Broadcasting with Nodes of Limited Memory

By: Saber Gholami

Supervisor: Professor Hovhannes Harutyunyan

Concordia University,

Department of Computer Science and Software Engineering

April 1^{st}, 2022

Outline

Outline

1. Introduction
2. Preliminaries and Literature Review
3. Contribution
   - Fully-adaptive Model
   - Broadcast graphs under fully-adaptive Model
   - Results on fully-adaptive model
   - Results on non-adaptive model
4. Conclusion and Future Works

Introduction

- Growth of using computer networks,
- Great attention to all major problems in this area,
- Information dissemination,
- Broadcasting:
  - Process of distributing a message starting from a single node (*originator*) to all other nodes of the network using the network's links.

Outline

Preliminaries

- The network: $G = (V, E)$, originator $u \in V$.

Preliminaries

- The network: $G = (V, E)$, originator $u \in V$.
- $B_{cl}(u, G)$: minimum time required to finish the broadcasting from u.
- $B_{cl}(G) = \max\{B_{cl}(u, G) | u \in V(G)\}$
  - For any graph: $B_{cl}(G) \geq \lceil \log n \rceil$

Preliminaries

- The network: $G = (V, E)$, originator $u \in V$.
- $B_{cl}(u, G)$: minimum time required to finish the broadcasting from u.
- $B_{cl}(G) = \max\{B_{cl}(u, G) | u \in V(G)\}$
  - For any graph: $B_{cl}(G) \geq \lceil \log n \rceil$
- Two major problems in this area:
  - Broadcast time problem,
  - Network design.

Literature Review - Broadcast time problem

- Finding $B_{cl}(u, G)$ or $B_{cl}(G)$,
- Broadcast scheme: ordering of the neighbours of each vertex, depending on the originator:
  - u: originator,
  - once v gets informed, it will follow its list l_v^u,
  - Each vertex has to maintain up to $|V|$ different lists and know the originator to perform broadcasting.

[1] Peter J. Slater, Ernest J. Cockayne, and Stephen T. Hedetniemi. Information dissemination in trees. *SIAM Journal on Computing, 10(4):692–701, 1981.*

Literature Review - Broadcast time problem

- Finding $B_{cl}(u, G)$ or $B_{cl}(G)$,
- Broadcast scheme: ordering of the neighbours of each vertex, depending on the originator:
  - u: originator,
  - once v gets informed, it will follow its list l_v^u,
  - Each vertex has to maintain up to $|V|$ different lists and know the originator to perform broadcasting.
- NP-Complete in arbitrary graphs [1],
- Directions to follow:
  - Exact solution for a specific graph,
  - Heuristic,
  - Approximation algorithms.

[1] Peter J. Slater, Ernest J. Cockayne, and Stephen T. Hedetniemi. Information dissemination in trees. *SIAM Journal on Computing, 10(4):692–701, 1981.*

Literature Review - Broadcast time problem - cont.

- Broadcasting with universal lists:
  - ◇ Each vertex v has a single list l_v to follow, regardless of the originator.

[1] Slater, P.J., Cockayne, E.J. and Hedetniemi, S.T., 1981. Information dissemination in trees. *SIAM Journal on Computing*, 10(4), pp.692-701..
[2] Diks, K. and Pelc, A., 1996. Broadcasting with universal lists. *Networks*, 27(3), pp.183-196.

Literature Review - Broadcast time problem - cont.

- Broadcasting with universal lists:
  - Each vertex v has a single list l_v to follow, regardless of the originator.
- Two sub-models:
  - Non-adaptive $B_{na}(G)$: send to all vertices on the list,
  - Adaptive $B_a(G)$: skip the ones you **received from**!

[1] Slater, P.J., Cockayne, E.J. and Hedetniemi, S.T., 1981. Information dissemination in trees. *SIAM Journal on Computing*, 10(4), pp.692-701..
[2] Diks, K. and Pelc, A., 1996. Broadcasting with universal lists. *Networks*, 27(3), pp.183-196.

Literature Review - Broadcast time problem - cont.

- Broadcasting with universal lists:
  - Each vertex v has a single list l_v to follow, regardless of the originator.
- Two sub-models:
  - Non-adaptive $B_{na}(G)$: send to all vertices on the list,
  - Adaptive $B_a(G)$: skip the ones you **received from**!
- Introduced indirectly by Slater [1]; for any Tree, $B_{cl}(T) = B_a(T)$.
- Diks and Pelc [2] distinguished between adaptive and non-adaptive models,
  - Also proposed several broadcast schemes for different graphs
- Long list of research ...

