On Minimum Circular Arrangement*

Murali K Ganapathy[1]** and Sachin P Lodha[2]

[1] Dept. of Computer Science, University of Chicago, Chicago, Illinois, USA.
[2] Tata Research Development and Design Centre, Pune, Maharashtra, India.

{gmkrishn@cs.uchicago.edu, sachin.lodha@tcs.com}

Abstract. Motivated by a scheduling problem encountered in multicast environments, we study a vertex labelling problem, called Minimum Circular Arrangement (MCA), that requires one to find an embedding of a given weighted directed graph into a discrete circle which minimizes the total weighted arc length. Its decision version is already known to be NP-complete when restricted to sparse weighted instances. We prove that the decision version of even un-weighted MCA is NP-complete in case of sparse as well as dense graphs.

We also consider complementary version of MCA, called MaxCA. We prove that it is MAX-SNP[π] complete and, therefore, has no PTAS unless P=NP. A similar proof technique shows that MCA is MAX-SNP[π]-Hard and hence admits no PTAS as well. Then we prove a conditional lower bound of $\sqrt{2} - \epsilon$ for MCA approximation under some hardness assumptions, and conclude with a PTAS for MCA on dense instances.

Keywords: Computational complexity, hardness of approximation, polynomial time approximation scheme, scheduling, multicast.

1 Introduction

Availability of very high-speed and large bandwidth networks, explosion in internetworking, and advent of cheap, low-power, portable computing devices have given rise to one-to-many asymmetric communication networks and huge client populations having commonality of interests [1]. In such environments, servers are endowed with much more computing power and have access to much larger bandwidth than clients. Therefore, it becomes cost effective to *push* data from server side rather than follow traditional client-server based *pull* model. This can be achieved using *multicast* where a server needs to send a data unit only once to reach arbitrary number of clients.

A common way to use multicast in data dissemination is to use server initiated *repetitive* multicast where a server cyclically multicasts data to a large

* Full version of the paper is available online [12].
** This work was done when the author was visiting Tata Research Development and Design Centre as a summer intern in 2003.

client population. This finds application in many diverse domains, *e.g.*, high-throughput database systems [15], data management in broadcast disks [1], in solving scalability problems of heavily loaded Web servers [3], content delivery networks (CDNs) [22], etc.

A fundamental question is the order in which the server should multicast data, that is, *scheduling*. In general, clients are seldom interested in individual data items, and attempt to download multiple items. For example, Web clients hardly ever access only one HTML resource, but access almost always the HTML document along with all its embedded images [17]. Database clients often access multiple items to complete a read transaction [24]. Thus client access patterns often show dependencies between consecutive requests, so that the request for a data unit will make it more likely or less likely that certain data unit will be requested next. These access patterns must be taken into account while designing a good cyclic multicast schedule that has low client-perceived latency while accessing multi-item objects [19].

One way to model this scenario is to treat the server data set as a weighted directed graph where nodes represent server data units and arc weights represent the strength of the dependency. Then the scheduling problem becomes following question in combinatorial optimization:

Minimum Circular Arrangement (MCA): Given a directed weighted graph $G = (V, E, w)$ with non-negative weights, find a surjection $f : V \mapsto \{0, 1, \ldots, n-1\}$ which minimizes $\sum_{e \in E} w(e)\ell(e)$, where $\ell(e) = (f(v) - f(u)) \mod n$, for $e = (u, v)$. Note that $\ell(e)$ is called the latency of the edge e in the arrangement f.

1.1 Related Problems

The MCA problem first appeared in work of Liberatore [19]. It falls under the class of vertex labelling problems where the question is to find a labelling of the vertices which optimizes some cost function. This class includes many interesting practical problems [7], *e.g.*, optimal linear arrangement problem, directed optimal linear arrangement problem, minimum bandwidth problem, folding labelling (also called minimum cut linear arrangement) problem, etc. We give more consideration to optimal linear arrangement problem and directed optimal linear arrangement problem since MCA is very closely related to them.

Optimal Linear Arrangement (OLA): Given an undirected weighted graph $G = (V, E, w)$ with non-negative weights, find a surjection $f : V \mapsto \{0, 1, \ldots, n-1\}$ which minimizes $\sum_{e \in E} w(e)\ell(e)$, where $\ell(e) = |f(v) - f(u)|$, for $e = (u, v)$.

