University of Pavia
Pavia, Italy
Fall, 2017

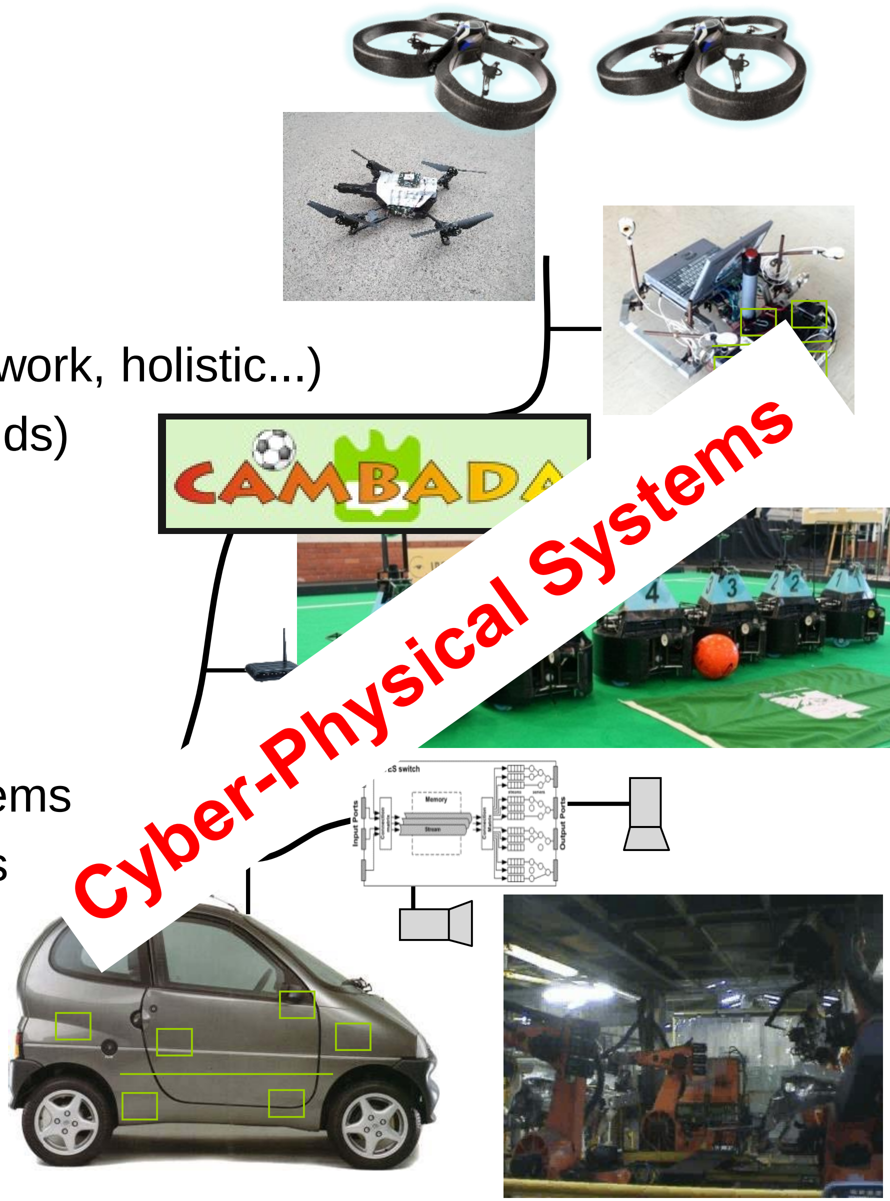
Coordination of Autonomous Mobile Nodes

Mobile Cyber-Physical Systems

Luis Almeida

www.fe.up.pt/~lda
(IT) Instituto de Telecomunicações – Porto
(DaRTES) Distributed and Real-Time Embedded Systems Lab
(FEUP) Faculdade de Engenharia – **Universidade do Porto**, Portugal

My background



- **Real-time systems:**
  - Scheduling aspects (processor, network, holistic...)
  - RT communication protocols (all kinds)
- **Dependable systems:**
  - Safety and reliability aspects
  - Error handling in communications
- **Main *battle fields*:**
  - Wired networks for embedded systems
  - Wireless comms for teams of robots
  - Dynamic reconfiguration
  - On-line QoS adaptation

F-T-T-
Flexible Time-Triggered architecture

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Some pointers to our work

- Luis Ramos Pinto, Luis Almeida, Hassan Alizadeh, Anthony Rowe. Aerial Video Stream over Multi-hop using Adaptive TDMA Slots. RTSS 2017 – 38th IEEE Real-Time Systems Symposium. Paris, France. 5-8 December 2017.
- Sidney Carvalho, Luis Pinto, Luis Almeida, Ubirajara Moreno. Improving Robustness of Robotic Networks using Consensus and Wireless Signal Strength. TA 2016 – 4th IFAC Symposium on Telematics Applications. Porto Alegre, Brazil. 6-9 November 2016.
- J. Aisa, H. Fotouhi, L. Almeida, and J.L. Villarroel. DoTHa - A Double-threshold Hand-off Algorithm for Managing Mobility in Wireless Mesh Networks. ETFA 2016 – 21st IEEE Int Conf Emerging Tech. and Factory Automation. Berlin, Germany. Sept 2016.
- Luis Ramos Pinto, Luis Oliveira, Luis Almeida and Anthony Rowe. Extendable Matrix-Camera using Aerial Networks. ICARSC 2016 - 10th IEEE International Conference on Autonomous Robot Systems and Competitions, Vila Real, Portugal. 4-6 May 2016.
- L. Almeida, F. Santos, L. Oliveira. Structuring Communications for Mobile Cyber-Physical Systems. In Management of Cyber Physical Objects in the Future Internet of Things: Methods, Architectures and Applications. A. Guerrieri, V. Loscri, A. Rovella, G. Fortino (Eds), Springer, Series on Internet of Things, Vol. 1: ISBN 978-3-319-26867-5, 2016. (DOI: 10.1007/978-3-319-26869-9)
- Daniel Ramos, Luis Oliveira, Luis Almeida, Ubirajara Moreno. Network Interference on Cooperative Mobile Robots Consensus. ROBOT 2015: 2nd Iberian Robotics Conference. Lisboa, Portugal. 19-21 November 2015.
- Luis Oliveira, Luis Almeida, Pedro Lima. Multi-hop routing within TDMA slots for teams of cooperating robots. WFCS 2015 – 11th IEEE World Conference on Factory Communication Systems. Palma de Mallorca, Spain. 27-29 May 2015.
- P. Lima, A. Ahmad, A. Dias, A. G. S. Conceição, A.P. Moreira, E. Silva, L. Almeida, L. Oliveira, T. P. Nascimento. Formation Control Driven by Cooperative Object Tracking. Robotics and Autonomous Systems (ISSN: 0921-8890), 63(P1):68-79. Elsevier, Jan 2015.
- B. Ordoñez, U. F. Moreno, J. Cerqueira, L. Almeida. Generation of trajectories using predictive control for tracking consensus with sensing and connectivity constraint. In Cooperative Robots and Sensor Networks, Anis Koubaa, Abdelmajid Khelil (Eds.), Springer, Series on Studies in Computational Intelligence, Vol. 507:19-37. ISBN 978-3-642-39300-6, 2014. (DOI:10.1007/978-3-642-39301-3)
- L. Oliveira, H. Li, L. Almeida, T. E. Abrudan. RSSI-based Relative Localisation for Mobile Robots. Ad Hoc Networks (ISSN 1570-8705), 13-B:321-335. DOI:10.1016/ j.adhoc.2013.07.007. Elsevier. February 2014.
- L. Oliveira, C. Di Franco, T. E. Abrudan, L. Almeida. Fusing Time-of-Flight and Received Signal Strength for Adaptive Radio-Frequency Ranging. ICAR 2013 -16th Int. Conf. on Advanced Robotics. Montevideo, Uruguay. 25-19 Nov 2013.
- D. Tardioli, L. Almeida, J. L. Villarroel. Adding alien traffic endurance to wireless token-passing real-time protocols. APCSS 2010, 5th IEEE Asia-Pacific Services Computing Conf. Hangzhou, China. 6-10 Dec 2010
- A.J.R. Neves, J.L.Azevedo, B.Cunha, N.Lau, J.Silva, F.Santos, G.Corrente, D.A. Martins, N.Figueiredo, A.Pereira, L.Almeida, L.S. Lopes, A.J. Pinho, J.Rodrigues, P. Pedreiras. CAMBADA soccer team: from robot architecture to multiagent coordination. in Robot Soccer, Vladan Papić (ed), INTECH, pp:19-46. ISBN 978-953-307-036-0, January 2010. (DOI: 10.5772/7353)
- H. Li, L. Almeida, Y. Sun. Dynamic Target Tracking with Integration of Communication and Coverage using Mobile Sensors. IECON 2009, 35th Conf of the IEEE Ind. Electronics Soc., Porto, Portugal, 3-5 Nov 2009.
- F. Santos, L. Almeida, L.S. Lopes, J.L. Azevedo, M.B. Cunha. Communicating among robots in the RoboCup Middle-Size League. RoboCup Symposium 2009, Graz, Austria. June 29-July 5, 2009 (LNCS 5949, Springer 2010).
- T Facchinetti, G. Buttazzo, L. Almeida. Dynamic Resource Reservation and Connectivity Tracking to Support Real-Time Communication among Mobile Units, EURASIP J. on Wireless Comm. and Networking, 2005(5):712-730, Dec 2005.

Acknowledgments

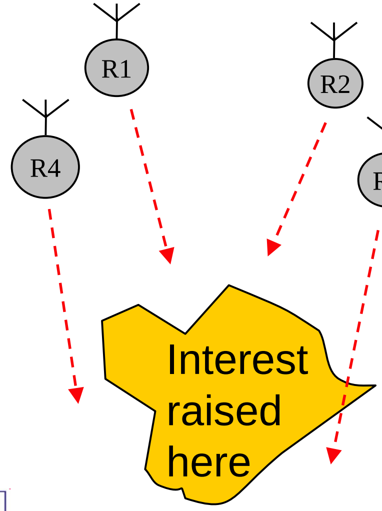
The work that supports this course involved several direct and indirect contributors among which I would like to acknowledge

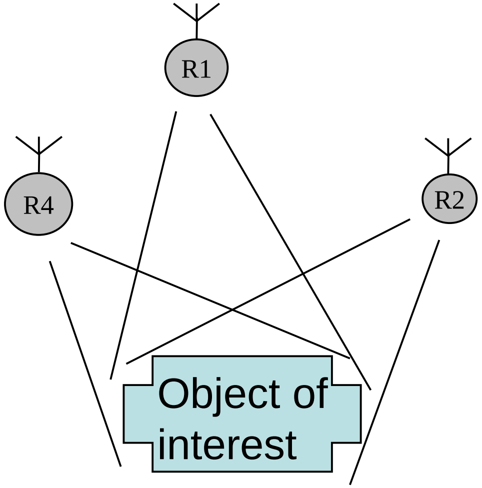
Luis Pinto, Sydney Carvalho, Daniel Ramos, Carmelo di Franco, Shuguo Zhuo, Bernardo Ordoñez, Danilo Tardioli, Shantao Chen, Luís Oliveira, Ana Ponte, Hongbin Li, Zhi Wang, Tullio Facchinetti, Frederico Santos, the CAMBADA (Aveiro) and 5DPO (Porto) RoboCup MSL teams and the PCMMC project team

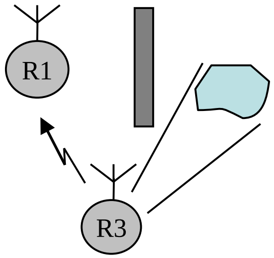
Mobile Cyber-Physical Systems

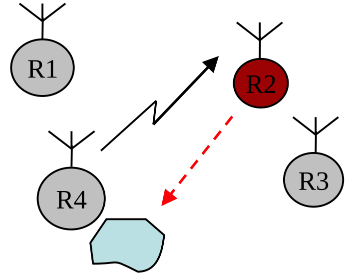
- **Cyber-Physical Systems (CPS)**
  - Anything that involves computer-environment interaction !

- **Mobile CPS = CPS + moving sensors / actuators**
  - Robust sensing
  - Cooperative sensing & control
  - Efficient actuation, ...





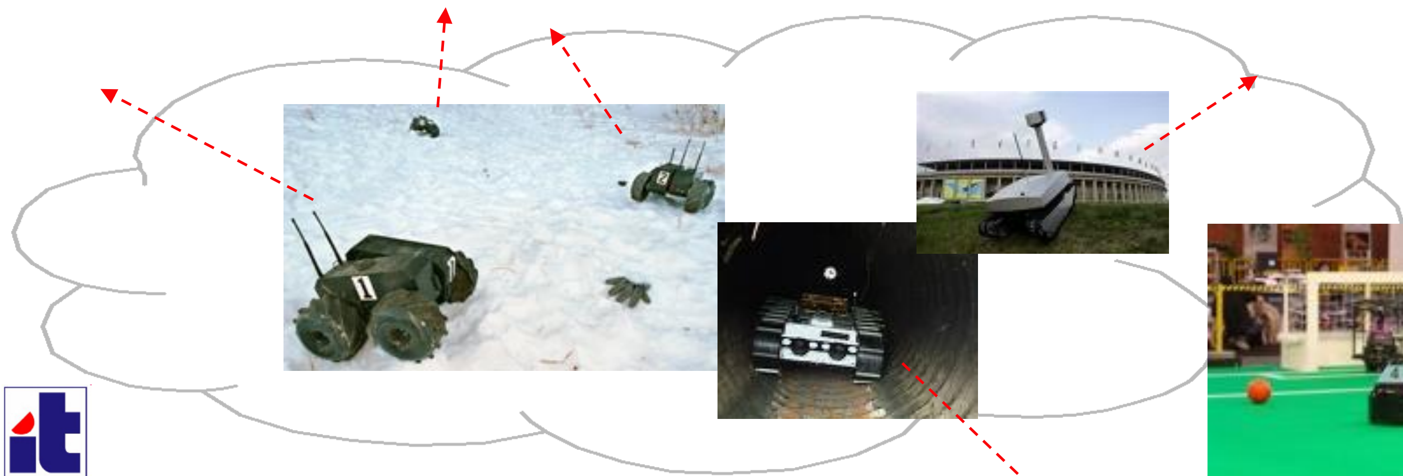




Pictures taken from diverse websites

Mobile Cyber-Physical Systems

- **A desirable target**
  - To deploy a **team** of **heterogeneous autonomous agents,** provided with appropriate sensing and actuating capabilities, in a given operational scenario, which are capable of **cooperating towards a global goal without needing** any
    - **specific** agents **configurations**
    - previous **environment preparation**

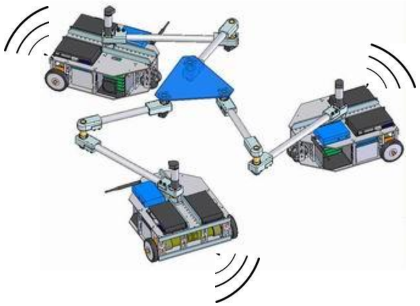


Pictures taken from diverse websites

Robotica 2008, Aveiro, Portugal

Architectural issues

- **Number of nodes: variable**, normally small (<20)
- **Network topology:** with (**star**) / without (**mesh**) infrastructure
  - Must be **clustered** for larger numbers of nodes
- **Synchronization:** Beneficial for **communications** and **actions**
- **Dynamic membership:** notion of **team**
- **Inter-agent information sharing:** normally **global** in the team
- **Location-awareness: relative** (no anchors) /absolute if possible
- **Combination of** coordinated with autonomous **behaviors**
- **Communication medium: wireless!**



