The Heathland Experiment
Results And Experiences

V. Turau, Ch. Renner, M. Venzke, S. Waschik, Ch. Weyer, M. Witt

Hamburg University of Technology, Germany

Introduction

- Wireless sensor networks very active field
- But:
  - Number of theoretical publications >> number of deployments
  - Vast majority algorithms only simulated & not implemented on real sensor networks
  - Current simulation tools are not capable to completely represent sensor networks in harsh environment over long period of time
- Real deployments are indispensable!

Heathland Experiment

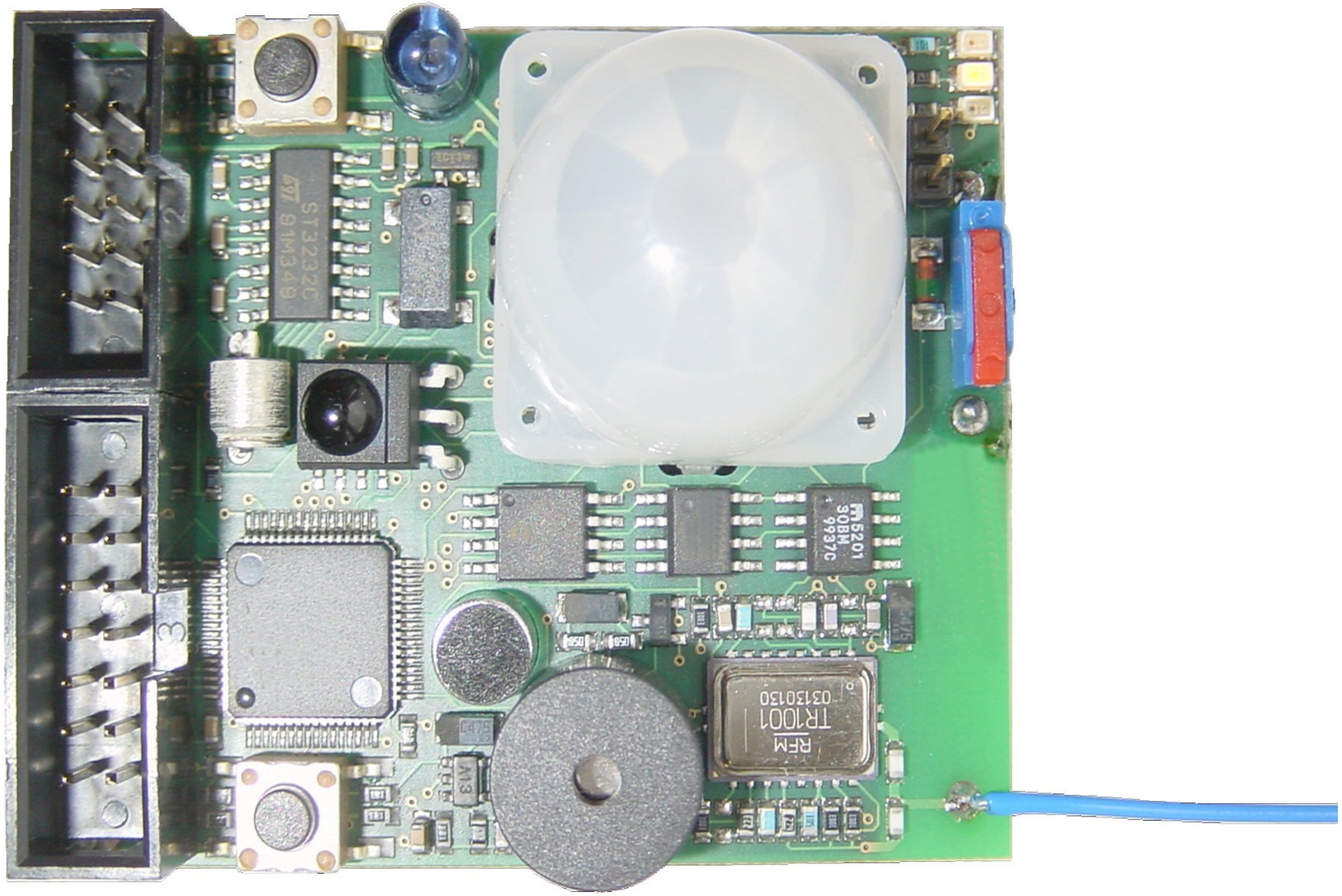
- Real deployment of a sensor network
- 24 nodes running for 2 weeks in March 2005
- Location: Heathlands of Northern Germany
- First outdoor long term usage of Embedded Sensor Board (ESB)
- Application ran without any human attention
- Objective:
  Reveal problems that can only be discovered through experience

Goals

- General radio performance (packet loss, packet errors)
- Constancy of transmission ranges
- Performance of our neighborhood protocol
- Communication in a multi-hop environment
- Dynamic adaption to changes of transmission ranges
- Test depth-first search implementation
- Increased expectation of life was not a goal

Platform: ESB Nodes

Source: www.scatterweb.net



Platform: ESB Nodes

- Texas Instruments micro controller MSP 430
- Transceiver TR1001, 868 MHz, 19.2 kbit/s
- RS232 serial interface
- Sensors: light, passive infrared, temperature, vibration, microphone
- Tunable radio transmit power
- 2KB RAM, 8 KB EEPROM
- Powered by 3 AA batteries
- Power consumption: 8 µA (sleep mode) up to 12 mA (all sensors on)

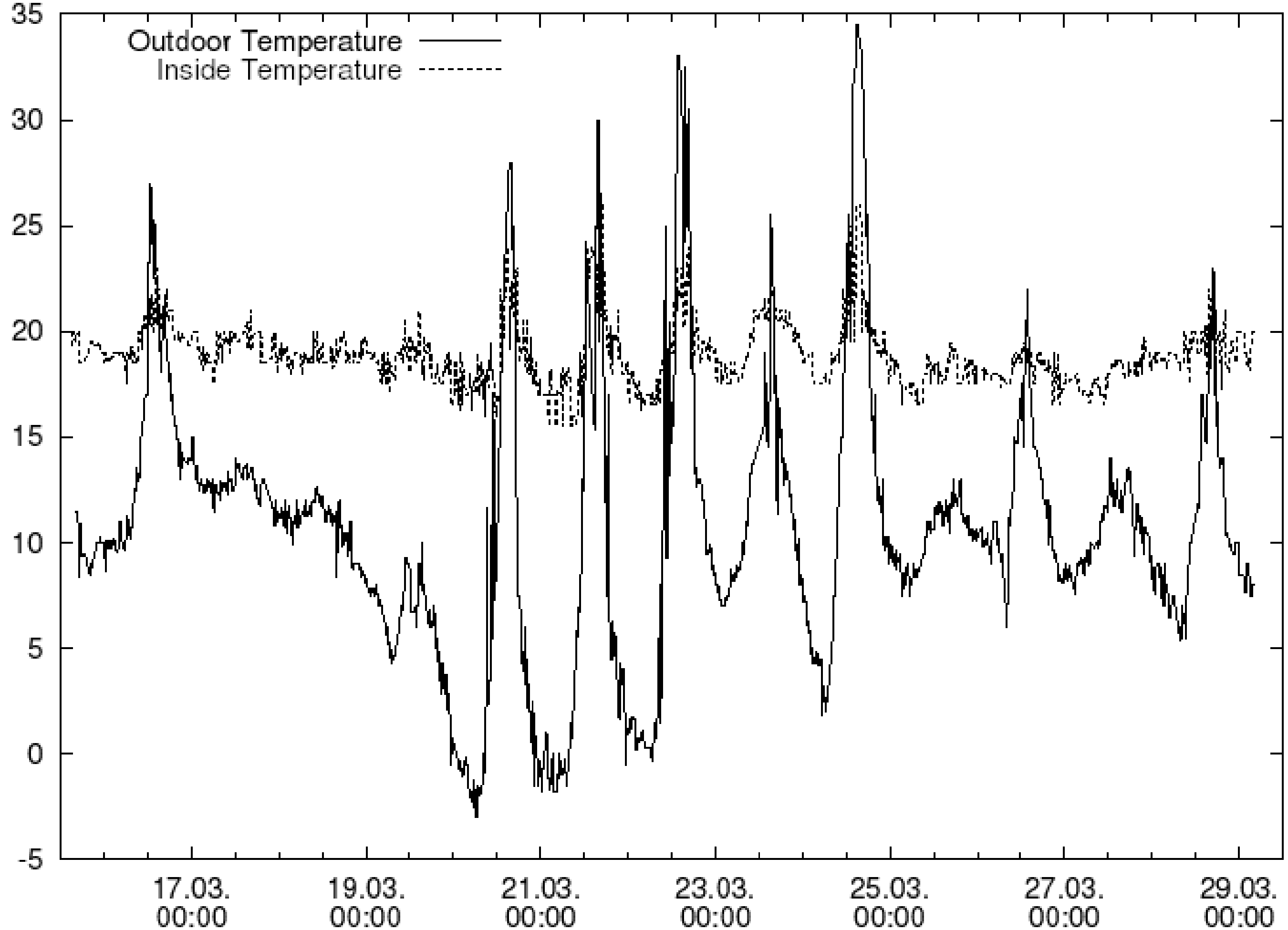
Packaging

Packaging

Packaging

Weather

Recorded Temperature



Application

- Sense & send: Sensors deployed in wide-ranging area take periodic readings & report to central repository
- Nodes periodically send readings from their 5 sensors to sink
- To account for topology changes, routing trees and neighbor relationships were recomputed regularly
- Scalable application
- Application could cope with node failures, deployment of new nodes, decreasing communication radii due to fading battery energy
- Multi-hop network