[1] Slater, P.J., Cockayne, E.J. and Hedetniemi, S.T., 1981. Information dissemination in trees. *SIAM Journal on Computing*, 10(4), pp.692-701..
[2] Diks, K. and Pelc, A., 1996. Broadcasting with universal lists. *Networks*, 27(3), pp.183-196.

Literature Review - Network Design

- Graph G on n vertices is a broadcast graph *(bg)* under classical model if $B_{cl}(G) = \lceil \log n \rceil$,
- A *bg* with minimum number of edges is called a minimum broadcast graph *(mbg)*,
- The number of edges of an *mbg* on n vertices: $B(n)$ or $B^{(cl)}(n)$.

Literature Review - Network Design - cont.

- $B^{(cl)}(n)$ is known for very few n,
- Exact values:
  - ◇ $n \leq 32$, except for $23, 24, 25$.
  - ◇ $n = 2^k$, Hypercubes | Knodel Graph
  - ◇ $n = 2^k - 2$, Knodel Graph
- Several upper bounds and lower bounds ...
- No result on universal lists model!

Outline

Fully-adaptive Model

- Another sub-model for universal lists,
- A universal list L_u is maintained at each vertex u,
- Once informed, follow the list and **skip all informed vertices!**
  - ◇ Similarly to the classical model: No unnecessary calls!

Fully-adaptive Model

- Another sub-model for universal lists,
- A universal list L_u is maintained at each vertex u,
- Once informed, follow the list and **skip all informed vertices!**
  - ◇ Similarly to the classical model: No unnecessary calls!
- **Theorem 3.1.** $B_{cl}(G) \leq B_{fa}(G) \leq B_a(G) \leq B_{na}(G)$, *for any graph* G.
-

Model	Symbol	No. of unnecessary calls	Space Complexity	Speed
Non-adaptive	$B_{na}(G)$	Many	Very Low	Very Slow
Adaptive	$B_a(G)$	Few	Low	Slow
Fully Adaptive	$B_{fa}(G)$	0	Moderate	Moderate
Classical	$B_{cl}(G)$	0	Very High	Very Fast

Fully-adaptive Model - Definitions

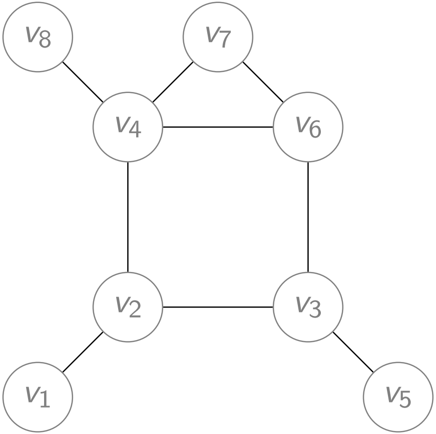
- A broadcast scheme: Matrix $\sigma_{n \times \Delta}$,
  - Row i of σ corresponds to an ordering for vertex v_i.
- Set of all possible schemes: Σ.

Fully-adaptive Model - Definitions

- A broadcast scheme: Matrix $\sigma_{n \times \Delta}$,
  - Row i of σ corresponds to an ordering for vertex v_i.
- Set of all possible schemes: Σ.
- Let $M \in \{na, a, fa\}$ be a model:
  - $B_M^\sigma(v, G)$: the time steps needed to inform all the vertices in G from v while following σ under M,
  - $B_M^\sigma(G) = \max_{v \in V}\{B_M^\sigma(v, G)\}$,
  - $B_M(G) = \min_{\sigma \in \Sigma}\{B_M^\sigma(G)\}$.

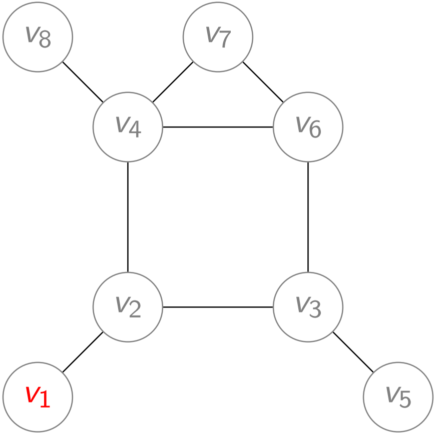
Fully-adaptive model - Example

Sender	Ordering of receivers			
v_1	v_2	Null	Null	Null
v_2	v_3	v_4	v_1	Null
v_3	v_2	v_6	v_5	Null
v_4	v_2	v_6	v_8	v_7
v_5	v_3	Null	Null	Null
v_6	v_3	v_7	v_4	Null
v_7	v_6	v_4	Null	Null
v_8	v_4	Null	Null	Null



