Embeddings into Hyper Petersen Networks: Yet Another Hypercube-Like Interconnection Topology*

SAJAL K. DAS, SABINE ÖHRING and AMIT K. BANERJEE
Department of Computer Science, University of North Texas, Denton, TX 76203, USA

A new hypercube-like topology, called the hyper Petersen (HP) network, is proposed and analyzed, which is constructed from the well-known cartesian product of the binary hypercube and the Petersen graph of ten nodes.

This topology is an attractive candidate for multiprocessor interconnection having such desirable properties as regularity, high symmetry and connectivity, and logarithmic diameter. For example, an n-dimensional hyper Petersen network, $HP_n$, with $N = 1.25 \times 2^n$ nodes is a regular graph of degree and node-connectivity $n$ and diameter $n - 1$, whereas an $(n - 1)$-dimensional binary hypercube, $Q_{n-1}$, with the same diameter covers only $2^{n-1}$ nodes, each of degree $(n - 1)$. Thus the HP topology accommodates 2.5 times extra nodes than $Q_{n-1}$ at the cost of increasing the node-degree by one. With the same degree and connectivity of $n$, the diameter of the $HP_n$ network is one less than that of $Q_n$, yet having 1.25 times larger number of nodes.

Efficient routing and broadcasting schemes are presented, and node-disjoint paths in $HP_n$ are computed even under faulty conditions. The versatility of the hyper Petersen networks is emphasized by embedding rings, meshes, hypercubes and several tree-related topologies into it. Contrary to the hypercubes, rings of odd lengths, and a complete binary tree of height $n - 1$ permit subgraph embeddings in $HP_n$.

Key Words: Communication, embedding, fault tolerance, hypercube, interconnection network, Petersen graph

1 INTRODUCTION

One of the important aspects of a message-passing parallel computer is its interconnection network topology, which can be represented as an undirected graph such that each node represents a processor-memory module and an edge is a bidirectional communication link between two processors. Several network topologies have been proposed including trees, meshes, shuffle-exchange [1], hypercubes, cube connected cycles [2], pyramids [3], recursive networks [4], stars and pancakes [5], de Bruijn networks [6], folded Petersen networks [7] and so on.

Among these networks, the hypercubes have received significant attention because of their rich topological properties such as symmetry, high node-connectivity, logarithmic degree and diameter. Also, several commercial parallel machines (e.g. Intel iPSC, NCUBE, Connection machine) use this topology to interconnect processors. In the recent years, various modifications have been incorporated into the hypercube structure to enhance its performance while preserving the useful properties. Twisted hypercube [8] and multiply twisted hypercube [9] provide two such examples, for which the diameter is reduced without increasing node degrees, whereas for folded hypercubes [10] and bridged hypercubes [11] the diameter is reduced with increasing node-degree.

Now, the fact that the well-known Petersen graph [12] of ten nodes has degree three and diameter two compared to a three-dimensional hypercube (with only eight nodes, each of degree three) having diameter equal to three, motivates us to design a new topology which makes use of the Petersen graph as a building block. At the same time our goal is to exploit the existing properties of the popular hypercube network. Thus a natural way to extend a hypercube is to replace each of its nodes by a Petersen graph, which is precisely the effect of a cartesian product. The proposed network constructed this way is called the hyper Petersen (HP) network, which
turns out to be an attractive alternative of the hypercubes for multiprocessor interconnection with slightly smaller diameter and higher packing density. This paper also emphasizes the importance of the HP network as a versatile interconnection topology, by embedding various other structures into it. The guest networks include rings, grids, complete binary trees [13], full-ringed binary trees [14], meshes of trees [15], tree machines [16], X-trees [17], hyper-trees [18], and pyramids [3]. These mapping problems are of practical importance. For example, an algorithm developed for any such guest network can be efficiently implemented onto the HP network, if there exists an efficient mapping of that guest into the HP host. Also if a problem has a tree-related task or data structure as in divide-and-conquer, branch-and-bound or image processing (quadtrees) algorithms, an efficient mapping of that structure onto the HP network is needed for an efficient parallel implementation of these algorithms on the HP.

The paper is organized as follows. In Section 2 we introduce various graph-theoretic terminology and definitions. Section 3 gives a formal definition of the HP network, while Section 4 discusses its topological properties. In Section 5 routing and broadcasting techniques are given. Section 6 shows how to embed other topologies into the HP network. Section 7 compares the embedding results with those in other host networks, and Section 8 concludes this paper.

2 DEFINITIONS AND TERMINOLOGY

An interconnection network is usually modeled as an undirected graph \( G = (V, E) \) where \( V \) is the node-set representing processors and \( E \) the edge-set representing the communication links. We will use standard graph-theoretic terminology [12].

The degree, \( \deg(u) \), of a node \( u \) is the number of edges incident on \( u \), and it is a measure of the network connection cost. If \( \deg(u) = \delta \), for \( 1 \leq u \leq |V| = N \), then \( G \) is called \( \delta \)-regular, or a regular graph of degree \( \delta \). The diameter (\( d \)) of a network is a measure of its performance in terms of worst-case communication delay. It is defined as \( d = \max_{i,j \in V} \text{dist}_G(i,j) \), where \( \text{dist}_G(i,j) \) is the shortest distance between nodes \( i \) and \( j \) in \( G \). The node-connectivity (\( \kappa \)) is the number of nodes whose removal results in a disconnected network. It is a measure of fault-tolerance of the network. The \( f \)-fault diameter is defined as the worst-case diameter by removing at most \( f \) nodes. For a binary string \( x = x_{k-1} \ldots x_0 \) we denote with \( x(i) := x_{i+1} \ldots x_{k-1}x_i x_{i-1} \ldots x_0 \) the string \( x \) complemented in the \( i \)th bit, \( 0 \leq i \leq k - 1 \).

There are several graph-theoretic operations which enable us to aggregate large networks from smaller networks of known properties. One such operation is the so called cartesian product [12] of two graphs \( G_1 = (V_1, E_1) \) and \( G_2 = (V_2, E_2) \). The resulting network \( G = G_1 \times G_2 = (V, E) \) has the node-set 

\[
V = V_1 \times V_2
\]

and an edge \( \{ (x, y), (u, v) \} \in E \) iff either

\[
x = u \quad \text{and} \quad \{ y, v \} \in E_2
\]

or

\[
y = v \quad \text{and} \quad \{ x, u \} \in E_1.
\]

Clearly, \( G_1 \times G_2 \) satisfies the following relations on the number of nodes, degree, diameter, and node-connectivity: \( N = N_1 \times N_2 \), \( \delta = \delta_1 + \delta_2 \), \( d = d_1 + d_2 \), and \( \kappa_1 + \kappa_2 \leq \kappa \leq \delta \).

An \( n \)-dimensional binary hypercube, \( Q_n \), consisting of \( 2^n \) nodes has the node-set \( V_n = \{ x_i \mid x_i \in \{ 0, 1 \}^n \} \), the set of binary strings of length \( n \). Two nodes \( u \) and \( v \) are adjacent iff their Hamming distance is \( H(u, v) = 1 \), i.e. the binary labels of \( u \) and \( v \) differ in exactly one bit position. In terms of the cartesian product operation, hypercubes can be recursively defined as \( Q_n = Q_{n-1} \times K_2 \), where \( K_2 \) is the complete graph on two nodes. It is known that \( \delta = d = \kappa = n \) for \( Q_n \).

The incomplete hypercube [19] \( I_n \), \( 0 \leq k < n \), has \( 2^n + 2^k \) nodes consisting of a front cube \( Q_n \) and a back cube \( Q_k \). The node-set is given by the set of \( (n + 1) \)-bit labels such that \( (0^w)^* \) and \( (1(0)^{n-1-k}^* \) denote the front cube and the back cube, respectively, where \( * \) is either 0 or 1. An edge between two nodes \( x \) and \( y \) of \( I_n \) exists iff \( x \) and \( y \) differ in exactly one bit.

The undirected binary de Bruijn graph [20] \( DG(n) \) of order \( n \) is defined by the node-set \( V_n = \{ x_i \mid x_i \in \{ 0, 1 \}^n \} \) and the edge-set

\[
E_n = \{ \{ x_{i-1} \ldots x_0, x_{n-2} \ldots x_0 \} \mid p, x_i \in \{ 0, 1 \} \}
\]

The order-(\( m, n \)) hyper-de Bruijn graph [21] is the product graph \( HD(m, n) = Q_m \times DG(n) \).

