Electromagnetic communication in nanonetworks (computer science challenges of the miniaturisation in minisensor networks)

Eugen Dedu

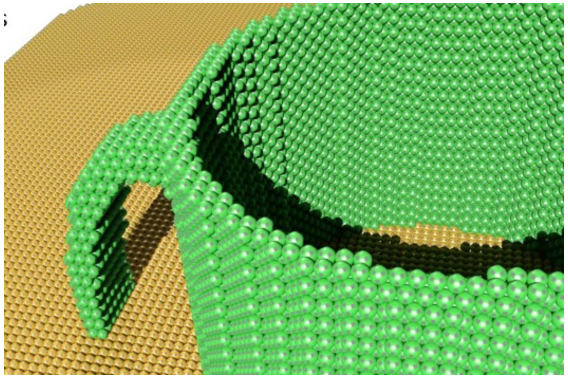
Maître de conférences HDR
eugen.dedu@univ-fcomte.fr
http://eugen.dedu.free.fr

IoT master, 2nd year
Univ. Franche-Comté
Montbéliard, France
2023–2024

Context: making modular robots communicate

- Modular robots = an ensemble of tiny robots, usually connected each other
- Existing modular robots:
  - sliding cubes, 1 cm side
  - rotating/jumping cubes, 5 cm side
  - rotating cylinders, 1 mm diameter, 5 mm long
  - 3D catoms, 3.6 cm radius
  - ...
- Flooding (broadcast to all the network)
  - with or without ack
- Questions answered by the current course:
  - what technology can be used? Nanonetworks
  - what protocols can be used? Backoff flooding, density estimator etc.

Dense networks



- Cup: 5300 nodes in mono hop
- Dense networks = nodes have so numerous neighbours (hundreds, thousands, ...) that classical protocols are inefficient
  - Wi-Fi is inefficient when tens of neighbours communicate simultaneously: slots, contention window, backoff
  - the same for RFID tag counting
  - nanonetworks
- More and more appliances and users connect to networks (IoT), therefore we consider that dense networks will not be the exception anymore, but the norm

Data communication types

- Unicast
- Multicast
- Zone-cast
- Broadcast (flooding): pure, probabilistic, adaptive
  - need to discover number of neighbours
- ... (functionality-based, ...)

Planning

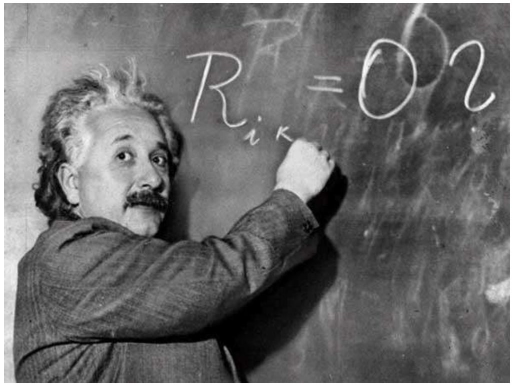
- 6h lectures
- 6h labs, using BitSimulator on GNU/Linux or macOS
- Exam (not yet sure): ~30', written (a few questions to answer in detail)

Summary

- Context: our nanonetwork group
- 1. Introduction to nanonetworks
- 2. Simulation tools (BitSimulator)
- 3. Broadcasting: backoff flooding
- 4. Estimating number of neighbours (DEDeN)
- 5. Deviation-based routing
- 6. Duty cycling
- Conclusion

FEMTO-ST institute

- 760 members: 260 researchers and teachers-researchers, 80 ITA/BIATSS + 20 associates, 280 own funding (mainly PhD students), 120 external (internship students, invited researchers etc.) (12/2022)
- 7 scientific departments in science and technology:
  - AS2M – automatic control and mechatronics micro systems
  - DISC – computer science
  - Energy
  - Applied mechanics
  - MN2S – micro nano sciences and systems
  - Optics
  - Time & Frequency



- Lead by CODIR (Direction Council): the director, the deputy directors, and the directors of the 7 departments
- Locations: Besançon (80% of people), Belfort (15%), Montbéliard (5%)

DISC department (computer science)

- 57 teachers-researchers + 53 PhD students (01/2022)
- 4 teams:
  - AND (Belfort) –distributed numerical algorithmic
  - DEODIS (Besançon) – design and evaluation of distributed systems
  - OMNI (Montbéliard, Belfort) – optimization, mobility, network
  - VESONTIO (Besançon) – verification and validation of software and embedded systems

OMNI team members

- 18 T/R, 1 temporary T/R, 1 post-doc, 15 PhD students, 0.15 engineer, 0.15 secretary
- 2018:
  - 11 journals, 17 conferences (1 CORE A*, 5 CORE A)
  - conference organisation: steering committee, workshop chair
  - 2 keynotes in international conferences
  - national and international funding


NUMERICA
PÔLE NUMÉRIQUE
DE BOURGOGNE
FRANCHE-COMTÉ

OMNI team – research axis

- Programmable matter:
  - distributed algorithmics for modular µrobots
  - communication in electromagnetic nanonetworks (nanonetwork group)
- Transport and mobility
  - people, goods and data transport
  - mobility models
  - geopositioning
  - operational research (optimisation)

Nanonetwork group

Current members:

Carole Al Mawla
PhD student
routing

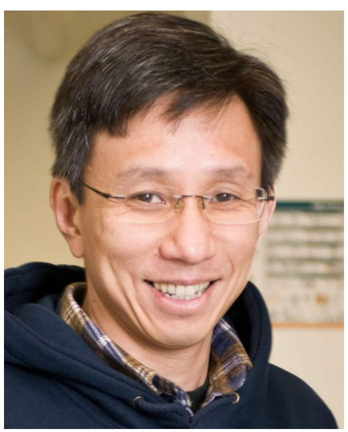
Farah Hoteit
PhD student
routing

Ali Medlej
PhD student
routing

Eugen Dedu
Assoc. pr.

Dominique Dhoutaut
Assoc. pr.

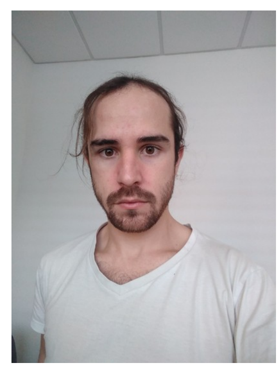
PhD thesis co-supervisors:



Winston Seah
Professor, New Z.

Kamal Beydoun
Assoc. pr., Lebanon

Past members:



Thierry Arrabal
PhD student

Muhammad Agus Zainuddin
PhD student
coding

Yacine Benchaïb
ATER
duty cycling

Publications of the nano group

- F. Hoteit, E. Dedu, W. Seah, D. Dhoutaut. Ring-based forwarder selection to improve packet delivery in ultra-dense networks. WCNC, Austin, TX, USA, 2022.
- A. Medlej, K. Beydoun, E. Dedu, D. Dhoutaut. Scaling up Routing in Nanonetworks with Asynchronous Node Sleeping. SoftCOM, Hvar, Croatia, 2020.
- T. Arrabal, F. Büther, D. Dhoutaut, E. Dedu. Congestion Control by Deviation Routing in Electromagnetic Nanonetworks. ACM NanoCom, Dublin, Ireland, 2019.
- T. Arrabal, D. Dhoutaut, E. Dedu. Efficient multi-hop broadcasting in dense nanonetworks. NCA, Cambridge, MA, USA, 2018.
- D. Dhoutaut, T. Arrabal, E. Dedu. BitSimulator, an electromagnetic nanonetworks simulator. NanoCom, Reykjavik, Iceland, 2018.
- T. Arrabal, D. Dhoutaut, E. Dedu. Efficient Density Estimation Algorithm for Ultra Dense Wireless Networks. ICCCN, Hangzhou, China, 2018.
- A. Tsioliaridou, C. Liaskos, E. Dedu, S. Ioannidis. Packet routing in 3D nanonetworks: A lightweight, linear-path scheme. Nano Communication Networks, 2017.
- M. A. Zainuddin, E. Dedu, J. Bourgeois. Simple and energy efficient image compression for pulse-based communication in THz band. AINA, Taipei, Taiwan, 2017.
- M. A. Zainuddin, E. Dedu, J. Bourgeois. SBN: Simple Block Nanocode for nanocommunications. ACM Nanocom, New York City, NY, USA, 2016.
- A. Tsioliaridou, C. Liaskos, E. Dedu, S. Ioannidis. Stateless Linear-path Routing for 3D Nanonetworks. ACM Nanocom, New York City, NY, USA, 2016.
- M. A. Zainuddin, E. Dedu, J. Bourgeois. Low-Weight Code Comparison for Electromagnetic Wireless Nanocommunication. IEEE Internet of Things Journal, 2016.
- M. A. Zainuddin, E. Dedu, J. Bourgeois. The Effects of Nanosensors Movements on Nanocommunications. ACM Nanocom, Boston, MA, USA, 2015.
- M. A. Zainuddin, E. Dedu, J. Bourgeois. Nanonetwork Minimum Energy coding. UIC, Bali, Indonesia, 2014.
- E. Dedu, J. Bourgeois, M. A. Zainuddin. A first study on video transmission over a nanowireless network. ACM Nanocom, Atlanta, GA, USA, 2014.

Collaborations in nanonetworks

- C. Liaskos, A. Tsioliaridou (FORTH institute, Greece), routing layer, 2 common publications
- F. Büther (Lübeck univ., Germany), routing and transport layers, 1 common publication, simulator co-developer
- W. Seah (Victoria univ., New Zealand), routing layer, PhD student's co-supervisor
- K. Beydoun (Lebanese univ., Lebanon), routing layer, PhD student's co-supervisor

1. Introduction to nanonetworks
nanomachine
nanonetwork
peculiarities
applications

Miniaturization race

IBM 704 (1964) mainframe

Arduino

?

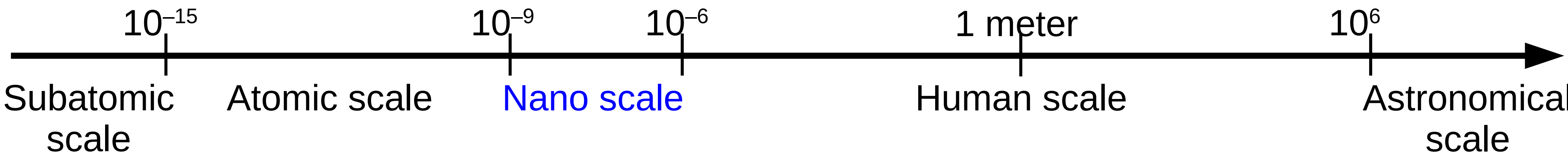
Next steps in component miniaturization: micro, nano

Advantages:

- reduced manufacturing cost
- reduced energy consumption
- reduced space

How to make our applications work in this context?

Nanometer scale



- Definition of a nanothing: whose size is 1..1000 nm (< 1 µm)
  - not to be confounded with "nanocomputer" (whose name comes from smaller than minicomputer, whose fundamental parts are smaller than a few nm), whose size is comparable to a credit card (Arduino, Raspberry Pi etc.)
- Examples of things at nano scale:
  - molecular field: typical carbon-carbon bond lengths (spacing between these atoms in a molecule) is 0.12–0.15 nm
  - biology: DNA double helix diameter is 2 nm, smallest cellular life-form is 200 nm in length
  - electrical circuits: current technology for µP is ~8 nm

Nanomachine, some current values

- Our vision is to use nanomachines
- Smallest computers: 1 mm² (IBM, 03/2018), .3*.3 mm² (CMU, 06/2018)
- CPU:
- Memory: areal density of magnetic discs is 1.34 Tb/in² (as of 2015, Samsung, cf. wikipedia areal_density) ~= 256 B/µm²
  - buffer in/out: we need 1–2 kB (?)
- Sensor/actuator:
- Energy harvester:
- No battery, but capacitors
- Antenna: see later slide
- Density of nodes: 1 mm³ contains 1 million fully packed machines of 10x10x10 µm³, and 1 cm³ contains 1 billion (milliard)

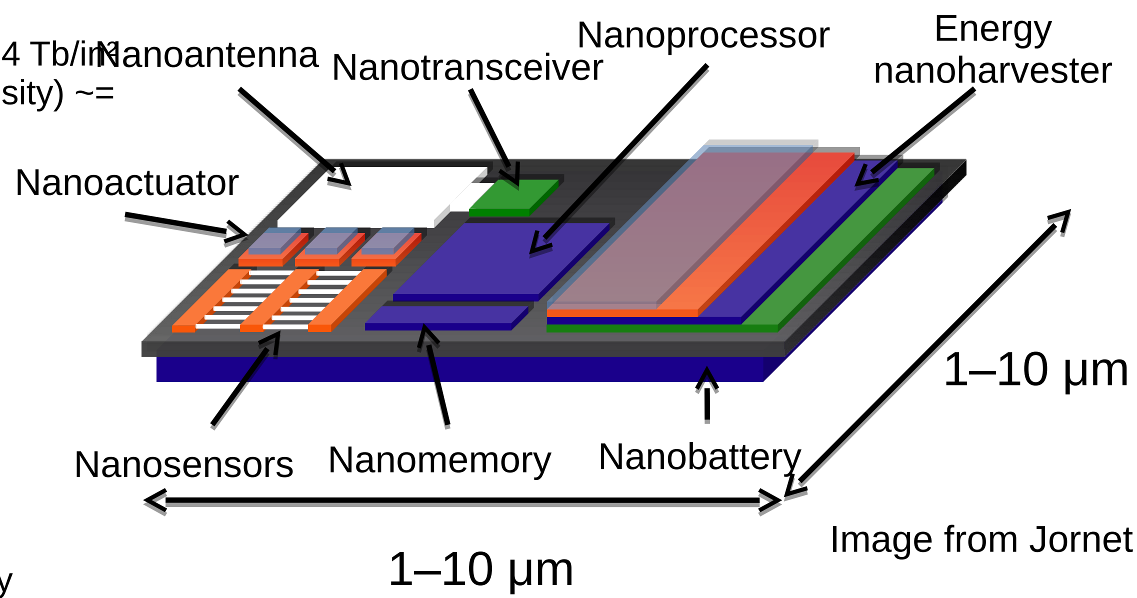


Image from Jornet

Nanodevices designed based on current and feasible technologies have sufficient capability to run ad-hoc communication protocols and perform simple tasks

Positioning – nanoscience, nanotechnology

When people hear nano they think at:

- Nanoscience: study of phenomena at nanoscale, where materials can show different properties compared to macroscale (from w nanotechnology):
  - opaque substances can become transparent (copper)
  - stable materials can turn combustible (aluminium)
  - insoluble materials may become soluble (gold)
  - in this context, nano means between 0.2 (atomic level) and 100 nm (in this range materials exhibit different properties)
- Nanotechnology: how to exploit these new phenomena to create nanothings
  - often, nanotechnology word includes also nanoscience
  - National Nanotechnology Initiative, USA (government program for nanoscale projects) considers as nanotechnology the manipulation of matter that has at least one dimension in 1..100 nm range, since at <100 nm quantum effects become important (w nanotechnology)
- This is not our topic

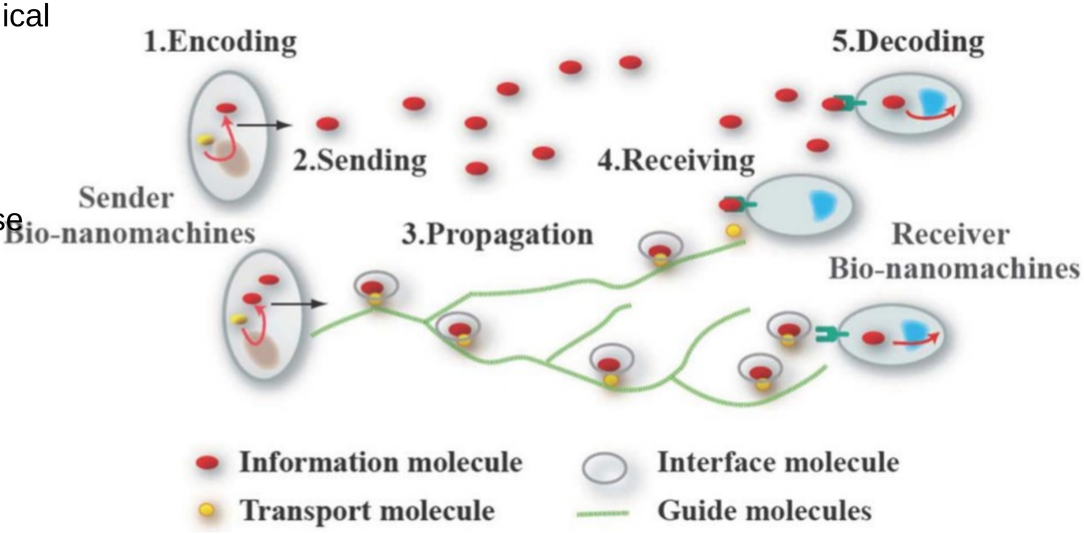
Definition of a nanonetwork

- A nanonode alone is of little interest => let's network them
- For us, following Jornet, nanonetwork means network of nanodevices
- Networking technologies:
  - wired, uses a physical wire: electrical cable (e.g. Ethernet), fibre optic etc.
  - wireless: electromagnetic (radio waves) – in-body, free space, on-chip etc.
  - others: molecular (molecules as data carriers in the human body, restricted to biological environments) etc.

Molecular nanonetworks (communication)

2010 Nakano et al., Molecular communication and networking: Opportunities and challenges

- What it is
  - engineered nanomachines communicate with biological systems
  - new communication paradigm: uses molecules to convey information
  - sender encodes information in molecules and release them in the environment, and receiver decodes the information upon reception
- "A single CNT (1 nm diameter) is small enough to penetrate a cell without triggering the cell's defensive responses" (Bush's book)
- Applications envisioned:
  - health monitoring, drug delivery
  - environment monitoring (toxic molecules)
  - create novel patterns of molecules
- Very very slow



Communication	Telecommunication	Molecular Communication
Devices	Electronic devices	Bio-nanomachines
Signal types	Optical/electrical	Chemical
Propagation speed	Speed of light	Extremely slow
Propagation range	m - km	nm - μm
Media	Air/cables	Aqueous

Electromagnetic nanonetworks (communication)

- Uses EM waves to transmit data
- Pioneer: J. Jornet, who proposed:
  - physical layer protocol
  - MAC (collision avoidance) protocol
- Relatively young research field (since 2012)

Peculiarities

- Nanonodes have not yet been built, given technological challenges at this tiny size => simulation
- Nanonetwork = wireless network of nanonodes, multi-hop possible
  - small communication range: ~cm => need multi-hop for longer comm distances
- Peculiarities:
  - very tight energy budget (and high frequency): no signal carrier, but pulses and silence => TS-OOK modulation
  - tiny size: waiting time between pulses => packet interleaving (temporal multiplexing)
  - tiny size: ultra-dense networks (thousands–millions of neighbours possible)
  - tiny size: small antennas, high frequency (0.1–10 THz), big signal attenuation, Tb/s
  - harvesting is an option given the small energy consumption
  - limitations: nodes have limited resources (computation, power, memory, …), e.g. no floating point

UWB, ultra-wideband

- IEEE 802.15.4 2020 defines LR-WPAN (low rate wireless personal area network)
- UWB is a technology for transmitting information across a wide bandwidth (>500 MHz) or >20% of the centre frequency (for a signal at –10 dB)
- Pulse-based systems are UWB ?

