Minimum-Latency Aggregation Scheduling in Multihop Wireless Networks

Peng-Jun Wan*
Department of Computer Science
Illinois Institute of Technology
Chicago, IL 60616
wan@cs.iit.edu

Scott C.-H. Huang†
Department of Computer Science
City University of Hong Kong
Kowloon, Hong Kong
shuang@cityu.edu.hk

Lixin Wang
Department of Computer Science
Illinois Institute of Technology
Chicago, IL 60616
wanglix@iit.edu

Zhiyuan Wan
Department of Computer Science
City University of Hong Kong
Kowloon, Hong Kong
zhiyuwan@cityu.edu.hk

Xiaohua Jia‡
Department of Computer Science
City University of Hong Kong
Kowloon, Hong Kong
jia@cs.cityu.edu.hk

ABSTRACT

Minimum-latency aggregation schedule (MLAS) in synchronous multihop wireless networks seeks a shortest schedule for data aggregation subject to the interference constraint. In this paper, we study MLAS under the protocol interference model in which each node has a unit communication radius and an interference radius $\rho \geq 1$. All known aggregation schedules assumed $\rho = 1$, and the best-known aggregation latency with $\rho = 1$ is $23R + \Delta - 18$ where R and Δ are the radius and maximum degree of the communication topology respectively. In this paper, we first construct three aggregations schedules with $\rho = 1$ of latency $15R + \Delta - 4$, $2R + O(\log R) + \Delta$ and $\left(1 + O\left(\log R / \sqrt[3]{R}\right)\right) R + \Delta$ respectively. Then, we obtain two aggregation schedules with $\rho > 1$ by expanding the first two aggregation schedules with $\rho = 1$. Both aggregation schedules with $\rho > 1$ have latency within constant factors of the minimum aggregation latency.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*wireless communication*; F.2.0 [**Theory of Computation**]: Analysis of Algorithms and Problem Complexity—*General*

General Terms

Algorithms, Theory

Keywords

Aggregation, communication latency, approximation algorithms

1. INTRODUCTION

Data aggregation in multihop wireless networks is a primitive communication task in which a distinguished sink node collects a packet from every other node and every intermediate node combines all received packets with its own packet into a single packet of fixed-size according to some aggregation function such as logical and/or, maximum, or minimum. A routing for an aggregation is a spanning inward arborescence of the communication topology rooted at the sink of the aggregation. Assume that all communications proceed in synchronous time-slots and each node can transmit at most one packet of a fixed size in each time-slot. A link schedule of an spanning inward arborescence is an assignment of time-slots to all links in this arborescence subject to two constraints: (1) A node can only transmit after all its children complete their transmissions to itself; and (2) all links assigned in a common time-slot are interference-free. Thus, an *aggregation schedule* specifies not only a spanning in-arborescence for routing but also a link schedule of such spanning in-arborescence. The *latency* of an aggregation schedule is the number of time-slots during which at least one transmission occurs. The problem of computing an aggregation schedule with minimum latency in a multihop wireless network is referred to **Minimum-Latency Aggregation Schedule** (MLAS).

In this paper, we study the problem MLAS under the following model for wireless networks. All the networking

*This work was supported in part by NSF under grant CNS-0831831.

†This work was supported in part by the Research Grant Council of Hong Kong under the project CERG CityU 113807.

‡This work was partially supported by Research Grants Council of Hong Kong under Project No. CityU 114307 and by NSF China Grant No. 60633020.

MobiHoc'09, May 18–21, 2009, New Orleans, Louisiana, USA.

nodes are located in a plane and are each equipped with an omnidirectional antenna. Each node has a fixed transmission radius which is normalized to one and an interference radius $\rho \geq 1$. The communication range and the interference range of a node v are the two disks centered at v of radius one and ρ respectively (see Figure 1). Let V denote the set of networking nodes, and G be the unit-disk graph (UDG) on V. Then the communication topology of the network is the digraph $\overrightarrow{G}$ obtained from G by replacing any edge uv in G with two oppositely oriented links (u, v) and (v, u). A pair of communication links (u_1, v_1) and (u_2, v_2) in $\overrightarrow{G}$ are said to be conflict-free if the two line segments $u_1 v_2$ and $u_2 v_1$ are both longer than ρ. A subset of links in $\overrightarrow{G}$ scheduled in a same time-slot are inteference-free if they are pairwise conflict-free. Such interference model is referred to as the *protocol interference model* [3] and is widely used because of its generality and tractability.

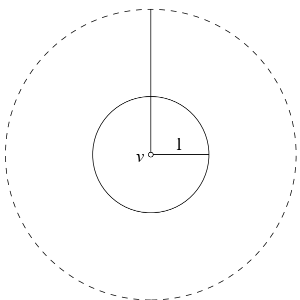

Figure 1: The protocol interference model: each node has a unit transmission radius and an interference radius $\rho \geq 1$.

MLAS with $\rho = 1$ is NP-hard [1]. Let n be the number of nodes, and the s be the sink node of the aggregation. The radius of G with respect to s, denoted by R, is the maximum (hop) distance between s and other node in G. Both R and $\log n$ are two lower bounds on the minimum aggregation latency regardless of ρ. For $\rho = 1$, two aggregation schedules of latency at most $(\Delta - 1)R$ and $23R + \Delta - 18$ respectively have been developed in [1] and [4] respectively, where Δ is the maximum degree of G. Note that Δ contributes to an *multiplicative* factor in the former aggregation schedule, while contributes to *additive* factor in the latter aggregation schedule. This paper makes the following contributions to MLAS:

- For $\rho = 1$, we develop three approximation algorithms which produce aggregations schedules of latency at most $15R + \Delta - 4$, $2R + \Delta + O(\log R)$ and $\left(1 + O\left(\log R / \sqrt[3]{R}\right)\right) R + \Delta$ respectively. The first one has the simplest implementation and may outperform the other two when the radius R is small. For large radius R, the latter two speed up the aggregation schedule by using the pipelining technique.
- For $\rho > 1$, we develop two approximation algorithms which produce aggregations schedules of latency at most $\beta_{\rho+1}(15R + \Delta - 4)$ and $\beta_{\rho+1}(2R + \Delta + O(\log R))$ respectively, where
$$\beta_r = \frac{\pi}{\sqrt{3}} r^2 + \left(\frac{\pi}{2} + 1\right) r + 1.$$
We also prove that both algorithms have constant approximation ratios.