An \( n \)-dimensional twisted hypercube, \( TQ_n \), is constructed as follows. We select two distinct edges, say \( (u, v) \) and \( (w, z) \), in a 4-cycle of the hypercube \( Q_n \) which have no nodes in common, and replace them with two new edges \( (u, z) \) and \( (v, w) \) [22]. Clearly, such a twist does not affect the degree of a node (i.e., \( \delta = \kappa = n \)), but the diameter of \( TQ_n \) is reduced to \( d = n - 1 \). A recursive application of the twist operation results in a multiply twisted hypercube, \( MTKQ_n \), having a significant reduction in the diameter (see Table 1).

3 HYPER PETERSEN NETWORKS

The Petersen graph, \( P \), with ten nodes has an outer 5-cycle, an inner 5-cycle and five spokes joining them (see Figure 1a)). This is an important graph
TABLE I

Comparison of HP Network with Other Hypercube-like Networks

<table>
<thead>
<tr>
<th>Network</th>
<th>HPₙ</th>
<th>Qₙ</th>
<th>TQₙ</th>
<th>MTQₙ</th>
<th>DG(n)</th>
<th>HD(m, n)</th>
<th>I⁺</th>
</tr>
</thead>
<tbody>
<tr>
<td># nodes (N)</td>
<td>1.25(2ⁿ)</td>
<td>2ⁿ</td>
<td>2ⁿ</td>
<td>2ⁿ</td>
<td>2ⁿ⁺²</td>
<td>2ⁿ + 2⁻³</td>
<td></td>
</tr>
<tr>
<td>degree (δ)</td>
<td>n</td>
<td>n</td>
<td>n</td>
<td>n</td>
<td>n + 4</td>
<td>k + 1 ≤ δ ≤ n + 1</td>
<td></td>
</tr>
<tr>
<td>diameter (d)</td>
<td>n - 1</td>
<td>n</td>
<td>n - 1</td>
<td>[n + 1] / 2</td>
<td>n</td>
<td>m + n</td>
<td>n + 1</td>
</tr>
<tr>
<td>connectivity (κ)</td>
<td>n</td>
<td>n</td>
<td>n</td>
<td>n</td>
<td>n</td>
<td>n</td>
<td>k + 1</td>
</tr>
<tr>
<td>cost (C = d ⋅ δ)</td>
<td>n(n - 1)</td>
<td>n²</td>
<td>n² - n</td>
<td>(n(n + 1)) / 2</td>
<td>4n</td>
<td>(n + 4)(m + n)</td>
<td>(n + 1)²</td>
</tr>
</tbody>
</table>

belonging to the rare "Moore graphs" family, and has been studied extensively. For example, P is the smallest 3-regular graph in which the shortest cycle has length five. Therefore, it is also called "(3, 5)-cage." Its diameter d = 2 and node-connectivity κ = 3. It is highly symmetric, but neither Hamiltonian nor planar. For more details on the Petersen graph, refer to [23].

There are several ways to characterize the node-adjacency of the Petersen graph. One way to construct this graph is by considering the ten subsets of size two (called two-subsets) of the set {1, 2, 3, 4, 5} as nodes, and joining those pairs of subsets that are disjoint [23]. For example, there exists an edge between the pair of nodes {1, 2} and {3, 4} because they are disjoint subsets of size two. However, nodes {1, 2} and {1, 3} are not adjacent. To make the drawing less cumbersome, we designate a node {1, 2} as 12 in Figure 1a). In the rest of the paper, we will instead use the decimal labeling of the nodes as also shown in Figure 1a).

The proposed n-dimensional (for n ≥ 3) hyper Petersen network, HPₙ = Qₙ⁻³ × P, is a cartesian product of a (n - 3)-dimensional hypercube and P. Informally, it is obtained from Qₙ⁻³ such that each of its nodes is replaced by a 10-node Petersen graph, P. For the sake of computational convenience, each node in HPₙ is represented as a two-tuple: (u, v), where u corresponds to an (n - 3)-bit binary label of the hypercube part, and v is a node in the Petersen graph. The label u is called the prefix label of a node (denoted either in binary or in decimal encoding in the following) and the corresponding Petersen graph will be denoted as Pᵤ.

One advantage of constructing a large network with the help of the cartesian product is that the characteristics of the component networks are preserved. For example, since both graphs Qₙ⁻³ and P are node- and edge-symmetric, HPₙ is also node- and edge-symmetric.

4 PROPERTIES OF HP NETWORKS

An n-dimensional hyper Petersen network, HPₙ, is a regular graph with N = 2ⁿ + 2ⁿ⁻² = 1.25 + 2ⁿ nodes, each of degree δ = n, and having diameter d = n - 1. The prefix labels of any two nodes, say (x, u) and (y, v), in HPₙ can differ in at most (n - 3) bits. Thus by complementing one bit at a time, the Petersen graph Pᵤ to which (y, v) belongs can be reached from Pₓ in at most n - 3 steps. Then it requires at most two more steps to locate v within Pᵤ.

Theorem 1: The node-connectivity of HPₙ is n. Its (n - 1)-fault diameter is n + 2.

Proof: The node-connectivity of HPₙ is κ = n, since there are n node-disjoint paths between two arbitrary nodes (x, y) and (x', y') of length at most dist_HPₙ(x, y), (x', y')) + 3 as shown below.

Let i₁, ... , iₘ be the m differing bits between x and x', and let k ∈ {0, 1}, be the number of different
positions between \( y \) and \( y' \) such that \( \alpha = \text{dist}_P(y, y') \). Furthermore, \( y' \), for \( 1 \leq i \leq 3 \), are the neighbors of \( y \) in \( P \). Let \( \gamma = 1 \) for \( \alpha = 2 \), otherwise \( \gamma = 3 \) for \( \alpha = 1 \), where \( \gamma \) is the difference between the lengths of a non-shortest and a shortest path in \( P \) connecting \( y \) and \( y' \). Assume \( p_j, j = 1, 2 \), be the non-shortest paths in \( P \) between \( y \) and \( y' \). Then there are \( m + k \) node-disjoint shortest paths \( p_j, 1 \leq j \leq m + k \), (of length \( r = m + k \)) between \( (x, y) \) and \( (x', y') \) by equalizing the differing positions in cyclic order \((i_1, \ldots, i_m, i_{m+k})\). Further \( n - 3 - m \) paths of length \( r + 2 \) can be constructed for \( j \not\in \{i_1, \ldots, i_m\} \) by the sequence \((x, y), (x(j), y'), (x', y')\), and \((1 - k)\) paths of length \( r + 2 \) by the sequence \((x, y), (x, y'), (x', y')\), equalizing the bits in \( x \) and \( x' \) by using the path \( p_j, (x', y), (x', y') \). And \( 2k \) paths of length \( r + \gamma \) can be constructed by the node-sequence \((x, y), (x', y'), (x', y')\), equalizing the \( j \)th element as in \( p_j \), \((x', y'), (x', y')\). The node-disjointness of these constructed paths is proven by us in [24]. If follows that there exists still a path of length at most \( \text{dist}_P((x, y), (x', y')) + 3 \) even after removal of \( n - 1 \) nodes.

Let us now compare the topological properties of \( HP_n \) with several other hypercube-like networks (Table 1). For the facts stated in this table, refer to [1, 25, 21, 19].

Maintaining the same degree \( \delta = n \), the \( HP_n \) network can accommodate 1.25 times extra nodes than in \( Q_n \), yet having the diameter reduced by one. On the other hand, considering diameter \( d = (n - 1) \) as the criterion for comparison, \( HP_n \) has better connectivity and covers 2.5 times more nodes than the hypercube \( Q_{n-1} \), only at the cost of increasing the node-degree by one. Although the degree and diameter of \( HP_n \) is identical to those of a twisted hypercube, \( TQ_n \), the packing density is higher in the former network. As can be seen from Table 1, a special case of the incomplete hypercube, namely \( p_{n-2} \), consisting of the same number of nodes as \( HP_n \), has larger diameter and poorer connectivity.

The de Bruijn graph \( DG(n) \) provides a bounded degree network and has poor reliability, as compared to \( HP_n \), which turns out to be a versatile network. It is worth pointing out that the de Bruijn graph is very efficient for (dynamic) embeddings of trees [26], but not for grids [27].

