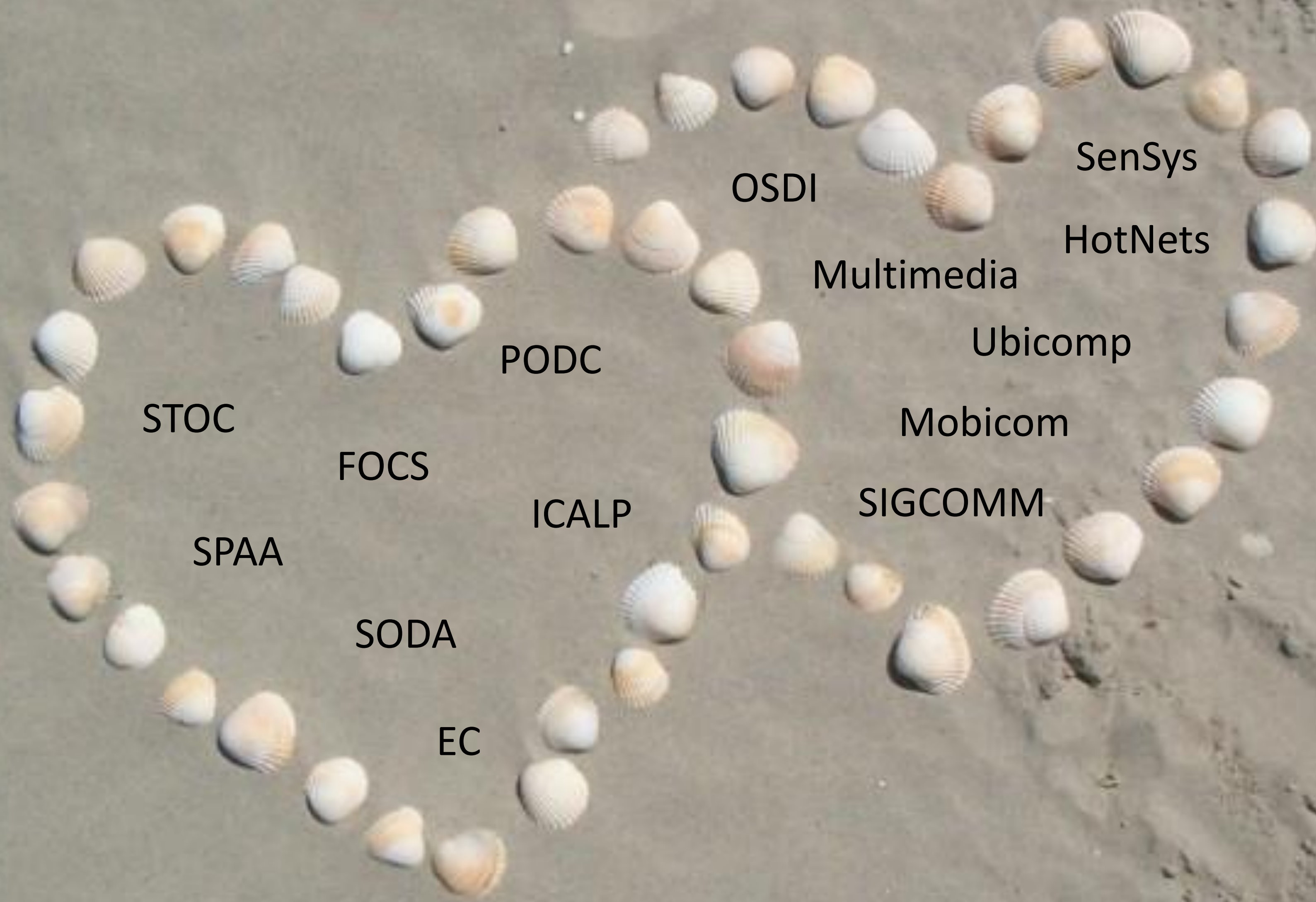
Sensor Networks
Where Theory Meets Practice

Roger Wattenhofer

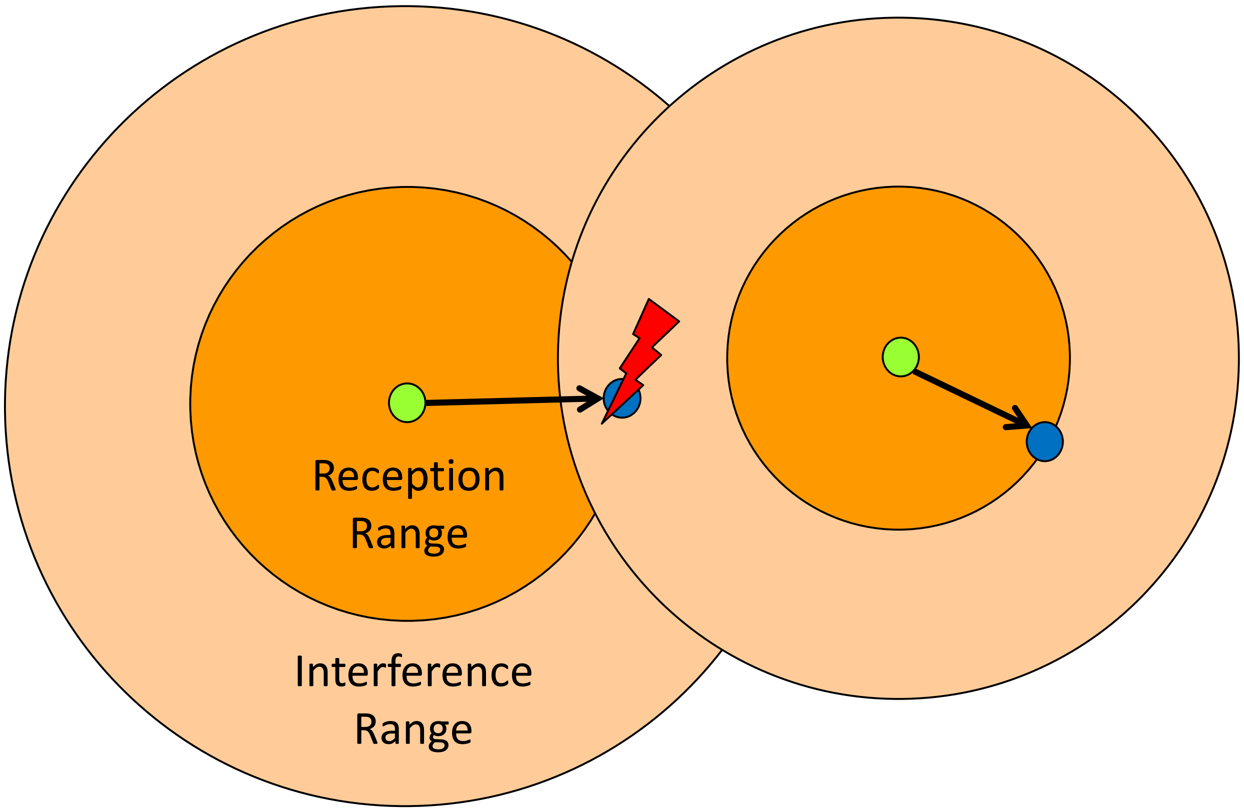
Theory Meets Practice



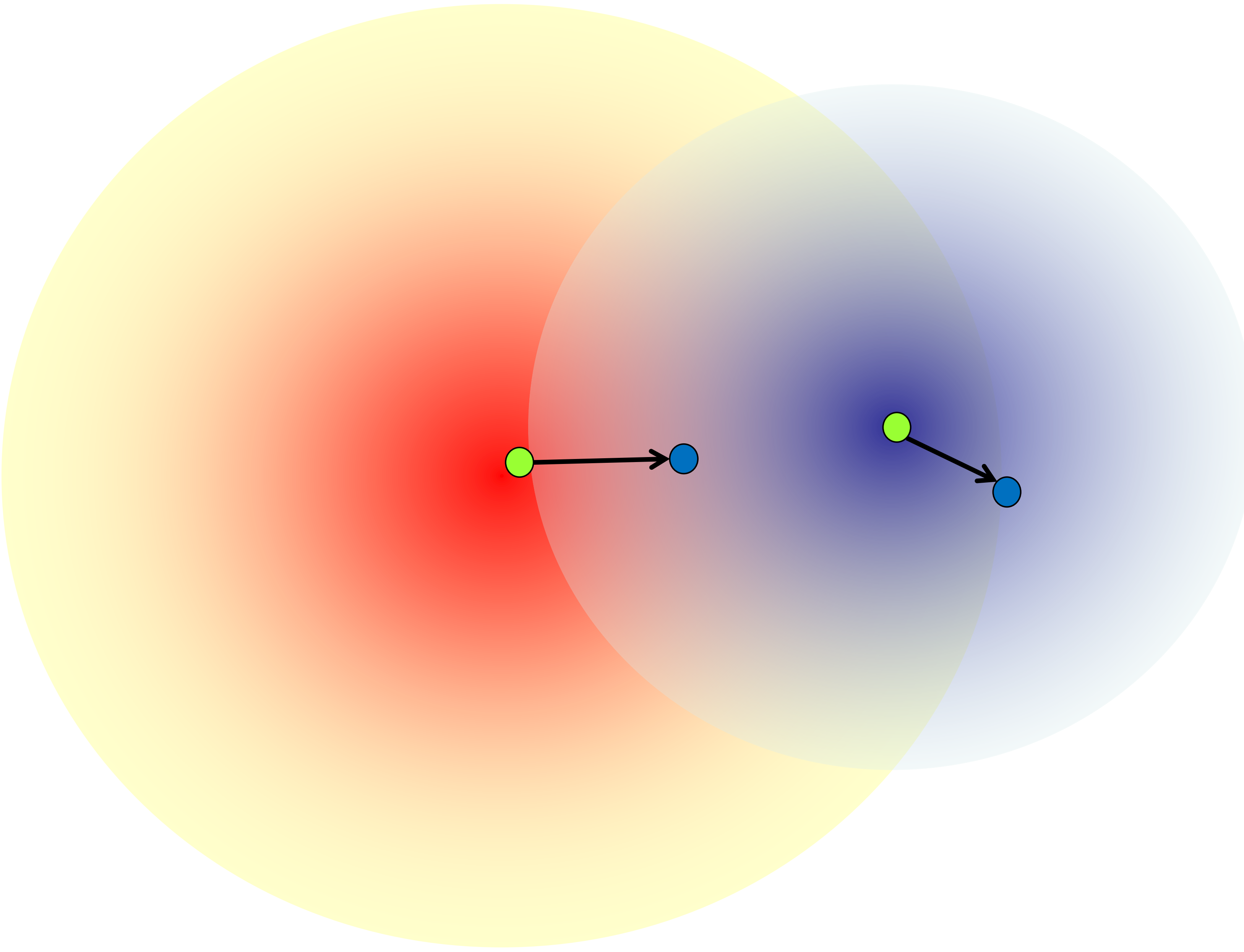
Wireless Communication?

Capacity!

Protocol Model



Physical (SINR) Model



Signal-To-Interference-Plus-Noise Ratio (SINR) Formula

$$\frac{\frac{P_u}{d(u,v)^\alpha}}{N + \sum_{w \in V \setminus \{u\}} \frac{P_w}{d(w,v)^\alpha}} \geq \beta$$

Received signal power from sender

Power level of sender *u*

Path-loss exponent

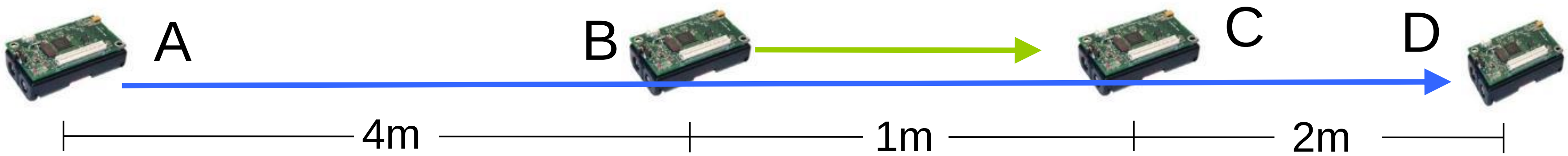
Noise

Minimum signal-to-interference ratio

Received signal power from all other nodes (=interference)

Distance between two nodes

Example: Protocol vs. Physical Model



Assume a single frequency (and no fancy decoding techniques!)

Is spatial reuse possible?
- NO — Protocol Model
- YES — With power control

Let α=3, β=3, and N=10nW

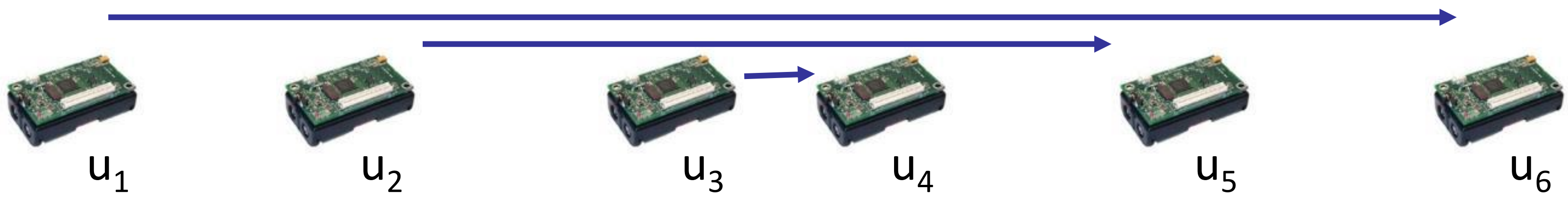
Transmission powers: P_B= -15 dBm and P_A= 1 dBm

SINR of A at D: $$\frac{1.26mW/(7m)^3}{0.01\mu W+31.6\mu W/(3m)^3}\approx 3.11\geq\beta$$ 

SINR of B at C: $$\frac{31.6\mu W/(1m)^3}{0.01\mu W+1.26mW/(5m)^3}\approx 3.13\geq\beta$$ 

This works in practice

… even with very simple hardware



Time for transmitting 20‘000 packets:

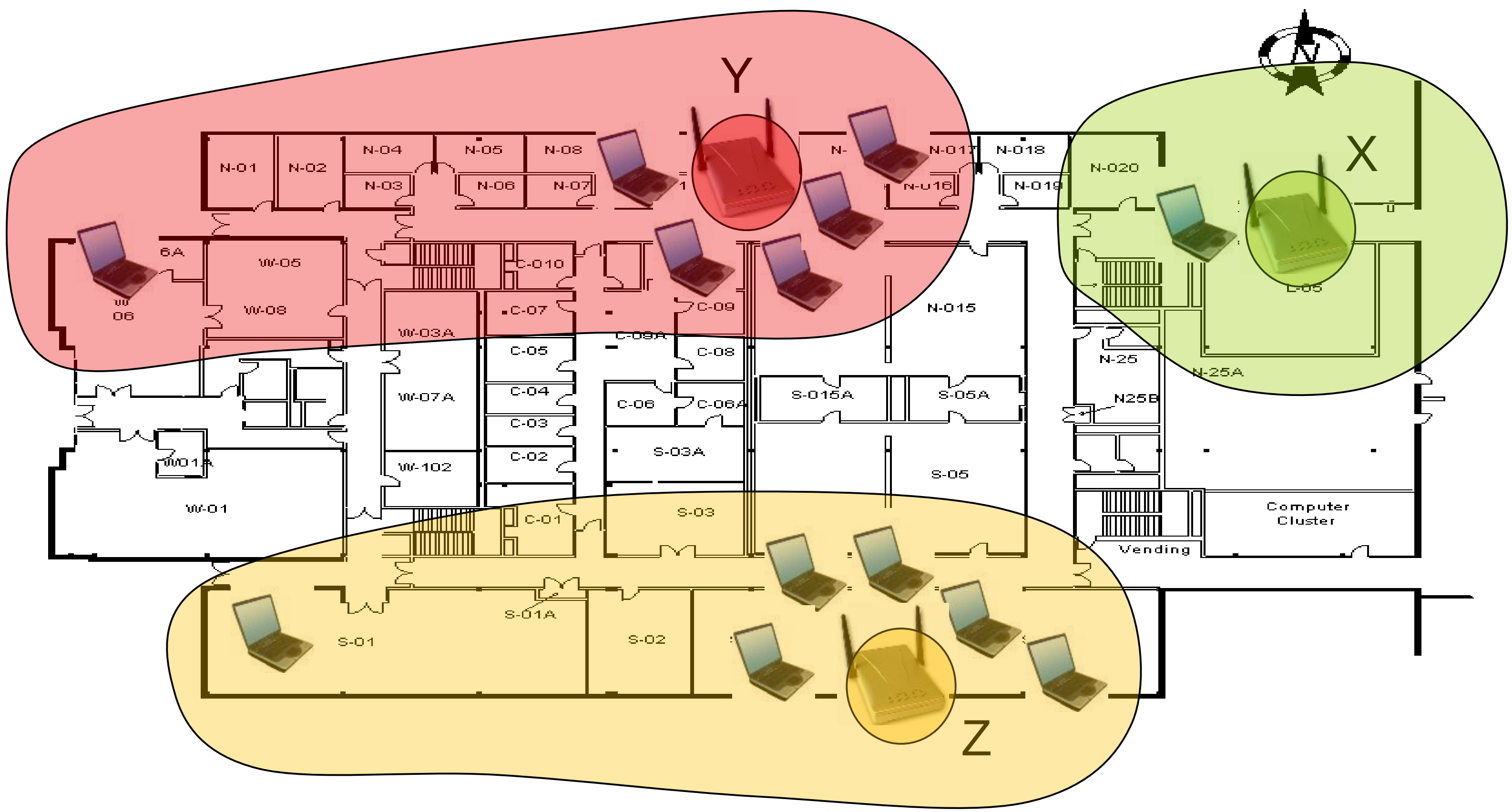
	Time required	
	standard MAC	“SINR-MAC”
Node u_1	$721s$	$267s$
Node u_2	$778s$	$268s$
Node u_3	$780s$	$270s$

