Faster Communication in Known Topology Radio Networks

Λέων-Χαράλαμπος Σταματάρης

INTRODUCTION

- Two classical problems of information dissemination in computer networks:
  - The broadcasting problem:

    Distributing a particular message from a distinguished source node to all other nodes in the network
  - The gossiping problem:

    Each node *u* in the network initially holds a message m_u, and it is required to distribute all messages m_u to all nodes in the network

INTRODUCTION

- The model of the network:
  - An undirected connected graph $G = (V, E)$, $(u, w) \in E$ iff the transmissions of node u can directly reach node w and vice versa.
  - The number of neighbors of a node w is called its degree. the maximum degree of any node in the network is called the max-degree of the network and is denoted by Δ.
  - each node u either transmits or listens.
  - a node w adjacent to u successfully receives this message iff in this step w is listening and u is the only transmitting node among w's neighbors.

INTRODUCTION

- Part 1:
  - Deterministic schedule $O(D + \Delta \log n)$ for the gossiping task
    Best known gossiping schedule $O\left(D + \sqrt[i+2]{D}\Delta log^{i+1} n\right)$

  - Deterministic schedule $D + O(\log^3 n)$ for the broadcast task
    Randomized schedule $D + O(\log^2 n)$ for the broadcast task
    Best known broadcasting schedule $D + O(\log^4 n)$

INTRODUCTION

- Part 2:
  - Broadcasting schedule for planar graphs ≥ 3D
    Best known broadcasting schedule $D + O(\log^3 n)$

GOSSIPING IN GENERAL GRAPHS WITH KNOWN TOPOLOGY

- The gossiping task can be performed in two consecutive stages.
  - First Stage: gather all individual messages in one (central) point of the graph.
  - Second Stage: the collection of individual messages is broadcast to all nodes in the network.

GOSSIPING IN GENERAL GRAPHS WITH KNOWN TOPOLOGY

- We start this section with the presentation of a simple gathering procedure that works in time $O((D + \Delta) \log n)$ in free trees.
- Later we show how to choose a spanning breadth-first (BFS) tree in an arbitrary graph G in order to gather (along its branches) all messages in G also in time $O((D + \Delta) \log n)$
- Finally, we show how the gathering process can be pipelined and sped up to time $O(D + \Delta \log n)$

GOSSIPING IN GENERAL GRAPHS WITH KNOWN TOPOLOGY

- **Ranking procedure:**
  - Given an arbitrary tree, we choose as the root its central node *c*. The nodes in a tree rooted in *c* are partitioned into consecutive layers $L_i = \{v \mid dist(c, v) = i\}$ for $i = 0, \ldots., r$ where *r* is a radius of the tree. We denote the size of each layer L_i by $|L_i|$
  - Each leaf v has rank(v) = 1
  - A non-leaf node determines its rank as rank(v) = r_{max} where r_{max} is the maximum rank of its children.
  - If there are at least two children with the rank r_{max} then the rank of node v is set to rank(v) = $r_{max} + 1$

GOSSIPING IN GENERAL GRAPHS WITH KNOWN TOPOLOGY

- **Ranking procedure:**
  - Lemma 2.1. *The largest rank in a tree of size n is bounded by* $\lceil \log n \rceil$

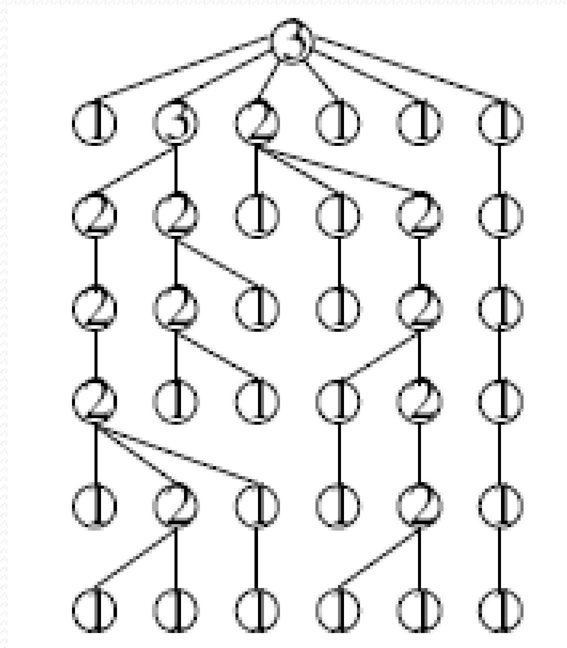


Figure 1: A ranked tree of size n = 37

GOSSIPING IN GENERAL GRAPHS WITH KNOWN TOPOLOGY

- The schedule is now defined in stages using the ranked tree by partitioning the nodes into different rank sets $R_i = \{v \mid rank(v) = i\}$ for $1 \leq i \leq r_{max} \leq \lceil \log n \rceil$

- The meaning of this partition is that the nodes in R_i are involved in transmissions only during the stage i.

GOSSIPING IN GENERAL GRAPHS WITH KNOWN TOPOLOGY

- We now define two key subsets of nodes for each stage.

The *fast transmission set*:

$$F_i^k = \{v \mid v \in L_k \cap R_i \text{ and } parent(v) \in R_i\}.$$

Also define $F_i = \bigcup_{k=1}^{D} F_i^k$ and $F = \bigcup_{i=1}^{r_{max} \leq \lceil \log n \rceil} F_i$.

The *slow transmission set*:

$$S_i^k = \{v \mid v \in L_k \cap R_i \text{ and } parent(v) \in R_j, j > i\}.$$

Also define $S_i = \bigcup_{k=1}^{D} S_i^k$ and $S = \bigcup_{i=1}^{r_{max} \leq \lceil \log n \rceil} S_i$.

- Lemma 2.2. *During the ith stage, all nodes in F_i^k can transmit to their parents simultaneously without any collisions, for i = 1,...., $r_{max} \leq \lceil \log n \rceil$ and k = 1,....,D.*

GOSSIPING IN GENERAL GRAPHS WITH KNOWN TOPOLOGY

- The following procedure moves messages from all nodes with rank *i* into their parents with ranks $i + 1$ or higher.
  - Procedure Gathering(*i*);

1. Move messages from nodes in F_i to S_i from F_i^D down to F_i^1 layer by layer.
2. Move messages from nodes in S_i to their parents, all parents collect their messages from their children in S_i one by one.

