Discrete Mathematics Letters
www.dmlett.com

Discrete Math. Lett. **3** (2020) 50–56

A linear time algorithm for embedding chord graphs into certain necklace and windmill graphs

T. M. Rajalaxmi[1], N. Parthiban[2], Joe Ryan[3], A. Arul Shantrinal[4], R. Sundara Rajan[4,*]

[1]*Department of Mathematics, SSN College of Engineering, Chennai 603 110, India*
[2]*Department of Computer Science and Engineering, SRM Institute of Science and Technology, Chennai 603 203, India*
[3]*School of Electrical Engineering and Computing, University of Newcastle, NSW 2308, Australia*
[4]*Department of Mathematics, Hindustan Institute of Technology and Science, Chennai 603 103, India*

(Received: 17 January 2020. Received in revised form: 23 April 2020. Accepted: 4 May 2020. Published online: 8 May 2020.)

Abstract

Wirelength is a salient feature to authenticate the quality of an embedding of a guest graph into a host graph and is used specifically in Very Large Scale Integration (VLSI) layout designs. The chord graph is an influential topology in the sphere of peer-to-peer networks. Thus, it is interesting to study the embedding of chord graphs into networks. In this paper, we have computed the exact wirelength of chord graphs into necklace and windmill graphs. Further, we have developed a linear time algorithm to compute the wirelength.

Keywords: embedding; wirelength; chord graph; necklace graph; windmill graph.

2020 Mathematics Subject Classification: 05C12, 05C60, 05C85.

1. Introduction

In the implementation of any algorithm, it is necessary that the code should be compilable and executable on any machine. However, it is far more complicated in the case of parallel algorithms as the properties of parallel machines highly depend on their interconnection structure [24]. Therefore the implementation of algorithms is often restricted to certain classes of networks. In order to overcome this dependency, it is necessary to emulate one network by another by embedding one into another. Efficient parallel algorithms existing for some source architecture are implemented on a target architecture by embedding the source into the target. However the efficiency level determined by certain cost measures associated with an embedding.

The aspects of an embedding can be measured using certain cost criteria. Congestion and the wirelength rank among the most significant criteria [13]. The cardinality of a largest set of edges from the guest graph that are charted on a single edge from the host graph is explained as the preceding one. Consequently, the primary point here is many problems may arise while we are facing a large congestion. Furthermore, it also leads to the problem of circuit switching, longest communication delay as well as the existence of uncontrolled noise itself. It represents the effects of packet loss or blockage of new connections in data networking. Accordingly, the minimum congestion is of the utmost importance in network embedding [16]. The other measure, wirelength, is the sum of the congestions. These sources inclusive of interest ranging from VLSI design to data structures and more [16]. In recent times, graph embeddings have been carefully scrutinized for a variety of networks such as circulant networks and grids [6, 15], windmill and necklace graphs [10]. Although there are myriad results and discussions found on the wirelength problem, the approximate results and the estimation of lower bounds are dealt by most of them [5, 9, 20]. The embedding in the present paper provides exact wirelength.

An overlay network can be defined as a computer network that is constructed on the top of another network. Virtual or logical links provide connections to the nodes in the overlay as each node communicates to a path, reasonably through many other forms of physical links existing in the underlying network. For instance, overlay networks includes distributing systems of cloud computing, peer-peer networks and client-server applications. Chord graph introduced by Stocia *et al.* [23], is a structured peer-to-peer architecture based on Distributed Hash Tables (DHTs) [25]. In [18], an overlay network is designed as the chord graph. In addition hypercubes, generalized hypercubes are subgraphs of chord graphs.

The notion of the concept that is essential is defined in the next section and remembrance two key lemmas for embedding algorithm is also focused. The two key lemmas are as follows: Modified Congestion Lemma and the Partition Lemma. In Section 3, we obtained the accurate wirelength of chord graphs into necklace and windmill graphs. In Section 4, we have shown a linear time algorithm to compute the wirelength. Conclusion and future work are given in Section 5.

*Corresponding author (vprsundar@gmail.com)

2. Preliminaries

We begin with the following definitions and related results.

Definition 2.1. [2] An embedding (f, P_f) of a graph $G(V_G, E_G)$ into a graph $H(V_H, E_H)$ is defined by a mapping f from V_G to V_H, together with a mapping P_f that maps each edge $uv \in E_G$ onto a path $P_f(uv)$ in H that connects $f(u)$ and $f(v)$. The load on a node $v \in V_H$ is the number of nodes of G that are mapped onto v, the max-load of an embedding is the maximum load over all nodes of H. The expansion of an embedding f is the ratio of the number of vertices of H to the number of vertices of G.