Peculiarity: pulse-based communication

Information from 2014 Jornet et al., Femtosecond-Long Pulse-Based Modulation...

- Because of tiny size, energy is very scarce in nanonodes => cannot use carriers to transmit data => pulses
  - OOK technologies: IR/UWB (impulse radio or ultra wide band), FSO (free space optical)
- Technology limitation => pulses cannot be sent in burst, need relaxation time
- Proposition: TS-OOK, Time Spread On-Off Keying modulation, i.e. pulse or silence at big intervals
- Characteristics:
  - T_p = 100 fs-long pulses
  - T_s time between two symbols,

    proposed $\beta = T_s/T_p = 1000$
    - β can vary (time hopping)
  - if $\beta = 10$, then rate = 1 Tb/s
  - pulse energy = a few aJ, peak power = a few µW
  - need extremely high synchronisation between

    src and dest nodes
- Model validated using COMSOL Multiphysics

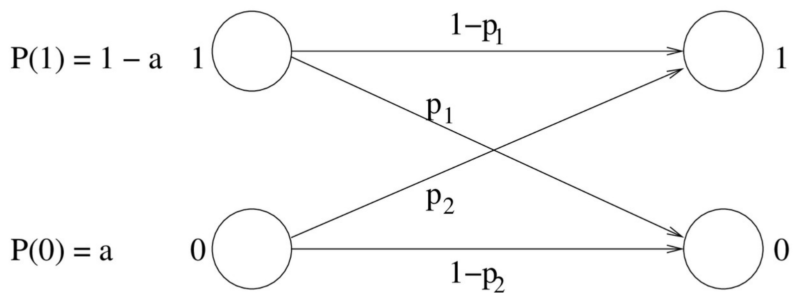
On sender:

```
Signal:   /\____/\____.____/\____
Bit sent: 1     1     0    1
```

Signal on receiver:

```
Expected:   /\____/\____.____/\____
```

Nanonetworks use BAC channel model:

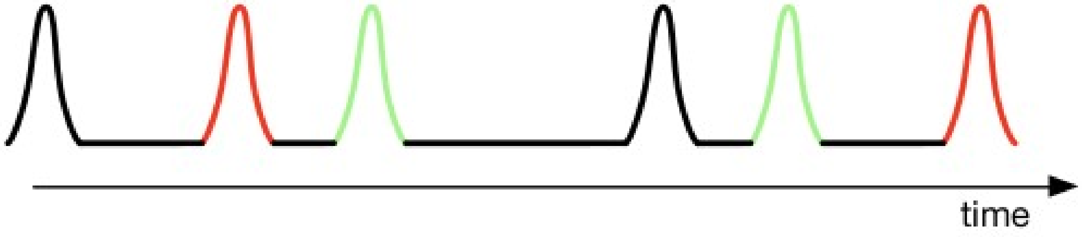


TS-OOK, Time-Spread On-Off Keying

- Bits are coded as the **presence** or the **absence** of a very short (e.g. ~100 femtoseconds long) **radio pulse**. A tight synchronization is required. A pulse means "1" and its absence means "0".

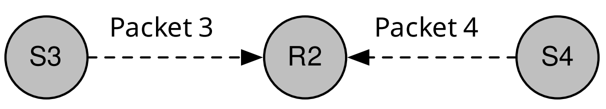
- Bits from a given frame are separated by a fixed interval, much longer than the duration of a pulse. The ratio between pulse duration and interval between pulses is called β. Typical values of β is usually high (1000 or even more).

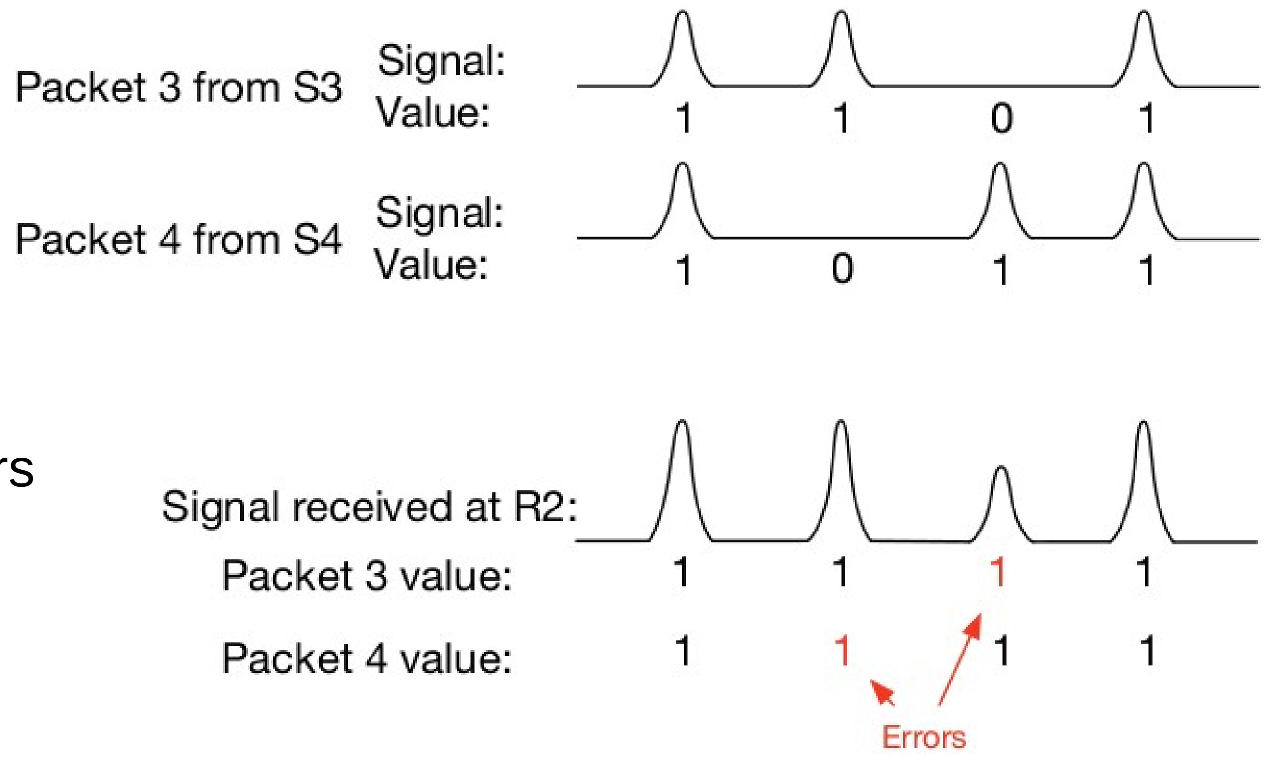
- This high spreading ratio allows frames from different communications to overlap



Collisions

- Collisions appear when several bits from different frames arrive at a receiver at (almost) the same time

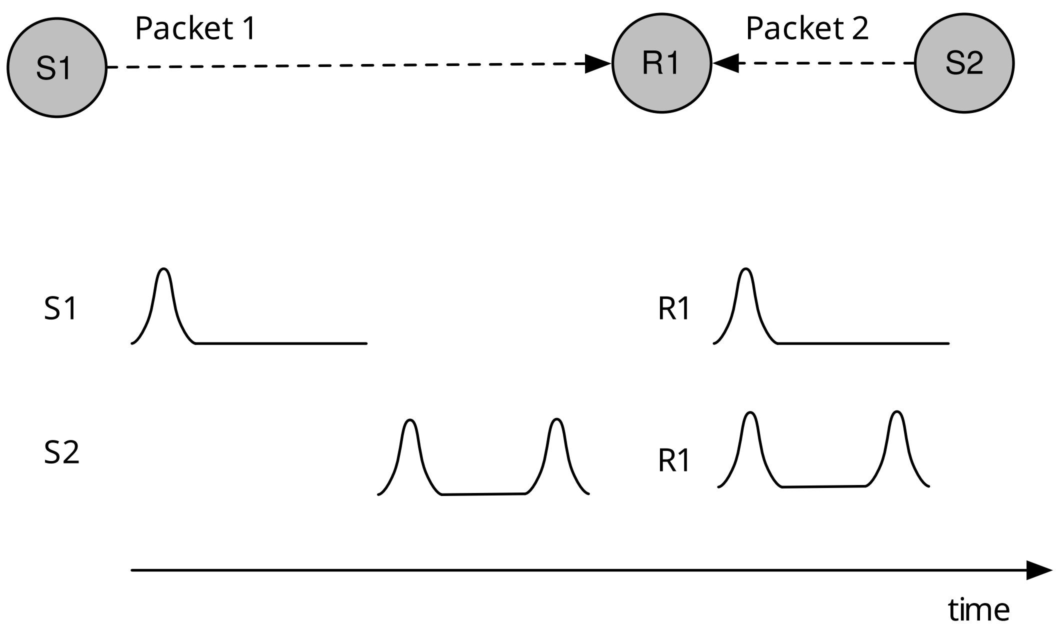




- Not all collisions lead to errors
- Here, the only error occurs when the 1 bit from S3 hides the 0 bit (silence) from S4 at the receiver R2

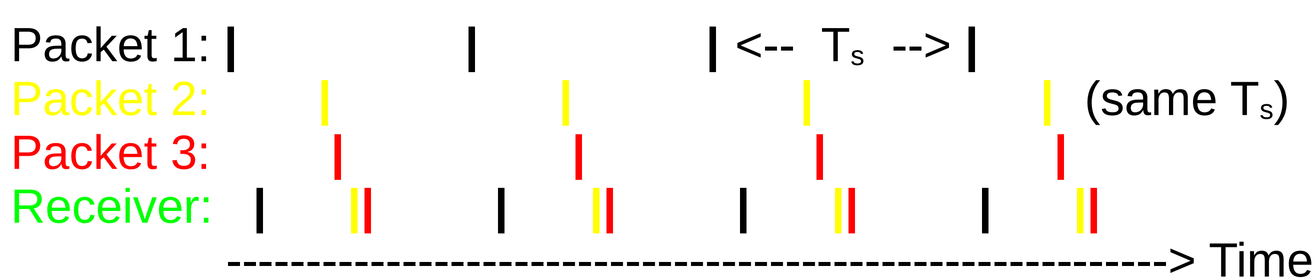
Propagation delay

- Pulse duration is so small that even for very short distances, propagation delay cannot be neglected
- Node positions influence the reception date and consequently the occurrence of collisions



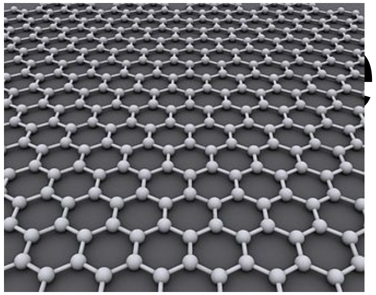
Peculiarity: packet overlapping (bit switching communication)

- Overlapping level:
  - circuit switching – no overlap
  - packet switching – communications overlap – IP
  - bit switching – packets overlap – TS-OOK
  - ... – bits overlap – collisions (coding, error correction codes?)
- Multiplexing of frames over time on the channel can go be very high (potentially hundreds or more frames being sent in parallel
- This is very specific to pulse-based nanonetworks (in other networks, frames of different origin are sent sequentially)
- Number of concurrent frames can go well beyond the capability of a node to track and decode all of them (more on this later)



Peculiarity: dense network

- Multi-hop wireless networks
- Dense networks ®: so many neighbours (e.g. thousands) that the classical protocols handle common tasks inefficiently, or even do not work anymore
  - Wi-Fi DCF channel access method does not work anymore for >100 nodes communicating simultaneously
  - RFID tag counting for numerous tags
- High density: one of the properties of nanonetworks we specifically take into account
- We believe this scenario will be common in the future:
  - 5G mMTC (massive Machine Type Communications)
  - long range technologies with classical machines (e.g. LoRa)
  - short range technologies with small machines (nanonetworks)
- Related definitions:
  - dense communications ®: numerous concurrent communications
  - large network ?? ®: numerous hops



eculiarity: high frequency – nanoantenna

201 Jornet et al, Graphene-based plasmonic nano-transceiver for terahertz band...

- What is the resonating frequency of a classical nanoantenna?
  - following classical antenna theory, the size of a conventional (metallic) antenna resonating at wavelength λ should be at least λ/2
  - an antenna of 1 µm (max size for a nanoantenna) can process signals of max 2 µm waveleng

$$f = c/\lambda = \frac{3 \times 10^8 m/s}{2 \times 10^{-6} m} = 1.5 \times 10^{14}\ \text{Hz} = 150\ \text{THz}$$

  - frequency >150 THz => big propagation loss, even classical antenna theory needs to be revised
- A method to reduce nanoantenna's frequency is to use graphene [Jornet], a one atom-thick layer of carbon, which would allow 100 times smaller frequencies for the same size => irradiating in the THz band (0.1–10 THz)
  - graphene, isolated in 2004 by Geim and Novoselov (Nobel prize in 2010) has remarkable characteristics, such as 100 times stronger than steel and conducts very efficiently electricity

Spectrum background

Optical wireless comm.
- UV
- visible light comm.: Li-Fi
- IR

THz (0.1–10 THz, far&thermal IR)
- nano EM comm.

RF (radio frequency), 3 kHz–300 GHz:
- Wi-Fi, ISM (2.4 GHz)
- cellular network 3G (UMTS, ...), 1.9 GHz
- Zigbee, ISM (2.4 GHz)
- Bluetooth, ISM

(ISM = industrial, scientific and medical bands)

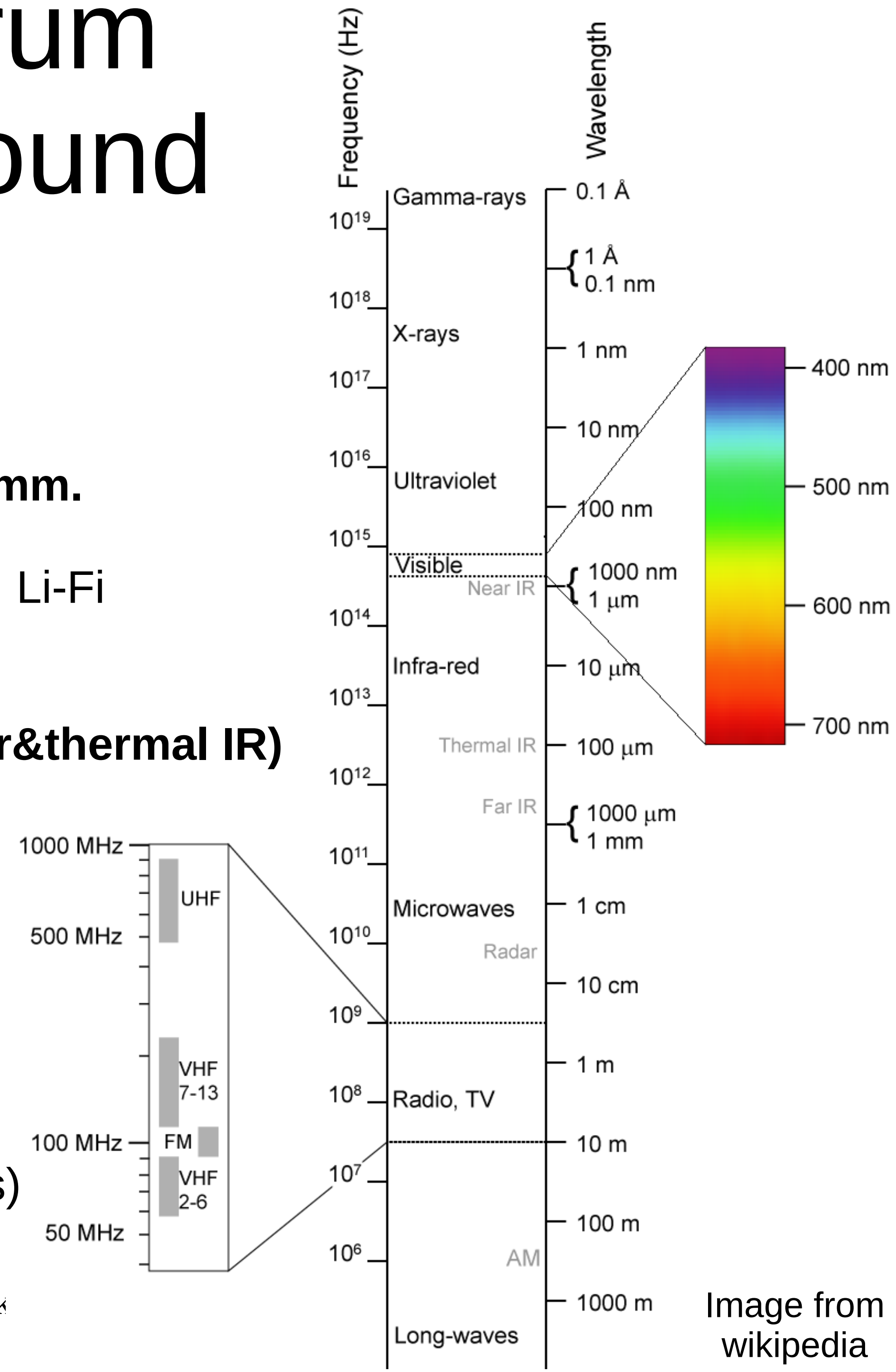
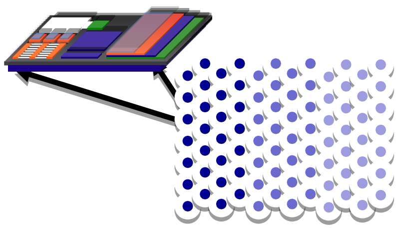


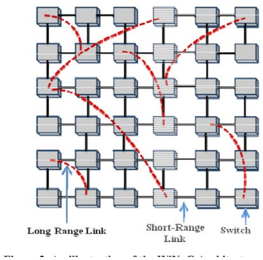
Image from wikipedia

Applications

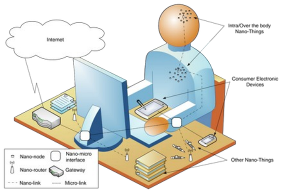
Catom



- Programmable matter: each nanomachine can play the role of a fundamental unit, e.g. a µrobot, or a catom in claytronics
- Wireless network-on-chip (WNoC) in multi-core processors:
  - long range links
  - cache coherence