	Messages received	
	standard MAC	“SINR-MAC”
Node u_4	19999	19773
Node u_5	18784	18488
Node u_6	16519	19498

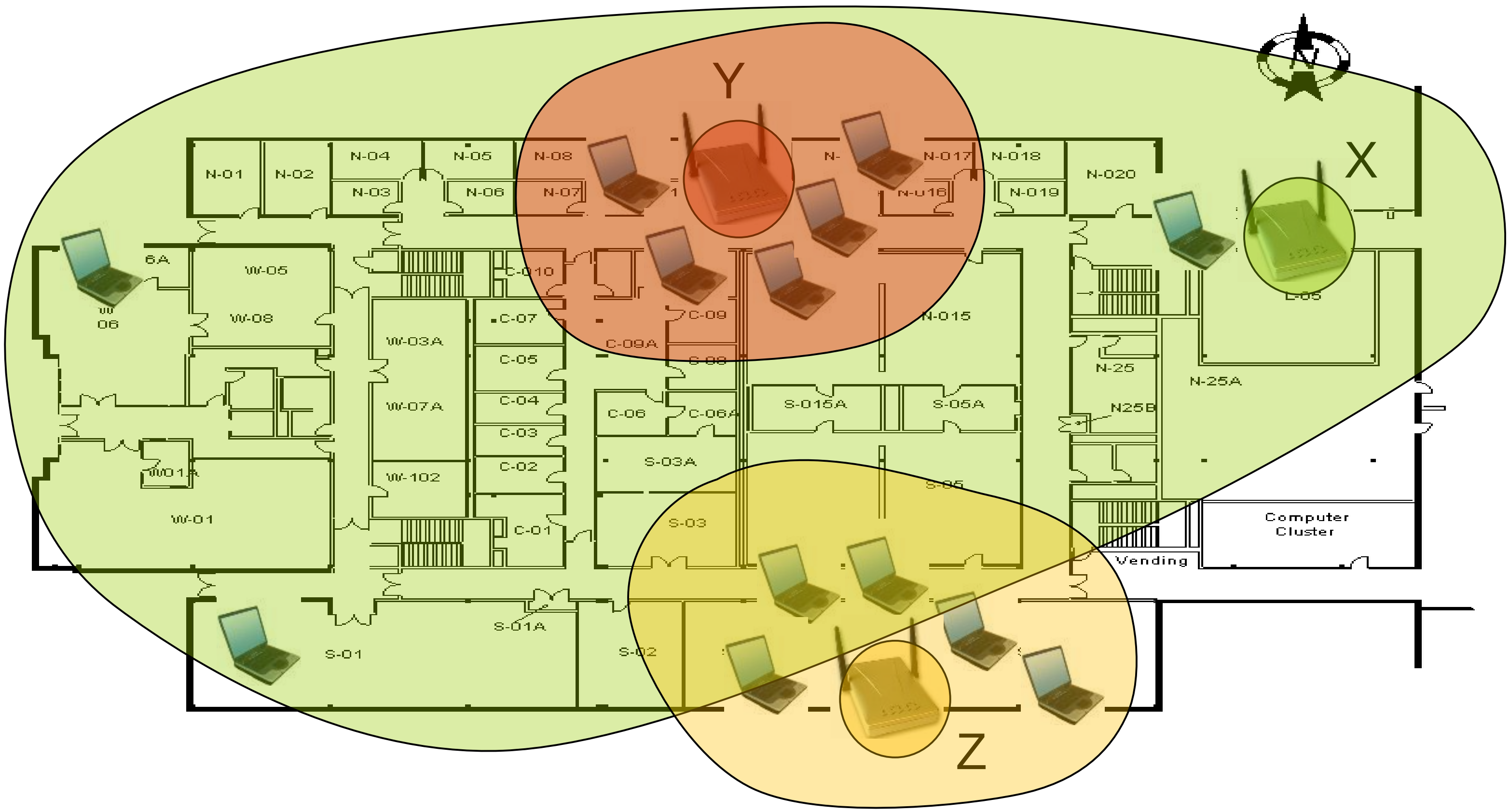
Speed-up is almost a factor 3

[Moscibroda, W, Weber, Hotnets 2006]

Possible Application – Hotspots in WLAN



Possible Application – Hotspots in WLAN



The Capacity of a Network

(How many concurrent wireless transmissions can you have)

Convergecast Capacity in Wireless Sensor Networks

[Moscibroda, W, 2006]

Worst-Case Capacity

[Giridhar, Kumar, 2005]

Classic Capacity

Topology / Model/Power	Max. rate in arbitrary, worst-case deployment	Max. rate in random, uniform deployment
Protocol Model	$\Theta(1/n)$	$\Theta(1/\log n)$
Physical Model (power control)	$\Omega(1/\log^3 n)$	$\Omega(1/\log n)$

Capacity of a Network

Real Capacity

How much information can be transmitted in any network?

Classic Capacity

How much information can be transmitted in nice networks?

Worst-Case Capacity

How much information can be transmitted in nasty networks?

Core Capacity Problems

Given a set of arbitrary communication links

One-Shot Problem

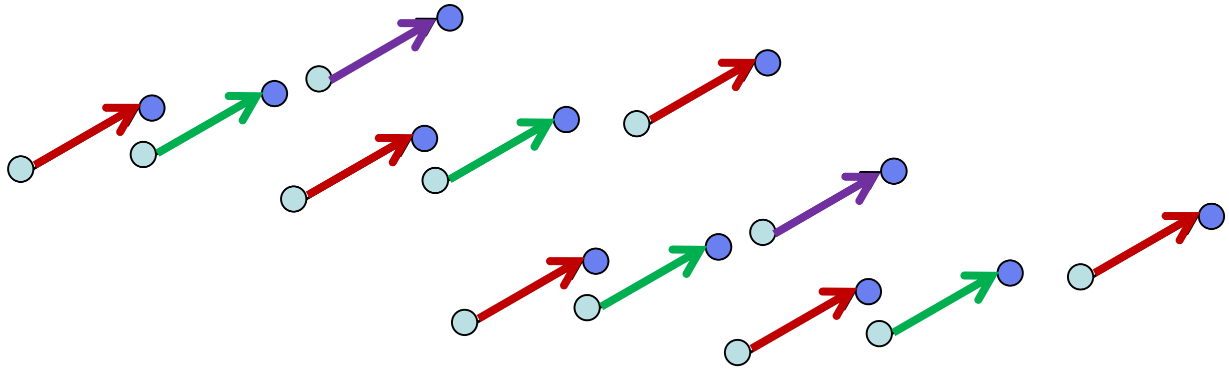
Find the maximum size feasible subset of links

O(1) approximations for uniform power [Goussevskaia, Halldorsson, W, 2009 & 2014] as well as arbitrary power [Kesselheim, 2011]

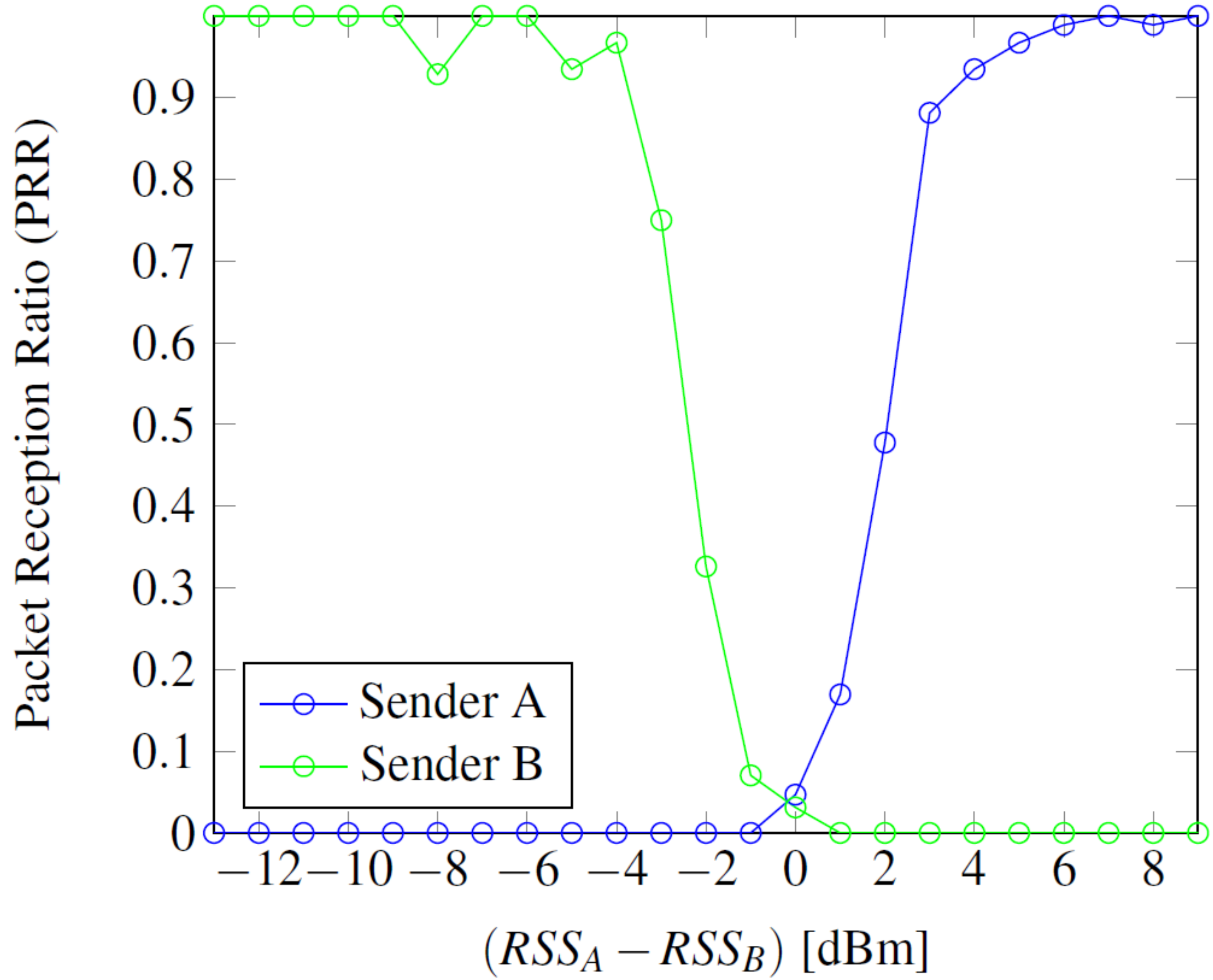
Scheduling Problem

Partition the links into fewest possible slots, to minimize time

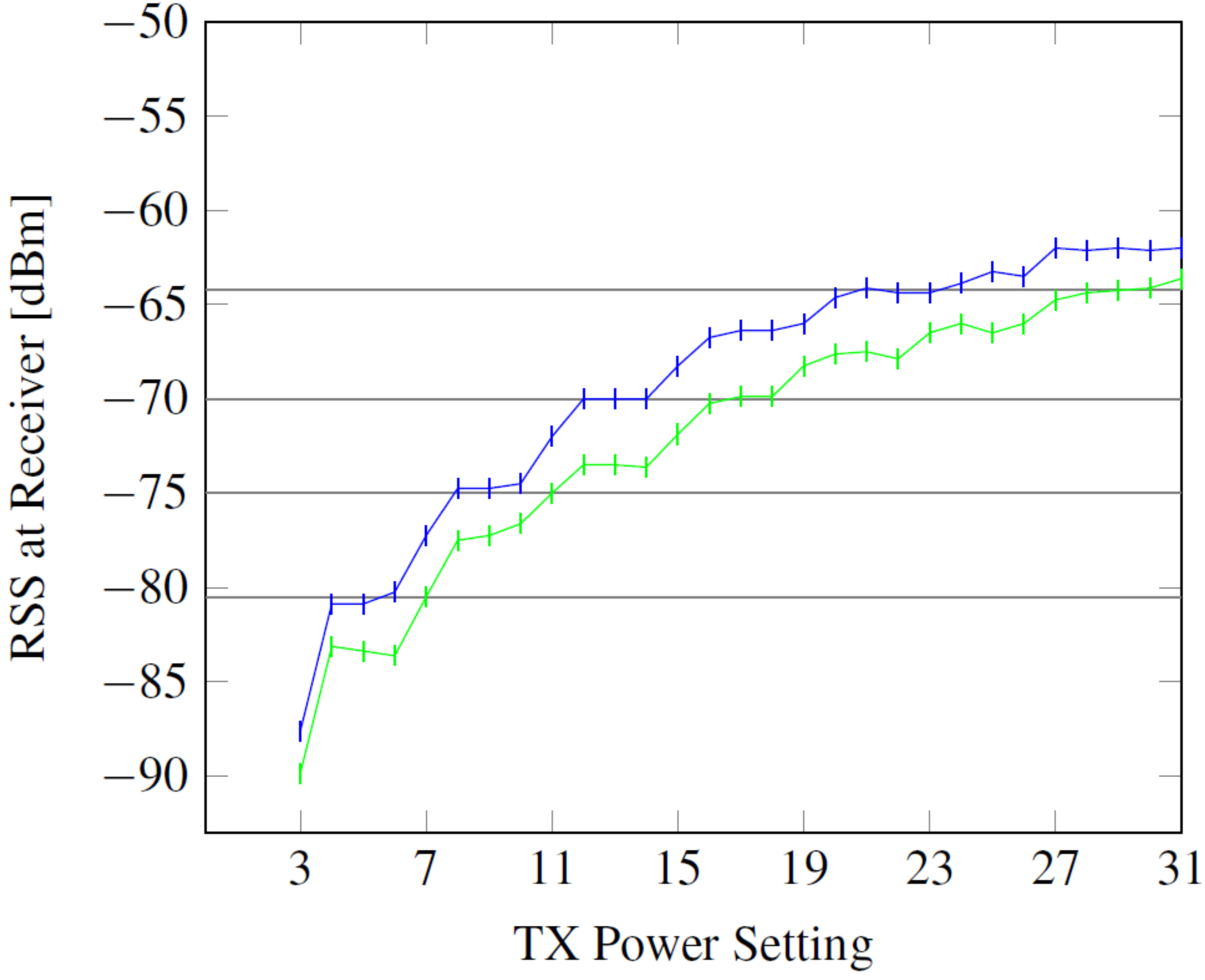
Open problem: Only $O(\log n)$ approximation using the one-shot subroutine