Pictures taken from diverse websites

At the team level

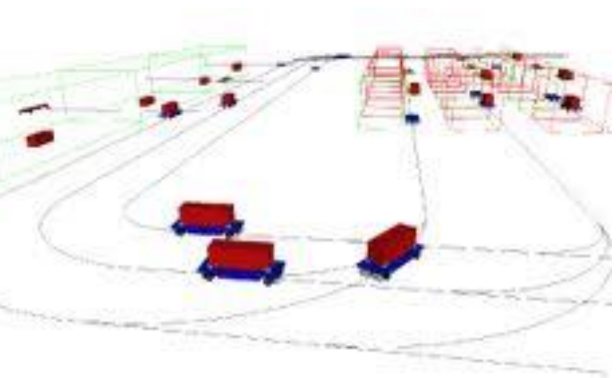
- **Coordination must be adaptive and tolerant to**
  - **communication unavailability**
  - **changes in team composition**

- **Some useful techniques**
  - Use **dynamic role** assignment
  - Define set of **safe autonomous behaviors**
    - Guarantee **safe switch** to such states when needed
  - Use **positioning** to control **connectivity** / **topology**
  - Use **synchronization** for effective **communications** and **actions**

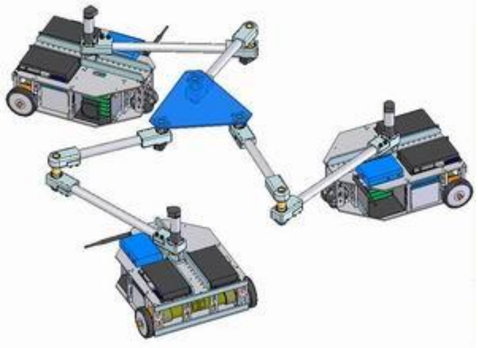
Wireless communications

- **Openness**
  - Ad-hoc connections based on proximity
  - Prone to intrusion and denial-of-service
- **Freedom of relative movements**
- **Bit error rate higher than with wired comm.**
- **Potential for unavailability periods**
  - Other uncontrolled traffic, interference, ...
- **Common real-time assumptions** of **bounded delays, connectivity, medium availability** and **cooperative environment**
  - Typical in wired systems
  - Have **lower coverage in wireless** systems

Communication requirements

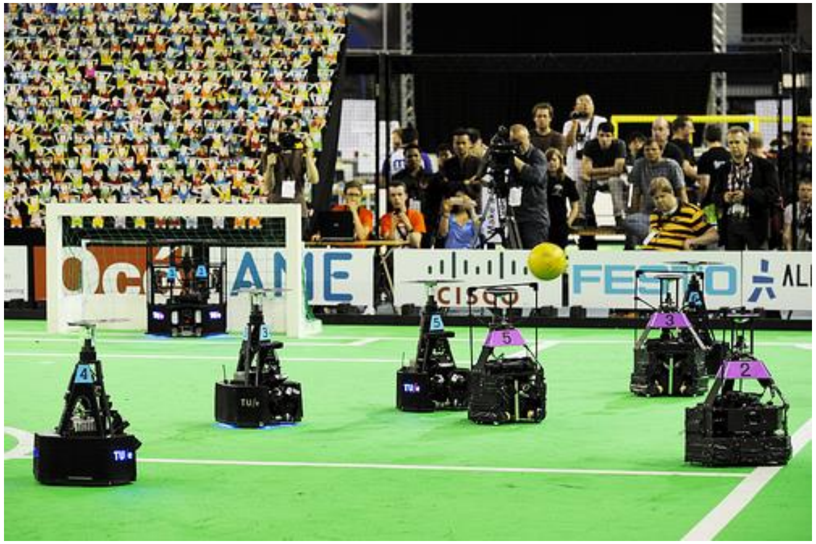


- **Sharing state + sensing**
  - Periodic short/medium size data
- **Communicating events**
  - Aperiodic short data
- **Streaming multimedia**
  - Periodic medium/long data



In this course

- **Building blocks** for Coordinating Autonomous Mobile Nodes
  - Wireless communications
  - Information dissemination and access
  - Clock synchronization and synchronization of communications
  - Relative localization
- Examples of **communication protocols**
  - Reconfigurable and Adaptive TDMA
  - RT-WMP
- Examples of **coordination**
  - Coordinated sensing and tracking
  - Distributed consensus
  - Robotic soccer (RoboCup)



RoboCup 2013, Eindhoven, Netherlands

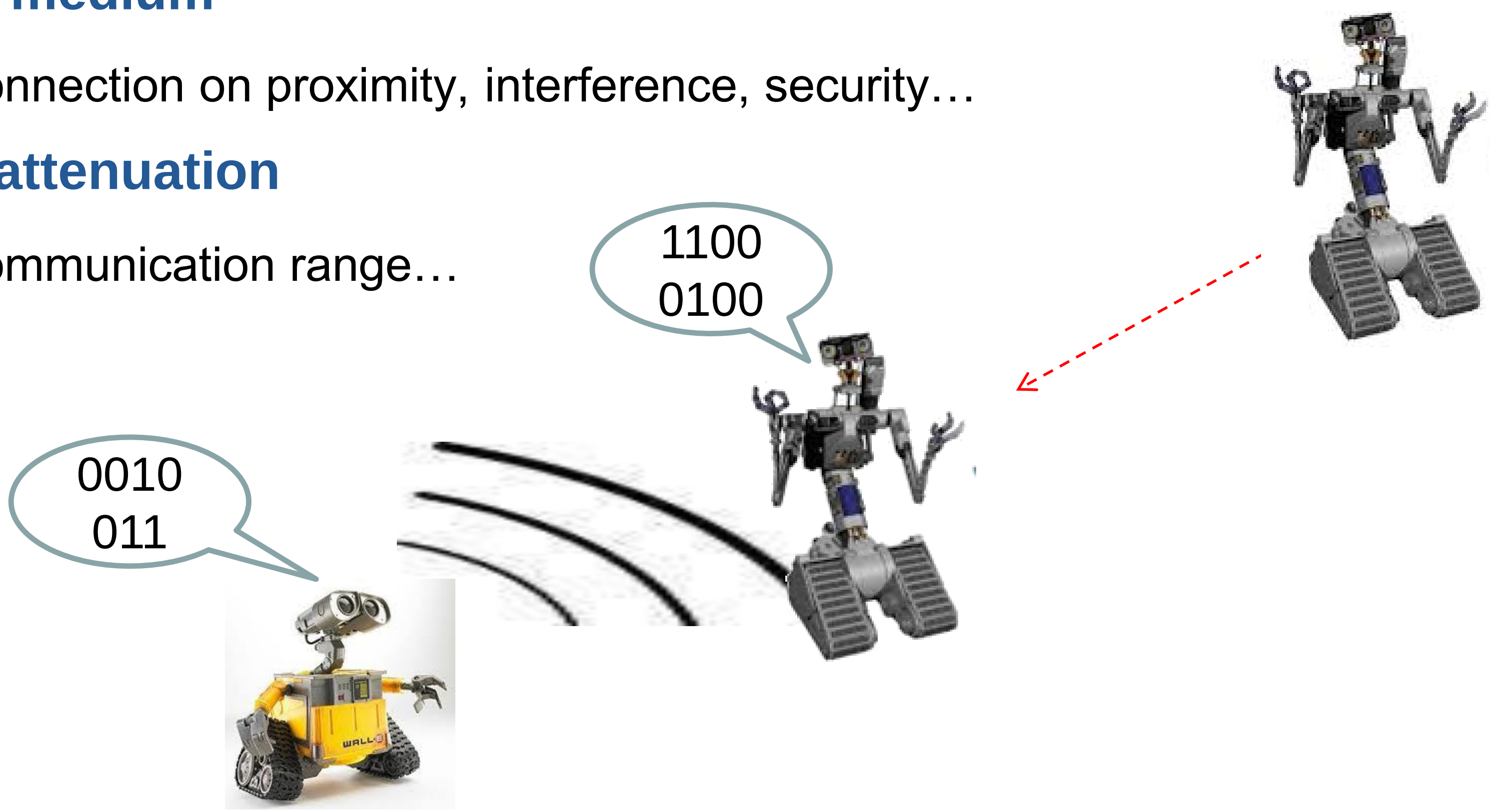
Basic concepts in wireless communications

wireless media

- (infra-red) light,
- (ultra-) sound,
- **radio-frequency**
- …

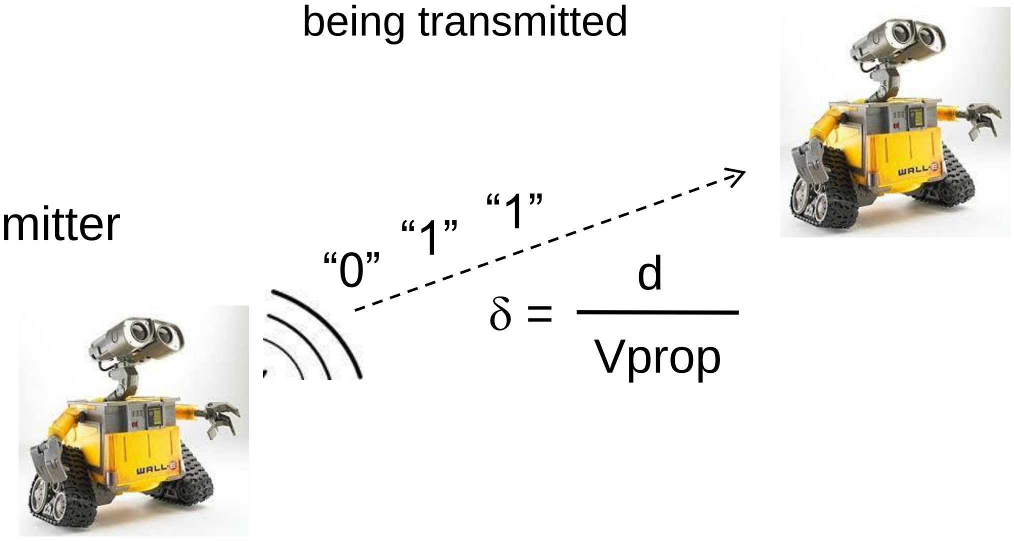
Particularities of wireless communication

- **Freedom from physical connection**
  - No cabling, flexible deployment, mobility…
- **Open medium**
  - Connection on proximity, interference, security…
- **High attenuation**
  - Communication range…



Radio-frequency transmission

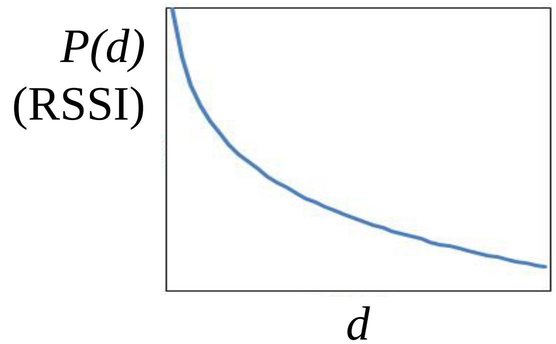
- **Electromagnetic wave**
- Propagates at about **3.3ns/m** in the air
  - **Propagation delay δ**



$$\delta = \frac{d}{Vprop}$$

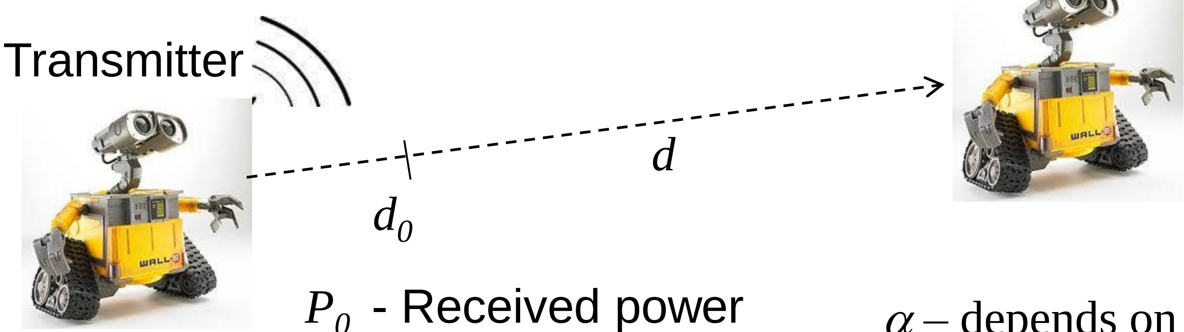
Free-space attenuation model

- **Received power** varies with **log of distance**



Power at receiver (dBm)
(**RSSI:** Received Signal Strength Indicator)

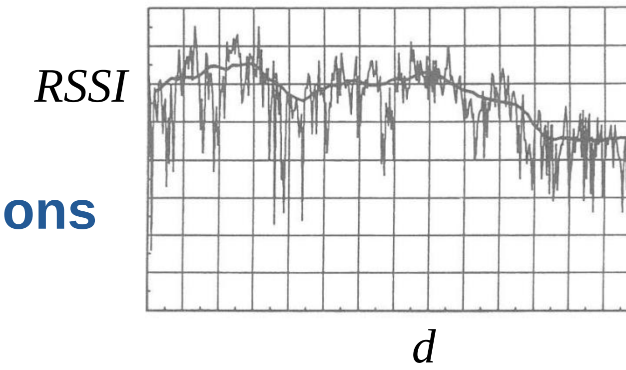
$$P(d) = P_0 - 10\alpha \log\frac{d}{d_0} + X_\sigma$$



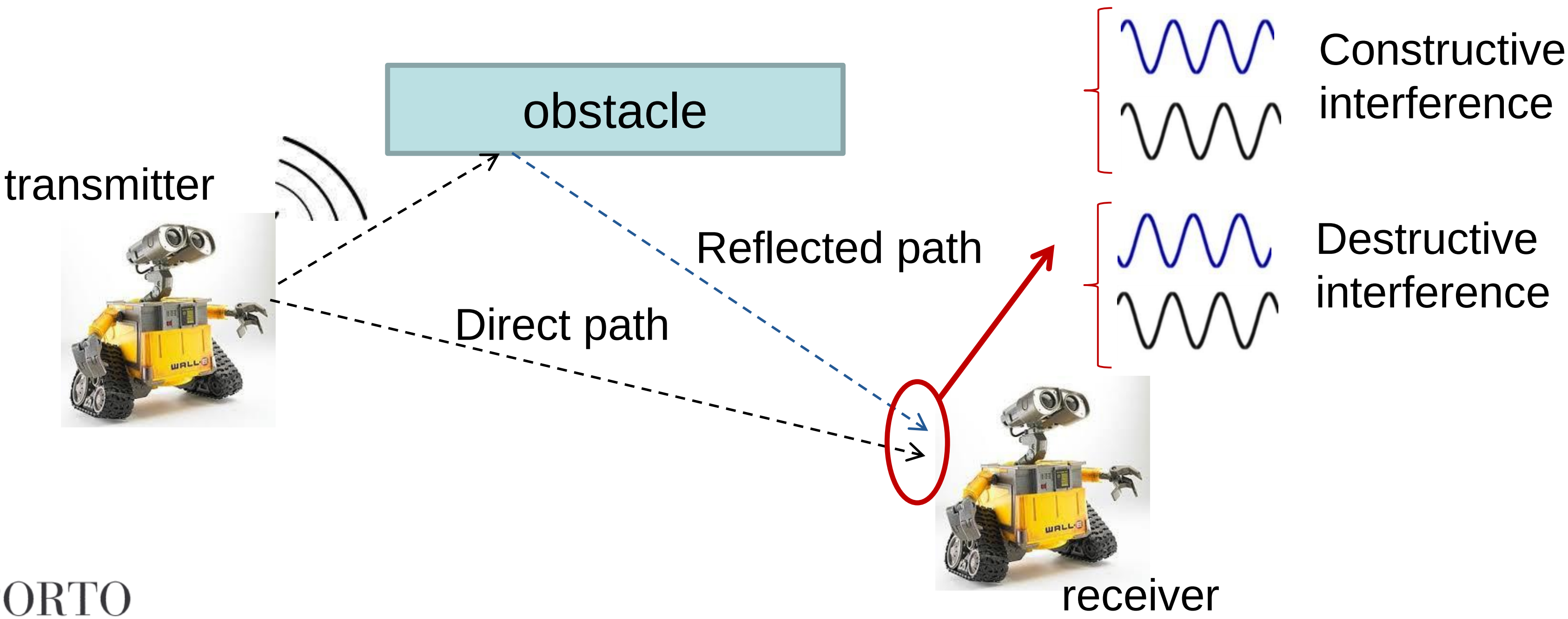
P_0 - Received power at reference d_0 (dBm)