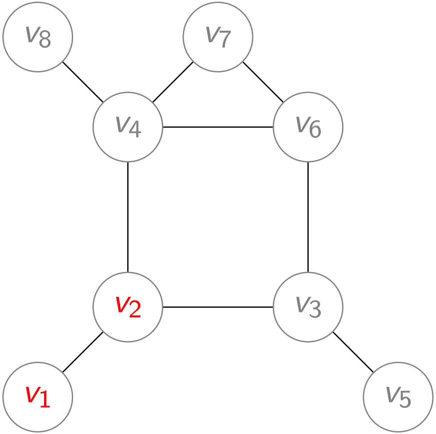
Fully-adaptive model - Example

Sender	Ordering of receivers			
v_1	v_2	Null	Null	Null
v_2	v_3	v_4	v_1	Null
v_3	v_2	v_6	v_5	Null
v_4	v_2	v_6	v_8	v_7
v_5	v_3	Null	Null	Null
v_6	v_3	v_7	v_4	Null
v_7	v_6	v_4	Null	Null
v_8	v_4	Null	Null	Null



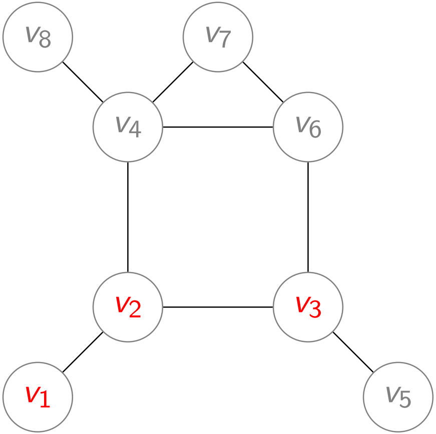
Fully-adaptive model - Example

Sender	Ordering of receivers			
v_1	v_2	Null	Null	Null
v_2	v_3	v_4	v_1	Null
v_3	v_2	v_6	v_5	Null
v_4	v_2	v_6	v_8	v_7
v_5	v_3	Null	Null	Null
v_6	v_3	v_7	v_4	Null
v_7	v_6	v_4	Null	Null
v_8	v_4	Null	Null	Null



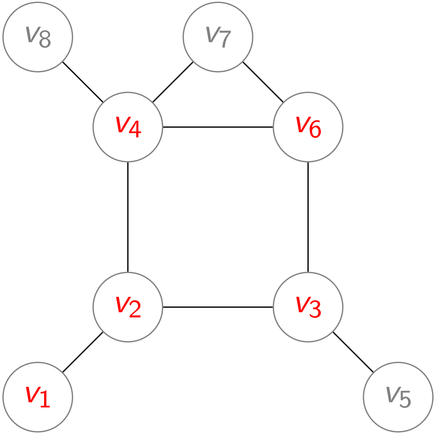
Fully-adaptive model - Example

Sender	Ordering of receivers			
v_1	v_2	Null	Null	Null
v_2	v_3	v_4	v_1	Null
v_3	v_2	v_6	v_5	Null
v_4	v_2	v_6	v_8	v_7
v_5	v_3	Null	Null	Null
v_6	v_3	v_7	v_4	Null
v_7	v_6	v_4	Null	Null
v_8	v_4	Null	Null	Null



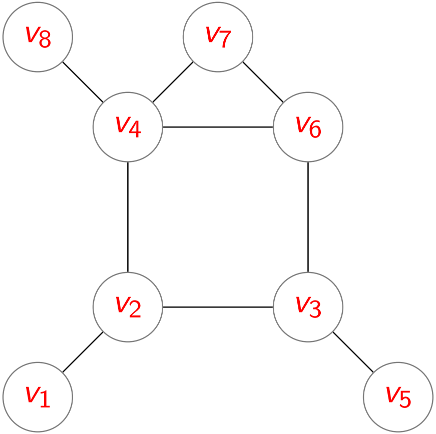
Fully-adaptive model - Example

Sender	Ordering of receivers			
v_1	v_2	Null	Null	Null
v_2	v_3	v_4	v_1	Null
v_3	v_2	v_6	v_5	Null
v_4	v_2	v_6	v_8	v_7
v_5	v_3	Null	Null	Null
v_6	v_3	v_7	v_4	Null
v_7	v_6	v_4	Null	Null
v_8	v_4	Null	Null	Null



