Journal of Discrete Mathematical Sciences & Cryptography
ISSN 0972-0529 (Print), ISSN 2169-0065 (Online)
Vol. 29 (2026), No. 4, pp. 1695–1706
DOI : 10.47974/JDMSC-2394

Optimal embedding of extended Sierpiński networks into windmill and necklace graphs

P. Leo Joshwa [†]
Department of Mathematics
Hindustan Institute of Technology & Science
Chennai 603103
Tamil Nadu
India

R. Sundara Rajan *
Department of Mathematics
Rajalakshmi Engineering College
Chennai 602105
Tamil Nadu
India

T. M. Rajalaxmi [§]
Department of Mathematics
Sri Sivasubramaniya Nadar College of Engineering
Chennai 603110
Tamil Nadu
India

[†] *ORCID-ID:* 0000-0002-0365-2309
* *ORCID-ID:* 0000-0002-1851-6334
[§] *ORCID-ID:* 0000-0001-9700-8110

[†] *E-mail:* joseleowa18@gmail.com
* *E-mail:* vprsundar@gmail.com (Corresponding Author)
[§] *E-mail:* laxmi.raji18@gmail.com

© Taru Publications

D. Ahima Emilet [‡]
Department of Mathematics
Hindustan Institute of Technology & Science
Chennai 603103
Tamil Nadu
India

Abstract

The Sierpińki network $S(p, q)$ is a hierarchical, self-similar structure with broad applications in multiprocessor interconnection and VLSI design. In this work, we study its enhanced variant, $S^{++}(p, q)$ for $p \geq 2$, $q \geq 3$, focusing on the problem of minimum wirelength embedding into windmill and necklace graphs. These results offer practical advances for hierarchical network layout, contributing to the design of scalable and low-latency parallel computing architectures.

Subject Classification: *05C10, 05C60, 05C62, 05C75.*

Keywords: *Embedding, Wirelength, Extended Sierpiński network S^{++} (p, q), Windmill, Necklace graph.*

1. Introduction

In parallel computing environments, interconnection networks are fundamental as they facilitate communication between processors and links. These systems are often effectively represented using graph structures. Optimizing such networks contributes significantly to improving data flow efficiency, minimizing delay, and enhancing overall performance. Graph embedding is a key approach in this domain, wherein a guest graph is mapped onto a host graph in a way that retains both topological and performance features. This technique is critical in numerous applications such as VLSI physical layout design, parallel processing systems, distributed computing, and biological modeling [1, 2].

A central metric in graph embedding is *wirelength*, which measures the total communication cost in the host graph. Minimizing wirelength directly impacts circuit design, especially in VLSI and interconnection networks. This leads to the edge isoperimetric problem (EIP), an NP-complete problem with rich theoretical and practical implications [3]. The EIP arises in scenarios such as graph partitioning, minimum cut analysis, and layout optimization. Prior studies have explored minimum wirelength embeddings for classical network topologies

[‡] *ORCID-ID:* `0000-0002-3424-2126`

[‡] *E-mail:* `ahi.180492@gmail.com`

including hypercubes, circulant networks, and various tree-based and multipartite structures [4].

Among emerging models, the *extended Sierpiński network* $S^{++}(p,q), p \geq 2,\ q \geq 3$ stands out for its hierarchical structure, recursive construction, and favorable properties such as low diameter, fault tolerance, and efficient routing [5]. Inspired by puzzles like the Towers of Hanoi, these graphs have attracted attention in both theoretical graph theory and applied computer science due to their scalability and structural elegance [6]. They are particularly promising for designing modular, robust interconnection schemes in parallel and distributed systems [7].

This paper investigates the minimum wirelength embedding of $S^{++}(p,q)$ into key host graphs specifically, *windmill* and *necklace* graphs. These host structures are significant in the context of fault-tolerant routing and Hamiltonian resilience, with direct implications for layout design in hardware systems. Section 2 outlines preliminary definitions and the topological features of $S^{++}(p,q)$. Section 3 presents minimum wirelength embedding results into windmill and necklace graphs. Sections 4 and 5 discuss time complexity analysis findings and propose directions for future research.

2. Basic Concepts

We now outline some basic terminology and notation that will be used in the sequel. Each concept is given formally and followed by a short explanation.