Application

Sense & Send Application

Depth-first Algorithm Routing

Wireless Neighborhood Exploration WNX

Radio Transmission Protocol Firmware

WNX

- Modified implementation of TND, the proactive neighbor discovery protocol of TBRPF (RFC 3684)
- Simple, agile yet stable
- Uni- & bidirectional links
- Each node records last 32 expected hellos from each neighbor
- Quality descriptor: weighted moving average
- Hellos associated with weights → Link-quality between 0 and 255
- Radio signal strength indicator, not included

WNX

- At most 8 neighbors per node
- It takes 30 sec to discover new bidirectional link / to purge lost link
- Small memory footprint : 152 Bytes per node
- Currently protocol is extended

Application

- Uncertainty in radio communication → time triggered scheme
- Activities are initiated by progression of globally synchronized time-base
- Due to clock drift between individual nodes, sink sends periodically its local time into network
- All nodes including the sink have the same code
- Simplified deployment process
- Sink node sent data over a standard serial port interface to a PC (about 6MB per day)

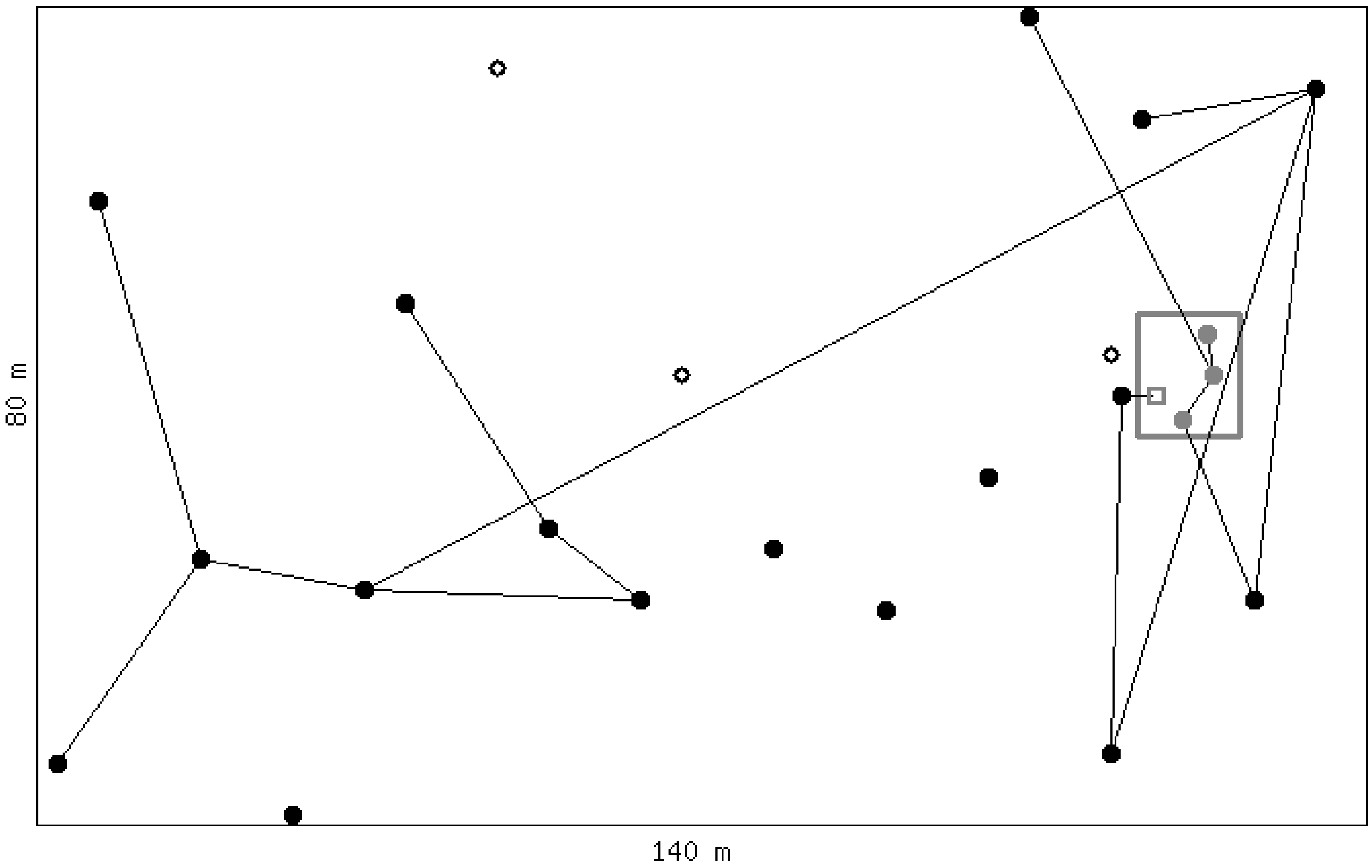
At start of every clock hour:

Time	Action	Result
0	Reset	Resets state of node
1	Start WNX	Nodes send Hello packets, compute link qualities and determine bidirectional links
9	Suspend WNX	List of bidirectional links with link quality
12	Start DFS	Depth-first search tree
14	Start measurements	Leaf nodes turn off radio, inner nodes turn off sensors In intervals of 10 minutes: - leaf nodes turn on radio - all nodes send measured data & link states to sink - the sink sends its local time to all nodes in tree - leaf nodes turn off radio

Application

- To reduce interferences, send time of leaf nodes randomly distributed in interval
- Every packet includes:
  - time-stamp with respect to local clock
  - remaining battery energy
  - number of packets received and sent
  - number of packet transmission retries
  - information about clock drift
- Packet size between 70 and 80 Bytes