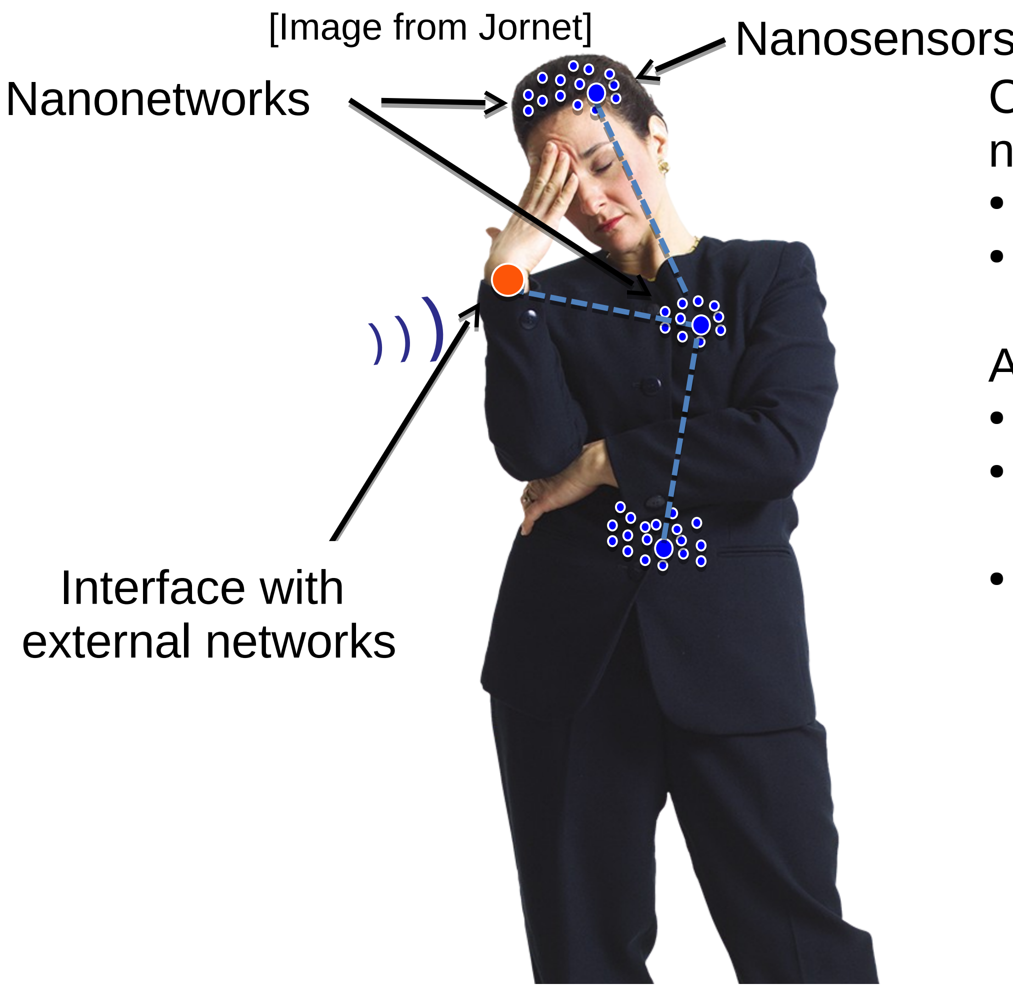


- Internet of nano-things, wireless LAN
- High-speed wireless links in data center: [4] B. Peng and T. Kürner, “A stochastic channel model for future wireless thz data centers,” in Wireless Communication Systems (ISWCS), 2015 International Symposium on. IEEE, 2015, pp. 741–745



- Wireless high-definition TV transmissions
- In agriculture, plant monitoring and plague defeating systems (Alan Davy)
- Software-defined metamaterials (Christos Liaskos)
- Chemical defense: detect biological and chemical hazards, faster and in lower concentrations, than existing microsensors

Applications: enhanced medicine



Chemical and biological nanosensors, nanocameras feature:

- high sensitivity and low detection limits
- low power consumption

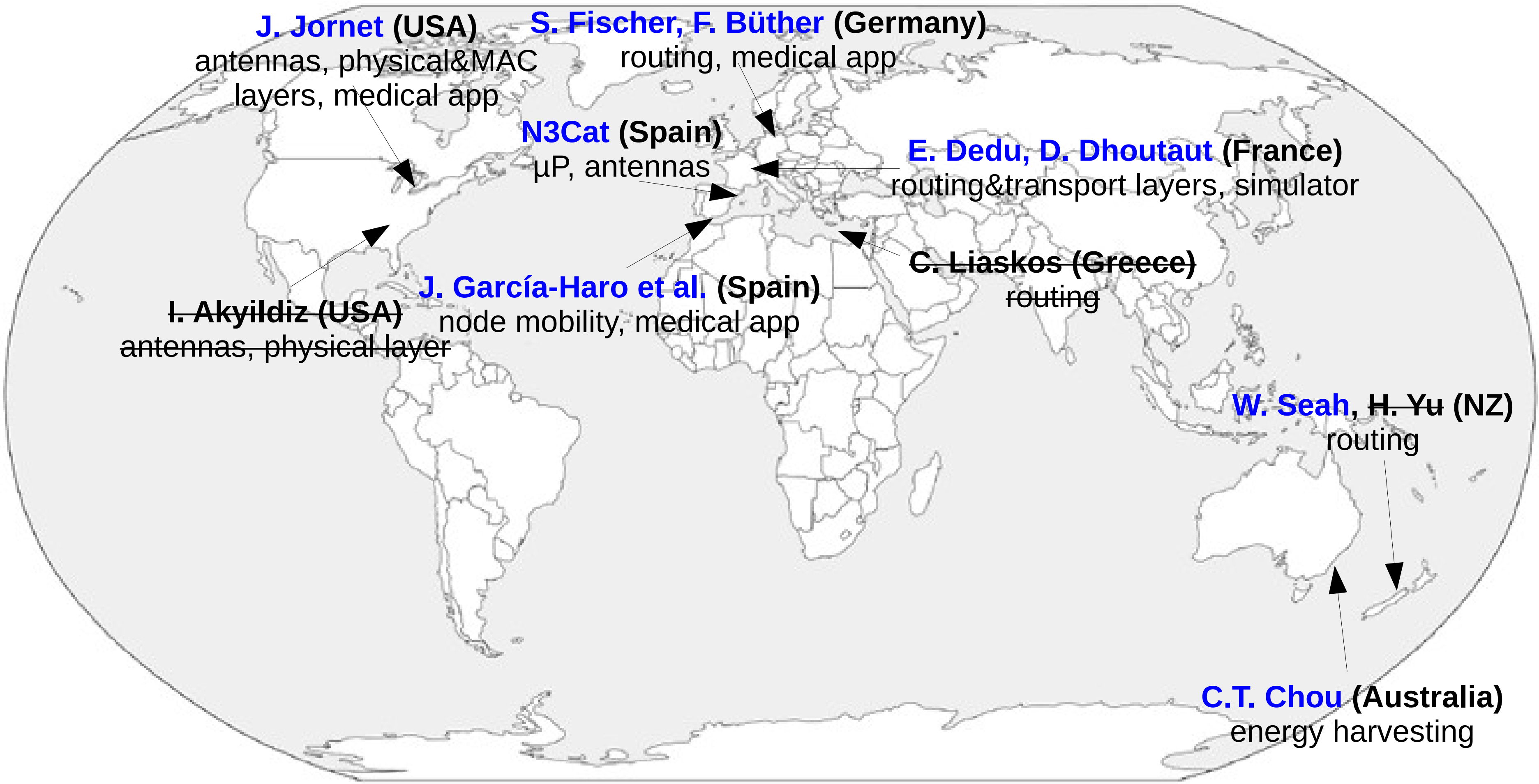
Applications:

- monitor glucose, sodium, cholesterol
- detect infectious agents in small concentrations
- localise and kill cancerous cells
  - colorectal cancer 5 year survival rate: 90% if localised, 68% if regional, 10% if distant metastases
  - they can cross barriers such as blood-brain and gastrointestinal ones

Journals, conferences

- Journals
  - Elsevier Nano Communication Networks (1st number in 2010)
  - but also: IEEE Internet of Things Journal (1st number in 2014)
- Conferences
  - ACM NanoCom (1st edition in 2014)
- Awards:
  - LCN 2017, best paper (NZ): *Pulse Arrival Scheduling for Nanonetworks Under Limited IoT Access Bandwidth*
  - ICT 2016 (Int. Conf. on Telecommunications):
    - 1 of 3 best papers *N3: Addressing and Routing in 3D Nanonetworks* (Greece)
    - 1 of 3 keynotes: *Molecular communication for future nanonetworks* (Germany)

People and labs close to this field



Bottom layers

MAC layer protocol – PHLAME

2012 Jornet et al. PHLAME: A Physical Layer Aware MAC Protocol...

- Rate Division Time Spread On-Off Keying (RD TS-OOK)
  - same as TS-OOK, but βs are different for different nanonodes and for different types of packets => collisions are temporary
- PHysical Layer Aware MAC protocol for EM nanonetworks (PHLAME)
  - handshaking request, packet contains: sync, src, dest, packet id, β, CRC
  - handshaking acknowledgement, packet is similar: sync, src, dest, packet id, coding scheme (according to perceived quality of received pulse), CRC
  - data, packet contains: src, dest, data

Low-weight coding

- Sending a 1 bit consumes energy, contrary to bit 0
- Goal: reducing the number of bits 1 transmitted
- Our proposition: NME (Nanonetwork Minimum Energy) coding
  - create a table of used symbols and their frequency
  - sort it in decreasing order of frequency and map high frequency symbols with low-weight codewords
  - => energy efficient for uncompressed data, but less robust (a 1-bit error leads to several bits in error)

Algorithm (Huffman-like):

```
Bits to send:                 Dict.:     Bits actually sent:
11 10 00 11 10 01 11   -> 11 3 00   -> 00 01 10 00 01 11 00
(9 bits 1)                10 2 01           (5 bits 1)
                          00 1 10      => 45% energy reduction
                          01 1 11
```

Physical layer – low-weight coding

2016 Zainuddin et al., Low-Weight Code Comparison...

Depending on channel, bits to be sent are better to be replaced

Mapping table for various codes:

Input symbol	Symbol freq.	ME	NME	PG	NPG	MTE	MEC	LWC	Unary
111	80	000	000	1000000	0000000	1100	0…0011	10010	1
110	70	001	001	0100000	0000001	0110	0…1100	10001	01
101	60	010	010	0010000	0000010	0011	.	01100	001
100	50	100	100	0001000	0000100	1000	.	01010	0001
011	40	011	101	0000100	0001000	0100	.	01001	00001
010	30	101	011	0000010	0010000	0010	.	00110	000001
001	20	110	110	0000001	0100000	0001	0011…0	00101	0000001
000	10	111	111	0000000	1000000	0000	1100…0	00011	00000001

Performance of the analysed codes:

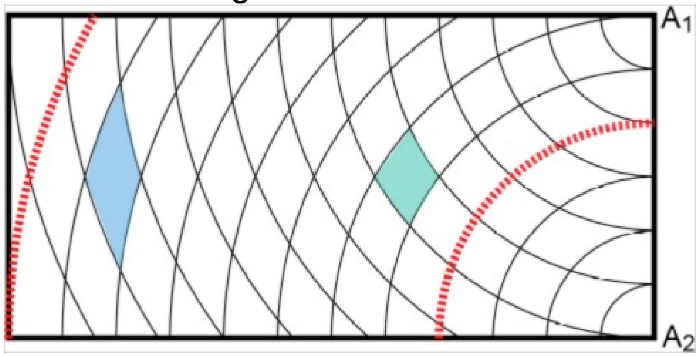
Code	Energy efficiency	Bandwidth expansion	Sequential bits 1	Multi-user interference	Robustness
ME, NME	++	++	−	−	++
PG, NPG	++	− −	++	++	+
MTE	++	−	−	+	+
MEC	+	− −	− −	++	+
LWC	+	−	−	+	+
Unary	++	+	+	+	− −

Conclusion: NPG and PG are better on all criteria except bw expansion

SLR (Stateless Linear-path Routing) coordinate system and routing protocol

SLR coordinate system

2016 A. Tsioliaridou et al. Stateless Linear-path Routing for 3D Nanonetworks



- Creates of a grid-like coordinate system
- Broadcasting a beacon from an anchor
- Nodes use the distance from this anchor as coordinate
- Generates a curvilinear coordinate system
- 2 anchors for a 2D network, 3 anchors for a 3D network

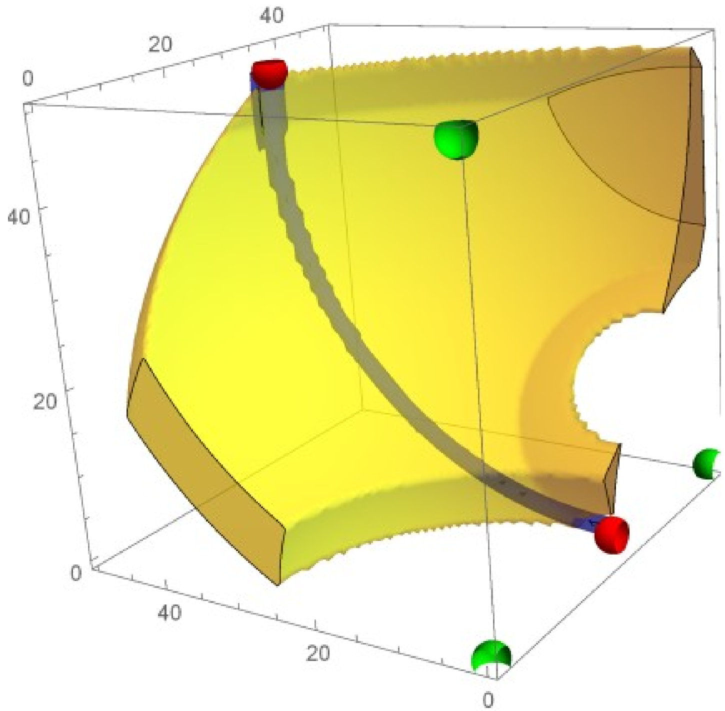
SLR routing

2016 A. Tsioliaridou et al. Stateless Linear-path Routing for 3D Nanonetworks

To forward a message from a zone to another, nodes checks if they belong to the path between source and destination zones
This mechanism uses only integer computations, which keeps SLR light

SLR is a geographical routing relying on a light infrastructure (anchors)

Routing:
Packet contains sender and destination
Nodes receiving a packet retransmit it iff:
(0≤diffs<n AND tj=sj+diffs)
  OR (diffs==n AND n≤diffr≤n+k AND tj=sj+diffr)
where diffs=ti-si, diffr=ti-ri
ri=si+n and rj=sj+n+k, with n,k≥0



Characteristics:
- 3D
- configurable path width for increased robustness
- choice of best viewport (anchors)

2. Simulation tools
BitSimulator, an electromagnetic nanonetwork simulator

Motivation

- Theory and numerical analysis (e.g. MATLAB) have limits:
  - analysing only a part of the dense network does not help
  - cannot take into account nodes with algorithmic behaviour
- Experiments are not possible:
  - no open platform with thousands of neighbouring nodes
- => We use simulations

Nobel committee (chemistry), 2013:

- "Simulations are so realistic that they predict the outcome of traditional experiments"

A. Legrand's HDR dissertation, 2015 [1, 2]

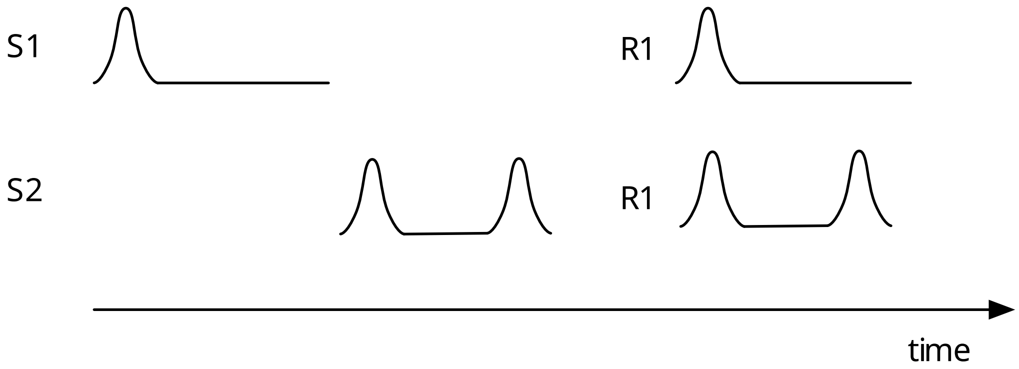
- These systems are so complex that solely evaluating through equations has become impossible
- Performing experiments on such infrastructures is costly and sometimes not even possible
- Other sciences experiment with real systems but also routinely use computers to understand complex systems
- Theoretical analysis like the ones we did [...] are typically tractable only when using stringent and ultimately unrealistic assumptions

Simulators

- No nanonode has ever been built, so we need simulators for nanonetworks
- 2013 Nano-Sim (Modena, Italy): ns3 module
  - simple propagation model (all or nothing)
  - 3 types of node: nanonode (sensor), nanorouter (collects data and forwards it), nanointerface (process data, gateway to external world)
  - 2 types of routing: flooding, random next hop
  - buggy
- 2016 Vouivre (Montbéliard, France): standalone library
  - realistic propagation and channel
  - scalable (1 million nodes)
  - see Julien's presentation for more
- 2018 TeraSim (Buffalo/NewYork, USA): ns3 module
  - implements all analytical models known to date
- 2018 BitSimulator (Montbéliard, France)

State of the art
ns3: Nano-Sim

- Nano-Sim: an ns3 extension
- Does not consider payload and propagation delay
  => collisions cannot be correctly computed

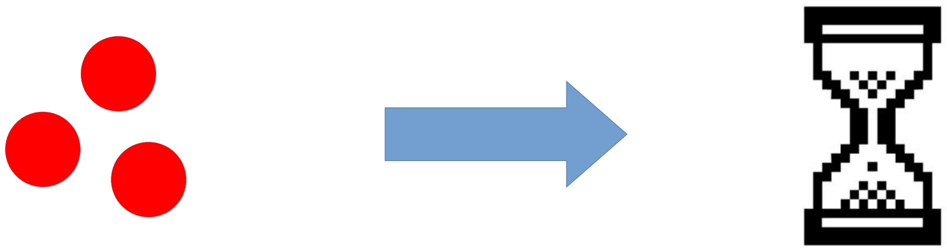


- Cannot handle networks with very high number of nodes

ns2 and ns3: “my colleague uses scenarios of 2000 nodes
 and they take days to complete” (hypothetical quote)
=> We have developed our own simulator
 we still have the same problem, but for 2 million nodes... :o)

State of the art
COMSOL Multiphysics

- COMSOL Multiphysics is a very low level simulator: it can simulate physics behavior
- Extremely precise
- Consequently very slow
  - can take several hours to simulate a scenario with a few nodes. It is useless on scenarios involving huge number of nodes.