Definition 2.1: [1] Consider a graph $G = (V, E)$ and fix an integer l with $1 \leq l \leq |V|$. The maximum subgraph problem (MSP) consists of finding an l-vertex subset $M \subseteq V$ such that the induced subgraph $G[M]$ contains as many edges as possible. The optimal value is written as

$$I_G(l) = \max\{\ |E(G[M])| : M \subseteq V, |M| = l\ \}.$$

In simple terms, $I_G(l)$ tells us how many edges the densest subgraph of size l can have.

Definition 2.2: [8] An embedding of a graph G into another graph H is a one-to-one mapping $\varphi: V(G) \to V(H)$ such that each edge of G is represented by a path in H connecting the images of its endpoints. This ensures that vertices remain distinct and the adjacency structure is preserved through the paths.

Definition 2.3: [8] Suppose $\varphi: G \to H$ is an embedding and let $e \in E(H)$. The congestion of e under φ is the number of edges of G whose image paths make use of e. If $P_\varphi(e')$ denotes the path in H representing an edge $e' \in E(G)$, then

$$C_\varphi(G, H(e)) = \left|\{\ e' \in E(G) : e \in P_\varphi(e')\ \}\right|.$$

In other words, congestion measures the load carried by an edge of H when simulating G inside it.

Definition 2.4: [9] For an embedding $\varphi: G \to H$, the wirelength is the total sum of all edge congestions:

$$WL_{\varphi}(G,H) = \sum_{e \in E(H)} C_{\varphi}(G,H(e)).$$

The *minimum wirelength* is obtained by minimizing over all embeddings:

$$WL(G,H) = \min_{\varphi} WL_{\varphi}(G,H).$$

This value expresses how efficiently G can be arranged inside H with respect to edge usage.

Lemma 2.5: [10] *Consider an edge cut* $S \subseteq E(H)$ *separates* H *into two components* H^1 *and* H^2*. Let* $G^j = G[\varphi^{-1}(V(H^j))]$ *for* $j = 1,2$*, and assume:*

- *Each edge between vertices in* G^j *has no edge in* S*.*
- *Each edge between vertices in* G^1 *and* G^2 *use exactly one edge from* S*.*
- $V(G^j)$ *are optimal subsets.*

Then,

$$C_{\varphi}(S) = \sum_{a \in V(G^1)} \deg_G(a) - 2|E(G^1)| = \sum_{a \in V(G^2)} \deg_G(a) - 2|E(G^2)|.$$

Lemma 2.6: [8] *Let* $\varphi: G \to H$ *be an embedding, and let* $[tE(H)]$ *denote the multiset where each edge appears* t *times. Let* $\{S_y : 1 \leq y \leq k\}$ *be a partition of* $[tE(H)]$*, with each* S_y *satisfying the Lemma 2.5. Then,*

$$WL(G,H) = \frac{1}{t} \sum_{y=1}^{k} C_{\varphi}(S_y),$$

where $C_{\varphi}(S_y) = \sum_{e \in S_y} C_{\varphi}(G,H(e))$*.*

Extended Sierpiński Network $S^{++}(p,q)$: Sierpiński network are recursive, self-similar structures with significant applications in discrete mathematics, parallel computing, and network design. Inspired by the Sierpiński triangle, the classical form $S(p,q)$, introduced by Klavžar and Milutinović, in Figure 1 offers symmetry, scalability, and hierarchical organization, making it suitable for interconnection networks.

To improve robustness and connectivity, Klavžar and Mohar [12] proposed the extended Sierpiński network $S^{++}(p,q)$, which augments $S(p,q)$ with additional edges, see Figure 2. This enhanced topology supports efficient communication and fault tolerance, making $S^{++}(p,q)$ well-suited for VLSI systems and distributed architectures.

Definition 2.1.1: [13] Let $p,q \in \mathbb{N}$ with $p \geq 1$, $q \geq 1$. The vertex set of generalized Sierpiński network $S(p,q)$ is $V(S(p,q)) = \{0,1,\ldots,q-1\}^p$, i.e., all p-tuples over the alphabet $\{0,1,\ldots,q-1\}$. Vertices $(a_1,\ldots,a_p)$ and $(b_1,\ldots,b_p)$ are connected if $\exists$ an index $t \in \{1,\ldots,p\}$ such that:

1. $a_i = b_i \ \forall \ t > i$,
2. $a_t \neq b_t$,
3. $a_i = b_t,\ b_i = a_t \ \forall \ t < i$.