Deployment



Territory

Deployment

Deployment

Deployment

Deployment

Deployment

Configuration

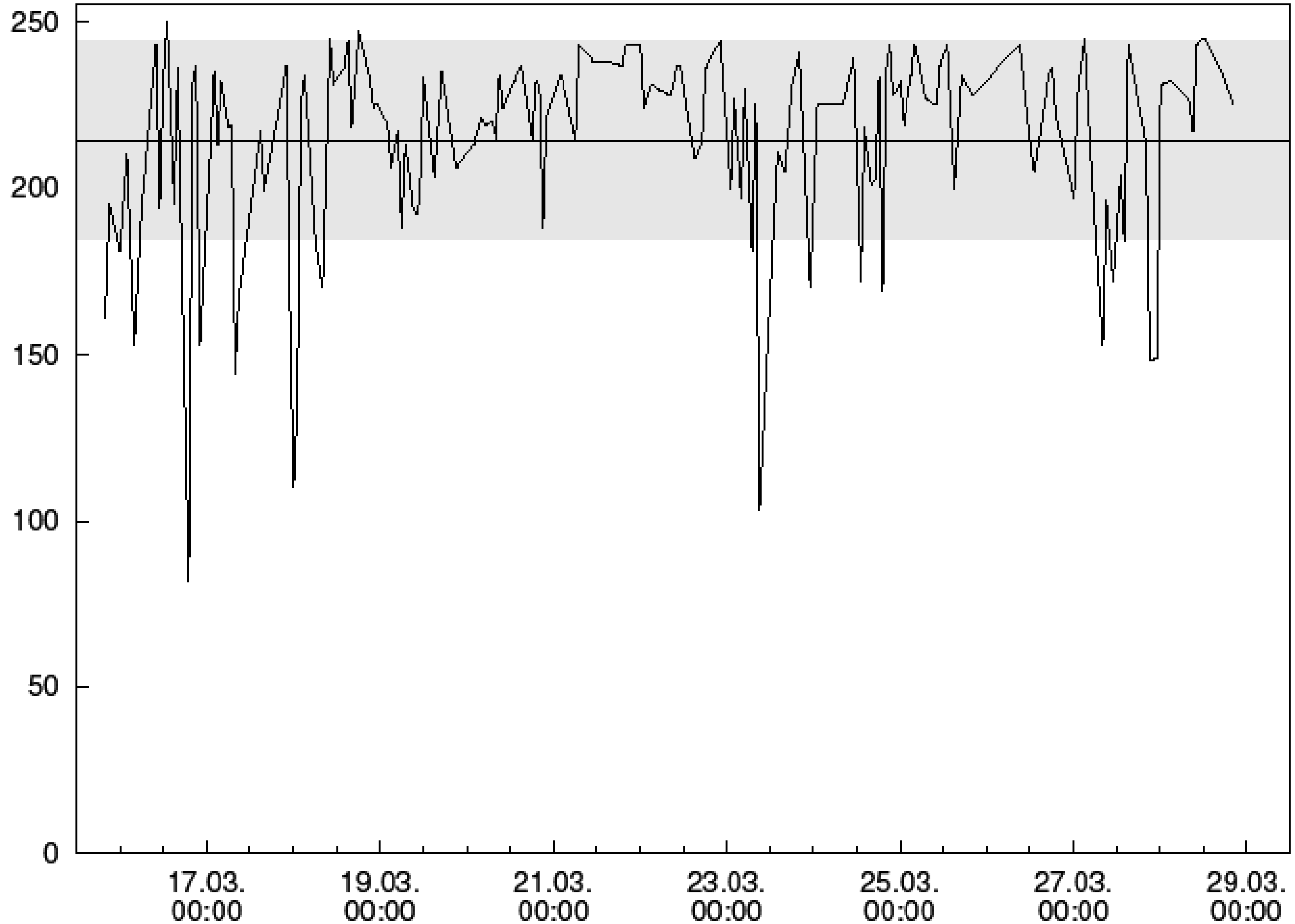
Radio Link Features

- Asymmetrical links: Connectivity from node A to node B might differ significantly from B to A
- Non-isotropical connectivity: connectivity need not be same in all directions (at same distance)
- Non-monotonic distance decay: Nodes geographically far away from source may get better connectivity than geographically closer nodes
- Link asymmetries may be caused by differences in hardware calibration
- Communication gray zones: Data messages cannot be exchanged although the HELLO messages indicate neighbor reachability

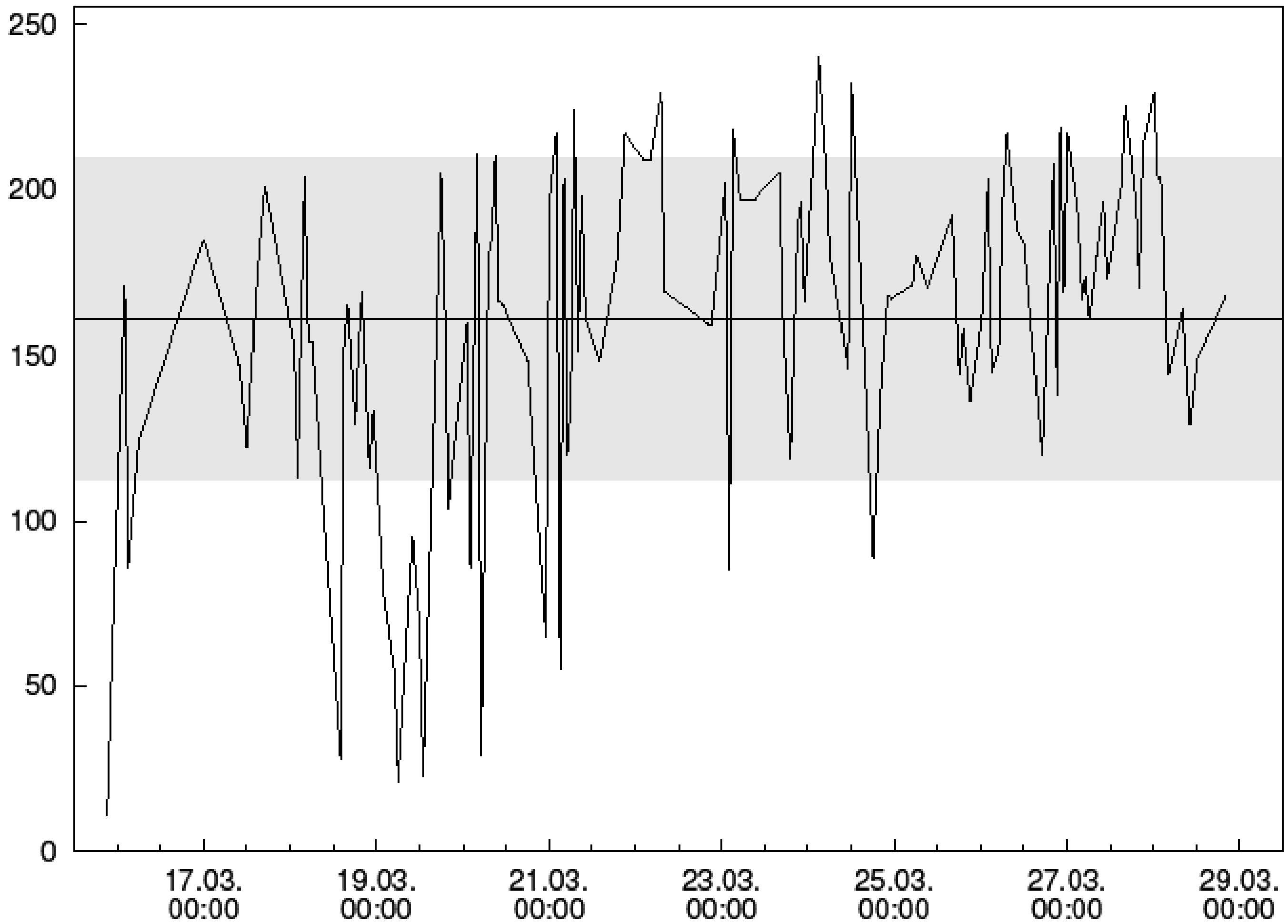
Analysis

- Altogether 24 nodes were used
- 3 nodes did not send significant amount of data after the first day (reasons are unknown)
- Remaining 21 nodes
- Out of the 210 different pairs of nodes, 45 appeared as links in a search tree
- Quality of these links varied considerably