OLA problem naturally arose from applications in VLSI design. Garey, Johnson and Stockmeyer [14] proved NP-completeness of the decision version of OLA. Today we know how to solve OLA problem exactly for some special cases of graphs, *e.g.*, un-weighted trees [25, 8], outer planar graphs [10], cycles, wheels, complete bipartite graphs [16], etc. For arbitrary graphs, the currently best known guarantee of $O(\log n)$-approximation is due to Rao and Richa [23]. Meanwhile, there has also been some work done on polynomial time approximation schemes for un-weighted OLA of dense graphs, namely, [4] and [11].

Directed Optimal Linear Arrangement (DOLA): Given a directed acyclic weighted graph $G = (V, E, w)$ with non-negative weights, find a surjection $f : V \mapsto \{0, 1, \ldots, n-1\}$ such that $(u, v) \in E \implies f(u) < f(v)$, i.e. a topological sort, which minimizes $\sum_{e \in E} w(e)\ell(e)$, where $\ell(e) = f(v) - f(u)$, for $e = (u, v)$.

Not much is known about DOLA. Its decision version was shown to be NP-complete by Even and Shiloach [9]. On the algorithmic front, Adolphson and Hu [2] gave an $O(n \log n)$-time algorithm to solve DOLA exactly on rooted trees, where all the edges are oriented towards (or away from) the root. Otherwise neither any approximation algorithms nor any hardness of approximation results are known for it.

1.2 Current Status

The MCA problem is pretty recent [19]. Very few theoretical results are known about it. One of them is the proof of NP-completeness for the decision version of MCA problem restricted to sparse weighted graphs by Liberatore [18]. He also demonstrates an $\widetilde{O}(\sqrt{n})$-approximation algorithm on any arbitrary graph instance using divide-and-conquer strategy in [18]. This result has recently been improved by Naor and Schwartz in [20] where they present $O(\log n \log \log n)$-approximation algorithm for the MCA problem.

1.3 Our Results

In this paper, we start out by proving some preliminary lemmas in section 3 that bound MCA cost. In section 4, we draw comparison between MCA cost and OLA cost (DOLA cost), throwing light on the relative hardness of these problems.

We prove that the decision version of even un-weighted MCA is NP-complete in case of sparse as well as dense graphs (section 5), a stronger result than [18]. We also consider complementary version of MCA, called MaxCA in section 6. We prove that it is MAX-SNP[π] complete [21] and, therefore, has no PTAS unless P=NP [5]. A similar proof technique would then show that MCA is MAX-SNP[π]-Hard and hence there is no PTAS for MCA too [5]. In section 7, we prove a conditional lower bound of $\sqrt{2} - \epsilon$ for MCA approximation under the assumption that DOLA does not admit constant factor approximation. Finally we conclude with a PTAS for MCA on dense instances in section 8.

2 Notation

By a graph G, we mean a directed graph without parallel edges and loops. V and E, as always, stand for vertex-set and edge-set of G respectively. $|V| = n$.

An un-weighted graph is considered as a graph with edges of unit weight. When we talk about the OLA problem on a directed graph we mean the OLA problem on the underlying undirected graph.

Definition 1. *A graph G is* ***dense*** *if $|E| = \Omega(n^2)$. More specifically, G is δ-****dense*** *if $|E| \geq \delta n^2$. Similarly G is* ***sparse*** *if $|E| = O(n)$.*

Consider a graph G and $f : V \mapsto \{1, \ldots, n\}$ an arrangement of G.

Definition 2. *An edge $e = (u, v)$ of G is said to be a* ***forward edge with respect to*** *f if $f(u) < f(v)$. Similarly e is a* ***backward edge with respect to*** *f if $f(u) > f(v)$. Note that the forward/backward status of an edge can be changed by rotating f.*

Let $\mathrm{MCOST}(f)$, $\mathrm{LCOST}(f)$ and $\mathrm{DCOST}(f)$ respectively denote the circular, linear and the directed linear cost of the arrangement f as defined in the problem definitions. For an edge e, $\mathrm{MCOST}_e(f)$ denotes the cost of the edge e in the arrangement f. Similarly for $\mathrm{DCOST}_e(f)$ and $\mathrm{LCOST}_e(f)$. Set $\mathrm{DCOST}_e(f) = \mathrm{DCOST}(f) = \infty$, if any edge $e = (u, v)$ is a backward edge.