For brevity, we denote in the rest of the paper the pair (f, P_f) simply as f. If f is an embedding of G into H and $e_H \in E(H)$, then $EC_f(e_H) = |\{e_G \in E(G) : e_H \in E(P_f(e_G))\}|$.

The *congestion of an embedding* f of G into H is

$$EC_f(G, H) = \max_{e_H \in E(H)} EC_f(e_H)$$

and the *congestion of embedding* G into H is

$$EC(G, H) = \min_{f:G \to H} EC_f(G, H)\,.$$

Further, if f is an embedding of G into H and $S \subseteq E(H)$, then we set $EC_f(S) = \sum_{e_H \in S} EC_f(e_H)$.

The *wirelength of an embedding* f of G into H is

$$WL_f(G, H) = \sum_{e_H \in E(H)} EC_f(e_H)\,,$$

and the *wirelength of embedding* G into H is

$$WL(G, H) = \min_{f:G \to H} WL_f(G, H)\,.$$

The following two problems, so called Maximum Subgraph Problem (MSP) and Minimum Cut Problem (MCP) are considered in the literature [1], and proved to be NP-complete [7].

For a subgraph M of G of order n,

- $I_G(M) = \{uv \in E \mid u, v \in M\}$, $I_G(k) = \max_{M \subseteq V(G),\ |M|=k} |I_G(M)|$
- $\theta_G(M) = \{uv \in E \mid u \in M, v \notin M\}$, $\theta_G(k) = \min_{M \subseteq V(G),\ |M|=k} |\theta_G(M)|$

The maximum subgraph problem for a given k, $k \in [n]$ is a problem of computing a subset M of $V(G)$ such that $|M| = k$ and $|I_G(M)| = I_G(k)$. Further, the subsets M are called the *optimal set* [3, 7, 8]. Similarly, we define the minimum cut problem for a given k, $k \in [n]$ is a problem of computing a subset M of $V(G)$ such that $|M| = k$ and $|\theta_G(M)| = \theta_G(k)$. For a regular graph, say r, we have $2I_G(k) + \theta_G(k) = rk$, $k \in [n]$ [3].

The following lemmas are efficient techniques to find the exact wirelength using MSP [12].

Lemma 2.1. *Let f from G to H be an embedding with same order. Let T be the set of all edges (or edge cut) of H such that $E(H) \setminus T$ has exactly two connected graphs, say H_1 and H_2 and let $G_q = G[f^{-1}(V(H_q))]$, $q = 1, 2$. In addition, T must satisfy the following:*

1. *For each edge $uv \in E(G_q)$, $q = 1, 2$, $P_f(uv)$ has no edges in the set T.*
2. *For each edge $uv \in E(G)$ with u in $V(G_1)$ and v in $V(G_2)$, $P_f(uv)$ has only one edge in the set T.*
3. *$V(G_1)$ and $V(G_2)$ are optimal sets.*

Then $EC_f(T)$ is minimum over all embeddings $f : G \to H$ and $EC_f(T) = \sum_{v \in V(G_1)} deg_G(v) - 2|E(G_1)| = \sum_{v \in V(G_2)} deg_G(v) - 2|E(G_2)|$, where $deg_G(v)$ is the degree of a vertex v in G.

Remark 2.1. If G is regular, then it is easy to see that, $V(G_2)$ is optimal if and only if $V(G_1)$ is optimal [11].

Lemma 2.2. *For an embedding f from G into H, let $\{T_1, T_2, \ldots, T_g\}$ be a partition of $[kE(H)]$ such that each T_q is an edge cut of H and it satisfies all the conditions of Lemma 2.1. Then*

$$WL_f(G,H) = \frac{1}{k}\sum_{q=1}^{g} EC_f(T_q)$$

where $[k\ E(H)]$ denote the collection of edges of H, each repeated exactly k times, $k \geq 1$.