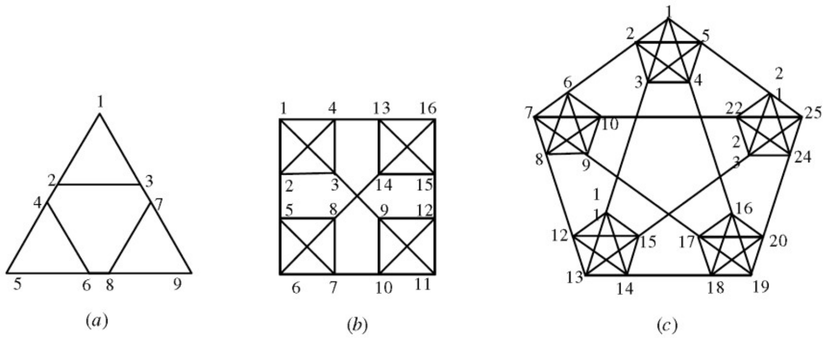

Figure 1
Sierpiński networks

Definition 2.1.2: [5] The extended Sierpiński network $S^{++}(p, q)$ is constructed by augmenting $S(p, q)$ with additional vertices and edges as follows:

- $V(S^{++}(p, q)) = q^p \cup \{q\overline{s} | \overline{s} \in q^{p-1}\}$, where $q^p = \{(a_1, \dots, a_p) | a_i \in \{0, \dots, q-1\}\}$ and $q\overline{s}$ denotes the concatenation of q and a $(p-1)$-tuple $\overline{s} \in q^{p-1}$.
- $E(S^{++}(p, q)) = E(S(p, q)) \cup \left\{\{q\overline{s}, q\overline{t}\} | \{\overline{s}, \overline{t}\} \in \binom{q^{p-1}}{2}\right\} \cup \{\{qi^{p-1}, i^p\} | i \in \{0, \dots, q-1\}\}$.

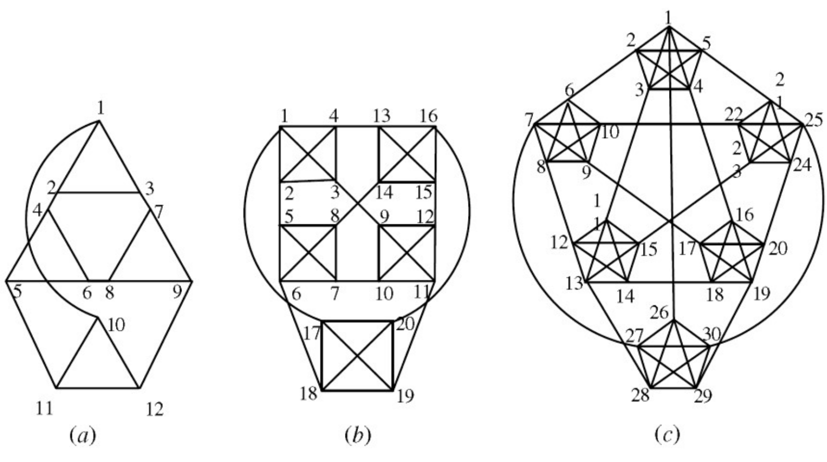

Figure 2
Extended Sierpiński networks

3. Main Results

To determine the optimal wirelength necessary to embed the extended Sierpiski network $S^{++}(p, q)$ into graph structures such as windmill and necklace graphs, we explore its influence on distributed algorithms. Efficient embeddings play a vital role by minimizing communication overhead between nodes. This leads to lower latency and better overall performance. With well-optimized embeddings, we can achieve faster message passing and reduced consumption of system resources in distributed environments.

In this study, we conduct a detailed investigation to compute the exact wirelength necessary for embedding $S^{++}(p,q)$ into structured graphs like windmill and necklace configurations. This analysis plays a key role in understanding how to minimize inter-node communication costs in distributed systems. Ultimately, such optimizations contribute to better algorithmic performance and more efficient resource allocation.

Theorem 3.1: [13] *For all values of* p *and* q, *the set of initial* l*-segments in the lexicographic sequence yields an optimal embedding for* $S(p,q)$.