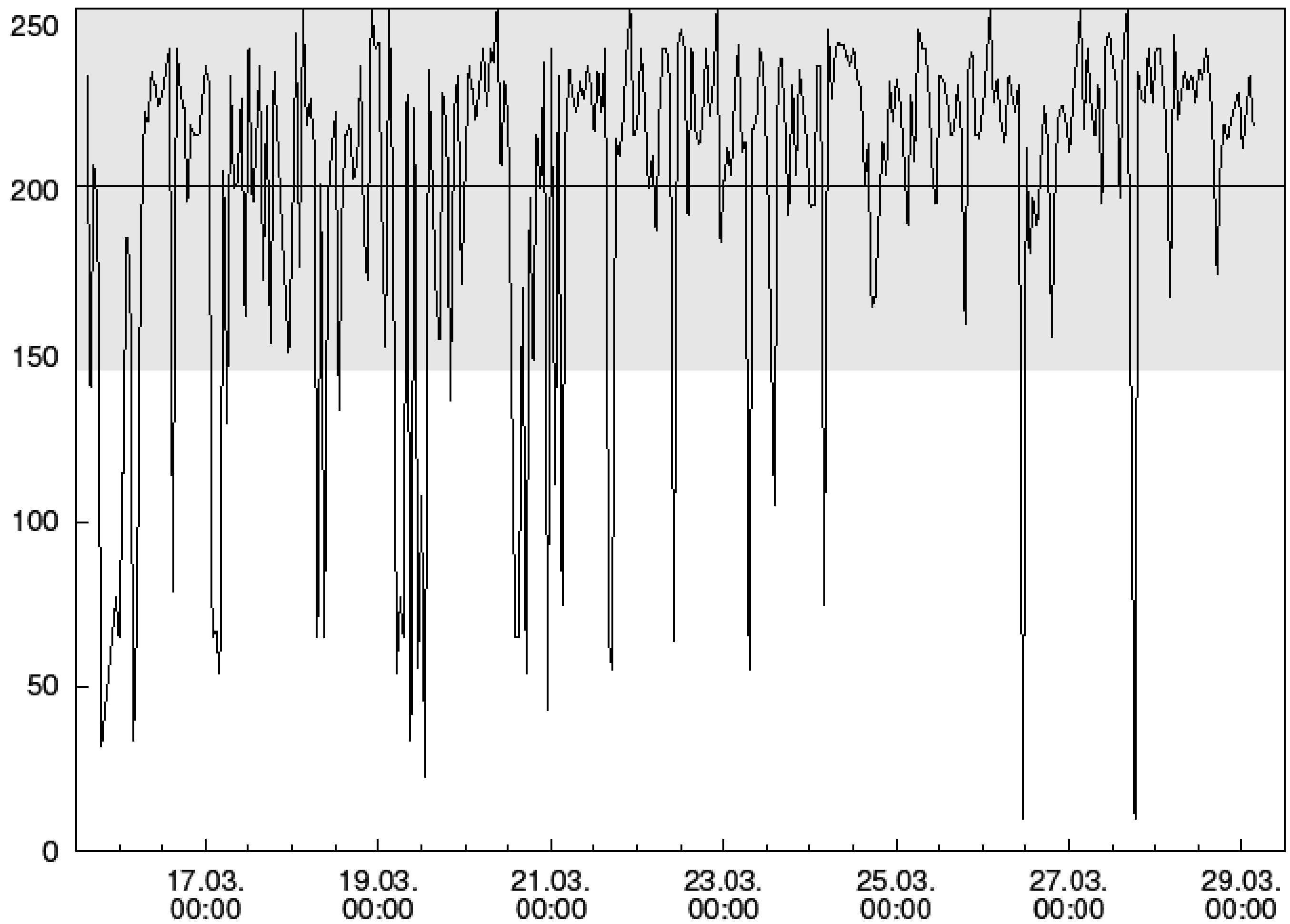
2 Indoor Nodes



2 Outdoor Nodes

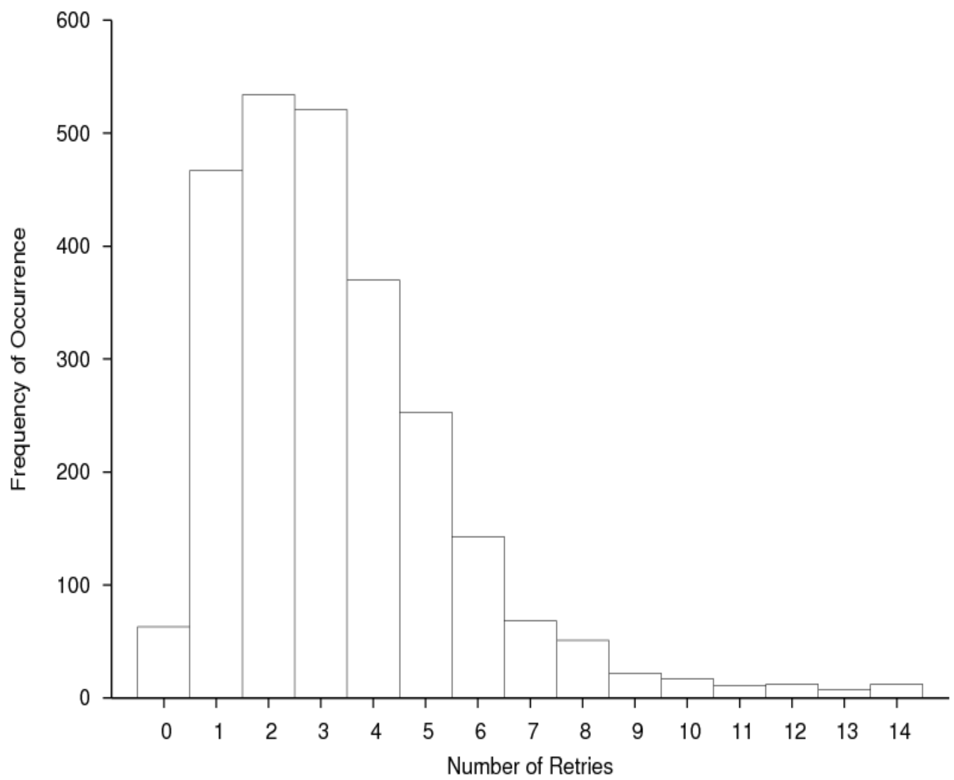


2 Outdoor Nodes



Retries in Unicast Transmission Scheme

- 15 retransmissions
- Maximal latency: 1143 ms, not including waiting time for channel access
- Observed latencies up to 3 s
- ~ 50% of all unicasts were successful
- Average number of retries: 3.89 → limit was too high
- Gnawali et al. max. 3 retries

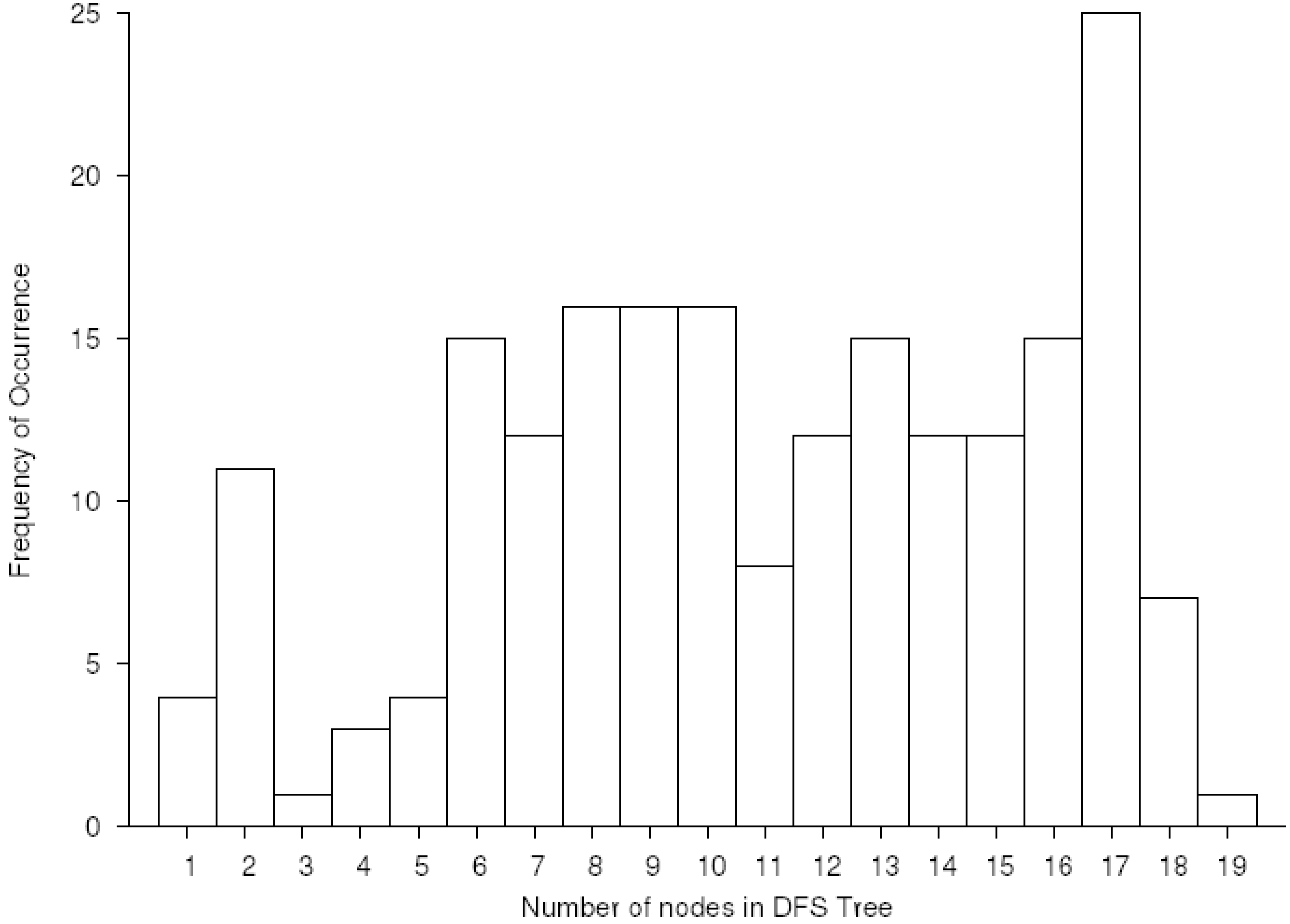


Lower value → less congestion & better overall success rate?