The key ingredients of the three approximation algorithms with $\rho = 1$ are a special inward arborescence and two novel connected dominating sets. The inward arborescence is associated with a properly defined link labelling and node ranking. It enables the application of the pipelining technique for speeding up the aggregation schedule. In any aggregation routing, the set of relaying nodes together with the sink node s form a connected dominating set (CDS) of G. For achieving shorter aggregation latency, the CDS should have stronger structural properties such as graph radius and maximum degree than small size only. The constructions of these structures are presented in Section 3 and Section 4 respectively. The three approximation algorithms with $\rho = 1$ are then described in Section 5. The two approximation algorithms with $\rho > 1$ exploits a generic expansion technique which adapts a "well-separated" communication schedule with $\rho = 1$ to a communication schedule with $\rho > 1$. Such expansion technique is described in Section 6. We expect that these structures and the expansion technique can be applied in the scheduling of other communications.

2. PRELIMINARIES

In this section, we first introduce some standard graph-theoretic terms and notations adopted throughout this paper. Let $G = (V, E)$ be a connected graph. The subgraph of G induced by a subset U of V is denoted by $G[U]$, and the bipartite subgraph of G induced by two disjoint subsets U and W of V is denoted by $G[U, W]$. The maximum (respectively, minimum) degree of G is denoted by $\Delta(G)$ (respectively, $\delta(G)$). The *inductivity* of G is defined by
$$\delta^*(G) = \max_{U \subseteq V} \delta(G[U]).$$
The graph distance between any two nodes u and v in G is denoted by $dist_G(u, v)$. The radius of G with respect to a specific node $v \in V$ is denoted by $Rad(G, v)$. Now, fix a node $s \in V$. The *depth* of a node v (with respect to s) is $dist_G(s, v)$. For each $0 \leq i \leq Rad(G, s)$, the set of nodes in V of depth i is referred to as the *i-th layer* of G.

A subset U of V is an *independent set* of G if no two nodes in U are adjacent. If U is a independent set of G but no proper superset of U is a independent set of G, then U is called a *maximal independent set* (MIS) of G. Any node ordering $v_1, v_2, \cdots, v_n$ of V induces an MIS U in the following first-fit manner: Initially, $U = \{v_1\}$. For $i = 2$ up to n, add v_i to U if v_i is not adjacent to any node in U. A subset U of V is a *dominating set* of G if each node not in U is adjacent to some node in U. Clearly, every MIS of G is also a dominating set of G. If U is a dominating set of G and $G[U]$ is connected, then U is called a *connected dominating set* (CDS) of G.

Consider an ordering $\langle v_1, v_2, \cdots, v_n \rangle$ of V. For each $1 < i \leq n$, let V_i denote the set of nodes v_j with $1 \leq j < i$ adjacent to v_i. The *inductivity* of the ordering $\langle v_1, v_2, \cdots, v_n \rangle$ is defined to be $\max_{1<i\leq n} |V_i|$. A natural question is whether a vertex ordering of the smallest inductivity can be computed in polynomial time. The answer to this question is positive. A special vertex ordering, known as *smallest-degree-last ordering* [5], achieves the smallest inductivity. It is produced iteratively as follows: Initialize H to G. For

$i = n$ down to 1, let v_i be a vertex of the smallest degree in H and delete v_i from H. Then the ordering $\langle v_1, v_2, \cdots, v_n \rangle$ is a smallest-degree-last ordering. The following theorem was proven in [5].

THEOREM 2.1. *The smallest-degree-last ordering achieves the smallest inductivity $\delta^*(G)$ among all vertex orderings.*

A vertex coloring of G is an assignment of colors to V satisfying that adjacent vertices are assigned with distinct colors. Given a vertex ordering $\langle v_1, v_2, \cdots, v_n \rangle$ of V, a coloring of V with colors represented by natural numbers can be produced in the following first-fit manner: Assign the color 1 to v_1. For $i = 2$ up to n, assign to v_i with the smallest color which is not used by any neighbor of v_i which precedes v_i. Such coloring of V is referred to as the *first-fit coloring* in the ordering $\langle v_1, v_2, \cdots, v_n \rangle$. It's easy to see the number of colors used by the first-fit coloring in the ordering $\langle v_1, v_2, \cdots, v_n \rangle$ is no more than one plus the inductivity of the ordering $\langle v_1, v_2, \cdots, v_n \rangle$. In particular, the first-fit coloring in smallest-degree-last ordering uses at most $1+\delta^*(G)$ colors.

Let X and Y be two disjoint subsets of V. Y is a *cover* of X if each node in X is adjacent to some node in Y, and a *minimal cover* of X if Y is a cover of X but no proper subset of Y is a cover of X. Any ordering $y_1, y_2, \cdots, y_m$ of Y induces a minimal cover $W \subseteq Y$ of X by the following sequential pruning method: Initially, $W = Y$. For each $i = m$ down to 1, if $W \setminus \{y_i\}$ is a cover of X, remove y_i from W. Suppose that Y is a cover of X. A node $x \in X$ is called a *private* neighbor of a node $y \in Y$ with respect to Y if y is the only node in Y which is adjacent to x. Clearly, if Y is a minimal cover of X, then each node in Y has at least one private neighbor with respect to Y.

In the remaining of this section, we introduce a classic geometric result on disk packing.

THEOREM 2.2 (GROEMER INEQUALITY [2]). *Suppose that C is a compact convex set and U is a set of points with mutual distances at least one. Then*

$$|U \cap C| \leq \frac{area\,(C)}{\sqrt{3}/2} + \frac{peri\,(C)}{2} + 1,$$

where $area\,(C)$ and $peri\,(C)$ are the area and perimeter of C respectively.

When the set C is a disk or a half-disk, we have the following packing bound.

COROLLARY 2.3. *Suppose that C (respectively, C') is a disk (respectively, half-disk) of radius r, and U is a set of points with mutual distances at least one. Then*

$$|U \cap C| \leq \frac{2\pi}{\sqrt{3}} r^2 + \pi r + 1,$$

$$|U \cap C'| \leq \frac{\pi}{\sqrt{3}} r^2 + \left(\frac{\pi}{2} + 1\right) r + 1.$$

3. CANONICAL INWARD ARBORESCENCE

Let $G = (V, E)$ be a connected undirected graph and s be a distinguished node in V. In this section, we present a spanning inward s-arborescence of $\overrightarrow{G}$, which is associated with a link labelling and node ranking. The arborescence itself would be utilized later in aggregation routing, and the associated link labelling and node ranking will be utilized in the link scheduling. Such arborescence is referred to as a canonical inward arborescence.