- The time complexity of step 1 is *O(D)* due to Lemma 2.2. The time complexity of step 2 is bounded by *O(Δ)*
- Theorem 2.3. *In any tree of size n, diameter D and maximum degree Δ, the gossiping task can be completed in time* $O((D + \Delta) \log n)$.

GOSSIPING IN GENERAL GRAPHS WITH KNOWN TOPOLOGY

- **Gathering messages in arbitrary graphs:**
  - Gathering spanning tree (GST)

In an arbitrary graph $G = (V, E)$, any BFS spanning tree T_G of G s.t.

(1) T_G is rooted at the central node c of G,

(2) T_G is ranked, and

(3) all nodes in F_i^k of T_G are able to transmit their messages to their parents simultaneously without any collision, for all $1 \le k \le D$ and $1 \le i \le r_{max} \le \lceil \log n \rceil$

is called a gathering spanning tree, or simply GST

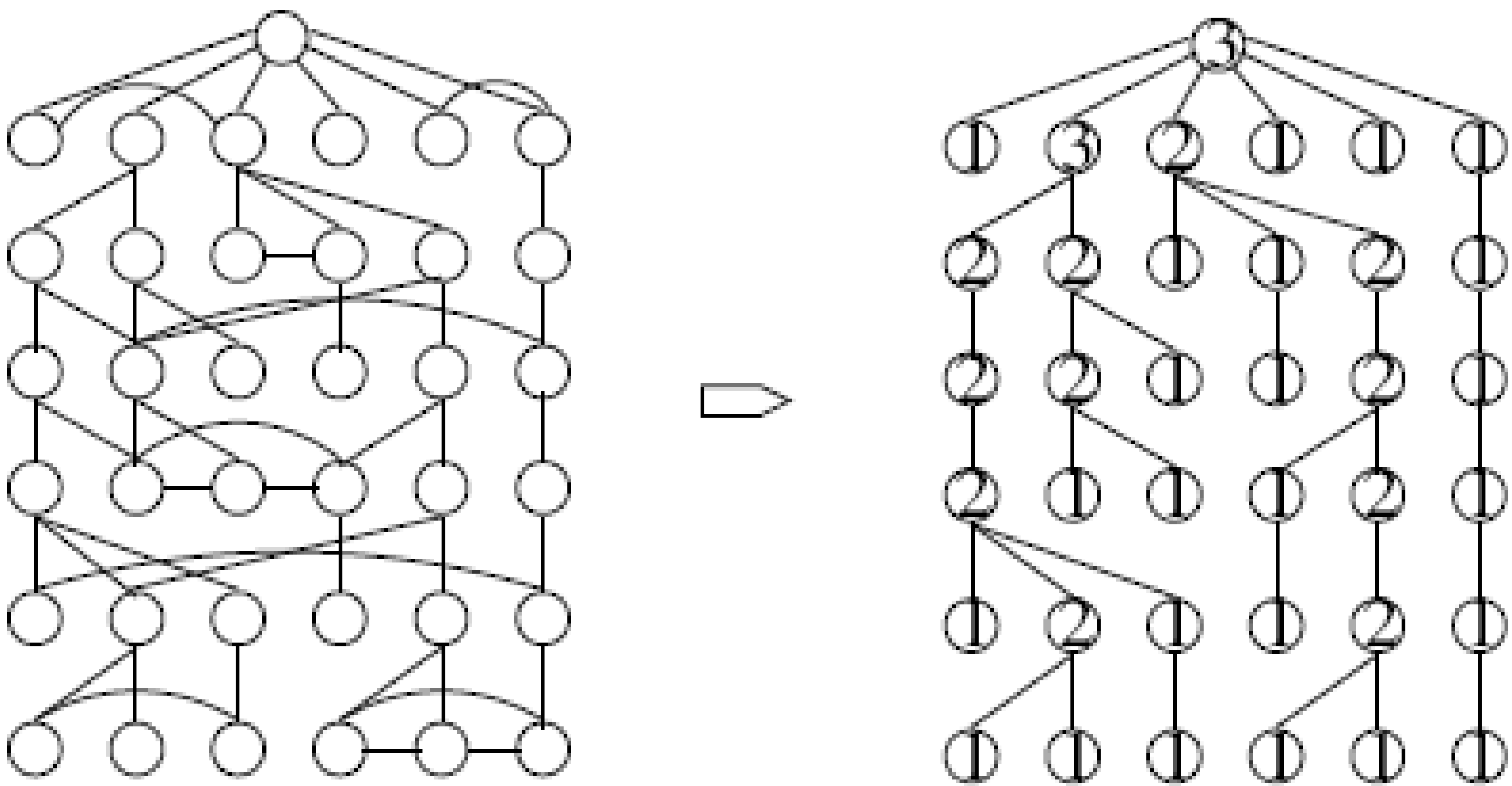
GOSSIPING IN GENERAL GRAPHS WITH KNOWN TOPOLOGY

- **The construction of GST.**
  - In a graph $G = (V, E)$, an arbitrary ranked BFS spanning tree rooted in the central node c is called a pre-gathering-tree $T_{PGT} = (V, E_{PGT})$
  - Function Check-collision(i, j): a pair of nodes;

    (1) if $\exists u, v \in F_j^i$ and $(u, parent(v)) \in E$, where $u \neq v$ then return(u, v);

    else return('null');