Remark 2.2. [14] *For an embedding f from G into H, where f satisfies Lemma 2.2. Then the wirelength of an embedding from G into H is equal to the wirelength of an embedding from G into H with respect to f.*

Definition 2.2. [18] A graph $CH_t(V,E)$ is a chord graph on $n = 2^t$ nodes with the following vertex and edge sets: $V(CH_t) = \{v_0, v_1, v_2, \ldots, v_{2^t-1}\}$ and $e = (v_q, v_l) \in E(CH_t)$ iff $q + 2^k =_{mod_{2^t}} l$ or $l + 2^k =_{mod_{2^t}} q$, $\forall k \in \{0, 1, \ldots, t-1\}$, we say that the length of e is 2^k.

Definition 2.3. [18] A k - subchord is a subgraph of a CH_t, induced by vertices v_q where $q + s2^{t-k} = mod_{2^t} c$ or $c + s2^{t-k} = mod_{2^t} q$ for constant c $(0 \leq c \leq 2^t - 1)$ and $s \in N$.

Theorem 2.1. [19] For each $1 \leq m \leq 2^t$ and $S \subseteq V(CH_t)$, if S is an Ichord, then it maximizes $E(S)$ for its cardinality m and it is denoted by L_m.

3. Main results

In this section, we calculate the exact wirelength of embedding chord graphs into certain necklace and windmill graphs.

3.1 Necklace graphs

Before we prove the main theorem, we first start with the following definitions and remarks.

Definition 3.1. [17] Let K_p and K_{t_q} be complete graphs on p (say $v_1, v_2, ..., v_p$) and t_q vertices respectively. Let $t_q = 2^{r_q}$, $1 \leq q \leq p$ and $r_1 = r_2 .... = r_p$, such that $K_p \uplus K_{t_q}$ has just v_q as a cut vertex, where r_q is an integer and $1 \leq q \leq p$. The resultant graph $K_p \uplus (\bigcup_{q=1}^{p} K_{t_q})$ is a circular necklace denoted by $CN(K_p; K_{t_1}, K_{t_2}, ..., K_{t_p})$.

Remark 3.1. *$CN(K_p; K_{t_1}, K_{t_2}, ..., K_{t_p})$ has $2^t = \sum_{q=1}^{p} t_q$ vertices, where $t_q = 2^{r_q}$. We denote $\sum_{q=0}^{k} t_q$ by s_k, $0 \leq k \leq p$, where $t_0 = 0$. For brevity, the circular necklace $CN(K_p; K_{t_1}, K_{t_2}, ..., K_{t_p})$ will be represented by $CN(K_p, K)$.*

Definition 3.2. [17] Let $K_{1,p}$ be a star graph on $p+1$ vertices, say $v_0, v_1, ..., v_p$. Let K_{t_q} be a complete graph on t_q vertices and $t_q = 2^{r_q}, q = 1, 2, ..., p-1$, $r_1 = r_2, r_{q+1} = r_q + 1$ for all $q = 2, ..., p-1$ and $t_p = 2^{r_p} - 1$ such that $K_{1,p} \uplus K_{t_q}$ has just v_q as a cut vertex, where r_q is an integer. The resultant graph $K_{1,p} \uplus (\bigcup_{q=1}^{p} K_{t_q})$ is a necklace denoted by $N(K_{1,p}; K_{t_1}, K_{t_2}, ..., K_{t_p})$.

Remark 3.2. *$N(K_{1,p}; K_{t_1}, K_{t_2}, ..., K_{t_p})$ has $2^t = \sum_{q=1}^{p} t_q + 1$ vertices, where $t_q = 2^{r_q}$. We denote $\sum_{q=0}^{k} t_q$ by s_k, $0 \leq k \leq p$, where $t_0 = 0$. For brevity, the necklace $N(K_{1,p}; K_{t_1}, K_{t_2}, ..., K_{t_p})$ will be represented by $N(K_{1,p}, K)$.*

Embedding Algorithm A

Input : The chord graph CH_t and a circular necklace $CN(K_p, K)$.

Algorithm : Label the vertices of CH_t by Algorithm 1 [19] from 0 to $2^t - 1$. Label the vertices of K_{t_q} in $CN(K_p, K)$ as $s_{q-1} + l, l = 0, 1, 2, ..., t_q - 1$ such that $s_q - 1$ is the label of v_q, $1 \leq q \leq p$.

Output : An embedding f of CH_t into $CN(K_p, K)$ given by $f(x) = x$ with minimum wirelength.