We begin with a key building block of the construction algorithm. Two links (u_1, v_1) and (u_2, v_2) in $\overrightarrow{G}$ are said to be conflicting if at least one of $u_1 v_2$ and $u_2 v_1$ is an edge in G. A subset A of links in $\overrightarrow{G}$ is said to be conflict-free if any pair of links in A are not conflicting. Suppose that X and Y are two disjoint subsets of V and X is covered by Y. A single-hop (X, Y)-aggregation schedule consists of a set A of links in $\overrightarrow{G}$ and a labeling of the links in A by natural numbers satisfying that (1) for each link $a = (x, y) \in A$, $x \in X$ and $y \in Y$; (2) each node in X is the tail of exactly one link in A; (3) all the links in A with the same label are conflict-free. For each link $(x, y) \in A$, x is said to be a child of y while y is said to be a parent of x. Table 1 outlines an algorithm called *iterative minimal covering* (**IMC**). It takes as input a pair (X, Y) of disjoint subsets X and Y of V satisfying that X is covered by Y and outputs a single-hop (X, Y)-aggregation schedule.

IMC:
$A \leftarrow \emptyset, l \leftarrow 0,\ X' \leftarrow X,\ Y' \leftarrow Y$;
while $X \neq \emptyset$,
$C \leftarrow$ a minimal cover of X' contained in Y';
for each $y \in C$,
$x \leftarrow$ a private neighbor of y in X',
$A \leftarrow A \cup \{(x, y)\}$;
$\ell\,(x, y) \leftarrow l$;
$X' \leftarrow X' \setminus \{x\}$;
$Y' \leftarrow C$;
output A and ℓ.

Table 1: Outline of the algorithm IMC.

Figure 2 is an illustration of the algorithm **IMC**. In this example, $X = \{x_i : 1 \leq i \leq 7\}$ and $Y = \{y_i : 1 \leq i \leq 5\}$. Their adjacency is depicted in Figure 2 (a). In the first iteration, y_2, y_3, y_5 form the minimal cover, and x_1, x_4, x_6 are their private neighbors respectively (see Figure 2 (b)). So, the three links $(x_1, y_2), (x_4, y_3)$ and (x_6, y_5) are added to A and all receive the label one. After that we remove x_1, x_4, x_6 (see Figure 2 (c)) and proceed to the second iteration. In the second iteration, y_2, y_5 form the minimal cover, and x_2, x_5 are their private neighbors respectively (see Figure 2 (d)). So, the two links (x_2, y_2) and (x_5, y_5) are added to A and both receive the label two. After that we remove x_2, x_5 (see Figure 2 (e)) and move on to the third iteration. In the third iteration, the two links (x_3, y_2) and (x_7, y_5) are added to A and both receive the label three (see Figure 2 (f)). This is the last iteration as every node has been assigned as a parent. Figure 2 (g) shows all the links in A together with their labels.

LEMMA 3.1. *Let A and ℓ be the output by **IMC**. For each link $(x, y) \in A$, $\ell\,(x, y)$ is no more than the number of children of y, i.e.,*

$$\ell\,(x, y) \leq \left|\{x' \in X : (x', y) \in A\}\right|.$$

PROOF. Suppose the algorithm runs in L iterations. For each $1 \leq l \leq L$, let C_l be the minimum cover C computed

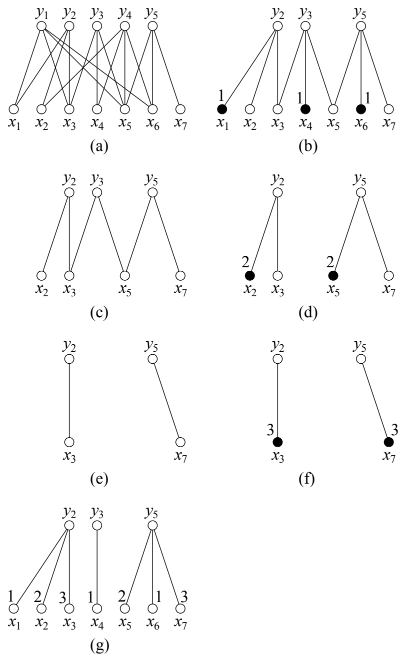

Figure 2: An illusrtation of the algorithm IMC.

in the l-th iteration. Then,

$$C_1 \supseteq C_2 \supseteq \cdots \supseteq C_L.$$

Consider an arc $a = (x, y) \in A$. Let k be the largest index such that $y \in C_k$. Then, $\ell(x, y) \leq k$ and y has exactly k children, each from one of the first k iterations. So, the lemma follows. □

CBFS:
$R \leftarrow$ radius of G w.r.t. s;
for each $0 \leq i \leq R$, $V_i \leftarrow \{v \in V : dist_G(v, s) = i\}$;
$R \leftarrow (V, \emptyset)$;
for each $u \in V_R$, $rank(u) \leftarrow 0$;
for each $i = R$ down to one
  $J \leftarrow \{rank(v) : v \in V_{i-1}\}$
  for each $j \in J$
    $V_{ij} \leftarrow \{v \in V_i : rank(v) = j\}$;
    augment T by applying **IMC** on V_{ij} and V_{i-1};
  for each $u \in V_{i-1}$,
    if u has no child, $rank(u) \leftarrow 0$;
    else
      $r \leftarrow$ maximum rank of the children of u;
      if only one child of u has rank r, $rank(u) \leftarrow r$;
      else $rank(u) \leftarrow r + 1$;
output T and $rank$.

Table 2: Outline of the algorithm IMC.

Next, we apply the algorithm **IMC** to construct a canonical inward s-arborescence T. Our algorithm **CBFS** is outlined in Table 2. Let R be the radius of G with respect to s. For each $0 \leq i \leq R$, let V_i be the set of nodes in V of depth i. The construction is in the bottom-up manner. Initially, T is empty and $rank(v) = 0$ for each node v in the bottom layer. For each layer i from R down to one, we first compute the links from layer V_i to V_{i-1} and their associated labeling by using the algorithm **IMC**. Specifically, let J be the set of ranks of the nodes in V_i. For each $j \in J$, let V_{ij} be the set of nodes in V_i with rank j and apply the algorithm **IMC** on (V_{ij}, V_{i-1}) to augment T. After that, we compute the ranks of all nodes in V_{i-1} in a standard manner. For each node u in V_{i-1},we assign the ranks as follows. If u has no child, $rank(u)$ is set to zero. If u has at least one child, let r be the maximum rank of its children. If u has only one child of rank r, then $rank(u)$ is set to r; otherwise $rank(u)$ is set to $r + 1$. Figure 3 is an example of the canonical inward arborescence output by the algorithm **CBFS**.