Definition 3. *Let g be a circular arrangement of G. By $\mathrm{ROT}(g)$ we mean an arrangement h obtained by rotating g so that the total weight of the backward edges is minimized. Note that $\mathrm{MCOST}(\mathrm{ROT}(g)) = \mathrm{MCOST}(g)$.*

Let $\mathrm{MCA}(G)$ be the set of all optimal circular arrangements of G. Similarly define $\mathrm{OLA}(G)$ and $\mathrm{DOLA}(G)$. Sometimes, by abuse of notation, $\mathrm{MCA}(G)$ also stands for some optimal circular arrangement. Similarly for $\mathrm{OLA}(G)$ and $\mathrm{DOLA}(G)$. Finally, let $\mathrm{MCOST}(G) := \mathrm{MCOST}(\mathrm{MCA}(G))$ denote the cost of the optimal arrangement. Similarly for $\mathrm{LCOST}(G)$ and $\mathrm{DCOST}(G)$.

By P_m we mean a directed path $p_1, \ldots, p_{m+1}$ of length m (on $m+1$ vertices), with unit weight edges. By $\overrightarrow{K}_n$ we mean the complete directed acyclic graph on n vertices, i.e. for $1 \leq i < j \leq n$, there is an edge (i, j) of unit weight.

By $\overleftarrow{G}$ we mean the graph *anti-parallel* to G, that is, $V(\overleftarrow{G}) = V(G)$ and $E(\overleftarrow{G}) = \{(v, u) | (u, v) \in E\}$. The edges in $\overleftarrow{G}$ carry same weight as their counterparts in G.

If G and H are two graphs, then $G + H$ is the graph which has G and H as its two components.

3 Bounding MCA Cost

In this section we show some upper and lower bounds on MCA cost and highlight its peculiar features that would help us derive our hardness results.

Proposition 1. *The total weight of the backward edges of* $\mathrm{ROT}(g) \leq \frac{\mathrm{MCOST}(g)}{n}$.

Proposition 2. *Let $G = H_1 + \cdots + H_k$ be a graph with k components. Put $n = |V(G)|$ and $n_i = |V(H_i)|$. Then $\sum_{i=1}^{k} \mathrm{MCOST}(H_i) \leq \mathrm{MCOST}(G) \leq n \sum_{i=1}^{k} \mathrm{MCOST}(H_i)/n_i$. Moreover these inequalities are tight.*

Proof (sketch). For $1 \leq i \leq k$, let $f_i \in \mathrm{MCA}(H_i)$ and consider $g = \mathrm{ROT}(f_1) \circ \cdots \circ \mathrm{ROT}(f_k)$. $\square$

This behavior of MCA (with components) enables us to derive our hardness results. The fundamental difference between MCA and OLA (or DOLA) is the issue of connectedness. In case of a graph with more than one component it is easy to the see that the optimal arrangement (in case of OLA and DOLA) is obtained by concatenating the optimal arrangements of the components. However, such is not the case with MCA. If there are any backward edges in the optimal circular arrangement of one of the components, then the latency of that edge is increased due to the presence of the other components.

Definition 4. *Let G be a weighted directed graph. For a vertex u, let $w_1 \geq \cdots \geq w_d$ denote the weights of the outgoing edges from u. Define $X^+(u) = \sum_i iw_i$, and $X^+(G) = \sum_{v \in V(G)} X^+(v)$. Similarly define $X^-(u)$ and $X^-(G)$ by replacing outgoing with incoming.*

Proposition 3. $\mathrm{MCOST}(G) \geq \max\{X^+(G), X^-(G)\}$.

4 Comparison of MCA with OLA and DOLA

4.1 Comparison with OLA

Proposition 4. *For any graph G,* $\mathrm{LCOST}(G) \leq 2(1 - 1/n) \cdot \mathrm{MCOST}(G)$.

