Computing Local Structures in Radio Networks

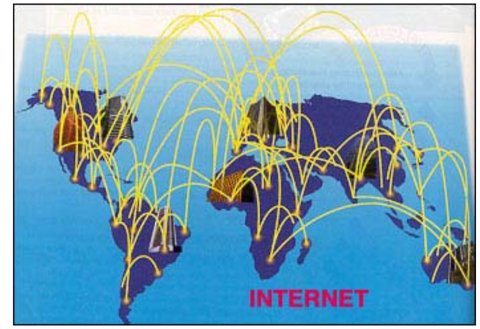
Thomas Moscibroda

ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed
Computing
Group

Locality...

- Many modern networks are large-scale and highly complex
  - Internet
  - Peer-to-Peer Networks
  - Wireless Sensor Networks
  - Human Brain, Society...?



LOCALITY

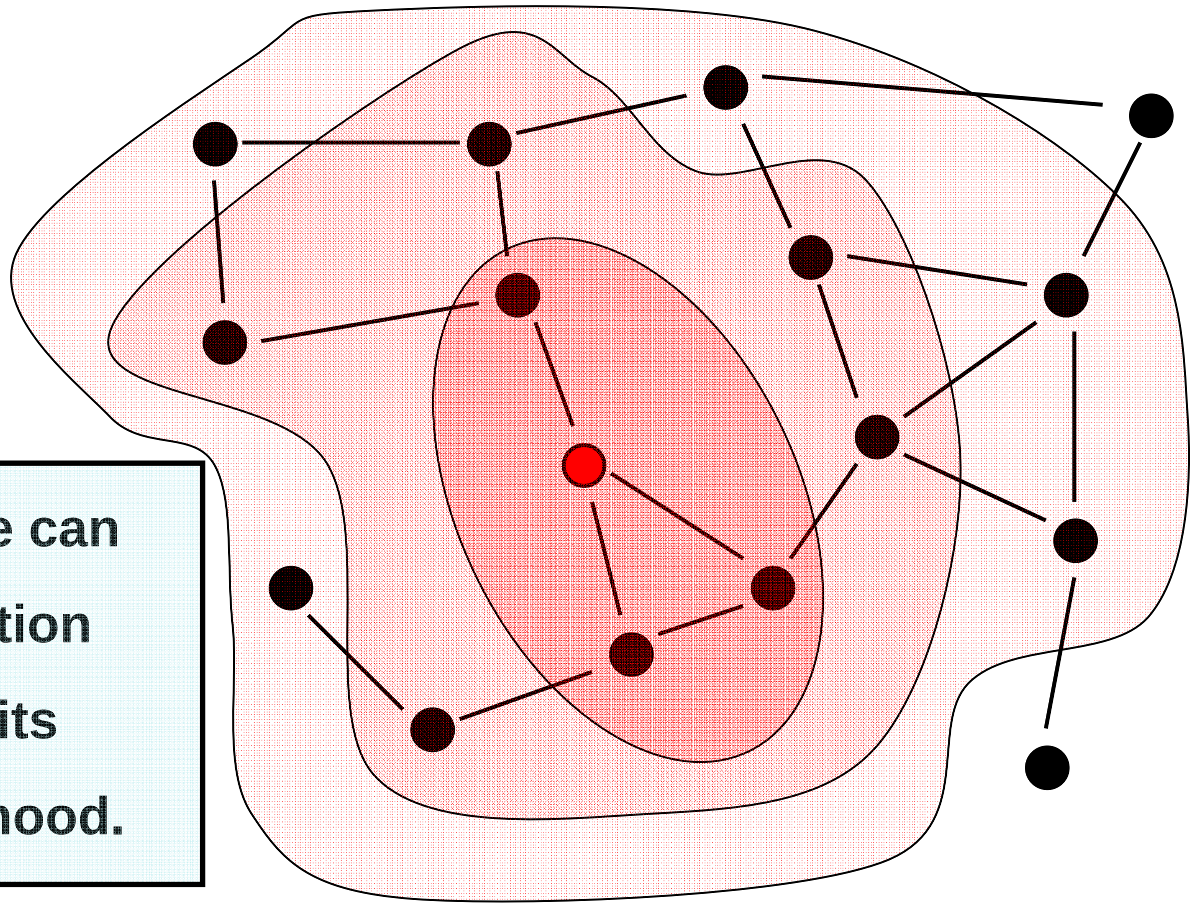
→ **No node has global information**

→ **Each node can gather information from its neighborhood only (local information)**

→ **Yet, nodes have to come up with a global goal!**

Local Computation

In 3 communication round(s):



In time k, a node can obtain information from at most its k-hop neighborhood.

- Fundamental trade-off between the amount of communication and the quality of the global solution!

LOCALITY
Communication Rounds

TRADE-OFF

GLOBAL GOAL
e.g. Approximation

- **Upper Bounds:**

  → Local Distributed (Approximation) Algorithms

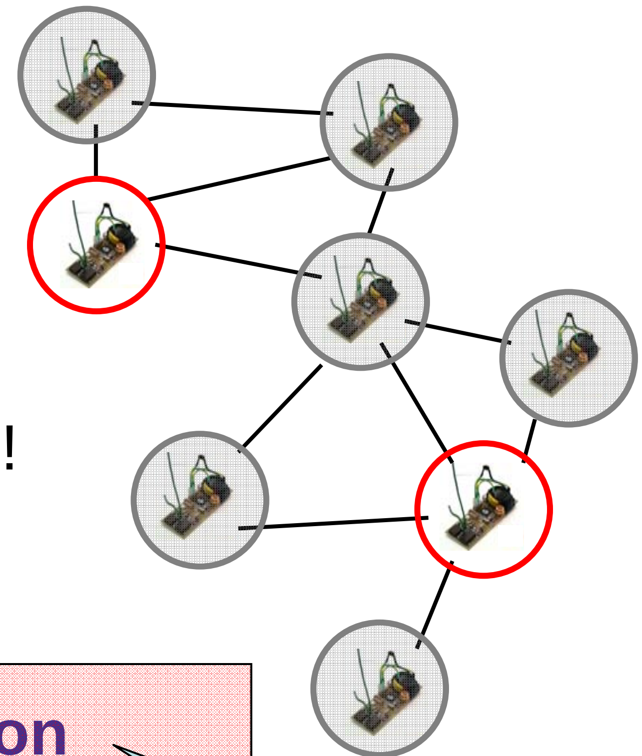
- **Lower Bounds:**

  → Time Lower Bounds

  → Hardness of Distributed Approximation

Clustering

- Clustering in Radio Networks
- Choose clusterheads such that
  - Every node is either a clusterhead or...
  - ...has a clusterhead in its neighborhood.



- Goal: We want only few clusterheads!

Inherent Problem:

→ Nodes have only local information

→ Nodes have to optimize a global goal

If we want fast algorithms!

The Importance of Being Clustered...

- In wireless multi-hop networks,...
- ... clustering helps in structuring the network.

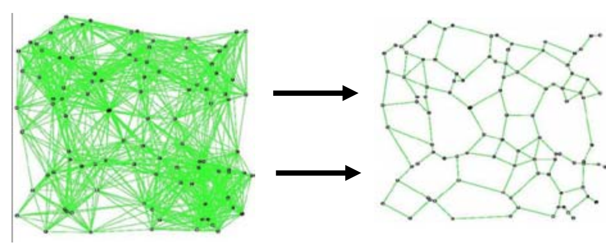
- Particularly, clustering helps in...