Theorem 3.2: [11] *Let* G *be* $S^{++}(p,q)$ *and let* l *be an integer that satisfies* $1 \leq l \leq q^p + q^{p-1}$. *For any subset* $M \subseteq \{0,1,\dots,q^p+q^{p-1}-1\}$ *with* $|M| = l$, *the induced subgraph* $G[M]$ *reaches its optimum edges when* $M = \{0,1,\dots,l-1\}$.

Theorem 3.3: [11] *For* $l = \sum_{j=0}^{t} a_j q^j$ *with* $a_j \in \{0,1,\dots,q-1\}$, *we set, for* $j = 1,2,\dots,t$,

$\beta_j = \min\left\{a_j, \left\lceil \frac{q-1}{q^j-1} \sum_{i=0}^{j-1} a_i q^i \right\rceil\right\}; \lambda_l = \frac{1}{2}\sum_{j=0}^{t} a_j\left(q^{j+1} - q + a_j - 1\right) + \sum_{j=1}^{t} \beta_j.$

Then

$$I_G(l) = \begin{cases} \lambda_l & if\ \ 1 \leq l \leq q^p + \frac{1}{q-1}(q^{p-1}-1) \\ \lambda_l + t & if\ \ q^p + \frac{t}{q-1}(q^{p-1}-1) + 1 \leq l \leq q^p + \\ & \frac{t+1}{q-1}(q^{p-1}-1), 1 \leq t \leq q-2 \\ \lambda_l + q - 1 & if\ \ l = q^p + q^{p-1}. \end{cases}$$

3.1 *Circular Necklace Graph*

Let K_{q+1} be the complete graph with $q+1$ vertices, say $v_1, v_2, \dots, v_{q+1}$ and K_{t_y} be a complete graph with t_y vertices, where $1 \leq y \leq q+1$. Define $t_y = q^{p-1}$ for all $1 \leq y \leq q+1$, such that $K_{q+1} \uplus_{y=1}^{q+1} K_{t_y}$ shares a common vertex v_y. The resulting graph is called the circular necklace graph and is denoted by $CN(K_{q+1})$ [14].

Embedding Algorithm A

Input: The extended Sierpiński network $S^{++}(p,q), p \geq 2, q \geq 3$ and the circular necklace graph $CN(K_{q+1})$.

Algorithm: The vertices of $S^{++}(p,q)$ are indexed following lexicographic ordering from 1 to $q^p + q^{p-1}$. The vertices of K_{t_y} in $CN(K_{q+1})$ are labeled in a circular fashion, starting with a vertex in K_{t_y}. The labels are regarded as identifiers of the corresponding vertices. Define an embedding of $S^{++}(p,q)$ into $CN(K_{q+1})$ given by $\varphi(x) = x$.

Output: The algorithm computes the exact wirelength.

Proof of Correctness: Let $S^y = \{(yq^{p-1}, y'q^{p-1}): y \neq y', 1 \leq y, y' \leq q+1\}$ and $S'_y = \{(yq^{p-1}, a): a \in V(K_{t_y}) \backslash \{yq^{p-1}\}, 1 \leq y \leq q+1\}$ and for $S_y^n = \{((y-1)q^{p-1} + n, a) \ : \ a \in V(K_{t_y}) \setminus \{(y-1)q^{p-1} + n\}, 1 \leq y \leq q + 11 \leq n \leq t_y - 1\}$. Then the set $\{S^y, S'_y, S_y^n\}$ Yields a partition $[2E(CN(K_{q+1}))]$.

For each $1 \leq y \leq q+1$, let H^y and $\overline{H^y}$ denote the two connected components obtained by $E(CN(K_{q+1}))\backslash S^y$ where $V(H^y) = \{K_{t_y}\}$. Let G^y and $\overline{G^y}$ denote the preimages of $V(H^y)$ and $V(\overline{H^y})$, respectively, under the map φ. Applying Theorem 3.3 results in G^y being an optimal subset of $S^{++}(p,q)$. Hence, each S^y satisfies the conditions of Lemma 2.5, ensuring that $C_\varphi(S^y)$ is minimized.

Similarly, for each $1 \leq y \leq q+1$, let H'_y and $\overline{H'_y}$ denote the two connected components obtained by $E(CN(K_{q+1}))\backslash S'_y$ where $V(H'_y) = \{K_{t_y} \backslash \{yq^{p-1}\}\}$ then G'_y and $\overline{G'_y}$ denote the preimages of $V(H'_y)$ and $V(\overline{H'_y})$, respectively, under the map φ. Again, G'_y is an optimal subgraph of $S^{++}(p,q)$, satisfying Lemma 2.5.