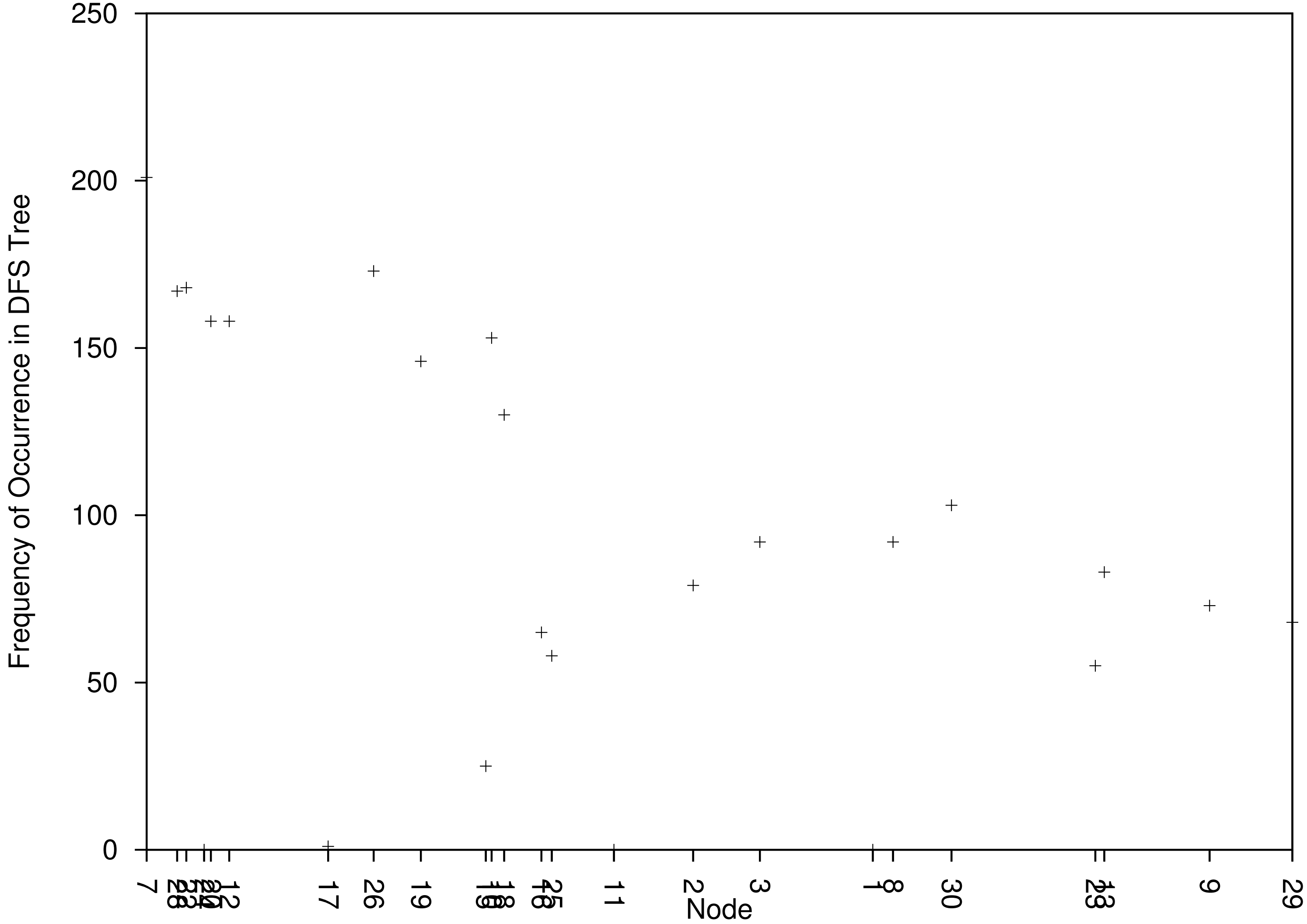
Depth-First Algorithm

- Sink started distributed depth-first search 317 times in 2 weeks (~22.5 per day)
- Links were chosen in descending quality order
- Search successfully terminated in 205 of these cases
- Successful depth-first does not visit all nodes of network
- Largest tree: 19 nodes (~ 90% of active nodes)
- More than 50% of all successfully built trees included more than 50% of nodes