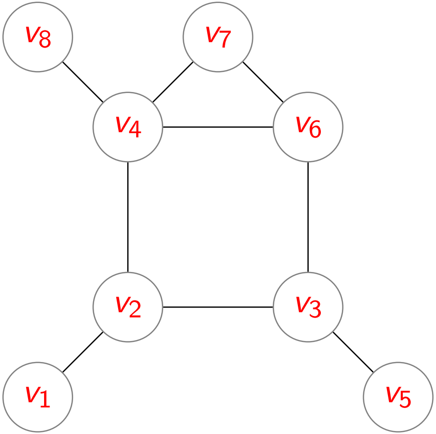
Fully-adaptive model - Example

Sender	Ordering of receivers			
v_1	v_2	Null	Null	Null
v_2	v_3	v_4	v_1	Null
v_3	v_2	v_6	v_5	Null
v_4	v_2	v_6	v_8	v_7
v_5	v_3	Null	Null	Null
v_6	v_3	v_7	v_4	Null
v_7	v_6	v_4	Null	Null
v_8	v_4	Null	Null	Null



Fully-adaptive model - Example

Sender	Ordering of receivers			
v_1	v_2	Null	Null	Null
v_2	v_3	v_4	v_1	Null
v_3	v_2	v_6	v_5	Null
v_4	v_2	v_6	v_8	v_7
v_5	v_3	Null	Null	Null
v_6	v_3	v_7	v_4	Null
v_7	v_6	v_4	Null	Null
v_8	v_4	Null	Null	Null



- $B^{\sigma}_{fa}(v_1, G) = 4$, while $B^{\sigma}_{a}(v_1, G) = 5$ and $B^{\sigma}_{na}(v_1, G) = 6$.

Fully-adaptive Model - AAA

- **Assumptions:**
  - ◇ None-faulty network with established links,
  - ◇ Unique and heavy message,
  - ◇ The message: header + payload,

Fully-adaptive Model - AAA

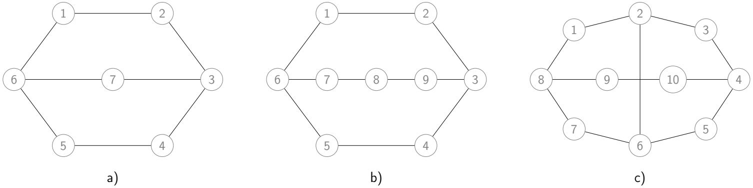
- **Assumptions:**
  - None-faulty network with established links,
  - Unique and heavy message,
  - The message: header + payload,
- **Architecture:**
  - How to know the state of each neighbour?
    - Push model,
    - Pull model,

Fully-adaptive Model - AAA

- **Assumptions:**
  - None-faulty network with established links,
  - Unique and heavy message,
  - The message: header + payload,
- **Architecture:**
  - How to know the state of each neighbour?
    - Push model,
    - Pull model,
- **Applications:**
  - Update procedure in SDNs:
    - Changing routing policies, adjusting links' weights, etc.
    - The data plane only forwards packets,
    - Routing and load balancing decisions are made in a centralized controller,
    - The network manager must optimize the forwarding tables (broadcast schemes).

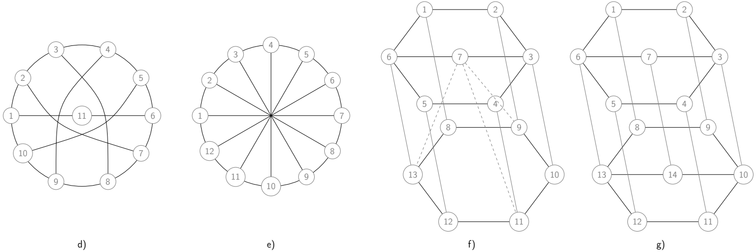
Broadcast graphs under fully-adaptive model

- **Lemma 3.2.** *If there is a graph G on n vertices for which $B_{fa}(G) = \lceil \log n \rceil$, then $B^{(cl)}(n) \leq B^{(fa)}(n)$.*
- mbg's for $n \leq 10$:



Broadcast graphs under fully-adaptive model - cont.

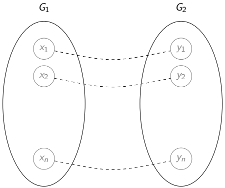
- bg's for $11 \leq n \leq 14$:



n	3	4	5	6	7	8	9	10	11	12	13	14
Lower bound on $B^{(fa)}(n)$	2	4	5	6	8	12	10	12	13	15	18	21
Upper bound on $B^{(fa)}(n)$	2	4	5	6	8	12	10	12	16	18	23	23

Broadcast graphs under fully-adaptive model - cont.