Proof of correctness : We assume that the labels represent the vertices to which they are assigned. For $1 \leq q \leq p$, let $T_q = \{(s_q - 1, s_l - 1) : 1 \leq l \leq p, q \neq l\}$. For $1 \leq q \leq p$, let $T'_q = \{(s_q - 1, s_q - 1 - l) : 1 \leq l \leq t_q - 1\}$. For $1 \leq q \leq p$ and $0 \leq l \leq t_q - 2$, let $T^l_q = \{(s_{q-1} + l, s_{q-1} + k) : 0 \leq k \leq t_q - 1 \text{ and } l \neq k\}$. Then $\{T_q, T'_q : 1 \leq q \leq p\} \cup \{T^l_q : 1 \leq q \leq p, 0 \leq l \leq t_q - 2\}$ is a partition of $[2E(CN(K_p, K))]$.

For each q, $1 \leq q \leq p$, $E(CN(K_p, K)) \backslash T_q$ has two components H_{q1} and H_{q2}, where $V(H_{q1}) = \{s_{q-1}, s_{q-1} + 1, ..., s_q - 1\}$. Let $G_{q1} = CH_t[f^{-1}(V(H_{q1}))]$ and $G_{q2} = CH_t[f^{-1}(V(H_{q2}))]$. By Theorem 2.1, G_{q1} is an optimal set and each T_q satisfies all the conditions of Lemma 2.1. Therefore $EC_f(T_q)$ is minimum.

For each q, $1 \leq q \leq p$, $E(CN(K_p, K)) \setminus T'_q$ has two components H'_{q1} and H'_{q2}, where $V(H'_{q1}) = \{s_{q-1}, s_{q-1}+1, ..., s_{q-1}+t_q-2\}$. Let $G'_{q1} = CH_t[f^{-1}(V(H'_{q1}))]$ and $G'_{q2} = CH_t[f^{-1}(V(H'_{q2}))]$. By Theorem 2.1, G'_{q1} is an optimal set and each T'_q satisfies all the conditions of Lemma 2.1. Therefore $EC_f(T'_q)$ is minimum.

For each q, l, $1 \leq q \leq p$ and $0 \leq l \leq t_q-2$, $E(CN(K_p, K)) \setminus T^l_q$ has two components H^l_{q1} and H^l_{q2}, where $V(H^l_{q1}) = \{s_{q-1}+l\}$. Let $G^l_{q1} = CH_t[f^{-1}(V(H^l_{q1}))]$ and $G^l_{q2} = CH_t[f^{-1}(V(H^l_{q2}))]$. By Theorem 2.1, G^l_{q1} is an optimal set, each T^l_q satisfies all the conditions of Lemma 2.1. Therefore $EC_f(T^l_q)$ is minimum. By Lemma 2.2 implies that the wirelength is minimum.

Now, we have the following theorem.

Theorem 3.1. *The exact wirelength of embedding CH_t into $CN(K_p, K)$ is given by*

$$WL(CH_t, CN(K_p, K)) = \frac{1}{2}\sum_{q=1}^{p}[(2t-1)t_q - 2|I_{CH_t}(t_q)|] + \frac{1}{2}\sum_{q=1}^{p}[(2t-1)(t_q-1) - 2|I_{CH_t}(t_q-1)|] + \frac{2t-1}{2}(2^t - p).$$

Proof. By Embedding Algorithm A,

(i) $EC_f(T_q) = (2t-1)t_q - 2|I_{CH_t}(t_q)|$, $1 \leq q \leq p$

(ii) $EC_f(T'_q) = (2t-1)(t_q-1) - 2|I_{CH_t}(t_q-1)|$, $1 \leq q \leq p$ and

(iii) $EC_f(T^l_q) = 2t-1$, $1 \leq q \leq p$ and $0 \leq l \leq t_q-2$.

Then by Lemma 2.2,

$$\begin{aligned} WL(CH_t, CN(K_p, K)) &= \frac{1}{2}\left[\sum_{q=1}^{p} EC_f(T_q) + \sum_{q=1}^{p} EC_f(T'_q) + \sum_{q=1}^{p}\sum_{l=0}^{t_q-2} EC_f(T^l_q)\right] \\ &= \frac{1}{2}\sum_{q=1}^{p}[(2t-1)t_q - 2|I_{CH_t}(t_q)|] + \frac{1}{2}\sum_{q=1}^{p}[(2t-1)(t_q-1) - 2|I_{CH_t}(t_q-1)|] \\ &\quad + \frac{2t-1}{2}(2^t - p). \end{aligned}$$

□

Embedding Algorithm B

Input : The chord graph CH_t and a necklace $N(K_{1,p}, K)$.