GOSSIPING IN GENERAL GRAPHS WITH KNOWN TOPOLOGY



Original Graph

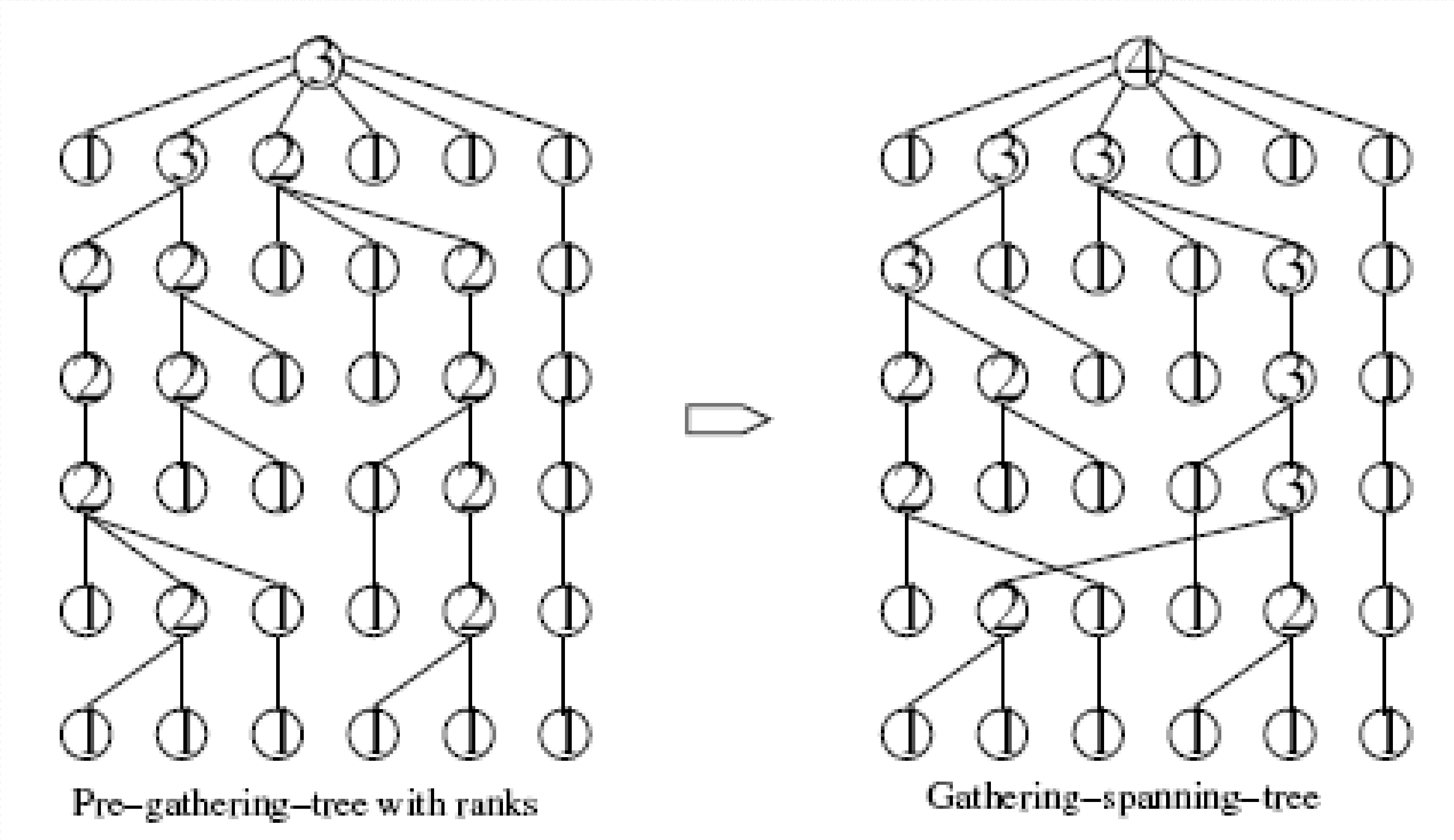
Pre–gathering–tree with ranks

GOSSIPING IN GENERAL GRAPHS WITH KNOWN TOPOLOGY

- **Procedure Gathering-Spanning-Tree:**

For $i := D$ down to 1 do
begin
  For $j := r_{max}$ down to 1 do
  begin
    While CHECK-COLLISION$(i, j) \neq$ '*null*' do
    begin
      $\mathbf{rank}(parent(v)) = j + 1;$
      $F_j^i = F_j^i - \{v, u\};$
      $S_j^i = S_j^i \cup \{v, u\};$
      $E_{PGT} = E_{PGT} - \{(u, parent(u))\};$
      $E_{PGT} = E_{PGT} \cup \{(u, parent(v))\};$
      re-rank T_{PGT} only at the top BFS layers from $i - 1$ down to 0;
      recompute sets in F and S in new T_{PGT};
    end
  end
end

GOSSIPING IN GENERAL GRAPHS WITH KNOWN TOPOLOGY



GOSSIPING IN GENERAL GRAPHS WITH KNOWN TOPOLOGY

- Lemma 2.4. *After completing the pruning process at layer i in T_{PGT}, the structure of edges in T_{PGT} between layers i -1,...,D is fixed, i.e., the transmissions within layers i,...,D in all sets F_j, for j = 1,...,$r_{max} \leq \lceil \log n \rceil$ are free of collisions.*

- Theorem 2.5. *There exists an efficient polynomial time construction of a GST on an arbitrary graph G.*

GOSSIPING IN GENERAL GRAPHS WITH KNOWN TOPOLOGY

- Using the ranks of the GST nodes (constructed in the previous section), all nodes get partitioned into distinct rank sets $R_i = F_i \bigcup S_i$, where $1 \leq i \leq r_{max} \leq \lceil \log n \rceil$ Initially, all messages are gathered into the central node *c*, stage by stage, using the structure of the GST.

GOSSIPING IN GENERAL GRAPHS WITH KNOWN TOPOLOGY

- During the ith stage, all messages from nodes in F_i are first moved to the nodes in S_i.
- We divide the sequence of transmission time slots into three separate (interleaved) subsequences of time slots. Specifically, the nodes in S_i transmit in time slots: $t \equiv 0 \pmod 3$ iff $i \equiv 0 \pmod 3$; $t \equiv 1 \pmod 3$ iff $i \equiv 1 \pmod 3$; and $t \equiv 2 \pmod 3$ iff $i \equiv 2 \pmod 3$. Later, we move all messages from nodes in Si to their parents in GST.

GOSSIPING IN GENERAL GRAPHS WITH KNOWN TOPOLOGY

- Lemma 2.6. *In stage i, nodes in set S_i of the GST transmit their messages to the parents in time $O(\Delta)$.*

- Theorem 2.7. *In any graph G, the gossiping task can be completed in time $O((D + \Delta) \log n)$.*

GOSSIPING IN GENERAL GRAPHS WITH KNOWN TOPOLOGY

- In this section we show how to pipeline the transmissions of different stages.
- The pattern of transmissions of a node v at layer i and with rank j in GST depends on whether it belongs to the set F or to the set S, and it is as follows:

  (1) if $v \in F$, then v transmits within the time block
  $(D - i) + j\Delta$

  (2) otherwise ($v \in S$), v transmits within the time block
  $(D - i) + j\Delta + s(v)$, $1 \leq s(v) \leq \Delta$

GOSSIPING IN GENERAL GRAPHS WITH KNOWN TOPOLOGY

- Lemma 2.8. *A node v transmits its message as well as all messages collected from its descendants towards its parent in GST successfully during the time block allocated to it by the pattern of transmissions.*

- Theorem 2.9. *In any graph G, the gossiping task can be completed in time* $O(D + \Delta \log n)$.