State of the art
Vouivre

- Developed by our team
- Simulates a high number of nodes
- Does not take payload into account
- Uses a statistical model to compute collisions on frames
- Lacks a protocol stack to study interactions between various layers (e.g. MAC and routing)

BitSimulator

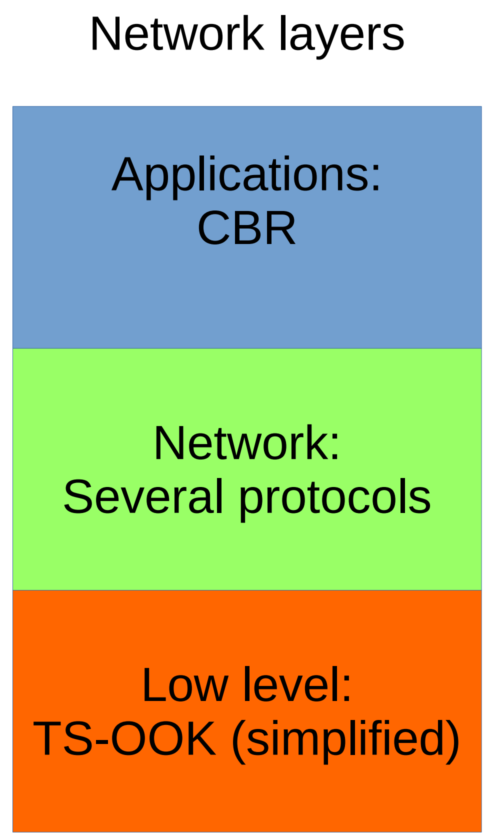
- Dedicated to nanonetworks
- Discrete event simulator (no artificial synchronisation barrier)
- Can handle hundreds of thousands nodes and very dense networks (tens of hundreds neighbors)
- Time precision: 1 femtosecond
- Space precision: 1 nanometer
- Correctly simulates propagation delay
- Simulates each communication at bit level: can compute each collision individually
- Allowed us to discover non intuitive properties which we were able to explain afterwards

DEMO

- broadcast
- 1 flow
- sleep

Implemented features

- Node memory (reception queue) can be configured. It models the ability to process a limited number of concurrent frames
- Simplified (but sufficient) protocol stack, with example and useful implementations already provided
  - Physical and MAC layer
    - TS-OOK
  - Routing / network layer
    - Pure flooding, probabilistic flooding, ...
    - SLR
    - Backoff flooding
    - Density Estimator for Dense Networks (DEDeN)
  - Application layer
    - Constant Bit Rate (CBR) source
    - Data sinks
- Under active development, http://eugen.dedu.free.fr/bitsimulator



Log system

- Various events are traced during the simulation (receptions, transmissions, collisions, …)

```
eventType    date(fs)  nodeID  packetType      flowId  sequenceNumber  collidedBits
4(reception) 3900234   2       3(densityInit)  1       0               0
```

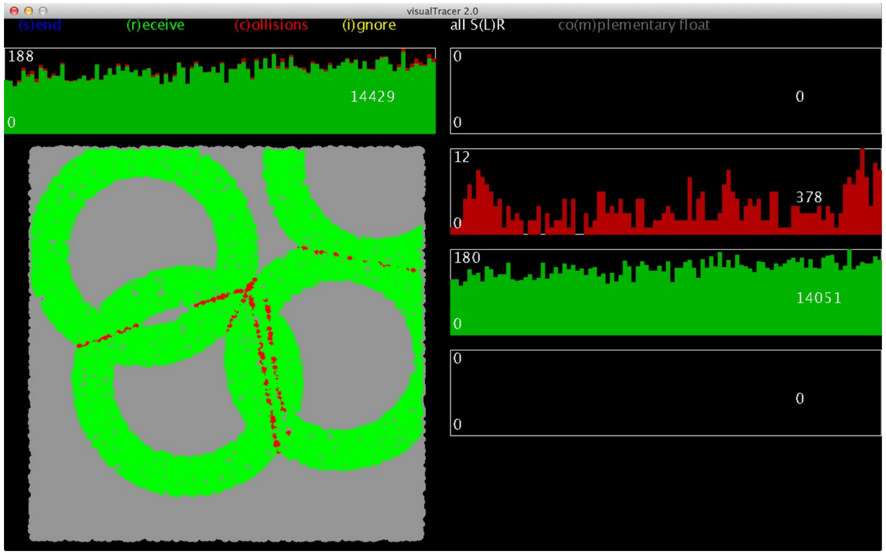
- Easy to add/remove items in log lines
- Easy to add/remove new types of line to trace various information during the simulation
- A log reader library is provided
to automatically read and process logs
- Is used by VisualTracer …

VisualTracer

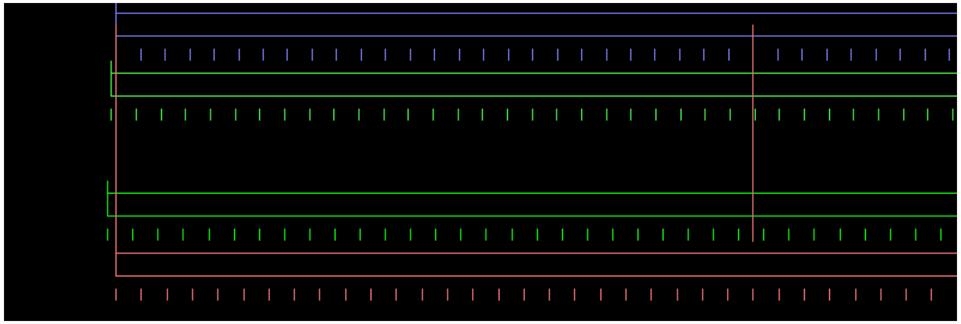
- Visualization tool for BitSimulator logs

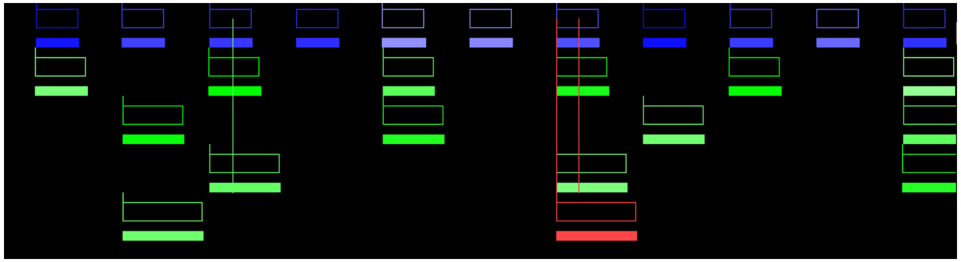
- Shows step by step the propagation of frames through the network

- Separately displays node currently sending, correctly receiving and receiving a corrupted frame (due to a collision)



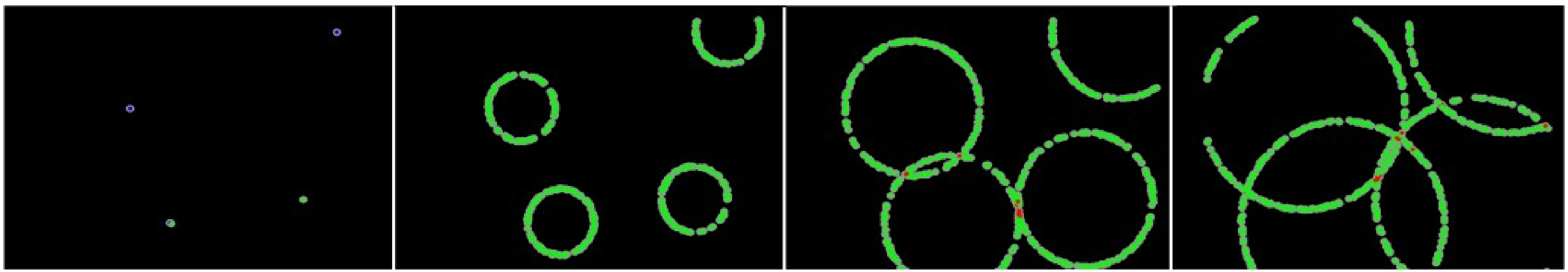
- Can also follow a node point of view in a chronogram mode



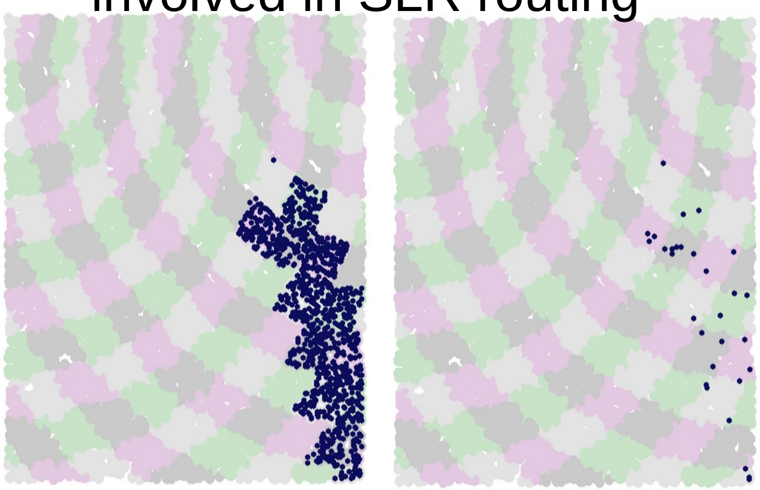


VisualTracer examples/interest/usefulness

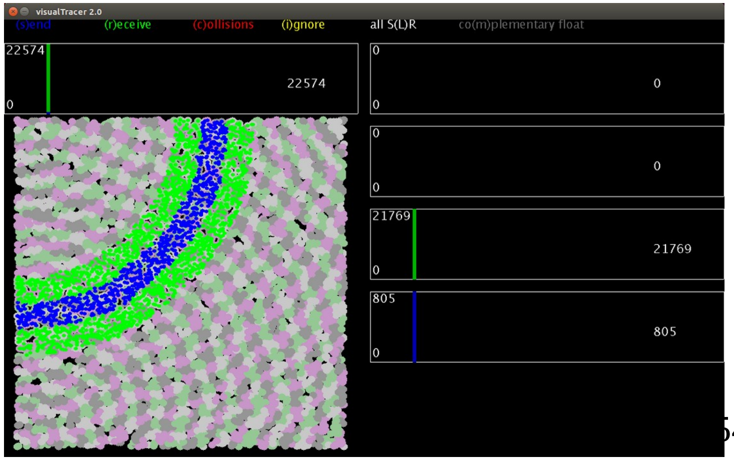
30000 neighbours receiving 4 bits over time
(4 very small intervals of time)



Many vs few nodes
involved in SLR routing



Histograms for sent, received,
and collided bits in an interval of time:
numerous nodes send at the same time



3. Broadcasting: backoff flooding

The problem: overcrowding

- A possibly huge number of nanonodes

- Even with very small communication range nodes can have thousands of neighbours and much more

- To transmit an information to whole network: broadcast

  - Pure flooding: all nodes repeat the message, a lot of resources are wasted
    - energy
    - channel usage

- Some technique is needed to replace the naive pure flooding approach in order to reduce the number of forwards when broadcasting in nanonetworks

Classical methods: Adaptive probabilistic flooding

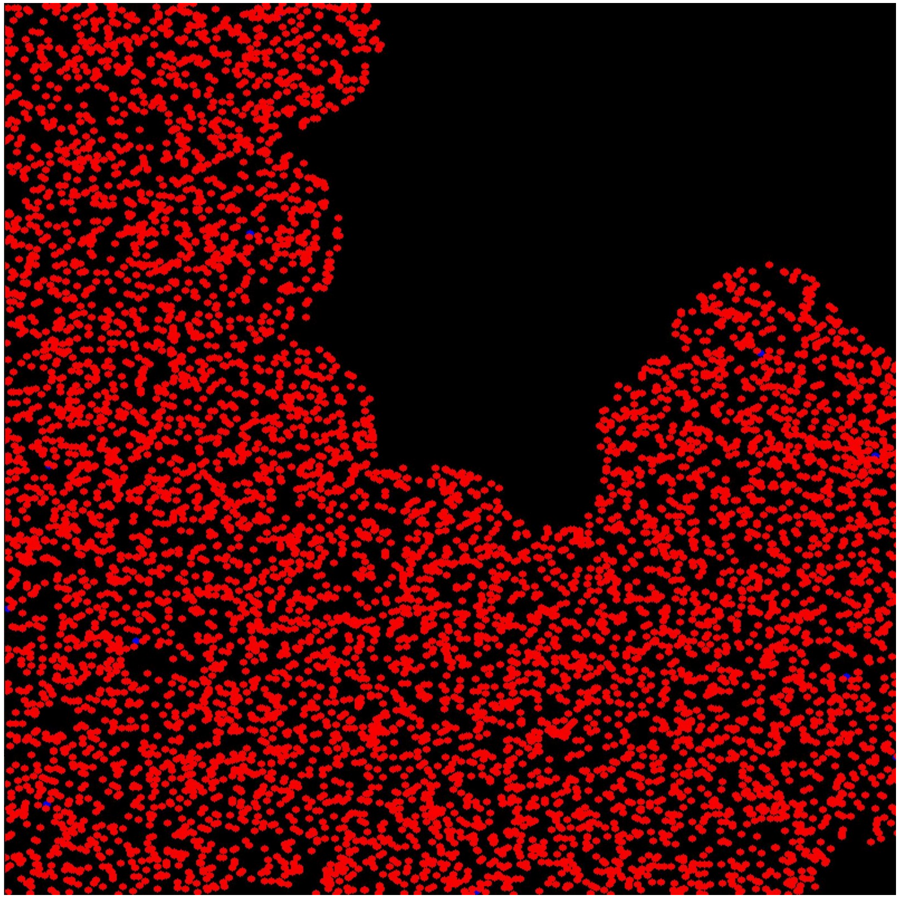
- Use probability to broadcast a packet

- The number of forwarded messages is fixed and tuned by the probability

- Very simple
  $p = f / n$
  p the forwarding probability
  f the desired number of forward
  n the number of neighbours

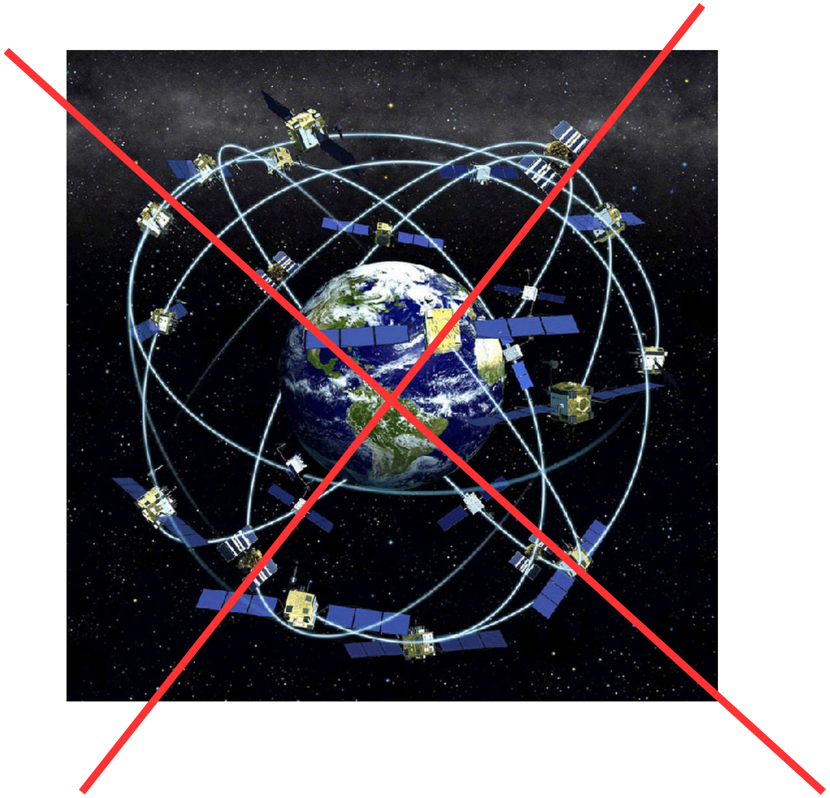
- Zero memory footprint

- May cause die out



Classical methods: Geoforwarding and OLSR

- No GPS => No geoforwarding
  Nodes are too small to embed GPS
- No infrastructure => No relative positioning
- No memory => No OLSR
  Too many neighbours to select precisely
  Maybe no unique IDs



Classical methods: Adaptive counter-based schemes

- Counting the number of transmissions to take the forwarding decision
- Backoff and waiting time not appropriate
  Have to be tuned correctly
- Density in nanonetworks varies widely
  Needs to take density into account
- Backoff flooding is adaptive counter-based

Our method: Backoff flooding

When a node receives a packet it waits for a random time and checks the number of copies he receives during this time. If the number of copies is below a threshold r (aka redundancy), the node forwards the packet, otherwise drops it

Waiting time window:

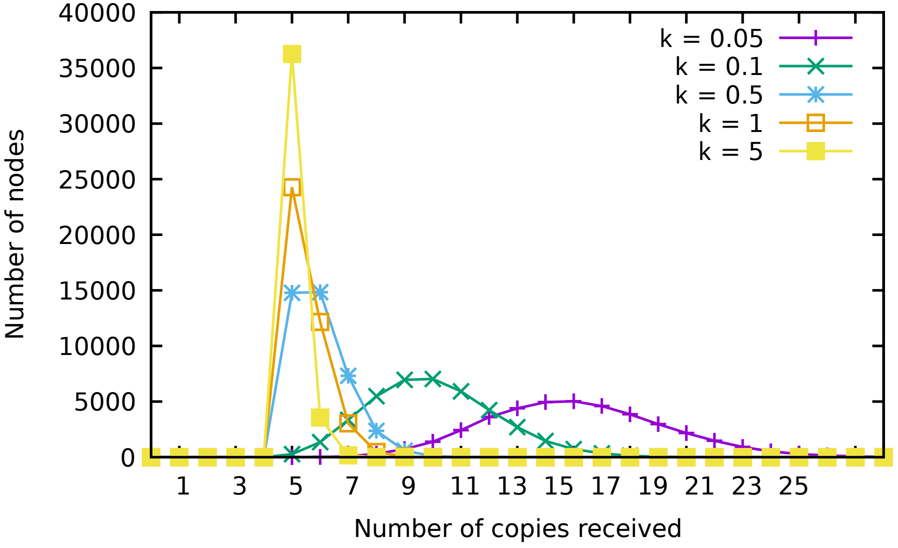
twait = n * k * 2(Tpkt)

- n is the number of neighbours and k a multiplier factor discussed later
- 2(Tpkt) is the time for the furthest neighbours to receive and send back the packet
- k is a forwarding factor

Properties: Window size

- Theoretical results
- r = 5 specifies the number of copies to be received
- If k is too small, the waiting time before transmitting is not large enough and nodes forward the message before noticing that 5 copies have already been sent

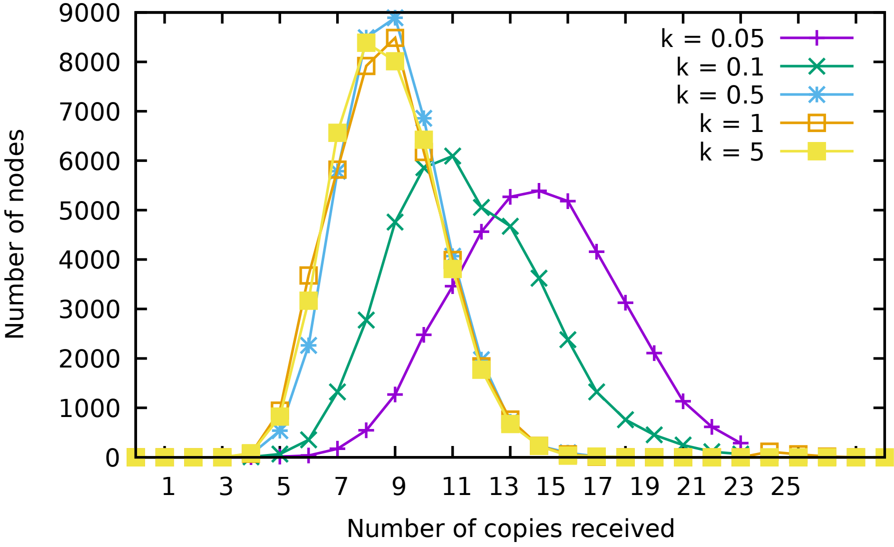
neighbours: 1150
twait: 8 nanoseconds
r: 5
k: various values