Proof (sketch). Let $f \in \mathrm{MCA}(G)$ be an optimal circular arrangement. Denote by f_i the arrangement got by rotating f by i-positions. Now consider any edge $e = (u, v)$ of weight w with latency p with respect to the f ordering. The cost of this edge in the linear arrangement f_i is pw if $f_i(u) < f_i(v)$ and $(n - p)w$ if $f_i(u) > f_i(v)$. Averaging over all n rotations settles the claim. $\square$

To see that the above result is tight, consider $G = C_n$, a directed cycle. On the other hand, $\mathrm{MCOST}(G) \leq (n - 1)\,\mathrm{LCOST}(G)$ trivially. By considering appropriately directed sunflowers, one can show examples [12] which achieve $\mathrm{MCOST}(G) \geq (n/12)\,\mathrm{LCOST}(G)$.

4.2 Comparison with DOLA

From the definition, any legal DOLA arrangement is a legal MCA arrangement. Hence we trivially have $\mathrm{MCOST}(G) \leq \mathrm{DCOST}(G)$. On the other hand, $\mathrm{DCOST}(G)$ is trivially $\leq (n - 1)\,\mathrm{MCOST}(G)$. In case of weighted graphs this is optimal as shown in [19]. We can get little more sophisticated bound if we restrict ourselves to un-weighted graphs.

Proposition 5. *Let G be an un-weighted directed acyclic graph and f be any DOLA arrangement of G. Then* $\mathrm{DCOST}(f) \leq |E|n - \frac{2\sqrt{2}}{3}|E|\sqrt{|E|} + \frac{7}{3}|E|$. *Moreover for interesting E (i.e. $|E| \geq 28$),* $\mathrm{DCOST}(f) \leq |E|n - |E|\sqrt{|E|}/2$.

Proof (sketch). Note that in any legal DOLA arrangement of G, there can be at most $n - i$ edges with latency i for each i. □

Corollary 1. *For any un-weighted directed acyclic graph G,*

$$\mathrm{DCOST}(G) \leq \mathrm{MCOST}(G) \cdot \frac{n(2n - \sqrt{|E|})}{|E| + n}.$$

Thus in order to get a $\Omega(n)$ separation between DCOST(G) and MCOST(G), we only need to look at sparse graphs in the un-weighted case. Moreover any approximation algorithm for DOLA on dense graphs automatically yields an approximation algorithm for MCA on dense graphs. However an approximation algorithm for MCA does not apriori give rise to a DOLA approximation algorithm since an MCA arrangement need not be a legal DOLA arrangement.

5 NP Completeness

Theorem 1 (Proposition 3.1 in [18]). *The decision version of the MCA problem is NP-complete.*

Liberatore [18] proves that *weighted* MCA problem is NP-complete by a reduction from an un-weighted DOLA. Since the MCA instance in his proof has $|E| = O(|V|)$, we infer that MCA is NP-complete even when restricted to sparse graphs. In this section we prove that even *un-weighted* MCA is NP-complete in case of sparse as well as dense graphs, stronger result than Liberatore [18]. We too make use of reduction from an un-weighted DOLA.

5.1 Straightening Algorithm

We start with an algorithm which allows us to normalize optimal solutions in a special case.

Theorem 2 (Straightening Algorithm). *Let G be a weighted directed graph, and $m > 2$. Let f be any circular arrangement of $G + P_m$. We can transform f (in time polynomial in $m + n$) to an arrangement g in which all the vertices in P_m appearing in the order $p_1, \ldots, p_{m+1}$. Moreover* MCOST(g) $\leq$ MCOST(f).

Proof (sketch). For this proof it is more convenient to think of an arrangement as a mapping from $[n]$ to V or as an ordered list of vertices, rather than the other way around. Let f be any circular arrangement of $G + P_m$. We define a sequence of arrangements $g_1, \ldots, g_{m+1}$ with the following properties:

- $g_i(j) = p_j$ for all $1 \leq j \leq i \leq m + 1$
- $\mathrm{MCOST}(g_{i+1}) \leq \mathrm{MCOST}(g_i)$ for $1 \leq i \leq m$

Thus $g = g_{m+1}$ is the required arrangement. To start, let $g_1 = f$ suitably rotated so that $g_1(1) = p_1$. Note that $\mathrm{MCOST}(g_1) = \mathrm{MCOST}(f)$. Assume we know g_i and $i \le m$ (else we are done). If $g_i(i+1) = p_{i+1}$, then set $g_{i+1} = g_i$ and continue with the next i.