Contrary to the hyper-de Bruijn topology, \( HP_n \) provides higher symmetry and regularity. Also, the hyper Petersen network \( (HP_n) \) has reduced degree and diameter and higher connectivity than the hyper-de Bruijn network \( HD(n - 3, 3) \) consisting of \( 2^n \) nodes.

### 4.1 Fault Tolerance

Fault-tolerance capabilities of a network can be improved with the existence of a wide container, providing a large number of node-disjoint paths between any two nodes [10]. The width of a container is the number of node-disjoint paths it includes. The length of a container is the longest path length in it. The best container of a given width between two nodes is the one of the shortest length. The container quality between two nodes at a distance \( r \) in a node-symmetric network is given by \( \text{CQ}(r) = (\Gamma_1/\Gamma_2) \), where \( \Gamma_1 \) is the number of node-disjoint paths and \( \Gamma_2 \) is the average length of all disjoint paths between those two nodes in the best container. A large value of \( \text{CQ}(r) \) means a short container that includes many paths.

The container quality of \( HP_n \) (between two nodes \((x, y)\) and \((x', y')\) in \( HP_n \) at distance \( r \)) is given by

\[
\text{CQ}_{HP}(r) = \frac{2^r}{(m + k) \cdot r + (n - m - 3k) \cdot (r + 2) + 2k(r + \gamma)} = \frac{2^r}{(n \cdot r + 2n - 2m - 2\gamma)}
\]

If \( k = 0 \) and \( \alpha = 2 \), then

\[
\text{CQ}_{HP}(r) = \frac{2^r}{(n \cdot r + 2n - 2m - 2m - 2\gamma)} = \frac{2^r}{(n \cdot r + 2n - 4m)}
\]

Thus, \( \text{CQ}_{HP}(r) = 1 \) for \( \alpha = 2 \) and \( \text{CQ}_{HP}(r) = 1 \) for \( \alpha = 1 \). This means that \( HP_n \) provides a container that is approximately as good as in \( Q_n \).

Let us now compute the probability of failure of \( HP_n \). Since its node-connectivity, upto \( (n - 1) \) node faults can be tolerated without disconnecting the network. Also, \( HP_n \) will be disconnected if all the neighboring processors of a particular node are faulty. Let us denote the probability of occurrence of such a faulty event as \( F(HP_n) \), and let the corresponding parameter for a binary hypercube, \( Q_n \), be \( F(Q_n) \). Suppose the binomial coefficient, \( C(i, j) \), denotes the number of ways of choosing \( j \) objects out of \( i \) objects.

Then considering a fault set of \( n \) nodes in a \( Q_n \), there exists \( C(2^n, n) \) different faults. Among these only \( 2^n \) faults result in disconnecting the network, and again each such set consists of nothing but the set of all neighbors of a given node. Similar logic applies to a hyper Petersen topology. Thus, \( F(Q_n) = \ldots \)
(2^n/C(2^n, n)), and $F(HP_n) = (1.25 \times 2^n) / (C(1.25 \times 2^n, n))$ such that $(F(HP_n)/F(Q_n)) = (1.25 \times C(2^n, n)) / (C(1.25 \times 2^n, n)) << 1$. It implies that for a given pattern of faulty nodes, the probability of disruption of an $n$-dimensional hyper Petersen network is significantly less than that of $Q_n$. In other words, on the average the fault tolerance capability of $HP_n$ is better.

### 4.2 Routing and Broadcasting

For routing, let $S = (x, y)$ and $D = (x', y')$ be the source and destination nodes, respectively, where $S \neq D$. If $x = x'$, the routing is restricted within a single Petersen graph, $P^x$, from node $y$ to $y'$ and requires at most two hops due to the diameter of the Petersen graph. When $x \neq x'$, we equalize the differing bit-positions between $x$ and $x'$ in a sequential order until we reach the node $(x', y')$. If $y = y'$, accept the message; otherwise route it within the Petersen graph $P^x$ until the destination is found. This procedure ensures that point-to-point message routing in $HP_n$ is via a shortest path and requires at most $(n - 1)$ hops.

In the case of at most $n - 1$ faults, and when each node knows the status of another node in $HP_n$, the routing is done along one of the $n$ node-disjoint paths of length at most $dist_{HP_n}(S, D) = 3$ (cf. Theorem 1). On the other hand, if each node $(u, v)$ has only local knowledge about the status of $2^{n-3} + 8$ other nodes (having the hypercube-part label $u$ or the Petersen-part label $v$), a path of length $\leq n + 7$ can be found. First try to send the message to the intermediate node $(x, y')$ using routing in $P$ for $y' \neq y'$. If this is not possible, since $(x, y')$ or at least three nodes in the subgraph $P^x$ are faulty, try to route the message from $(x, y)$ to $(x', y)$ using hypercube routing, if $x \neq x'$. If this is not possible due to $(x', y)$ or at least $n - 3$ nodes in $Q^x_{n-3}$ being faulty, request a path from the hypercube part neighbors $(x(i), y)$, $0 \leq i \leq n - 4$, and the Petersen-part neighbors $(x, y')$, $0 \leq i \leq 2$. Because of at most $n - 1$ faulty nodes, one of these paths is operational and its length is at most $n + 7$.

The broadcasting commonly used in a hypercube can be easily adopted for the hyper Petersen configuration. In $Q_n$, the nodes identify the communication channels according to their dimensions, i.e., channel $i$ for the neighbor at dimension $i$, for $0 \leq i \leq n - 1$. Broadcasting in $Q_n$ requires $n$ hops. Now, in $HP_n$, broadcasting will require a total of $(n - 1)$ hops because within a Petersen graph only two hops are required to broadcast to all 10 nodes, and $(n - 3)$ hops are sufficient to broadcast to all nodes in $2^{n-3}$ Petersen graphs. This implies that in the $HP_n$ network, we can broadcast a message to 2.5 times additional nodes than in $Q_{n-1}$ with the same number of hops. A broadcast tree is depicted in Figure 2. The broadcasting scheme suggests us to calculate the average distance in $HP_n$.

**Theorem 2:** The number $Z^i_n$ of nodes at distance $i$ from any node in $HP_n$ is $Z^i_n = \binom{n}{i} + 3 \cdot \binom{n}{i-1} + 6 \cdot \binom{n}{i-2}$. The average distance in $HP_n$ is given by $d_n = (2^n + 2^{n-2}) \cdot n / (2^n + 2^{n-2} - 1) \cdot 2 \approx (n/2)$.

**Proof:** $HP_n$ is node-symmetric. Thus, w.l.o.g., let us compute the number of nodes $Z^i_n$ at distance $i$, where $1 \leq i \leq n - 1$, from the node $((0)_{n-3}, 0)$. It is obvious that $Z^i_n = \binom{n}{i} + 3 \cdot \binom{n}{i-1} + 6 \cdot \binom{n}{i-2}$, where $Z^i_n = \binom{n}{i}$ is the number of nodes of distance $i$ from any node in the hypercube $Q_n$. Thus, $Z^i_n = \binom{n}{i} + 3 \cdot \binom{n}{i-1} + 6 \cdot \binom{n}{i-2}$. The average distance $d_n = (\sum_{i=1}^{n-1} i \cdot [\binom{n}{i} + 3 \cdot \binom{n}{i-1} + 6 \cdot \binom{n}{i-2}]) / (2^n + 2^{n-2} - 1) = (2^n + 2^{n-2}) \cdot n / (2^n + 2^{n-2} - 1) = (n/2)$, which is almost the same as in the binary hypercube, $Q_n$.

### 5 EMBEDDINGS INTO HP NETWORKS

This section deals with the mapping of several standard topologies into the HP network. An embedding of a guest graph $G = (V_G, E_G)$ into a host graph $H = (V_H, E_H)$ is a one-to-one mapping of their vertices along with a mapping $\psi: E_G \rightarrow \{\text{paths in } H\}$, from edges in $G$ to paths in $H$. The dilation of an edge, $e \in E_G$, is the length of path $\psi(e)$ in $H$. The dilation of an embedding is the maximum dilation over all edges. The expansion of an embedding is $|V_G| / |V_H|$. An embedding with expansion of one is said to be an embedding of a graph into the optimal sized host [28]. The edge-congestion is the maximum number of edges of $G$ that are routed by $g$ over a single edge of $H$. An embedding is modular, if the embedding of a guest graph $G$ of size or dimension $n + 1$ into the optimum-sized host is an extension of the embedding of $G$ of size $n$ into its optimum-sized host.