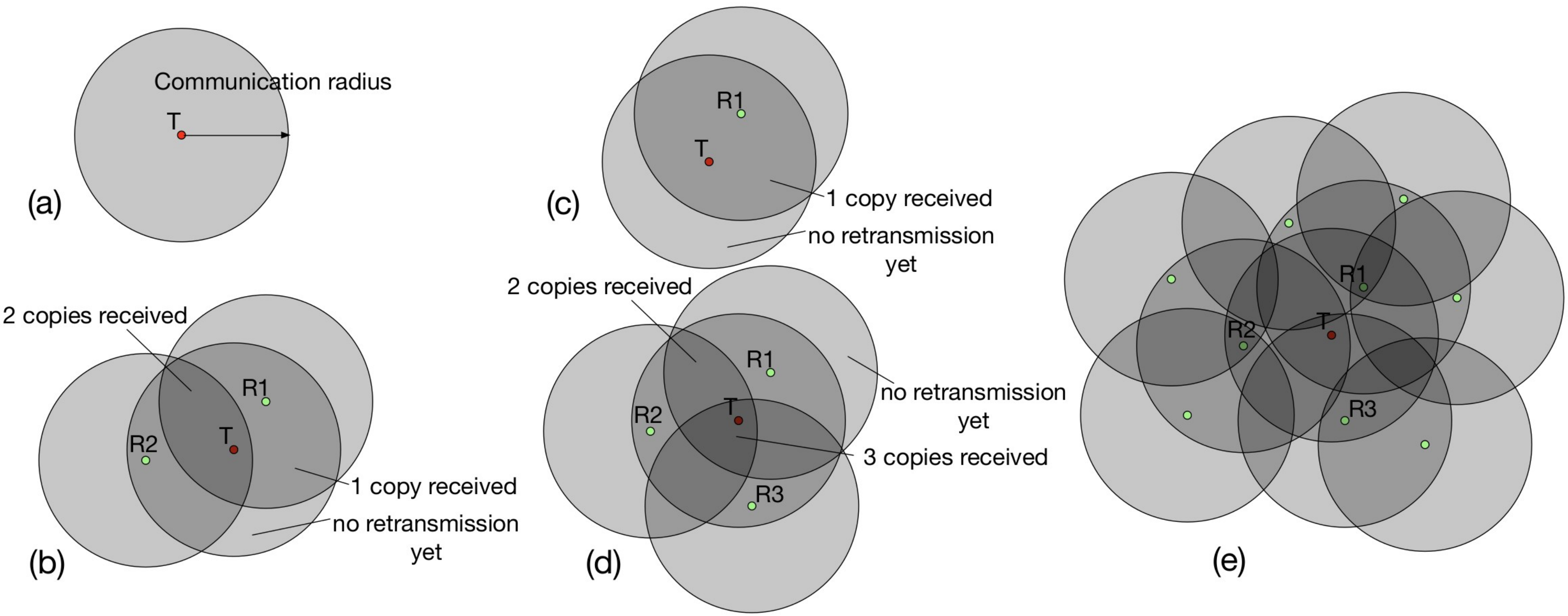
Properties:
Number of packets received

- The number of copies actually received is higher than r when simulated due to the “geographical effect”
- Even with high waiting time, nodes receive more than r copies of the packet
- No node received LESS than r copies of the packet

neighbours: 1150
twait: 8 nanoseconds
r: 5
k: various values

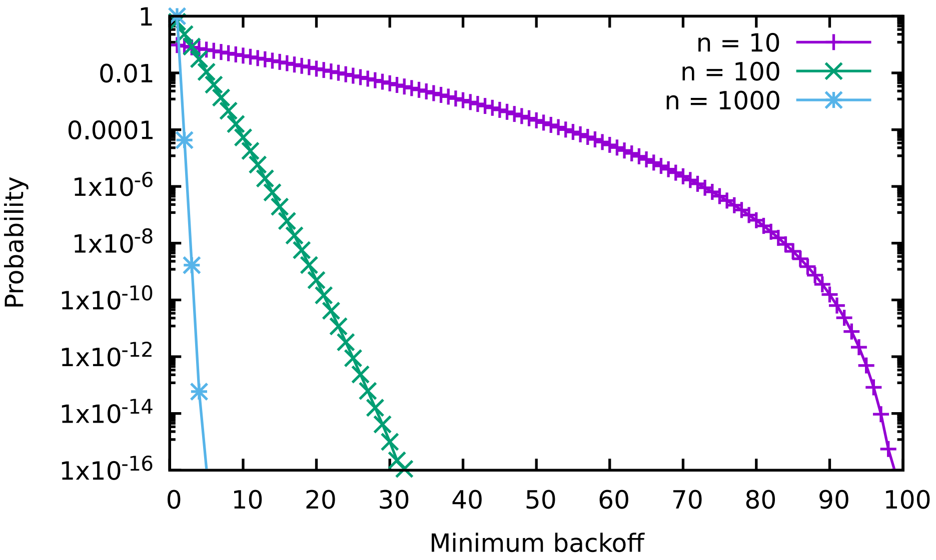


Properties: Geographical effect



Properties:
Minimum backoff probabilities

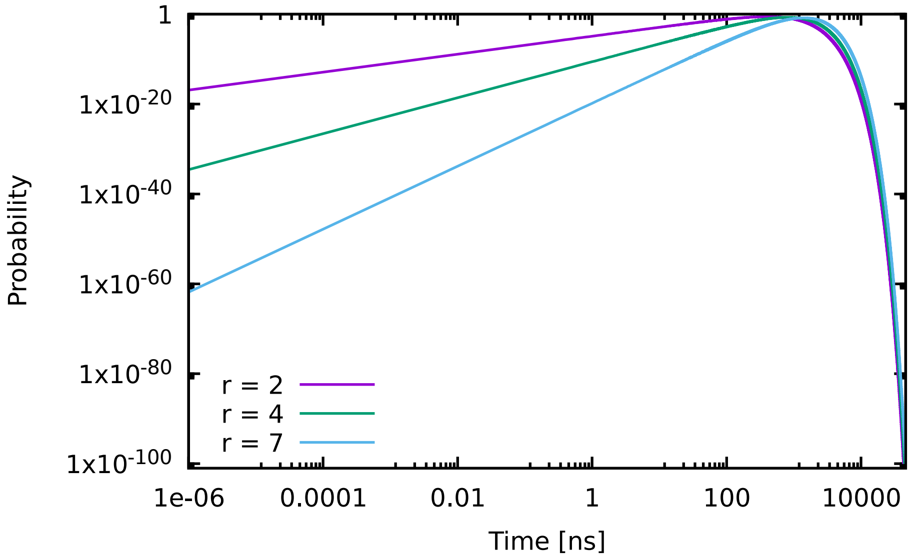
- Different node densities
- Show the probability for the minimum backoff (the first transmission) to be at x^{th} percentage of the window
- The probability quickly decreases: the mean backoff is smaller than the usual window / 2
  => Because the message progresses with the minimum backoff among neighbours



Properties: Delay

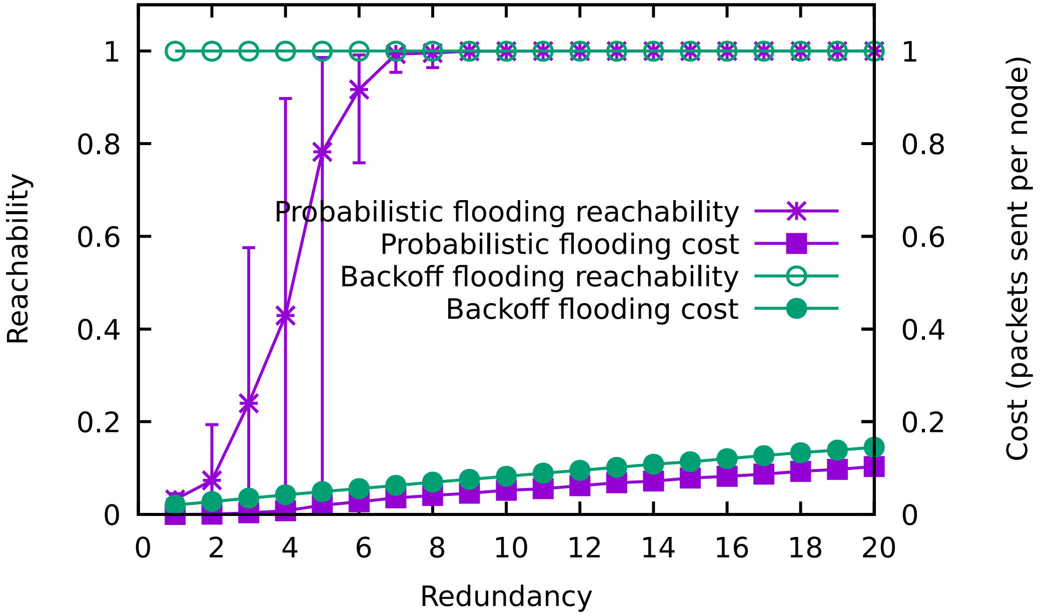
- Backoff flooding induces a predictable delay
- Figure show the probability (y axis) for the rth node to transmit its copies after the time of the x axis
- Most of the probable values are in a narrow range. The redundancy does not affect the delay
- It is a small percentage of the total window

neighbours: 1150
twait: 8 nanoseconds
r: various values
k: 1



Properties: Reachability

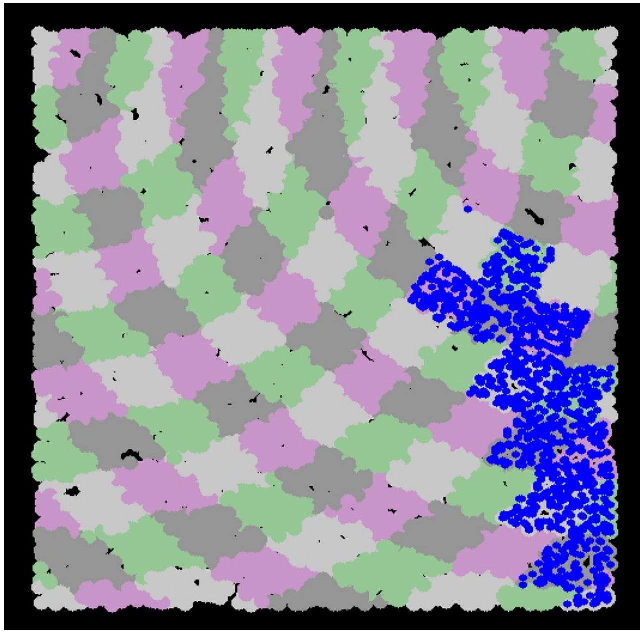
- Reachability comparison between probabilistic flooding and backoff flooding
- Backoff flooding is steady and reaches the whole network even with a redundancy of 1
- The backoff flooding sends fewer packets than the probabilistic flooding to reach the whole network



SLR / Backoff flooding

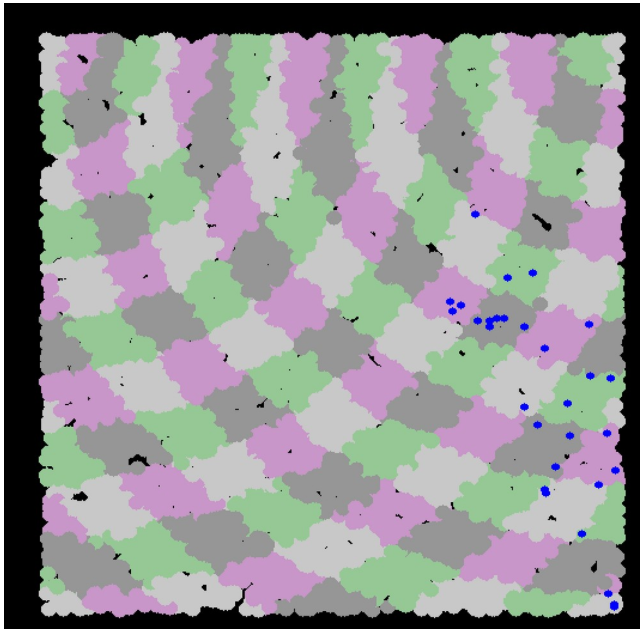
SLR: Stateless Linear Routing

- Designed for nanonetworks
- Geocasting routing using relative positioning



Backoff flooding

- Enhancement of pure flooding that reduces the number of packet sent
- Compatible with SLR



Backoff flooding conclusion

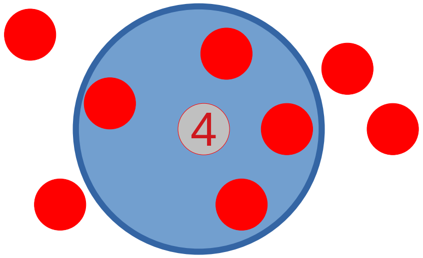
- Backoff flooding is a counter-based forwarding scheme adapted to nanonetworks
- Guarantees a minimum number of forwards
- Limits the number of forwarders
- Very high reachability
- Takes network density into account => Needs neighbours information
- Introduces a small and predictable delay
- Does not need any location system
- No die out problem, even with low redundancy
- Future work => Sleeping nodes: femtoseconds cycles

Backoff flooding – questions

- What major issue of the probabilistic flooding does the backoff flooding fix? Give an example of the issue, to better understand it.
- Backoff flooding adds some delay to packet forwarding; what is this delay for a number of neighbours of 100 and a window of 200 µs?

4. DEDeN, estimating the number of neighbours

DEDeN



- Density Estimator for Dense Networks
- An algorithm implemented in BitSimulator
- Designed for very dense networks (as nanonetworks can be)
- Can handle very wide range of densities: from few to several thousands of neighbors
- Divided in several rounds with increasing probabilities of sending local probes
  Tunable confidence using pre-computed probe threshold
- High performance in terms of estimation error and number of packets generated

How to count neighbours, current solutions

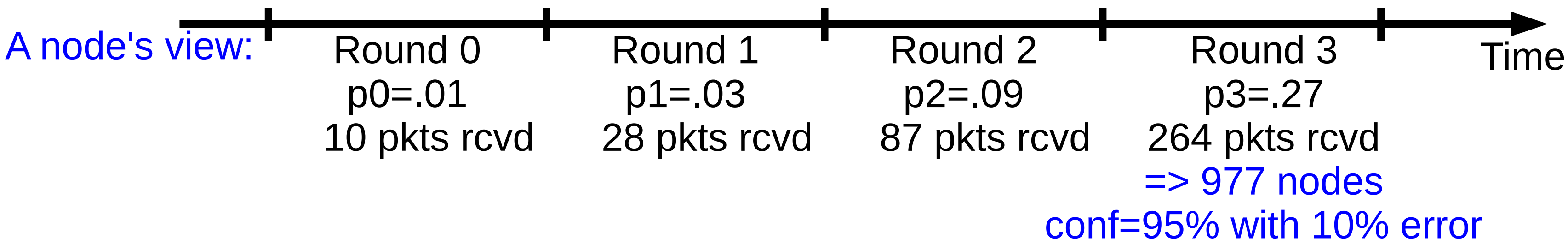
- So:
  - it is about neighbour counting, not about their identification, position, distance etc.
  - about counting all nodes, not only the communicating ones (got using collision probability)

- Count people in this room (imagine numerous participants)
  - hello method: I say "ping!", and each of you replies "pong!"
    - drawbacks: 1 packet per node, collisions (depending on message length compared to number of persons)
  - better: answering probability of 10% => .1 sent packets per node
    - drawbacks: still collisions; but which is the best probability? chicken and egg problem!
  - the more the neighbours (i.e. the bigger the density), the bigger the drawbacks!

Properties of the estimator

- Executed each time is needed:
  - either during network setup
  - or at hard coded intervals of time
  - or upon reception of a 1-hop signal received from a macro equipment
  - or when a given node broadcasts a message...
- Not precise, but gives an (maximum likelihood) estimation
- Tunable: application can set a guaranteed error of estimation for a given confidence
- Estimation obtained by only one node, or by all the nodes
- Small overhead in terms of number of exchanged packets
- Unaffected by background traffic
- Tiny memory footprint: only a counter per node is needed
- Targets high density networks, but works in low densities too

Algorithm of the estimator



- Rounds of identical duration, known by all nodes
- Each round i, each node, after a backoff, sends a packet with probability pi
  - initially, p0 close to 0
  - pi grows exponentially with the round: pi = p0*growthRate^i, with growthRate>1
- The algorithm ends when:
  - either the number of packets received k during last round exceeds a threshold precomputed as a function of the desired confidence and pi
  - or pi≥1
- Then, each node estimates the number of neighbours as k/pi

Computation of the estimation

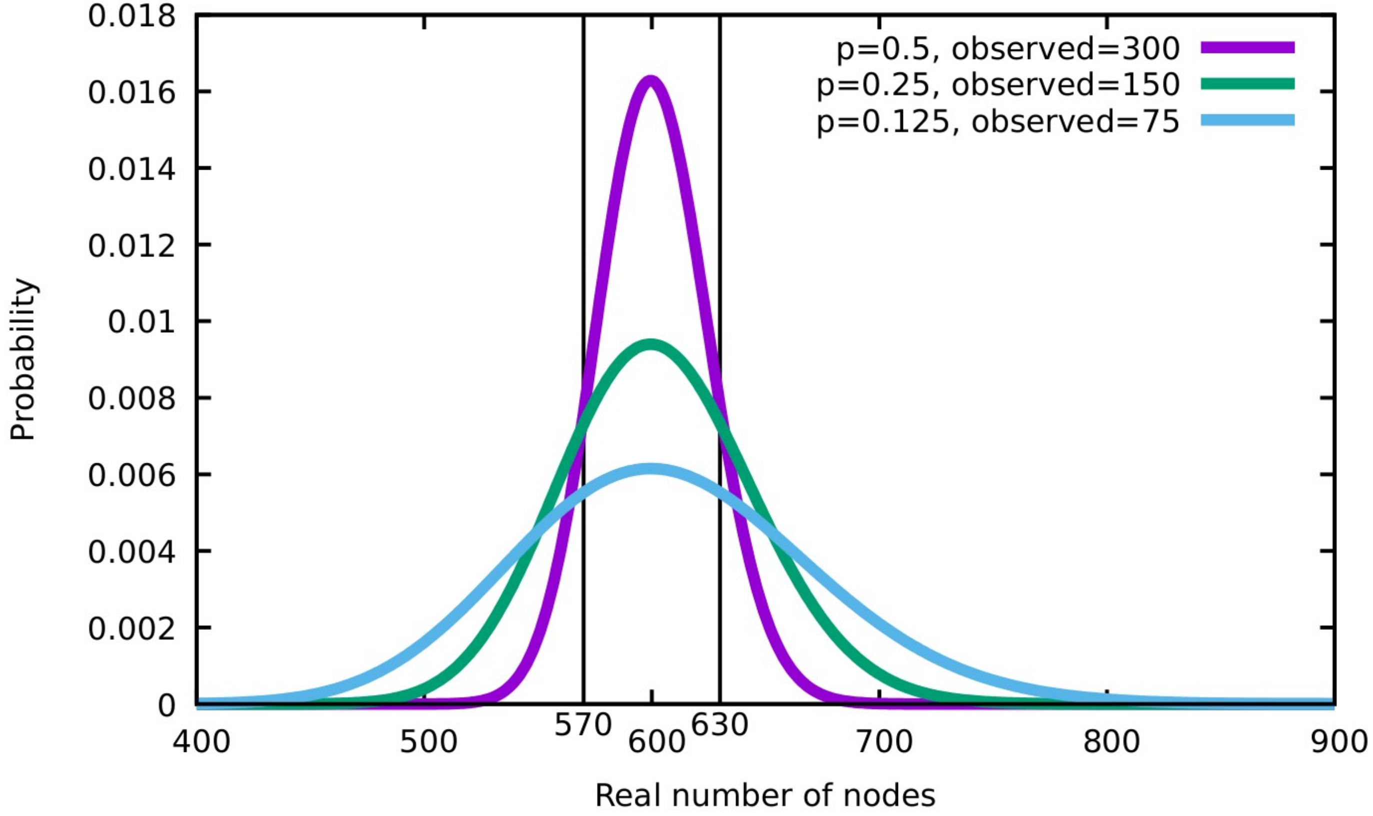
- Input: cmin, emax, pi, k (nb of pkts received)
- Output: n (number of nodes in reality) with given confidence and error
- Most likely k/pi neighbours, but is it within required error and confidence?
- We want n be in the interval:
  - n_min = k/pi * (1+emax)
  - n_max = k/pi * (1-emax)
- with a confidence (probability) P ≥cmin
  - where Pr is the probability that there are n nodes in reality