- General construction of bg's:
  - **Lemma 3.4.** *Consider a graph $G = (V, E)$ with n vertices, m edges, and $B_{fa}(G) = \tau$. It is always possible to construct a graph $G' = (V', E')$ with $2n$ vertices, $2m + n$ edges, and $B_{fa}(G') = \tau + 1$.*
  - $(G, n, m, \tau) \to (G', 2n, 2m + n, \tau + 1)$.



  - **Corollary 3.7.** *For any positive k,* $(G, n, m, \lceil \log n \rceil) \to (G', 2^k n, 2^k m + k2^{k-1} n, \lceil \log n \rceil + k)$.

Broadcast graphs under fully-adaptive model - cont.

- This yeilds 4 infinite families of bg's under fully-adaptive model:
  - **Theorem 3.9.** *For any integer* $k = \lceil \log n \rceil \geq 4$*:*

$$B^{(fa)}(n) = B^{(fa)}(2^{k-1} + 2^{k-4}) \leq \frac{n\lceil \log n \rceil}{2} - \frac{8n}{9},$$

$$B^{(fa)}(n) = B^{(fa)}(2^{k-1} + 2^{k-3}) \leq \frac{n\lceil \log n \rceil}{2} - \frac{4n}{5},$$

$$B^{(fa)}(n) = B^{(fa)}(2^{k-1} + 2^{k-2}) \leq \frac{n\lceil \log n \rceil}{2} - \frac{n}{2},$$

$$B^{(fa)}(n) = B^{(fa)}(2^{k-1} + 2^{k-2} + 2^{k-3}) \leq \frac{n\lceil \log n \rceil}{2} - \frac{5n}{14}.$$

Results on fully-adaptive model

- Trees T:
  - **Theorem 3.10.** $B_{cl}(T) = B_{fa}(T) = B_a(T)$.

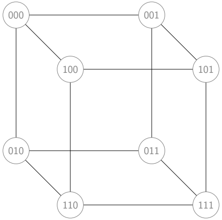
Results on fully-adaptive model

- Trees T:
  - **Theorem 3.10.** $B_{cl}(T) = B_{fa}(T) = B_a(T)$.
- Grids $G_{m\times n}$:
  - **Corollary 3.11.** $B_{fa}(G_{m\times n}) = m + n - 2$.

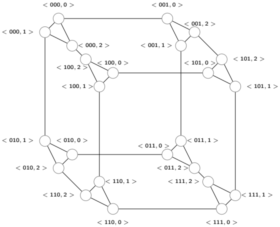
Results on fully-adaptive model

- Trees T:
  - **Theorem 3.10.** $B_{cl}(T) = B_{fa}(T) = B_a(T)$.
- Grids $G_{m\times n}$:
  - **Corollary 3.11.** $B_{fa}(G_{m\times n}) = m + n - 2$.
- Tori $T_{m\times n}$:
  - **Theorem 3.12.**
    - $B_{fa}(T_{m\times n}) = \lfloor \frac{n}{2} \rfloor + \lfloor \frac{m}{2} \rfloor$, if n and m are even,
    - $B_{fa}(T_{m\times n}) = \lfloor \frac{n}{2} \rfloor + \lfloor \frac{m}{2} \rfloor + 1$, if one of m and n is even and the other one is odd,
    - $\lfloor \frac{n}{2} \rfloor + \lfloor \frac{m}{2} \rfloor + 1 \leq B_{fa}(T_{m\times n}) \leq \lfloor \frac{n}{2} \rfloor + \lfloor \frac{m}{2} \rfloor + 2$, if both m and n are odd.

Results on fully-adaptive model - cont.



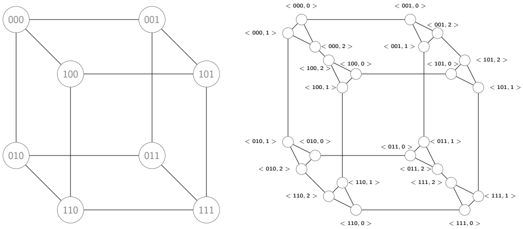
a) H_3



b) CCC_3

- Hypercubes H_d:
  - **Theorem 3.13.** $B_{fa}(H_d) = d$.