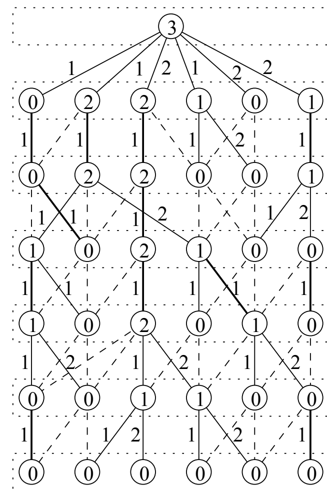

Figure 3: A canonical inward arborescence and the associated node ranking produced by the algorithm CBFS.

The arborescence T and the associated ranking have a number of interesting properties. Clearly, each node has rank no more than its parent in T. It's also easy to prove by induction in the bottom-up manner that for each node v, $rank(v) \leq \lfloor \log |T_v| \rfloor$, where T_v is the subtree of T induced by v and all its descendents. In particular, for each node v, $rank(v) \leq \lfloor \log |V| \rfloor$. A link in the canonical BFS tree is said be an *express link* if its two endpoints have the same rank. By Lemma 3.1, all express links are labelled with *one*.

4. CONNECTED DOMINATING SETS

Let $G = (V, E)$ be a connected UDG and s be the sink node of aggregation. The problem of computing a minimum CDS of G has been well-studied. While it is NP-hard, it admits constant approximations. However, for our later application in the aggregation scheduling, a CDS of small size is not sufficient. In this section, we construct two CDS's with stronger properties. Both constructions follow a general two-phased approach [6]. The first phase constructs a dominating set, and the second phase selects additional nodes, called connectors, which together with the dominators induce a connected topology. The two algorithms have the same first phase, which selects an MIS U induced by a breadth-first-search (BFS) ordering with respect to s as the dominating set. By Corollary 2.3,

$$|U| \leq \frac{2\pi}{\sqrt{3}} R^2 + \pi R + 1,$$

In the next two subsections, we describe the selections of connectors in the second phase.

4.1 The First Set of Connectors

Let H be the graph on U in which there is an edge between two dominators if and only if they have a common neighbor. Then, H is connected and $Rad(H,s) \leq R-1$. For each $0 \leq i \leq Rad(H,s)$, let U_i be the set of dominators of depth i in H. Then, $U_0 = \{s\}$. For each $0 \leq i < Rad(H,s)$, let P_i be the set of nodes adjacent to at least one node in U_i and at least one node in U_{i+1}, and compute a minimal cover $W_i \subseteq P_i$ of U_{i+1}. Set

$$W = \bigcup_{i=0}^{Rad(H,s)-1} W_i.$$

Then, $G[U,W]$ is connected and $U \cup W$ is a CDS of G. We refer to all nodes in W as *connectors* and all nodes not in $U \cup W$ as *dominatees*.

Clearly, $|W| \leq |U| - 1$ and hence

$$\begin{aligned} |U \cup W| &\leq 2|U| - 1 \\ &\leq 2\left(\frac{2\pi}{\sqrt{3}} R^2 + \pi R + 1\right) - 1 \\ &= \frac{4\pi}{\sqrt{3}} R^2 + 2\pi R + 1. \end{aligned}$$

Furthermore,

$$Rad(G[U,W], s) = 2Rad(H,s) \leq 2(R-1).$$

The lemma below presents some additional properties of the output CDS.

LEMMA 4.1. *The following statements are true.*

1. *For each $0 \leq i < Rad(H,s)$, each connector in W_i is adjacent to at most 4 dominators in U_{i+1}.*
2. *For each $1 \leq i < Rad(H,s) - 1$, each dominator in U_i is adjacent to at most 11 connectors in W_i.*
3. $|W_0| \leq 12$.

PROOF. The first part of the lemma follows from the fact that every node is adjacent to at most five independent nodes. We prove the second part by contradiction. Assume to the contrary that some dominator $u \in U_i$ is adjacent to $k \geq 12$ nodes $w_1, w_2, \cdots, w_k$ in W_i. By the minimality of W_i, for each $1 \leq j \leq k$ there is a node $v_j \in U_{i+1}$ such that v_j is adjacent to w_j but not to any other node in W_i. Let v_0 be a dominator in U_{i-1} which is adjacent to u in H, and w_0 be the node which is adjacent to both v_0 and u. Then, all these $k+1$ nodes $v_0, v_1, \cdots, v_k$ are distinct, and so are these $k+1$ nodes $w_0, w_1, \cdots, w_k$. In addition, for each $0 \leq j \leq k$, v_j is the only node in $\{v_0, v_1, \cdots, v_k\}$ which is adjacent to w_j. Among the $k+1$ nodes $v_0, v_1, \cdots, v_k$, there exist two, say v_{j_1} and v_{j_2}, satisfying that $\angle v_{j_1} u v_{j_2} \leq \frac{2\pi}{13}$. Denote by $B(x)$ the disk of unit radius centered at x. Since the distance between v_{j_1} and v_{j_2} is greater than one, either $B(u) \cap B(v_{j_1}) \subseteq B(v_{j_2})$ or $B(u) \cap B(v_{j_2}) \subseteq B(v_{j_1})$ (see Lemma 4 in [7]). In the former case, $w_{j_1} \in B(v_{j_2})$, and hence v_{j_2} is adjacent to w_{j_1}, which is a contradiction. In the latter case, $w_{j_2} \in B(v_{j_1})$, and hence v_{j_1} is adjacent to w_{j_2}, which is again a contradiction. Thus, the second part of the lemma holds. By the same argument, we can show that the third part of the lemma holds. □

4.2 The Second Set of Connectors

In this section, we present another set W of connectors such that $G[U \cup W]$ has shorter radius at the expense of higher maximum degree and larger $|W|$. Fix a positive integer parameter k. Let T be a BFS tree of G rooted at s. For each node v rather than s, we denote the parent of v in T by $p(v)$. In general, the node which is i hops away from v in the tree path from v to s is called the i-th ancestor of v and is denoted by $p^i(v)$. Since s is a dominator and is an ancestor of every other node, each node has at least one ancestor which is a dominator. Initialize W' to be empty. For each dominator u, let i be the smallest positive integer such that $p^i(u)$ is a dominator, and add each $p^j(u)$ with $1 \leq j \leq \min\{i-1, k\}$ to W'. Next, we compute the shortest-path tree T' from s to all other dominators in $G[U \cup W']$. In other words, all the leaves of T' are dominators. Let W be the subset of nodes in W' contained in T'. Then, $U \cup W$ is still a CDS. We refer to all nodes in W as *connectors* and all nodes not in $U \cup W$ as *dominatees*.