Suppose $g_i(i+1) \ne p_{i+1}$. Let $i + \ell$ denote the position of the vertex p_{i+1} $(2 \le \ell \le m + n - i)$. Partition the vertices as follows: $L = \{p_1, \ldots, p_i\}$, $M = \{g_i(i+1), \ldots, g_i(i+\ell-1)\}$, $R = \{g_i(i+\ell+1), \ldots, g(m+n+1)\}$. Thus the arrangement g_i is $L\,M\,p_{i+1}\,R$.

Let W_{MR} be the total weight of all the edges going from M to R and W_{RM} be the total weight of all edges going from R to M. Define

$$g_{i+1} = \begin{cases} L\,p_{i+1}\,M\,R & \text{if } W_{MR} \ge W_{RM} \\ L\,p_{i+1}\,R\,M & \text{if } W_{MR} < W_{RM} \end{cases}$$

The verification that $\mathrm{MCOST}(g_{i+1}) \le \mathrm{MCOST}(g_i)$ is left to the reader. □

Note that $g = p_1 p_2 \ldots p_{m+1} \circ \mathrm{ROT}(f|G)$, where $f|G$ is the arrangement f restricted to vertices in G. Hence this transformation can be implemented in time $O(m + n^3)$. A generalization of this is proved in [18].

Proposition 6 (Lemma 3.4 of [18]). *Let $G = H_1 + \cdots + H_k$ be a directed graph with k components. Then there is an optimal circular arrangement of G which can be obtained by concatenating (not necessarily optimal) circular arrangements of H_i.*

We now have a corollary of Theorem 2 which gives us a technique to force an optimal MCA arrangement to have only forward edges.

Corollary 2. *Let G be an un-weighted directed acyclic graph, $m \ge \mathrm{DCOST}(G)$. Let g be the circular arrangement obtained by concatenating P_m with the optimal DOLA arrangement of G. Then $\mathrm{MCOST}(G + P_m) = \mathrm{DCOST}(G) + m = \mathrm{MCOST}(g)$, i.e. g is an optimal circular arrangement.*

Proof (sketch). First note that $\mathrm{MCOST}(g) = \mathrm{DCOST}(G) + m \le 2m$. Moreover, G cannot have any backward edges in the *straightened* optimal circular arrangement of $G + P_m$, for that implies the cost of the arrangement is $> 2m$. □

We conclude this section with a couple of NP Completeness proofs of un-weighted MCA.

Theorem 3. *The decision version of the un-weighted MCA problem is NP-complete.*

Proof. Proof by reduction from un-weighted DOLA. Let (G, K) be a DOLA instance. Let $m = n^3$ be an upper bound for cost of optimal DOLA arrangement. By Corollary 2, $\mathrm{MCOST}(G + P_m) = DOLA(G) + m$. So if $G' = G + P_m$ and $K' = K + m$, we have $DOLA(G) \le K \iff \mathrm{MCOST}(G + P_m) \le K'$. □

Since the MCA instance in this proof has $|E| = O(|V|)$, we infer that un-weighted MCA is NP-complete even when restricted to sparse graphs. We now prove a generalization of Corollary 2 and use it to show that un-weighted MCA is NP-complete even when restricted to dense instances.

Proposition 7. *Let G and H be un-weighted directed acyclic graphs such that $|V(H)| = m \geq \mathrm{DCOST}(G)$. Assume further that there is an optimal MCA arrangement h of H which does not contain any backward edges. Let g be the circular arrangement obtained by concatenating h with the optimal DOLA arrangement of G. Then $\mathrm{MCOST}(G+H) = \mathrm{DCOST}(G) + \mathrm{DCOST}(H) = \mathrm{MCOST}(g)$, i.e. g is an optimal circular arrangement.*

Proof (sketch). First apply Proposition 6 to separate out vertices of G and H in the optimal arrangement. Then rearrange the H portion to be h, since h has no backward edges. Then proceed as in Corollary 2 to show that G cannot have any backward edges. □

Theorem 4. *Decision version of the un-weighted MCA is NP-complete even when restricted to dense instances.*

Proof (sketch). Proof by reduction from un-weighted DOLA. Let (G, K) be a DOLA instance. Let $m = n^3$ be an upper bound for cost of optimal DOLA arrangement. Then consider MCA instance $G' = G + \overrightarrow{K}_m$, where $\overrightarrow{K}_m$ is the complete DAG on m vertices, and use proposition 7! □

6 MAX-SNP[π] and PTAS

Papadimitriou and Yannakakis [21] show that the complementary version of OLA, called Maximum Linear Arrangement, is in MAX-SNP[π]. We show that same is true for the following complementary version of MCA as well.