Algorithm : Label the vertices of CH_t by Algorithm 1 [19] from 0 to 2^t-1. Label the vertices of K_{t_q} in $N(K_{1,p}, K)$ as $s_{q-1}+l$, $l = 0, 1, 2, ..., t_q-1$ such that, $s_q - 1$ is the label of v_q, $1 \leq q \leq p$ and v_0 as 2^t-1.

Output : An embedding f of CH_t into $N(K_{1,p}, K)$ given by $f(x) = x$ with minimum wirelength.

Proof of correctness : We assume that the labels represent the vertices to which they are assigned. For $1 \leq q \leq p$, let $T_q = T_{q'} = \{(s_q-1, 2^t-1)\}$. For $1 \leq q \leq p$, let $T'_q = \{(s_q-1, s_q-1-l) : 1 \leq l \leq t_q-1\}$. For $1 \leq q \leq p$ and $0 \leq l \leq t_q-2$, let $T^l_q = \{(s_{q-1}+l, s_{q-1}+k) : 0 \leq k \leq t_q-1 \text{ and } q \neq k\}$. Then $\{T_q, T_{q'}, T'_q : 1 \leq q \leq p\} \cup \{T^l_q : 1 \leq q \leq p, 0 \leq l \leq t_q-2\}$ is a partition of $[2E(N(K_{1,p}, K))]$.

For each q, $1 \leq q \leq p$, $E(N(K_{1,p}, K)) \setminus T_q$ has two components H_{q1} and H_{q2}, where $V(H_{q1}) = \{s_{q-1}, s_{q-1}+1, ..., s_q-1\}$. Let $G_{q1} = CH_t[f^{-1}(V(H_{q1}))]$ and $G_{q2} = CH_t[f^{-1}(V(H_{q2}))]$. By Theorem 2.1, G_{q1} is an optimal set, each T_q satisfies all the conditions of Lemma 2.1. Therefore $EC_f(T_q)$ is minimum. Similarly, $EC_f(T_{q'})$ is minimum.

For each q, $1 \leq q \leq p$, $E(N(K_{1,p}, K)) \setminus T'_q$ has two components H'_{q1} and H'_{q2}, where $V(H'_{q1}) = \{s_{q-1}, s_{q-1}+1, ..., s_{q-1}+t_q-2\}$. Let $G'_{q1} = CH_t[f^{-1}(V(H'_{q1}))]$ and $G'_{q2} = CH_t[f^{-1}(V(H'_{q2}))]$. By Theorem 2.1, G'_{q1} is an optimal set, each T'_q satisfies all the conditions of Lemma 2.1. Therefore $EC_f(T'_q)$ is minimum.

For each q, l, $1 \leq q \leq p$ and $0 \leq l \leq t_q-2$, $E(N(K_{1,p}, K)) \setminus T^l_q$ has two components H^l_{q1} and H^l_{q2}, where $V(H^l_{q1}) = \{s_{q-1}+l\}$. Let $G^l_{q1} = CH_t[f^{-1}(V(H^l_{q1}))]$ and $G^l_{q2} = CH_t[f^{-1}(V(H^l_{q2}))]$. By Theorem 2.1, G^l_{q1} is an optimal set, each T^l_q satisfies all the conditions of Lemma 2.1. Therefore $EC_f(T^l_q)$ is minimum. Then by Lemma 2.2 implies that the wirelength is minimum.

As a consequence of Lemma 2.2 and Embedding Algorithm B, we have the following result.

Theorem 3.2. *The exact wirelength of embedding CH_t into $N(K_{1,p}, K)$ is given by*

$$WL(CH_t, N(K_{1,p}, K)) = \frac{1}{2}\sum_{q=1}^{p}[(2t-1)t_q - 2|I_{CH_t}(t_q)|] + \frac{1}{2}\sum_{q=1}^{p}[(2t-1)(t_q-1) - 2|I_{CH_t}(t_q-1)|] + \frac{2t-1}{2}(2^t - (p+1)).$$

3.2 Windmill graphs

In this section, we calculate the exact wirelength of chord graphs into windmill graphs. To prove the main result, we require the following definition and a remark.