  A) ...facilitating communication between distant nodes
  - Virtual Backbone routing

  B) ...organizing communication between adjacent nodes
  - MAC layer, spatial multiplexing, topology control

  C) ...improving energy efficiency
  - Synchronized Sleep/Awake schedules within a cluster

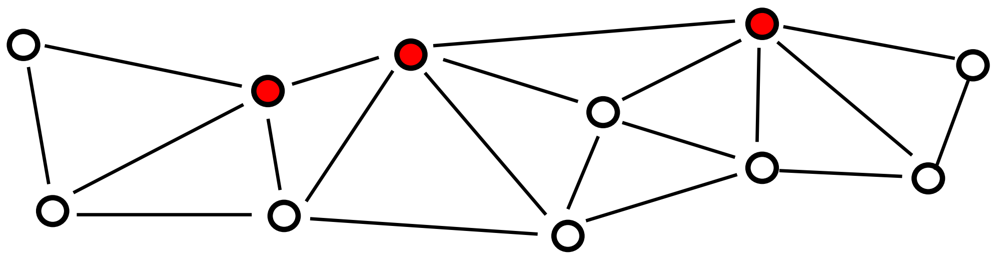
  D) ...helps in initializing the network



What kind of clustering...?

- Minimum Dominating Set (MDS)

  (Choose minimum S ⊆ V, s.t. each v∈V is in S or has at least one neighbor in S)

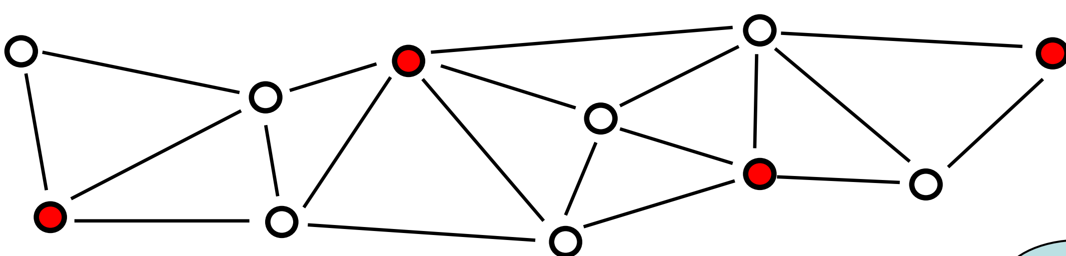


What can be computed locally?

[Naor, Stockmeyer, 93]

- Maximal Independent Set (MIS)

  (Choose a dominating set without neighboring dominators)



Both problems appear to be local in nature!

The Locality of Clustering

LOCALITY ⟺ GLOBAL GOAL

In this talk, I give an overview of recent results on the locality of clustering.

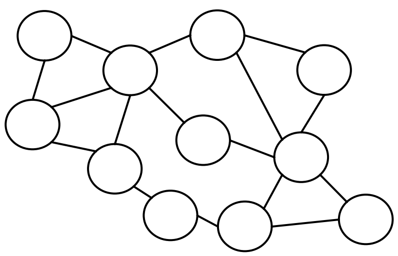
Unfortunately, no time for proofs...

Outline:

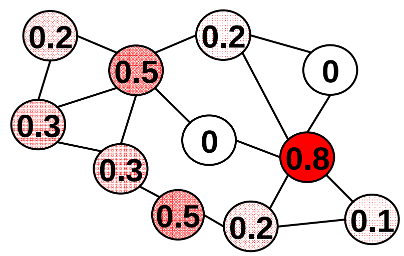
1. Locality and Distributed Algorithms
2. Clustering in Radio Networks
3. Results and Techniques
   a) Unit Disk Graphs vs. General Graphs
   b) Graphs with Bounded Independence
   c) Unstructured Radio Networks
4. Conclusions

Distributed MDS Algorithm - Overview

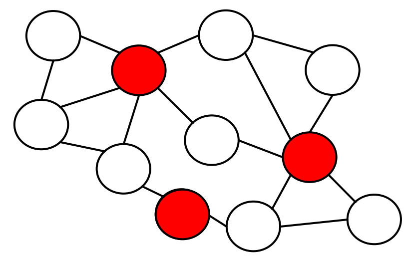
Input:
Local Graph



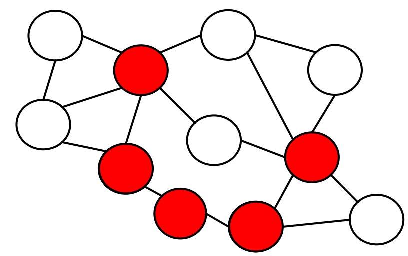
Fractional
Dominating Set



Dominating
Set



Connected
Dominating Set



Phase A:
Distributed
linear program
rel. high degree
gives high value
O(k^2) rounds

Phase B:
Probabilistic
Algorithm
O(1) rounds

Phase C:
Connect DS
by “tree” of
“bridges”
O(1) rounds

Upper Bounds

- First algorithm by Kuhn and Wattenhofer [PODC 2003]
- Improved algorithms [Kuhn, Moscibroda, Wattenhofer @ SODA 2006]:

  a) Algorithm computes an $O(\Delta^{1/k})$-approximation of phase A with logarithmic sized messages in $O(k^2)$ rounds

  → $O(\log^2\Delta / \varepsilon^4)$ time for a $(1+\varepsilon)$-approximation

  b) If messages can be of unbounded size, algorithm computes an $O(n^{1/k})$-approximation in $O(k)$ rounds

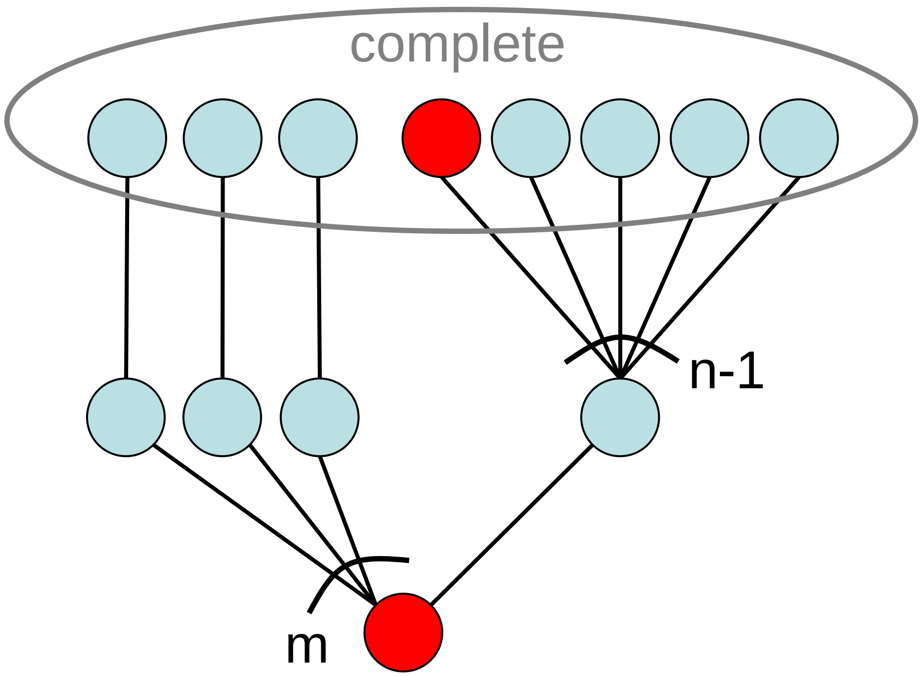
  → constant approximation in $O(\log n)$

Locality is only constraint!