- Corollary 2.10. *The gossiping can be completed in time* $O(D)$ *in all graphs with* $\Delta = O(D / \log n)$.

BROADCASTING IN GRAPHS WITH KNOWN TOPOLOGY

- The deterministic algorithm B uses the concept of the ranked gathering spanning tree
- Let us start with an overview of the broadcast process from the point of view of a copy of the message that was eventually received at some leaf a of the tree.
- Let us denote that the message follow the shortest path $p(a)$

$$p(a) = \langle p_1^F(a), p_1^S(a), p_2^F(a), p_2^S(a), \ldots, p_q^F(a), p_q^S(a) \rangle$$

BROADCASTING IN GRAPHS WITH KNOWN TOPOLOGY

- During the broadcasting process the nodes in the tree use the following pattern of transmissions:
  - Consider a node υ of rank $1 \leq i \leq r_{max}$ on BFS layer L_i with a child w of the same rank at the next BFS layer. Then υ is set to perform a fast transmission to w in time steps t satisfying $t \equiv i + 6j \pmod{6\, r_{max}}$
  - The slow transmissions at the BFS layer L_i are performed in time steps t satisfying $t \equiv i + 3 \pmod{6}$
  - For slow transmissions, algorithm B uses the $O(\log^2 n)$ transmission Procedure CW

BROADCASTING IN GRAPHS WITH KNOWN TOPOLOGY

- The total time required for the broadcast message to reach a leaf *a* in the tree can be bounded as follows. Let D_i, for $1 \leq i \leq r_{max}$, denote the length of $p^F(a)$, the ith fast segment of the route $p(a)$ used by the broadcast message that has reached *a*. Thus the time required to communicate a is bounded by $O(\log n)+D_1+: : :+O(\log n)+D\, r_{max}\ \ D+ O(\log^2 n)$ for the fast transmissions plus $r_{max}\ \ O(\log^2 n) = O(\log^3 n)$ for the slow transmissions, yielding a total of $D+ O(\log^3 n)$.

BROADCASTING IN GRAPHS WITH KNOWN TOPOLOGY

- Theorem 3.1. *There exists a deterministic polynomial time algorithm that constructs, for any n node radio network of diameter D, a broadcasting schedule of length* $D + O(\log^3 n)$.

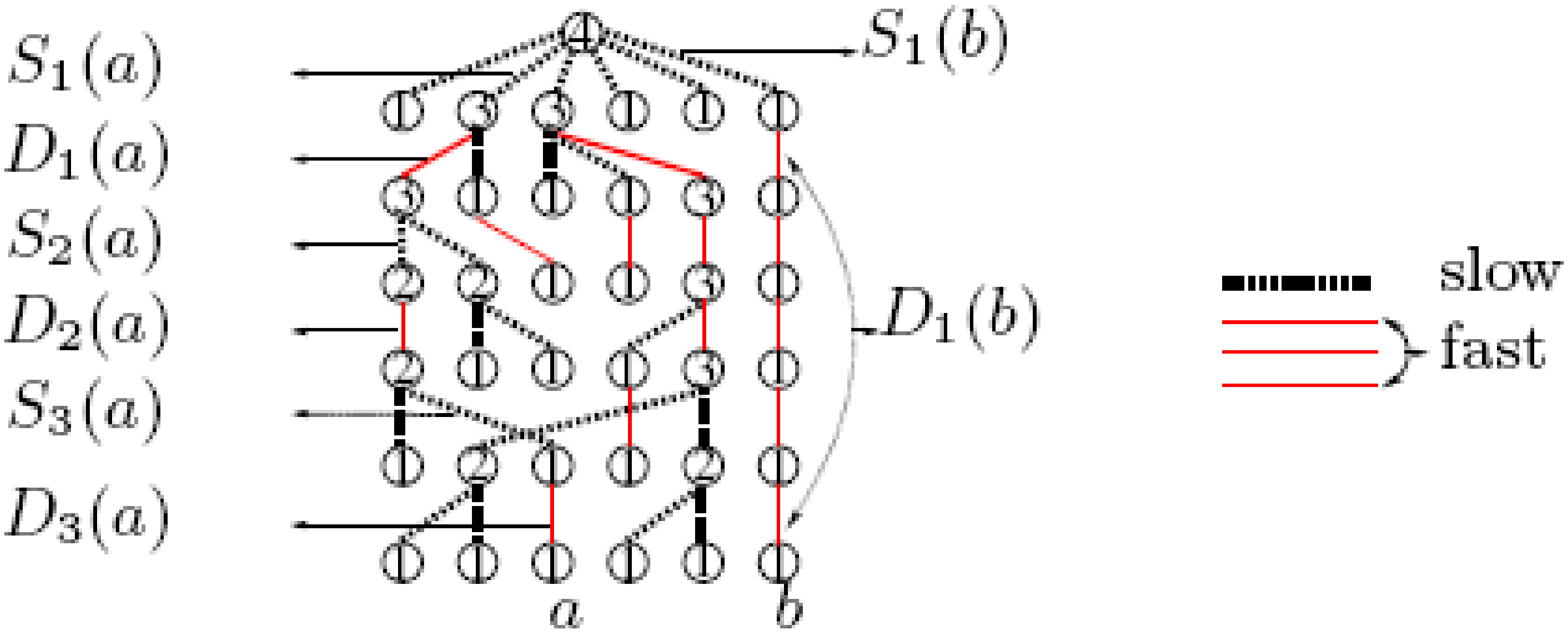
- Claim 3.2. *Consider an uninformed node w in* L_j+1. *Suppose that at the beginning of the current activation of procedure RCW, w has some informed neighbors on layer* L_j . *Then w will get the message during the current activation of procedure RCW with constant probability* $p \geq 1/(4e)$.