LEMMA 4.2. *The following three inequalities are true:*

$$\begin{aligned} |U \cup W| &\leq (k+1)\left(\frac{2\pi}{\sqrt{3}} R^2 + \pi R\right) + 1, \\ Rad(G[U \cup W], s) &\leq (1 + 1/k) R, \\ \Delta(G[U \cup W]) &\leq 2\sqrt{3}\pi k^2 + 3\pi k + 3 + 4\pi/\sqrt{3}. \end{aligned}$$

PROOF. For each dominator rather than s, at most k connectors are added to W'. Thus,

$$|W'| \leq k(|U| - 1).$$

Hence,

$$\begin{aligned} |U \cup W| &\leq |U \cup W'| \\ &\leq |U| + k(|U|-1) = (k+1)|U| - k \\ &\leq (k+1)\left(\frac{2\pi}{\sqrt{3}} R^2 + \pi R + 1\right) - k \\ &= (k+1)\left(\frac{2\pi}{\sqrt{3}} R^2 + \pi R\right) + 1. \end{aligned}$$

Now, we prove the second inequality in the lemma holds. Let H be the subgraph of G induced by $U \cup W$. It is sufficient to show that for each dominator u,

$$dist_H(u,s) \leq \left(1 + \frac{1}{k}\right) dist_G(u,s).$$

We prove this inequality by induction on $dist_G(u,s)$. Clearly, if $dist_G(u,s) \leq k$ then

$$dist_H(u,s) = dist_G(u,s).$$

So, we assume that $dist_G(u,s) > k$. Let i be the smallest integer such that $p^i(u)$ is a dominator. We consider two cases:

Case 1: $i \le k+1$. Then,

$$dist_G(u,s) = dist_G\left(p^i(u), s\right) + i.$$
$$dist_H(u,s) \le dist_H\left(p^i(u), s\right) + i.$$

By induction hypothesis,

$$\begin{aligned} dist_H(u,s) &\le dist_H\left(p^i(u), s\right) + i \\ &\le \left(1+\frac{1}{k}\right) dist_G\left(p^i(u), s\right) + i \\ &< \left(1+\frac{1}{k}\right)\left(dist_G\left(p^i(u), s\right) + i\right) \\ &= \left(1+\frac{1}{k}\right) dist_G(u,s). \end{aligned}$$

Case 2: $i > k+1$. If $p^k(v)$ is adjacent to some dominator at the same layer as $p^{k+1}(v)$, then using the same argument as in Case 1, we can show the inequality holds. So, we assume that $p^k(v)$ is not adjacent to some dominator at the same layer as $p^{k+1}(v)$. Then, $p^k(v)$ must be adjacent to some dominator v at the same layer as itself. Then,

$$dist_G(u,s) = dist_G(v,s) + k.$$
$$dist_H(u,s) \le dist_H(v,s) + k + 1.$$

By induction hypothesis,

$$\begin{aligned} dist_H(u,s) &\le dist_H(v,s) + k + 1 \\ &\le \left(1+\frac{1}{k}\right) dist_G(v,s) + (k+1) \\ &= \left(1+\frac{1}{k}\right)\left(dist_G(v,s) + k\right) \\ &= \left(1+\frac{1}{k}\right) dist_G(u,s). \end{aligned}$$

Finally, we prove the third inequality in the lemma. Each connector v in W must have a descendent in T which is a dominator, and we denote by $q(v)$ the descendant dominator of v which is closest to v. For each dominator v, we set $q(v)$ to v itself. Then, $dist_G(v, q(v)) \le k$. Consider a node $u \in U \cup W$. Let $N(u)$ denote the set of nodes in $U \cup W$ adjacent to u, and let

$$S(u) = \{q(v) : v \in N(u)\}.$$

Then each dominator in $S(u)$ is at most $k+1$ hops away from u in G. Now, let $S_1(u)$, $S_2(u)$ and $S_3(u)$ be the set of nodes in $S(u)$ which are at most $k-1$, k and $k+1$ hops away from u respectively. Notice that each node in $N(u)$ must be either at the same layer as u, or at the layer above u,or at the layer below u. Thus, for each u' in $S_1(u)$ (respectively, $S_2(u) \setminus S_1(u)$, $S_3(u) \setminus S_2(u)$), the set

$$\left\{v \in N(u) : q(v) = u'\right\}$$

consists of at most three (respectively, two, one) nodes. Consequently,

$$\begin{aligned} &|N(u)| \le 3|S_1(u)| + 2|S_2(u) \setminus S_1(u)| + |S_3(u) \setminus S_2(u)| \\ &= |S_1(u)| + |S_2(u)| + |S_3(u)|. \end{aligned}$$

By Corollary 2.3, we have

$$\begin{aligned} |N(u)| &\le \sum_{i=k-1}^{k+1}\left(\frac{2\pi}{\sqrt{3}}i^2 + \pi i + 1\right) \\ &= 2\sqrt{3}\pi k^2 + 3\pi k + 3 + \frac{4\pi}{\sqrt{3}}. \end{aligned}$$

Thus, the third inequality in the lemma holds. □

5. AGGREGATION SCHEDULING WITH $\rho = 1$

In this section, we present three aggregation scheduling algorithms with $\rho = 1$. All of them utilize a connected dominating set (CDS) for routing, which consisting of an MIS U induced by a BFS ordering (with respect to s) of V and a set W of connectors. The set W adopted by the first two schedules is the first set of connectors, and the set W adopted by the third schedule is the second set of connectors with an integer parameter $k = \Theta\left(\sqrt[3]{R}/\log R\right)$. The three schedules all consist of two phases. The first phase is a single-hop $(V \setminus (U \cup W), U)$-aggregation schedule, which can be constructed by applying the algorithm **IMC** presented in Section 3. Thus, the latency of the first phase is at most $\Delta - 1$. The second phase is an aggregation schedule in the graph $G[U \cup W]$. In the next, we describe the aggregation schedules for the second phase.

5.1 Sequential Aggregation Scheduling

Our first algorithm is called **Sequential Aggregation Scheduling** (**SAS**). Let W be the first set of connectors. We first construct an inward s-arborescence T on $U \cup W$ by specifying the parent $p(v)$ for each node v other than s. Let $R' = Rad(G[U, W], s)$. Then, R' is an even number no more than $2(R-1)$. For each $0 \le i \le R'$, we denote the set of nodes in the i-th layer of $G[U, W]$ by V_i'. Note that for even (respectively, odd) i, V_i' consists of dominators (respectively, connectors). For each $1 \le i \le R'$, each node $v \in V_i'$ sets its parent $p(v)$ to be the node of the smallest ID in V_{i-1}' which is adjacent to v. For each $1 \le i \le R'$, A_i denotes the set of links from the nodes in V_i' to their parents.