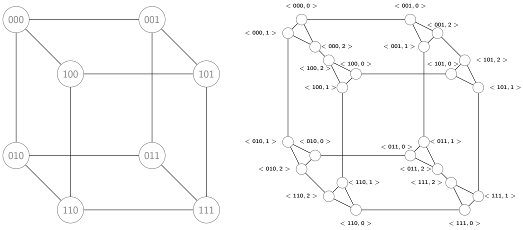
Results on fully-adaptive model - cont.



a) H_3 b) CCC_3

- Hypercubes H_d:
  - **Theorem 3.13.** $B_{fa}(H_d) = d$.
    - **Corollary 3.14.** *Hypercube H_d is an mbg on 2^d vertices under the fully-adaptive model.*

Results on fully-adaptive model - cont.



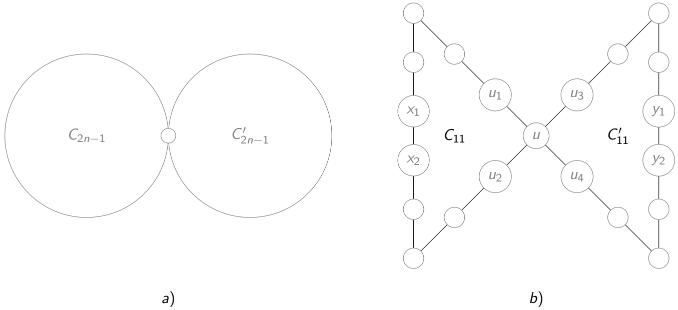
a) H_3 b) CCC_3

- Hypercubes H_d:
  - **Theorem 3.13.** $B_{fa}(H_d) = d$.
    - **Corollary 3.14.** *Hypercube H_d is an mbg on 2^d vertices under the fully-adaptive model.*
- Cube Connected Cycles CCC_d:
  - **Theorem 3.15.** $B_{fa}(CCC_d) = \lceil \frac{5d}{2} \rceil - 1$.

Results on fully-adaptive model - cont.

- Is $B_{cl}(G) = B_{fa}(G)$ always?

- Is $B_{cl}(G) = B_{fa}(G)$ always?
  - No!
  - **Proposition 3.16.** *There exists graph G with $B_{cl}(G) < B_{fa}(G)$:*



Results on non-adaptive model

- Complete $k-$ary trees $T_{k,h}$:
  - From known bounds on trees:
  - **Observation 3.17.** $\lceil \frac{6h-1}{2} \rceil \leq B_{na}(T_{k,h}) \leq kh + 2h - 1$.
  - We will show that the upper bound is tight.

Results on non-adaptive model

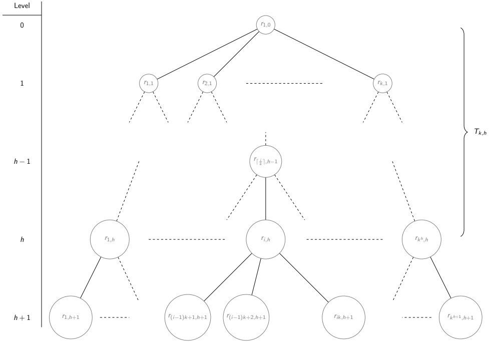
- Complete $k-$ary trees $T_{k,h}$:
  - From known bounds on trees:
  - **Observation 3.17.** $\lceil \frac{6h-1}{2} \rceil \leq B_{na}(T_{k,h}) \leq kh + 2h - 1.$
  - We will show that the upper bound is tight.
- $B_{na}(T_{k,h}) \geq kh + 2h - 1.$
  - Proof by induction on h:

Results on non-adaptive model

- Complete $k-$ary trees $T_{k,h}$:
  - From known bounds on trees:
  - **Observation 3.17.** $\lceil \frac{6h-1}{2} \rceil \leq B_{na}(T_{k,h}) \leq kh + 2h - 1$.
  - We will show that the upper bound is tight.
- $B_{na}(T_{k,h}) \geq kh + 2h - 1$.
  - Proof by induction on h:
    - Base case is easy $h = 1$...
    - I.H: $B_{na}(T_{k,h}) \geq kh + 2h - 1$,
    - I.S: $B_{na}(T_{k,h+1}) \geq kh + 2h + k + 1$.

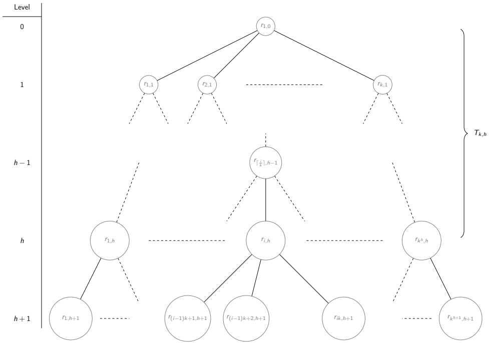
Results on non-adaptive model

- Fix a vertex at level $h+1$ as the originator,
- At $t=1$ it sends to its parent under any scheme,
- Until $t = kh + 2h$ all vertices in the first h level will be informed (based on I.H).