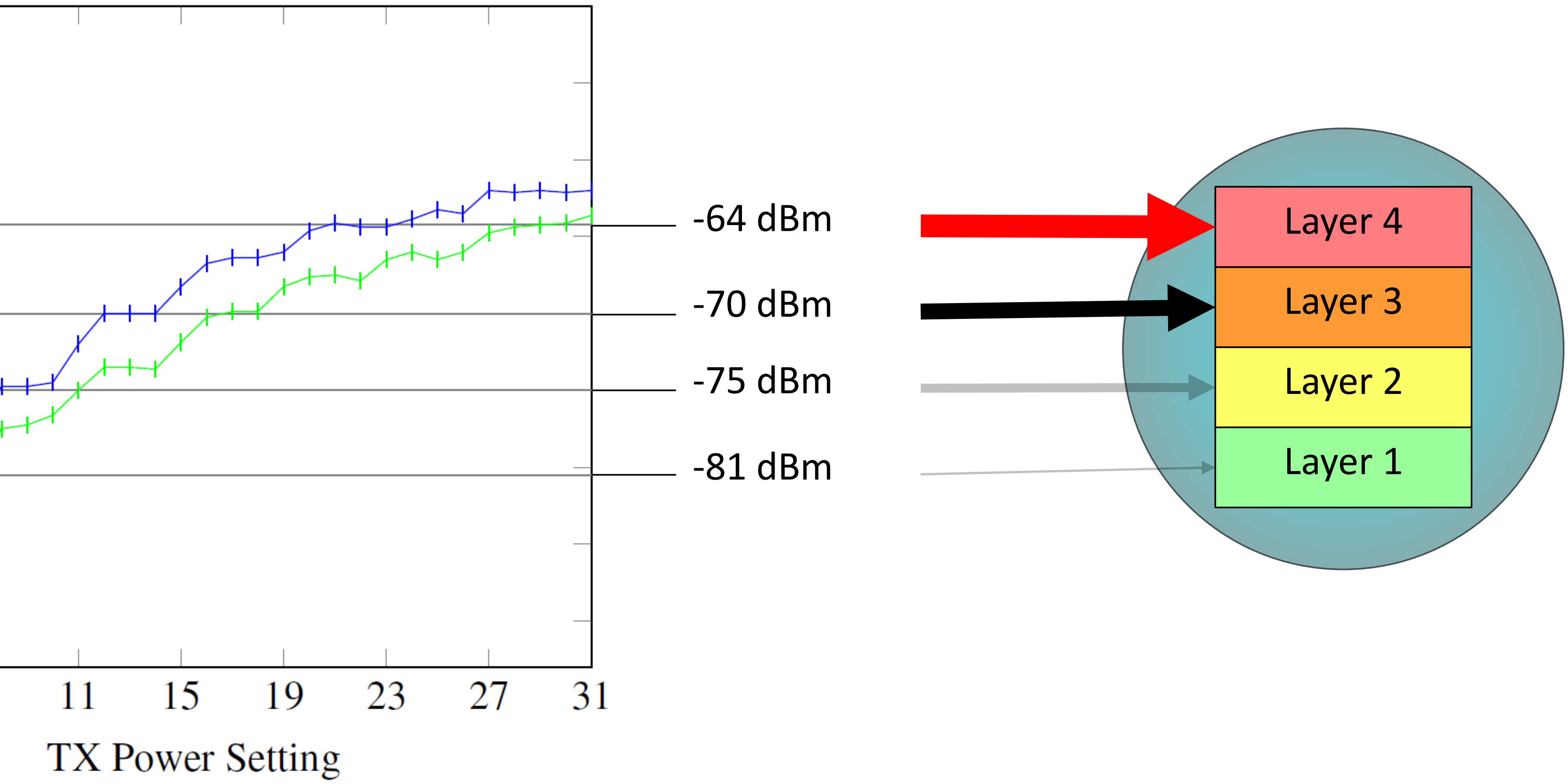
The Capture Effect



Receiving Different Senders

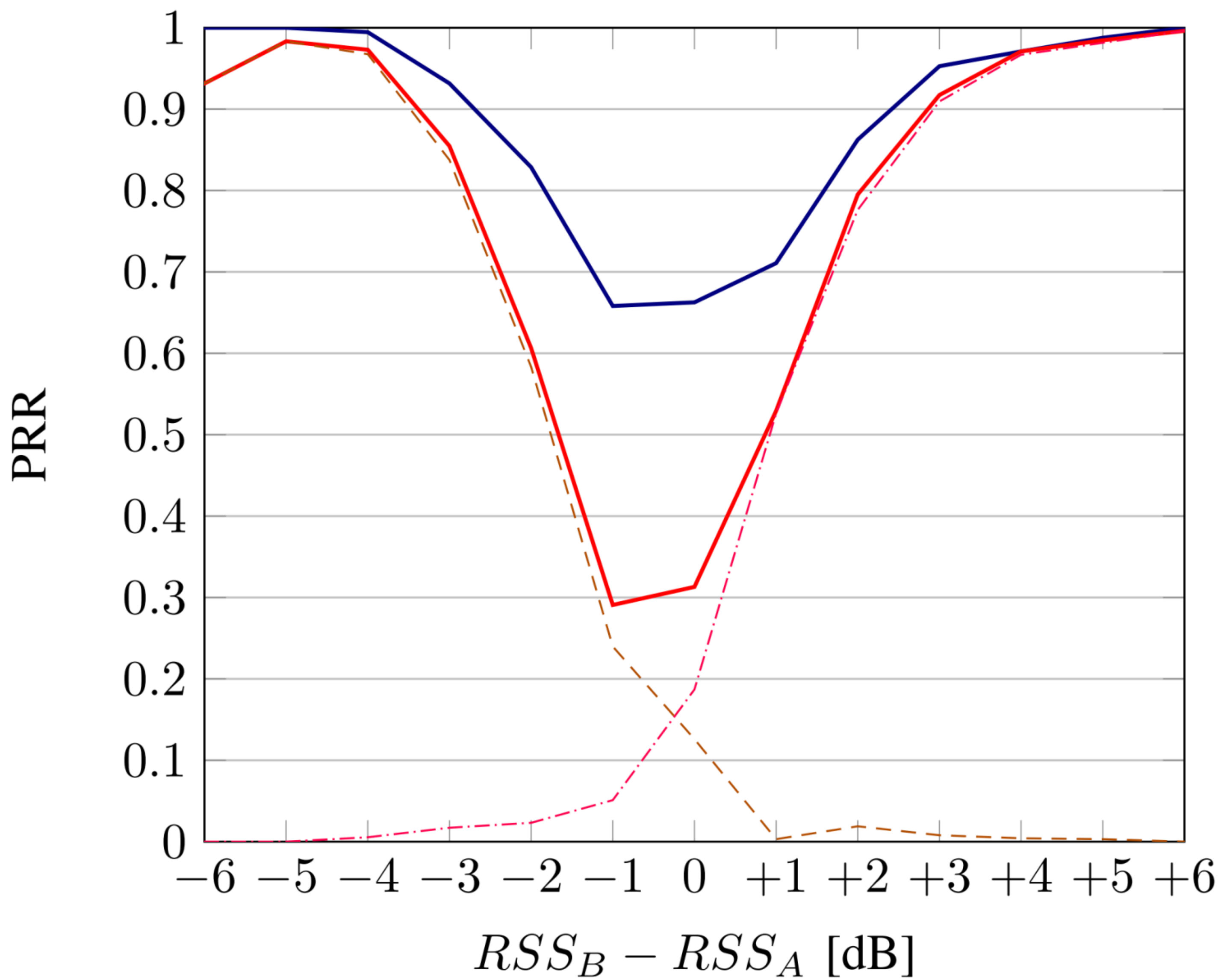


“Layer” Abstraction



[König, W, 2016]

Constructive Interference



Energy Efficiency?

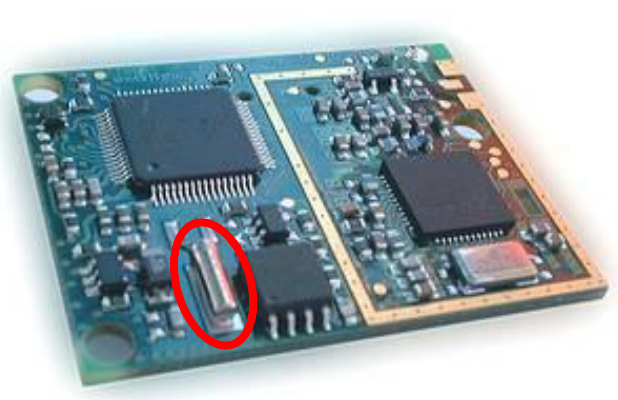
Clock Synchronization!

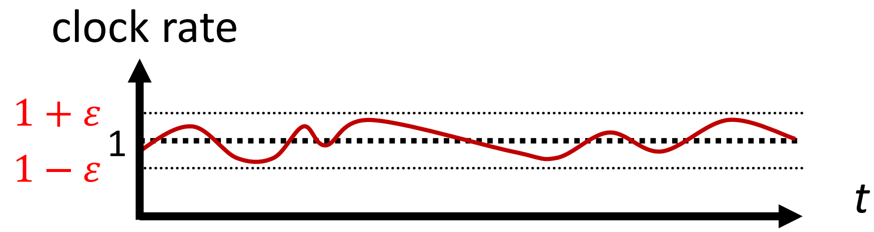
Clock Synchronization Example: Dozer

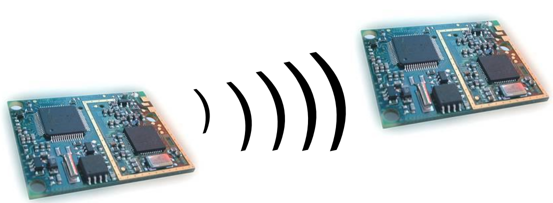
- Multi-hop sensor network with duty cycling
- 10 years of network life-time, mean energy consumption: 0.066mW
- High availability, reliability (99.999%)
- Many different applications use Dozer: TinyNode, PermaSense, etc.

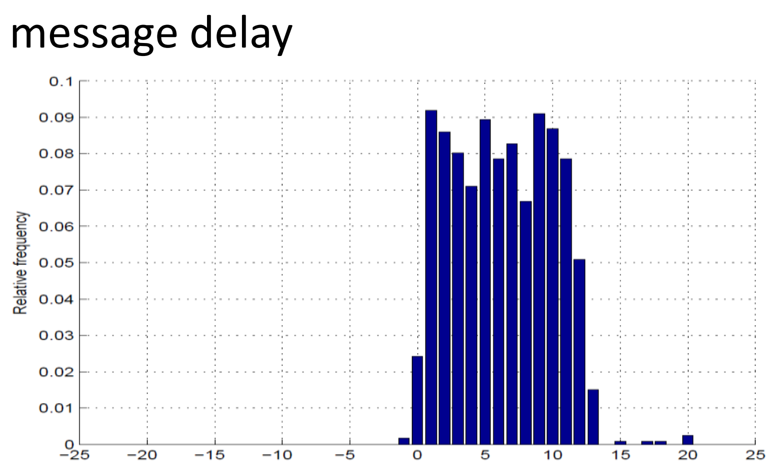
[Burri, von Rickenbach, W, 2007]

Problem: Physical Reality







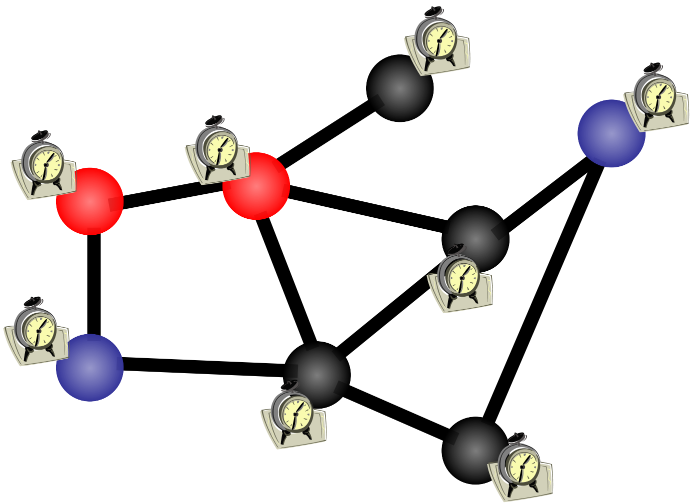


Clock Synchronization in Theory?

Given a communication network

1. Each node equipped with hardware clock with drift
2. Message delays with jitter

worst-case (but constant)



Goal: Synchronize Clocks ("Logical Clocks")

- Both global and local synchronization!

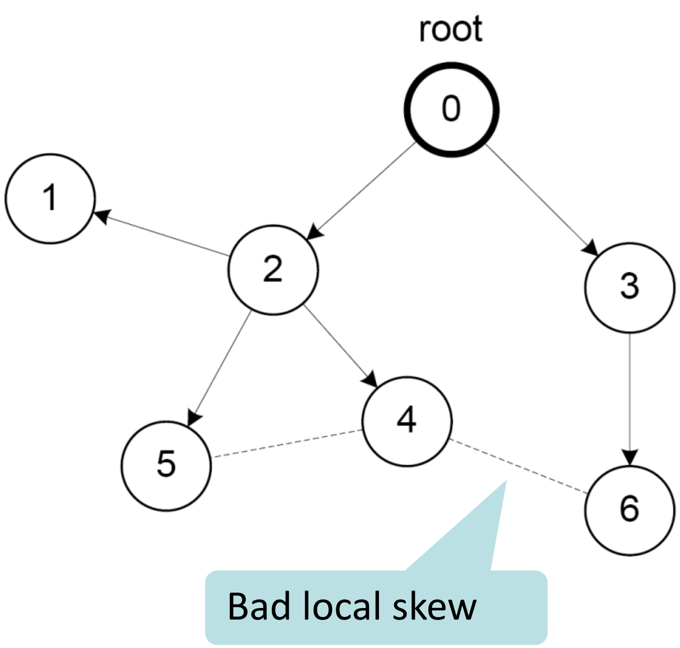
Time Must Behave!