Definition 3.3. [17] Let K_{t_q} be a complete graph on t_q vertices and $t_q = 2^{r_q} + 1, r_{q+1} = r_q + 1$ for all $q = 2, 3, ..., p-1$ and $t_1 = 2^{r_1}, t_2 = 2^{r_2} + 1, r_1 = r_2$ such that $\uplus K_{t_q}$ has just v_1 is a cut vertex. The resultant graph $\biguplus_{q=1}^{m} K_{t_q}$ is a windmill graph incident with a common vertex v_1 denoted by $WM(K_{t_1}, K_{t_2}, ..., K_{t_p})$.

Remark 3.3. *$WM(K_{t_1}, K_{t_2}, ..., K_{t_p})$ has $2^t = \sum_{q=1}^{p} t_q - p + 1$ vertices. We denote $\sum_{q=0}^{k} t_q$ by s_k, $0 \leq k \leq p$, where $t_0 = 0$. For brevity, the graph $WM(K_{t_1}, K_{t_2}, ..., K_{t_p})$ will be represented by $WM(K)$.*

Embedding Algorithm C

Input : The chord graph CH_t and the windmill graph $WM(K)$.

Algorithm : Label the vertices of CH_t by Algorithm 1 [19] from 0 to $2^t - 1$. Label the vertices of K_{t_q} in $WM(K)$ as $s_{q-1} + l$, $l = 0, 1, 2, ..., t_q - q$ such that $s_1 - 1$ is the label of v_1, $1 \leq q \leq p$.

Output : An embedding f of CH_t into $WM(K)$ given by $f(x) = x$ with minimum wirelength.

Proof of correctness : We assume that the labels represent the vertices to which they are assigned. Let $T_1 = \{(s_1 - 1, s_q - 1 - l) : 1 \leq q \leq t_q - 1\}$ and for $2 \leq q \leq p$, let $T_q = \{(s_1 - 1, s_q - (q-1) - l) : 1 \leq l \leq t_q - 1\}$. For $1 \leq l \leq t_q - 1$, let $T_1^l = \{(l-1, k-1) : 1 \leq k \leq t_q - 1$ and $l \neq k\}$ and for $2 \leq q \leq p$, $1 \leq q \leq t_q - 1$, let $T_q^l = \{(s_{q-1} + l - (q-1), s_1 - 1), (s_{q-1} + l - (q-1), s_{q-1} + k - (q-1)) : 1 \leq k \leq t_q - 1$ and $l \neq k$. Then $\{T_q : 1 \leq q \leq p\} \cup \{T_q^l : 1 \leq q \leq p, 1 \leq l \leq t_q - 1\}$ is a partition of $[2E(WM(K))]$.

For each q, $2 \leq q \leq p$, $E(WM(K)) \backslash T_q$ has two components H_{q1} and H_{q2}, where $V(H_{q1}) = \{s_q - q - t_q + 2, s_q - q - t_q + 3, ..., s_q - q\}$. Let $G_{q1} = CH_t[f^{-1}(V(H_{q1}))]$ and $G_{q2} = CH_t[f^{-1}(V(H_{q2}))]$. Since G_{q1} is an optimal set, each T_q satisfies all the conditions of Lemma 2.1. Therefore $EC_f(T_q)$ is minimum. Similarly, $EC_f(T_1)$ is minimum.

For each q, l, $2 \leq q \leq p, 1 \leq l \leq t_q - 1$, $E(MC(P, K)) \backslash T_q^l$ has two components H_{q1}^l and H_{q2}^l, where $V(H_{q1}^l) = \{s_{q-1} + l - (q-1)\}$. Let $G_{q1}^l = CH_t[f^{-1}(V(H_{q1}^l))]$ and $G_{q2}^l = CH_t[f^{-1}(V(H_{q2}^l))]$. Since G_{q1}^l is an optimal set, each T_q^l satisfies all the conditions of Lemma 2.1. Therefore $EC_f(T_q^l)$ is minimum. Similarly, $EC_f(T_1^l)$ is minimum. Then by Lemma 2.2 implies that the wirelength is minimum.

Now, we have the following result.

Theorem 3.3. *The exact wirelength of CH_t into $WM(K)$, is given by*

$$WL(CH_t, WM(K)) = \frac{1}{2}\sum_{q=1}^{p} \left[(2t-1)(t_q - 1) - 2|E(CH_t[L_{t_q - 1}])|\right] + \frac{2t-1}{2}(2^t - 1).$$

Proof. By Embedding Algorithm C,

(i) $EC_f(T_q) = (2t-1)(t_q - 1) - 2|E(CH_t[L_{t_q-1}]), 1 \leq q \leq p$ and

(ii) $EC_f(T_q^l) = 2t - 1$, $1 \leq q \leq p$ and $1 \leq l \leq t_q - 1$.