α – depends on environment (~1-4)
X_σ – noise parameter

Attenuation in closed spaces

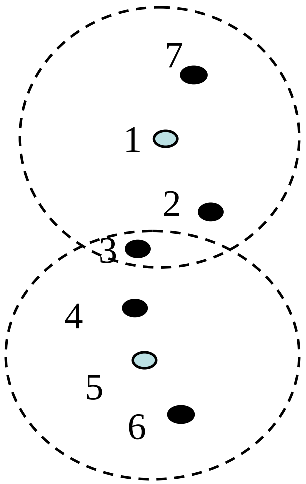


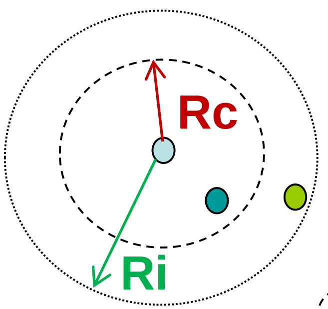
- **Depends** too much on the **environment**
- Suffers from **multi-path fading** and **obstructions**
  - **Deteriorates power-distance** relationship
  - Varies strongly with materials, geometry, ambient parameters…

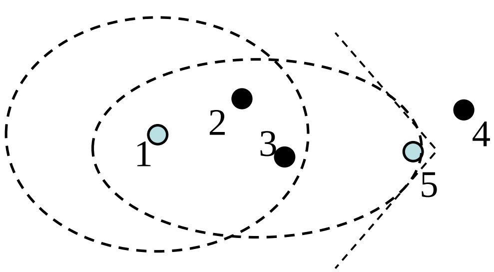


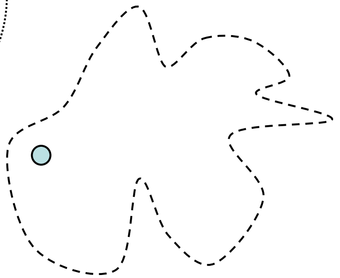
Consequences of attenuation

- **Communication range**
  - Receiver **cannot decode** transmission after **distance Rc**
  - **Rc** varies with **direction, Rx Tx antennas, modulation, electronics**
  - Cause of **hidden-nodes** and **spatial channel reuse**
- **Interference range**
  - Receiver **cannot detect** transmission after **distance Ri**
  - Transmissions are **detected** but **not received**
- **Impact of antennas**
  - Anisotropic → directional
  - Asymmetric links





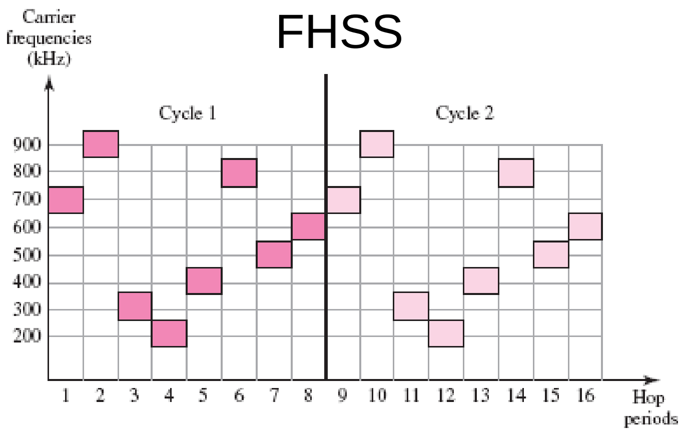




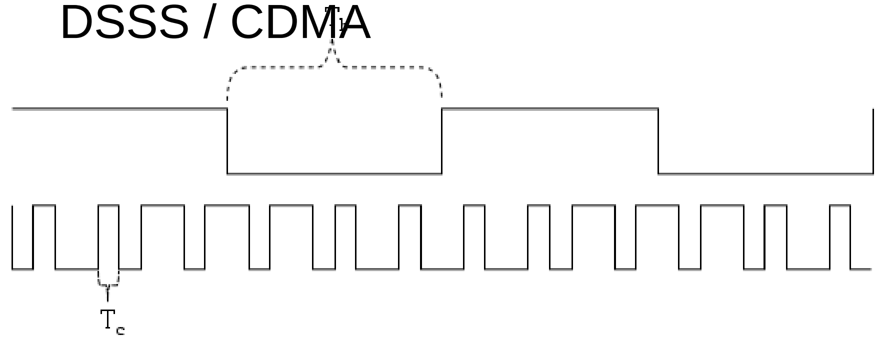
Modulation techniques

- Allowed **huge gains** in **bit rate** and **reliability**

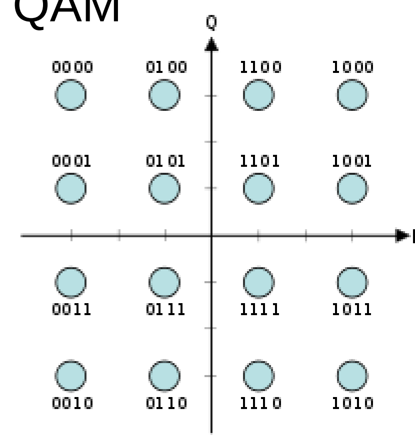
FHSS



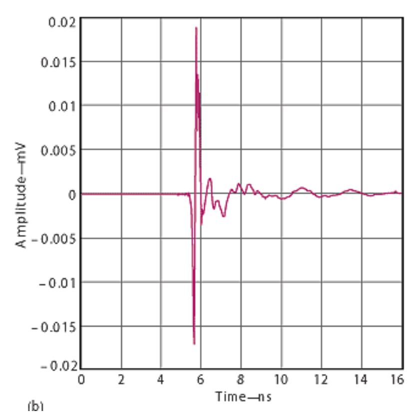
DSSS / CDMA



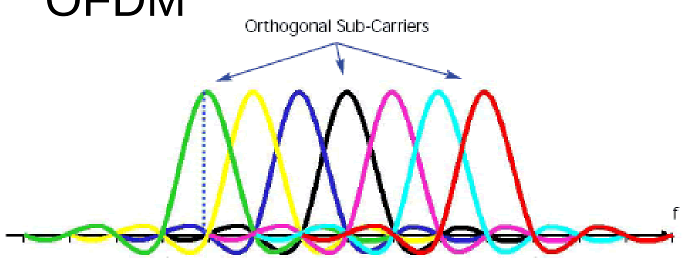
QAM



UWB



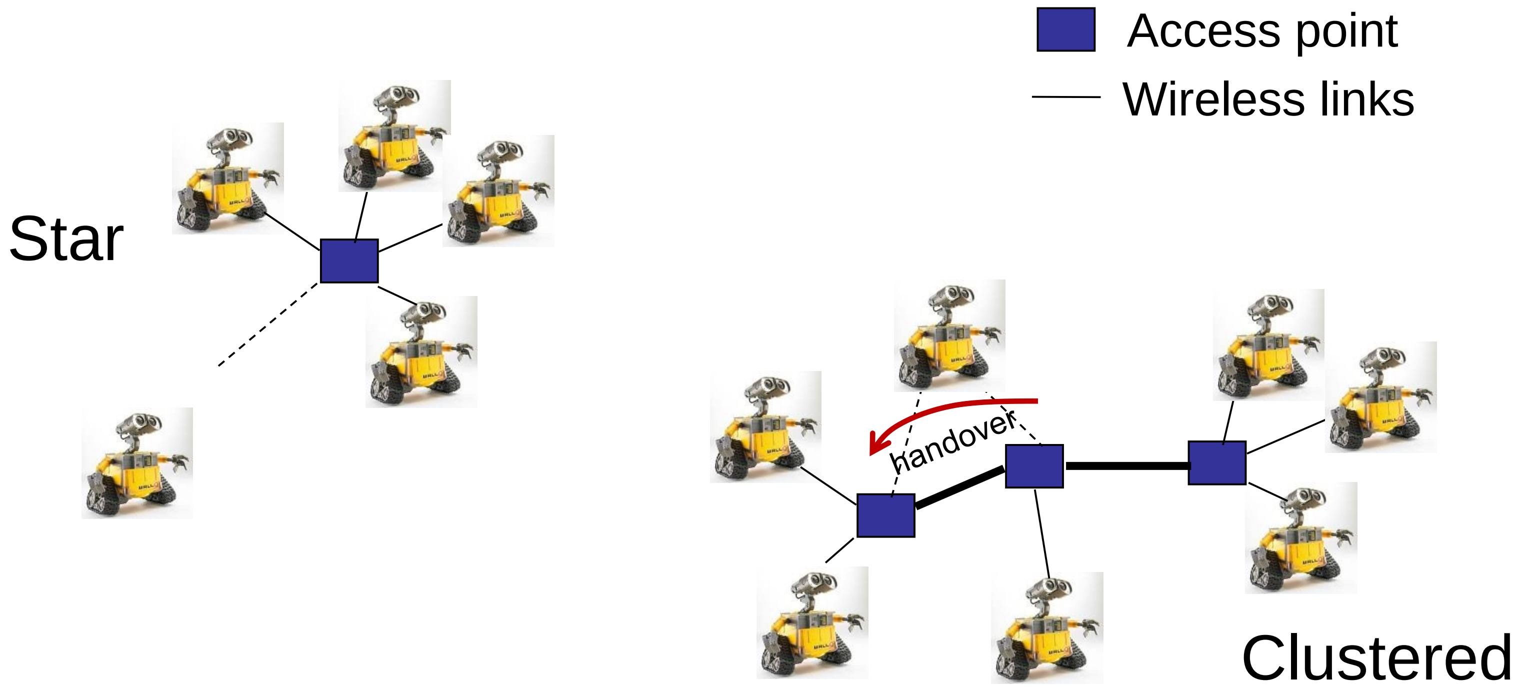
OFDM



Figures taken from diverse websites

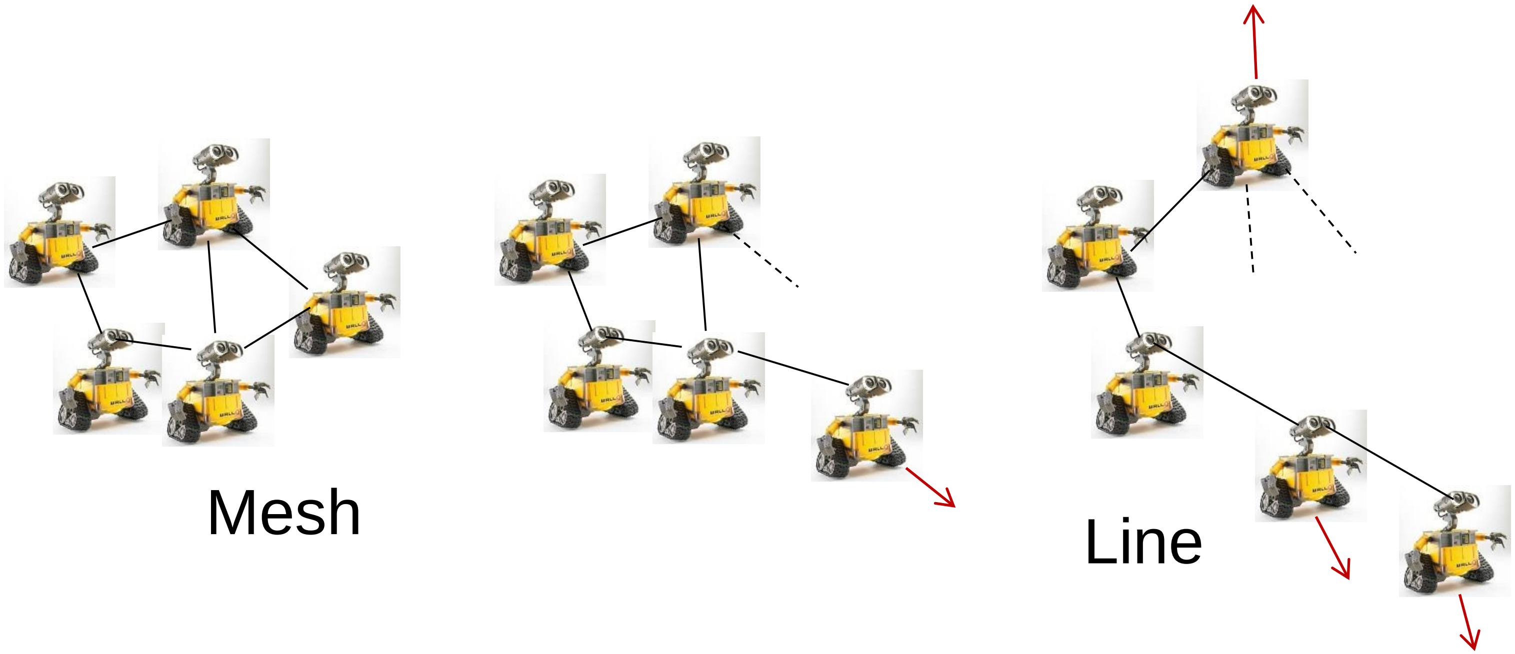
Network topology

- **Infrastructured**
  - **Star:** Defines team
  - **Clustered:** Larger reach, requires hand-over and routing across clusters



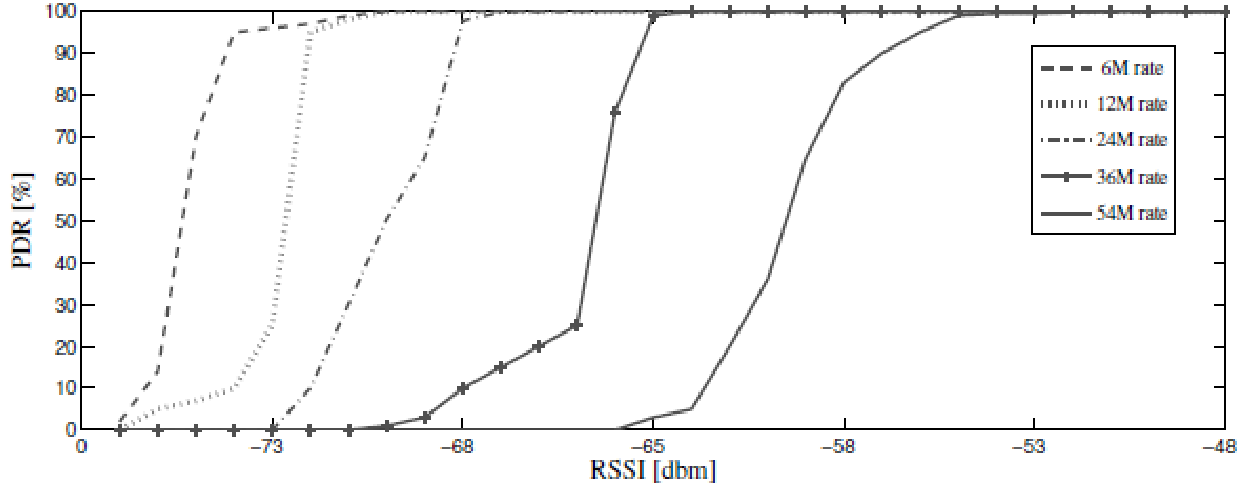
Network topology

- **Ad-hoc**
  - **Mesh**
    - Mobility → **Flexible topology**, reach versus redundancy
    - **Multi-hop** communication: Routing? Information dissemination?