- Time (logical clocks) should not be allowed to stand still or jump

Local Skew

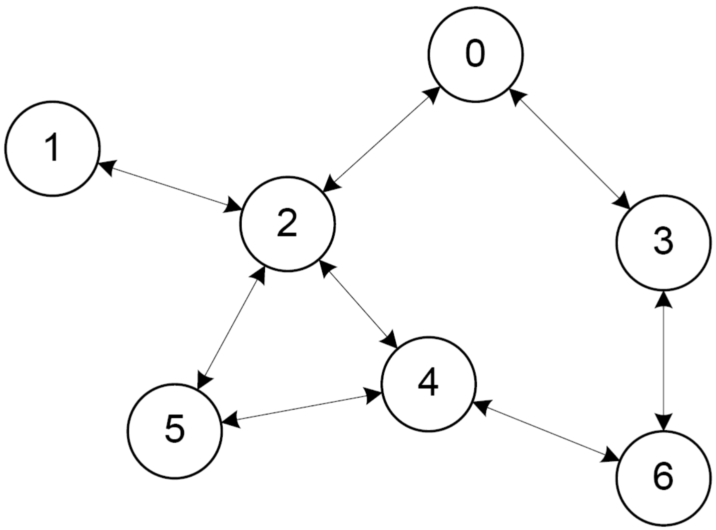
Tree-based Algorithms

e.g. FTSP



Neighborhood Algorithms

e.g. GTSP

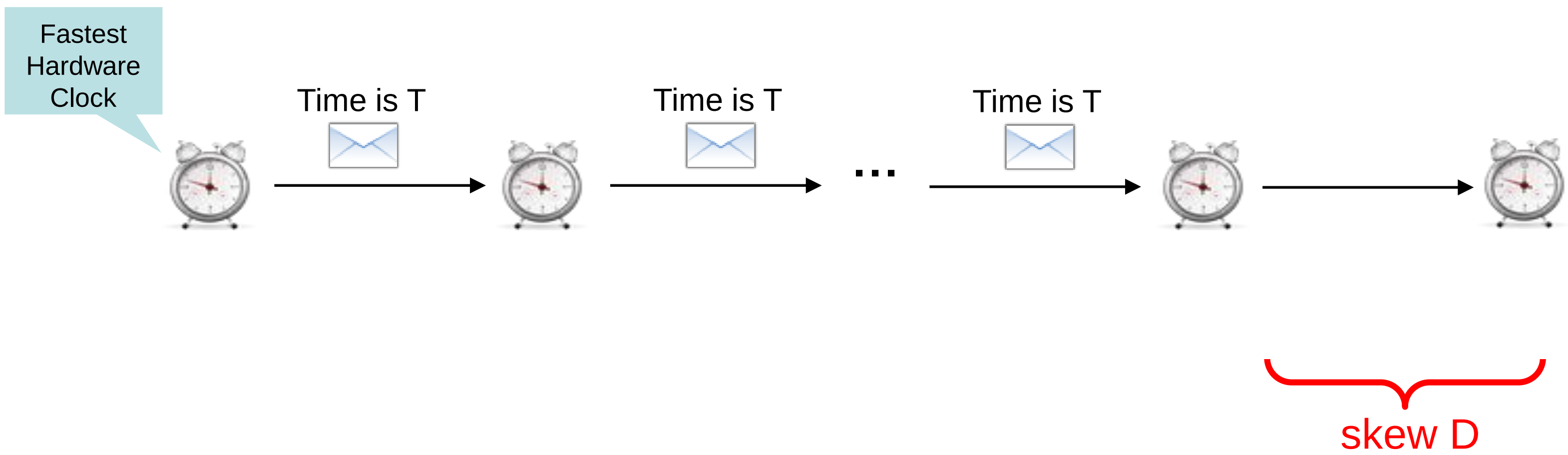


Synchronization Algorithms: An Example ("A^{max}")

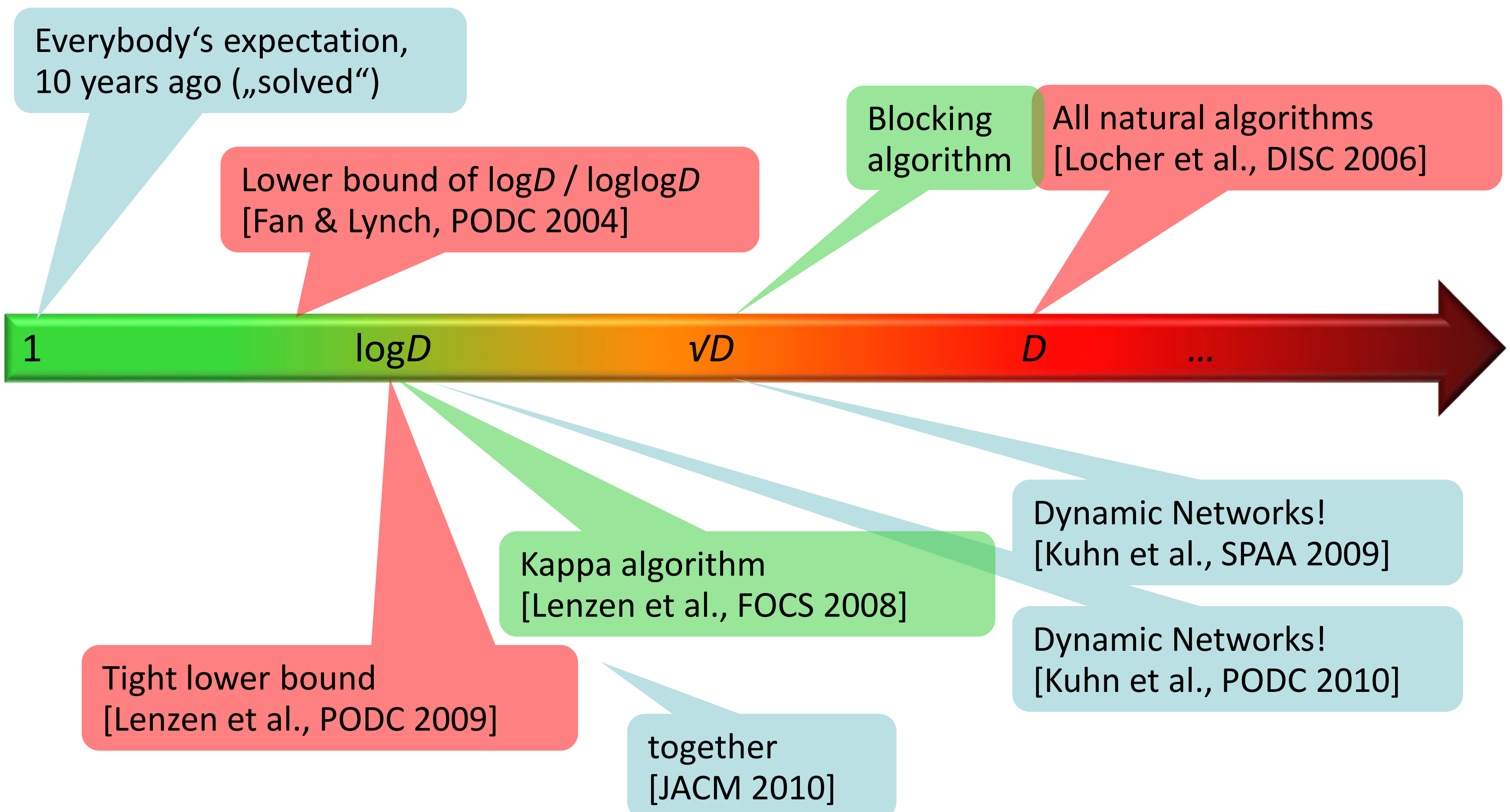
- Question: How to update the logical clock based on the messages from the neighbors?

- Idea: Minimizing the skew to the fastest neighbor
  - Set clock to maximum clock value you know, forward new values immediately

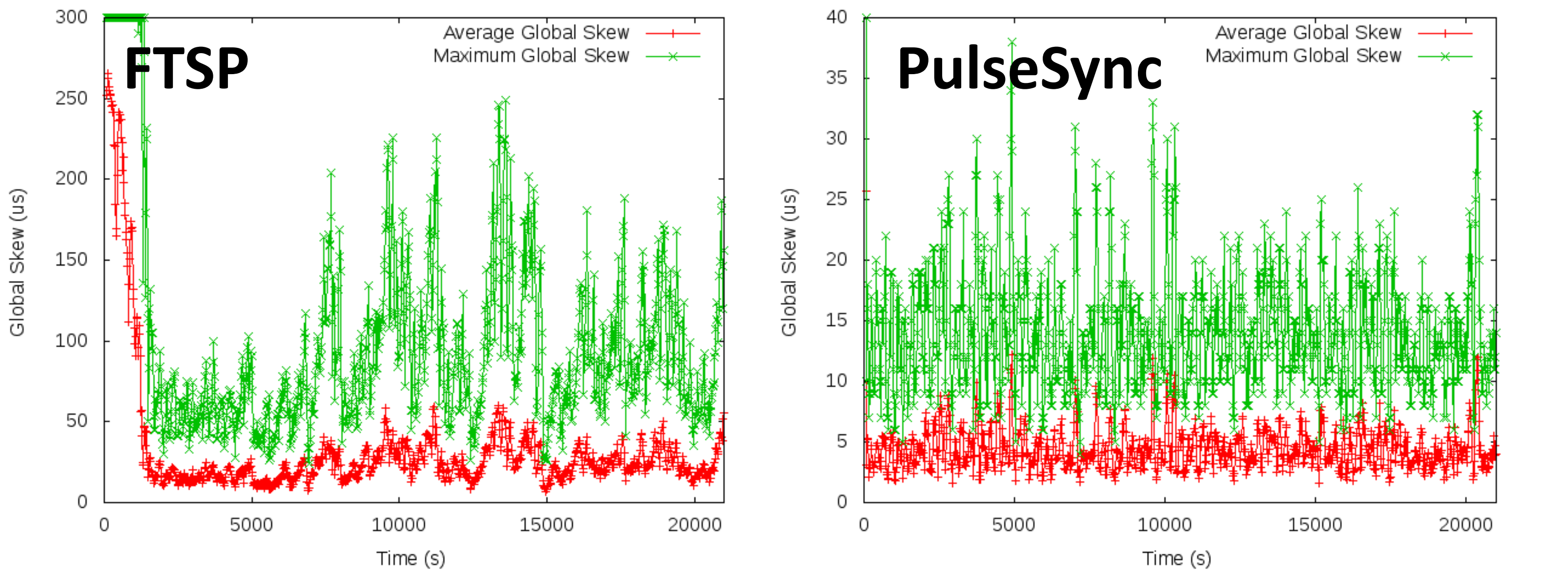
- First all messages are slow (1), then suddenly all messages are fast (0)!



Local Skew: Overview of Results

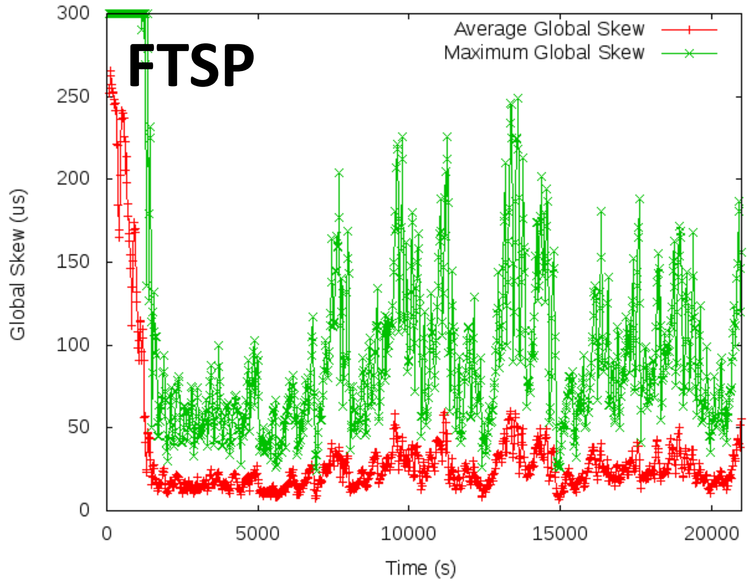


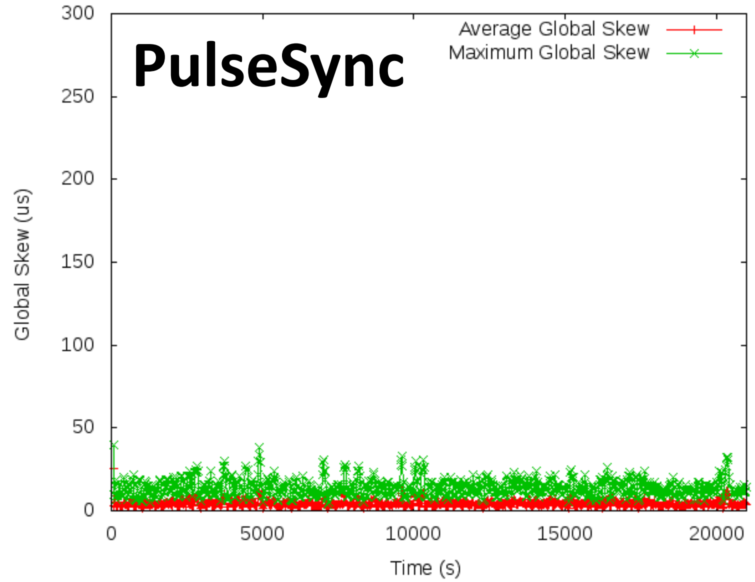
Experimental Results for Global Skew



[Lenzen, Sommer, W, 2014]

Experimental Results for Global Skew

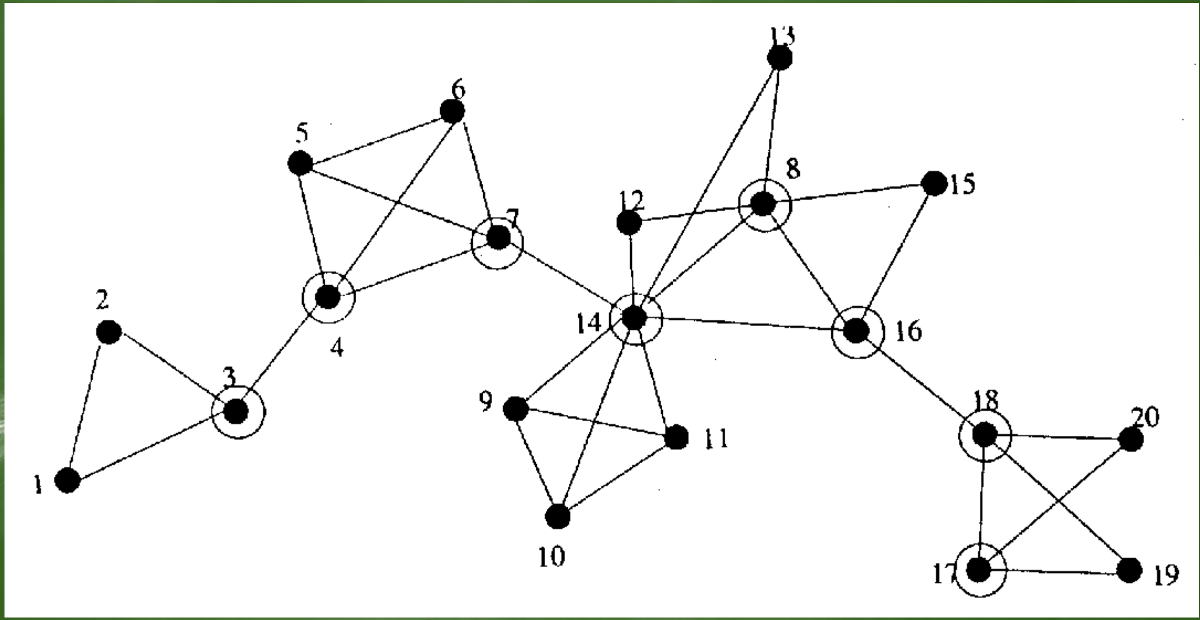




[Lenzen, Sommer, W, 2014]

Network Dynamics?

Distributed Control!


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20

Complexity Theory

Can a Computer Solve Problem *P* in Time *t*?

Distributed

Complexity Theory

Network

Can a ~~Computer~~ Solve

Problem *P* in Time *t*?

Network

~~Distributed~~

Complexity Theory

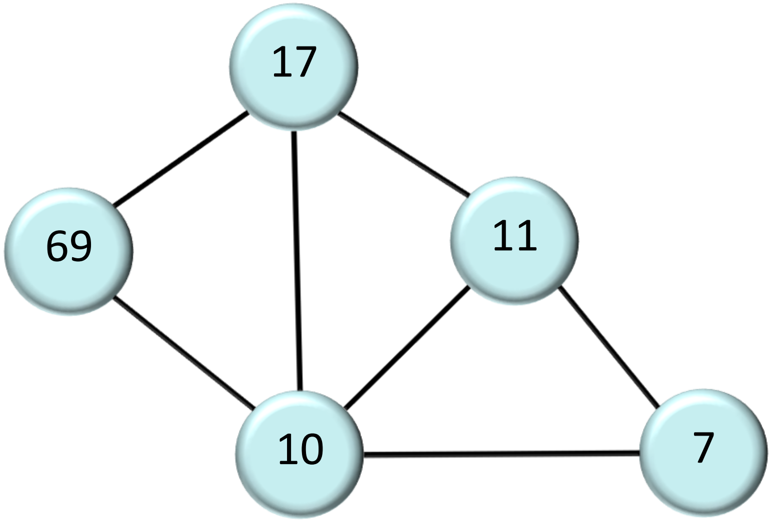
Network

Can a ~~Computer~~ Solve

Problem *P* in Time *t*?