Now for $1 \leq y \leq q+1$ and $1 \leq n \leq t_y - 1$, removing S_y^n from $E(CN(K_{q+1}))$ results in two components H_y^n and $\overline{H_r^n}$, where $V(H_y^n) = \{(y-1)q^{p-1} + n\}$. Let G_y^n and $\overline{G_y^n}$ denote the preimages of $V(H_y^n)$ and $V(\overline{H_y^n})$, respectively, under the map φ. Applying Theorem 3.3 results in G_y^n being an optimal subset of $S^{++}(p,q)$, meeting the criteria in Lemma 2.5. Hence, S_y^n minimizes $C_\varphi(S_y^n)$, and from Lemma 2.6, the wirelength $WL_\varphi(S^{++}(p,q), CN(K_{q+1}))$ is minimized.

Theorem 3.4: *The wirelength of $S^{++}(p,q)$ into $CN(K_{q+1})$ is followed by*

$$WL(S^{++}(p,q), CN(K_{q+1})) = \frac{1}{2}[(q+1)(q^p + 2q - 2)].$$

Proof: By Lemma 2.5 and 2.6 we have,

$$\begin{aligned} WL(S^{++}(p,q), CN(K_{q+1})) &= \frac{1}{2}\left[\sum_{y=1}^{q+1} \sum_{n=1}^{t_y-1} C_\varphi(S_y^n) + \sum_{y=1}^{q+1} C_\varphi(S_{y'}) + \sum_{y=1}^{q+1} C_\varphi(S^y)\right] \\ &= \frac{1}{2}\left[\sum_{y=1}^{q^p+q^{p-1}-q-1} q + \sum_{y=1}^{q+1} 2(q-1) + \sum_{y=1}^{q+1} 3\right] \\ &= \frac{1}{2}[(q+1)(q^p - q) + 2(q^2 - 1) + q(q+1)] \\ &= \frac{1}{2}[(q+1)(q^p + 2q - 2)]. \end{aligned}$$

3.2 *Windmill*

A windmill graph, is formed by taking $q+1$ copies of the complete graph denoted as $K_{t_y}, 1 \leq y \leq q+1$. Let $t_y = q^{p-1}+1$ for all $1 \leq y \leq q$ and $t_{q+1} = q^{p-1}$ such that $\uplus_{y=1}^{q+1} K_{t_y}$ has one vertex v as common. The resultant graph is called a windmill graph and is denoted by $WM(K)$. The windmill graph is also known as the friendship graph [14].

Embedding Algorithm B

Input: An extended Sierpiński network $S^{++}(p,q)$ where $p \geq 2,\ q \geq 3$, and a windmill graph $WM(K)$.

Algorithm: Assign labels to the vertices of $S^{++}(p,q)$ sequentially from 1 to $q^p + q^{p-1}$ based on the lexicographic order. Label K_{t_y} in $WM(K)$ in a circular manner, starting with vertex 1 and increasing to $q^p + q^{p-1}$ such that $v = q^p + q^{p-1}$. The labels are regarded as identifiers of the corresponding vertices. Define an embedding of $S^{++}(p,q)$ in $WM(K)$ given by $\varphi(x) = x$.

Output: The algorithm produces the exact wirelength corresponding to the embedding.

Proof of correctness : Let $S^y = \{(q^p + q^{p-1}, a) : a \in V(K_{t_y}) \setminus \{q^p + q^{p-1}\}, 1 \leq y \leq q+1\}$ and $S_y^n = \{((y-1)q^{p-1}+n, a) : a \in V(K_{t_y}) \setminus \{(y-1)q^{p-1}+n\}, 1 \leq y \leq q+1 \text{ and } 1 \leq n \leq t_y - 1\}$. Then $\{S^y, S_y^n\}$ Yields a partition of $[2E(WM(K))]$.

For each $1 \leq y \leq q+1$, let H^y and $\overline{H^y}$ denote the two connected components obtained by $E(WM(K)) \setminus S^y$ where $V(H^y) = \{K_{t_y} \setminus q^p + q^{p-1}\}$. Let G^y and $\overline{G^y}$ denote the preimages of $V(H^y)$ and $V(\overline{H^y})$, respectively, under the map φ. Applying Theorem 3.3 results in G^y being an optimal subset of $S^{++}(p,q)$. Therefore, each S^y meets the conditions of Lemma 2.5, ensuring that $C_\varphi(S^y)$ is minimized.