BROADCASTING IN GRAPHS WITH KNOWN TOPOLOGY

- Theorem 3.3. *There exists a randomized algorithm that for any known topology of n node radio network of diameter D and any source node s; following a polynomial preprocessing stage, broadcasts a message from s with high probability in time* $D + O(\log^2 n)$.

- Corollary 3.4. *For any known topology n nodes radio network of diameter D; there exists a broadcasting schedule of length* $D + O(\log^2 n)$:

BROADCASTING IN GRAPHS WITH KNOWN TOPOLOGY



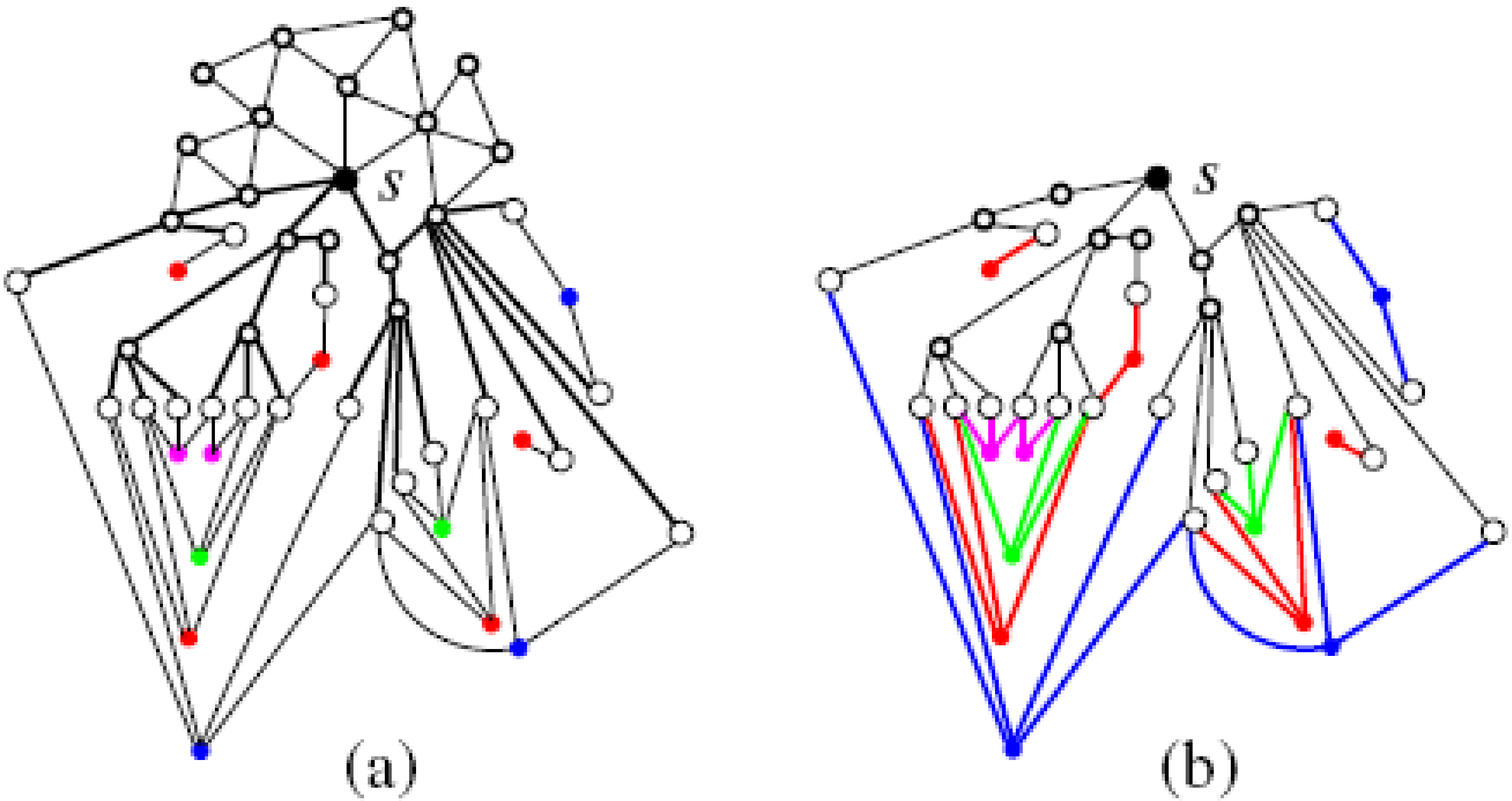
ASYMPTOTICALLY OPTIMAL RADIO BROADCAST ON PLANAR GRAPHS

- In this section we sketch an algorithm for constructing a transmission schedule for performing broadcast from a given source s on a known planar radio network G in asymptotically optimal $O(D)$ time. The schedule consists of D phases, each of up to 3 rounds.
- let L_p denote the set of vertices at distance p from s.
- Phase 1: Only s transmits, and by the end of this round, all the vertices of L_1 are informed.
- Assuming all the vertices of layer $U = L_{p-1}$ are informed, let us now describe the algorithm for constructing the sub-schedule of phase p, designed to inform all the vertices of $D = L_p$