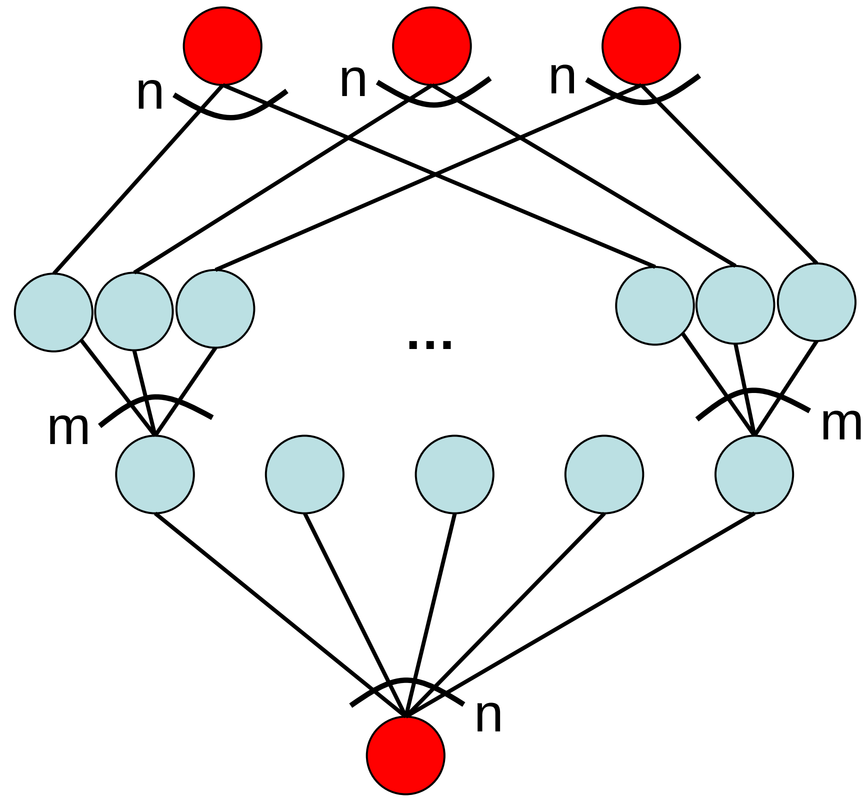
Is it any good?

Locality Lower Bounds: Intuition…

- Two graphs (m << n). Optimal dominating sets are marked red.



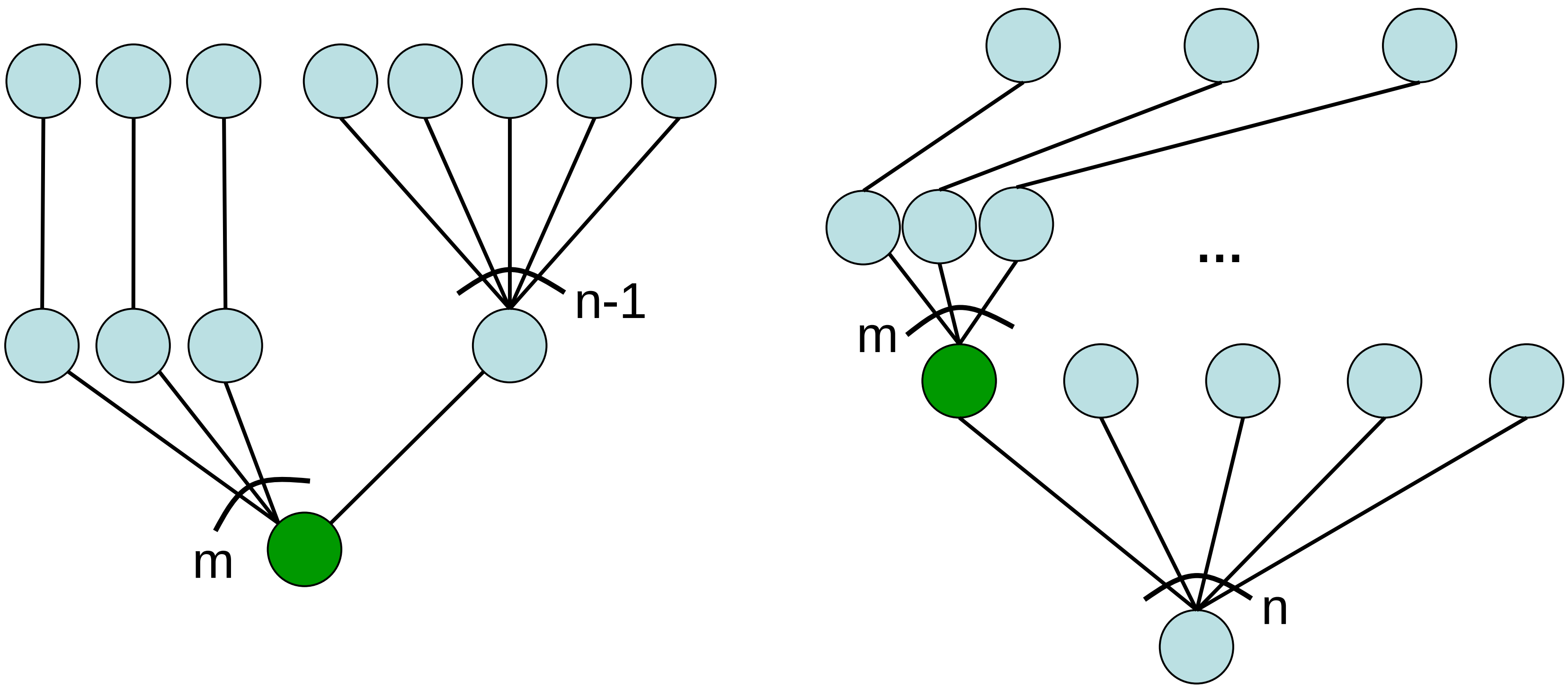
$|DS_{OPT}| = 2.$



$|DS_{OPT}| = m+1.$

Locality Lower Bounds: Intuition…

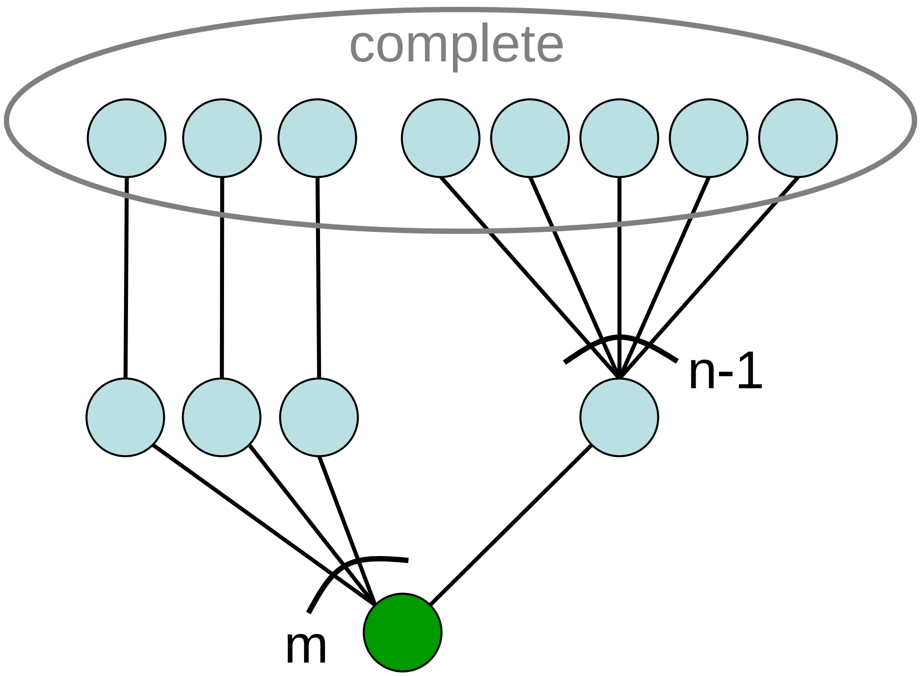
- In local algorithms, nodes must decide only using local knowledge.
- In the example green nodes see exactly the same neighborhood.