$$P = \frac{\sum_{n=n_{min}^{real}}^{n=n_{max}^{real}} Pr(k, n, p^{trans})}{\sum_{n=0}^{n=\infty} Pr(k, n, p^{trans})}$$

$$Pr(k, n, p) = Pr(X = k) = \binom{n}{k} p^k \, (1-p)^{n-k}$$

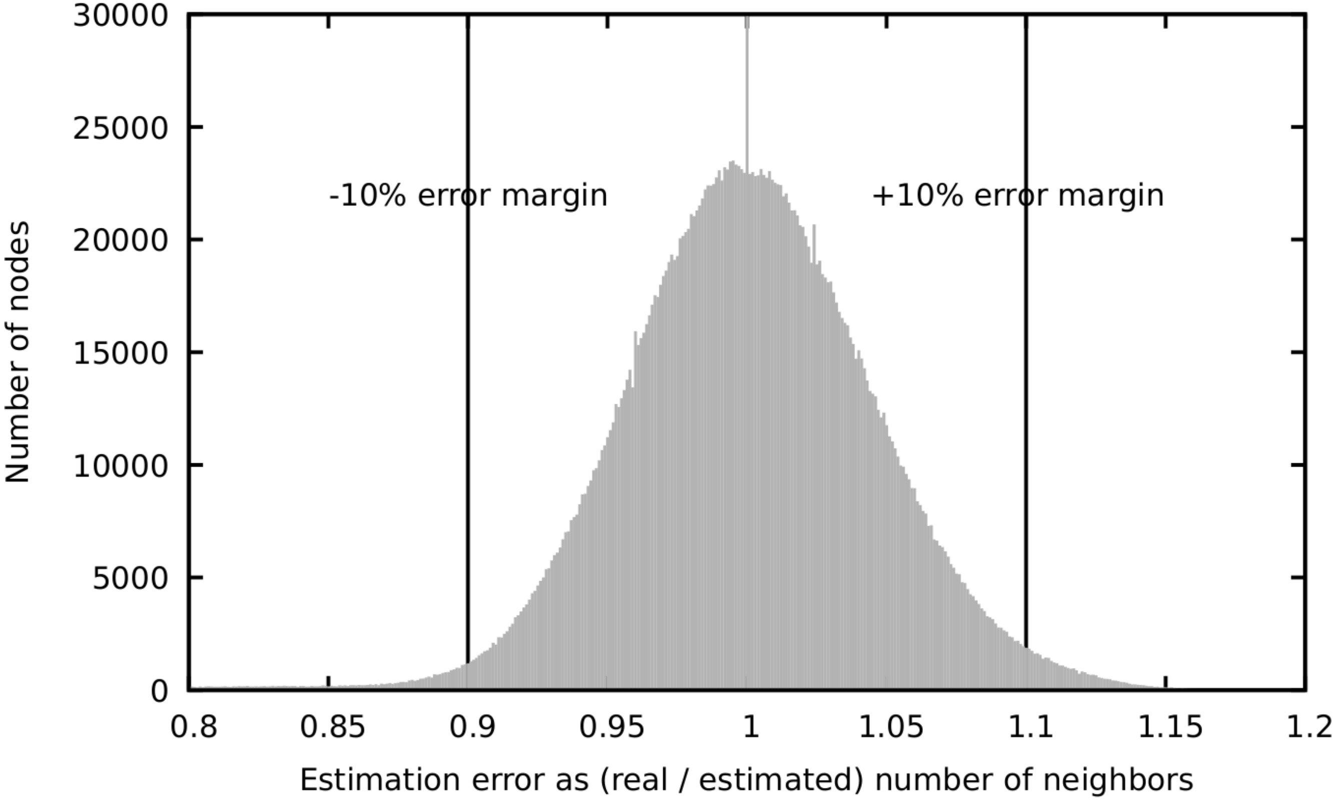
Theory: Confidence

Theory: probability distribution of the confidence based on pi ("p" in the figure) and number of received packets k ("observed" in the figure)



- 600 neighbours most probably
- vertical lines: 5% error
- the confidence increases with p

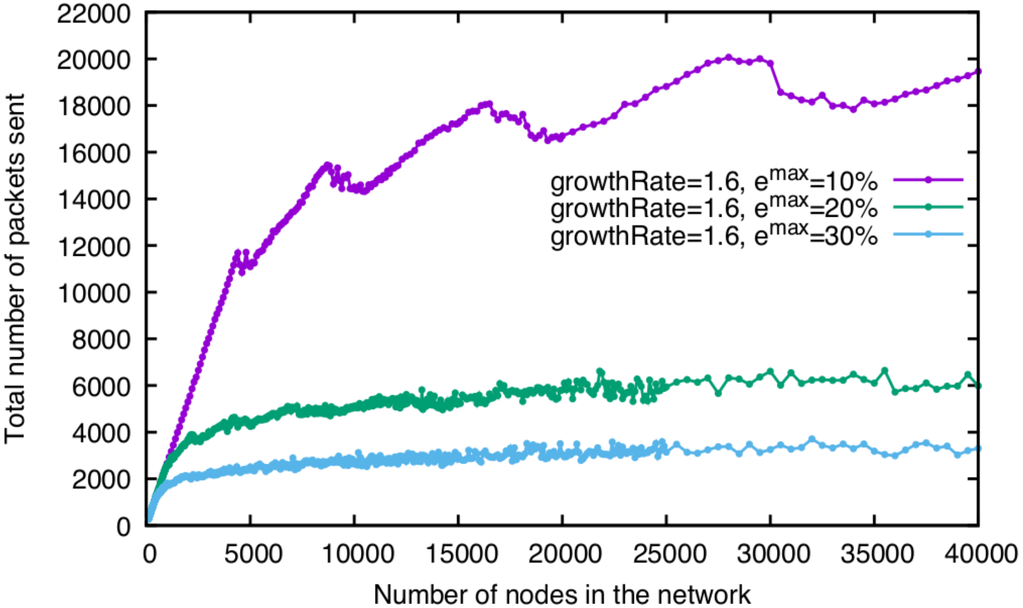
Simulation: Similarity of results



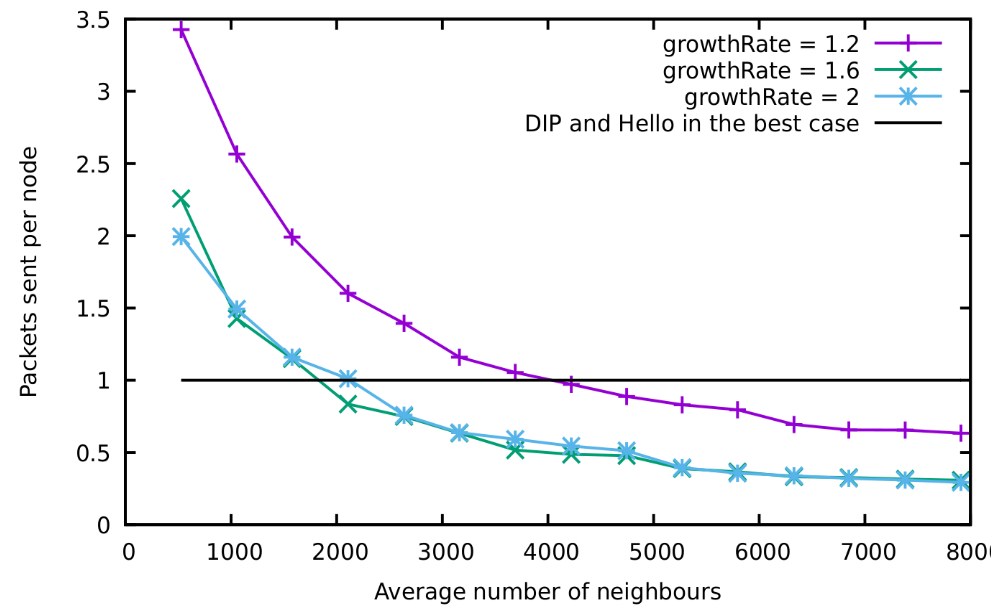
- 2.5mm x 2.5mm
  comm. range 0.5mm
- Emax=10%
  confidence=95%
  growthRate=1.6
- sum of 202 scenarios
  nb. neighbours 10..4200

Estimation cost: number of packets sent

- 2.5mm x 2.5mm, comm. range 0.5mm
- 200 scenarios, 100..40000 nodes placed randomly

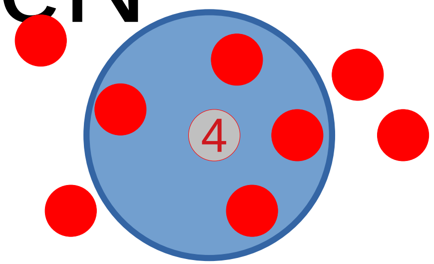


- emax has a big influence on the number of exchanged packets



- Smaller overhead for dense networks => less bandwidth used, fewer collisions, ...
- The overhead gets smaller with the density

Conclusion on DEDeN



- Proposed an algorithm allowing nodes to estimate the number of their neighbours
- Useful to flooding, various transmission schemes, sleep decisions etc.
- Functioning: several rounds, with increasing probability of replying
- Tunable confidence and estimation error
- Has a tiny memory footprint, and a small overhead (number of packets exchanged)
- The denser the network, the better the results
- Future work: continuous estimation, use it to reduce network congestion

5. Congestion control by deviation-based routing

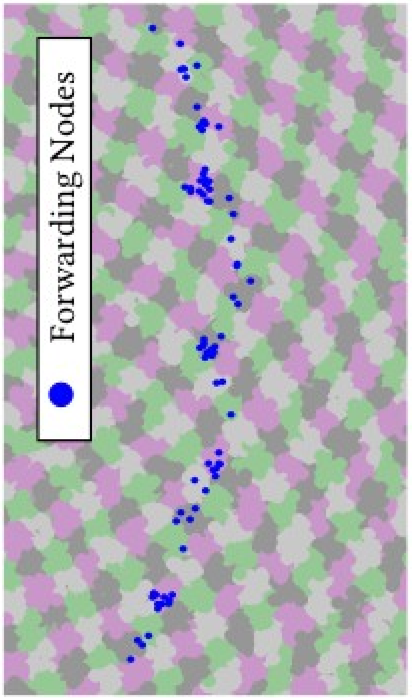
Transport protocol – none

- Specificities, challenges:
  - different methods to change throughput: beta, congestion avoidance, concurrent communication etc.
  - Tb/s throughput
  - little energy => should be much simpler than TCP
  - little energy => different modulation
    - can receive numerous packets in Ts interval and send only one packet
    - better to send 0s than 1s
  - tiny size => need to harvest enough energy to send/receive a packet => the channel may be free, but node has no energy to send the packet
  - propagation time bigger than emission time => several packets in the same time on the channel => classical ARQ (Wi-Fi) inappropriate

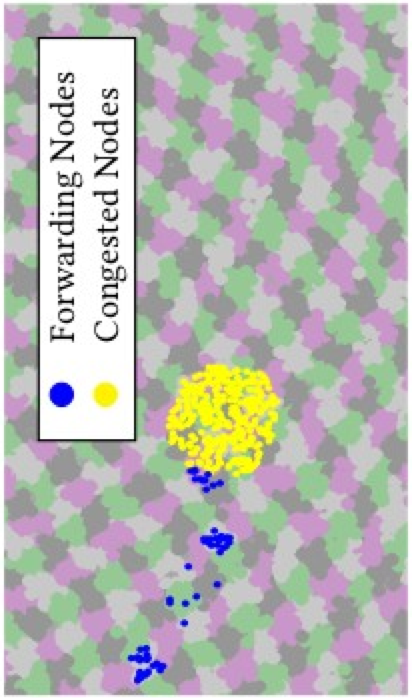
Challenges of congestion control

- Congestion detection
  - Tb/s => bandwidth is not a problem
  - limited resources => memory is a problem
  - cannot process all the incoming packets => maxConcurrentReceptions variable
  - if local congestion ≥ MCR/2, then deviate packet
- Congestion deviation
  - increment distance field in packet
  - routers at specified distance route this packet => deviation

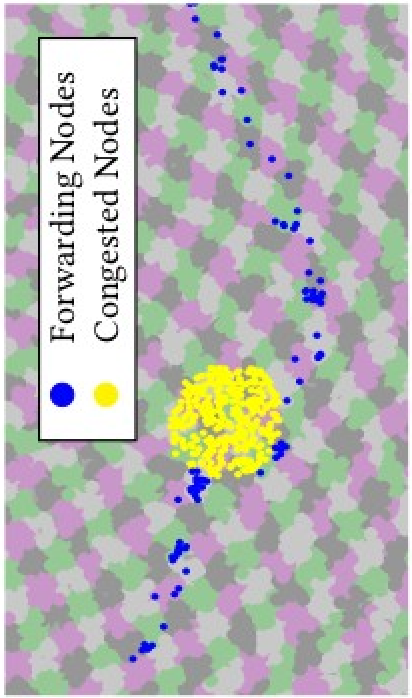
In a glance



(a) No congestion: Messages forward on an SLR path.



(b) Congestion with SLR: The message cannot pass the congestion.



(c) Congestion with deviation: Routing deviates as it encounters the congestion.

Packets arrive at the destination, at the cost of greater transmission time (compared to without congestion)

Reception buffers

- Due to fabrication constraints, those hardware resources are limited

  => This limits the number of simultaneous frames a node can follow

  Basically, a node will be unaware of any new packets arriving when all reception resources are in use, and consequently lose them, even in the absence of collision

  This is different from traditional routers, where frames are sequentially received by interfaces and can be lost only if the transmission buffers are full. In that case, routers are aware of the packet drop
- Using a simple counter, we consider that nodes are aware of the number of reception buffers currently in use

Congestion definition

- Knowing the maximum number of buffers and the number of reception buffers currently in use, we can compute the node congestion level C at time t:

$$C(t) = Rn(t) / Rmax$$

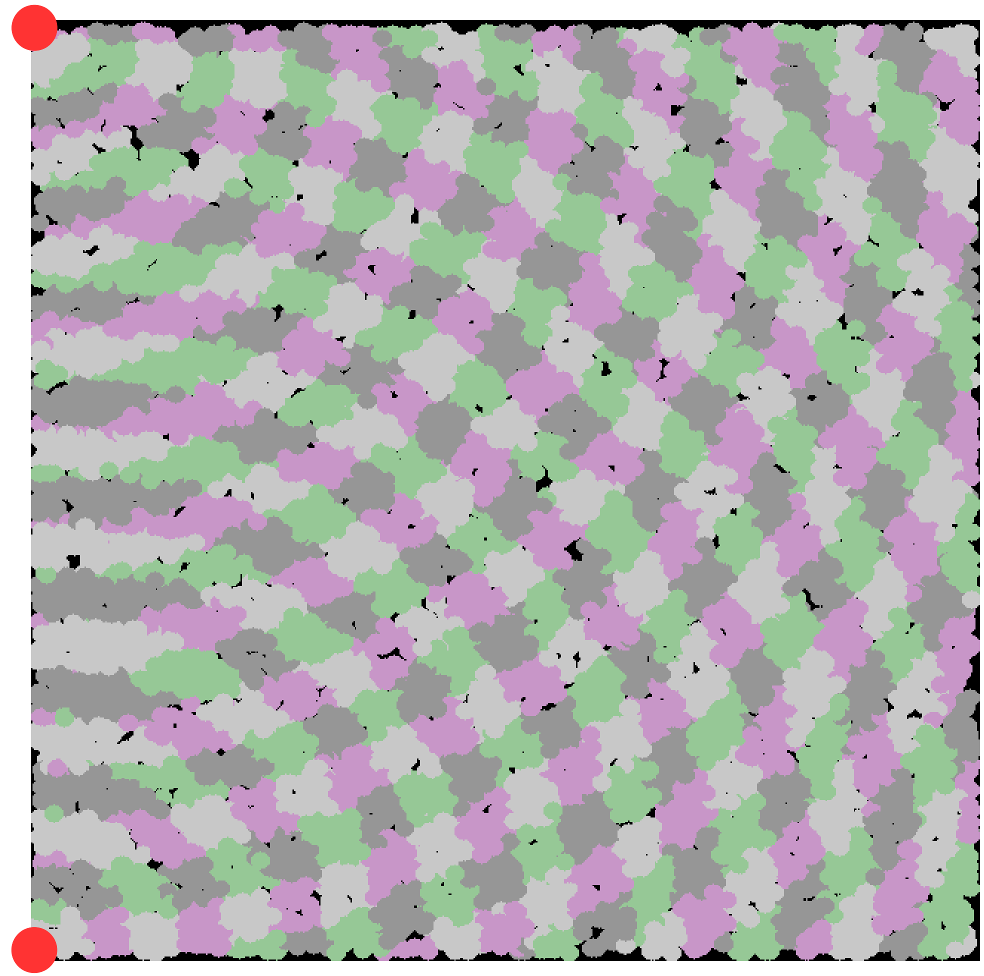
  with Rn the number of reception buffers in use at time t and Rmax the total number of buffers available on the node

Congestion thresholds

- We now define two congestion thresholds:
  - upper threshold Cu: node is congested, measure should be taken
  - lower threshold Cl: node is not congested anymore, back to the normal behaviour
- Those two thresholds can be equal and hence behave as a unique threshold