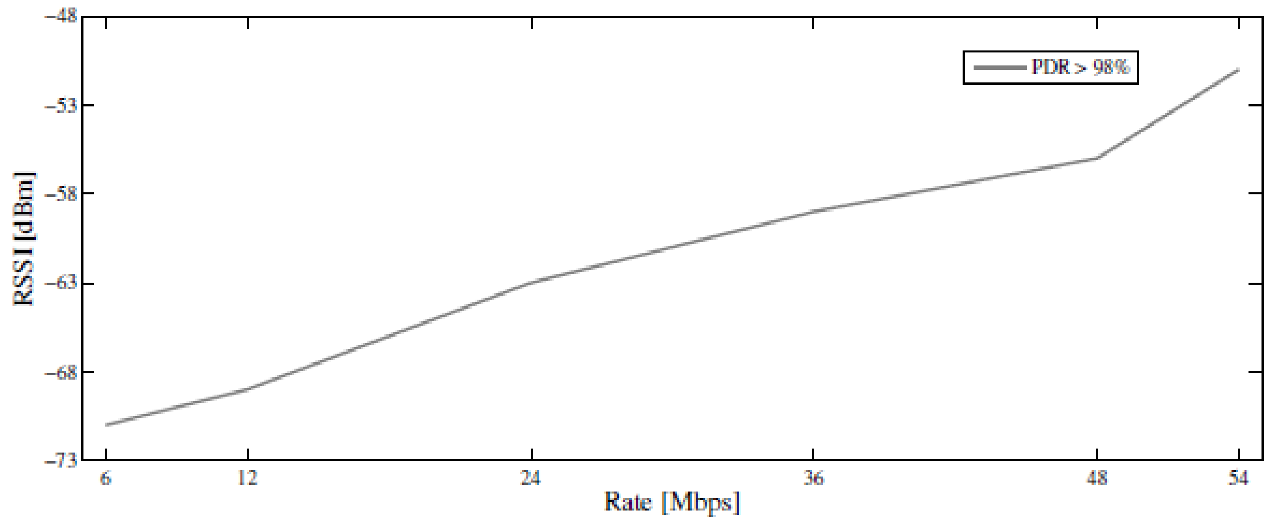
Errors, power and bit rate

IEEE 802.11a



For a given power,
higher bit rate
implies
higher error rate

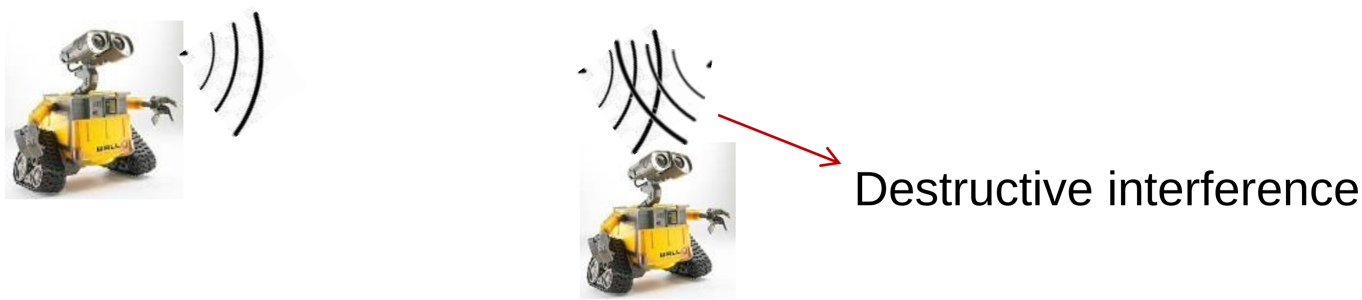
For a given error rate,
higher bit rate
requires
higher power



D. Sicignano, Univ. Zaragoza, Spain

Collisions in transmissions

- **Different transmissions** may **collide** in the receiver
  - Transmitters **not aware** of each other
    - Situation known as ***hidden nodes***
      - the transmitters are *hidden* from each other

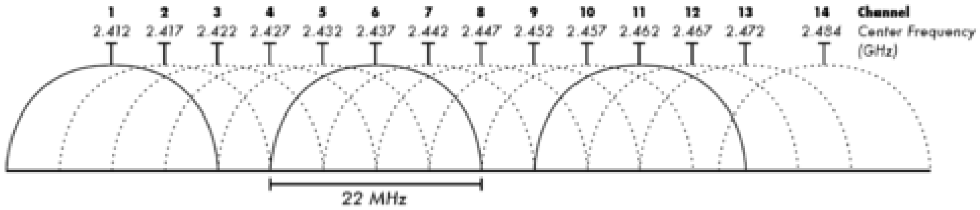


- **Detecting collisions** is the basis for **back-off and retry**
  - Sensed indirectly by means of **acknowledging**
    - **Ack received → OK**
    - **Ack not received → assume NOK and retry**

Accessing the shared medium

- Sharing in the **frequency domain**
  - **Transmissions go in parallel** using different frequency channels

Frequency Division Multiplexing



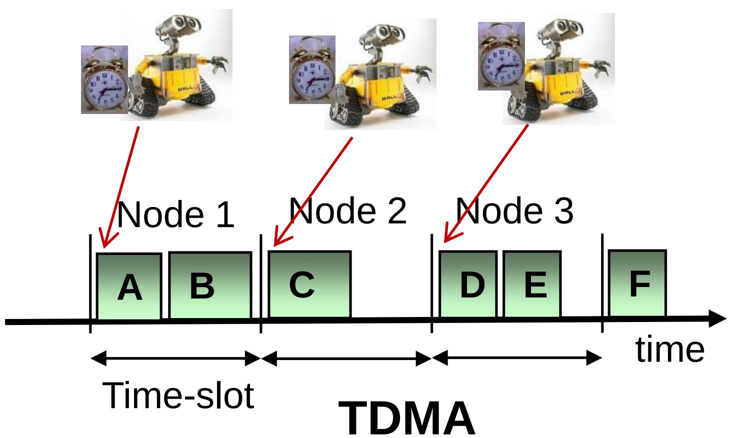
Frequency spectrum in 802.11b

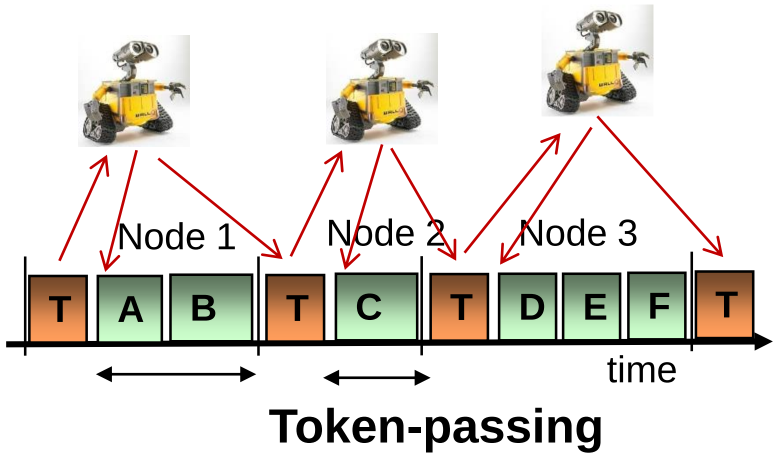
- Sharing in the **time domain**
  - Transmissions are **sequenced in time**
    - **Scheduled** (controlled) access
    - **Arbitrated** (uncontrolled) access
    - Hybrid control

Time Division Multiplexing

Medium access control

- **Scheduled (controlled) access**





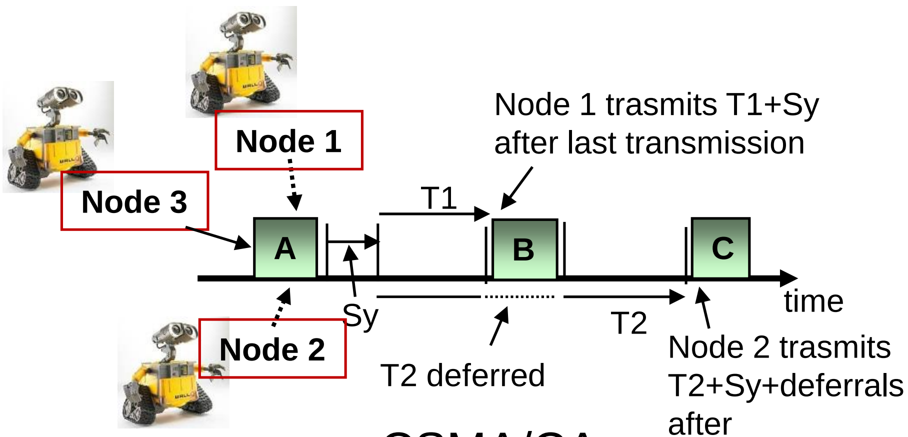
- Requires (clock) synchronization
- Slot are isolated in time
  - Even when nodes can go out of the group and back
- A slot is a dedicated fraction of the network
  - Real-time guarantees
  - Each slot is a periodic/polling server

- Many transmissions
  - the token is always going on
- High jitter in periodic transmissions
- Allows building prioritized access
  - Token round allows reaching consensus
  - Real-time guarantees

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Medium access control

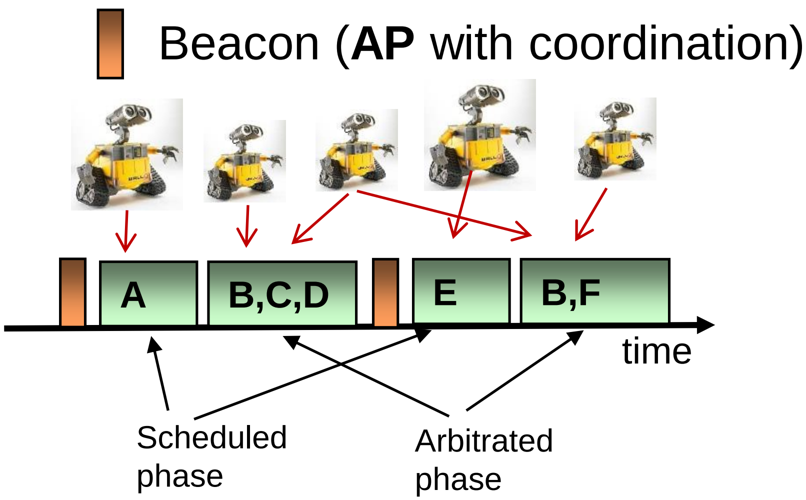
- **Arbitrated (uncontrolled) access**



CSMA/CA
(p-persistent)

- Immediate transmissions
  - If collision, back-off and retry
- Transmissions if needed, only
- Essentially configuration-free
- No real-time guarantees

- **Hybrid control**



- Requires AP
  - With coordination function for the scheduled phase (master-slave)
- Compromise btw jitter and latency

Other hybrid approaches, e.g. TDMA+CSMA
(relaxes control over tx instants)

Wireless communication technologies

(typical in Mobile-CPS)

Wireless technologies used in M-CPS

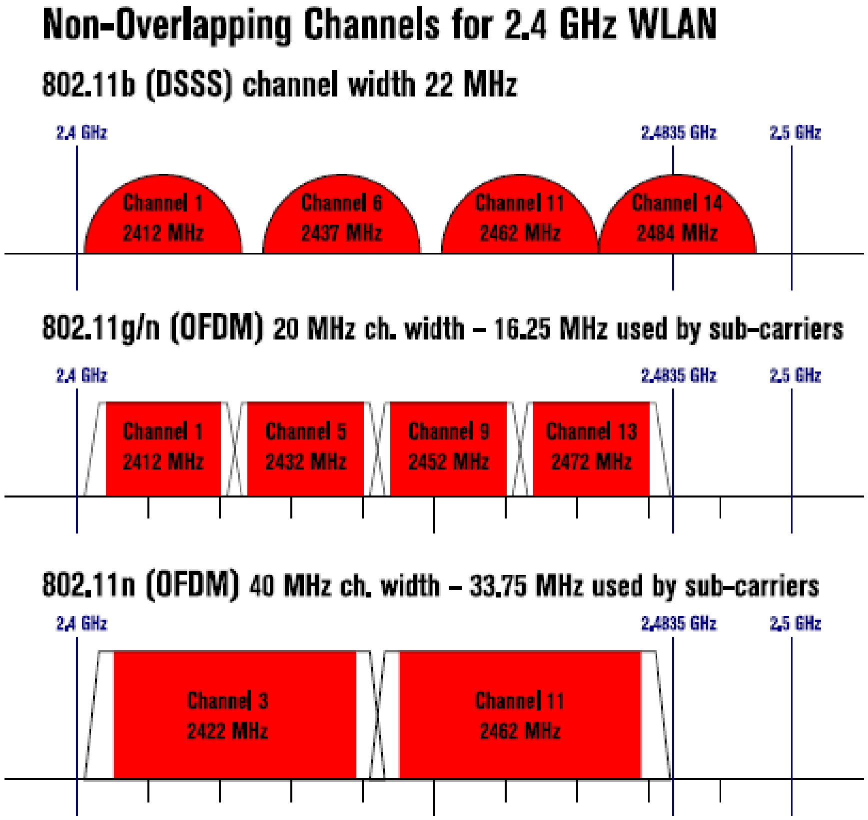
- IEEE 802.11 (WLAN)
  - **IEEE 802.11a/b/g/n - WiFi**
  - IEEE 802.11e – WiFi w/ enhanced QoS
  - IEEE 802.11p – WAVE
- IEEE 802.15.x (WPAN)
  - IEEE 802.15.1 (Bluetooth)
  - **IEEE 802.15.4** (ZigBee / WirelessHART /ISA 100)
  - IEEE 802.15.3 – UWB

Most popular, since it is coming together with every computer

2nd most popular, due to popularity in sensor networks

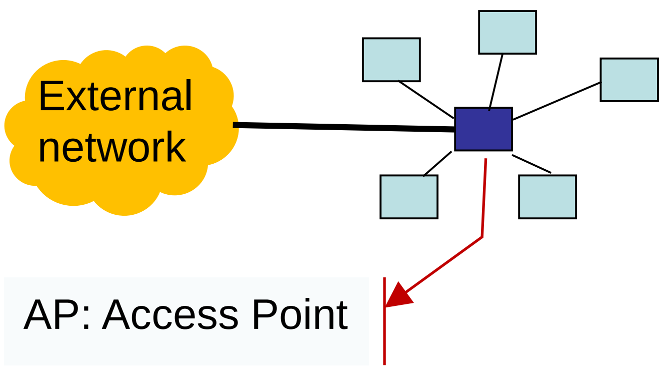
WiFi (IEEE 802.11)
standards.ieee.org/getieee802/802.11.html

- Created in the 90s as a **general purpose WLAN**
  - Very popular technology within **teams of robots**
- **4 modes** over **2 bands:**
  - IEEE 802.11b/g/n (ISM-2.4GHz) few non-overlapping channels
  - IEEE 802.11a (5GHz) several non-overlapping channels