#### 5.1 Rings and Linear Arrays

Although the Petersen graph does not contain a Hamiltonian cycle (one which contains all nodes exactly once), it is easy to show that the hyper Petersen network $HP_n$, for $n \geq 4$, is Hamiltonian. Furthermore, unlike the binary hypercubes, rings of odd...
lengths can be embedded into hyper Petersen networks and this property increases its flexibility.

**Theorem 3:** Any ring (hence linear array) of \( l \) nodes, \( 4 \leq l \leq 1.25(2^n) \), is a subgraph of \( HP_n \) for \( n \geq 4 \).

**Proof (by induction):** First, we prove that in \( HP_4 \) one can find a ring \( R_l \) of length \( l \), \( 4 \leq l \leq 20 \), containing the edge \( \{(0, 0), (0, 1)\} \). In the Petersen graph \( P \), there exists a linear array of length \( i \), \( 1 \leq i \leq 10 \), namely \((0, 1, 2, \ldots, 9)\). Thus the sequence of nodes \((0, 0), (0, 1), (1, 1), (1, 0)\) builds a ring of length \( 2l \) in \( HP_n \), where \( 2 \leq l \leq 10 \). Furthermore, one can easily prove that each ring \( R_l \), where \( l \) is odd and \( 5 \leq l \leq 19 \), can be embedded as subgraph in \( HP_4 \) such that \( R_l \) contains the edge \( \{(0, 0), (0, 1)\} \).

Now, as an induction hypothesis assume that in \( HP_n \) each ring of length \( l \), for \( 4 \leq l \leq 2^n + 2^n - 2 \), can be embedded as subgraph such that it contains the edge \( \{(x, 0), (x, 1)\} \) for some node \( x \in Q_{n-3} \). Since each ring \( R_l \) in \( HP_n \) is also contained in \( HP_{n+1} \), we only have to prove the existence of \( R_l \) in \( HP_{n+1} \) with an edge \( \{(x, 0), (x, 1)\}, x \in Q_{n-2}, \) for \( 2^n + 2^n - 2 + 1 \leq l \leq 2^{n+1} + 2^n - 1 \). Thus, \( l = l_1 + l_2 \) with \( 4 \leq l_1, l_2 \leq 2^n + 2^n - 2 \). Now, consider the rings \( R_{l_1} \) embedded in \( 0HP_n \) and \( R_{l_2} \) in \( 1HP_n \). According to the inductive assumption, \( R_{l_1} \) has an edge \( \{(0, x), (0, x)\} \), while \( R_{l_2} \) has an edge \( \{(x', 0), (x', 1)\} \) for some \( x' \in Q_{n-3} \). Since the hypercube is node-symmetric, we use a mapping \( f_1 \), on \( 0Q_{n-2} \), that maps the node \( 0x \) to the node \( 0(0)^{n-3} \) and a mapping \( f_2 \) on \( 1Q_{n-3} \) that maps the node \( 1x' \) to \( 1(0)^{n-3} \). Next, we divide the rings \( R_{l_1} \) and \( R_{l_2} \) at the edges \( \{(0(0)^{n-3}, 0), (0(0)^{n-3}, 1)\} \) and \( \{(1(0)^{n-3}, 0), (1(0)^{n-3}, 1)\} \), respectively, and connect the two rings as shown in Figure 3a).

To guarantee that the obtained ring contains an edge \( \{(y, 0), (y, 1)\} \), for some \( y \in \{0, 1\} \), we use an automorphism of \( P \) that maps an arbitrary edge \( \{(y, i), (y, j)\} \) to the edge \( \{(y, 0), (y, 1)\} \). This is always possible, since the Petersen graph \( P \) is edge-symmetric.

\[\square\]

**5.2 Hypercube Embedding**

**Theorem 4:** A Hypercube \( Q_n \) and additionally a \( Q_{n-2} \) can be embedded in a node- and edge-disjoint manner in \( HP_n \) with load 1, dilation and edge-congestion 2. There also exists an optimal embedding of the incomplete hypercube \( P_{n-2} \) into the optimum sized host \( HP_n \) with load 1 and dilation and edge-congestion 2.

**Proof:** Figure 3b) illustrates the embedding of \( Q_3 \) in the Petersen graph \( P \).

Then we continue with two copies of \( Q_3 \) embedded in \( P^0 \) and \( P^1 \), respectively, and connect them by combining the corresponding nodes in \( Q_4 \). Assume \( Q_{n-1} \) is embedded in \( HP_{n-1} \). Then two copies of \( Q_{n-1}'s \) embedded in \( 0HP_{n-1} \) and \( 1HP_{n-1} \) are combined through the corresponding nodes to yield \( Q_n \) in \( HP_n \). Since the nodes \( (x, 0) \) and \( (x, 8) \), where \( x \in \mathbb{Z}_2^{n-3} \), are not used so far, they can now be utilized to embed an additional \( Q_{n-2} \) in a node- and edge-disjoint fashion with load 1, dilation and edge-congestion 2.

To show the embeddings of the incomplete hypercubes, consider the embedding \( (f_3, g_3) \) of \( P_1 \) in \( P \) as in Figure 3c). Here \( g_3 \) is indicated with dashed.
lines and white nodes, as far as the edge is not mapped to an edge in the host. The nodes v in P are labelled with their image-nodes f3(v) in P. Now, I_{n-2} = F^1 \times Q_{n-3} can be embedded in HP_n = Q_{n-3} \times P using the node-mapping f^n(\langle xy \rangle) = (y, f_3(x)), where y \in Z_{n-3}^2 and x \in \{0(*), 100(*), \}, and (f_3, g_3) is the embedding defined above. We can additionally prove that this embedding is optimal, that is I_{n-2} cannot be embedded with adjacency preserving in HP_n. Both HP_n and I_{n-2} have equal-sized node- and edge-set. Thus, if I_{n-2} is a subgraph of HP_n, the two graphs would be isomorphic. This is a contradiction, since I_{n-2} is bipartite, but HP_n is not.

5.3 Mesh Embedding

Theorem 5: A 2^i \times 2^i \cdot 10 mesh, for 0 \leq i + j \leq n - 3, is a subgraph of HP_n. A 2^n-2 \times 5 mesh (or 3 \cdot 2^i \times 3 \cdot 2^i mesh), for i + j = n - 3, can be embedded in HP_n with dilation and edge-congestion 2 and expansion 1 (or 10/9).

Proof: We use the k-bit binary reflected Gray codes defined recursively as \sigma_i = \{0, 1\}, \sigma_k = \{g^0_k, \ldots, g^k_{k-1}\}, and \sigma_{k+1} = \{g^0_k, 1g^0_k, 1g^1_k, 0g^1_k, \ldots, 1g^k_{k-1}, 0g^k_{k-1}\}. A 2^i \times 2^i \cdot 10 mesh can be embedded as subgraph in the HP_n network using the binary reflected Gray codes \sigma_i, and \sigma_j, and the fact that a linear array of length 10 is a subgraph of P.

Figures 4a) and b) illustrate a 3 \times 3 mesh and a 2 \times 5 mesh obtained from the Petersen graph. Although it is an optimal sized embedding, the dilation is two. Figure 4a) and b) form the basic building block for our construction of meshes in higher dimensional hyper Petersen networks. Using the embedding of a 2 \times 5 mesh in P and the binary reflected Gray code \sigma_{n-3}, a 2^{n-2} \times 5 mesh can be embedded in HP_n with expansion 1 and dilation and edge-congestion 2. The embedding of a 3 \times 3 mesh in P together with the binary reflected Gray code \sigma_j, and \sigma_j leads to an embedding of a 3 \cdot 2^i \times 3 \cdot 2^i mesh in HP_n with expansion 10/9 and dilation and edge-congestion 2.

For illustration, let us construct a 6 \times 6 mesh (see Figure 4c)) in the HP_5 network. We first embed 3 \times 3 meshes in Petersen graphs P^{10} and P^{11} which differ by one bit in dimension 0, and then take vertical mirror reflection (we call it vertical swap) of the mesh formed in P^{10}. Now join nodes along the leftmost column of the mesh in P^{11} with the corresponding nodes along the rightmost column of the mesh contained in P^{10}. This procedure forms a mesh of order 3 \times 6. Again considering the Petersen graphs P^{10} and P^{11}, we similarly construct a 3 \times 6 mesh. This mesh is then swapped horizontally such that the nodes along the bottom row of the meshes obtained from P^{10} and P^{11} are now directly above their corresponding nodes along the uppermost row of the meshes obtained from P^{10} and P^{11}, respectively. This yields a mesh of order 6 \times 6.