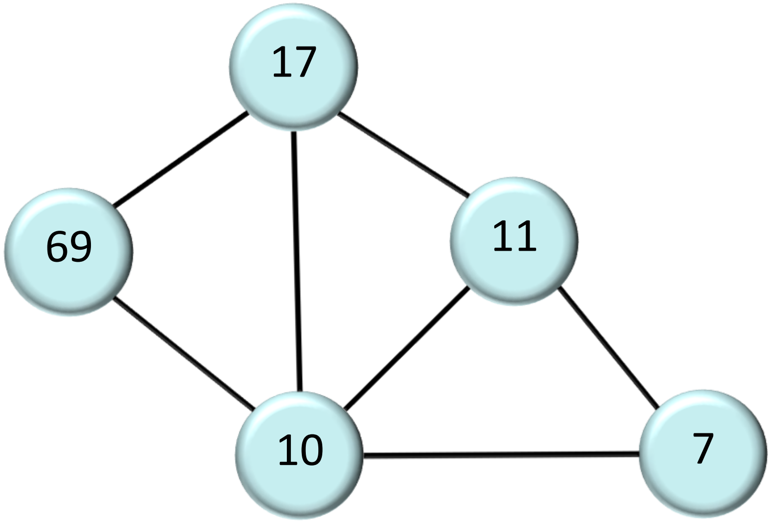
Distributed (Message-Passing) Algorithms

- Nodes are agents with unique ID's that can communicate with neighbors by sending messages. In each synchronous round, every node can send a (different) message to each neighbor.



Distributed (Message-Passing) Algorithms

- Nodes are agents with unique ID's that can communicate with neighbors by sending messages. In each synchronous round, every node can send a (different) message to each neighbor.



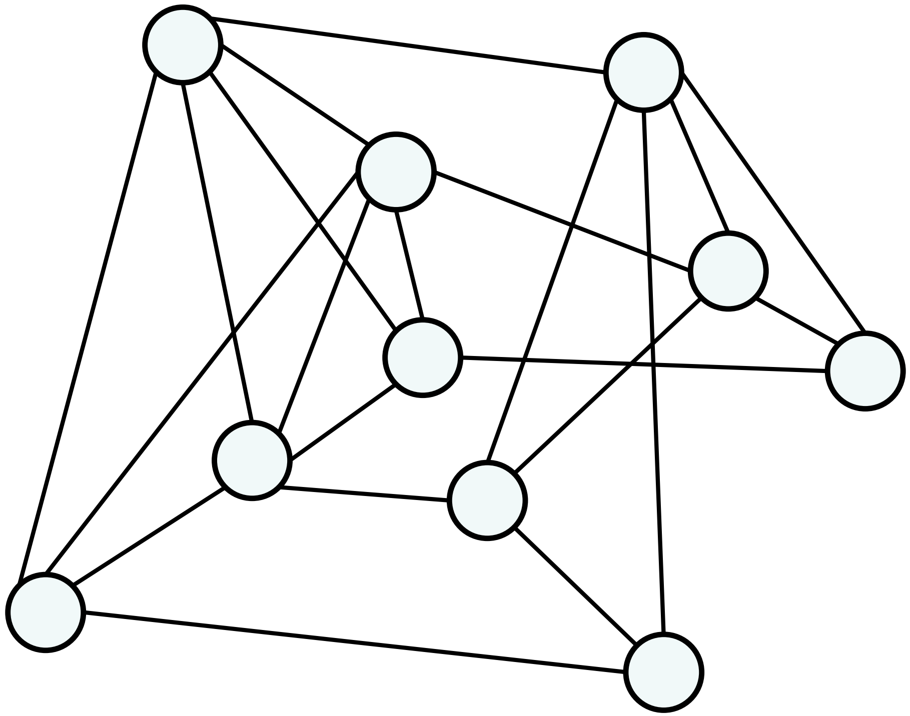

```
each round:
every node:
1. send msgs
2. rcv msgs
3. compute
```

- Distributed (Time) Complexity: How many rounds does problem take?

An Example

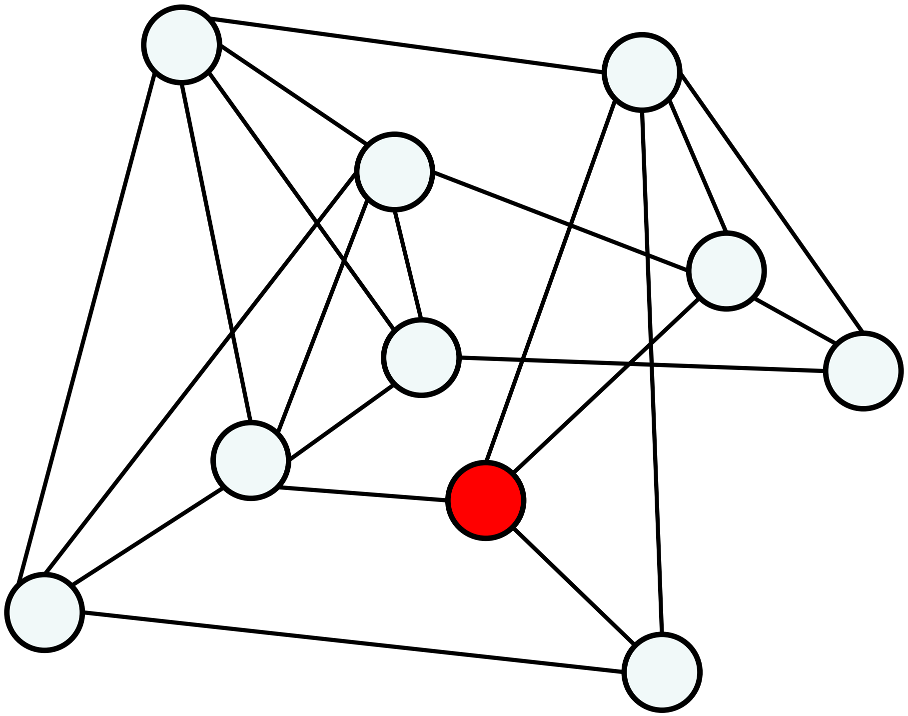
each round:
every node:
1. send msgs
2. rcv msgs
3. compute

How Many Nodes in Network?



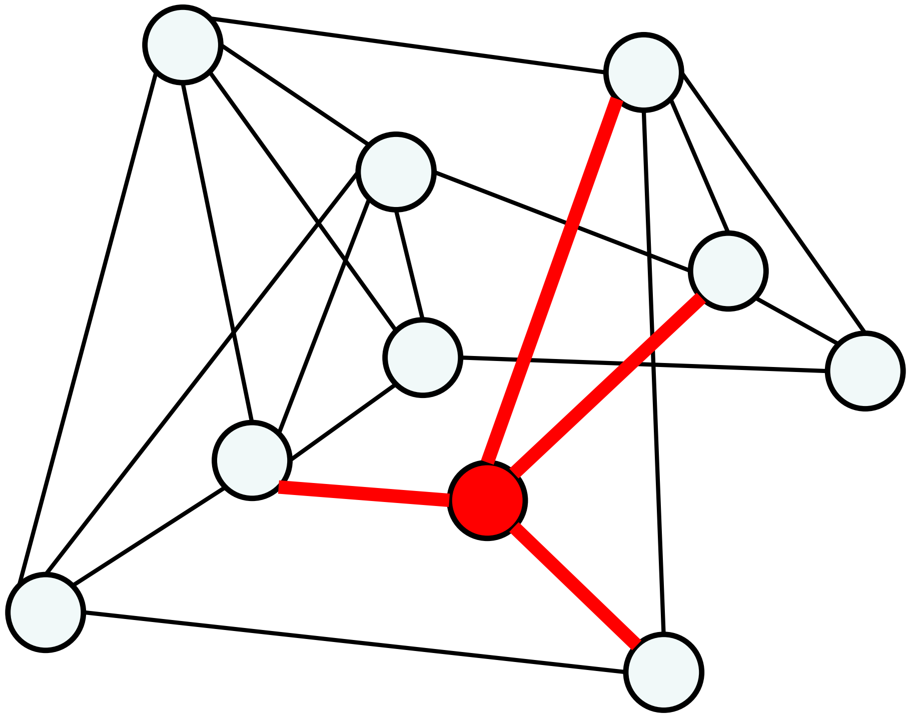
each round:
every node:
1. send msgs
2. rcv msgs
3. compute

How Many Nodes in Network?



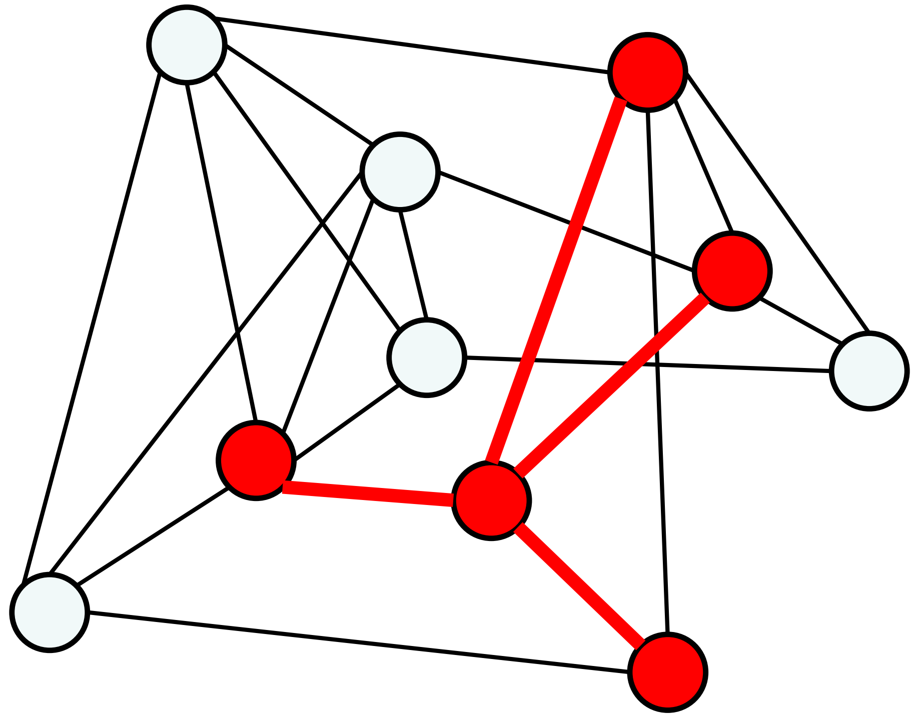
each round:
every node:
1. send msgs
2. rcv msgs
3. compute

How Many Nodes in Network?

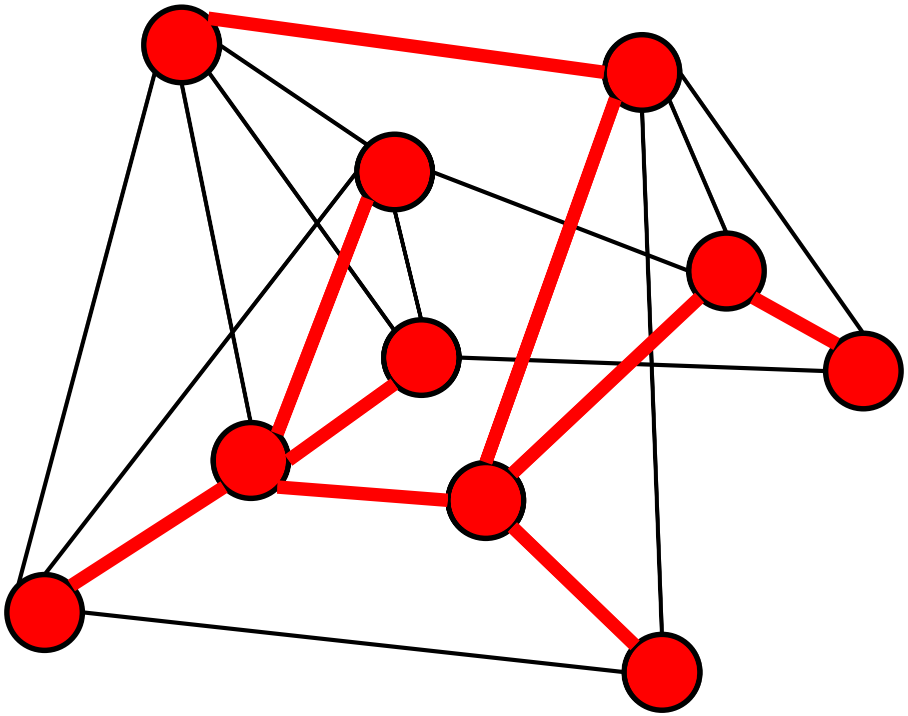


each round:
every node:
1. send msgs
2. rcv msgs
3. compute

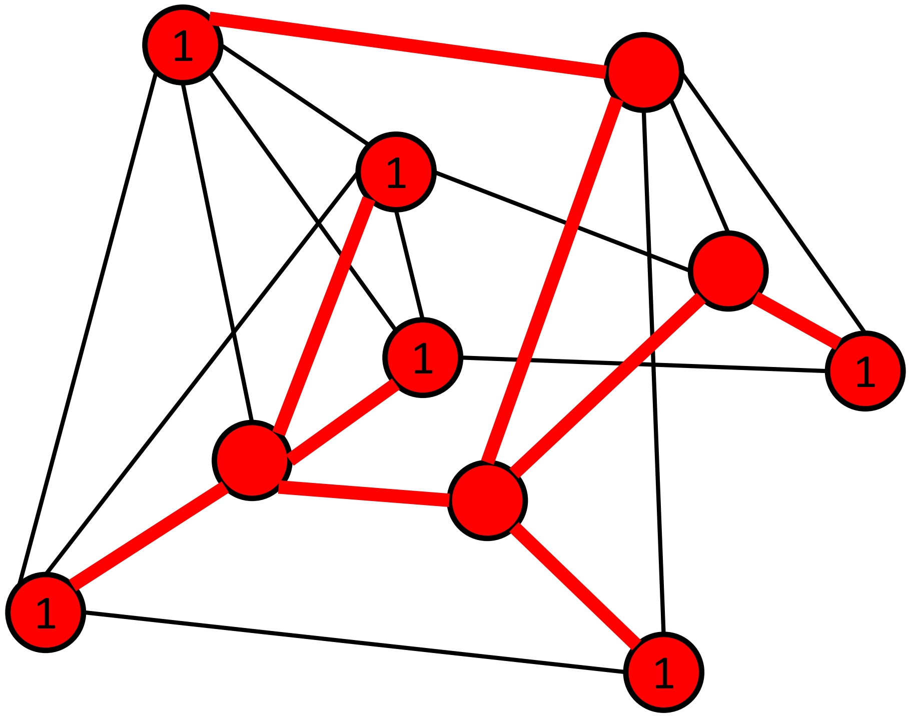
How Many Nodes in Network?



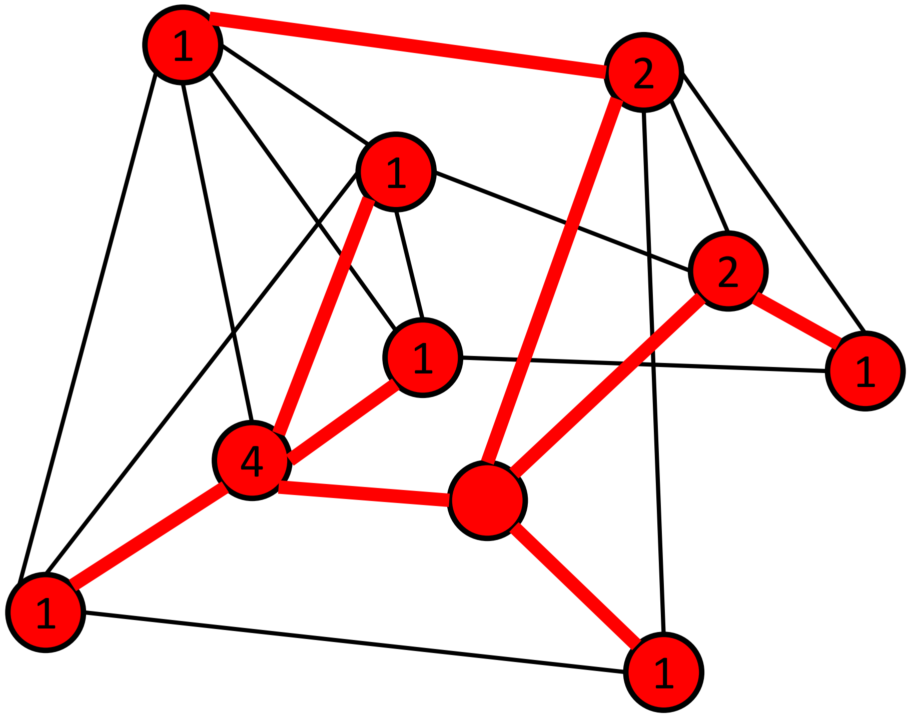
How Many Nodes in Network?