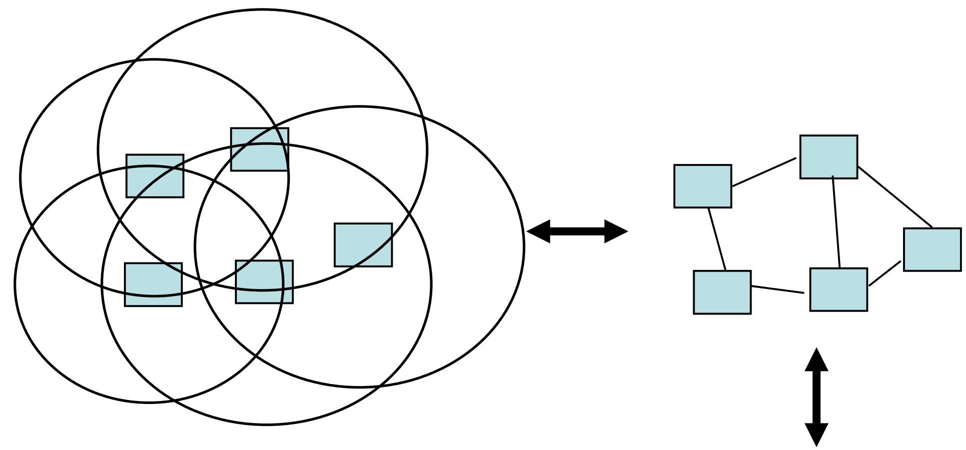


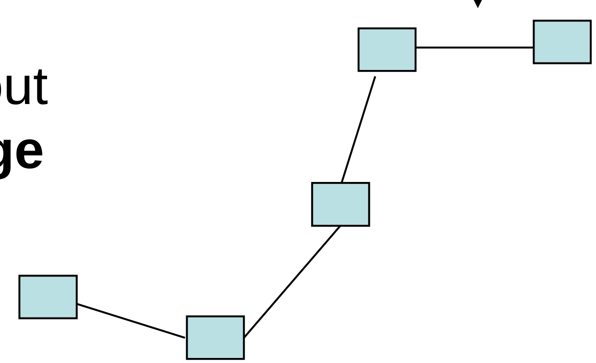
WiFi (IEEE 802.11)

- Infra-structured (star) or ad-hoc (mesh) architectures



Easy to know
who is **in** and **out**





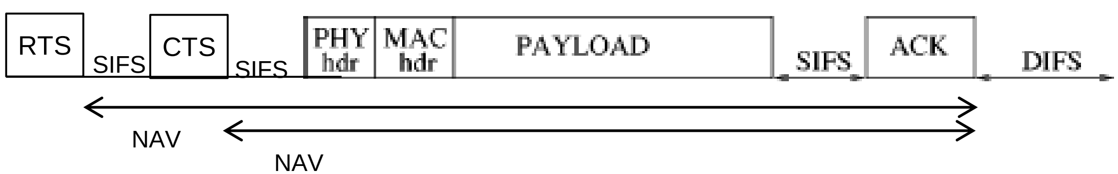
Flexible layout
and **coverage**

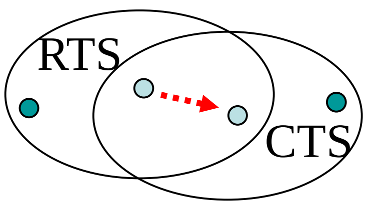
WiFi

- Many mechanisms to **reduce collisions** and **hidden-terminals**
  - Random interval to start of transmission (CSMA/CA)
  - RTS / CTS (request to send – clear to send)
  - Possibility to add transmission control (*overlay protocol*)
- Basic types of **transactions**

Unicast/Broadcast unacknowledged frame

Unicast acknowledged frame

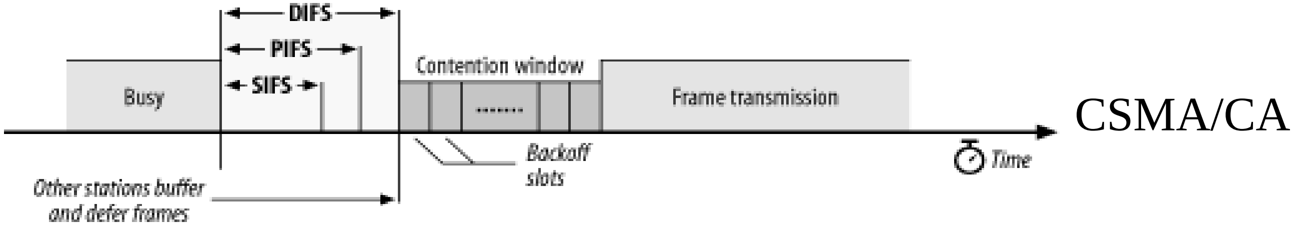




RTS/CTS for reduction of hidden terminals

WiFi

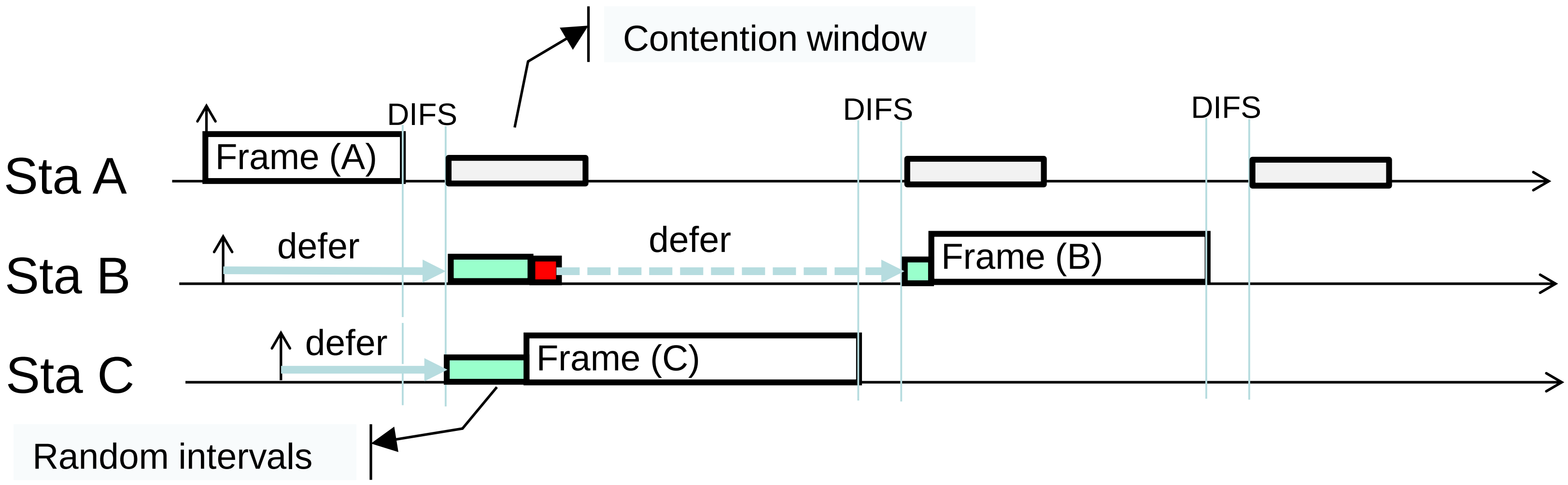
- Original MAC used different **Inter-Frame Spaces** to separate:
  - Protocol packets (ACK, RTS, CTS, ...) – **SIFS**
  - Contention-free access with master(AP)-slave (PCF) – **PIFS**
  - Contention access with CSMA/CA (DCF) – **DIFS**



- **QoS support** (802.11e)
  - Separate Video and VoIP, from e-mail and general Internet access
  - EDCA (DCF with 4 (typ) different Access Classes - AC), uses stochastically shorter inter-frame space – **AIFS** and **TXOP**
  - HCCA (AP schedules traffic)

WiFi

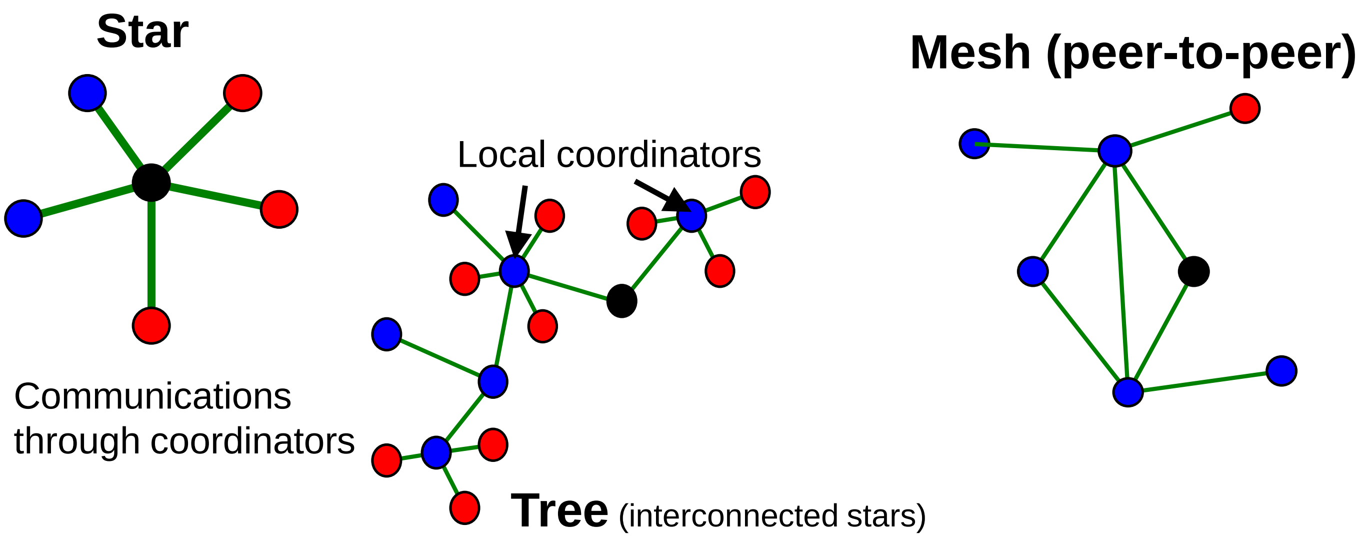
- Example of **CSMA/CA** operation



IEEE 802.15.4
standards.ieee.org/getieee802/802.15.html

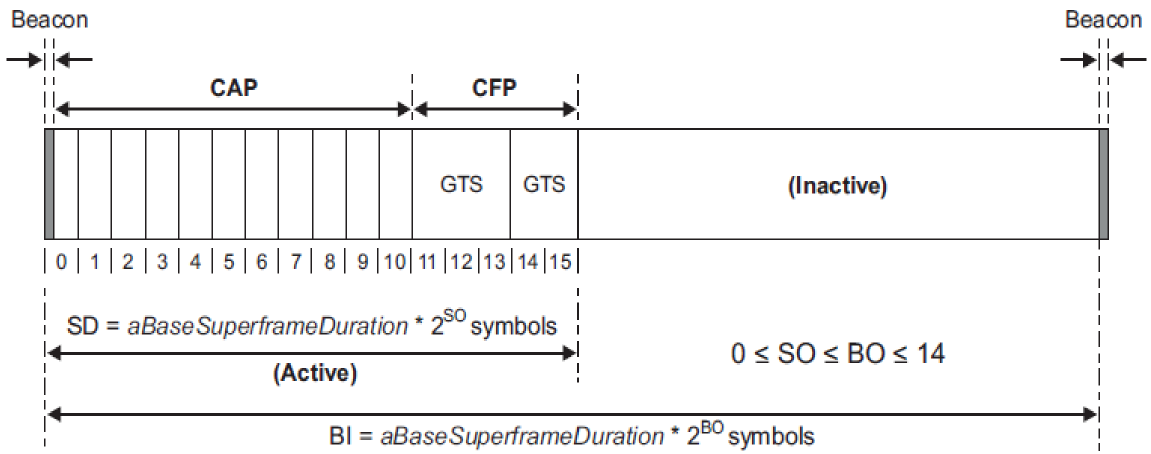
- Designed for **sensor networks**
  - Targets very low consumption
- **3 bands:** ISM-2.4GHz, 868/915 MHz
  - Several modulation schemes (DSSS is most common)
- Data rate of **250 Kbit/s** with range up to **300m**
- **Topologies:**

● **PAN Coordinator**
● **FFD** (full FD)
● **RFD** (reduced FD)
Up to 64k devices



IEEE 802.15.4

- **Multi-mode MAC**
  - Beaconless, just slotted CSMA/CA
  - Beacon-enabled but with CSMA/CA access, only
  - Beacon-enabled and with dedicated (guaranteed) slots -- GTS
    - Assigned by the coordinator on demand



$$Duty\ Cycle = \frac{SD}{BI}$$

Low duty cycle:
- Low energy
- Low reactivity

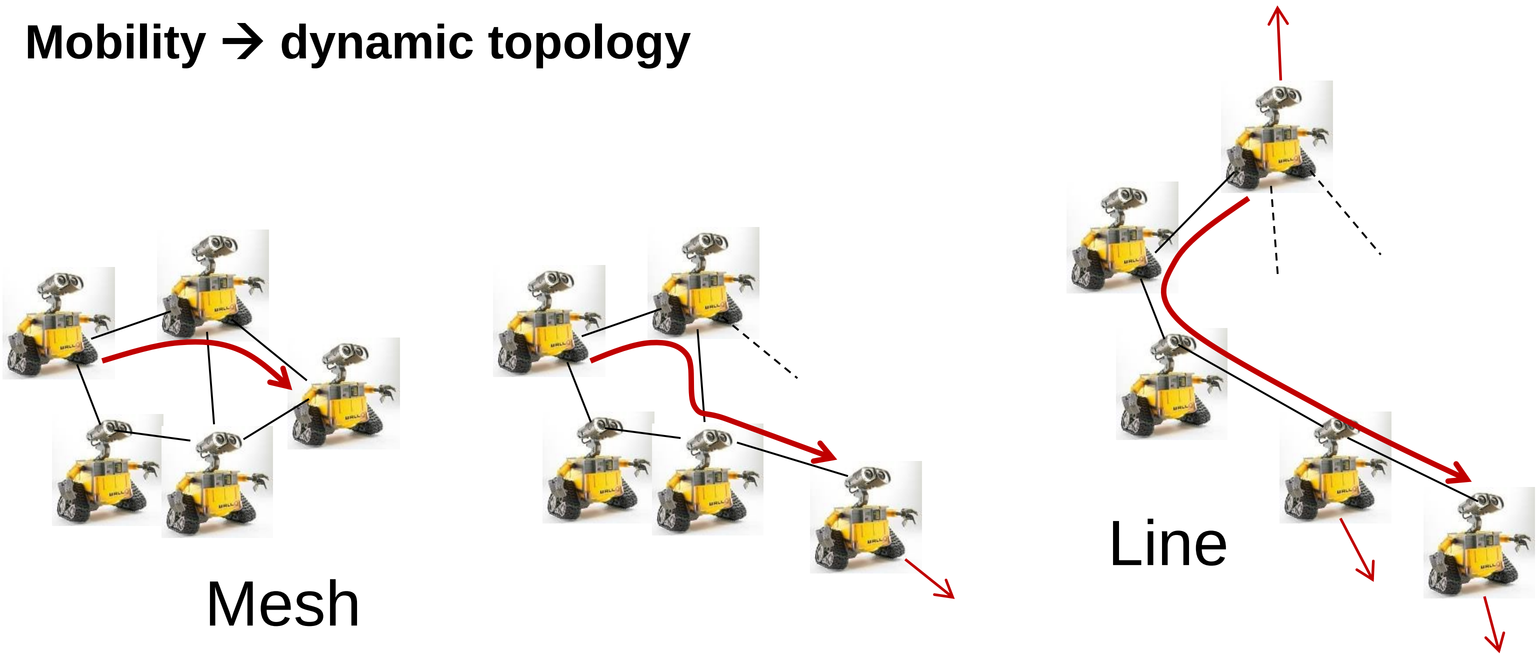
Routing for multi-hop topologies

(in Mobile-CPS)