The following subsections describe the embeddings of tree-related structures into the HP topology. The guest networks include complete binary trees, X-trees, full-ringed binary trees, tree-machines, hypertrees and pyramids into the HP topology.

5.4 Complete Binary Trees

Complete binary trees are useful computational structures both for broadcasting and applications in dictionary and database operations [14].

A complete binary tree CBT(n) = (V_n, E_n) of height n is defined by V_n = \{x_j, \ldots, x_0 \mid x_j \in Z_n, 0 \leq k \leq j, 0 \leq j \leq n - 1\} \cup \{\epsilon\} and E_n = \{\{x, xa\} \mid x \in Z_n, a \in Z_n, 0 \leq a \leq n - 1\}, where \epsilon is the empty string. The node \epsilon is called the root of the tree at
level 0, and a node \(x = x_i \ldots x_0\) is at level \(i + 1\). Let \(\text{MetaCBT}(n)\) denote the tree \(\text{CBT}(n)\) with an additional root, the so-called metaroot (cf. Figure 5a)). In this subsection, an optimum sized complete binary tree \(\text{CBT}(n)\) is embedded as a subgraph in \(\mathcal{P}_{n+1}\).

**Theorem 6:** A complete binary tree \(\text{CBT}(n)\) of height \(n\) is a subgraph of \(\mathcal{P}_{n+1}\).

**Proof:** Figure 5a) shows that \(\text{MetaCBT}(2)\) is a subgraph of the Petersen graph \(\mathcal{P}_n\). Due to the symmetry and degree of \(\mathcal{P}_n\), for each edge \(\{v, w\}\) of the Petersen graph, there exists exactly one \(\text{MetaCBT}(2)\) in \(\mathcal{P}_n\) with \((x, v)\) as the metaroot and \((x, w)\) as the root, if we additionally demand that each left child must have a smaller suffix-label than the corresponding right child.

Let \(\text{MetaCBT}(n)\) denote the tree \(\text{CBT}(n)\) with an additional root, the so-called metaroot (cf. Figure 5a)). In this subsection, an optimum sized complete binary tree \(\text{CBT}(n)\) is embedded as a subgraph in \(\mathcal{P}_{n+1}\).

**Theorem 6:** A complete binary tree \(\text{CBT}(n)\) of height \(n\) is a subgraph of \(\mathcal{P}_{n+1}\).

**Proof:** Figure 5a) shows that \(\text{MetaCBT}(2)\) is a subgraph of the Petersen graph \(\mathcal{P}_n\). Due to the symmetry and degree of \(\mathcal{P}_n\), for each edge \(\{v, w\}\) of the Petersen graph, there exists exactly one \(\text{MetaCBT}(2)\) in \(\mathcal{P}_n\) with \((x, v)\) as the metaroot and \((x, w)\) as the root, if we additionally demand that each left child must have a smaller suffix-label than the corresponding right child.

Let \(\text{MetaCBT}(n)\) denote the tree \(\text{CBT}(n)\) with an additional root, the so-called metaroot (cf. Figure 5a)). In this subsection, an optimum sized complete binary tree \(\text{CBT}(n)\) is embedded as a subgraph in \(\mathcal{P}_{n+1}\).

**5.5 X-trees and Full-Ringed Binary Trees**

An X-tree of height \(n\), denoted by \(X(n)\), consists of a complete binary tree \(\text{CBT}(n)\) of height \(n\) with additional edges connecting the vertices at the same level [17].

The network \(X(n)\) is a subgraph of the full-ringed binary tree of height \(n\), denoted by \(\text{FRBT}(n)\), which is a \(\text{CBT}(n)\) such that all nodes at each level are connected as a cycle. Figure 6 depicts \(X(3)\) and \(\text{FRBT}(3)\).

**Theorem 7:** There exists an optimal modular embedding \((f_n, g_n)\) of an X-tree (and full-ringed binary tree) of height \(n - 1\) into \(\mathcal{P}_n\) with load 1, dilation and edge-congestion 2.

**Proof:** The following construction defines the node mapping \(f_n, g_n\). An X-tree of height 2 with an additional metaroot can be embedded with load 1, edge-congestion 2, and dilation 2 into \(P^0\) (cf. Figure 7a)). This
holds also for FRBT(2), since \{2, 3\} is an edge in the Petersen graph \(P\). The dashed edge \([x, y]\) in Figure 7a) means that the edge between the two nodes \(x\) and \(y\) is mapped to a path of length 2 in \(P\), where the intermediate node is indicated by an arrow. Thereby \(P^0\) denotes the Petersen graph with prefix-label 0 in \(HP_n\).

When the embedding of X(2 + i + 1), for \(0 \leq i \leq n - 4\), is constructed, one instance of an embedded X(2 + i) is connected with the mirror image (by vertical reflection) of another instance of an embedded X(2 + i), where an offset-value of \(\alpha_i = 2^i\) is added to the prefix-labels. This construction is depicted in Figure 8a), where \(m(G)\) denotes the mirror image of the graph \(G\) by vertical reflection and \(f'(X(2 + i))\) is the embedding of \(X(2 + i)\), where the offset \(2^i\) is added to the prefix-labels. The root of \(X(2 + i + 1)\) is placed at the position of the metaroot of \(f(X(2 + i))\) and the metaroot of \(X(2 + i)\) is placed at the position of the metaroot of \(m(f'(X(2 + i)))\).

This implies for any two nodes \(v_1, v_2 \in X(2 + i + 1)\) with level \((v_1) = \text{level}(v_2) = l\) and \(v_1\) (or \(v_2\)) is the leftmost (or rightmost) node in level \(l\), that the edge \([v_1, v_2]\) is mapped to an edge in \(HP_n\).

The edge routing \(g_n\) is given as:

\[
g_n([x, y]) =
\begin{cases}
\{\hat{f}_n(x), \hat{f}_n(y)\} & \text{if } \text{dist}_{HP_n}(f_n(x), f_n(y)) = 1 \\
(f_n(x), f_n(parent(x), f_n(y)) & \text{if } f_n(x), f_n(y) \text{ differ only in the suffix-label} \\
\end{cases}
\]

\[
(f_n(x), (\Pi^i(f_n(x)), \Pi^i(f_n(y))) & \text{if } f_n(x), f_n(y) \text{ differ in the prefix- and the suffix-labels} \\
\]

\[
(f_n(x), (\Pi^i(f_n(x))(i + 1), \Pi^i(f_n(x)), f_n(y)) & \text{if } f_n(x), f_n(y) \text{ differ in bit } i \text{ and } i + 1 \text{ of the prefix-label}. \\
\]

Thereby, \(\Pi^i : N^k \rightarrow N\), for \(1 \leq i \leq k\), denotes the \(i\)th projection of \(N\) such that \(\Pi^i(n_1, \ldots, n_k) = n_i\). An example of this embedding is depicted in Figure 7b). There, each subtree of height two denoted by a triangle with label \(i\) stands for the embedding of a CBT(2) in \(P_i\) as in Figure 7a).

It is easy to see that the embedding is optimal. Let us assume on the contrary that an \(X(h)\), for \(h \geq 1\), is a subgraph of \(HP_n\). Then \(HP_n\) must contain at least one cycle of length 3 as subgraph, which is a contradiction.

**Remark:** \(\hat{f}_n\) does not use the host nodes \([x, j] \mid x \in \mathbb{Z}_2^{-3}, j \in \{1, 4\} \cup \{(1)^{n-3}, 0\}\). These \(2^{n-2} + 1\)
nodes of $HP_n$ can be used to embed another $X(n - 3)$ which is node- and edge-disjoint from $\hat{f}_n(X(n - 1))$. An example is illustrated in Figure 8b).

5.6 Hypertrees

A hypertree, $HT(n)$, of height $n$ is a complete binary tree $CBT(n)$ with additional links between the nodes at each level of the tree [18]. Assume that the nodes are numbered in breadth-first search manner (in binary encoding) as in Figure 9a). Let $r = z + 1$, where $z$ denotes the number of consecutive trailing zeros in the binary encoded level number. The specific bit number $b$ which is affected by the hypercube connections at level $m$ can be determined by $b = (m/2^r) + 1/2$. Thereby, the bits are counted from the left, such that the most significant bit in the address is counted as bit 0. Figure 9a) shows a hypertree $HT(3)$.