Then by Lemma 2.2,

$$\begin{aligned} WL(CH_t, WM(K)) &= \frac{1}{2}\left[\sum_{q=1}^{p} EC_f(T_q) + \sum_{q=1}^{p}\sum_{l=1}^{t_q - 1} EC_f(T_q^l)\right] \\ &= \frac{1}{2}\sum_{q=1}^{p} \left[(2t-1)(t_q - 1) - 2|E(CH_t[L_{t_q-1}])|\right] + \frac{2t-1}{2}(2^t - 1). \end{aligned}$$

□

4. Time complexity

The period of time to perform an algorithm is represented by the time complexity, specifically known as computational complexity in the domain of computer science. It is assumed that the execution of a specific fundamental operation takes a particular interval of time. Hence, the phenomena of time complexity is approximately calculated by the number of fundamental functions carried out by the algorithm. Therefore, the number of primary operations carried out by the algorithm and the interval of time required are drawn by a constant factor to vary.

The length of time for an algorithm to perform its task may differ over various inputs belonging to same sizes. The characteristic feature of time complexity is determined by the amount of time it takes to complete the given task. The maximum period of time needed for outputs of a given size is generally contemplated that it is the inadequate kind of time complexity. Less frequently and it is often peculiarly, the mean duration of time required for inputs of a specified size is the ideal-average complexity. In both of these instances, the performance of the size of the input is commonly articulated by the time complexity [22].

In the present section, we compute the time complexity of determining the accurate wirelength of embedding chord graphs into the circular necklace using Embedding Algorithm A. The algorithm is formally exhibited in the following way.

Time Complexity Algorithm

Input : The chord graph CH_t and the circular necklace $CN(K_p, K)$, $t, p \geq 3$.

Algorithm : Embedding Algorithm A.

Output : The time taken to compute the minimum (and exact) wirelength of embedding CH_t into $CN(K_p, K)$ is $O(t)$, which is linear.

Method : Since the chord graph, CH_t contains 2^t vertices, then for assigning the labels of 2^t vertices, we need $\lceil \log_2 2^t \rceil$ time units. By Embedding Algorithm A, we have $p(t_q) + p$, $1 \leq q \leq p$ edge cuts. For each edge cut, we require one unit of time and by Lemma 2.1, we require $m(r_q)$ time units. Again for finding the edge congestion of each edge cut, we require one unit of time. In addition, we require $p(r_q)$ units of time for finding the wirelength by using Lemma 2.2.

$$\begin{aligned} \text{Hence the total time needed to estimate the wirelength is} \quad &= \quad t + \sum_{q=1}^{p} p(t_q + 1)2^t \\ &= \quad O(t) \end{aligned}$$

which is linear. □

In this similar manner, we can estimate the exact wirelength of embedding chord graphs into necklace and windmill graphs in linear time.

5. Conclusion

In this paper, we computed the exact wirelength of chord into certain necklace and windmill graphs. Future directions for research of this nature would be to measure the exact wirelength and complexity of embedding chord graphs into such popular interconnection architectures as the rectangular grid and its variations, the cylinder and torus. In a more theoretical sense though still potential for valuable knowledge would be to study embedding chord graphs into trees.

Acknowledgments

The last author (R. S. Rajan) is supported by National Board of Higher Mathematics (NBHM), Department of Atomic Energy (DAE), Government of India, through Project No. 2/48(4)/2016/NBHM-R&D-II/11580. The authors thank the anonymous referees for their suggestions which improved the quality of the paper.

References

[1] S. L. Bezrukov, Embedding complete trees into the hypercube, *Discrete Appl. Math.* **110** (2001) 101–119.
[2] S. Bhatt, F. Chung, F. Leighton, A. Rosenberg, Efficient embeddings of trees in hypercubes, *SIAM J. Comput.* **21** (1992) 151–162.
[3] S. L. Bezrukov, S. K. Das, R. Elsässer, An edge-isoperimetric problem for powers of the Petersen graph, *Ann. Comb.* **4** (2000) 153–169.
[4] A. Caprara, M. Oswald, G. Reinelt, R. Schwarz, E. Traversi, Optimal linear arrangements using betweenness variables, *Math. Program. Comput.* **3(3)** (2011) 261–280.
[5] S. A. Choudum, R. Indhumathi, On embedding subclasses of height-balanced trees in hypercubes, *Inform. Sci.* **179(9)** (2009) 1333–1347.
[6] W. Fan, J. Fan, C-K. Lin, G. Wang, B. Cheng, R. Wang, An efficient algorithm for embedding exchanged hypercubes into grids, *J. Supercomput.* **75(2)** (2019) 783–807.