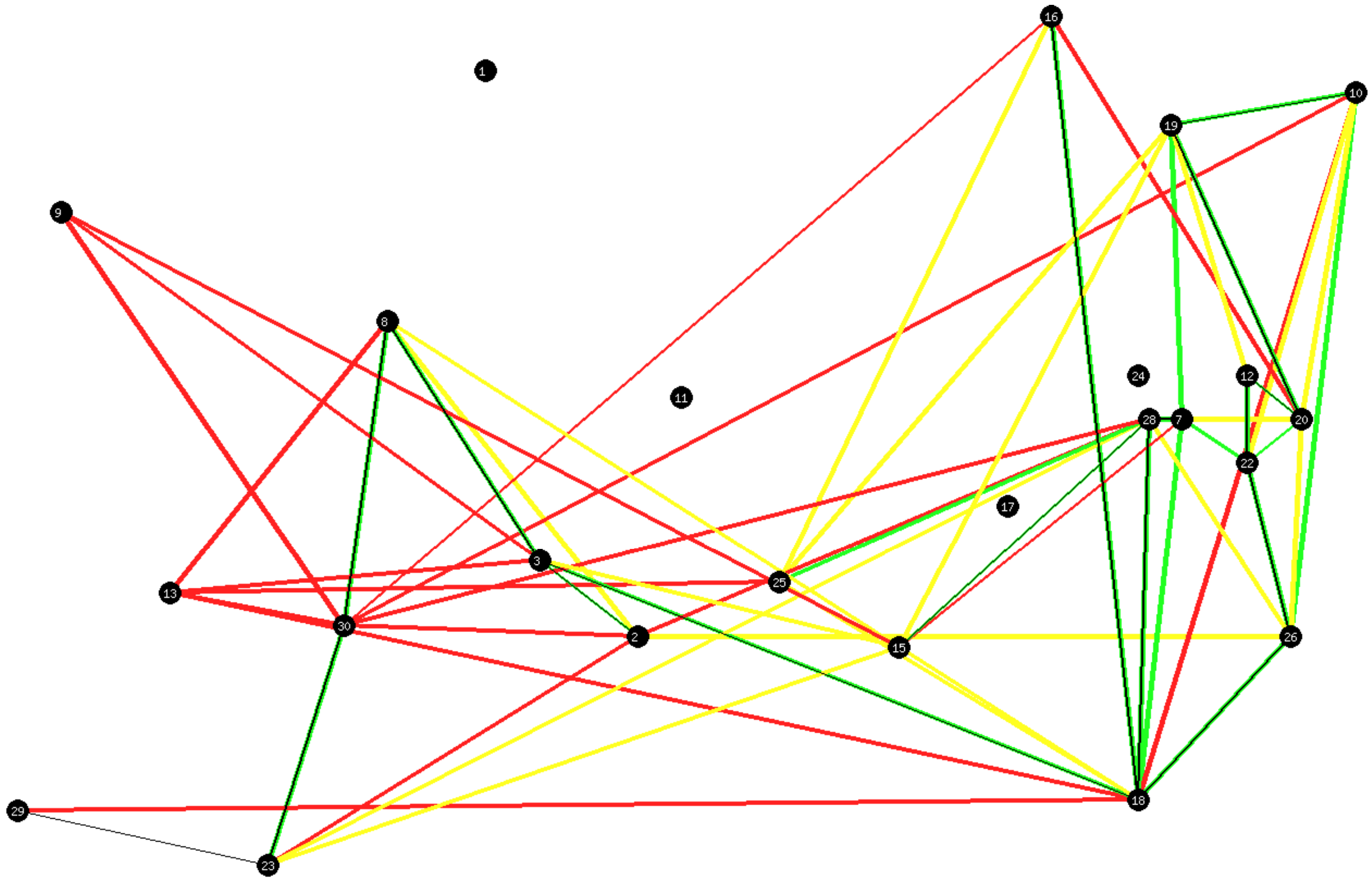
Depth-First Algorithm



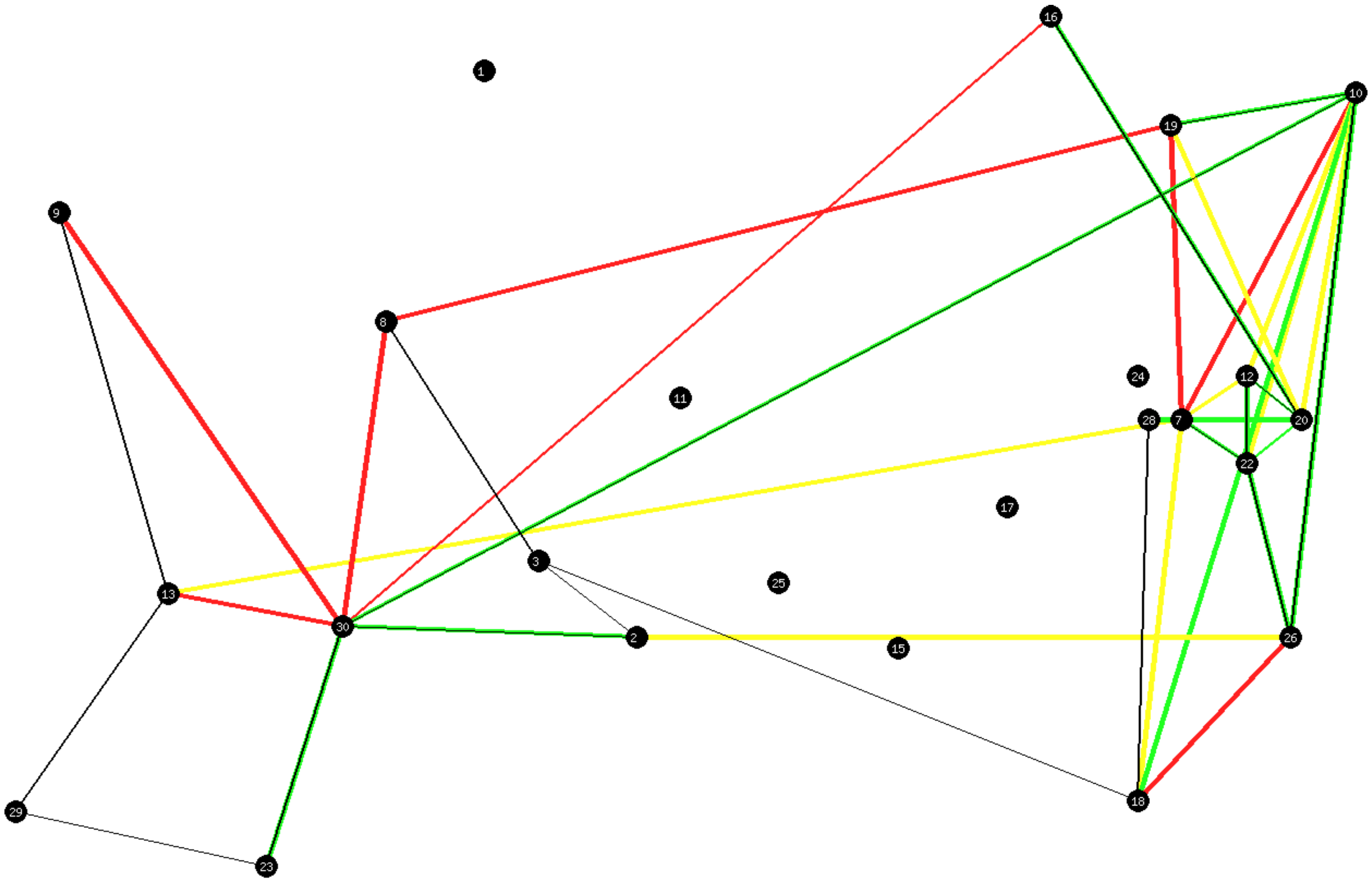
Occurrences of Nodes in DFS Trees



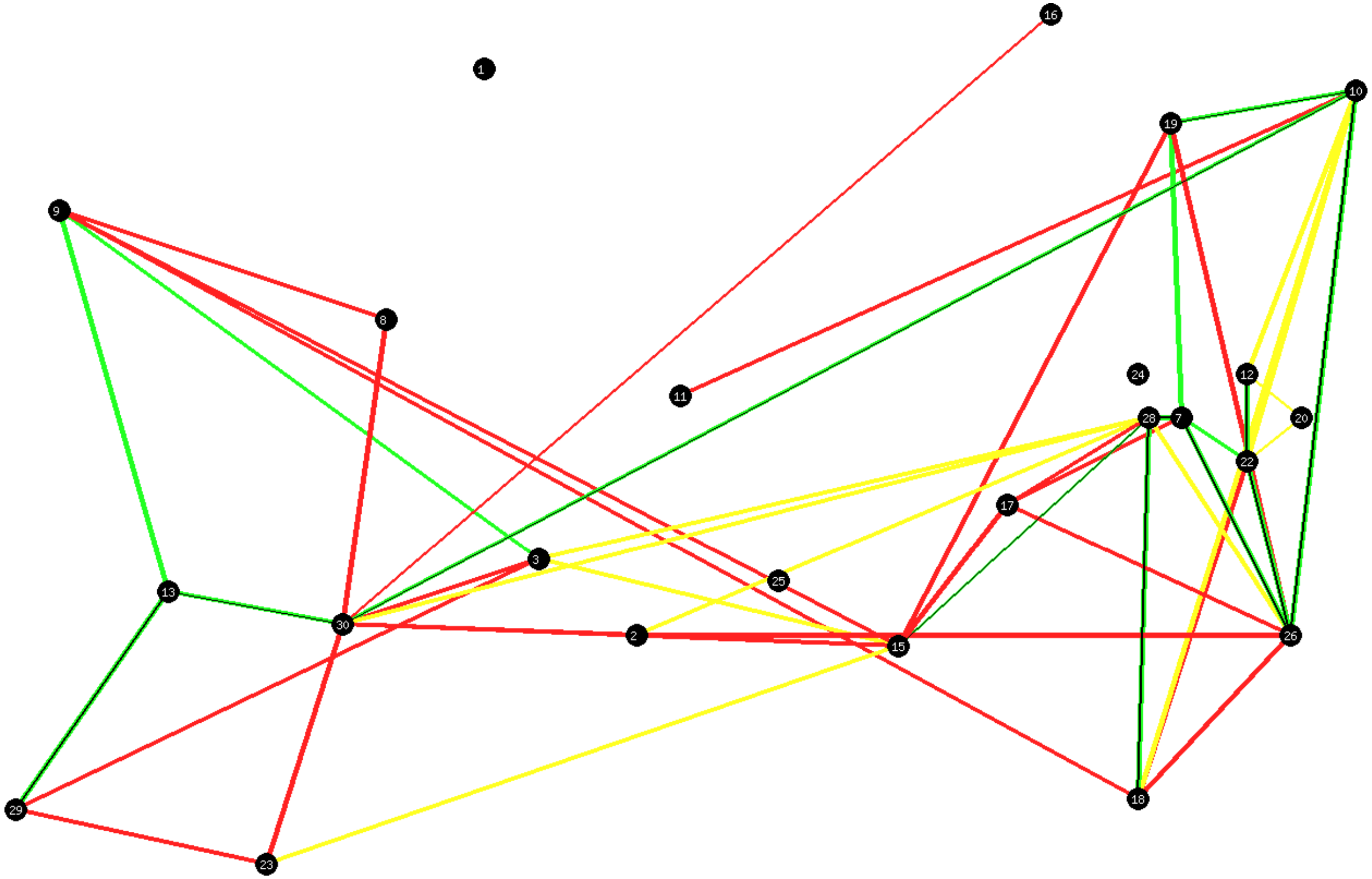
Changing Link Quality



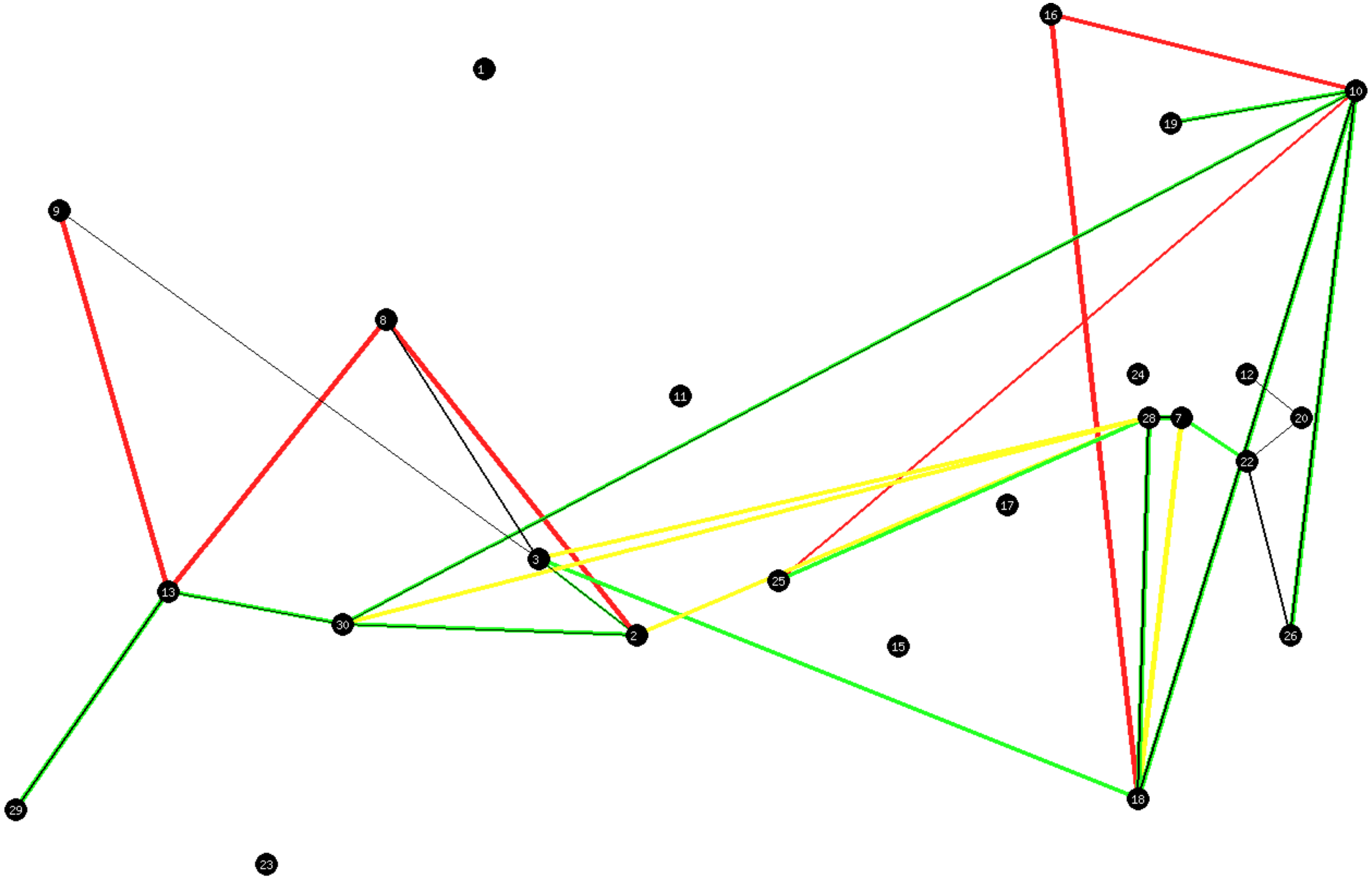
Changing Link Quality



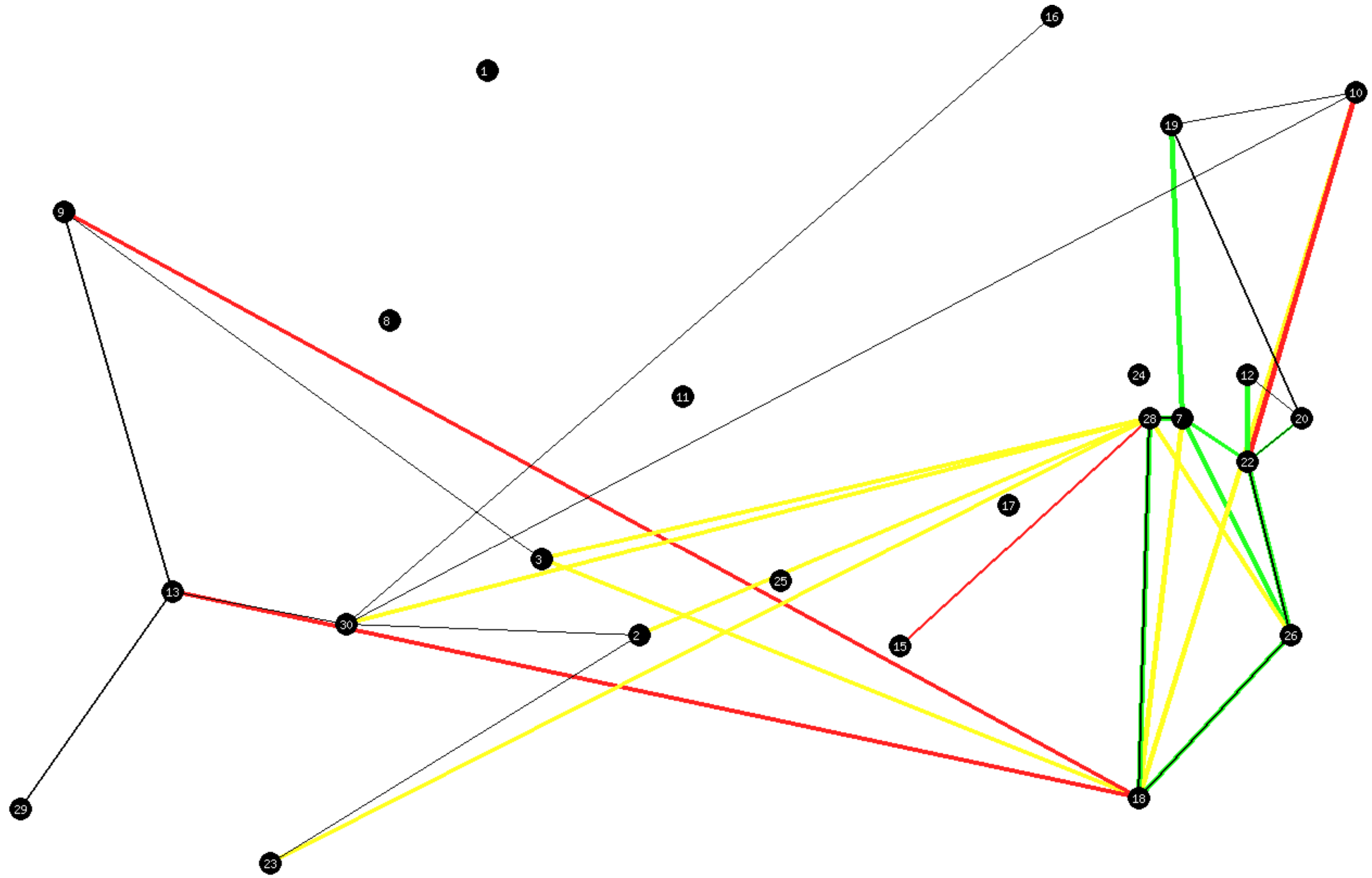