For $1 \leq y \leq q+1$ and $1 \leq n \leq t_y - 1$, removing S_y^n from $E(WM(K))$ yields two components: H_y^n and $\overline{H_y^n}$, where $V(H_y^n) = \{(y-1)q^{p-1}+n\}$. Let G_y^n and $\overline{G_y^n}$ denote the preimages of $V(H_y^n)$ and $V(\overline{H_y^n})$, respectively, under the map φ. Applying Theorem 3.3 results in G_y^n being an optimal subset of $S^{++}(p,q)$. Each S_y^n satisfies the conditions of Lemma 2.5, ensuring that $C_\varphi(S_y^n)$ is minimized. Thus, by Lemma 2.5, we conclude that $WL_\varphi(S^{++}(p,q), WM(K))$ is minimized.

Theorem 3.5: *The wirelength of* $S^{++}(p,q)$ *into* $WM(K)$ *is given by*

$$WL(S^{++}(p,q), WM(K)) = \frac{1}{2}\left[q^{p+1} + q^p - q + \sum_{y=1}^{q+1} q(t_y - 1) - 2I_G(t_y - 1)\right].$$

Proof: By Lemma 2.5 and 2.6 we have,

$$\begin{aligned} WL(S^{++}(p,q), WM(K)) &= \frac{1}{2}\left[\sum_{y=1}^{q+1}\sum_{n=1}^{t_y-1} C_\varphi(S_y^n) + \sum_{y=1}^{q+1} C_\varphi(S^y)\right] \\ &= \frac{1}{2}\left[\sum_{y=1}^{q^p+q^{p-1}-1} q + \sum_{y=1}^{q+1} q(t_y-1) - 2I_G(t_y-1)\right] \\ &= \frac{1}{2}\left[q^{p+1} + q^p - q + \sum_{y=1}^{q+1} q(t_y-1) - 2I_G(t_y-1)\right]. \end{aligned}$$

3.3 *Star Necklace Graph*

Consider $K_{1,q+1}$ as a star graph comprising $q+1$ nodes, and let K_{t_y} represent a complete graph with t_y vertices. Define $t_y = q^{p-1}$ for each $1 \le y < q+1$ and $t_{q+1} = q^{p-1} - 1$, such that $K_{1,q+1} \uplus_{y=1}^{q+1} K_{t_y}$ shares a unique common vertex v_y. This construction forms a complete star necklace graph, denoted by $SN(K_{1,q+1})$ [14].

Embedding Algorithm C

Input: The extended Sierpiński network $S^{++}(p,q)$, with $p \ge 3$ and $q \ge 3$, and the star necklace $SN(K_{1,q+1})$.

Algorithm: The vertices of $S^{++}(p,q)$ are indexed following lexicographic ordering, ranging from 1 to $q^p + q^{p-1}$. For each K_{t_y} in $SN(K_{1,q+1})$, assign labels clockwise manner, starting from an arbitrarily chosen vertex within K_{t_y} such that $q^p + q^{p-1}$ as a hub vertex. The labels are regarded as identifiers of the corresponding vertices. Define an embedding of $S^{++}(p,q)$ into $SN(K_{1,q+1})$ given by $\varphi(x) = x$.

Output: The algorithm produces the exact wirelength corresponding to the embedding.

Proof of correctness : Let $S_y^l = \{(K_{t_y}, q^p + q^{p-1}) : 1 \le y \le q+1 l = 1,2\}$. Let $S'_y = \{(K_{t_y}, a) : a \in V(K_{t_y}) \backslash \{K_{t_y}\}, 1 \le y \le q+1\}$ and for $S_y^n = \{((y-1)q^{p-1} + n, a) : a \in V(K_{t_y}) \backslash (y-1)q^{p-1} + n, 1 \le y \le q+1 \text{and} 1 \le n \le t_y - 1\}$. Then $\{S_y^l, S'_y, S_y^n\}$ yields a partition of $[2E(SN(K_{1,q+1}))]$.