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Routing within M-CPS

- **Ad-hoc routing in meshes**
  - Relaying messages between nodes within the team of mobile units.
  - All nodes have equal roles → **Flat routing**
  - **Mobility → dynamic topology**



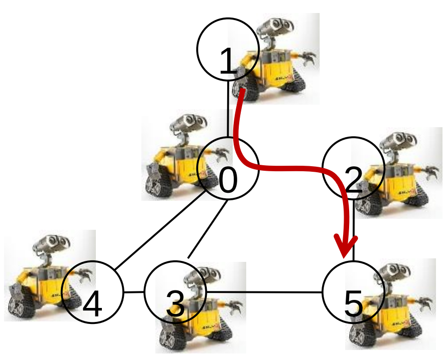
Routing within M-CPS

- **Ad-hoc routing in trees**
  - Mobile units are attached to one AP (cluster head / coordinator)
  - Messages are relayed by APs → **Hierarchical routing**
  - **Mobility → handover between Aps**
    - Tree must be maintained

Access point

Tracking the topology

- **When building the routes?**
  - On-demand → **Reactive routing**
    - **Initial latency** but permanent routing tables are not needed
      - **AODV – Ad-hoc On-demand Distance Vector**
  - Before hand → **Proactive routing**
    - Needs keeping **track of topology** → **routes** are available **on the fly**
      - Any routing method can used, e.g. **Dijkstra's shortest path** algorithm
      - Small teams → global structures viable
      - **Global connectivity (adjacency) matrix**



	0	1	2	3	4	5
0	0	1	1	1	1	0
1	1	0	0	0	0	0
2	1	0	0	0	0	1
3	1	0	0	0	1	1
4	1	0	0	1	0	0
5	0	0	1	1	0	0

Vision of node 0

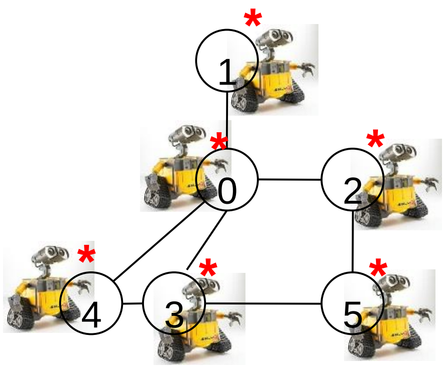
Neighbor

Not neighbor

Who receives from node 0

Propagating information

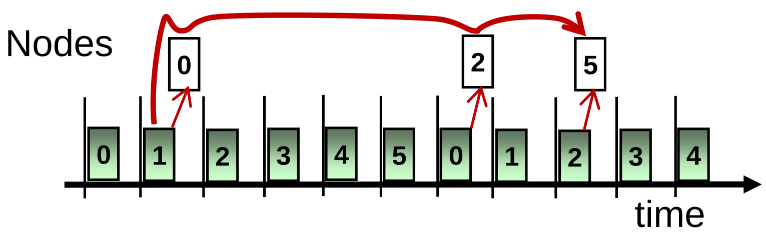
- **Flooding**
  - **All nodes broadcast** the messages they receive until a maximum number of hops or the destination is reached
  - Simple, old and most common technique in M-CPS. Presents:
    - **low requirements** for routing information (+)
      - Needs keeping track of messages already forwarded
    - high traffic levels with many duplicates (-)
      - **Traffic intensity** can be **reduced** transmitting in cycles, e.g. **TDMA**
    - not resource-aware

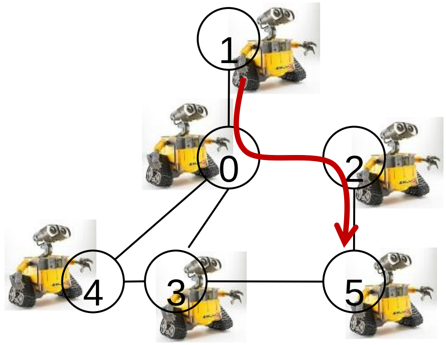


U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Propagating information

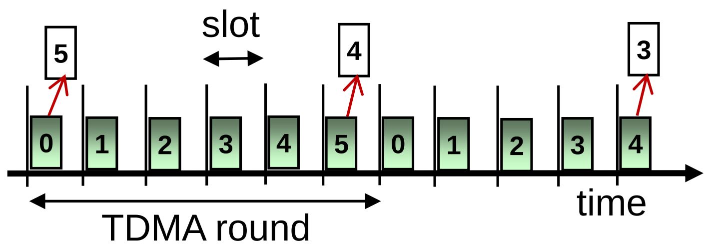
- **Flooding for routing**
  - Used to **track topology** or to **find a route** (AODV)
  - It is an **alternative to routing** to **spread *state* information**
    - Spreads any information to be shared by all

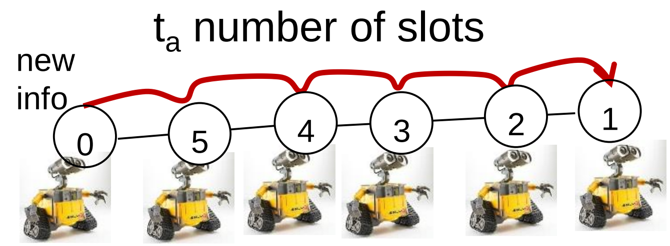




Propagating information

- **Flooding with TDMA**
  - A worst-case topology can be defined
    - Takes the longest time to spread the information





$$t_a = \min(S(n), \sigma(n,d))$$

$S(n) = n^2-n-1$ (worst-case number of **slots** with n nodes)

$\sigma(n,d) \leq 2(n-1)d$ (topology with diameter d)

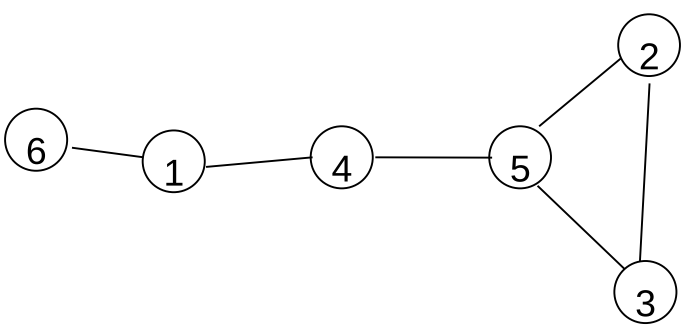
$$t_r = n-1$$

(worst-case number of **rounds** with n nodes)

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Connectivity matrix

- **Consolidating received and local matrices**
  - Using **TDMA** allows detecting **omissions**
    - Upon an **omission**, the **corresponding bit** in the local vision in the receiver matrix is **reset**
    - A column with all 0s means that **node is disconnected** from the team
  - Update the local matrix with info in received matrix for all **closer nodes**
    - Uses a **vector of shortest distances**
  - Reset distance when it increases through the same neighbor



$\delta^3(t)$

node	neigh	dist
1	5	3
2	2	1
3	3	0
4	5	2
5	5	1
6	5	4

$M^3(t)$

	1	2	3	4	5	6
1	0	0	0	1	0	1
2	0	0	1	0	1	0
3	0	1	0	0	1	0
4	1	0	0	0	1	0
5	0	1	1	1	0	0
6	1	0	0	0	0	0

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

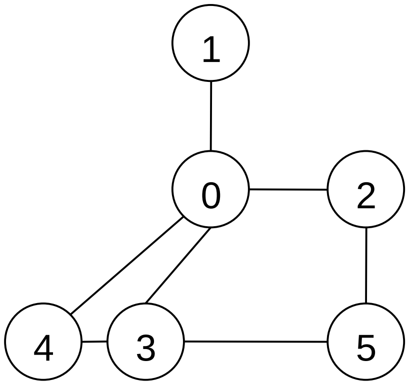
Extended connectivity matrix

- **How robust is the matrix?**
  - Binary information can vary unpredictably
    - Links in boundary conditions
  - Can be **extended** with the **strength of the links**
    - Use **RSSI** of the received message instead

	0	1	2	3	4	5
0	?	50	50	50	20	?
1	48	?	?	?	?	?
2	52	?	?	?	?	60
3	50	?	?	?	90	10
4	25	?	?	88	?	?
5	?	?	59	14	?	?

RSSI'

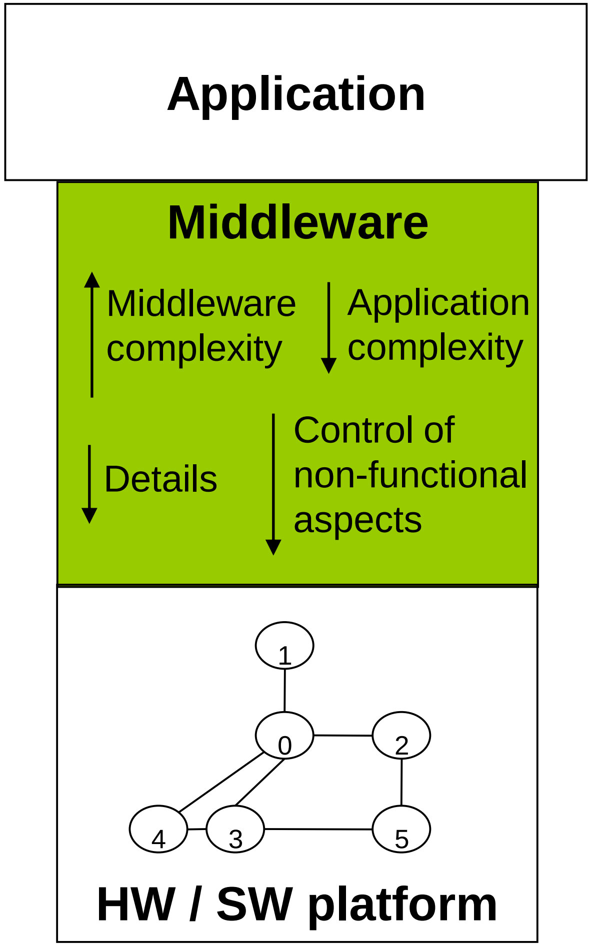
Out of Range



Middleware layers for M-CPS

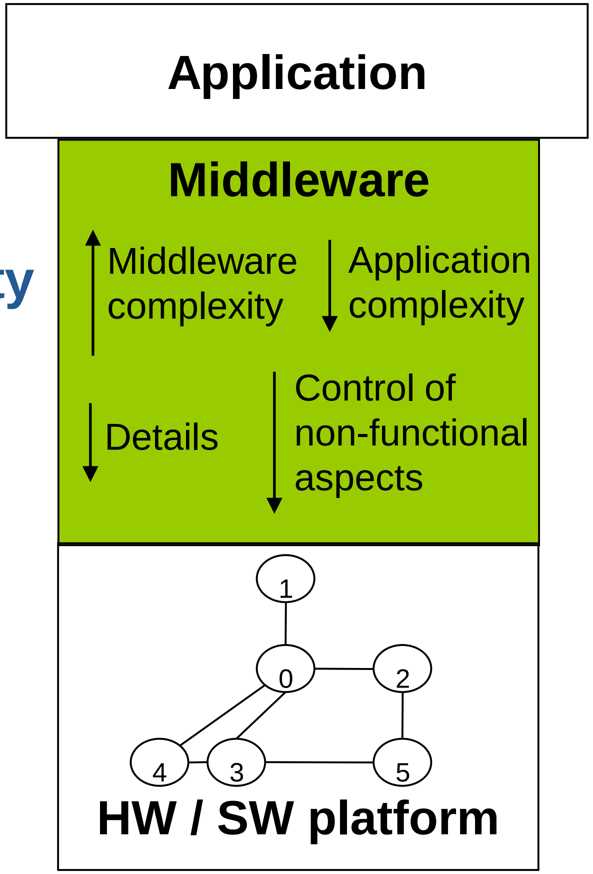
Middleware

- **The middleware is a SW layer that**
  - Hides unnecessary platform details
  - Simplifies development, adds new services
- **But it implies trade-offs**
  - The HW/ SW platform has a profound impact on non-functional properties
    - timing, performance, dependability...
  - The simpler it is to develop applications the more complex the middleware is
    - And vice-versa



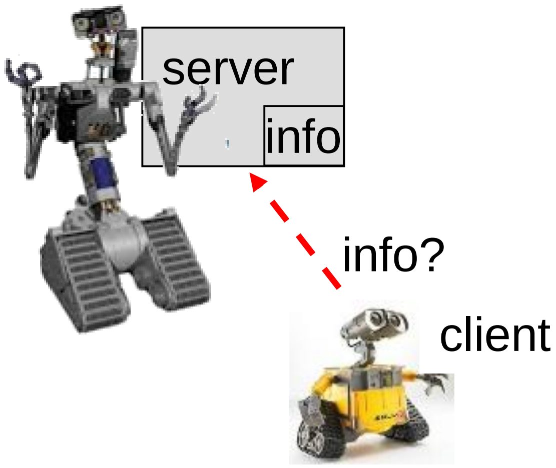
Robotics middleware requirements

- **Simplify application development**
  - High level abstractions and simplified interfaces
    - Hiding heterogeneity and low-level communication
- **Support communications and interoperability**
  - Integration of modules from different sources
    - Automatic discovery and configuration
- **Provide efficient resource utilization**
  - Processors, networks, memory…
- **Offer typically required services**
  - Navigation, filters, control…
- **Support integration of devices with low resources**
  - Connection to simple embedded devices



Cooperation models

- ## Client-Server
  - **Servers hold** information. **Clients request** it
    - Transactions **triggered by the receivers** (clients)
    - Typically based on **unicast** transmission (one to one comm.)
  - Transactions can be **synchronous** (client blocks until server answers) or **asynchronous** (client follows execution after issuing the request)
  - With synchronous transactions the **communication** time is **inside** the **computing loop**…
  - Requires **naming service**
  - Adequate for sporadic use of the data
  - Technologies: **RPC, RMI, CORBA, ROS**

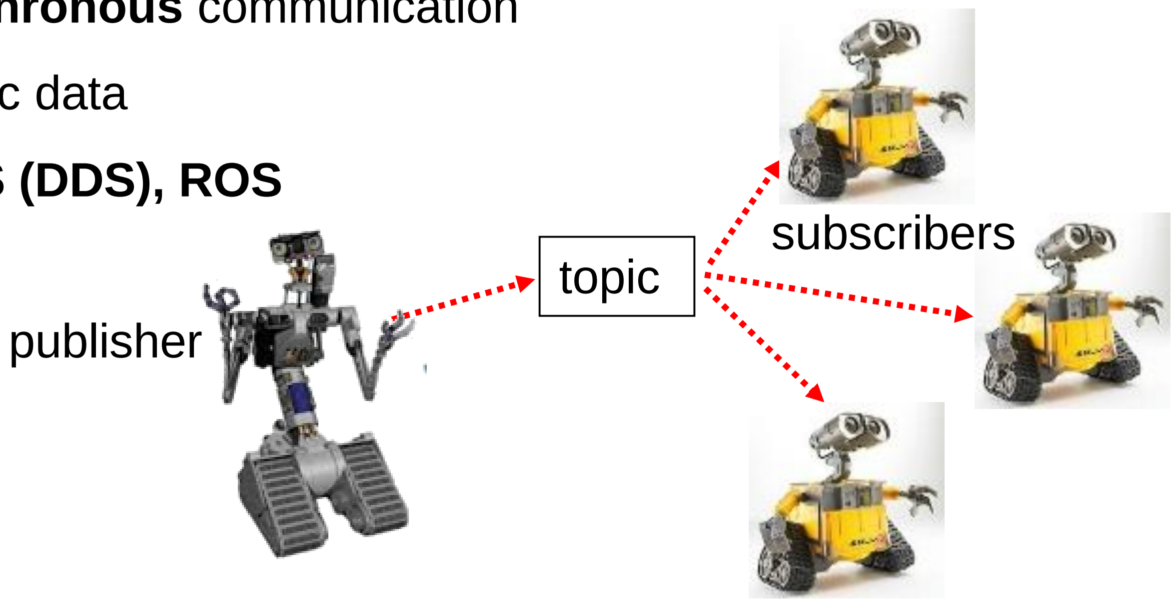


Cooperation models

- ## Producer-Consumer
  - **Producers disseminate** information. **Consumers use** it
    - Transactions **triggered by the senders** (producers).
    - Based on **broadcast** transmission (each message is received by all)
      - **Anonymous asynchronous** communication,
      - Security constraints?
  - **Communications** time is **outside** the **computing loop**
  - Adequate to regular state dissemination
  - Technologies: **CANopen**