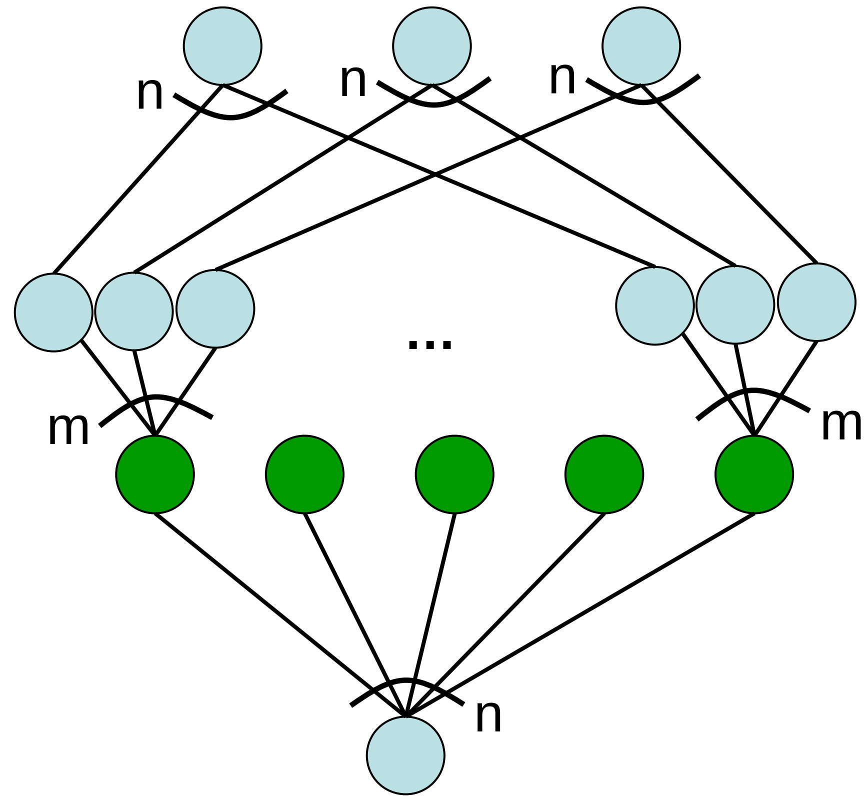


- So these green nodes must decide the same way!

Locality Lower Bounds: Intuition…

- But however they decide, one way will be devastating (with $n = m^2$)!





$|DS_{OPT}| = 2.$

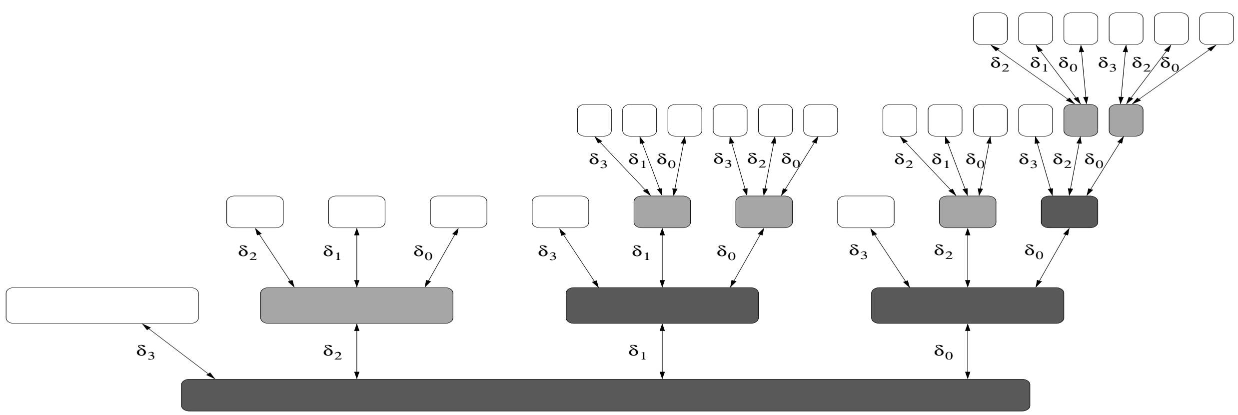
$|DS_{OPT\ without\ green}| \geq m.$

$|DS_{OPT}| = m+1.$

$|DS_{OPT\ with\ green}| > n$

The Lower Bound

- Locality lower bounds (Kuhn, Moscibroda, Wattenhofer @ PODC 04):
  - Model: In a network/graph G (nodes = processors), each node can exchange an unbounded message with all its neighbors for k rounds. After k rounds, node need to decide.
  - We construct the graph such that there are nodes that see the same neighborhood up to distance k. We show that node ID’s do not help, and using Yao’s principle also randomization does not.



The Lower Bound - Results

LOCALITY ⟷ **GLOBAL GOAL**

In k communication rounds, no algorithm can approximate MDS better than $\Omega(n^{c/k^2}/k)$ or $\Omega(\Delta^{1/k}/k)$.

holds even if...

- randomized...
- unbounded messages...
- unique IDs in [1..n]...
- synchronous model...

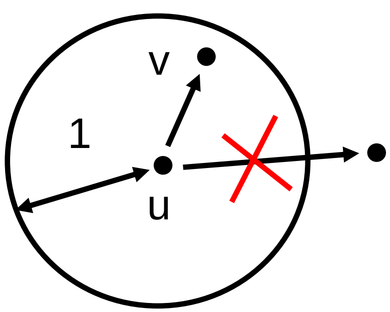
For polylogarithmic (or constant) approximation, every algorithm requires at least time

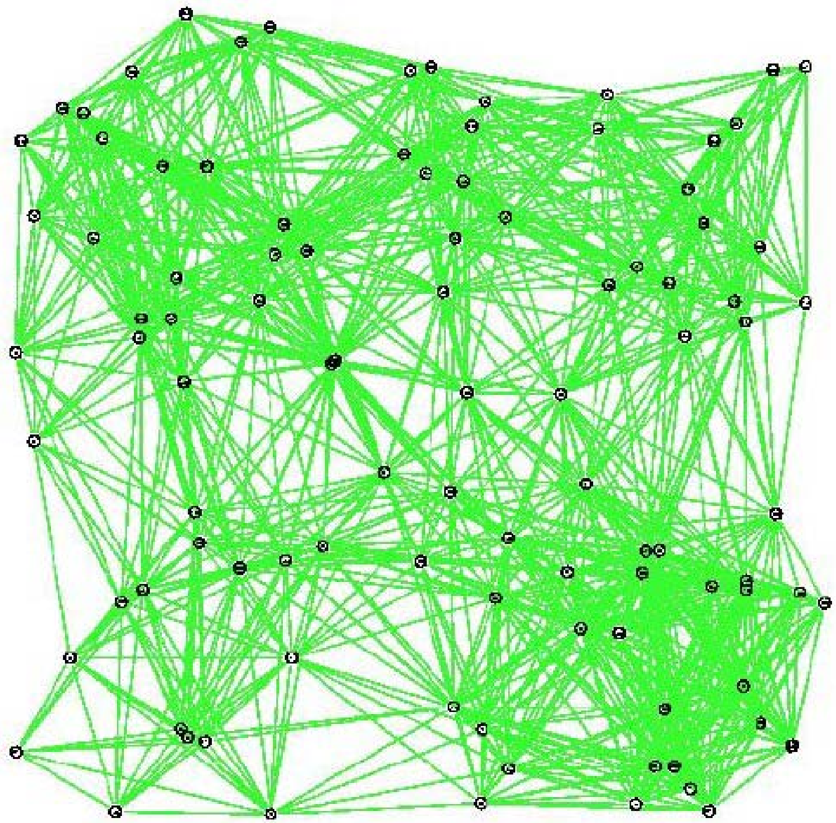
$$\Omega\left(\sqrt{\frac{\log n}{\log\log n}}\right) \quad or \quad \Omega\left(\frac{\log \Delta}{\log\log \Delta}\right)$$

The same time bounds hold for distributed MIS!