MaxCA Given a directed graph G, find an arrangement f that maximizes $\mathrm{MCOST}(f)$.

Theorem 5. *MaxCA is in MAX-SNP[π].*

Proof (sketch). To show that MaxCA is in MAX-SNP[π], consider the first-order quantifier-free predicate $\psi(\pi, u, w, v, G) := B(u, w, v) \wedge ((u, v) \in E(G))$, where $B(u, w, v)$ is true when $w = v$, or w occurs in between u and v in π order (considered cyclically). □

6.1 MAX-SNP[π] Completeness of MaxCA

We prove that MaxCA is complete for MAX-SNP[π] by showing a L-reduction [21] from the following MAX-SNP[π] complete problem. This problem is, in fact, the complementary version of minimum feedback arc set problem [13] and it does not admit a PTAS unless P=NP as shown in [5].

MAX SUBDAG: Given a directed graph $G = (V, E)$, find a subset $E' \subseteq E$ of maximum cardinality for which (V, E') is acyclic.

In [12], we prove the following theorem.

Theorem 6. *MaxCA is MAX-SNP[π] complete.*

6.2 MCA is MAX-SNP[π] Hard

We now show that MCA cannot have a PTAS by showing a similar reduction from MAX SUBDAG. In fact, this reduction can be easily modified into a L-reduction. It, then, proves that MCA is MAX-SNP[π]-Hard problem. But we prefer to put the proof in algorithmic form, since our main goal is to prove that MCA has no PTAS unless P=NP.

Theorem 7. *MCA does not have a PTAS unless P=NP.*

Proof (sketch). Suppose $\mathcal{M}$ is a $(1+\epsilon)$-approximation algorithm for MCA for some $0 < \epsilon < 1/3$.

Require: A directed graph G.

Ensure: $F \subseteq E$ for which (V, F) is acyclic.

$m \Leftarrow \frac{4n}{\epsilon}$.

$G' \Leftarrow G + P_m$.

$f \Leftarrow \mathcal{M}(G')$.

$f' \Leftarrow \mathcal{SA}(f)$. ($\mathcal{SA}$ is the straightening algorithm)

$F \Leftarrow$ edges of G which are forward edges in the arrangement f'.

Output F.

It can now be shown that $|F| \geq (1 - 3\epsilon)|F^*|$, where $F^* \subseteq E$ denotes the optimal solution to the MAX SUBDAG problem. This gives us a $(1 - 3\epsilon)$-approximation for MAX SUBDAG. Thus a PTAS for MCA gives a PTAS for the MAX SUBDAG. In the light of [5], this implies P=NP. □

7 Hardness of Approximation

We now turn to hardness of approximation. We use the straightening algorithm to prove a curious hardness result for MCA.

Proposition 8. *Suppose that un-weighted DOLA has a polynomial time α-approximation algorithm and un-weighted MCA has a polynomial time $(1+\delta)$-approximation algorithm ($\delta < 1$). Then un-weighted DOLA has a polynomial time $\mu(\alpha)$-approximation algorithm, where*

$$\mu(p) = (1+\delta) + \frac{\delta(1+\delta)}{1-\delta}\, p.$$

Proof (sketch). Let $\mathcal{D}$ denote the α-approximation algorithm for un-weighted DOLA, and $\mathcal{M}$ denote the $(1 + \delta)$-approximation algorithm for un-weighted MCA. Let $\mathcal{SA}$ denote the straightening algorithm of Theorem 2. Put $\theta = (1 + \delta)/(1 - \delta)$.

Require: Input un-weighted directed graph G.
Ensure: g is a β-approximate DOLA arrangement of G.

```
f ⇐ D(G).
m ⇐ θ · DCOST(f).
g1 ⇐ M(G + P_m).
g2 ⇐ SA(g_1).
g ⇐ g2 restricted to G.
Output g.
```

It can be shown that $\mathrm{DCOST}(G) \leq \mathrm{DCOST}(\mathtt{g}) \leq \mu(\alpha) \cdot \mathrm{DCOST}(G)$. □

One can view the above algorithm as a way of generating a $\mu(\alpha)$-approximate arrangement given the cost (we don't need the arrangement) of an α-approximate arrangement. This leads to the following theorem. See [12] for complete proof.