Changing Link Quality



Changing Link Quality



Changing Link Quality



Depth First Trees

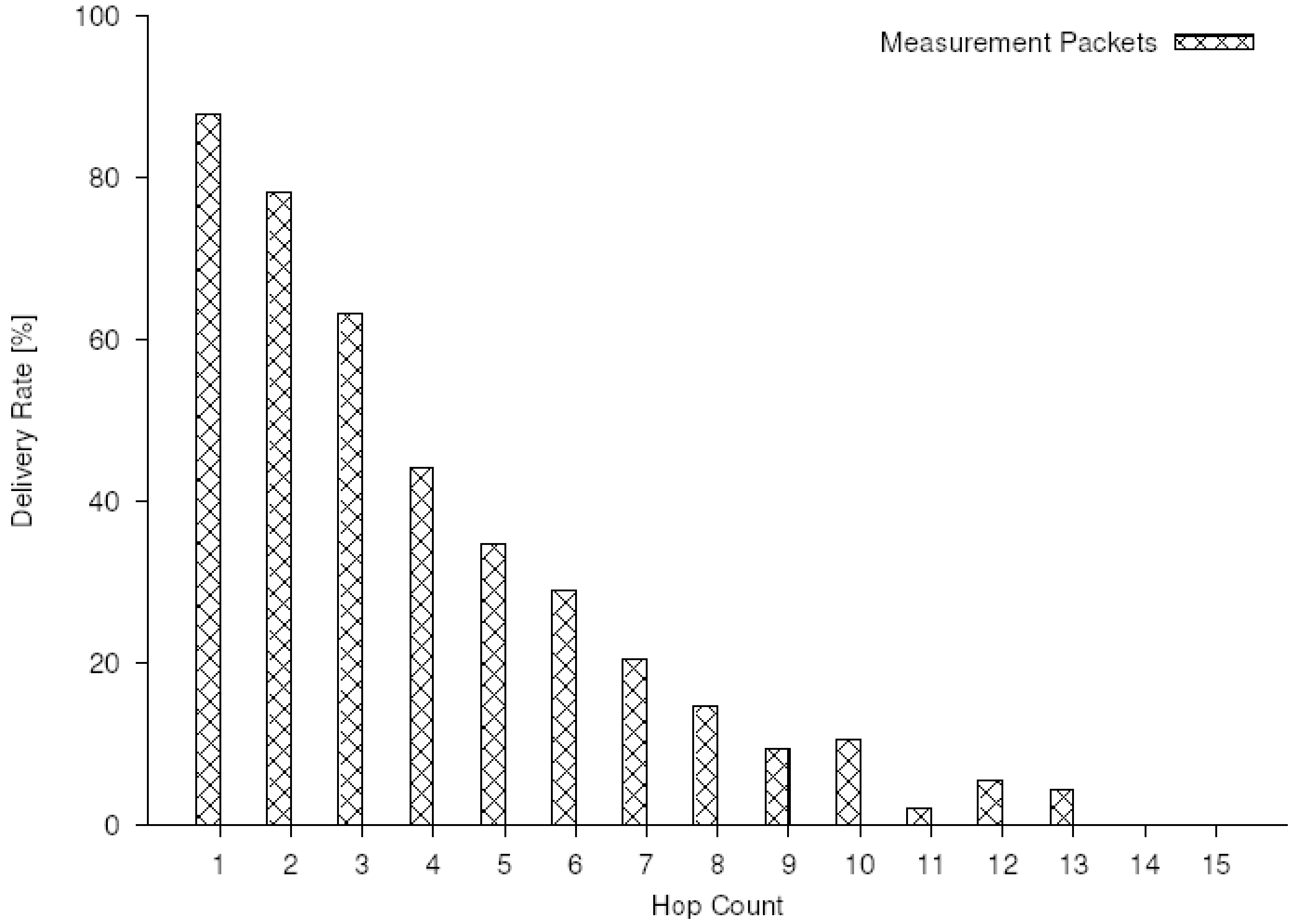
List of each node's successors is relatively stable over time

Node \ Nbs.	26	12	20	22	18	10	19	16	30
22	88	70	98						
26				55	7	72			
20		86		15			37	37	
10	41				26		82		71