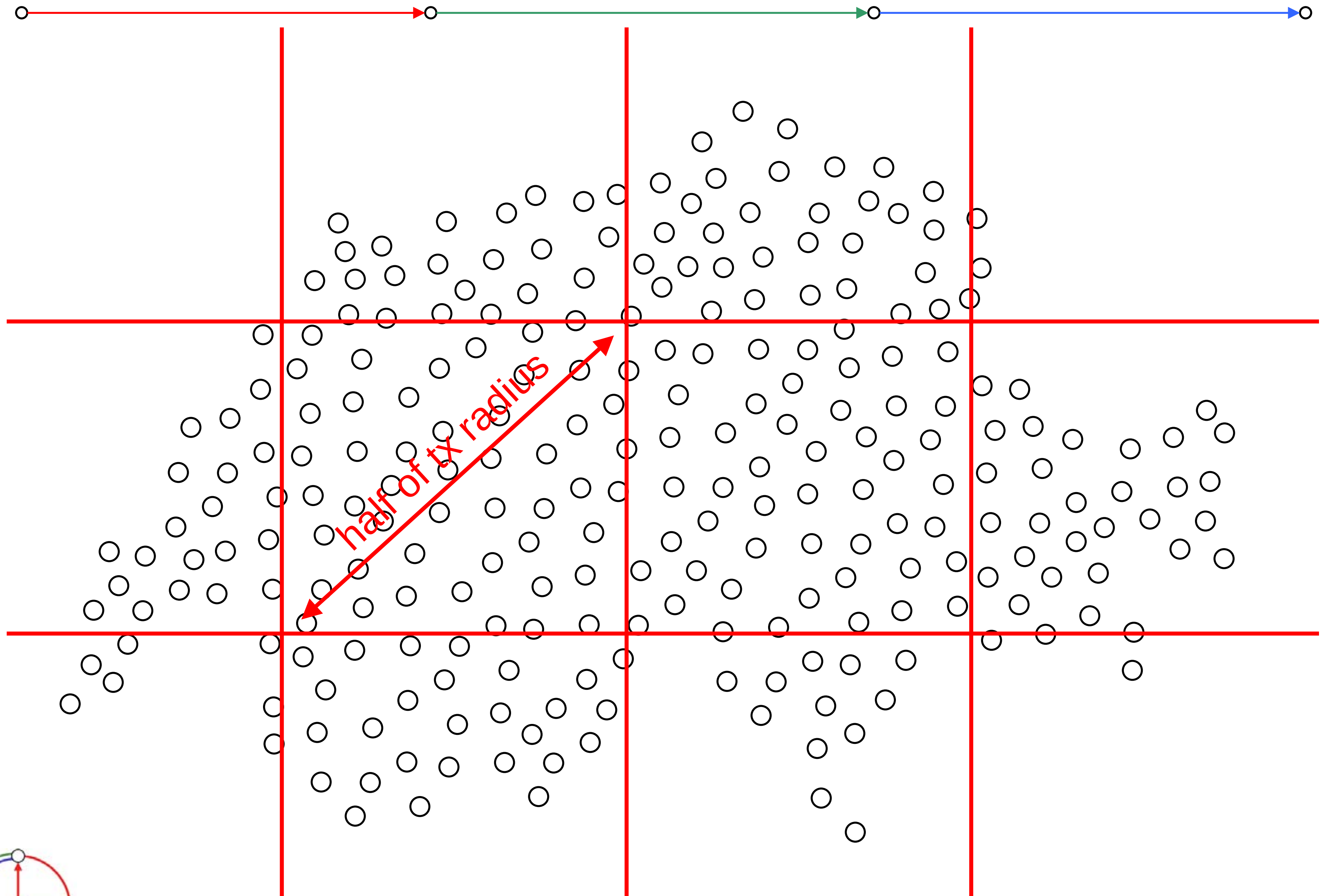
A much better, faster, and simpler algorithm!

- Assume that nodes know their position (GPS)
- Assume that nodes are in the plane; two nodes are within their transmission radius if and only if their Euclidean distance is at most 1 (UDG, unit disk graph)





A much better, faster, and simpler algorithm!



Comparison

- First algorithm (distributed linear program)
- Algorithm computes DS
- $k^2+O(1)$ transmissions/node
- $O(\Delta^{O(1)/k} \log \Delta)$ approximation
- General graph
- No position information

- Second algorithm (virtual grid)
- Algorithm computes DS
- 1 transmission/node
- $O(1)$ approximation
- Unit disk graph (UDG)
- Position information (UDG)

The model determines the distributed complexity (i.e., locality) of clustering!