Theorem 8 (Bootstrapping). *Suppose that un-weighted MCA has a polynomial time $(1 + \delta)$-approximation algorithm for some $\delta < \sqrt{2} - 1$. Put $\Gamma(\delta) = 1 + \frac{2\delta}{1-2\delta-\delta^2}$. Then for every $\epsilon > 0$, there is a polynomial time $(\Gamma(\delta) + \epsilon)$-approximation algorithm for un-weighted DOLA.*

As a corollary we have the following conditional hardness result.

Corollary 3. *For all $\eta \in (0, \sqrt{2} - 1)$, there is a constant c_η such that it is NP-hard to approximate un-weighted MCA to within $\sqrt{2} - \eta$ if it is NP-hard to approximate un-weighted DOLA within c_η.*

8 Polynomial Time Approximation Scheme

We conclude with a PTAS for un-weighted MCA on dense graphs. Arora, Frieze and Kaplan [4] give a PTAS for OLA on dense graphs. We show how the same algorithm with minor modifications works for MCA as well. The algorithm gives an arrangement which is at most ηn^3 away from the optimal solution. If the graph is dense, then Proposition 3 shows that the optimum value is $\Omega(n^3)$.

Definition 5. *For constant t, let $I_1, \ldots, I_t$ be a partition of $[n] := \{1, \ldots, n\}$ into consecutive equal sized intervals, such that $I_i = \{it, \ldots, (i+1)t - 1\}$. A **placement** is a mapping from the vertex set to the set $\{I_1, \ldots, I_t\}$. A placement f' is **proper** if $|f'^{-1}(I_i)| = |I_i|$ for each i. Given any mapping $f : V \mapsto [n]$, we denote by f' the induced placement. The cost of a placement f', denoted by $\mathrm{CP}(f')$ is defined to be $\sum_{(u,v)\in E}(f'(v) - f'(u) \mod t)$.*

Proposition 9. *If f is any arrangement, $|\mathrm{MCOST}(f) - \mathrm{CP}(f')n/t| \leq n^3/t$.*

Proof (sketch). Consider any edge which crosses an interval. If it has latency x w.r.t the arrangement f, then it has latency $\lfloor xt/n \rfloor$ or $\lfloor xt/n{+}1 \rfloor$ in the placement f'. This observation together with a generous upper bound on the number edges within an interval gives the result. □

Proposition 10. *If f and g are arrangements such that $|\,\mathrm{CP}(f') - \mathrm{CP}(g')| \leq \epsilon n^2$, then $|\,\mathrm{MCOST}(f) - \mathrm{MCOST}(g)| \leq (2+\epsilon)n^3/t$.*

Now proceed just like in [4]. See proof details in [12].

9 Discussion

We studied the MCA problem in this paper. Its motivation came from a problem related to design of cyclic multicast schedule. Considering current trend in technologies and applications, cyclic multicast that pays heed to data dependencies should play a pivotal role in the future [6].

Our research pointed out certain negative aspects of the MCA problem, namely, it does not have a polynomial time algorithm and it does not even admit a polynomial time approximation scheme for arbitrary graph instance (unless P=NP). Yet it is possible that MCA problem might be tenable if restricted to certain special kinds of graphs that have practical significance. Literature has many such instances of polynomial time algorithms for OLA problem, *e.g.*, unweighted trees [25, 8], outer planar graphs [10], wheels, complete bipartite graphs [16], etc. Can one hope for the same in case of MCA? Or is it also *too hard*?

Assuming DOLA to be non-approximable within any constant factor, we could show a lower bound of $\sqrt{2} - \epsilon$ for MCA approximation. We believe it to be far from being tight. In fact, there is a conspicuous lack of hardness of approximation results even for OLA and DOLA. They stand as natural open problems.

Liberatore provides few heuristics [19, 18] and a $\widetilde{O}(\sqrt{n})$-approximation algorithm [18] to solve MCA problem on arbitrary graphs. Similarly Naor and Schwartz present $O(\log n \log\log n)$-approximation algorithm in [20]. But these algorithms suffer either from no performance guarantee or from inherent inefficiency. Therefore it is an interesting open question to design an efficient approximation algorithm for MCA problem.