**Theorem 8:** There exists an optimal modular embedding of a hypertree $HT(n)$ of height $n$ into $HP_{n+1}$ with load 1, dilation 2 and edge-congestion 2.

**Proof:** We prove this theorem by induction on the height $h$ of the tree $HT(h)$. Let $\tilde{HT}(h)$ be the graph consisting of a $CBT(h)$ with additional links between the nodes $x$ numbered by the bitstrings (of length $i + 1$) of $2^i$ to $2^{i+1} - 1$ in level $i$, $0 \leq i \leq h$. Each node $x$ in level $i$ is connected with all nodes $x(k)$, $0 \leq k < [\log_2 i]$ in level $i$. Clearly, $HT(h)$ is a subgraph of $\tilde{HT}(h)$. We even prove that the graph $\tilde{HT}(n)$ can be embedded in $HP_{n+1}$ with the claimed cost measures. For a modular embedding (in each step of which an entire level of leaves is inserted), such a hypertree makes only sense when it grows from the root to the leaves. Accordingly, the mapping starts with the root of $\tilde{HT}(h)$ mapped to $(2^{n-3} - 1, 0)$. For level $l = 0$ to $n - 4$, the left (or right) son of an already embedded node $v$ in level $l$ with $f(v) = (x, 0)$ is mapped to the node $(x - 2^{n-4-l}, 0)$ or $(x +...
HYPER PETERSEN NETWORKS

5.7 Tree Machines

A tree machine of dimension \( n \), denoted as \( TM(n) \), consists of two \( CBT(n) \)'s connected back to back along the common leaves. The dark lines in Figure 10a) depict \( TM(2) \).

**Theorem 9:** There exists a modular embedding of \( TM(n) \) into \( HP_{n+2} \) for all \( n \geq 3 \) (into \( HP_3 \) for \( n \leq 2 \)) with load 1, dilation 2, edge-congestion 2 and expansion \( 1 \leq (2^{n+2} + 2^n)/(2^{n+1} + 2^n - 2) \leq 2 \).

**Proof:** For the embedding of \( TM(2) \) into \( HP_3 \), the mapping in Figure 10a) is used, where the edge between \((0, 5)\) and \((0, 1)\) is routed over node \((0, 9)\) and \(2^{n-4-i}, 0\), respectively. In level \( l = n - 3 \), the node \( v \) mapped to \((x, 0)\) has as left subtree the tree \( \tilde{HT}(2) \) embedded into \( P^x \) depicted in Figure 9b) (without the additional root) and as right subtree the tree \( \tilde{HT}(2) \) embedded into \( P^{x+1} \).

---

**FIGURE 8** a) Embedding of \( X(2 + i + 1) \) b) Embedding of an additional \( X(3) \) into \( HP_6 \).

**FIGURE 9** a) Hypertree of height 3 b) Embedding of \( HT(2) \) into \( P^{x+1} \).
the edge between \((0, 9)\) and \((0, 4)\) is routed over \((0, 5)\).

For \(2 \leq i \leq n - 1\), we take two instances of the embedded \(TM(i)\), add the off-set \(2^{i-1}\) to the prefix-labels of the second instance, and then join them inserting two roots \((2^{i-1} - 1, 7)\) and \((2^{i-1} - 1, 0)\) according to the decimal encoding of the prefix-labels (cf. Figure 10b). The preceding algorithm describes the node mapping \(f\). Let \((f_n, g_n)\) be the embedding of a \(CBT(n)\) in \(Q_{n+1}\) by inorder numbering of the vertices. For the edge routing \(g\), we define

\[
g((x, y)) = \begin{cases} 
\{f(x), f(y)\} & \text{if dist}_P(f(x), f(y)) = 1 \ldots (1) \\
\{f(x), f(y)\} & \text{if not (1), } \Pi_2(f(x)) = \Pi_2(f(y)) \\
((\Pi_1(f(x)), 5), (\Pi_1(f(x)), 9), (\Pi_1(f(y)), 1)) & \text{if not (1), } \Pi_2(f(x)) = 5, \Pi_2(f(y)) = 1 \\
((\Pi_1(f(x)), 9), (\Pi_1(f(x)), 5), (\Pi_1(f(y)), 4)) & \text{if not (1), } \Pi_2(f(x)) = 9, \Pi_2(f(y)) = 4
\end{cases}
\]

Thereby, \(g_n(e, x) = ((v_1, x), (v_2, x), (v_3, x))\), with \(g_n(e) = (v_1, v_2, v_3)\).

Remarks:

(1) There remain \(2^{n+1} + 2\) nodes of \(HP_{n+2}\) which are not used by the embedding just discussed. These nodes are \(\{(2^n - 1, 0), (2^n - 1, 7)\} \cup \{(x, j) \mid x \in \mathbb{Z}_2^n, y \in \mathbb{Z}_2, j \in \{1, 2, 3, 4, 5, 6, 8, 9\}\}\), and they can be used for embedding either an additional \(TM(n - 1)\) or another complete binary tree \(CBT(n)\). For details, refer to [29].

(2) For \(k = 2, 3\) we can show that \(TM(k)\) is a subgraph of \(HP_{k+2}\). The question whether this approach can be generalized for \(k > 2\) is still open.

### 5.8 Mesh of Trees

The two-dimensional \(2^n \times 2^n\) mesh of trees [13], denoted by \(MT(n, n)\), is the graph whose nodes are all pairs \((x, y)\), where \(x\) and \(y\) are binary strings of length at most \(n\) and at least one of them has length exactly \(n\). The edges in \(MT(n, n)\) connect \((x, y)\) with \((xa, y)\), where \(a \in \{0, 1\}\), if the length of \(x\) is less than \(n\), and \((x, y)\) with \((x, ya)\), if the length of \(y\) is less than \(n\). Figure 11 illustrates \(MT(2, 2)\).

The network \(MT(n, n)\) is obtained from the \(2^n \times 2^n\) mesh by eliminating all mesh edges, erecting a copy of the complete binary tree \(CBT(n)\) along each column, using the column nodes as leaves of the tree, and erecting a copy of \(CBT(n)\) along each row, using the row nodes as leaves of the tree [30].

**Theorem 10:** A mesh of trees \(MT(n, n)\) can be embedded into \(HP_{2n+2}\) with load 1, dilation 2, edge-congestion 2 and expansion \((1.25.2^{n+2}/(2^{n+1} + 2^{n+1} - 2^{n-1} - 1)) = (5/3)\).
Proof: Let \((x, y)\) be a node in \(MT(n, n)\), and \((f_n, g_n)\) denote the dilation-2 embedding of an X-tree \(X(n)\) into \(HP_{n+1}\) as described in Section 5.5. Furthermore, let \((f_n, g_n)\) denote the embedding of \(CBT(n)\) in \(Q_{n+1}\) using inorder-numbering of the nodes. Now, using the notation \(\Pi^\alpha\) as given in Section 5.5 and writing \(\text{dist}(x, y)\) as short for \(\text{dist}_{HP_{n+1}}(x, y)\), the embedding \((f, g)\) of \(MT(n, n)\) into \(HP_{2n+2}\) can be defined as follows.

\[
f((x, y)) = (\Pi^2(f_n(x)), \Pi^2(f_n(y))),
g((x, y), (x', y)) = \\
\begin{cases} 
(f(x, y), f(x', y)) & \text{if dist}(f(x, y), f(x', y)) = 1 \\
(f(x, y), (uf_n(y), v), f(x', y)) & \text{otherwise}
\end{cases}
\]

and

\[
g'((x, y), (x, y')) = \\
\begin{cases} 
g((x, y), (x, y')) & \text{if dist}(f(x, y), f(x', y')) = 1 \\
g((x, y'), (\Pi^2(f_n(x)), z, \Pi^2(f_n(y))), f(x, y')) & \text{otherwise}
\end{cases}
\]

where \(g_n((x, x')) = (f_n(x), (u, v), f_n(x'))\), if nodes \(f_n(x)\) and \(f_n(x')\) are not adjacent in \(HP_{n+1}\) and \(g_n((y, y')) = (f_n(y), z, f_n(y'))\), if \(f_n(y)\) and \(f_n(y')\) are not adjacent in \(Q_{n+1}\).

The required values for load, dilation and edge-congestion of \((f, g)\) follow from those of the subembeddings \((f_n, g_n)\) and \((f_n, g_n)\).