Models

General Graph

UDG No GPS

UDG GPS

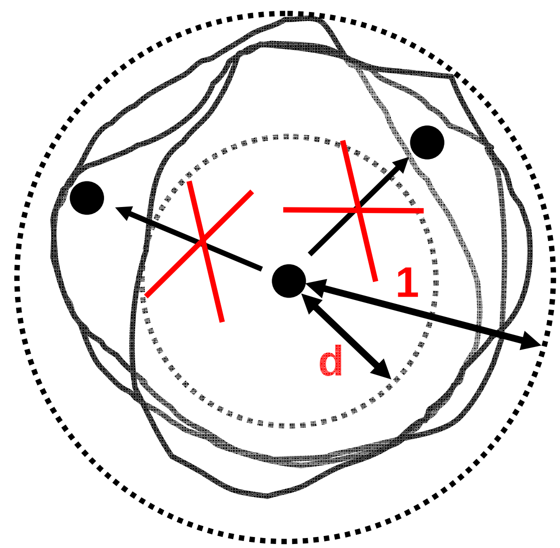
too pessimistic

too optimistic

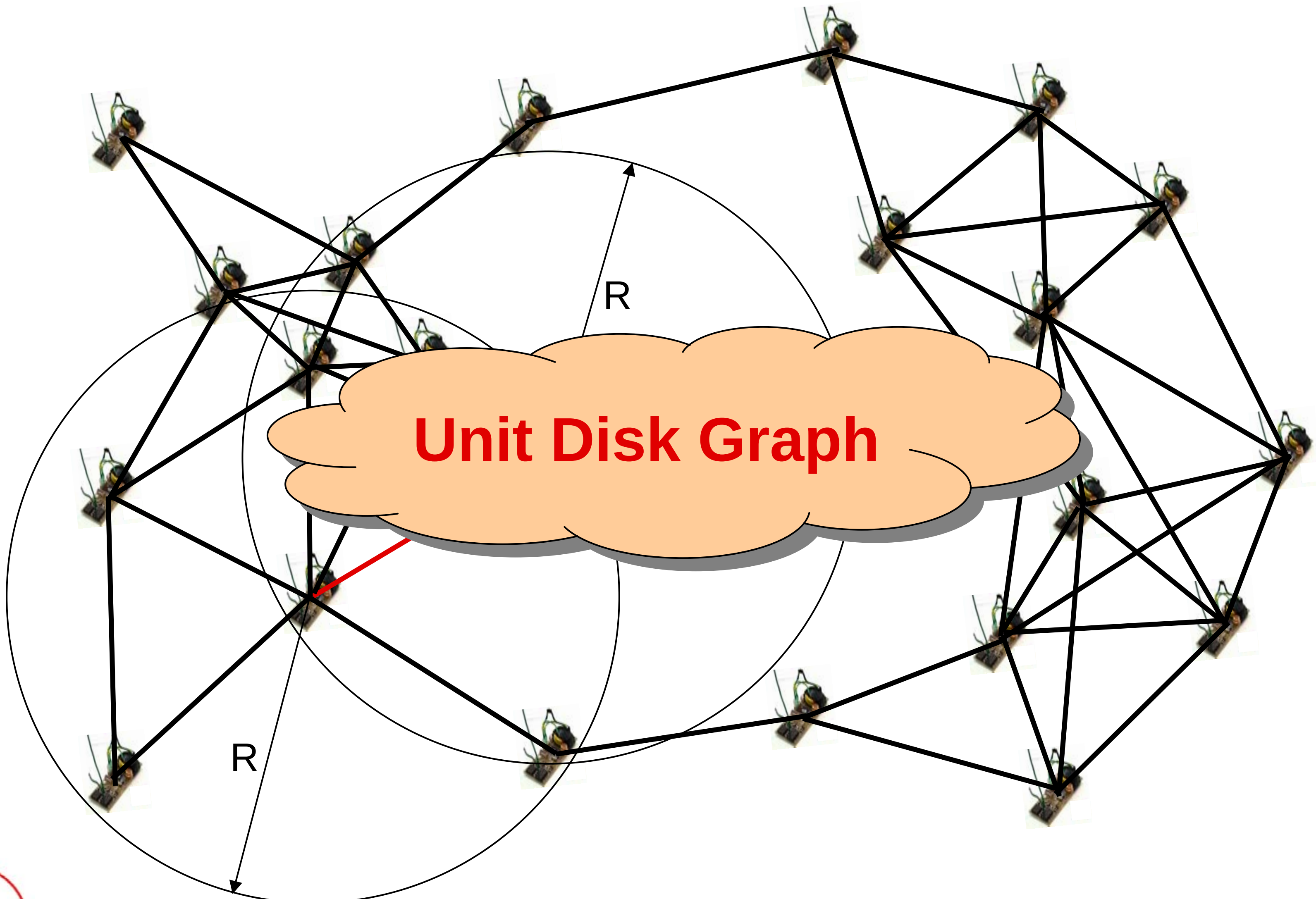
Bounded Independence

Unit Ball Graph

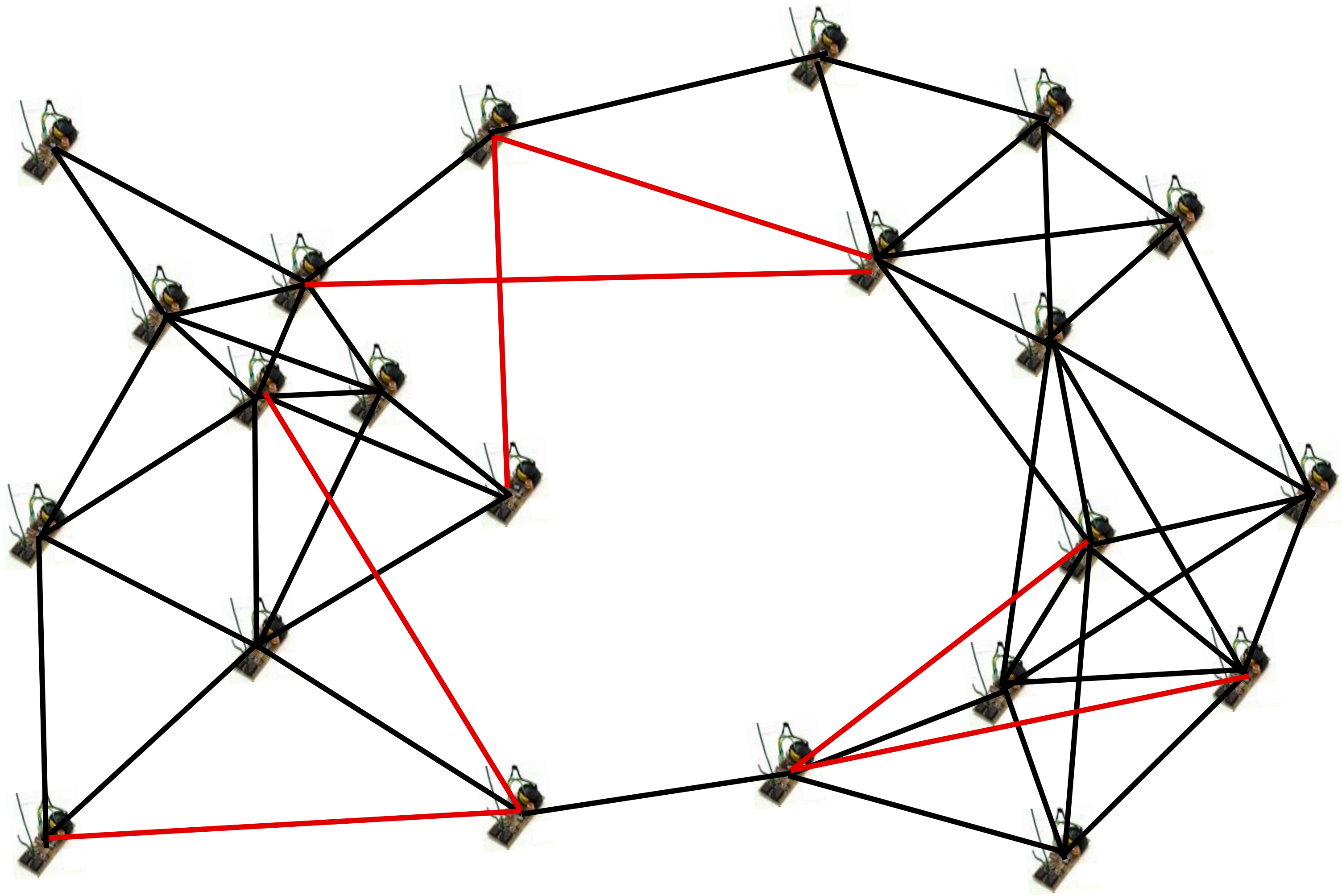
Quasi UDG



Locality in Real Networks



Locality in Real Networks



Locality in Real Networks

Wireless Networks are not unit disk graphs, but:

- No links between far-away nodes
- Close nodes tend to be connected
- In particular: Densely covered area → many connections

We want to understand the complexity distributed algorithms in real networks!

LOCALITY!

Unit Ball Graphs

- ∃ metric (V,d) describing distances between nodes u,v ∈ V

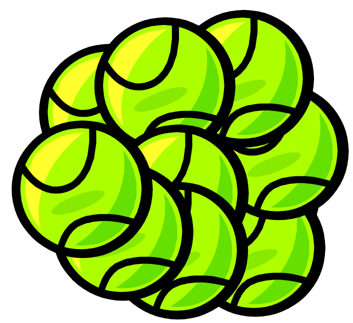
  such that: $d(u,v) \leq 1 : (u,v) \in E$
  $d(u,v) \geq 1 : (u,v) \notin E$

  Unit Ball Graph

- Assume that doubling dimension of metric is constant

- Doubling Dimension: log(#balls of radius r/2 to cover ball of radius r)

UBG based on underlying doubling metric.



Dominating Set Algorithm

d_{min} := min. distance between 2 nodes;
d := $2d_{min}$;
while ($d<1/2$) do
    G_d := graph induced by edges of length at most d;
    compute MIS S on G_d;
    only keep nodes of S;
    d := 2d
od

- Algorithm computes a dominating set:
  - Initially: all nodes are active
  - Distance to next active node incremented by $\leq$ d in each iteration
  - Sum of all d values is smaller than 1
- Constant approximation:
  - Distance between two dominators >1/4
  - Underlying metric is doubling

Dominating Set Algorithm

```
d_min := min. distance between 2 nodes;
d := 2d_min;
while (d<1/2) do
    G_d := graph induced by edges of length at most d;
    compute MIS S on G_d;
    only keep nodes of S;
    d := 2d
od
```

- Number of while loop iterations: $O(\log(1/d_{min}))$
- On doubling UBG: G_d has bounded degree
- Computing MIS S: $O(\log^* n)$ rounds → $O(\log^* n)$ time per iteration

Naive Implementation has time complexity of $O(\log^* n \log(1/d_{min}))$

Exploiting Locality

- Every k-round local algorithm can be transformed into the following canonical form:

  1. Collect complete k-neighborhood
  2. Compute solution locally be simulating relevant part of algorithm

- Using this transformation, we achieve: [KMW @ PODC 05]

For MIS, this is tight! (Due to $\Omega(\log^* n)$ lower bound on ring by Linial)

Time Complexity: O(log*n)

Approximation Ratio: O(1)

Compare with much stronger lower bound on general graphs!