For each $1 \le y \le q+1$ and $l = 1,2$, let H_y^l and $\overline{H_y^l}$ denote the two connected components obtained by $E(SN(K_{1,q+1})) \backslash S_y^l$, where $V(H_y^l) = \{K_{t_y}\}$. Let G_y^l and $\overline{G_y^l}$ denote the preimages of $V(H_y^l)$ and $V(\overline{H_y^l})$, respectively, under the map φ. Applying Theorem 3.3 results in G_y^l being an optimal subset of $S^{++}(p,q)$. Therefore, each S_y^l meets the conditions of Lemma 2.5, ensuring that $C_\varphi(S_{yn})$ is minimum.

Similarly, for each $1 \le y \le q+1$, let H'_y and $\overline{H'_y}$ denote the two connected components obtained by $E(SN(K_{1,q+1})) \backslash S'_y$ where, $V(H'_y) = \{K_{t_y}\{v_y\}\}$. Let G'_y and $\overline{G'_y}$ denote the preimages of $V(H'_y)$ and $V(\overline{H'_y})$, respectively, under the map φ. By Theorem 3.3. G'_y is an largest subgraph of $S^{++}(p,q)$. Therefore, each S'_y meets the conditions of Lemma 2.5, ensuring that $C_\varphi(S'_y)$ is minimum.

For $1 \leq y \leq q+1, 1 \leq n \leq t_y - 1$, removing S_y^n from $E(SN(K_{1,q+1}))$ yields two components: H_y^n and $\overline{H_y^n}$, where $V(H_y^n) = \{(y-1)q^{p-1} + n\}$. Let G_y^n and $\overline{G_y^n}$ denote the preimages of $V(H_y^n)$ and $V(\overline{H_y^n})$, respectively, under the map φ. By Theorem 3.3. G_y^n is an largest subgraph of $S^{++}(p,q)$. Therefore, each S_y^n meets the conditions of Lemma 2.5. Hence, S_y^n meets the conditions of Lemma 2.5, ensuring that $C_\varphi(S_y^n)$ is minimized.

Theorem 3.6: *The wirelength of the embedding of $S^{++}(p,q)$ into $SN(K_{1,q+1})$ can be expressed as*

$$WL(S^{++}(p,q), SN(K_{1,q+1})) = \frac{1}{2}[q(q^{p-1} + q^p - q - 2) + \sum_{y=1}^{q+1} q(t_y - 1) - 2I_G(t_y - 1)] + \frac{1}{2}[\sum_{y=1}^{2} \sum_{y=1}^{q+1} q(t_y) - 2I_G(t_y)].$$

Proof. Using Lemmas 2.5 and 2.6, the wirelength is computed as follows

$$WL(S^{++}(p,q), SN\left(K_{1,q+1}\right)$$
$$= \frac{1}{2}\left[\sum_{y=1}^{q+1} \sum_{n=1}^{t_y-1} C_\varphi(S_y^n) + \sum_{y=1}^{q+1} C_\varphi\left(S_y'\right) + \sum_{y=1}^{q+1} \sum_{n=1}^{2} C_\varphi(S_{yn})\right]$$
$$= \frac{1}{2}\left[q(q^{p-1} + q^p - q - 2 + \sum_{y=1}^{q+1} q\left(t_y - 1\right) - 2I_G\left(t_y - 1\right)\right]$$
$$+ \frac{1}{2}\left[\sum_{y=1}^{2} \sum_{y=1}^{q+1} q\left(t_y\right) - 2I_G(t_y)\right].$$

4. Time Complexity Analysis

Input: The extended Sierpiński network $S^{++}(p,q)$, with parameters $p \geq 2$ and $q \geq 3$, and the circular necklace $CN(K_{q+1})$.

Algorithm: Embedding Algorithm A.

Output: Total computational time required to determine the wirelength of $S^{++}(p,q)$ when embedded into $CN(K_{q+1})$.

Analysis: The graph $S^{++}(p,q)$ contains $q^p + q^{p-1}$ vertices. Assigning labels to all vertices requires $q^p + q^{p-1}$ time units. Using Algorithm A, the total number of edge cuts is $q^p + q^{p-1} + q + 1$, and since processing each cut takes constant time, the edge congestion calculations require an additional $2(q^p + q^{p-1} + q + 1)$ time units. Finally, by applying Lemma 2.6, computing the total wirelength takes 1 additional time unit. Therefore, the overall time complexity for determining the minimum wirelength of $S^{++}(p,q)$ into $CN(K_{q+1})$ is $= O(q^p + q^{p-1})$, Wich is linear. This approach can similarly be extended to compute the wirelength for embedding extended Sierpiński graphs into other topologies, such as the star necklace or windmill networks, while maintaining linear time complexity.