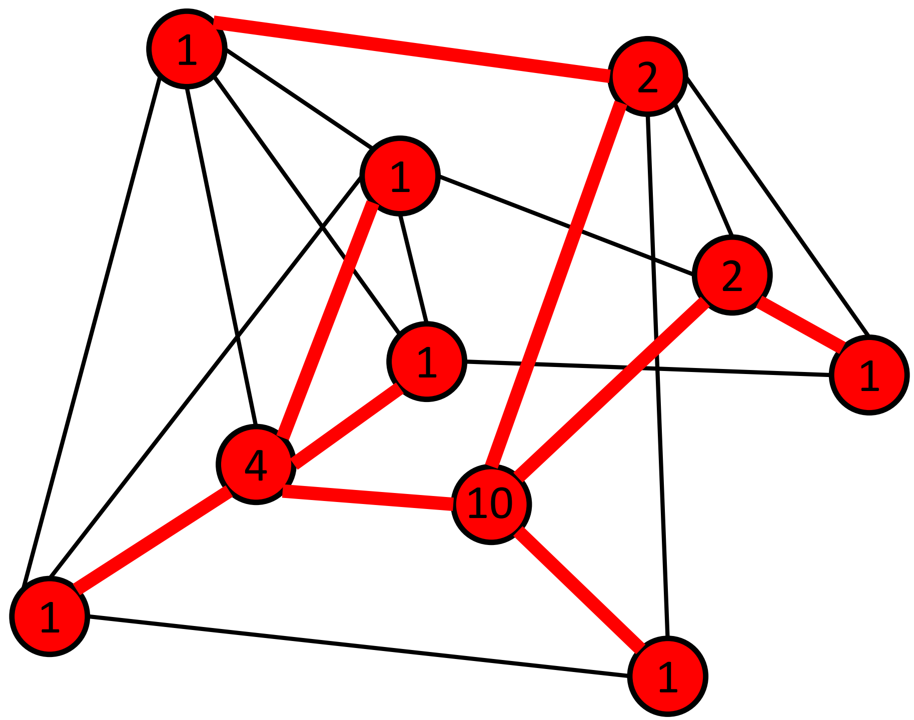
How Many Nodes in Network?



How Many Nodes in Network?

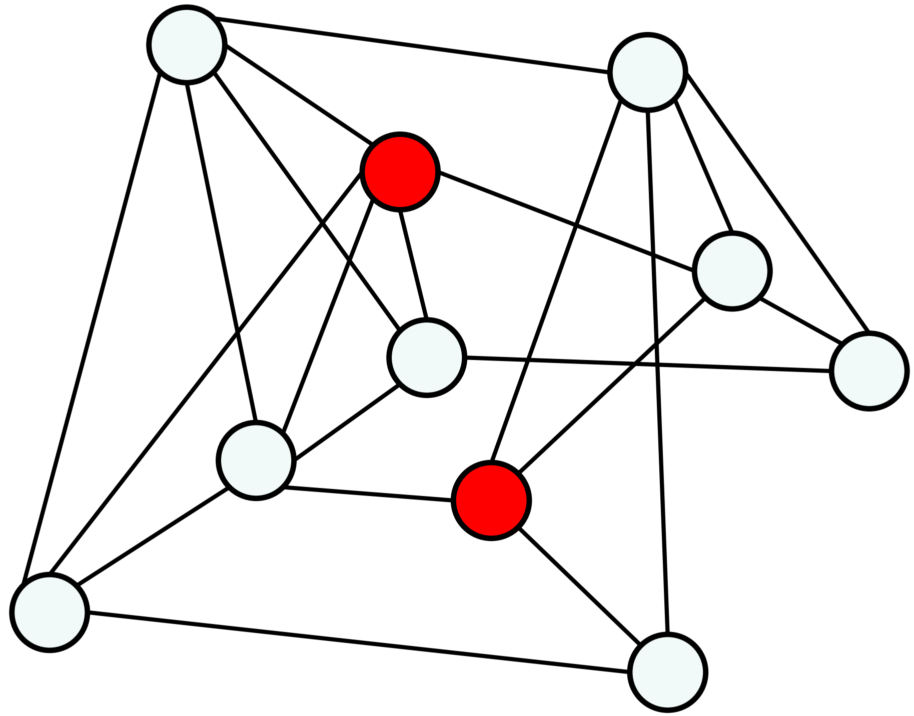


How Many Nodes in Network?



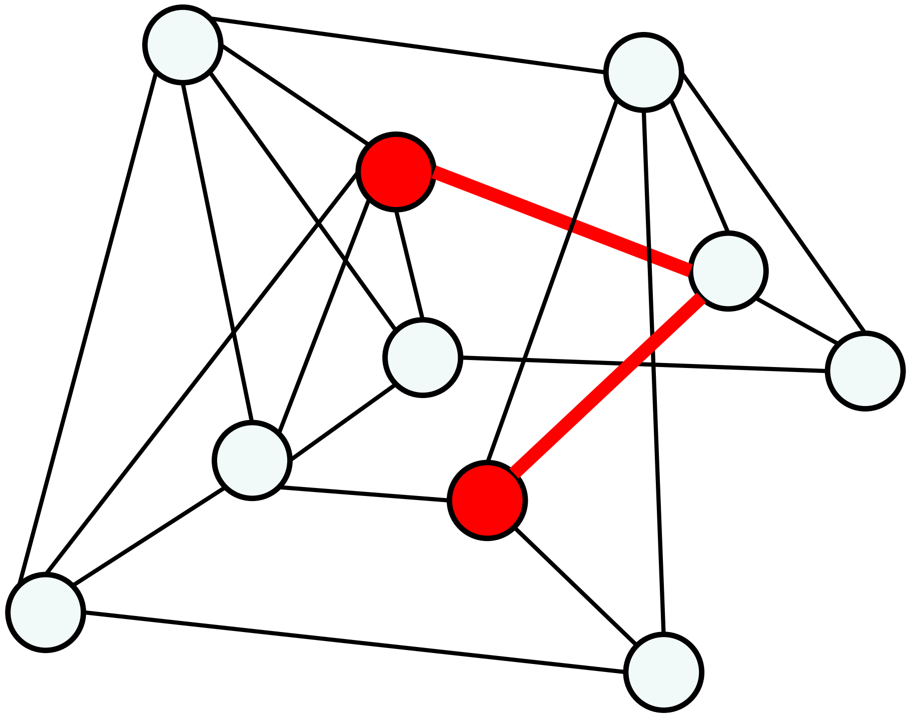
With a simple flooding/echo process, a network can find the number of nodes in time $O(D)$, where D is the diameter (size) of the network.

Diameter of Network?



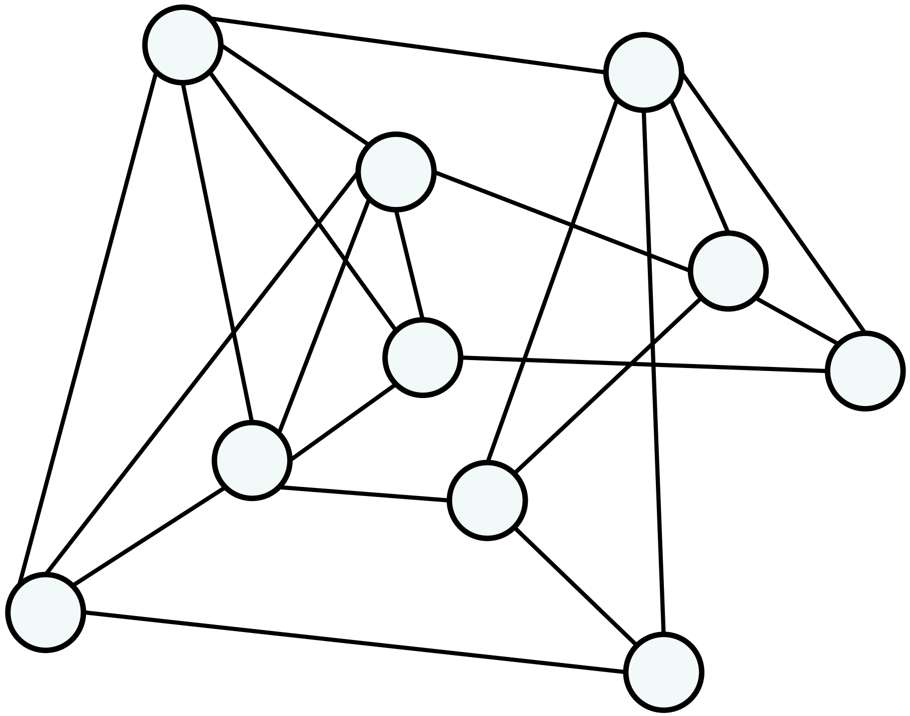
- Distance between two nodes = Number of hops of shortest path

Diameter of Network?



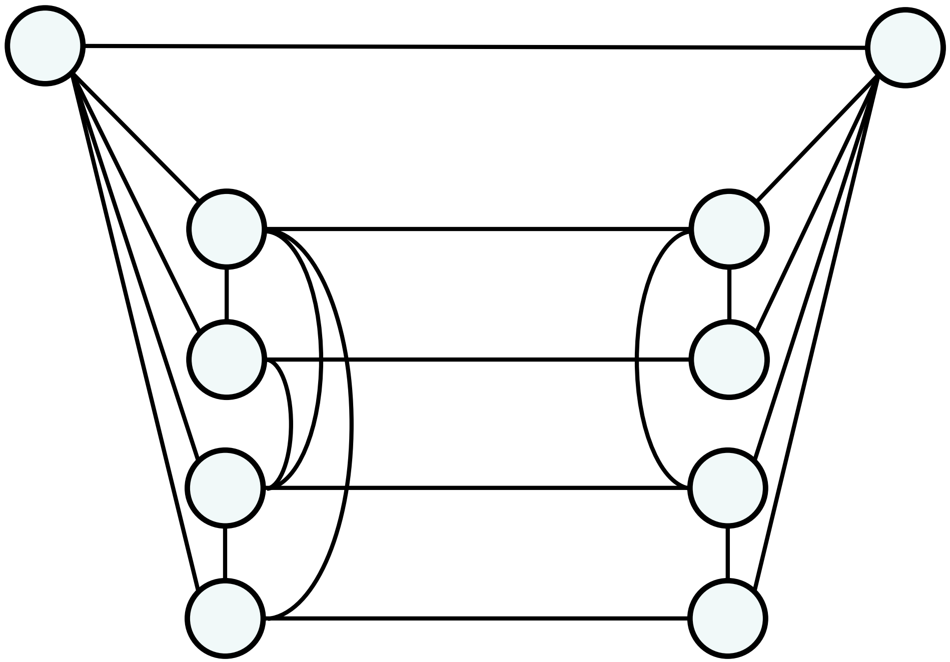
- Distance between two nodes = Number of hops of shortest path

Diameter of Network?

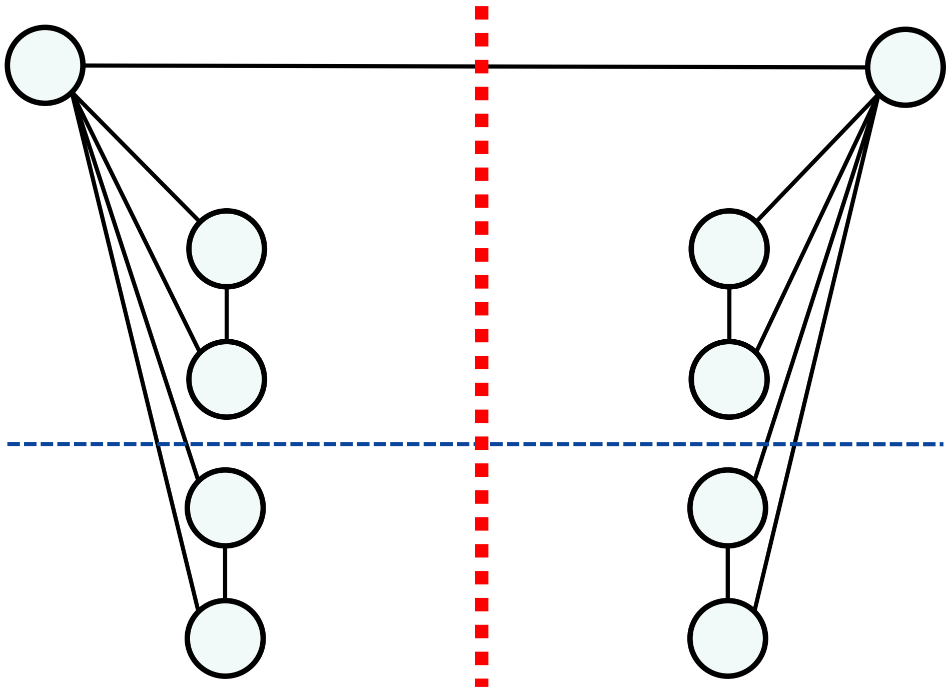


- Distance between two nodes = Number of hops of shortest path
- Diameter of network = Maximum distance, between any two nodes

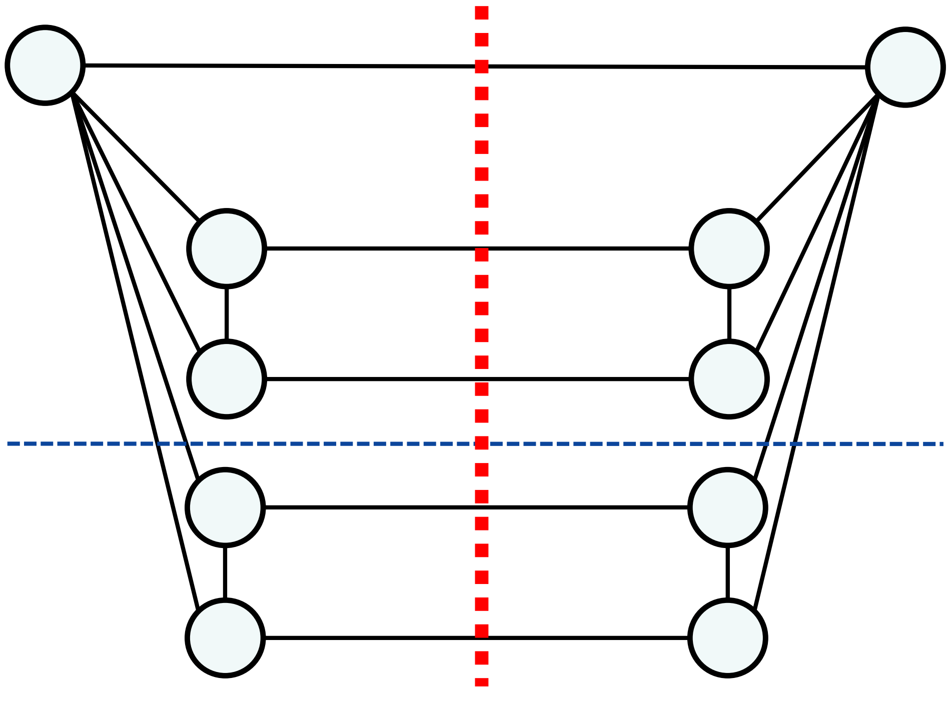
Diameter of Network?