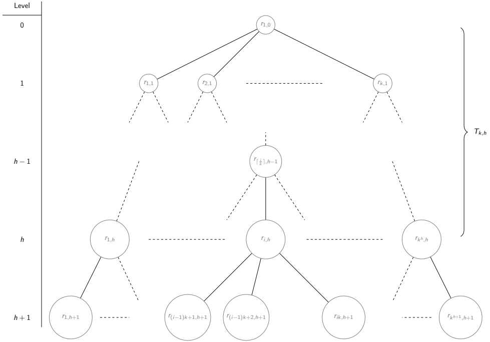
Results on non-adaptive model

- Fix a vertex at level $h+1$ as the originator,
- At $t=1$ it sends to its parent under any scheme,
- Until $t=kh+2h$ all vertices in the first h level will be informed (based on I.H).
- $\exists$ a vertex s.t. $r_{i,h}$ that gets informed at $t=kh+2h$.



Results on non-adaptive model

- Fix a vertex at level $h+1$ as the originator,
- At $t=1$ it sends to its parent under any scheme,
- Until $t=kh+2h$ all vertices in the first h level will be informed (based on I.H).
- $\exists$ a vertex s.t. $r_{i,h}$ that gets informed at $t=kh+2h$.
  - If $r_{i,h}$ first sends to its parent, I.S. follows.
  - If not, pick the first vertex that $r_{i,h}$ sends the message to as the originator. I.S. follows.



Results on non-adaptive model

- Complete $k-$ary trees $T_{k,h}$:
  - **Theorem 3.18.** $B_{na}(T_{k,h}) = kh + 2h - 1$.

Results on non-adaptive model

- Complete $k-$ary trees $T_{k,h}$:
  - **Theorem 3.18.** $B_{na}(T_{k,h}) = kh + 2h - 1$.
- Binomial trees T_d:
  - **Proposition 3.19.** $B_{na}(T_d) = 3d - 2$.

Results on non-adaptive model

- Complete $k-$ary trees $T_{k,h}$:
  - **Theorem 3.18.** $B_{na}(T_{k,h}) = kh + 2h - 1$.
- Binomial trees T_d:
  - **Proposition 3.19.** $B_{na}(T_d) = 3d - 2$.
- Complete Bipartite graph $K_{m\times n}$:
  - **Theorem 3.21.**
  - $$B_{na}(K_{m\times n}) \leq \begin{cases} \lceil \log n \rceil + 1 + \max\{\lceil \frac{m-2^{\lceil \log n \rceil}}{n} \rceil, 0\} + \\ 3 \times \lceil \sqrt{\lceil \log n \rceil + 1 + \max\{\lceil \frac{m-2^{\lceil \log n \rceil}}{n} \rceil, 0\}} \rceil, & \text{if } m > n. \\ \lceil \log n \rceil + 1 + 3 \times \lceil \sqrt{\lceil \log n \rceil + 1} \rceil, & \text{if } m = n. \end{cases}$$

Results on non-adaptive model - cont.

- A general upper bound for trees:
  - **Theorem 3.22.** $B_{na}(T) \leq B_{cl}(T) + \lfloor \frac{diam(T)}{2} \rfloor$.

Results on non-adaptive model - cont.

- A general upper bound for trees:
  - **Theorem 3.22.** $B_{na}(T) \leq B_{cl}(T) + \lfloor \frac{diam(T)}{2} \rfloor$.
- Tightest bounds on trees:
  - **Theorem 3.23.**
  - $$\max\left\{B_{cl}(T)+1, \lceil \frac{3.diam(T)-1}{2} \rceil\right\} \leq B_{na}(T) \leq \min\left\{B_{cl}(T) + \lfloor \frac{diam(T)}{2} \rfloor, b_{cl}(T) + diam(T)\right\}.$$

Outline

Conclusion

- Our contribution so far:
  - Suggesting fully-adaptive model,
    - *mbg*'s for $n \leq 10$,
    - *bg*'s for $11 \leq n \leq 14$,
    - The first infinite family of *bg*'s under universal lists model,
    - Exact value of $B_{fa}(G)$ for: trees, grids, hypercubes, cube connected cycles.
    - Upper bound on $B_{fa}(G)$ for tori.