5. Concluding Remarks

This work establishes the exact wirelength for embedding extended Sierpiński graphs $S^{++}(p,q)$, with $p \geq 2$ and $q \geq 3$, into several key interconnection topologies: star necklace, circular necklace, and windmill. These results contribute to the efficient layout and analysis of hierarchical networks in parallel computing and VLSI design. Future research will explore embeddings of $S^{++}(p, q)$ in other structurally rich host graphs to further evaluate their applicability in fault-tolerant and scalable network architectures.

Acknowledgment

The work is supported by the I-DAPT HUB Foundation, IIT (BHU) Varanasi, for funding under Technology Innovation Hub (TIH), Department of Science & Technology (DST), Government of India, with No. I-DAPT / IIT (BHU) / 2025-26 / Project Sanction / 30.

References

[1] S. L. Bezrukov, "Edge isoperimetric problems on graphs," in *Graph Theory and Combinatorial Biology,* vol. 7, pp. 157-197 (1999).

[2] A. Khobragade, R. Mahajan, H. Langi, R. Mundhe, and S. Ghumbre, "Effective negative triplet sampling for knowledge graph embedding," *Journal of Information and Optimization Sciences,* vol. 43, no. 8, pp. 2075-2087 (2022).

[3] M. R. Garey and D. S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness. San Francisco, CA, USA: Freeman (1979).

[4] R. S. Rajan, T. M. Rajalaxmi, J. B. Liu, and G. Sethuraman, "Wirelength of embedding complete multipartite graphs into certain graphs," *Discrete Applied Mathematics,* vol. 280, pp. 221-236 (2020).

[5] P. L. Joshwa, R. S. Rajan, T. M. Rajalaxmi, and I. N. Cangul, "Embedding of extended Sierpiński network S^{++} (k,m) into certain trees," *RAIRO - Operations Research*, vol. 59, no. 4, pp. 2279-2301 (2025).

[6] G. K. Nandini, S. Klavžar, T. M. Rajalaxmi, and R. S. Rajan, "A note on eccentricity based topological indices of honeycomb, oxide and 2-power interconnection networks," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 26, no. 7, pp. 231-253 (2021).

[7] S. Arulanand, R. S. Rajan, S. Prabhu, and S. Stephen, "Certain domination numbers for Cartesian product of graphs," *Journal of Discrete Mathematical Sciences and Cryptography,* vol. 27, no. 3, pp. 1045-1058.

[8] P. Manuel, I. Rajasingh, B. Rajan, and H. Mercy, "Exact wirelength of hypercubes on a grid," *Discrete Applied Mathematics,* vol. 157, no. 7, pp. 1486-1495 (2009).

[9] P. L. Joshwa, R. S. Rajan, and T. M. Rajalaxmi, "Maximum subgraph and wirelength optimization in cyclic bipartite networks $G_{k,3}$," *International Journal of Parallel, Emergent and Distributed Systems*, pp. 1-21 (2026).

[10] R. S. Rajan, A. B. Greeni, and P. L. Joshwa, "Maximum subgraph problem and minimum linear arrangement of generalized Sierpiński networks," *Journal of Graph Algorithms and Applications*, vol. 27, no. 9, pp. 767-782 (2023).

[11] P. L. Joshwa, R. S. Rajan, and T. M. Rajalaxmi, "Maximum subgraph and wirelength analysis of extended Sierpiński networks in parallel computing," *Discrete Applied Mathematics*, vol. 383, pp. 367-382 (2026).

[12] S. Klavžar and B. Mohar, "Crossing numbers of Sierpiński-like graphs," *Journal of Graph Theory*, vol. 50, no. 3, pp. 186-198 (2005).

[13] L. H. Harper, "The edge-isoperimetric problem on Sierpiński graph: Final resolution," arXiv preprint arXiv:1802.08355 (2018).

[14] I. Rajasingh, B. Rajan, and R. S. Rajan, "Embedding of hypercubes into necklace, windmill and snake graphs," *Information Processing Letters*, vol. 112, no. 12, pp. 509-515 (2012).

Received May, 2025