Recall: SLR addressing

Beaconing anchors

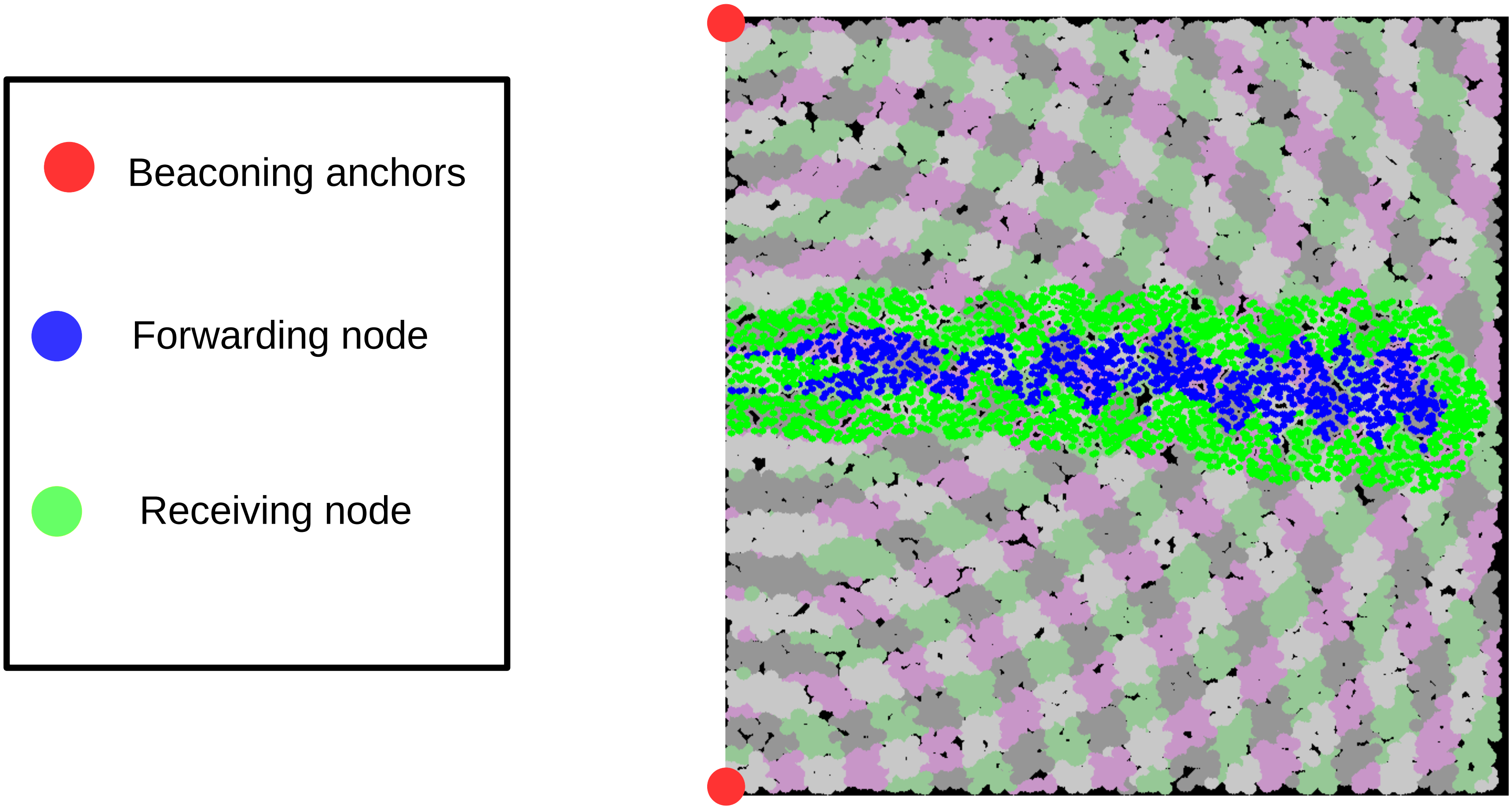


Every anchor broadcast a beacon containing a counter incremented at each hop

By observing those counters, each node can know how far it is from both anchors

With adequately placed anchors, this creates a coordinate system

Recall: SLR routing



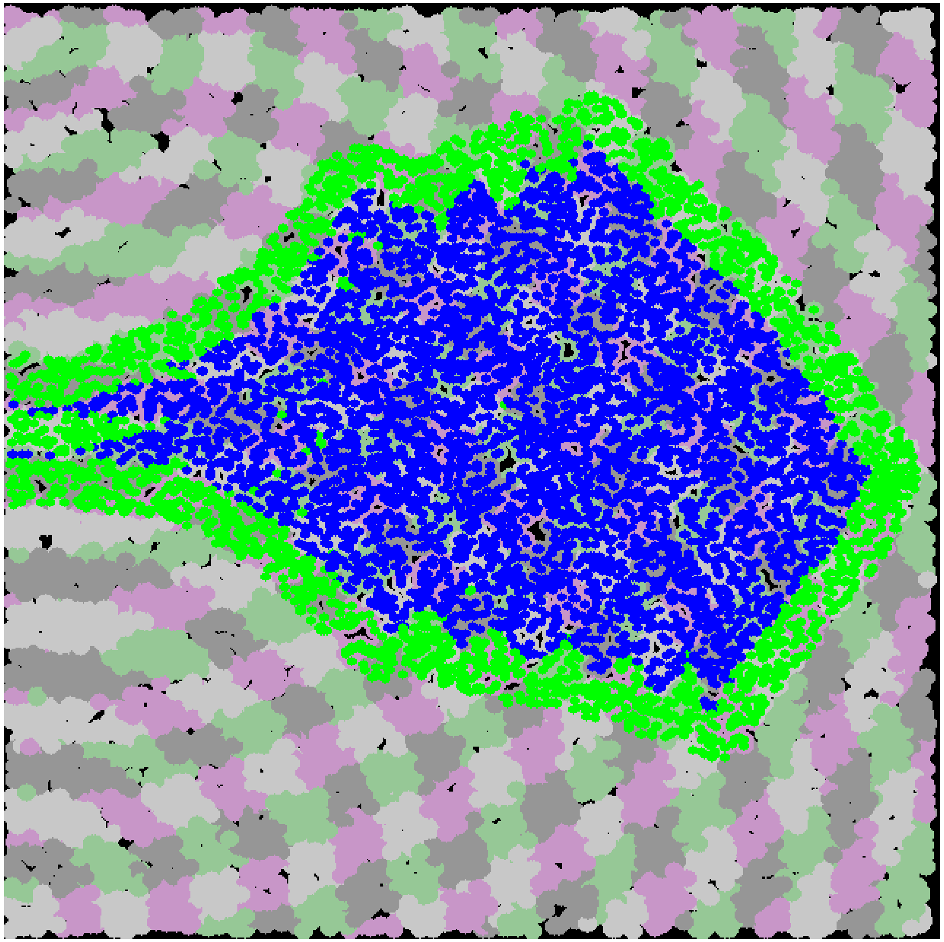
SLR path width

To increase reliability, SLR can be tuned to used a wider route

Worst for congestion: use more resources

But can be used wisely ...

```
bool isOnPath (node n,
               address src,
               address dst,
               int m) {
    return: n is on SLR path of
            width m from src to dst
}
```

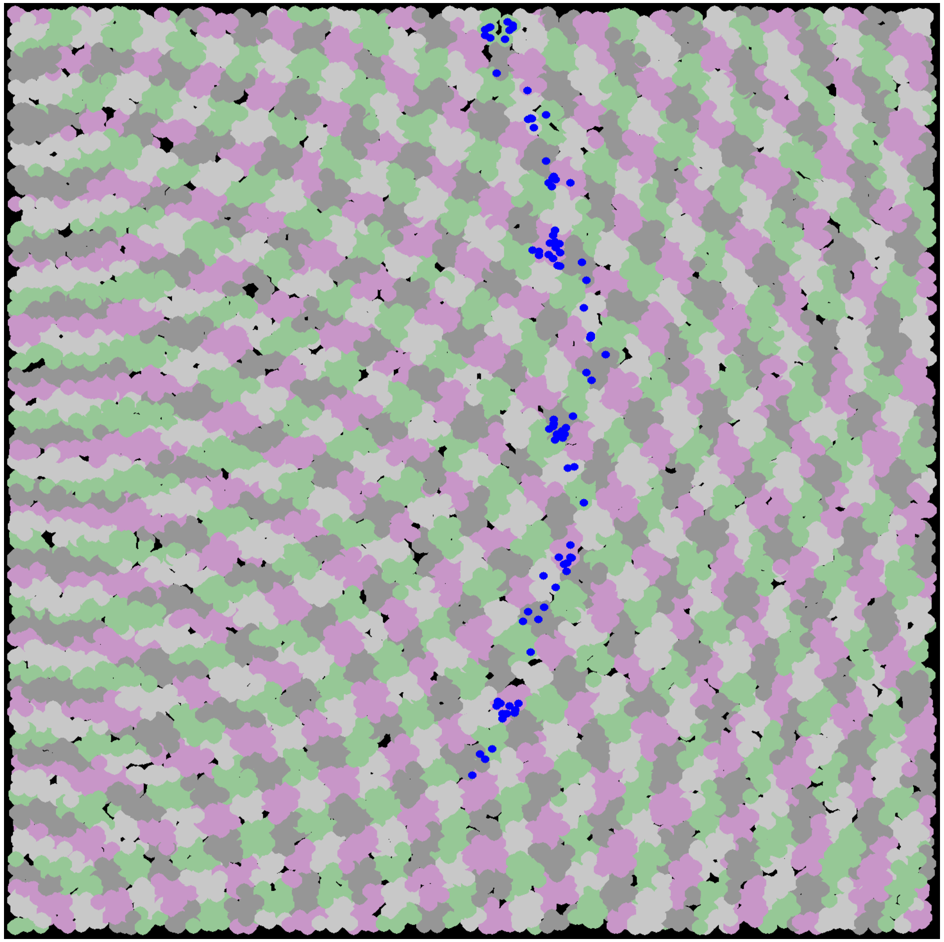


Reducing the number of forwarders

Need to reduce the number of forwarders to not detect congestion with only one message in the network

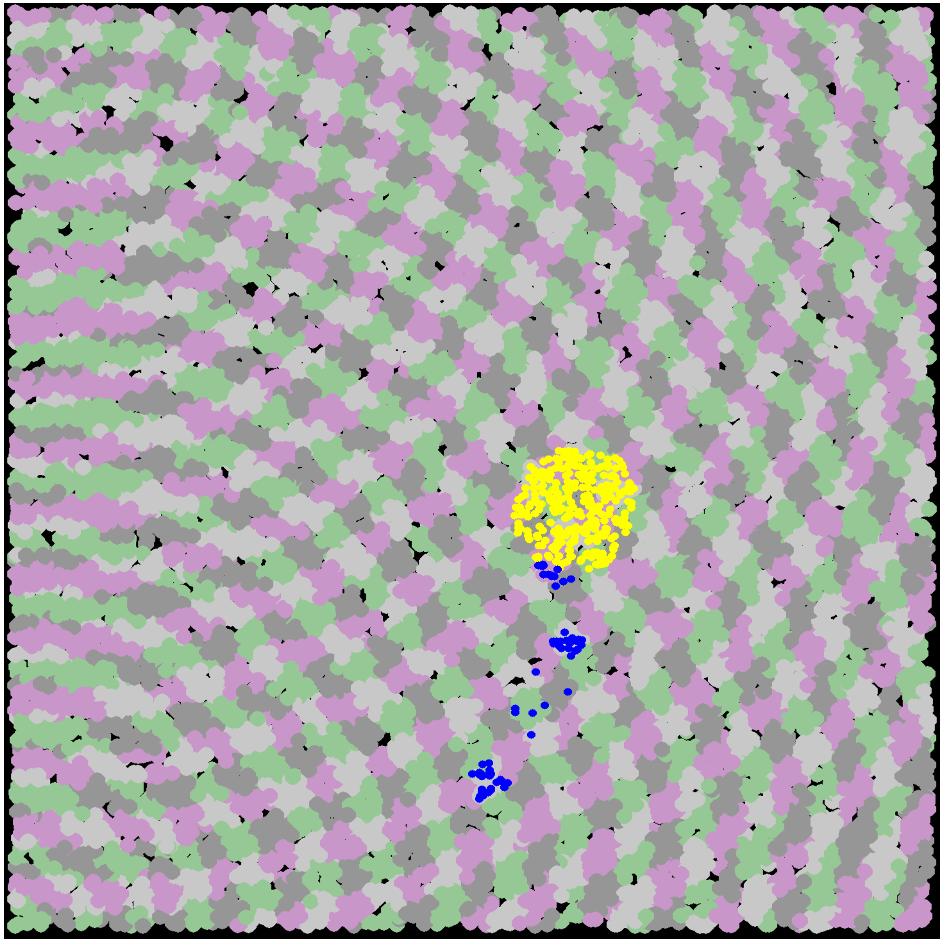
=> Random selection of the next forwarder via the backoff flooding

It is counter-based and designed to ensure that the message is forwarded while keeping the number of forwarders low



Congested area

- Wireless communications are broadcasted by nature
- Nodes are able to handle multiple simultaneous incoming frames, but if too many senders are active at the same time, they will saturate their reception capabilities
- In this example, independent and parasitic flows are active in the yellow area. All neighbouring nodes are affected



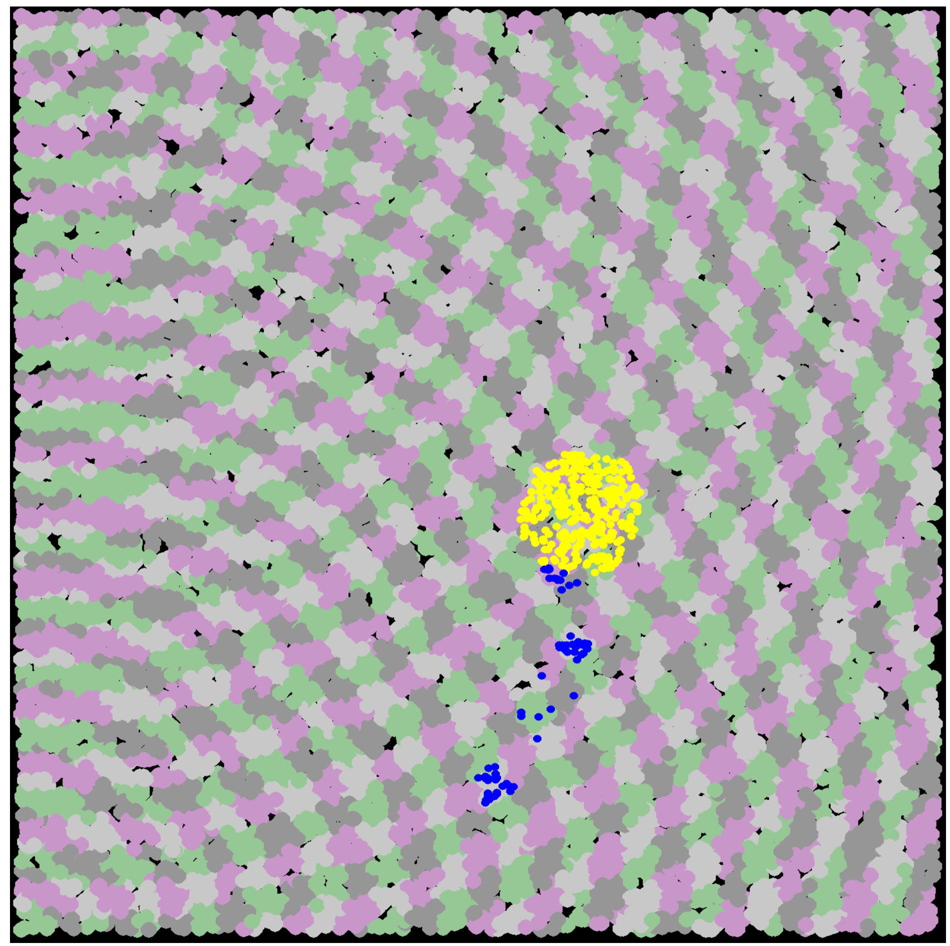
Deviation routing: Congestion blocking

Forwarding node

Congested node

Yellow nodes are completely congested, they cannot receive more packets. Node around have reached the Cu threshold due to local traffic, the prevention mechanism is triggered

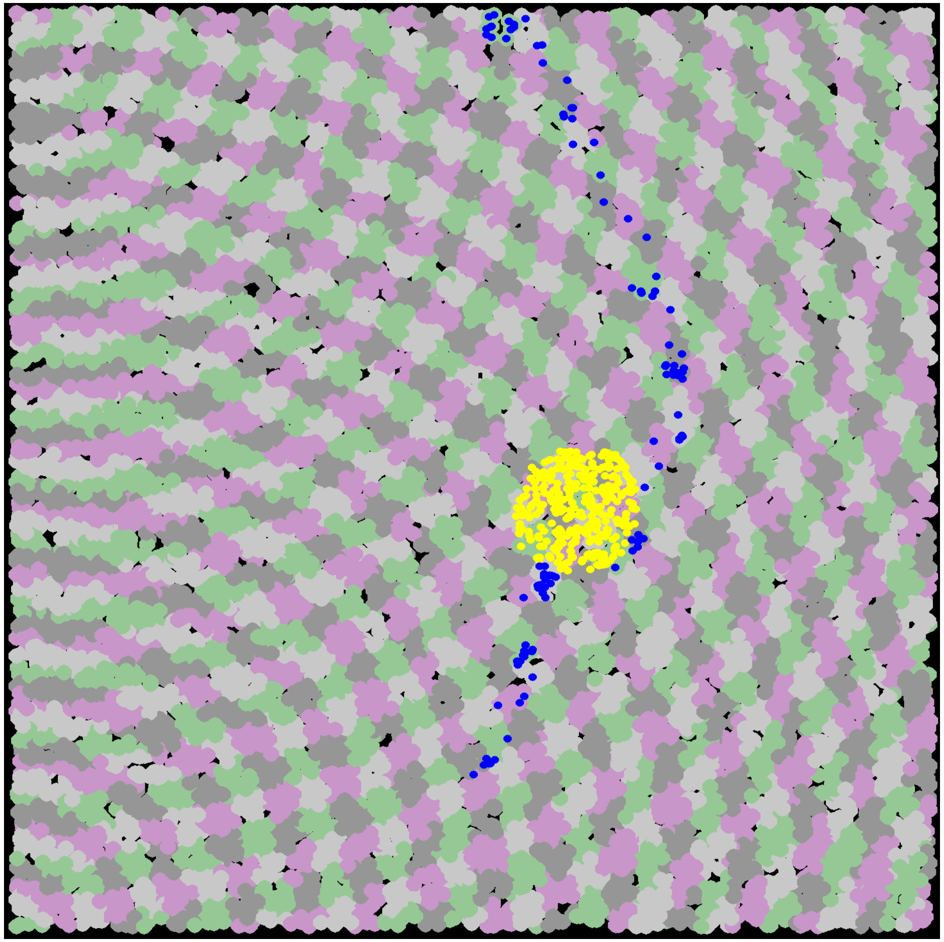
The flow cannot reach the destination because there are no resources available on yellow nodes: they cannot handle new packets