Our aggregation schedule proceeds in R' rounds, with the $(R'+1-i)$-th round devoted to the links in A_i for each $1 \le i \le R'$. Specifically, we sort all links in A_i in the increasing order of heads (i.e., receiving nodes) and break the ties with the increasing ordering of tails (i.e., transmitting nodes). Such ordering is referred to as *ID-lexicographic ordering* of A_i. The *conflict graph* of A_i, denoted by $CG(A_i)$, is an undirected graph on A_i in which there is an edge between each pair of conflicting links in A_i. We compute a first-fit coloring of $CG(A_i)$ in the ID-lexicographic ordering. Then, each link in A_i with color j is scheduled in the j-th time-slot of the $(R'+2-i)$-th round.

The next theorem gives an upper bound on the latency of the aggregation schedule produced **SAS**.

THEOREM 5.1. *Algorithm **SAS** produces an aggregation schedule with latency at most* $15R-3$.

We prove this theorem in the remaining of this subsection. For each $1 \le i \le R'$, we denote by δ_i^* the inductivity of the ID-lexicographic ordering in the graph $CG(A_i)$ and denote by Δ_i^* the maximum number of tails of the links in A_i adjacent in G to the head of some link in A_i.

Lemma 5.2. *For each* $1 \le i \le 2R'$, $\delta_i^* \le \Delta_i^* - 1$.

Proof. Suppose that $(u, p(u))$ and $(v, p(v))$ are two conflicting links in A_i and $(u, p(u))$ precedes $(v, p(v))$ in the ID-lexicographic ordering. We claim that u is adjacent to $p(v)$. This holds trivially if $p(u) = p(v)$. So, we assume that $p(u) \ne p(v)$. Then, $p(u)$ has smaller ID than $p(v)$. By the choice of parent, v is not adjacent to $p(u)$. Therefore, u must be adjacent to $p(v)$. So, our claim holds, from which the lemma follows immediately. □

The above lemma implies that the $(R' + 2 - k)$-th round takes at most Δ_i^* time-slots. By Lemma 4.1, for any $1 \le i \le R'$,

$$\Delta_i^* \le \begin{cases} 4 & \text{if } i \text{ is even;} \\ 11 & \text{if } i \text{ is odd and } i > 1; \\ 12 & \text{if } i = 1. \end{cases}$$

Thus, the total latency is at most

$$\begin{aligned} \sum_{k=1}^{R'} \Delta_i^* &\le (11 + 4) \cdot \frac{R'}{2} + 12 \\ &\le 15(R - 1) + 12 \\ &= 15R - 3. \end{aligned}$$

Thus, Theorem 5.1 holds.

Theorem 5.1 implies that the latency of the entire aggregation schedule is at most $15R + \Delta - 4$.

5.2 Pipelined Aggregation Scheduling

Our second algorithm is called **Piplelined Aggregation Scheduling** (**PAS**). Let W be the first set of connectors. We first apply the algorithm **CBFS** on the graph $G[U, W]$ to construct an inward s-arborescence T on $U \cup W$ together with a link labelling and a node ranking. The links in T are then scheduled as follows. Let $R' = Rad(G[U, W], s)$, and $r = rank(s)$. For each $0 \le i \le R'$ and $0 \le j \le r$, set

$$t_{ij} = (R' - i) + 44j.$$

Each link $(v, p(v))$ in T is scheduled in the the time-slot $t_{ij} + 4(l - 1)$, where i is the depth of v in T, j is rank of v, and l is the label of the link $(v, p(v))$.

Theorem 5.3. *The algorithm* ***PAS*** *produces an aggregation schedule of latency at most* $t_{0,r}$.

Proof. We first show that if u has the same depth as v but has a smaller rank than v, then u transmits earlier than v. Suppose that i is the depth of u and v, and j and j' are the ranks of u and v respectively. Then, $i \ge 1$ and $j < j'$. Let l be the label of $(u, p(u))$. We claim that $l \le 11$. This is true if $i > 1$. If $i = 1$, then $l \le |W_0| - 1 \le 11$ by Lemma 4.1. Thus, the claim is true. Hence,

$$t_{i,j} + 3(l - 1) \le t_{i,j} + 40 < t_{i,j+1} \le t_{i,j'},$$

which means u transmits earlier than v.

Now, we show that if u and v transmit in the same time-slot, then the two links $(u, p(u))$ and $(v, p(v))$ are independent. Let i and i' be the depths of u and v respectively. Then either $i = i'$ or $|i - i'| \ge 4$. In the former case, u and v must have the same rank by the previous claim and hence the two links are independent. So we assume that latter case. Since all dominators transmit in even time-slots and all connectors transmit in odd time-slots, either both of them are dominators or both of them are connectors. If they are both dominators, then the do not share a common neighbor in G as their depths differ by more than two, and consequently the two links are independent. If they are both connectors, then $p(u)$ and $p(v)$ do not share a common neighbor as their depths also differ by $|i - i'| \ge 4 > 2$, and hence the two links are independent as well.

Next, we show that if u is a node rather than s and v is a child of u, then u transmits later than v. Let i be the depth of u, and j and j' be the ranks of u and v respectively. Then, $j \ge j'$. If $j = j'$, then the label of the link (v, u) is **one**, and hence v transmits at the time slot $t_{i+1,j} < t_{i,j}$. If $j > j'$, then v transmits no later than the time-slot

$$t_{i+1,j'} + 40 \le t_{i+1,j-1} + 40 < t_{i+1,j} < t_{i,j}.$$

In either case, v transmits earlier than u.

Finally, we show that all nodes transmit before the time-slot $t_{0,r}$. Let v be a node last to transmit. Then $v \in W_0$. Let j be the rank of v,and l be the label of $(v, p(v))$. If $j = r$, then v is the only child of s with the rank r, and hence $l = 1$. So, v transmits in the time-slot $t_{1,r} < t_{0,r}$. If $j < r$, then $l \le |W_0| \le 12$ by Lemma 4.1 and consequently

$$t_{1j} + 4(l - 1) \le t_{1,r-1} + 44 = t_{1,r} < t_{0,r}.$$

Therefore, in either case the transmission by v ends before the time-slot $t_{0,r}$. □

In the next, we show that

$$t_{0,r} = 2R + O(\log R).$$

Since

$$|U \cup W| \le \frac{4\pi}{\sqrt{3}} R^2 + 2\pi R + 1,$$

we have

$$r \le \log |U \cup W| = O(\log R).$$

As $R' \le 2(R - 1)$, we have

$$t_{0,r} = R' + 44r \le 2(R - 1) + 44r = 2R + O(\log R).$$

Theorem 5.3 implies that the latency of the entire aggregation schedule is at most $2R + \Delta + O(\log R)$.