ASYMPTOTICALLY OPTIMAL RADIO BROADCAST ON PLANAR GRAPHS

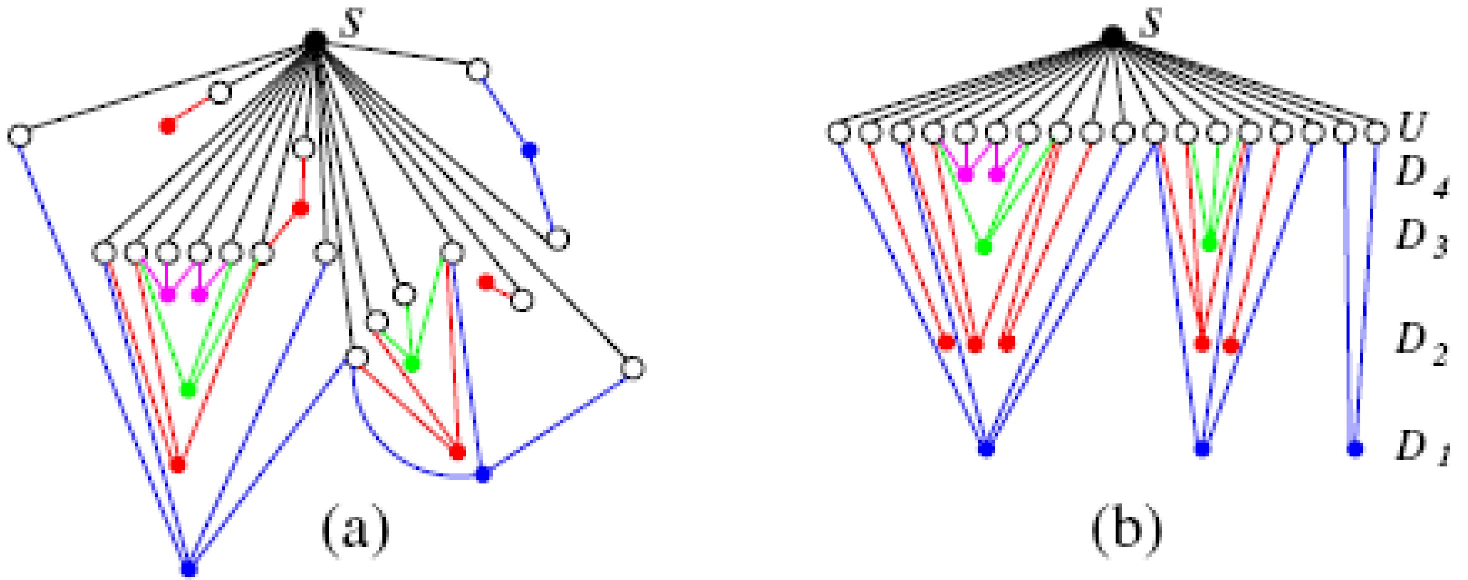
- Preprocessing stage: Constructs a bipartite graph consisting only of the nodes of the two layers *U* and *D* and the edges connecting them.
  - Construct a planar embedding of *G* with *s* at the top (on the outer face) and all other vertices below it.
  - Erase from the graph all the vertices of layers L_j for $j > p$ and their edges, as well as all the edges connecting vertices of *D*
  - mark on the graph a shortest paths tree *T* rooted at s and leading to all the vertices of *U*
  - Next, erase from the graph all the vertices of layers other than *D* and *U* that do not participate in this tree

ASYMPTOTICALLY OPTIMAL RADIO BROADCAST ON PLANAR GRAPHS



ASYMPTOTICALLY OPTIMAL RADIO BROADCAST ON PLANAR GRAPHS

- Next, we replace the tree *T* by a star connecting *s* directly to the vertices of layer *U*.
- Next, we modify the embedding so that the vertices of layer *U* occur on a straight horizontal line and the vertices of layer *D* occur below this line.



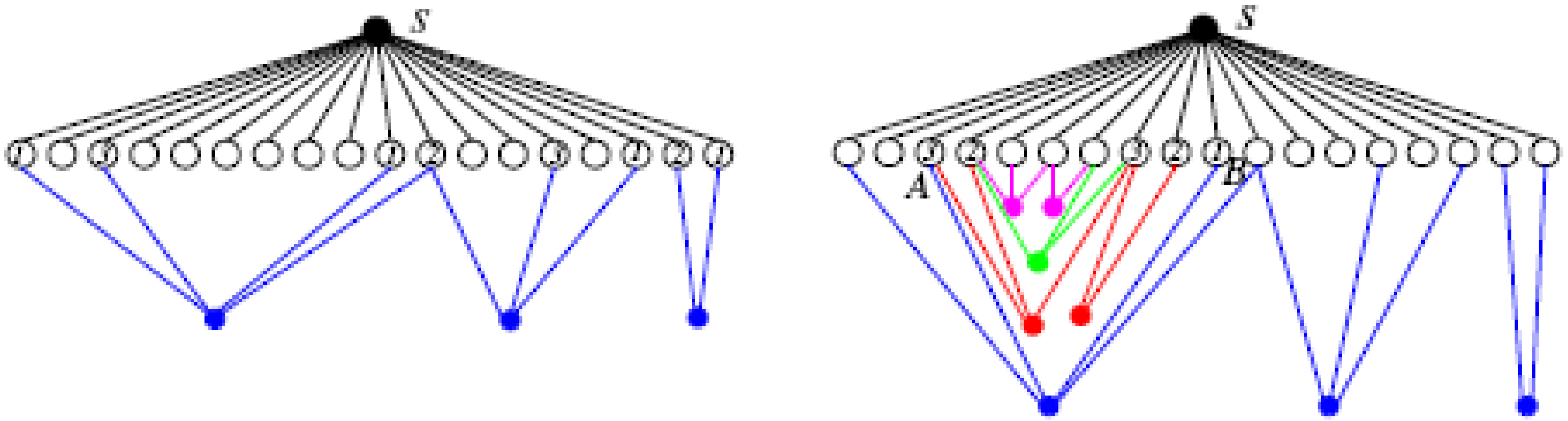
ASYMPTOTICALLY OPTIMAL RADIO BROADCAST ON PLANAR GRAPHS

- We now assign depth values to the vertices of D. This is done recursively as follows. Let $d = 1$. Assign each vertex $v \in D$ on the outer face a depth value depth(v) = d. Now erase all the vertices of D on the outer face and their edges, and increase d by 1. If D is still nonempty then recurse.
- For each vertex $v \in D$, denote its leftmost U neighbor by **left**(v), its rightmost U neighbor by **right**(v), and the list of its remaining neighbours (if any) by **rest**(v), taken from left to right.
- Finally, the schedule is defined as follows. The three time slots of the current phase p are $t_1 = 3p - 4$, $t_2 = 3p - 3$ and $t_3 = 3p - 2$.

ASYMPTOTICALLY OPTIMAL RADIO BROADCAST ON PLANAR GRAPHS

- Stage *l* starts with the depth 1 vertices, kept in the ordered list $D_1 = <\upsilon_1,\ldots, \upsilon_\kappa>$. Construct the ordered list of "breakpoint vertices"
- $B = <\text{left}(\upsilon_1), \text{right}(\upsilon_1), \ldots., \text{left}(\upsilon_\kappa), \text{right}(\upsilon_\kappa)>$
- Assign time slots t_1 and t_2 alternately to the vertices of the list *B*.
- Next, for each υ_i with nonempty list rest(υ_i), assign time slots to the vertices of that list as follows.
- If left(υ_i) was assigned the time slot t_1 (hence right(υ_i) was assigned the time slot t_2), then assign the time slots t_3 and t_1 alternately to the nodes of rest(υ_i) from left to right.