Models

General Graph

UDG No GPS

UDG GPS

too pessimistic — **too optimistic**

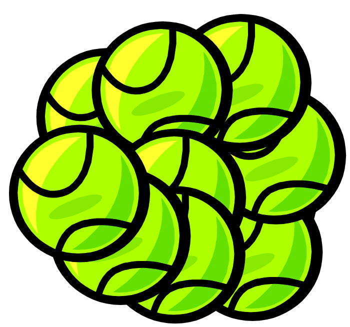
Bounded Independence

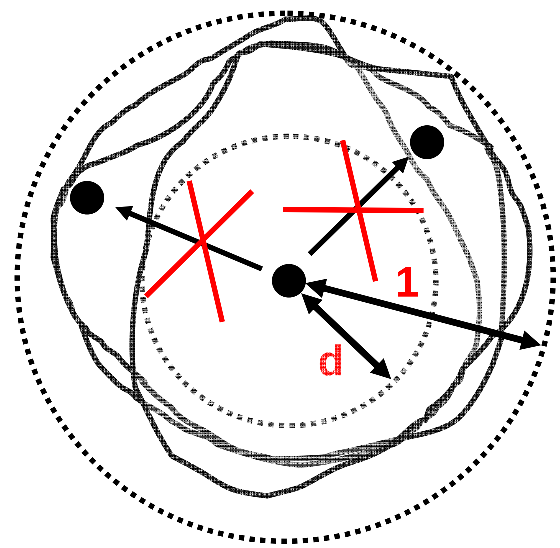
Unit Ball Graph

Quasi UDG

Number of independent neighbors is bounded (UDG: 5)

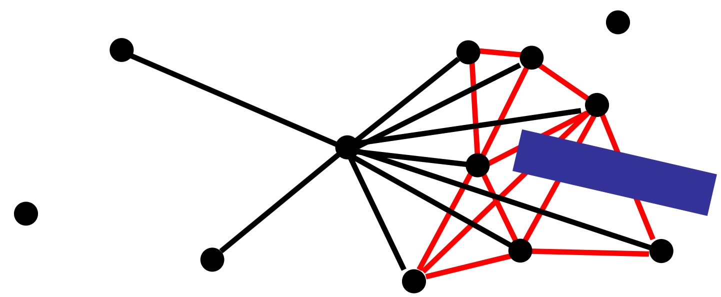
In a doubling metric:





Bounded Independence

- Def.: A graph *G* has bounded independence if there is a function *f(r)* such that every *r*-neighborhood in G contains at most *f(r)* independent nodes.
  - Note: *f(r)* does not depend on size of the graph !
  - Polynomially Bounded Independence: *f(r) = poly(r)*



1) A node can have many neighbors

2) But not all of them can be independent!

3) Can model obstacles, walls, ...

f(1) = 6

- Definition includes:
  - (Quasi) Unit Disk Graphs, Bounded Disk Graphs,...
  - Coverage Area Graphs, ...

Beyond Constant Approximation - Local PTAS

In graphs with bounded independence

An $(1+\varepsilon)$-approximation can be computed

in time $O(T_{MIS}+\log^{*}n/\varepsilon^{O(1)})$

[Kuhn, Moscibroda, Nieberg, Wattenhofer @ DIALM 05]

- $T_{MIS} \in O(\log \Delta \cdot \log^{*}n)$
  → in all graphs with bounded independence!

- $T_{MIS} \in O(\log^{*}n)$
  → in UBG with underlying doubling metric!
  → if nodes have distance information!

Deterministic Distributed MIS

Is there a distributed, deterministic MIS algorithm for general graphs?

- One of the outstanding questions in distributed computing theory [Linial 92]

- Partial affirmative answer:

KMNW @ DISC 2005
Talk: Wednesday 11:25 !!!

In graphs with polynomially bounded independence, we have a distributed deterministic $O(\log \Delta \cdot \log^* n)$ time MIS algorithm.

Models

General Graph

UDG No GPS

UDG GPS

too pessimistic — **too optimistic**

Bounded Independence

Unit Ball Graph

Quasi UDG

Physical Signal Propagation

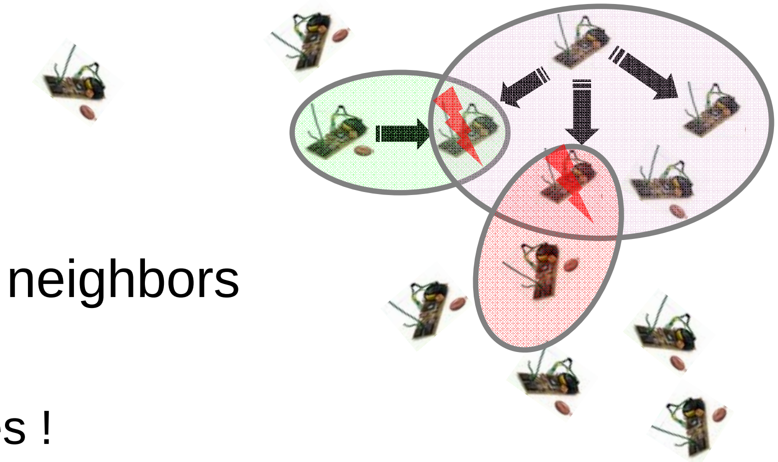
Radio Network Model

Message Passing Models

too realistic — **too simplistic**

Unstructured Radio Network Model [KMW, Mobicom 04]

Unstructured Radio Network Model



- Multi-Hop
- No collision detection
  - Not even at the sender!
- No knowledge about (the number of) neighbors
- Asynchronous Wake-Up
  - Nodes are not woken up by messages !
- Unit Disk Graph (UDG) to model wireless multi-hop network
  - Two nodes can communicate iff Euclidean distance is at most 1
- Upper bound n for number of nodes in network is known
  - This is necessary due to $\Omega(n / \log n)$ lower bound [Jurdzinski, Stachowiak, ISAAC 2002]