References

1. S Acharya. *Broadcast Disks: Dissemination-based Data Management for Assymetric Communication Environments.* PhD thesis, Brown University, May 1998.
2. D Adolphson and T C Hu. Optimal linear ordering. *SIAM Journal on Applied Mathematics*, 25(3):403–423, 1973.
3. K C Almeroth, M H Ammar, and Z Fei. Scalable delivery of web pages using cyclic best-effort multicast. In *IEEE INFOCOM*, pages 1214–1221, March 1998.

4. S Arora, A Frieze, and H Kaplan. A new rounding procedure for the assignment problem with applications to dense graph arrangement problems. In *37th Annual IEEE Symposium on Foundations of Computer Science*, pages 21–30, October 1996.
5. S Arora, C Lund, R Motwani, M Sudan, and M Szegedy. Proof verification and hardness of approximation problems. In *33rd Annual IEEE Symposium on Foundations of Computer Science*, pages 14–23, October 1992.
6. P K Chrysanthis, V Liberatore, and K Pruhs. Middleware support for multicast-based data dissemination: A working reality, 2001. White paper.
7. F R K Chung. *Theory and Applications of Graphs*, chapter Some Problems and Results in Labelings of Graphs, pages 255–264. John Wiley & Sons, New York, 1981.
8. F R K Chung. On optimal linear arrangements of trees. *Comp. & Maths with Applications*, 10(1):43–60, 1984.
9. S Even and Y Shiloach. NP-Completeness of several arrangement problems. Technical Report 43, Isreal Institute of Technology, 1975.
10. G N Frederickson and S E Hambrusch. Planar linear arrangements of outerplanar graphs. *IEEE Transactions on Circuits and Systems*, 35(3):323–332, 1988.
11. A M Frieze and R Kannan. Quick approximation to matrices and applications. *Combinatorica*, 19(2):175–220, 1999.
12. M K Ganapathy and S Lodha. On minimum circular arrangement, 2003. URL: http://www.research.rutgers.edu/˜lodha/research/ps/mca.ps.
13. M R Garey and D S Johnson. *Computers and Intractability: A guide to the theory of NP-Completeness*. W H Freeman and Company, 2nd edition, 1979.
14. M R Garey, D S Johnson, and L Stockmeyer. Some simplified NP-Complete graph problems. *Theoretical Computer Science*, 3(1):237–267, 1976.
15. G Herman, G Gopal, K C Lee, and A Weinrib. The datacycle architecture for very high throughput data systems. In *ACM SIGMOD International Conference on Management of Data*, pages 97–103, May 1987.
16. M Juvan and B Mohar. Optimal linear labelings and eigenvalues of graphs. *Discrete Applied Mathematics*, 36:153–168, 1992.
17. B Krishnamurthy and J Rexford. *Web Protocols and Practice*. Addison-Wesley, Boston, 2001.
18. V Liberatore. Circular arrangements. In *ICALP*, pages 1054–1066, 2002.
19. V Liberatore. Multicast scheduling for list requests. In *IEEE INFOCOM*, pages 1129–1137, June 2002.
20. J Naor and R Schwartz. The directed circular arrangement problem. In *15th Annual ACM-SIAM Symposium on Discrete Algorithms*, January 2004. To appear.
21. C H Papadimitriou and M Yannakakis. Optimmization, approximation and complexity classes. *Journal of Computer and System Sciences*, 43:425–440, 1991.
22. M Rabinovich. Resource management issues in content delivery networks (CDNs). In *DIMACS Workshop on Resource Management and Scheduling in Next Generation Networks*, 2001.
23. S Rao and A W Richa. New approximation techniques for some ordering problems. In *9th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 211–218. ACM-SIAM, January 1998.
24. J Shanmugasundaram, A Nithrakashyap, R Sivasankaran, and K Ramamritham. Efficient concurrency control for broadcast environments. In *ACM SIGMOD International Conference on Management of Data*, pages 85–96, June 1999.
25. Y Shiloach. A minimum linear arrangement algorithm for undirected trees. *SIAM Journal of Computing*, 8(1):15–32, 1979.