ASYMPTOTICALLY OPTIMAL RADIO BROADCAST ON PLANAR GRAPHS



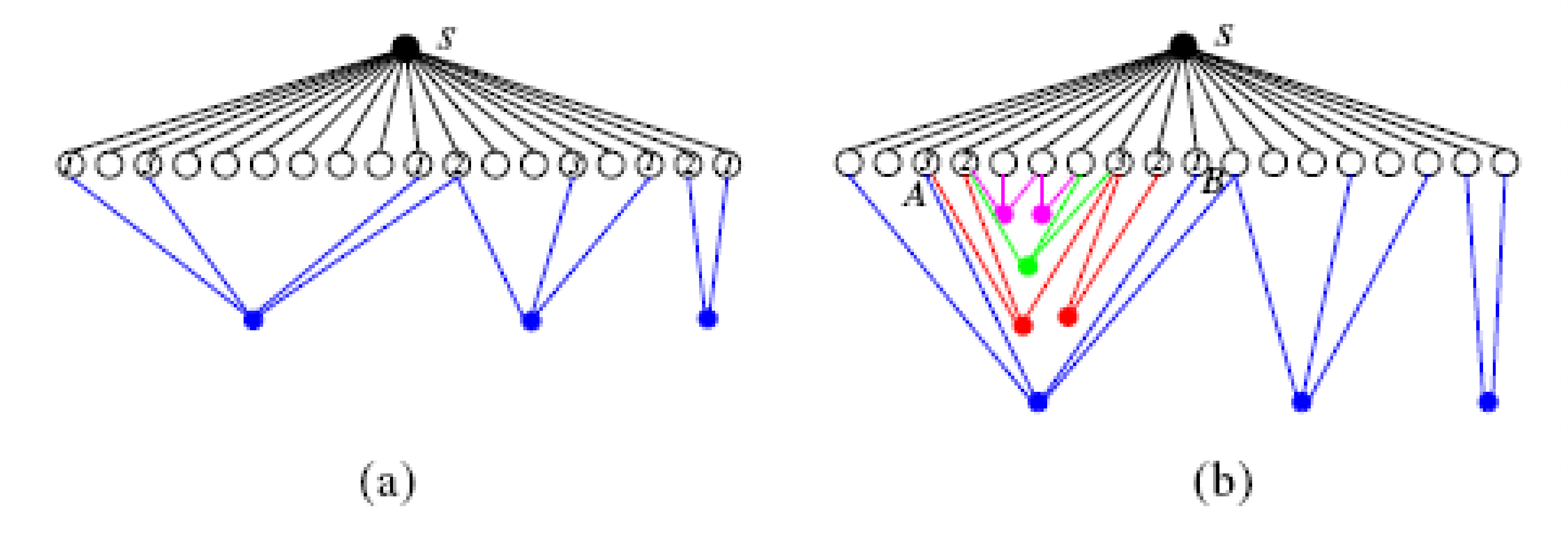
(a)

(b)

ASYMPTOTICALLY OPTIMAL RADIO BROADCAST ON PLANAR GRAPHS

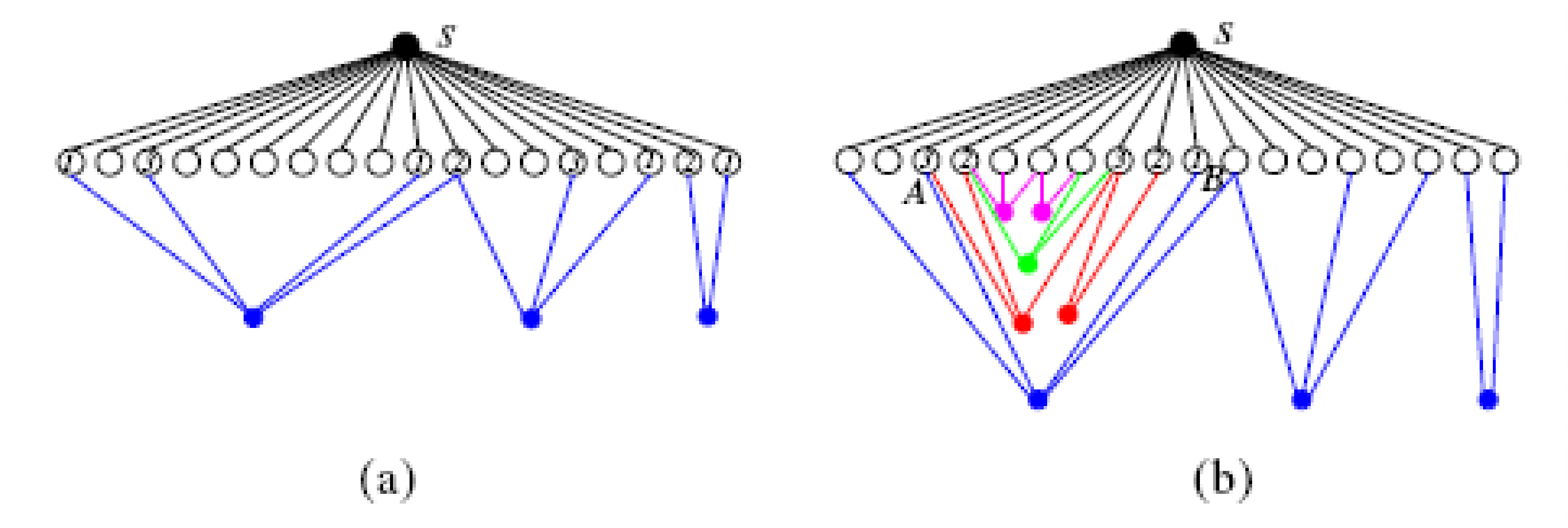
- Let us now describe stage $l \geq 2$, showing how to assign time slots to some U neighbors of vertices in D_l so as to ensure that they get the message during the current phase. The inductive hypotheses we rely on at the beginning of stage l are the following:
  - (P1) At the end of stage l -1, all the neighbors of vertices of D_k for $k < l$ were already assigned time slots,
  - (P2) these previously made assignments ensure that all the vertices of D_k for $k < l$ receive the message during the phase, and
  - (P3) at the end of stage l -1, every two consecutive vertices in $U_{assigned}$ are assigned different time slots