Conclusion

- Our contribution so far:
  - Suggesting fully-adaptive model,
    - *mbg*'s for $n \leq 10$,
    - *bg*'s for $11 \leq n \leq 14$,
    - The first infinite family of *bg*'s under universal lists model,
    - Exact value of $B_{fa}(G)$ for: trees, grids, hypercubes, cube connected cycles.
    - Upper bound on $B_{fa}(G)$ for tori.
  - For non-adaptive model,
    - Exact value of $B_{na}(G)$ for: k-ary trees, binomial trees,
    - Upper bound on $B_{na}(G)$ for complete bipartite graph,
    - A general upper bound for trees.

Future Works

- Developing a comprehensive framework based on Genetic Algorithm for all three models using universal lists.

Future Works

- Developing a comprehensive framework based on Genetic Algorithm for all three models using universal lists.
- A candidate solution to the problem: A matrix $\sigma_{n \times \Delta(G)}$.
- Generate several random solutions,
- Using crossover and mutation over multiple generations, a relatively good solution could be found.

Future Works

- Developing a comprehensive framework based on Genetic Algorithm for all three models using universal lists.
- A candidate solution to the problem: A matrix $\sigma_{n \times \Delta(G)}$.
- Generate several random solutions,
- Using crossover and mutation over multiple generations, a relatively good solution could be found.
- Pros:
  - Works for arbitrary graphs,
  - Works for all three models,
  - Several fitness functions could be defined,
  - Efficient in terms of time complexity,
  - Gives the actual broadcast scheme,
  - The scheme could be used separately for proving many results,
  - ...

Future Works - cont.

- Generalizing our idea published in [1]

[1] S. Gholami and H. A. Harutyunyan. A broadcasting heuristic for hypercube of trees. In *2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 355–361, 2021.

Future Works - cont.

- Generalizing our idea published in [1]
- Previously, we considered classical broadcasting in hypercube of trees,
- Now we replace hypercube with any broadcast graph (such as Knodel graph),

[1] S. Gholami and H. A. Harutyunyan. A broadcasting heuristic for hypercube of trees. In *2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 355–361, 2021.

Future Works - cont.

- Generalizing our idea published in [1]
- Previously, we considered classical broadcasting in hypercube of trees,
- Now we replace hypercube with any broadcast graph (such as Knodel graph),
- Goals:
  - ◇ Find exact broadcast time when there is only one tree,
  - ◇ Approximation algorithm with several trees,
  - ◇ A heuristic by improving our previous work.

[1] S. Gholami and H. A. Harutyunyan. A broadcasting heuristic for hypercube of trees. In *2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 355–361, 2021.

Time table

No.		Description	Date	Done?
T1	Task 1	Broadcasting heuristic in Hypercube of Trees	Nov. 2020	✓
O1	Outcome 1	Conference paper	Dec. 2020	✓ [1]
T2	Task 2	Optimal broadcasting in Fully Connected Trees	Aug. 2021	✓
O2	Outcome 2	Journal paper	Sep. 2021	✓ [2]
T3	Task 3	Fully-adaptive model for universal lists	Dec. 2021	✓
O3	Outcome 3	Journal paper	Jan. 2022	✓ [3]
T4	Task 4	Broadcast graphs under universal lists model	Mar. 2022	✓
O4	Outcome 4	Conference + Ph.D. Seminar	Apr. 2022	✓ [4]
T5	Task 5	Heuristic for graphs with base being a broadcast graph	May. 2022	
O5	Outcome 5	Conference paper	Jun. 2022	
T6	Task 6	GA for broadcasting with universal lists	Jul. 2022	
O6	Outcome 6	Journal paper	Aug. 2022	
T7	Task 7	Thesis writing	Sep. 2022	
O7	Outcome 7	Ph.D. defense	Oct. 2022	

[1] S. Gholami and H. A. Harutyunyan. A broadcasting heuristic for hypercube of trees. In *2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 355–361, 2021.
[2] S. Gholami and H. A. Harutyunyan. Optimal broadcasting in fully connected trees. Submitted to *Journal of Interconnection Networks*, 2021.
[3] S. Gholami and H. A. Harutyunyan. Fully-adaptive model for broadcasting with universal lists. Submitted to *Parallel Processing Letters*, 2022.
[4] S. Gholami and H. A. Harutyunyan. Broadcast graphs with nodes of limited memory. Submitted to *33rd International Workshop on Combinatorial Algorithms (IWOCA)*, 2022.

Thanks a bunch!