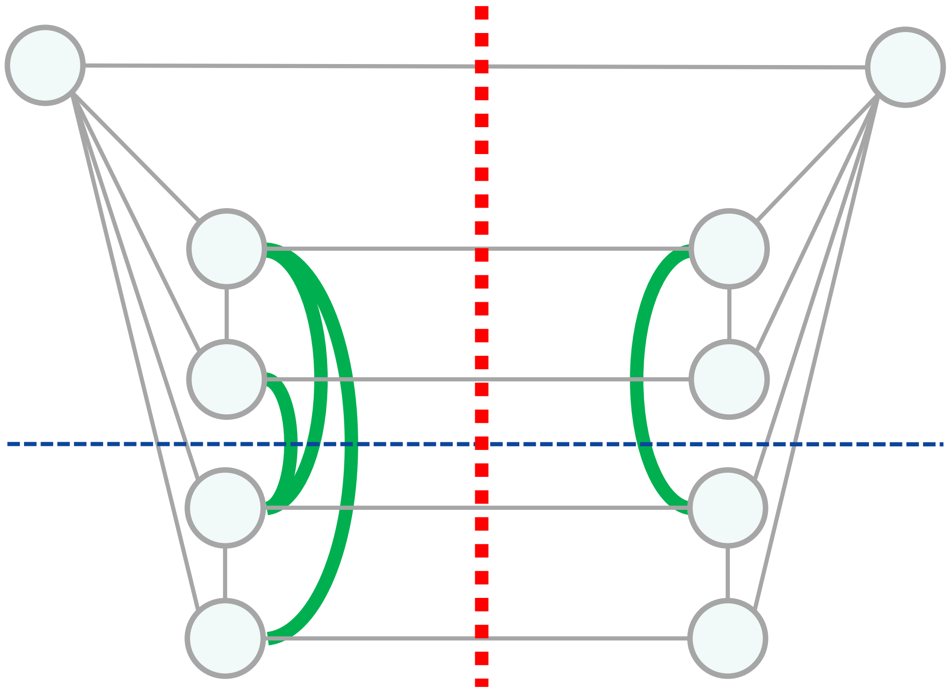
Diameter of Network?



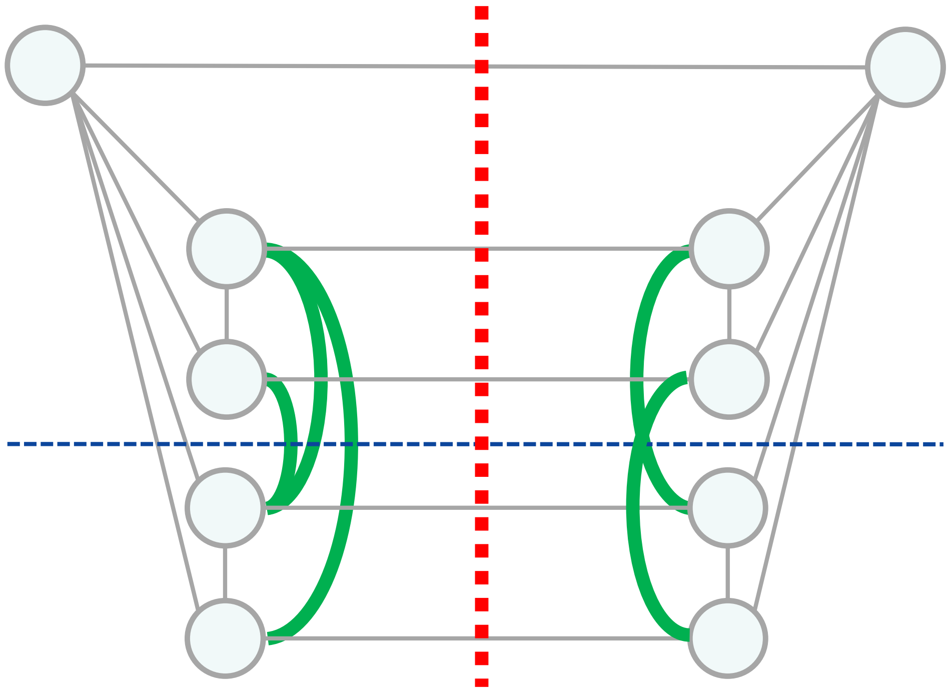
Diameter of Network?



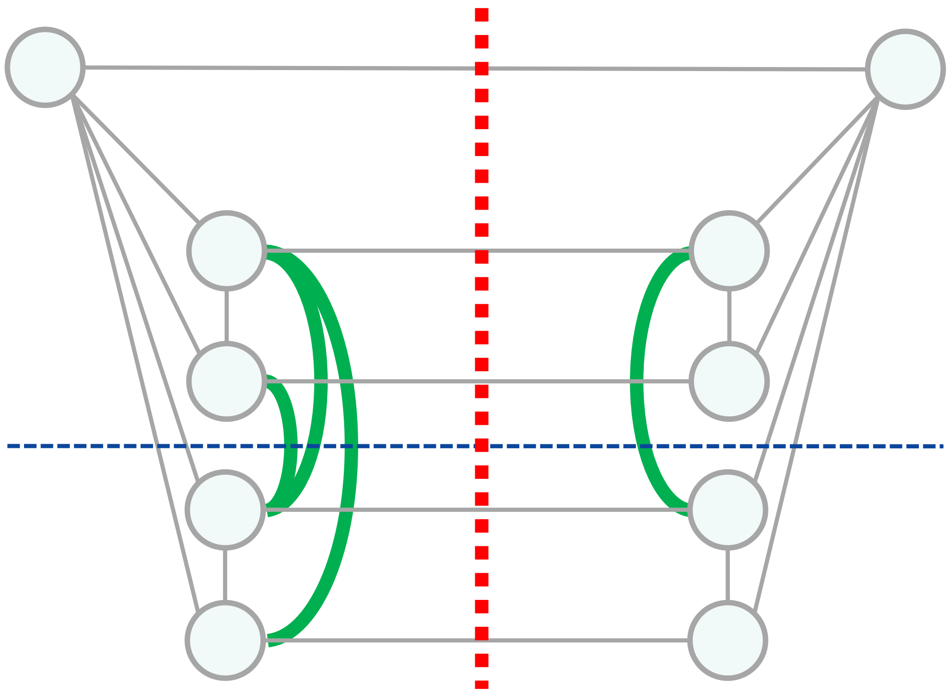
Diameter of Network?



Diameter of Network?

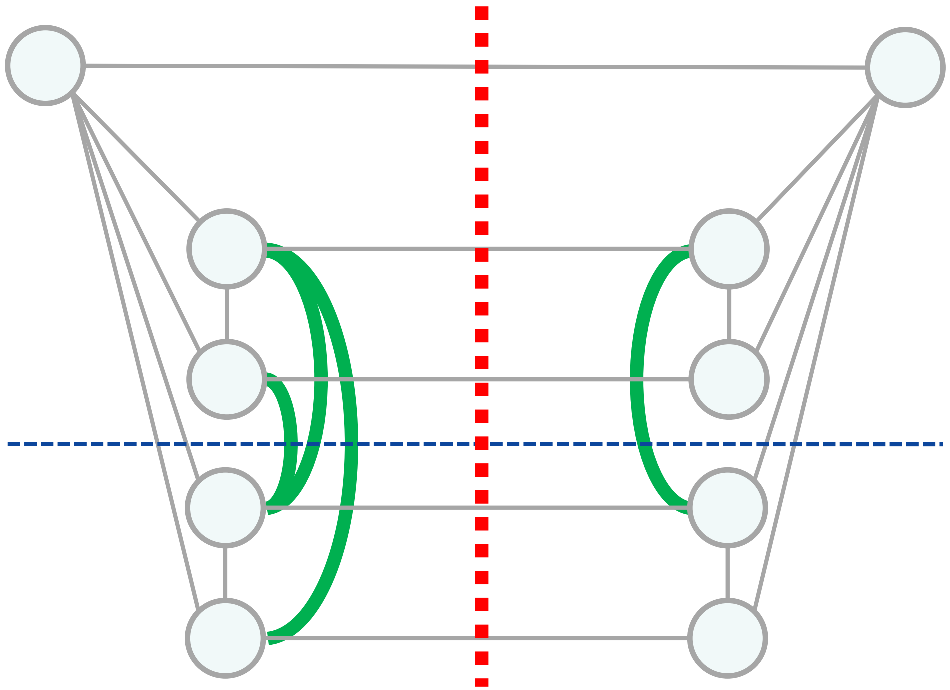


Diameter of Network?



Networks Cannot Compute Their Diameter in Sublinear Time!

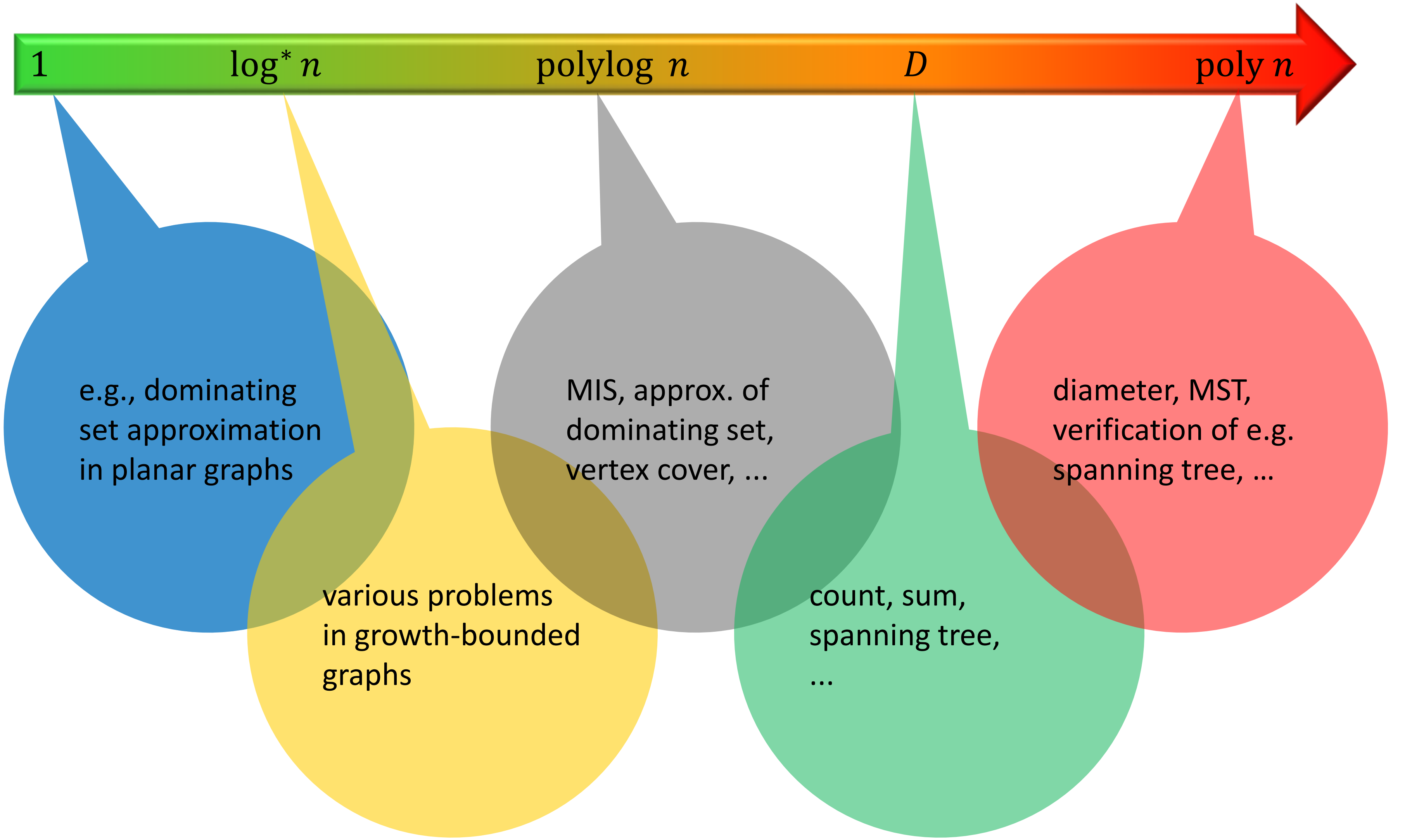
(even if diameter is just a small constant)



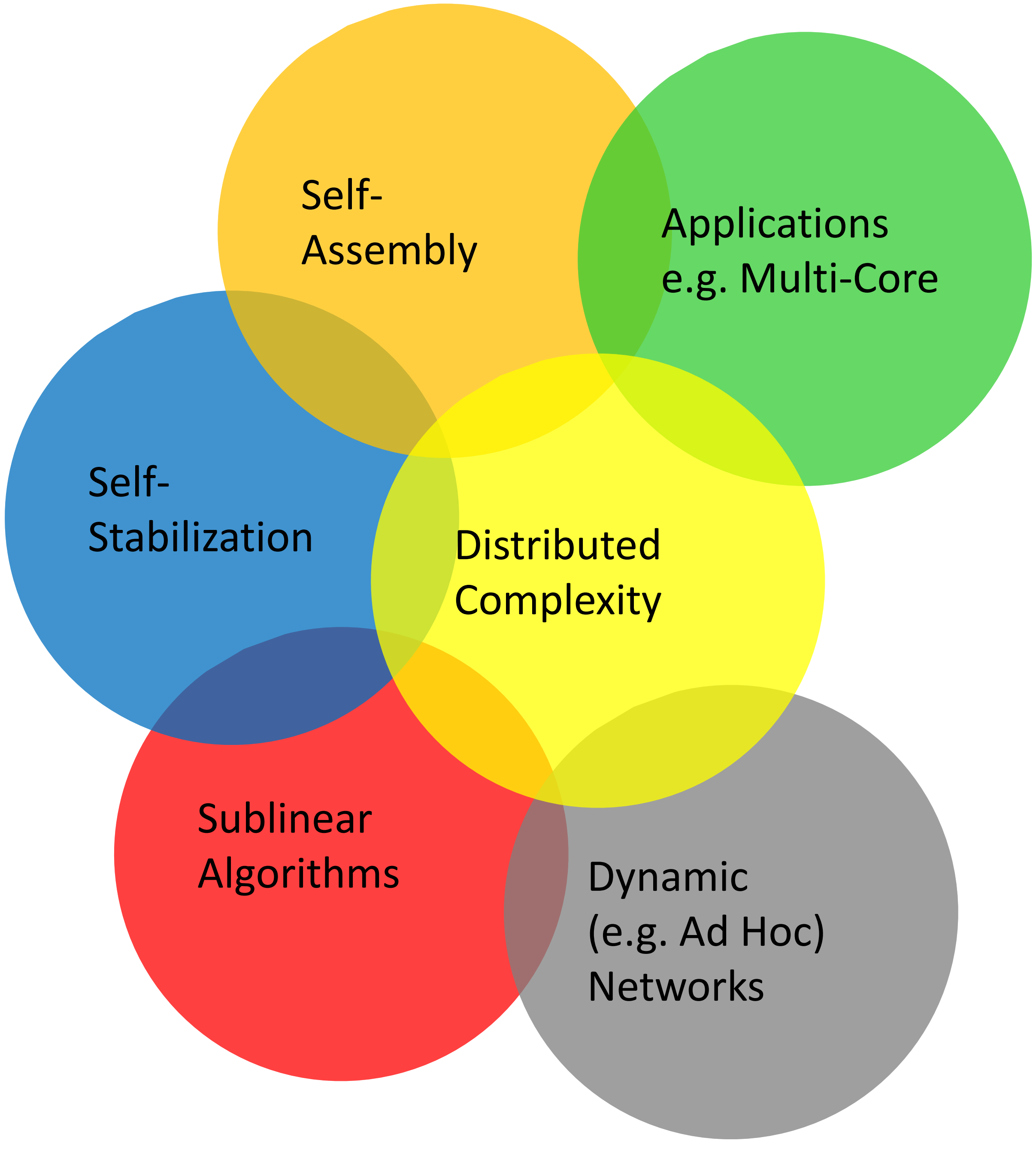
Pair of rows connected neither left nor right? Communication complexity:
Transmit $\Theta(n^2)$ information over $O(n)$ edges → $\Omega(n)$ time!