5.3 Enhanced Pipelined Aggregation Scheduling

Our third algorithm is called **Enhanced Pipelined Aggregation Scheduling** (**E-PAS**). Let W be the second set of connectors with an integer parameter $k = \Theta\left(\sqrt[3]{R}/\log R\right)$. We first apply the algorithm **CBFS** on the graph $G[U \cup W]$ to construct an inward s-arborescence T on $U \cup W$ together with a link labelling and a node ranking. The links in T are then scheduled as follows. Let $R' = Rad(G[U \cup W], s)$, $r = rank(s)$, L be the maximum value of the labels of the links in T. For each $0 \le i \le R'$ and $0 \le j \le r$, set

$$t_{ij} = (R' - i) + 3Lj.$$

Each link $(v, p(v))$ in T is scheduled in the the time-slot $t_{ij} + 3(l - 1)$, where i is the depth of v in T, j is rank of v, and l is the label of the link $(v, p(v))$.

THEOREM 5.4. *The algorithm **E-PAS** produces an aggregation schedule of latency at most $t_{0,r}$.*

PROOF. We first show that if u has the same depth as v but has a smaller rank than v, then u transmits earlier than v. Suppose that i is the depth of u and v, and j and j' are the ranks of u and v respectively. Then, $i \geq 1$ and $j < j'$. Hence, u transmits no later than the time-slot

$$t_{i,j} + 3(L-1) < t_{i,j+1} \leq t_{i,j'},$$

which means u earlier than v.

Now, we show that if u and v transmit in the same time-slot, then the two links $(u, p(u))$ and $(v, p(v))$ are independent. Let i and i' be the depths of u and v respectively. Then either $i = i'$ or $|i - i'| \geq 3$. In the former case, u and v must have the same rank by the previous claim and hence the two links are independent. So we assume that latter case, the two links are independent.

Next, we show that if u is a node rather than s and v is a child of u, then u transmits later than v. Let i be the depth of u, an j and j' be the ranks of u and v respectively. Then, $j \geq j'$. If $j = j'$, then the label of the link (v, u) is **one**, and hence v transmits at the time slot $t_{i+1,j} < t_{i,j}$. If $j > j'$, then v transmits no later than the time-slot

$$t_{i+1,j'} + 3(L-1) \leq t_{i+1,j-1} + 3(L-1) < t_{i+1,j} < t_{i,j}.$$

In either case, v transmits earlier than u.

Finally, we show that all nodes transmit before the time-slot $t_{0,r}$. Let v be a node last to transmit. Let j be the rank of v. If $j < r$, then v transmits no later than the time-slot

$$t_{1j} + 3(L-1) < t_{1,j+1} \leq t_{1,r} < t_{0,r}.$$

Now, we assume $j = r$. Then v is the only child of s with the rank r, and hence the label of v is one. So, v transmits in the time-slot $t_{1,r} < t_{0,r}$. Therefore, in either case the transmission by v ends before the time-slot $t_{0,r}$. $\square$

In the next, we show that

$$t_{0,r} = \left(1 + \Theta\left(\frac{\log R}{\sqrt[3]{R}}\right)\right) R.$$

By Lemma 4.2,

$$R' \leq (1 + 1/k) R = \left(1 + \Theta\left(\frac{\log R}{\sqrt[3]{R}}\right)\right) R,$$

and

$$\begin{aligned}
&|U \cup W| \\
&\leq (k+1)\left(\frac{2\pi}{\sqrt{3}} R^2 + \pi R\right) + 1 \\
&= \Theta\left(\frac{\sqrt[3]{R}}{\log R}\right)\left(\frac{2\pi}{\sqrt{3}} R^2 + \pi R\right) + 1 \\
&= \Theta\left(\frac{R^{7/3}}{\log R}\right).
\end{aligned}$$

Thus

$$r \leq \log |U \cup W| = O(\log R).$$

By Lemma 3.1 and Lemma 4.2,

$$\begin{aligned}
L &\leq \Delta(G[U \cup W]) \\
&\leq 2\sqrt{3}\pi k^2 + 3\pi k + 3 + 4\pi/\sqrt{3} \\
&= \Theta\left(\frac{\sqrt[3]{R^2}}{\log^2 R}\right).
\end{aligned}$$

So, we have

$$\begin{aligned}
t_{0,r} &= R' + 3Lr \\
&\leq \left(1 + \Theta\left(\frac{\log R}{\sqrt[3]{R}}\right)\right) R + 3 \cdot \Theta\left(\frac{\sqrt[3]{R^2}}{\log^2 R}\right) \cdot O(\log R) \\
&= \left(1 + \Theta\left(\frac{\log R}{\sqrt[3]{R}}\right)\right) R + O\left(\frac{\sqrt[3]{R^2}}{\log R}\right) \\
&= \left(1 + \Theta\left(\frac{\log R}{\sqrt[3]{R}}\right)\right) R.
\end{aligned}$$

Theorem 5.4 implies that the latency of the entire aggregation schedule is at most $\left(1 + O\left(\frac{\log R}{\sqrt[3]{R}}\right)\right) R + \Delta$.

6. AGGREGATION SCHEDULING WITH $\rho > 1$

In this section, we introduce a generic approach to extending a "well-separated" communication schedule with $\rho = 1$ to an aggregation schedule with $\rho > 1$ whose latency is increased by a $\Theta(\rho^2)$ factor. A communication schedule with $\rho = 1$ is said to be *well-separated* if at each time-slot of the schedule, either all transmitting nodes have mutual distances greater than one or all receiving nodes have mutual distances greater than one. Clearly, the first two algorithms in the previous section both produce a well-separated aggregation schedules produced by **SAS** and **PAS** are well-separated, while the aggregation schedule produced by **E-PAS** is not.

Fix $\rho > 1$. Suppose that A is a set of links in $\overrightarrow{G}$. A link schedule of A is a partition of A into subsets of links which are mutually conflict-free, and its latency is the number of subsets in the partition. The *conflict graph* of A is an undirected graph on A in which there is an edge between links in A if and only if these two links are not conflict-free. Then a link schedule for A is equivalent to a proper vertex coloring of its conflict graph of A, with the latency corresponding to the number of colors. Let $\delta^*(A)$ denote the inductivity of the conflict graph of A. Then, the first-fit coloring in the smallest-degree-last ordering of the conflict graph of A uses at most $1 + \delta^*(A)$ colors. Let

$$\beta_r = \frac{\pi}{\sqrt{3}} r^2 + \left(\frac{\pi}{2} + 1\right) r + 1.$$

The next lemma gives an upper bound on $\delta^*(A)$.

