges and internal edges is 1; thus, the average congestion of f is 1.

Mapping the edges of PSN(n) into the path of HPN(n)

PSN(n) is referred to as G, whereas HPN(n) is referred to as H in this section. The edge (u, v) of G is mapped to the path p(α(u), α(v)) of H by function 1, where a function β that creates a path p is proposed. Dilation and congestion are then analyzed. For a better understanding, the case of $n = 4$ is examined first, and then the function is generalized. Fig. 7 shows the factor x234, which is a portion of PSN(4) and HPN(4). Function 1 maps all nodes in Fig. 5(a) exactly to the nodes having the same address in Fig. 5(b).

The factor x234 of G is mapped to factor x234 of H; thus, the mapping of the internal edges is omitted. The external edges of G are mapped to the path of H. Edge (0234, 2034) of G is mapped to path p1 (0234, 2034) of H, whereas the edges (0234, 3204) and (0234, 4230) are mapped to path p2 (0234, 3204) and p3 (0234, 4230), respectively.

Paths p1, p2, and p3 can be expressed using notation 1 as follows:

p1 = 0234→LR→2340⇒PE⇒0340→RR→0034⇒PE⇒2034.

p2 = 0234→RR→4023→RR→3402→PE⇒0402⇒LR→4020 →LR→0204⇒PE⇒3204.

p3 = 0234→RR→4023⇒PE⇒0023→LR→0230⇒PE⇒4230.

The edge mapping function β can be generalized as follows: LR×i indicates that LR is repeated i times.

Function 3. Edge mapping β:

(a) when $2 \le i \le \lceil n/2 \rceil$, β((u, EEi(u))) : pi = u→LR×(i−1)⇒PE⇒RR×(i−1) ⇒PE⇒ EEi(u).

(b) when $\lceil n/2 \rceil < i \le n$, β((u, EEi(u))) : pi = u→RR×(n−i+1) ⇒PE⇒LR×(n−i+1) ⇒ PE⇒EEi(u).

(c) when β(u, PE(u)) : u⇒PE(u).

Theorem 4. *PSN(n) is embedded into HPN(n) with a dilation of n+4 and a congestion of 0.5.*

Proof of Theorem 4. All nodes in G have three internal edges. Owing to the function 3(a), the dilation and congestion of the internal edges are 1. The maximum number of external edges EE_i($2 \le i \le n$) is measured to be n−1. The path length of the PE is 2 in terms of the diameter of the Petersen graph. The worst case in function 3 refers to i $= \lceil n/2 \rceil + 1$. In this case, the length pi is (n/2)×2 + 2×2 ≤ n+4. Thus, the dilation of f is n+4. Next, the congestion is examined. There are no external edges if u = EE_i(u). Nodes 1111, 4444, and 6666 of PSN(4) have no external edges. Nodes 1123, 1413, and 1731 have two external edges. Similarly, nodes 2122, 2212, and 2224 have one external edge. Obviously, it is omitted in the embedding if there are no external edges. Node u = 1234567 in PSN(7) is examined next. The edge (u, EE_i(u)) is expressed as follows:

When $2 \le i \le \lceil n/2 \rceil$,

e2=(1234567, 2134567), e3=(1234567, 3214567), e4 = (1234567, 42314567).

When $\lceil n/2 \rceil < i \le n$,

e5=(1234567, 5234167), e6=(1234567, 6234517), e7=(1234567, 7234561).

Edges e2, e3, and e4 are mapped to paths p2, p3, and p4 by function 3(a), whereas edges e5, e6, and e7 are mapped to paths p5, p6, and p7 by function 3(b). The mapped path is explained as follows: several paths below overlap at the underlined parts.

p2=1234567→2345671⇒1345671→1134567⇒2134567.

p3=1234567→2345671→3456712⇒1456712→2145671→ 1214567⇒3214567.

p4=1234567→2345671→3456712→4567123⇒1567123→3156712→2315671→1231567⇒423 1567.

p5=1234567→7123456→6712345→5671234⇒1671234→6712341→7123416→1234167⇒523 4167.

p6=1234567→7123456→6712345⇒1712345→7123451→ 1234517⇒6234517.

p7=1234567→7123456⇒1123456→1234561⇒7234561.