ASYMPTOTICALLY OPTIMAL RADIO BROADCAST ON PLANAR GRAPHS



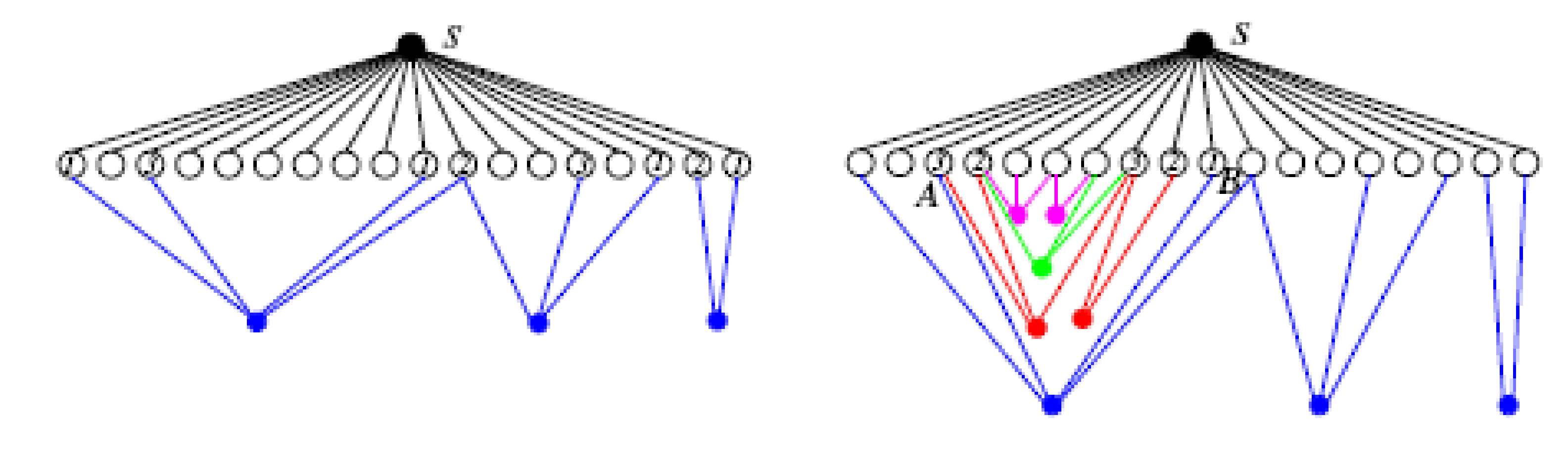
Let A be the rightmost U vertex to the left of left(v) that has already been assigned a time slot t_A previously. (A can possibly be left(v) itself.) Similarly, let B be the leftmost U vertex to the right of right(v) (possibly right(v) itself) that has already been assigned a time slot t_B previously.

ASYMPTOTICALLY OPTIMAL RADIO BROADCAST ON PLANAR GRAPHS



Note that left(v) and right(v) may have already been assigned a time slot previously, but the vertices of rest(v) (if any exist) are necessarily still unassigned at the beginning of stage l. Moreover, if both left(v) and right(v) have been assigned a time slot previously then these time slots must be different [(P3)], as A and B occur consecutively in $U_{assigned}$.

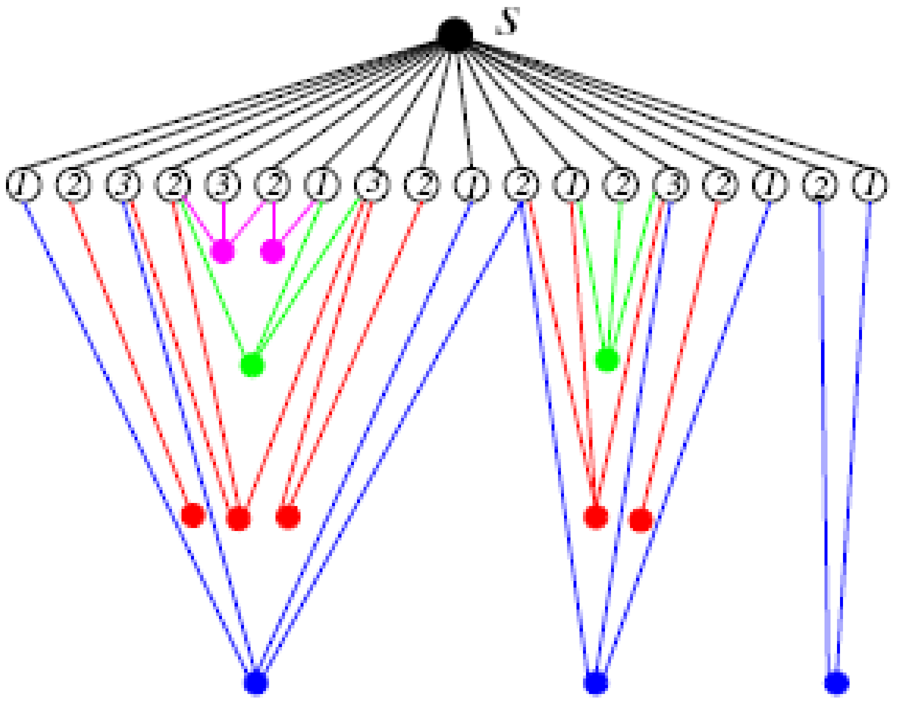
ASYMPTOTICALLY OPTIMAL RADIO BROADCAST ON PLANAR GRAPHS



(a) (b)

Now assign time slots to all unassigned U neighbors of υ as follows. Let $t_C \in \{t_1, t_2, t_3\}$ be a time slot different from t_A and t_B. Go over the unassigned U neighbors of υ from left to right, and assign them the time slots t_C and t_A alternately.

ASYMPTOTICALLY OPTIMAL RADIO BROADCAST ON PLANAR GRAPHS



: Final assignment of transmission times in D.

CONCLUSION

- We proposed here new efficient (polynomial time) construction of the deterministic schedule that performs the gossiping task in time $O(D + \Delta \log n)$.
- The new gossiping schedule is asymptotically optimal if $\Delta = O(D/\log n)$