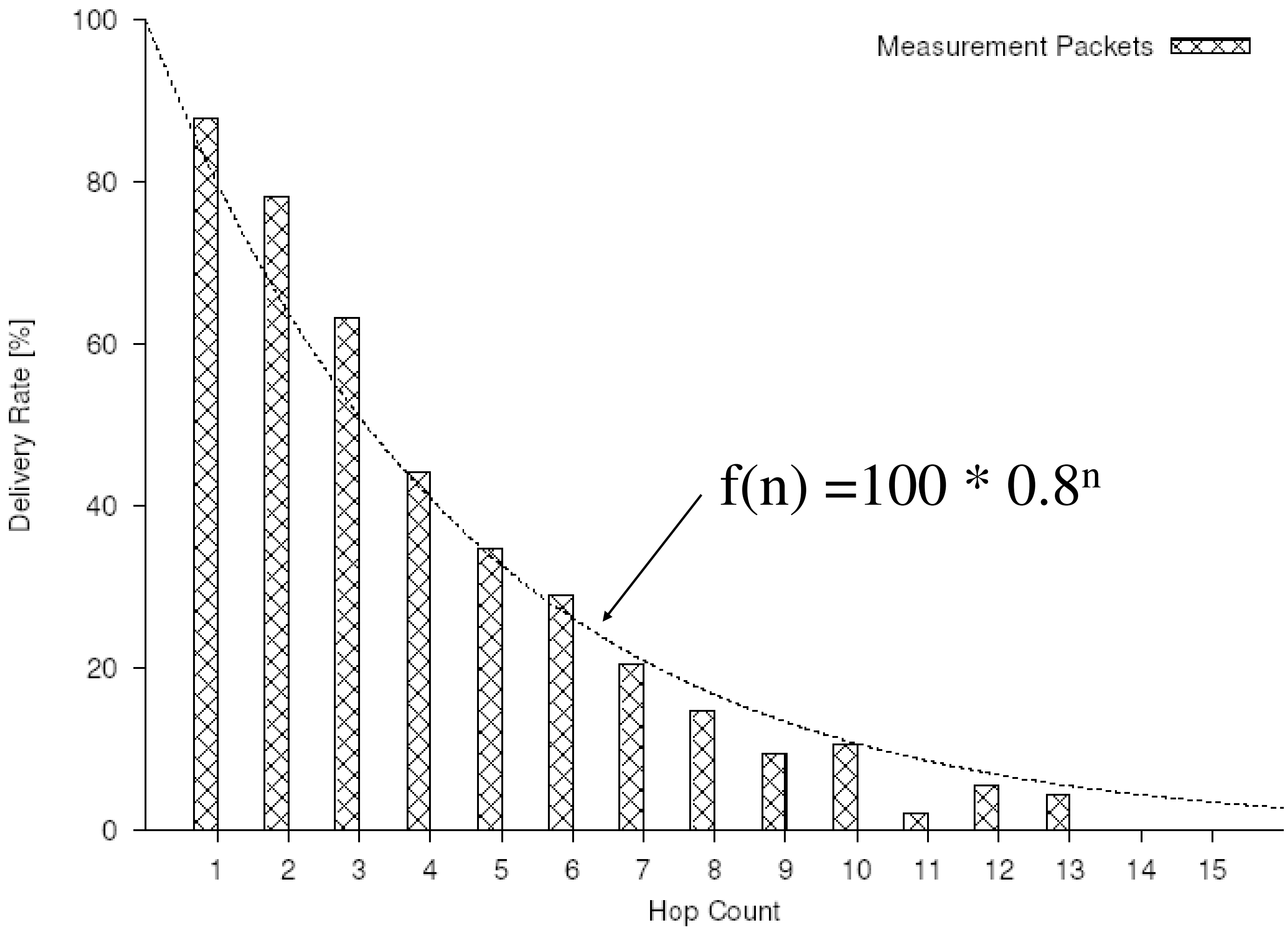
Routing

- DFS tree using for routing towards sink
- Success rate drops approximately exponentially with the depth d
- Transmission success of neighborhood packets exhibits a different run of the curve
- Size of a measurement packet ~ 70 bytes including packet header
- Neighborhood list packets 10% larger
- But significantly lower success rate

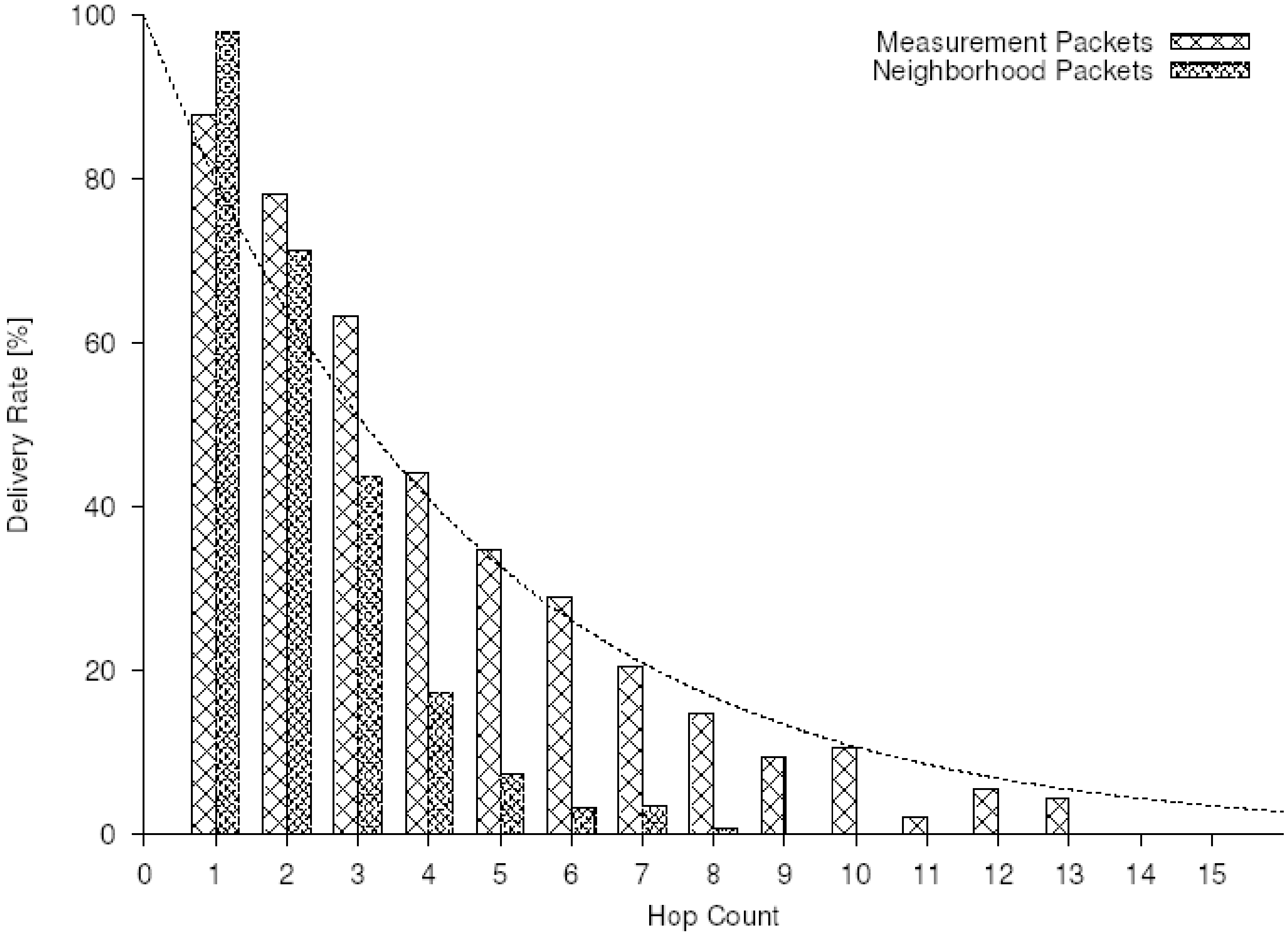
Success Rate



Success Rate



Success Rate



Reason

- Main reason:
  - probably not larger packet size
  - but is related to congestion
- Neighborhood packets were sent t seconds after measurement packets, t randomly chosen between 0 and15
- Due to retransmission of packets this time difference became very small after a few hops
- Reason for lower success rates of neighborhood packets for higher hop counts

Conclusion

- Lessons learned
  - Timing of actions very important to avoid congestion
  - Retransmission policy
  - Application has to deal with duplicate packets (lost ACKs)
  - High energy consumption for real time clock access
- We faced problems with the firmware implementation (e.g. reset of clock)
- Software installation procedure does not scale
- Unit disc graph model not appropriate
- Depth-first search implementation is robust and fault-tolerant
- Real deployments are a must to evaluate algorithms
- Careful planning: What data is needed, where and when to store it