[7] M. R. Garey, D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman, San Francisco, 1979.
[8] L. H. Harper, *Global Methods for Combinatorial Isoperimetric Problems*, Cambridge University Press, Cambridge, 2004.
[9] P. L. Lai, C. H. Tsai, Embedding of tori and grids into twisted cubes, *Theor. Comput. Sci.* **411** (2010) 3763–3773.
[10] J-B. Liu, M. Arockiaraj, J. N. Delaila, Wirelength of enhanced hypercube into windmill and necklace graphs, *Mathematics* **7(5)** (2019) 383–392.
[11] P. Manuel, I. Rajasingh, B. Rajan, H. Mercy, Exact wirelength of hypercube on a grid, *Discrete Appl. Math.* **157** (2009) 1486–1495.
[12] M. Miller, R. S. Rajan, N. Parthiban, I. Rajasingh, Minimum linear arrangement of incomplete hypercubes, *Comput. J.* **58** (2015) 331–337.
[13] N. Parthiban, J. Ryan, I. Rajasingh, R. S. Rajan, L. N. Rani, Exact wirelength of embedding chord graph into tree-based architectures, *Int. J. Netw. Virtual Organisations* **17** (2017) 76–87.
[14] R. S. Rajan, T. M. Rajalaxmi, J. B. Liu, G. Sethuraman, Wirelength of embedding complete multipartite graphs into certain graphs, *Discrete Appl. Math.*, DOI: 10.1016/j.dam.2018.05.034, In press.
[15] R. S. Rajan, T. M. Rajalaxmi, J. Ryan, M. Miller, Improved bound for dilation of an embedding onto circulant networks, In: B. Rushi Kumar, R. Sivaraj, B. Prasad, M. Nalliah, A. Reddy (Eds.), *Applied Mathematics and Scientific Computing*, Birkhäuser, Cham, 2019, pp. 623–632.
[16] R. S. Rajan, T. Kalinowski, S. Klavžar, H. Mokhtar, T. M. Rajalaxmi, Lower bounds for dilation, wirelength and edge congestion of embedding graphs into hypercubes, (2019), Manuscript.
[17] I. Rajasingh, B. Rajan, R. S. Rajan, Embedding of hypercubes into necklace, windmill and snake graphs, *Inform. Process. Lett.* **112(12)** (2012) 509–515.
[18] P. Raoufi, H. Rostami, H. Bagherinezhad, An optimal time algorithm for minimum linear arrangement of chord graphs, *Inform. Sci.* **238** (2013) 212–220.
[19] H. Rostami, J. Habibi, Minimum linear arrangement of chord graphs, *Appl. Math. Comput.* **203** (2008) 358–367.
[20] M. Rottger, U. P. Schroeder, Efficient embeddings of grids into grids, *Discrete Appl. Math.* **108** (2001) 143–173.
[21] J. P. I. Silvestre, *Approximation heuristics and benchmarkings for the MinLA problem*, Proceedings of Algorithms and Experiments (ALEX'98): Building Bridges Between Theory and Applications, Italy, Feb 9–11, 1998, pp. 112–128.
[22] M. Sipser, *Introduction to the Theory of Computation*, Thomson Course Technology, Boston, 2006.
[23] I. Stoica, R. Morris, D. L. Nowell, D. R. Karger, M. F. Kaashoek, F. Dabek, H. Balakrishnan, Chord: a scalable peer-to-peer lookup protocol for internet applications, *IEEE/ACM Trans. Netw.* **11** (2003) 17–32.
[24] J. Xu, *Topological Structure and Analysis of Interconnection Networks*, Springer, New York, 2001.
[25] C. Yu, D. Yao, X. Li, Y. Zhang, L. T. Yang, N. Xiong, H. Jin, Location-aware private service discovery in pervasive computing environment, *Inform. Sci.* **230** (2013) 78–93.