LEMMA 6.1. *Suppose that A is a set of links in $\overrightarrow{G}$ whose tails (respectively, heads) have mutual distances greater than one. Then, $\delta^*(A) \leq \beta_{\rho+1} - 1$.*

PROOF. By symmetry, we assume that the tails of the links in A have mutual distances greater than one. Consider and arbitrary subset A' of A. Let a be the link whose tail, denoted by u, is the rightmost one among all the tails of the links in A'. Then, all the tails of the links in A' which have conflict with a must lie in a half-disk of radius $\rho + 1$

centered at u. By Corollary 2.3, the number of these tails is at most $\beta_{\rho+1} - 1$, where the -1 term is due to that the tail of a is also in the half-disk. Hence, the minimum degree of the conflict graph of A' is at most $\beta_{\rho+1} - 1$. Thus, the lemma holds. □

Lemma 6.1 implies that if A is a set of links in $\overrightarrow{G}$ whose tails (respectively, heads) have mutual distances greater than one, the first-fit coloring in the smallest-degree-last ordering of the conflict graph of A gives a link schedule of latency at most $\beta_{\rho+1}$.

Now, consider a well-separated communication schedule with $\rho = 1$ given by

$$\mathcal{A} = \{A_k : 1 \leq k \leq \ell\}$$

where A_k is the set of links in the k-th time-slot for each $1 \leq k \leq \ell$. We construct a communication schedule $\mathcal{A}_\rho$ for $\rho > 1$ as follows. The schedule is partitioned into ℓ rounds. For each $1 \leq k \leq \ell$, the k-th round is a link schedule of A_k corresponding to the first-fit coloring of the conflict graph of A_k in the smallest-degree-last ordering. The schedule $\mathcal{A}_\rho$ is referred to as the ρ-*expansion* of $\mathcal{A}$. Since each round in $\mathcal{A}_\rho$ takes at most $\beta_{\rho+1}$ time-slots, the latency of $\mathcal{A}_\rho$ is at most $\beta_{\rho+1}\ell$. In summary, we have the following general theorem.

THEOREM 6.2. *For any $\rho > 1$, the ρ-expansion of a communication schedule with $\rho = 1$ and latency equal to ℓ has latency at most $\beta_{\rho+1}\ell$.*

Theorem 6.2 immediately implies the following corollary.

COROLLARY 6.3. *For any $\rho > 1$, the ρ-expansion of the aggregation schedule produced by **SAS** (respectively, **PAS**) has latency at most $\beta_{\rho+1}(15R + \Delta - 4)$ (respectively, $\beta_{\rho+1}(2R + \Delta + O(\log R))$).*

An interesting observation is that for $\rho > 1$, $\Theta(\Delta)$ is a lower bound on the minimum aggregation latency. Let Δ_ρ denote the maximum number of nodes within the interference range of a node. In other words, Δ_ρ is the maximum degree of the ρ-disk graph of the networking nodes. Clearly, $\Delta_\rho \geq \Delta_1 = \Delta$. For each $d > 0$,denote

$$\alpha_\rho = \frac{2\pi}{\sqrt{3}}\left(\frac{\rho}{\rho-1}\right)^2 + \pi\left(\frac{\rho}{\rho-1}\right) + 1.$$

We have the following bound on the minimum aggregation latency.

LEMMA 6.4. *For any $\rho > 1$, the minimum aggregation latency is at least Δ_ρ/α_ρ.*

PROOF. Let u be a node with maximum degree in the ρ-disk graph. Let C be the ρ-disk centered at u. Then, C contains $\Delta_\rho + 1$ nodes. If s is not in C, then all these $\Delta_\rho + 1$ nodes in C have to transmit; otherwise, exactly Δ_ρ nodes in C have to transmit. In either case, at least Δ_ρ nodes in C have to transmit. Since all nodes transmitting in the same time-slot must be apart from each other by a distance greater than $\rho - 1$, at most α_ρ nodes in C can transmit in a time-slot by Corollary 2.3. Hence, the Δ_ρ transmissions by the nodes in C takes at least Δ_ρ/α_ρ time-slots. □

By Lemma 6.4, the minimum aggregation latency with $\rho > 1$ is at least Δ_ρ/α_ρ. Since R is also a lower on the minimum latency regardless of ρ, the ρ-expansions of the aggregation schedules produced by **SAS** and **PAS** respectively are both constant approximations.

7. CONCLUSION

In this paper, we have developed three aggregation scheduling algorithms with $\rho = 1$, **SAS**, **PAS**, and **E-PAS**. All of them produce shorter aggregations schedules than those proposed in the past. Among those three, **SAS** has the simplest implementation and it may outperform the other two when the radius R is small. For large radius R, both **PAS** and **E-PAS** speed up the aggregation schedule by using the pipelining technique. Novel structures like the two connected dominating sets and the canonical inward arborescences used by these three algorithms are of independent interest and are expected to have applications in other communication scheduling. We also introduced a generic expansion technique which adapts a well-separated communication schedule with $\rho = 1$ to a communication schedule with with $\rho > 1$. With such expansion technique, we obtained two aggregation schedules with $\rho > 1$ whose latencies are within constant factors of the minimum latency. We also expect that this expansion technique can be applied for the scheduling of other communications.

Acknowledgement We would like to thank Dr. Anil Vullikanti for many helpful suggestions on the revision of this paper.

8. REFERENCES

[1] X.J. Chen, X.D. Hu, J.M. Zhu, Minimum data aggregation time problem in wireless sensor networks, *Lecture Notes in Computer Science* 3794, pp. 133-142, 2005.

[2] H. Groemer, Über die Einlagerung von Kreisen in einen konvexen Bereich. *Math. Z.*, 73:285–294, 1960.

[3] P. Gupta and P.Kumar, The capacity of wireless networks, *IEEE Trans. Inform. Theory*, vol. 46, pp. 388–404, Mar. 2000.

[4] S.C.-H. Huang, P.-J. Wan, C. T. Vu, Y. Li, and F. Yao: Nearly Constant Approximation for Data Aggregation Scheduling in Wireless Sensor Networks, *IEEE INFOCOM* 2007.

[5] D. W. Matula and L. L. Beck, Smallest-last ordering and clustering and graph coloring algorithms, *Journal of the Association of Computing Machinery*, 30(3):417-427, 1983.

[6] P.-J. Wan, K.M. Alzoubi, and O.Frieder: Distributed Construction of Connected Dominating Set in Wireless Ad Hoc Networks, *ACM Mobile Networks and Applications*, Vol. 9, No. 2, pp.141-149, 2004. A preliminary version of this paper appeared in *IEEE INFOCOM*, 2002.

[7] P.-J. Wan, C.-W. Yi, X. Jia, and D. Kim, Approximation Algorithms for Conflict-Free Channel Assignment in Wireless Ad Hoc Networks, *Wiley Journal on Wireless Communications and Mobile Computing* 6(2):201-211, 2006.