If the case above is generalized, congestion of the path u→LR×1 is $\lceil n/2 \rceil - 1$, while congestion of the path u→RR×1 is n−$\lceil n/2 \rceil$. When n is an even number, $\lceil n/2 \rceil - 1 <$ n −$\lceil n/2 \rceil$= 0.5n; when n is an odd number, $\lceil n/2 \rceil - 1$ = n −$\lceil n/2 \rceil \leq$ 0.5n. Therefore, the congestion of f is less than or equal to 0.5. □

Corollary 5. *PSN(n) is embedded into HPN(n) with an average dilation of 0.5n+6 and an average congestion of 0.5n+6.*

Proof of Corollary 5. The average dilation of G is the same as the average dilation of all the edges. Similarly, the average congestion refers to the average of all the congestion values of the edges. All nodes have three internal edges and n−1 external edges; therefore, the average dilation and congestion are examined sequentially from the edges connected to one node. The nodes of G do not have any external edges when the other symbols have the same value as the first symbol. Ten percent of the nodes do not have external edges in PSN(2) because each symbol can have a value from 0 to 9. For example, 11, 22, and 33 were the relevant cases here. Approximately 10% of the nodes do not partially have external edges in the PSN(n). The nodes that do not partially have edges are disregarded when calculating the average dilation. It was proven in Theorem 4 that the dilation of the internal edges is 1. According to functions 3(a) and 3(b), the sum of the dilation of edges where $2 \leq i \leq \lceil n/2 \rceil$ is$\sum_{i=1}^{\lfloor n/2 \rfloor} 2i + 4$, and the sum of dilation of edges where $\lceil n/2 \rceil < i \leq n$ is $\sum_{i=1}^{\lceil n/2 \rceil} 2i + 4$. Therefore, the average dilation of f is $(n^2/2 + 6n + 3) / (n+2) \leq 0.5n + 6$.

The congestion of the internal edges is 1, while the congestion of the external edges can be explained as follows.

When $2 \leq i \leq \lceil n/2 \rceil$.

The path u→LR×1 has a congestion of $\lceil n/2 \rceil - 1$.

The path LR×1→LR×2 has a congestion of $\lceil n/2 \rceil - 2$.

The path LR×($\lceil n/2 \rceil$ −2)→LR×($\lceil n/2 \rceil$ −1) has a congestion of $\lceil n/2 \rceil - (\lceil n/2 \rceil - 1) = 1$.

When $\lceil n/2 \rceil + 1 \leq i \leq n$.

The path RR×($\lceil n/2 \rceil$ −1)→RR×($\lceil n/2 \rceil$) has a congestion of $\lceil n/2 \rceil - (\lceil n/2 \rceil - 1) = 1$.

The path RR×1→RR×2 has a congestion of $\lceil n/2 \rceil - 2$.

The path RR×1 of f has a congestion of $\lceil n/2 \rceil - 1$.

Therefore, the average congestion of f is ≤ 0.5n + 6. □

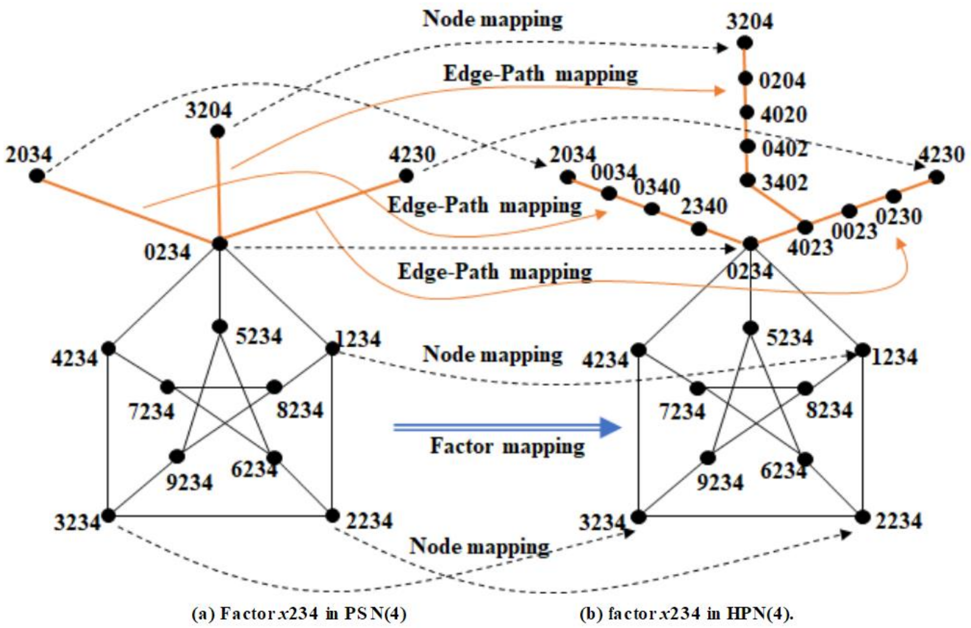

Figure 7. Embedding PSN(4) HPN(4)

Conclusions

Embedding algorithms between HPN and PSN were proposed, and the expansion, dilation, and congestion were derived in this study. The expansion of the two embeddings is 1. HPN(n) was embedded into PSN(n) with a dilation of n−1 and congestion of 1; PSN(n) was embedded into HPN(n) with a dilation of n+4 and congestion of 0.5n. Hence, it was proven that the algorithms developed in the two interconnection networks can be used interchangeably by paying the costs of dilation O(n) and congestion O(n). Previous studies conducted on embedding have been reviewed and summarized in Section II to examine the significance of this result. According to previous studies, embedding between HIN results on the dilation of O(n) corresponds to the worst case. In addition, further research is recommended to prove the upper and lower limits of dilation and congestion with respect to the network characteristics.