Deviation routing: Congestion avoidance

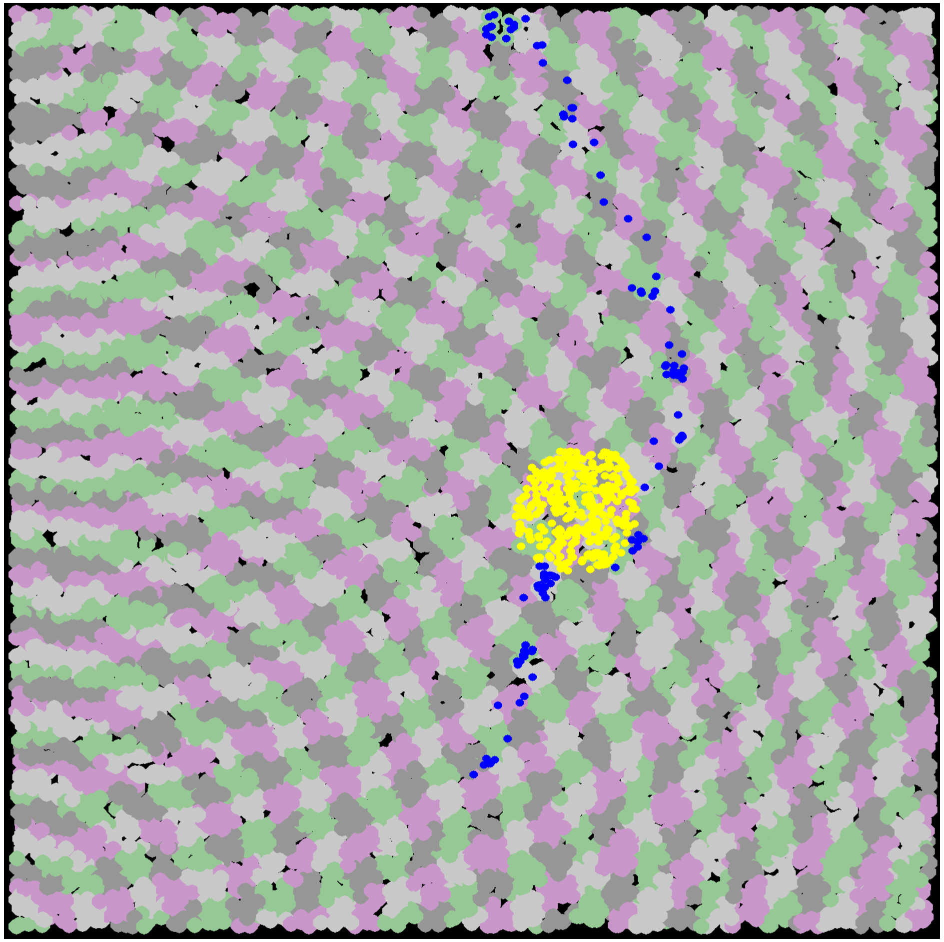
We use the capability of SLR to spatially occupy the network to create a new route and deviate from the original one

```
bool isOnPathDeviation ( n,
                         src,
                         dst,
                         m){
  return: isOnPath (n,src,dst,m) AND
    NOT isOnPath (n,src,dst,m-1)
}
```



Deviation routing: 3D

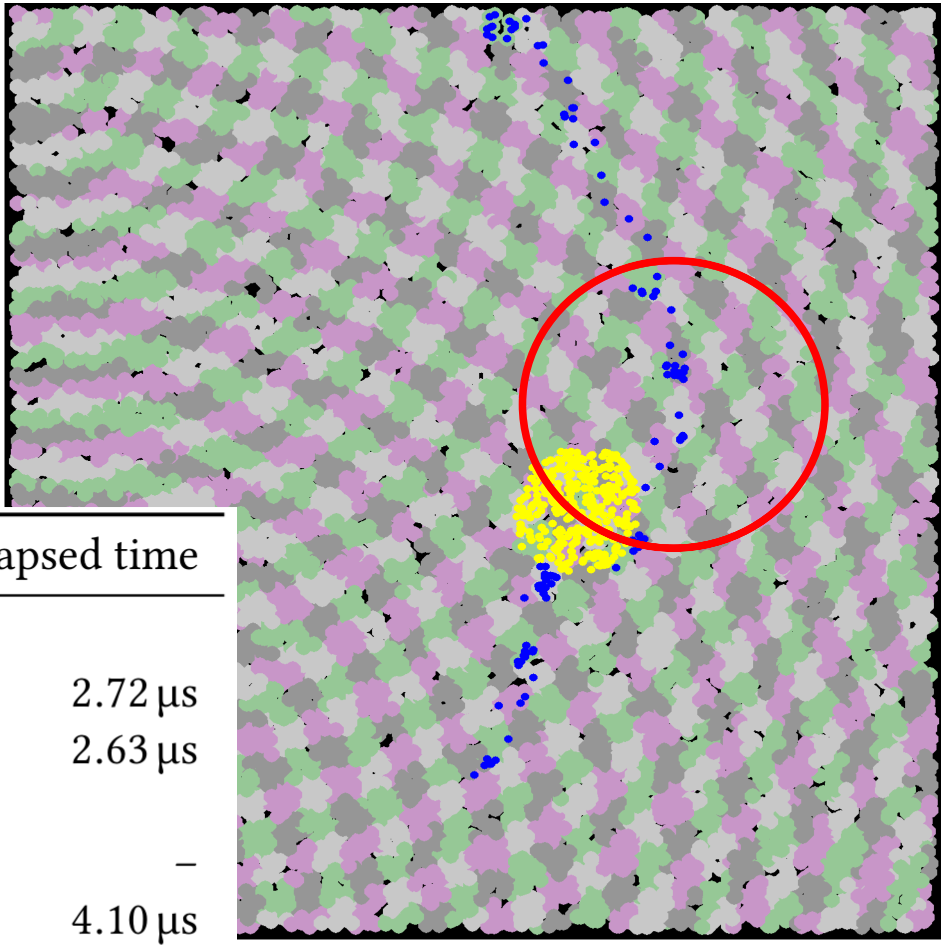
- It works in 2D networks, but rapidly reaches limits when no uncongested areas are available to deviate to
- Many nanonetworks (for example in programmable matter field) are in 3D
- In 3D, many more paths are available to deviate to, and the algorithm becomes much more efficient



Deviation routing: Congestion avoidance

Then, when the congestion gets below the CI threshold, the route automatically gets back to the normal route

This congestion control mechanism acts before packet loss. It prevents packet from being dropped by using resources in another part of the network



	Avg. packets sent	Avg. elapsed time
Without congestion:		
Modified SLR	96.4	2.72 µs
Deviating SLR	87.3	2.63 µs
With congestion:		
Modified SLR	–	–
Deviating SLR	135.0	4.10 µs

Conclusion

Two contributions

Congestion detection that takes into account TS-OOK specificities

It can be used separately from the deviation to serve others congestion control mechanisms

Route deviation based on SLR

It can be used separately from the congestion detection (hole avoidance for instance)

Might be very powerful in 3D...

6. Duty cycling

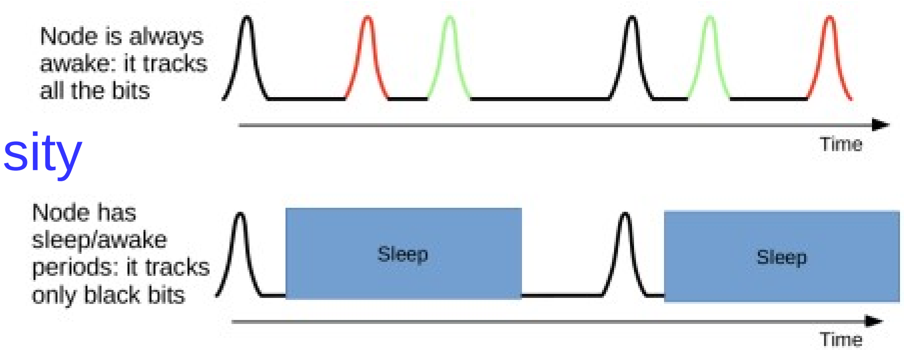
The problem: packet interleaving

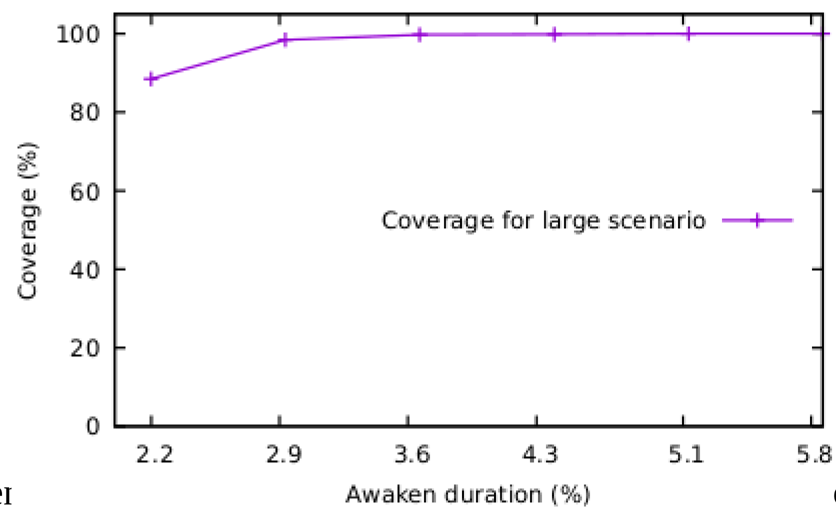
- TS-OOK allows a lot of packets to interleave
  - $\beta = 1000$ (commonly used value)

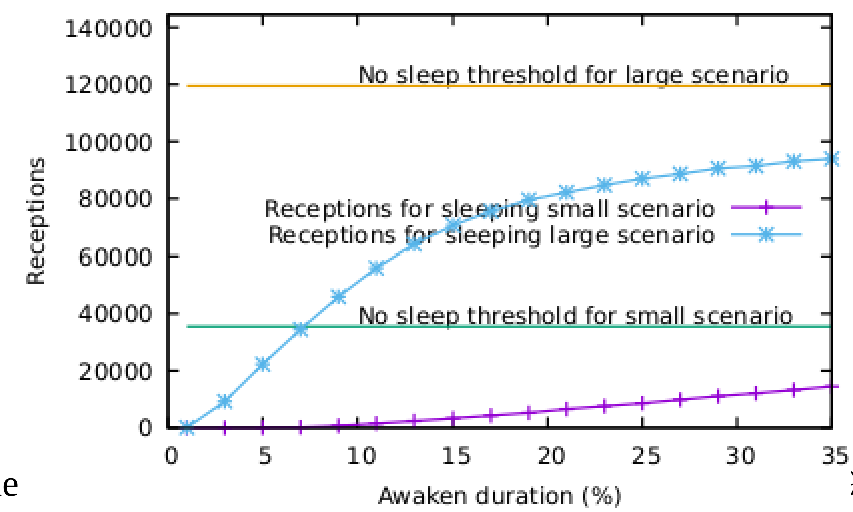
- Nodes have limited resources (memory, CPU, hardware constraints), they cannot store all the packets in-flight: maxConcurrentReceptions
  - even just the time to decode them

Proposed solution

- TDMA-like, but fine-grained
- Random sleeping period
- Awaken duration depends on node density
- Different from sleeping in WSN
- Avoid memory congestion
- Choose awaken duration so that coverage/connectivity is preserved

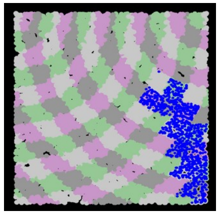




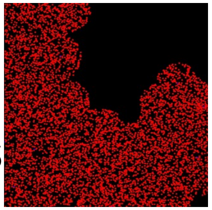


Conclusion

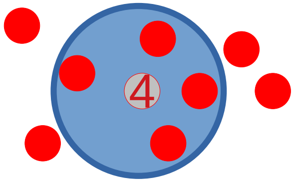
Putting together nano group's contributions...



- Need a multi-hop network, which implies a routing protocol
- Context of dense networks => complex system, so need a scalable simulator to understand phenomena



- Routing/flooding => many nodes receive the packet => numerous collis
=> backoff flooding



- The optimal window of backoff flooding depends on the number of neighbours => density (number of neighbours) estimation
- We reduced the number of exchanged packets (senders), but the number of receivers is still high (**all** the intermediary nodes receive the message) => they need a lot of memory to store messages => node sleeping
- All this was for only one initial packet sent => need a congestion control => deviation/routing-based congestion control

Conclusion

- Emerging topic: "We are lucky to live in an age in which we are still making discoveries" (R. Feynmann 1964...)
- Communication in electromagnetic nanonetworks
- **Very** different than classical networks, have unique features:
  - high density
  - tiny size, energy, capabilities
  - packet overlapping
  - sending 1 consumes energy, 0 is silence
  - different collisions, influence by propagation delay and receiver position
  - => need of novel algorithms and protocols
- IoT, complex systems, nanonetworks, routing and transport layers

Focus on network and transport layers, and on dense networks

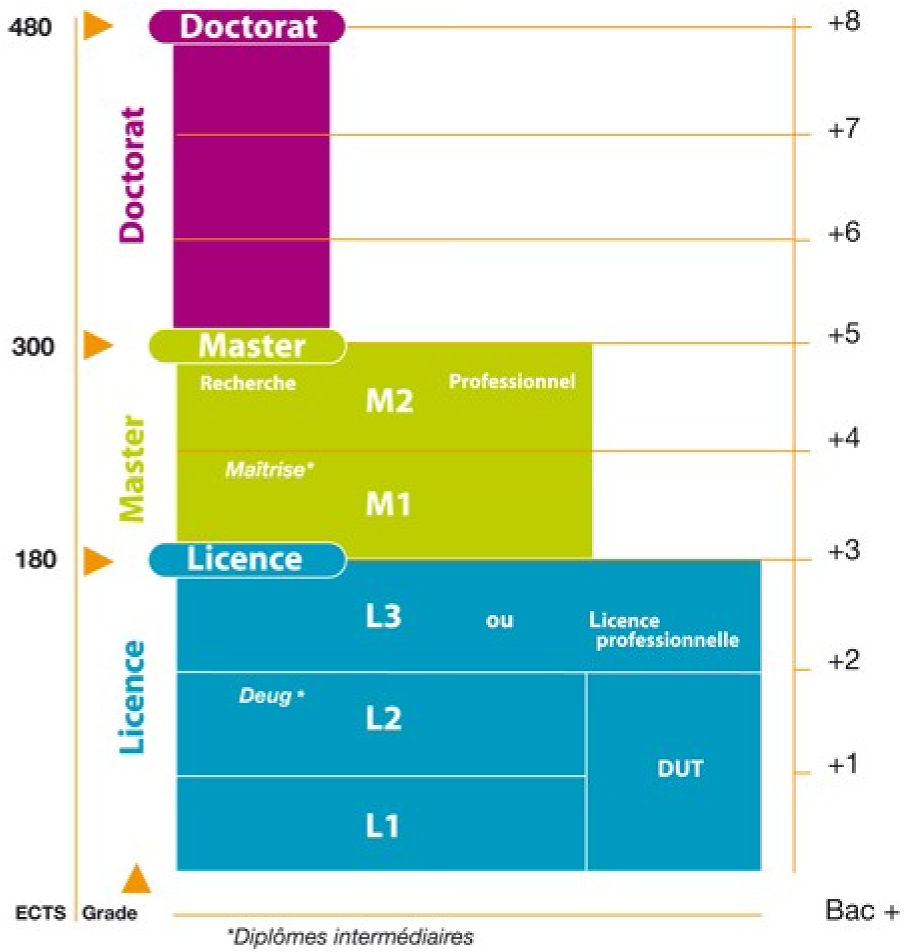
- simulator (BitSimulator)
- broadcasting: backoff flooding
- routing protocol (SLR)
- estimation of number of neighbours
- node sleeping to reduce memory
and energy used
- deviation-based routing

Perspectives

- Enumerate congestion control principles in nanonetworks
  - main limitation: memory, not bandwidth!
  - analyse deviation for 3D networks and numerous concurrent flows
- Density estimator:
  - continuous estimation
  - reduce packet overhead
  - apply to other fields
- Backoff flooding (Trickle) with sleep
- Communication patterns & application to programmable matter
- Nanogeopositioning
  - high precision/accuracy (nm?)
- Infrastructure-free routing
  - nodes are address-free
  - routing by functionality (temperature, pressure etc.)
- Reception
  - discovery of start and end of packet
  - for different betas
- Analyse influence of beta on communications
  - communications with different betas
- BitSimulator:
  - energy model, energy harvesting and communication limitations due to energy
- Dissemination of results to public:
  - 3D visualisation tool, using LEDs
  - demonstrator of dense networks
- Go with dense (massive) networks: numerous neighbours, numerous concurrent communications, numerous hops
  - enlarge application field to known dense networks: 5G (mMTC), UAV (drones), HPC
  - platform for real experiments

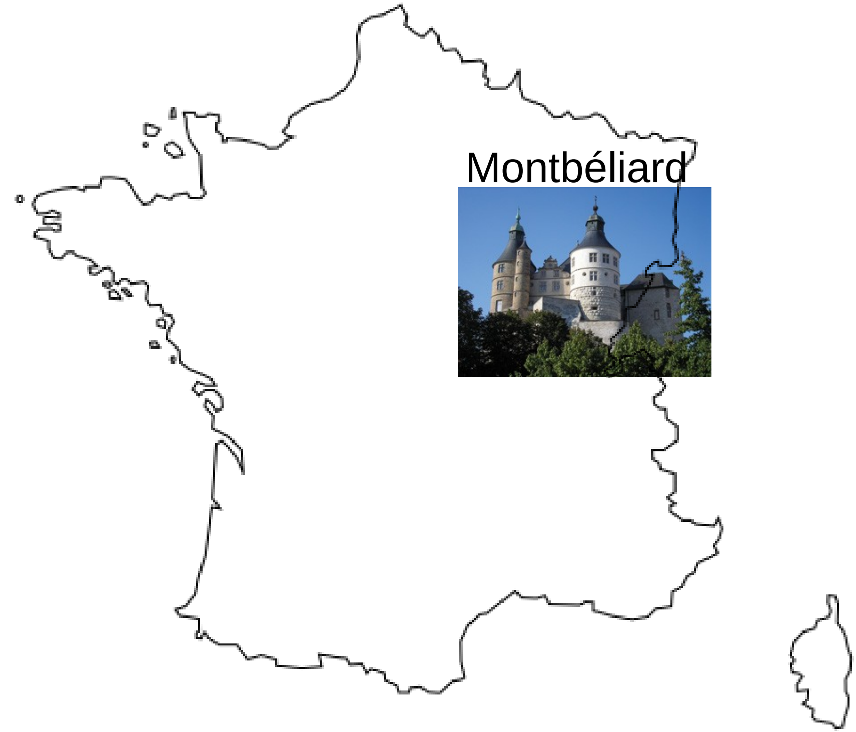
Master degree

- IoT master, M1 and M2, in English
- M1: October->June
  - Mobile development, Infrastructure and routing for connected objects, Data mining, Team management and communication, English, Radio networks, Positioning systems: techniques and applications, Embedded systems, Cloud infrastructure and virtualization, Mini project at the lab
- M2: Oct.->Jan. + internship February->August
  - choose 2 blocks out of 3:
  - Deep learning for IoT, Security for connected objects
  - Mobility in smart cities, Modular robots programming and swarm robotics
  - Agent-based Modeling and Simulation for IoT, Perception and interactions for IoT (with GUI)
- http://www.ubfc.fr/en/master-iot



Life

- FEMTO-ST Institute: biggest ST lab in France, affiliated to national research center (CNRS)
  - emphasises publication quality
- Montbéliard or Belfort: towns (~100,000 people) in East of France, near Germany and Switzerland
- University campuses: 2500 students each



mixer tr.25-28

Nouveau transparent money:
Until 2012, through its National Nanotechnology Initiative, the USA has invested 3.7 billion dollars, the European Union has invested 1.2 billion and Japan 750 million dollars (w nanotechnology)
TODO: compare with Alzheimer's disease funding, and others

Transparent nanoantenna:
- une des côtés est en nano ????
- ??? radiation in THz band is difficult to produce (THz gap), however such emitters do exist: CNT, canadianca, ... ???

- THz band is not regulated (yet?)
- app: kiosk downloading (file server), backhaul, data center network??, board to board communication (computer2computer, phone2phone etc.)
- quite often direct path may be blocked, see Dominique...

Other members:
- Julien Bourgeois, Pr. – MAC protocols and coding
- Hakim Mabed, Assoc. pr. – MAC protocols
- Lina Aliouat, PhD student – MAC protocols
- Past members:
  - Nicolas Boillot – group behaviour, simulator

Article co-authoring: C. Liaskos, A. Tsioliaridou, F. Büther

2D or 2D map
short/long term
difficulty
apply (BS) or research etc.