Remarks:

1. The node-embedding \(f\) uses only \(2^{n+1} \cdot (2^{2n+1} - 2^{n+1} - 1)\) nodes of the \(HP_{2n+2}\) network, which has a total of \(2^{2n+2} + 2^{2n}\) nodes. Since the embedding \((f_n, g_n)\) and hence \((f, g)\) do not use the host nodes with suffix-label 1 or 4, and because the node-set of \(MT(n, n)\) is only a subset of \(V_{CBT(n)} \times V_{CBT(n)}\), the following \(2^{2n+1} + 2^{2n}\) nodes of \(HP_{2n+2}\) are not used by \((f, g)\). They are grouped into four subsets: \(S_1 = \{(x(1)^{2^{n+1}}) i \mid x \in \mathbb{Z}_2^{2n+1}, i = 2, 3, 5, 9\}, S_2 = \{(x, j) \mid x \in \mathbb{Z}_2^{2n+1}, j = 0, 6, 7, 8\}, S_3 = \{(x, j) \mid x \in \mathbb{Z}_2^{2n+1}, j \in \{1, 4\}\}, and \(S_4 = \{((1)^{2n+1})^{i} 0 \mid x \in \mathbb{Z}_2^n\}\).

2. The \(2^n \times 2^n\) mesh of trees \(MT(p, q)\) is clearly a subgraph of \(CBT(p) \times CBT(q)\). Since \(HP_n = Q_{n-3} \times \ldots \times Q_0 = Q_n \times Q_{n-1} \times \ldots \times Q_0 \) contains \(CBT(k - 2) \times CBT(n - k - 1)\) as a subgraph for \(2 < k \leq n - 1\) (and hence \(CBT(1) \times CBT(n - 3)\) for \(k = 2\)), it follows immediately that \(HP_n\) contains a mesh of trees \(MT(k - 2, n - k - 1)\) for \(k > 2\) and \(MT(1, n)\) for \(k = 2\). More precisely, we can show that two node- and edge-disjoint instances of \(MT(k - 2, n - k - 1)\) can be found in \(HP_n\), since two node- and edge-disjoint \(CBT(n - 2)\)'s can be embedded as subgraphs in \(Q_n\) by using the subgraph embedding of a double rooted binary tree of height \(n - 1\) in \(Q_n\) (cf. [31]). Particularly, this implies that \(HP_{2n+2}\) contains \(MT(n - 1, n)\) as a subgraph for \(n > 1\) and \(MT(1, 1)\) for \(n = 1\). Whether \(MT(n, n)\) is generally a subgraph of \(HP_{2n+2}\) for all \(n \geq 1\), is still an open question.

5.9 Pyramid Embedding

The pyramid of height \(n\) [13], denoted by \(PR(n)\), is the graph with the node-set \(V_n = \{(i, x, y) \mid 0 \leq i \leq n, 0 \leq x, y \leq 2^i - 1\}\) and edge-set \(E_n = \{(i, x, y), (i+1, x, y) \mid 0 \leq i < n, x \in \{2x, 2x-1\}, y \in \{2y, 2y-1\} \cup \{(i, x, y), (i, u, v) \mid 0 \leq i \leq n, 0 \leq x, y \leq 2^i - 1, (x, y) and (u, v) differ in exactly one coordinate by one\}\). Informally, the pyramid network \(PR(n)\) is a 4-ary complete tree of height \(n\) and the nodes at each level form a square mesh. Clearly, \(P(n)\) has \(N = \Sigma_{i=0}^{n-1} (4^i + 1)\) nodes. Figure 12b) depicts a pyramid of height 2.

Theorem 11: There exists an optimal embedding of \(PR(n)\) into \(HP_{2n+1}\) with load 1, dilation 2, edge-congestion 2, and expansion 1.875.

Proof: Since a cycle of length three is not a subgraph of \(HP_k\) for all \(k \geq 3\), pyramid \(PR(i)\) cannot be embedded as subgraph into \(HP_k\) for all \(k \geq 2i + 1\). Theorem 11 can be proven by recursive construction, using an approach similar to that given in [13]. Figure 12a) shows that \(PR(1)\) with an alternate apex can be embedded in \(P\) with load 1, edge-congestion 2, and dilation 2. For the embedding \((f_n, g_n)\) of \(PR(i)\) into \(HP_{2i+1}\), we claim that:

1. \(f\) has load 1, dilation 2 and edge-congestion 2.
2. \(f\) maps the apex of \(PR(i)\) into a node of \(HP_{2i+1}\), that has an unassigned neighbor,
called the alternate apex, such that at most one edge of PR(i) is routed through the edge connecting the apex and the alternate apex.

(c) the embedding \( \hat{f}(x) = \begin{cases} f_1(\text{apex}) & \text{for } x = \text{apex of } PR(i) \\ f_1(\text{alternate apex}) & \text{for } x = \text{alternate apex of } PR(i) \\ f_1(x) & \text{for } x \neq \text{apex, alternate apex of } PR(i) \end{cases} \) also fulfills the properties (a) and (b).

As illustrated in Figure 12b), the nodes and edges of PR(2) are embedded into HP by the node-mapping \( f_2 \) and the edge routing \( g_2 \). We embed a copy of PR(1) into each of the two subgraphs \( P_0^0 \) and \( P_0^1 \) by the mapping \( f_1 \) and into each of \( P_{10} \) and \( P_{11} \) by \( \hat{f}_1 \). Then as apex of PR(2) we use the alternate apex of PR(1) mapped into \( P_0^0 \). Generally, the pyramid PR(i + 1), for 1 \( \leq i \leq n - 1 \), is embedded into HP as follows.

**Step 1:** HP is partitioned into four copies of HP with the node-sets \( \{(y, i) \mid 0 \leq i \leq 9, x \in \mathbb{Z}_4 \} \), where \( y \in \{00, 01, 10, 11\} \).

**Step 2:** Embed a copy of PR(i) into HP and HP using the embedding \( f_i \), and also embed a copy of PR(i) into HP and HP using \( \hat{f}_i \) as defined above.

**Step 3:** The apex of PR(i + 1) is mapped into the alternate apex of PR(i) embedded into HP, and the edges are routed as sketched in Figure 12c).

The cost measurements claimed in the theorem are easy to verify.

**Remarks:** We can show the procedure of embedding PR(n) into HP does not use node-set \( \mathcal{S} = \mathcal{S}_1 \cup \mathcal{S}_2 \cup \mathcal{S}_3 \), where \( \mathcal{S}_1 = \{(x, i) \mid x \in \mathbb{Z}_4, i \in \{3, 7, 8, 9\}\} \), \( \mathcal{S}_2 = \{(y, z, 0) \mid y \in \mathbb{Z}_4, z \in \mathbb{Z}_2\} \) and \( \mathcal{S}_3 = \{(x, y, 0, 0, 1) \mid x \in \mathbb{Z}_4, z \in \mathbb{Z}_2\} \). Now, the set \( \mathcal{S} \) of nodes can be used to embed two other PR(n)
- 1)'s, three \( PR(n - 2) \)'s and one \( PR(n - 3) \) in a node- and edge-disjoint fashion from the embedding of \( PR(n) \), such that the load is 1, dilation and edge congestion are 2. The details are provided in [29].

For a comprehensive outline of several embeddings on the remaining host networks, readers may refer to [33, 32, 26].

### 6 COMPARISON OF EMBEDDING RESULTS

This section compares the results on the embeddings of various tree-related topologies (or guest graphs) into such host graphs as hyper Petersen networks, binary hypercubes, incomplete hypercubes, de Bruijn graphs and hyper de Bruijn graphs. The results are displayed in Table 2, where the comparison metrics are dilation (\( \varphi \)), edge-congestion (\( \delta \)) or expansion. The results listed under the hyper Petersen networks are derived in this paper. Most of the results on incomplete hypercubes are also due to us [32].