Acknowledgment

This paper was supported by(in part) Sunchon National University Research Fund in 2021.(Grant number: 2021-0323)

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. 2020R1A2C1012363).

※ MSIT : Ministry of Science and ICT

References

[1] M. Moudi, "M. Othman. A survey on emerging issues in interconnection networks," Int. J. Internet Technol. Secur. Trans, vol. 11, no. 2, pp. 131–159, Jan 2021.

[2] R. K. Gupta, K. L. Jaiswal, "Study and analysis of graphs and models based interconnection network system," Int. J. Eng. Tech. Mgmt. Res, vol. 8, no. 1, pp. 53–56, Jan 2021.

[3] A. Bossard, “On solving the decycling problem in a torus network Wirel,” Commun. Mob. Comput, vol. Art, no. 5598173, Apr 2021.

[4] D. Sharma, K. Devasheesh, A. Punhani, “Heterogeneous channel allocation in mesh interconnection network using genetic algorithm in Proceedings of International Conference on Big Data,” Machine Learning and their Applications, Springer, Singapore, pp. 351–356, 2021.

[5] A.-E. Al-Ayyoub, K. Day, “The hyperstar interconnection network J. Parallel Distrib,” Comput vol. 48, no. 2, pp. 175–199, Feb 1998.

[6] K. Karthik, S. Jena, T. Venu Gopal, Performance evaluation, comparison and identification of efficient hypercube interconnection networks Wireless Pers. Commun, to be published.

[7] K. S. Ajish Kumar, D. Rajendraprasad, K. S. Sudeep, Oriented diameter of star graphs. Discrete Appl. Math., to be published. DOI:https://doi.org/10.1016/j.dam(2021.01.025)

[8] K. Ghose, K. R. Desai, “Hierarchical cubic network IEEE Trans,” Parallel Distrib. Syst Nov. vol. 6, no. 4, pp. 427–435, 1995.

[9] W. Shi, P. K. Srimani, “Hierarchical star: A new two-level interconnection network,” J. Syst. Archit, vol. 51, pp. 1–14, Jan 2005.

[10] S. K. Yun, K. H. Park, “Hierarchical hypercube networks (HHN) for massively parallel computers,” J. Parallel Distrib. Comput, vol. 37, pp. 194–199, Sep 1996.

[11] D. -R. Duh, G. -H. Chen, J. -F. Fang, “Algorithms and properties of a new two-level network with folded hypercubes as basic modules IEEE Trans,” Parallel Distrib, vol. 6, no. 7, pp. 714–723, Jul 1995.

[12] Q. M. Malluhi, M. A. Bayoumi, “The Hierarchical Hypercube: A New Interconnection Topology for Massively Parallel System IEEE Trans,” Parallel Distrib, vol. 5, no. 1, pp. 17–30, Syst Jan 1994.

[13] J. -H. Seo, H. O. Lee, “Design and analysis of the Petersen star interconnection network Far East,” J. Electron. Commun, vol. 17, no. 4, pp. 719–732, Aug 2017.

[14] S. -L. Zhao, R. -X. Hao, J. Wu, "The generalized 4-connectivity of hierarchical cubic networks Discrete Appl,” Math, vol. 289, pp. 194–206, Jan. 2021.

[15] J. -H. Seo, J. -S. Kim, H. J. Chang, H. -O. Lee, “The hierarchical Petersen network: a new interconnection network with fixed degree,” J. Supercomput, vol. 74, no. 4, pp. 1636–1654, Nov 2017.

[16] J. -H. Seo, H. O. Lee, “Design and analysis of the rotational binary graph as an alternative to hypercube and Torus,” J. Supercomput, vol. 76, pp. 7161–7176, Jan. 2020

[17] M. Alam, M. Shahid, “A load balancing strategy with migration cost for independent batch of tasks (BoT) on heterogeneous multiprocessor interconnection networks,” Int. J. Appl. Evol. Comput, vol. 8, no. 3, pp. 74–92, Jul. 2017.