Unstructured Radio Network Model

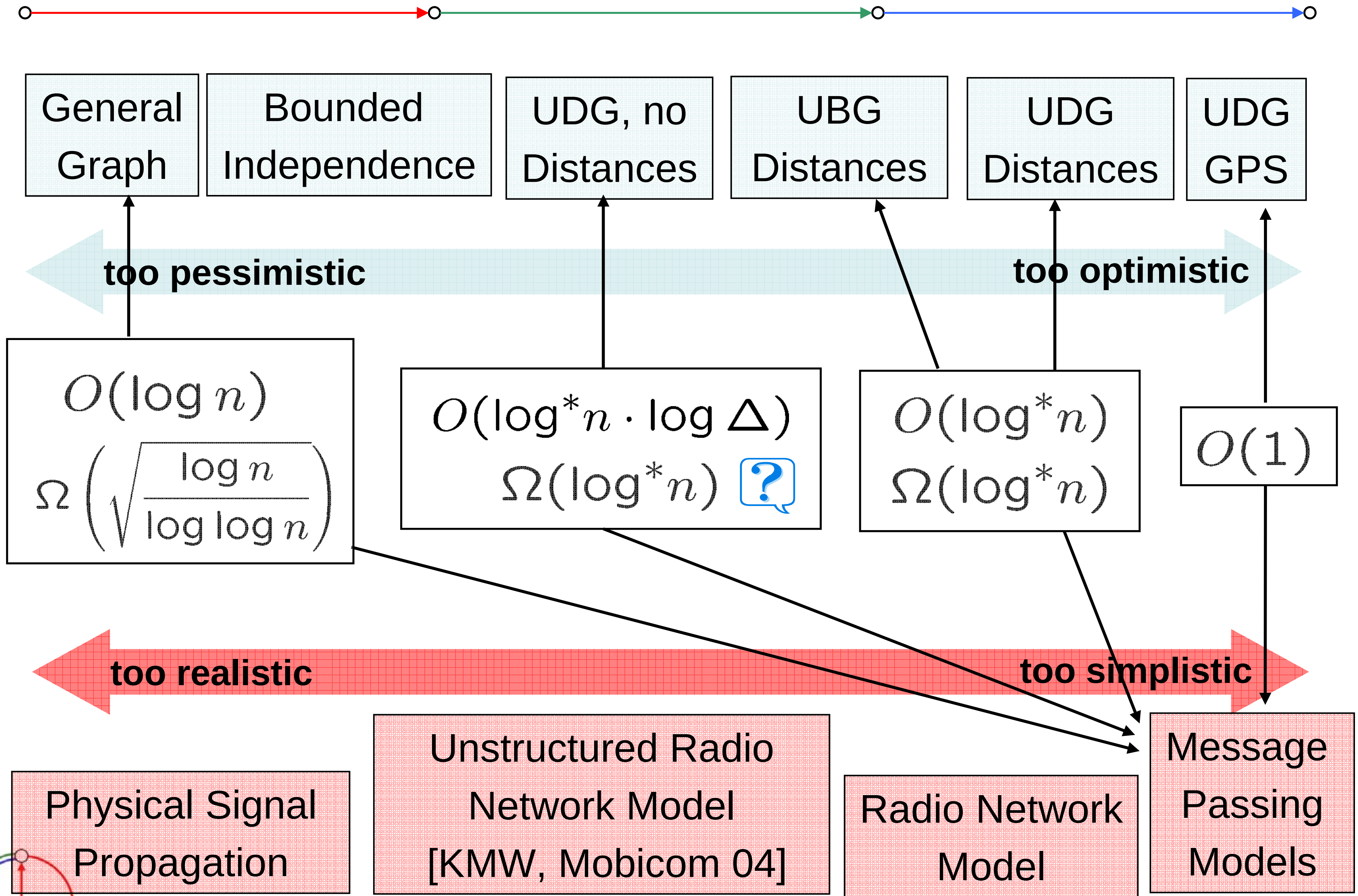
- Can MDS and MIS be solved efficiently in such a harsh model? [Moscibroda, Wattenhofer @ PODC 2005]

There is a MIS algorithm with running time $O(\log^2 n)$ with high probability.

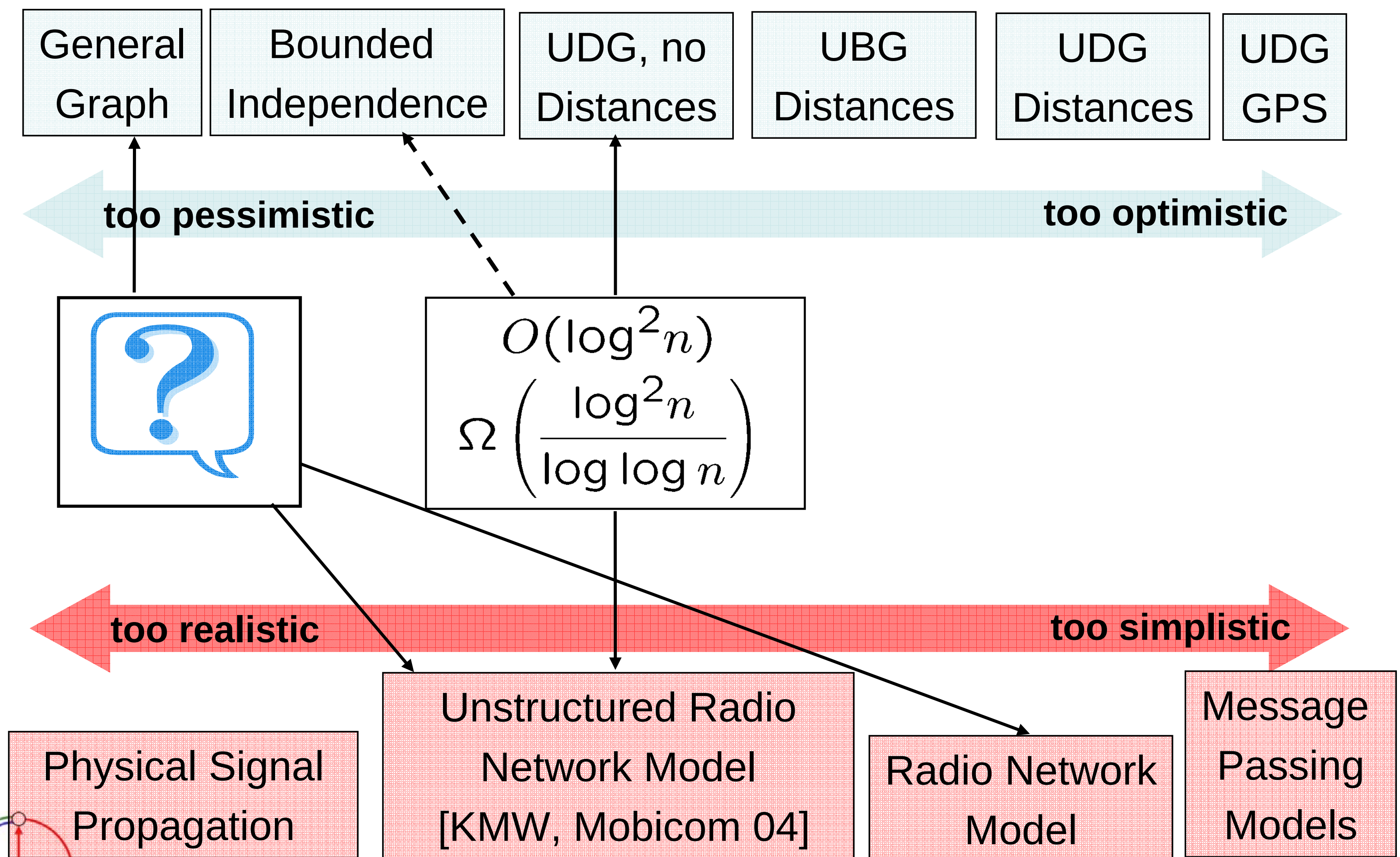
Optimal up to O(loglog n) factor

Compare with O(log n) or $O(\log^* n)$ in message passing model!

Summary (MIS)


General Graph
Bounded Independence
UDG, no Distances
UBG Distances
UDG Distances
UDG GPS
too pessimistic
too optimistic
$O(\log n)$
$\Omega\left(\sqrt{\frac{\log n}{\log\log n}}\right)$
$O(\log^* n \cdot \log \Delta)$
$\Omega(\log^* n)$
?
$O(\log^* n)$
$\Omega(\log^* n)$
$O(1)$
too realistic
too simplistic
Physical Signal Propagation
Unstructured Radio Network Model [KMW, Mobicom 04]
Radio Network Model
Message Passing Models

Summary (MIS)


General Graph
Bounded Independence
UDG, no Distances
UBG Distances
UDG Distances
UDG GPS
too pessimistic
too optimistic
$O(\log^2 n)$
$\Omega\left(\frac{\log^2 n}{\log\log n}\right)$
too realistic
too simplistic
Physical Signal Propagation
Unstructured Radio Network Model [KMW, Mobicom 04]
Radio Network Model
Message Passing Models

Theory of Locality

Locality is crucial in distributed computing!

- What can be computed locally?

Locally solvable problems!	Problems in the middle!	Locally unsolvable problems!
Count neighbors	MIS, MDS	MST
	Coloring	Count number of nodes

- Theory of Locality:
  - Key for designing fast algorithms
  - Allows a classification of problems!
  - Allows a classification of computational models!

$\Theta(\log^* n)$: **Ring, UDG w. distances**, **UBG w. distances**

$\Omega\left(\sqrt{\frac{\log n}{\log\log n}}\right)$: **General Graphs**

Questions? Comments?

Distributed
Computing
Group

Questions?
Comments?

Thomas Moscibroda