### 7 CONCLUSIONS

In this paper, we have proposed a family of regular and highly symmetric networks, called hyper Petersen networks, and explored their suitability for use in multiprocessor interconnections. Though these networks are derived from a simple modification of the standard hypercube topology, they exhibit distinct advantages over hypercubes and some of their variants in several aspects, especially in the packing density. Besides, various topologies can be efficiently embedded into them. We have constructed

### TABLE II

Comparison of Embedding Results

<table>
<thead>
<tr>
<th>Guest Graph</th>
<th>Hyper Petersen, ( HP_n )</th>
<th>Hypercube, ( Q_n )</th>
<th>Incomplete Cube, ( I_n^p )</th>
<th>de Bruijn, ( DG(n) )</th>
<th>Hyper-de Bruijn, ( HD(m, n) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>CBT(n)</td>
<td>( CBT(n - 1) ) in ( HP_n ) as subgraph [25] (optimal)</td>
<td>( CBT(n - 2) ) in ( Q_n ) as subgraph, ( CBT(n - 1) ) in ( Q_n ) with ( \varphi = 2 ) and ( \delta = 2 ) [31] (optimal)</td>
<td>( CBT(n - 1) ) in ( I_{n+1}^p ) as subgraph [34] (optimal)</td>
<td>( CBT(n - 1) ) in ( DG(n) ) as subgraph [6] (optimal)</td>
<td>( CBT(m + n - 2) ) in ( HD(m, n) ) as subgraph [21], ( CBT(m + n - 1) ) in ( HD(m, n) ) with ( \varphi = 2 ), ( \delta = 2 ) [26]</td>
</tr>
<tr>
<td>X(n)</td>
<td>( X(n - 1) ) in ( HP_n ) with ( \varphi = 2 ), ( \delta = 2 ) (optimal) [35]</td>
<td>( X(n - 1) ) in ( Q_n ) with ( \varphi = 2 ) and ( \delta = 2 ) (optimal) [35]</td>
<td>( X(n - 1) ) in ( I_n^p ) with ( \varphi = 2 ), ( \delta = 2 ) (optimal) [35]</td>
<td>( X(n - 1) ) in ( DG(k) ) with ( \varphi = \Theta(\log n) ), expansion ( \Theta(1) ) (optimal)</td>
<td>( X(n - 1) ) in ( HD(n - 3, 3) ) with ( \varphi = 2 ), ( \delta = 2 ) [33]</td>
</tr>
<tr>
<td>HT(n)</td>
<td>( HT(n - 1) ) in ( HP_n ) with ( \varphi = 2 ) and ( \delta = 2 ) (optimal)</td>
<td>( HT(n - 1) ) in ( Q_n ) with ( \varphi = 2 ), ( \delta = 2 ) (optimal)</td>
<td>( HT(n - 1) ) in ( I_n^p ) with ( \varphi = 2 ) and ( \delta = 2 ) (optimal) [32]</td>
<td>( HT(n) ) in ( de Brijn ) host with ( \varphi = \Theta(1) ), ( \delta = \Theta(n) ) [26]</td>
<td>( HT(n) ) in ( HD(n - 3, 3) ) with ( \varphi = 2 ), ( \delta = 2 ) [33]</td>
</tr>
<tr>
<td>TM(n)</td>
<td>( TM(n) ) in ( HP_{n+2} ) with ( \varphi = 2 ), ( \delta = 2 )</td>
<td>( TM(n) ) in ( Q_{n+2} ) as subgraph [36] (optimal)</td>
<td>( TM(n) ) in ( I_{n+2}^p ) with ( \varphi = 2 ), ( \delta = 2 ) (optimal) [32]</td>
<td>( TM(n) ) in ( DG(n + 2) ) as subgraph [6] (optimal)</td>
<td>( TM(m + n - 2) ) in ( HD(m, n) ) with ( \varphi = 2 ), ( \delta = 2 ) [33]</td>
</tr>
<tr>
<td>MT(n, n)</td>
<td>( MT(n, n) ) in ( HP_{2n+2} ) with ( \varphi = 2 ), ( \delta = 2 )</td>
<td>( MT(n, n) ) in ( Q_{2n+2} ) as subgraph [36] (optimal)</td>
<td>( MT(n, n) ) in ( I_{2n+2}^p ) with ( \varphi = 2 ), ( \delta = 2 ) (optimal) [32]</td>
<td>( MT(n, n) ) in ( DG(2, 2^{2n+2} ) with ( \varphi = 2 ), ( \delta \leq 8 ), load = 2 [37]</td>
<td>( MT(2^p, 2^q) ) subgraph of ( HD(m, n) ) for ( 1 \leq p \leq m - 2, 1 \leq q \leq n - 1 ) [21]</td>
</tr>
<tr>
<td>PR(n)</td>
<td>( PR(n) ) in ( HP_{2n+1} ) with ( \varphi = 2 ), ( \delta = 2 ) (optimal)</td>
<td>( PR(n) ) in ( Q_{2n+1} ) with ( \varphi = 2 ), ( \delta = 2 ) (optimal) [32]</td>
<td>( PR(n) ) in ( I_{2n+1}^p ) with ( \varphi = 2 ), ( \delta = 3 ) (optical) [32]</td>
<td>( PR(n) ) in ( de Brijn ) host with ( \varphi = \Theta(\log V N) ), expansion ( \Theta(1) ), where ( N = (4^{n+1} - 1)/3 ) (optimal) [26]</td>
<td>( PR(n) ) in ( HD(2n - 2, 3) ) with ( \varphi = 2 ), ( \delta = 2 ) [33]</td>
</tr>
</tbody>
</table>
optimal and modular embeddings of rings, meshes, hypercubes, complete binary trees, X-trees, hyper-trees, tree machines, mesh of trees, and pyramids.

Although we have studied only static embeddings in the hyper Petersen networks, they are also candidates for dynamic embeddings of dynamically evolving task graphs. Therefore, another direction of our research involves dynamic embeddings of dynamically growing and shrinking data and algorithm structures such as dynamic trees and grids [38].

Work is in progress in efficiently mapping algorithms on such networks. Future research can be aimed at analyzing the performance and embedding of other topologies in the presence of faults. It would be worthwhile to design and analyze a network using multiply-twisted hypercube [9] as the framework and the Petersen graph as the building block. This will lead to further reduction in the diameter.

Recently, we have designed the folded Petersen cube networks $FPQ_{n,k}$, a generalization of our approach presented in this paper, by applying iterative cartesian products on the Petersen graph [24]. It holds that $HP_n = FPQ_{n-3,1}$ is a special case of the folded Petersen cube. These networks have extremely large packing density without proportionate increase in the node-degree or diameter, yet having maximum connectivity. In fact, an folded Petersen cube network $FPQ_{n,k}$ is a regular and symmetric network consisting of $N = 2^n \cdot 10^k$ nodes with degree $\delta = 3k + n$, while the diameter is only $d = 2k + n$. Furthermore we have recently investigated incomplete folded Petersen networks having an arbitrary number of nodes [39]. Another direction of our research is to generalize the results derived for hyper Petersen or folded Petersen networks for a class of product networks in general [40].

Acknowledgments
We are grateful to the anonymous referees and the Guest Editor for their helpful comments which improved the quality of this paper.

References


Biographies

SAJAL K. DAS received the Ph.D. degree in Computer Science in 1988 from the University of Central Florida, Orlando. Currently he is an Associate Professor of Computer Science at the University of North Texas, Denton, and also a faculty of the Center for Research in Parallel and Distributed Computing. A recipient of Honor Professor Award at UNT for best teaching and scholarly research, he was invited in 1993 to visit The Leonardo Fibonacci Institute in Trento, Italy. His research interests include design and analysis of parallel algorithms, parallel data structures, parallel discrete event simulation, applied graph theory and combinatorics, and multiprocessor interconnection networks. He has published over 50 journal papers in these areas and presented his research work at many international and national conferences. He serves on the Editorial Boards of Parallel Processing Letters (World Scientific Pub) and Journal of Parallel Algorithms and Applications (Gordon and Breach). He has been a member of Program Committees for several international conferences. He is also a member of IEEE Computer Society, Association of Computing Machinery, New York Academy of Sciences, and Sigma Xi.

SABINE OHRING received the Diploma degree in mathematics and computer science in 1991 and the PhD degree in Computer Science in 1994 from the University of Würzburg in Germany. Currently, she is a Research Associate at the Department of Computer Science of the University of North Texas, Denton. Her research interests include graph embeddings, the design and analysis of interconnection networks for multicomputer systems, communication on interconnection networks, parallel data structures and parallel discrete event simulation. She is a member of IEEE Computer Society, Association of Computing Machinery, EATCS, 61 and PARS.

AMIT K. BANERJEE received the B. Tech. degree in Radiophysics and Electronics from Calcutta University in 1985 and the M.S. degree in Computer Science in 1992 from the University of North Texas, Denton. Currently he is working for the National Airport Authority, Guwahati, in India. His research interest is in interconnection networks and multiprocessor architectures.