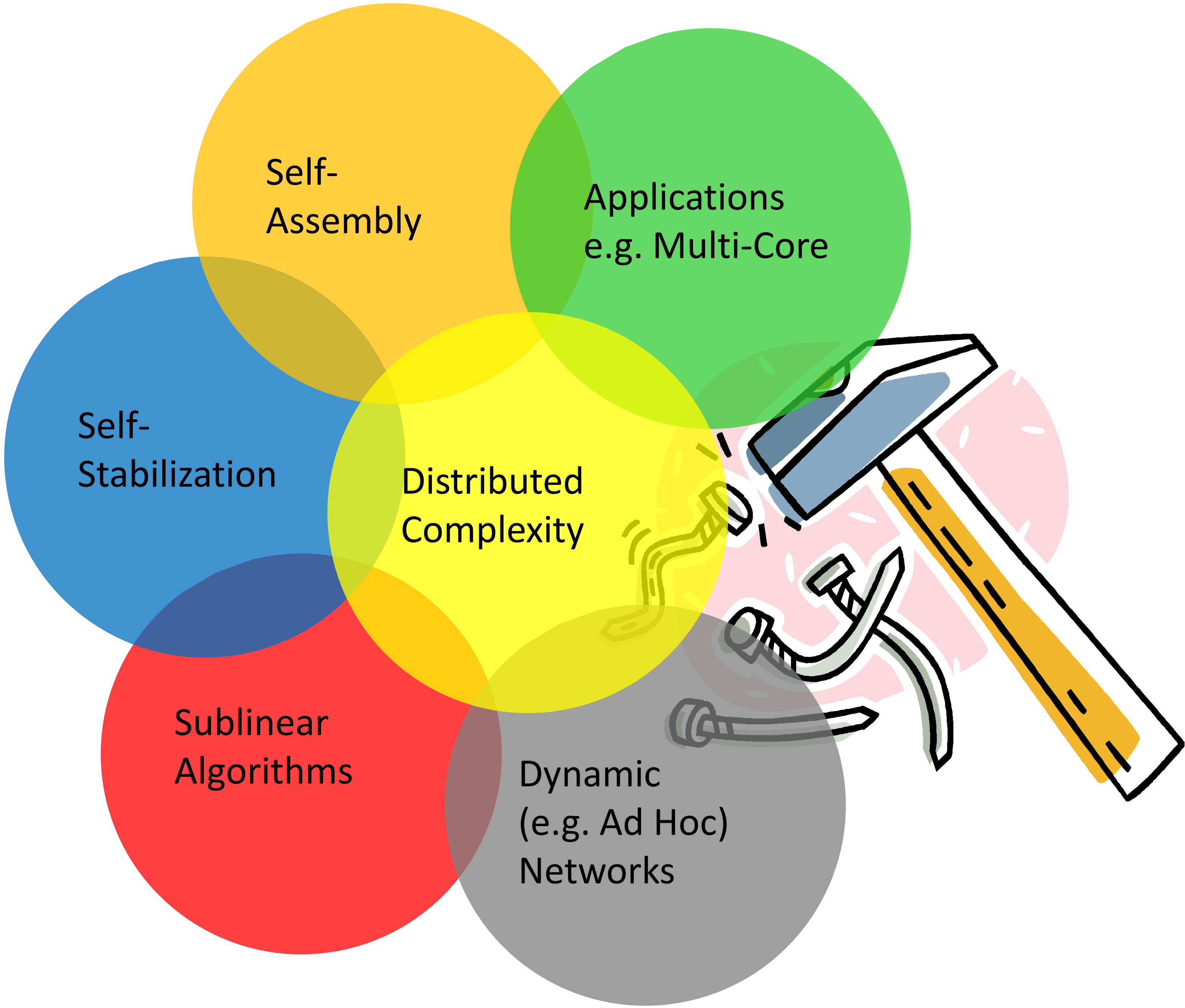
[Frischknecht, Holzer, W, 2012]

Distributed Complexity Classification

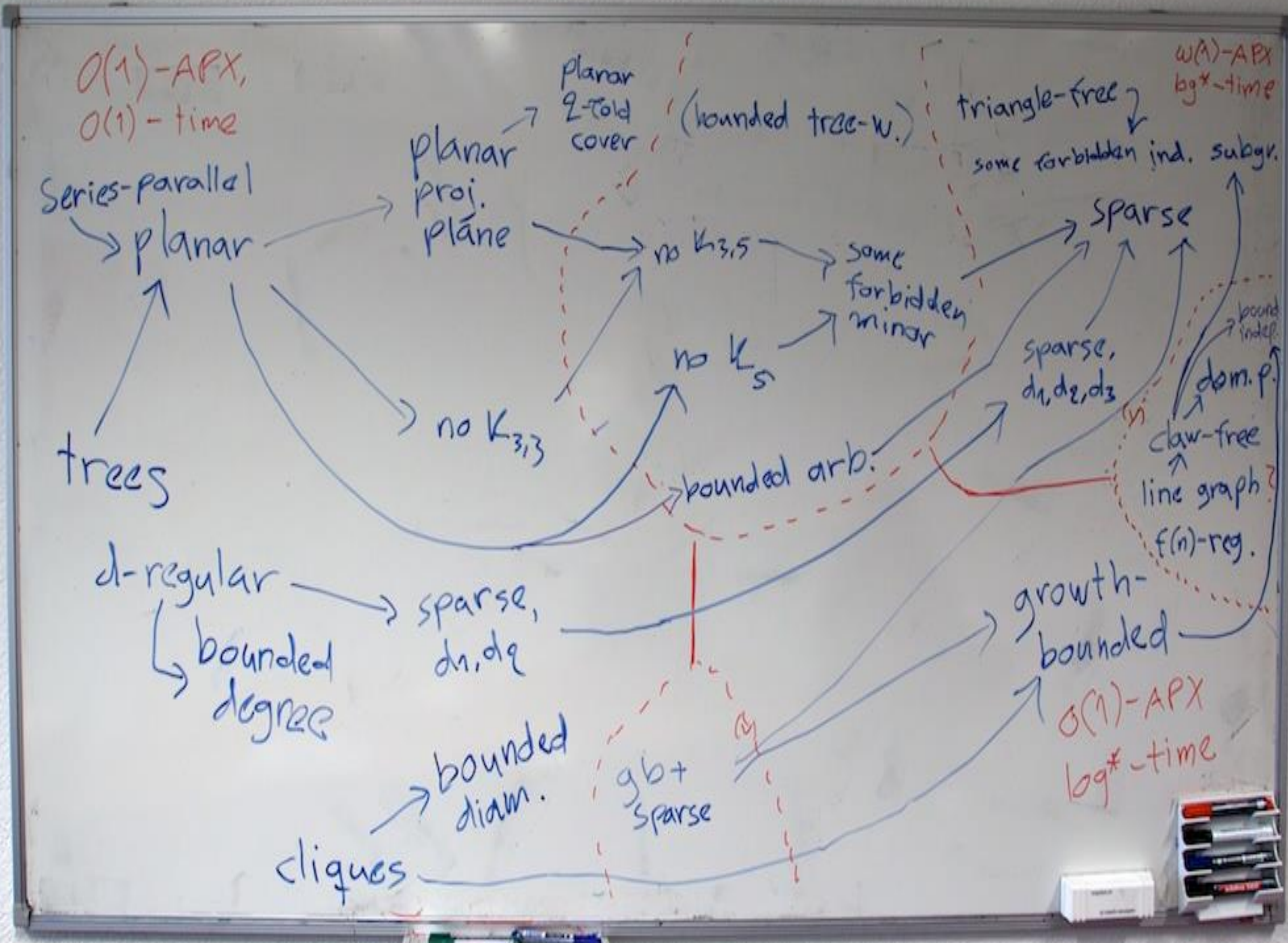


e.g., [Kuhn, Moscibroda, W, 2016]


Self-
Assembly
Applications
e.g. Multi-Core
Self-
Stabilization
Distributed
Complexity
Sublinear
Algorithms
Dynamic
(e.g. Ad Hoc)
Networks

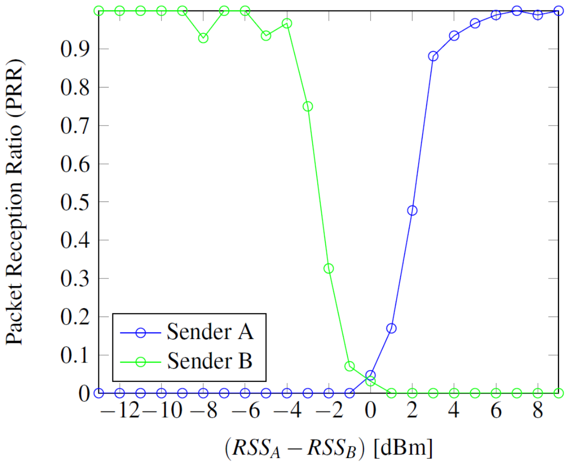


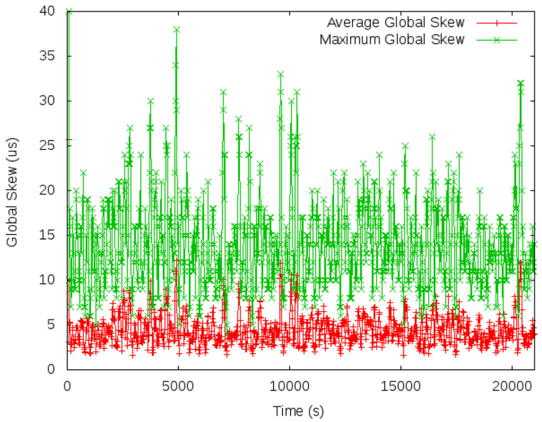
Self-
Assembly
Applications
e.g. Multi-Core
Self-
Stabilization
Distributed
Complexity
Sublinear
Algorithms
Dynamic
(e.g. Ad Hoc)
Networks

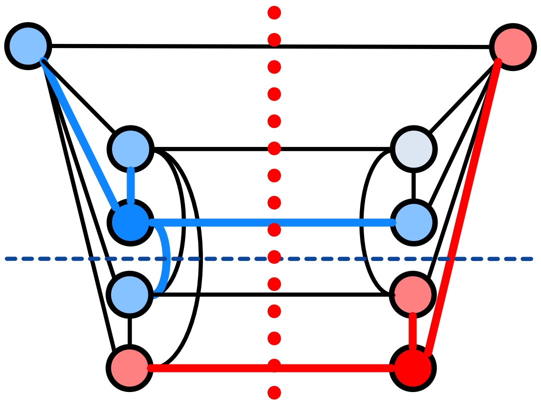


O(1)-APX,
O(1)-time
planar
2-fold
cover
(bounded tree-w.)
triangle-free
ω(1)-APX
log*-time
some forbidden ind. subgr.
series-parallel
planar
planar
proj.
plane
no $K_{3,5}$
some
forbidden
minor
sparse
no K_5
sparse,
d_1, d_2, d_3
bound
indep.
dom. p.
trees
no $K_{3,3}$
claw-free
bounded arb.
line graph?
f(n)-reg.
d-regular
sparse,
d_1, d_2
growth-
bounded
bounded
degree
O(1)-APX
log*-time
bounded
diam.
gb+
sparse
cliques

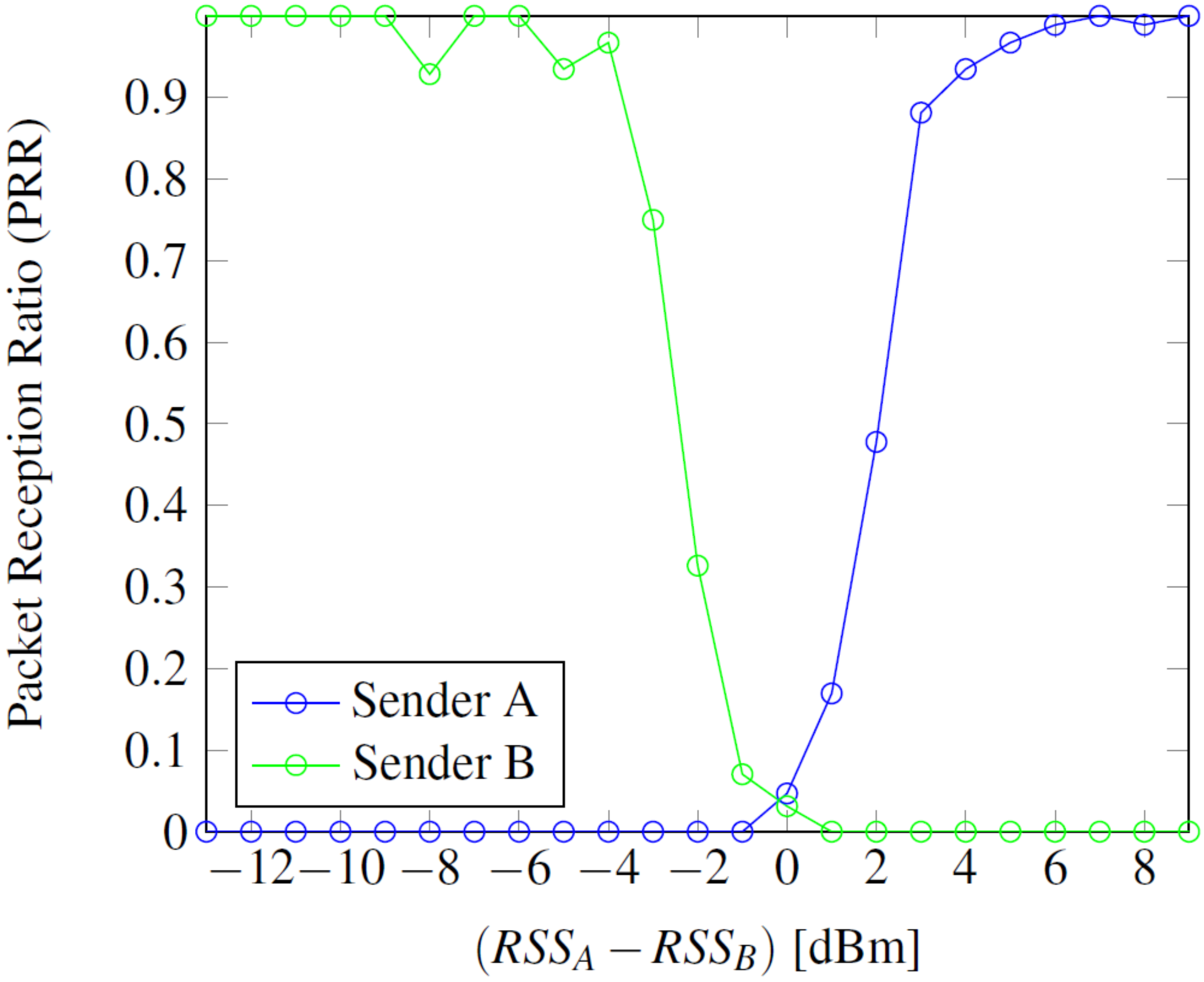
Summary


Packet Reception Ratio (PRR)
Sender A
Sender B
$(RSS_A - RSS_B)$ [dBm]


Average Global Skew
Maximum Global Skew
Global Skew (us)
Time (s)



The Capture Effect



Theory for sensor networks, what is it good for?!

How many lines of pseudo code
Can you implement on a sensor node?

The best algorithm is often complex
And will not do what one expects.

Theory models made lots of progress
Reality, however, they still don't address.

My advice: invest your research £££s
in ... impossibility results and lower bounds!

Thank You!

Questions & Comments?

Thanks to my co-authors, mostly
Silvio Frischknecht
Magnus Halldorsson
Stephan Holzer
Michael König
Christoph Lenzen
Thomas Moscibroda
Philipp Sommer