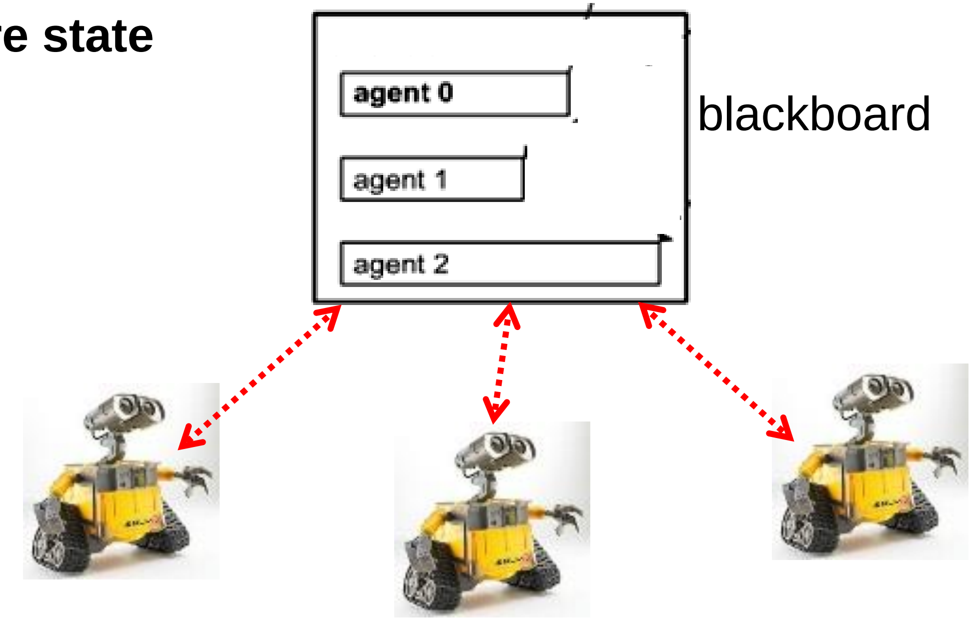
Cooperation models

- **Publisher-Subscriber**
  - Uses concept of **group communication** around data entities (topics)
  - Nodes must adhere to groups either as **publisher** (produces information) or as **subscriber** (consumes information)
  - Transactions are **triggered by the publisher** of a group and disseminated among the **respective subscribers**, only (multicast)
  - **Anonymous asynchronous** communication
  - Regular and sporadic data
  - Technologies: **RTPS (DDS), ROS**



Cooperation models

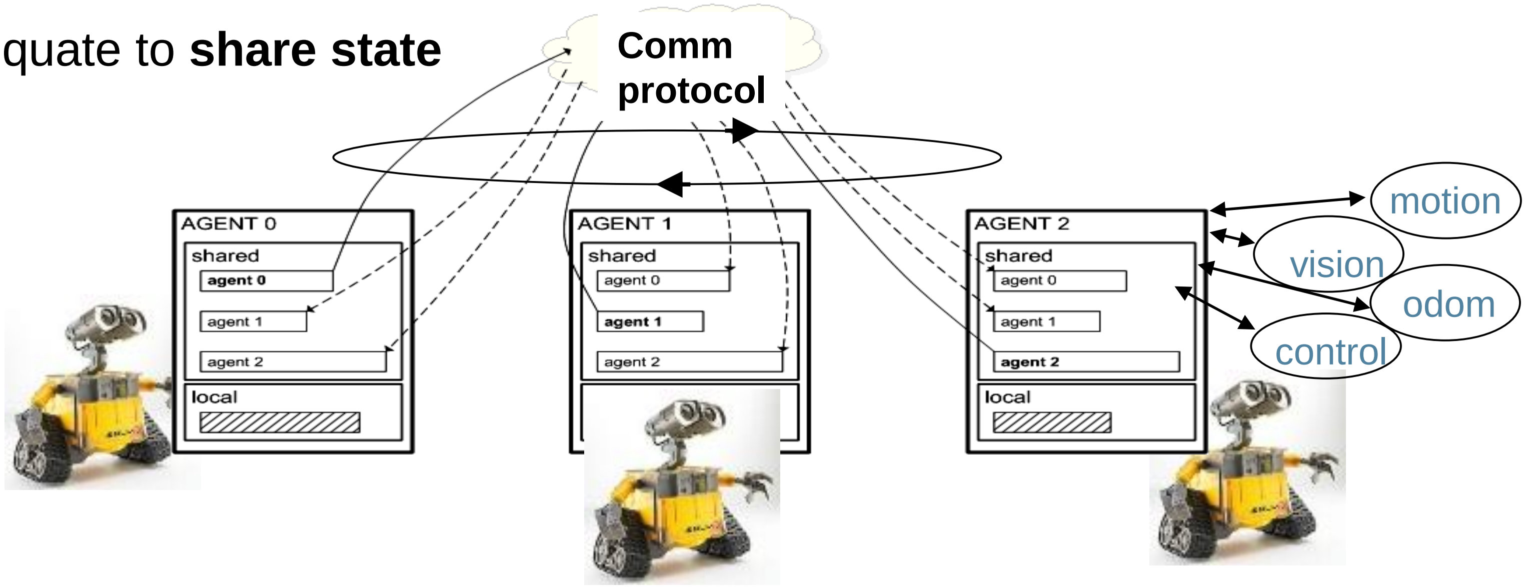
- **Shared memory – Blackboard**
  - Communicating processes **write and read from a common area**
    - This common area (**Blackboard**) may reside in a different computer
  - **Communication** time **inside** the **computing loop**
  - **Anonymous asynchronous** communication
  - Adequate to **share state**



FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

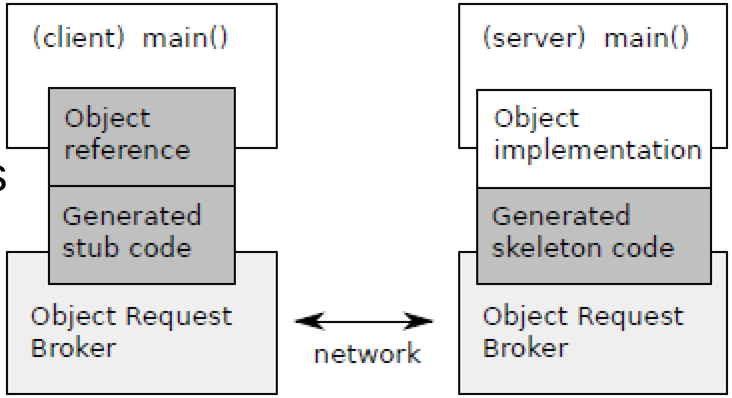
Cooperation models

- **Shared memory – Real-Time Database (RTDB)**
  - Communicating processes **write and read from a common area**
    - Common area is **replicated** in all agents providing **local data access**
      - **Real-time Database (RTDB)**
  - **Communications** time **outside** the **computing loop**
  - **Anonymous asynchronous** communication
  - Adequate to **share state**



Middleware technologies

- **CORBA – Common Object Request Broker Architecture**
  - www.corba.org
  - Open specification proposed by OMG
  - **Purpose:** Clients use remote objects as if they were local
- **Main features**
  - Interoperability between languages and platforms
    - Windows, Linux, Unix, MacOS, QNX, VxWorks, …
    - Ethernet, CAN, Internet, ...
    - Ada, C, C++, Java, Python, …
  - Multiple vendors & open-source products

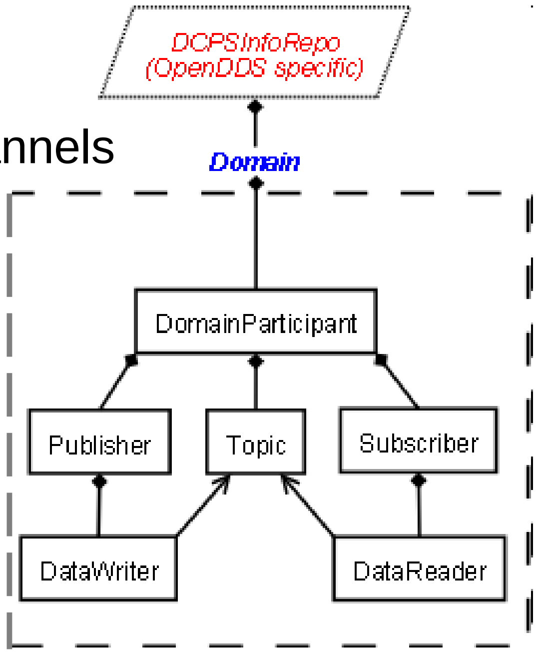


Middleware technologies

- **CORBA implementations / profiles used in robotics**
  - **RT-CORBA:** Support for applications with end-to-end timing constraints
  - **CORBA/e:** For embedded devices (Minimum CORBA and Micro CORBA)
  - **RTC – Robotics Technology Component**
    - Component model, with structural and behavior features typical in robotics
  - **TAO:** Open source, QoS support for real-time and embedded systems
  - **MIRO:** (TAO) sensor/actuator services as network transparent CORBA objects
  - **RT-Middleware:**
    - Component model with real-time functional elements – RT-Components
    - Applications designed in UML aggregating RT-Components

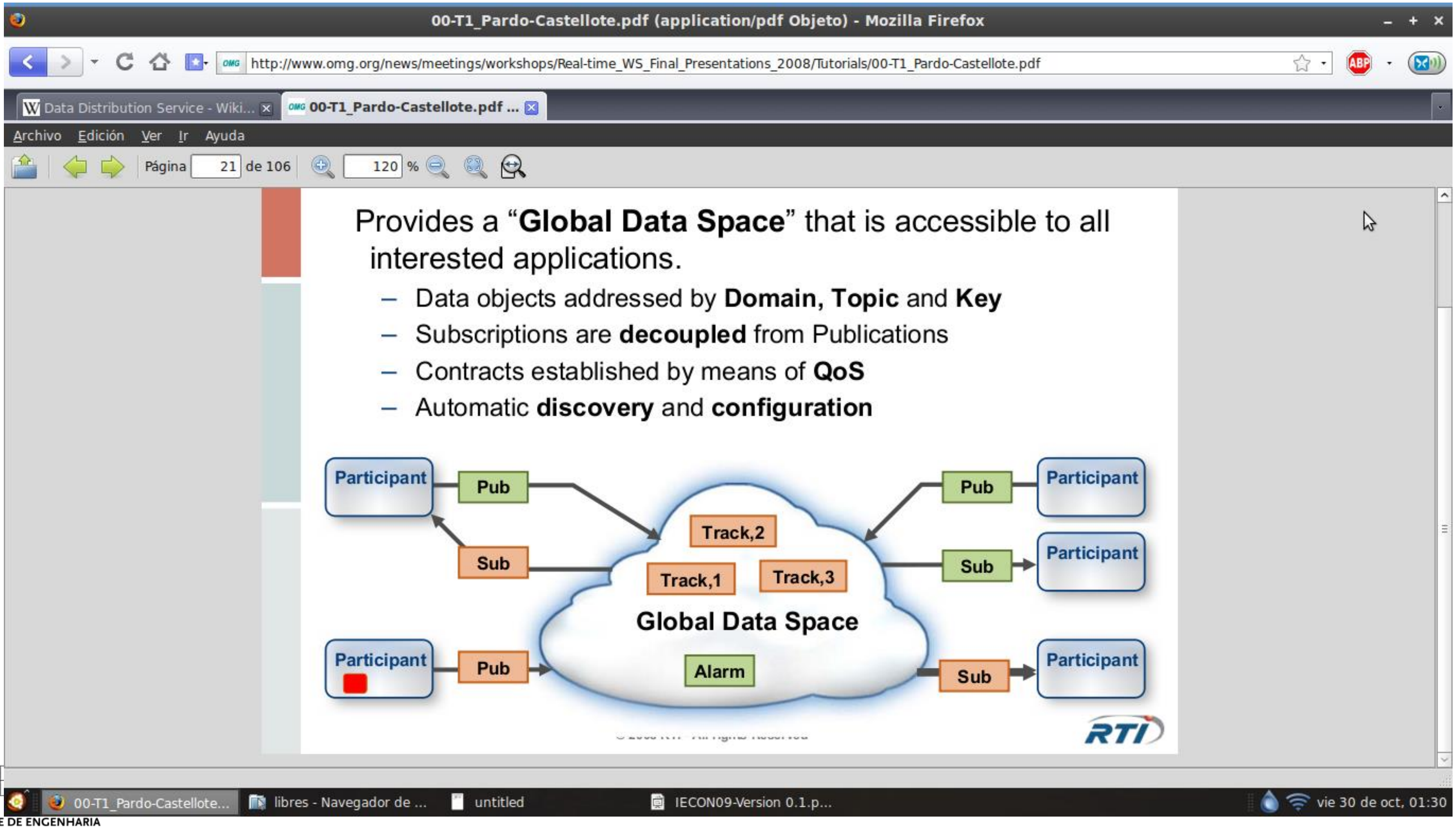
Middleware technologies

- **DDS – Data Distribution Service**
  - portals.omg.org/dds
  - Open specification proposed by OMG
  - **Purpose:** provide a Publisher-Subscriber data-centric model for distirbuted real-time applications
- **Main features**
  - Anonymous communication with asynchronous channels
    - Platform independence
    - Handles addressing, delivery, control flow
  - Global distributed database of **Topics**
    - Unique names, abstract data type, QoS parameters
    - **Signals**, **Streams** and **States**



Middleware technologies

- ## DDS – Global Data Space (GDS)



Middleware technologies

- **DDS implementations**
  - **Connext DDS**
    - most complete implementation (commercial, by RTI)
  - **RTPS**
    - Real-Time Publisher-Subscriber protocol
    - Provides interoperability to DDS
      - RT communication over IP (UDP/IP), Fault-tolerance, Extensibility, Plug&play, Configurability, Modularity, Scalability, Type-safety
  - **ORTE**
    - Open source RTPS implementation

Middleware technologies

- **SOAP – Simple Object Access Protocol**
  - www.w3.org/TR/soap
  - Open specification proposed by W3C
  - **Purpose:** exchange of structured and typed information based on XML
    - XML-RPC
- **Main features**
  - Stateless, asynchronous messaging system
  - Agnostic to application semantics
  - Modular packaging model and encoding mechanism
  - Consists of
    - **Envelope:** definition of what, who and whether optional/mandatory
    - **Encoding rules:** serialization mechanism
    - **RPC representation:** convention to represent RPCs and their responses

Middleware technologies

- SOAP implementations / profiles used in robotics
  - **ROS: Robot Operating System**
    - Hardware abstraction, low-level device control, commonly used functions, message-passing, package management
    - **Client-Server** and **Publisher-Subscriber** (ROS Topics) models
    - Framework with user contributed packages: SLAM, planning, perception, …
    - Free and Open-source

    Goals:

    - **Peer-to-peer** model supported on a *name service*
    - Focused **tools-based** approach, e.g. get/set configuration parameters, visualize the peer-to-peer connection topology, measure bandwidth utilization
    - **Language neutral** with specification at the messaging layer and peer-to-peer connection, negotiation and configuration in XML-RPC

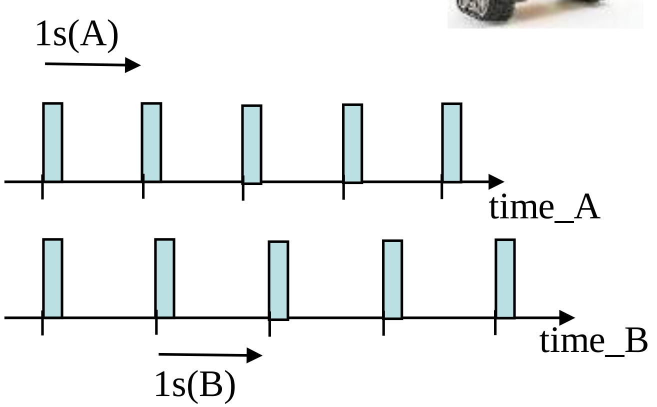
Middleware technologies

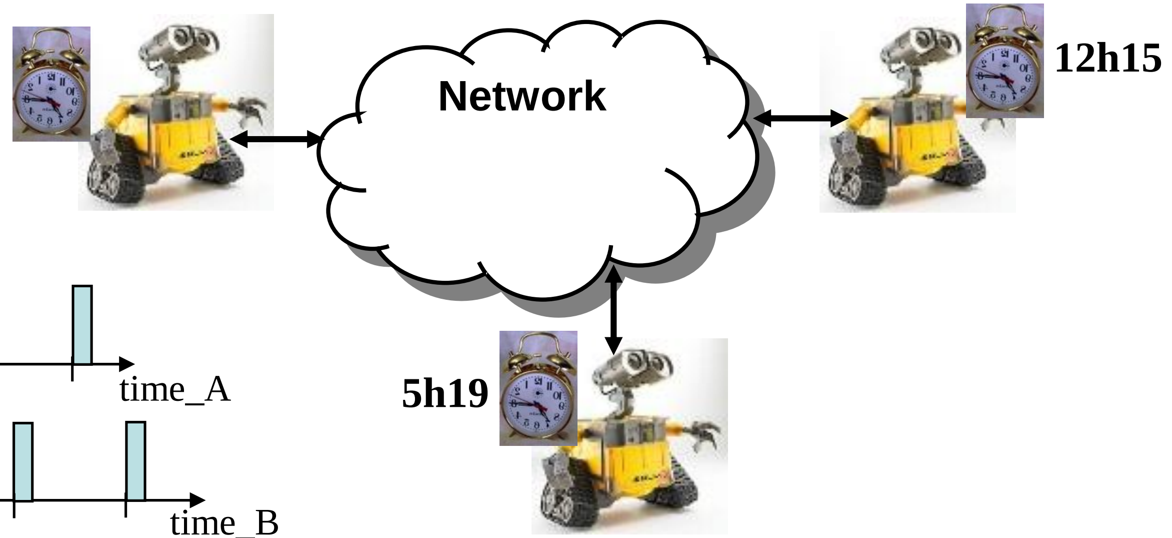
- **Different middleware models/paradigms imply**
  - Different communication **load**
    - CS, Blackboard: question response but unicast
    - PS, RTDB: one way but multicast
  - Different communication **pattern**
    - CS, PS – normally events
    - PC, PS, shared memory – normally periodic
  - Different level of **openness**
    - CS, Blackboard – directed unicasts
    - PS, RTDB – anonymous multicast
    - PC – broadcast

Time and physical clocks across the team

Time across a network

- In a **distributed system** each node has its **own clock**
  - Without specific support, there is **no explicit coherent notion of time** across a distributed system
  - Worse, due to **drift**, clocks tend to **permanently diverge**





Time across a network

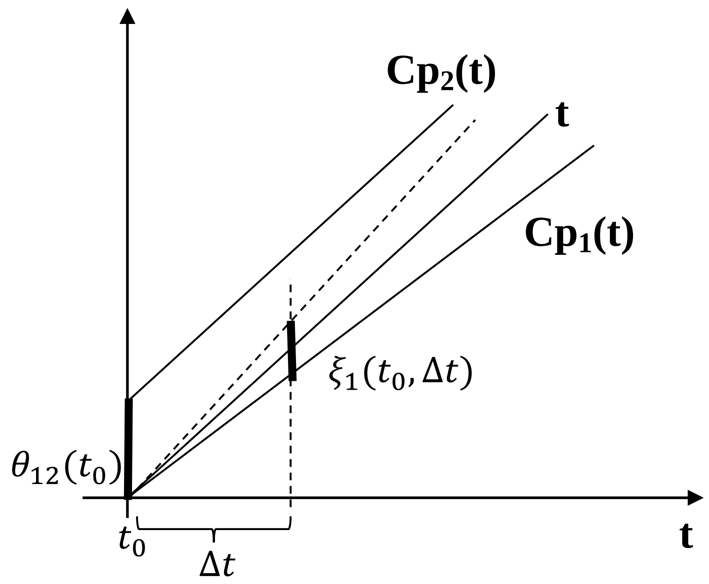
- **Why** developing a **coherent notion of time?**
  - Carry out **actions** at **desired time** instants
    - e.g. synchronous data acquisition, synchronous actuation, control formation
  - **Time-stamp** data and events
    - e.g. establish causal relationships that led to a system failure
  - Compute the **age** of data
  - **Coordinate** transmissions
    - e.g. TDMA clock-based systems

But how to synchronize the clocks across the network?

Few definitions

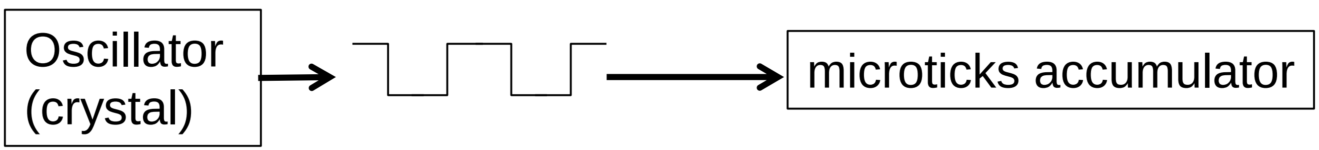
- **Offset**
  - $\theta_{ij}(t) = \left|Cp_i(t) - Cp_j(t)\right|$
- **Drift rate** and **drift**
  - $\rho_i(t) = \left|\frac{(Cp_i(t+\Delta t) - Cp_i(t))}{\Delta t} - 1\right|$
  - $\xi_i(t, \Delta t) = 2 * \rho_i(t) * \Delta t$
- **Accuracy** α
  - $\max_{it}|Cp_i(t) - t| \leq \alpha$
- **Precision** δ
  - $\max_{ijt}\left|Cp_i(t) - Cp_j(t)\right| \leq \delta \Leftrightarrow \max_{ijt}\left(\theta_{ij}(t)\right) \leq \delta$

Cpi(t) is the **clock** of **node i** at **instant t**



Digital clocks

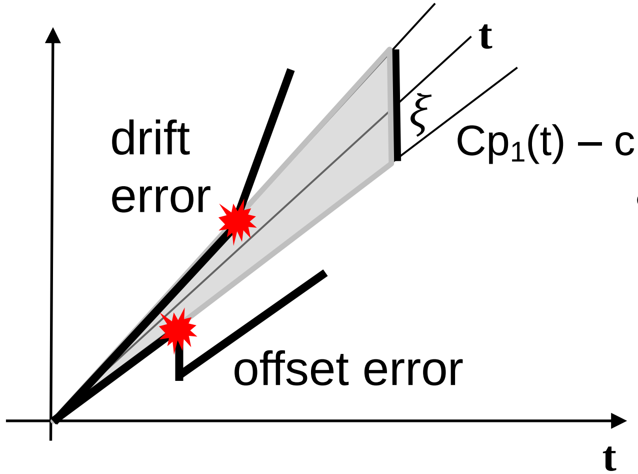
- A digital **clock** is a **counter** incremented every **tick** (fixed interval)
  - A **tick** is implemented counting a **fixed number** (n) of **microticks** that represent oscillator pulses



- Main **clock parameter** is the **drift rate**
  - Real clocks have drift rates between **10^{-2}** and **10^{-8}** depending on the quality of their oscillators

Fault model of digital clocks

- Clocks can suffer **offset errors** and **drift errors**



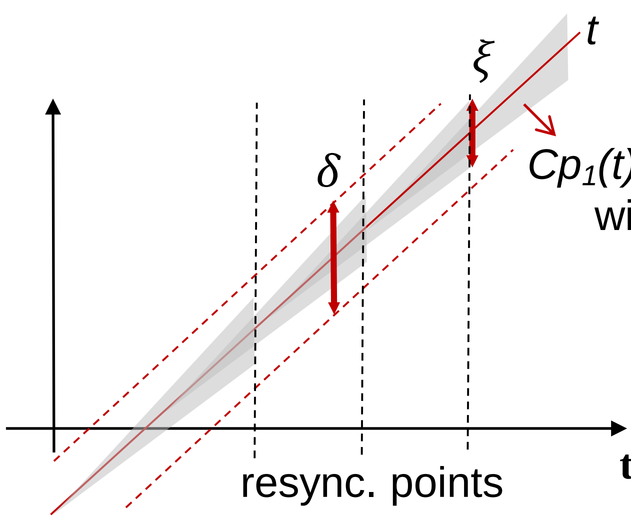
  - Offset errors are stochastic errors in tick counting
  - Drift errors can be **sistematic** (due to inherent drift rate) and **stochastic**
  - **Sistematic drift >>** stochastic drift
    - allows algorithmic correction → **clock synchronization**

Clock synchronization

- Clocks can be synchronized:
  - **Externally** – an external source sends a time update regularly (e.g. GPS)
    - Quality metric: **accuracy**
  - **Internally** – nodes exchange messages to come up with a global clock
    - **Master-Slave** – The time master spreads its own clock to all other nodes
    - **Distributed** – All nodes perform a similar role and agree on a common clock, typically an average
    - Quality metric: **precision**
  - Both methods are complementary
    - **Internal** synchronization provides **high availability** and **good short-term stability**
    - **External** synchronization provides **long-term stability** but has lower availability
- Standards:
  - NTP, SNTP, IEEE 1588 (PTP)

Synchronizing clocks

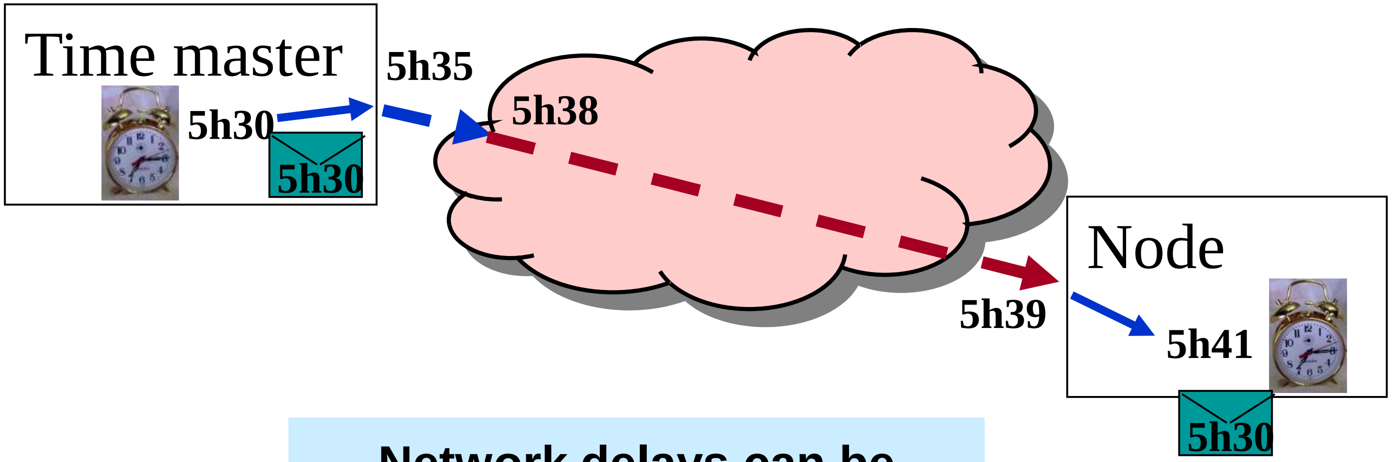
- Requires **regularly**
  - **Exchanging** clock values
  - Measuring **differences**
  - Computing and applying a **correction term**
    - In the form of **increment/decrement** of the **microticks** counter (n)
    - (not so common) Directly in the clock tick value



$Cp_1(t)$ – clock synchronized with t with drift ξ and precision δ

Network delay and precision

- Impact of **network delay jitter** on the **achievable precision**



Network delays can be rather variable and large

Network delay and precision

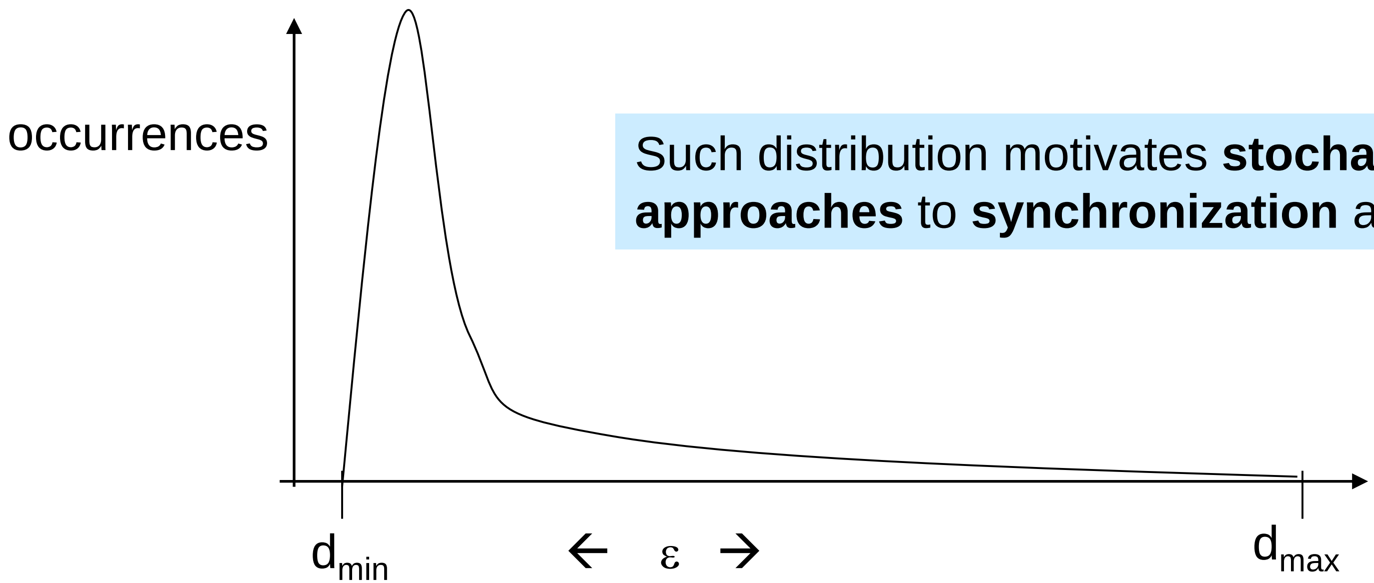
- **Network delay jitter** (ε) limits the achievable **precision** (δ)
  - **Clock synchronization** algorithms allow correcting **systematic errors**
    - not the impact of ε **(ε = d_{max} – d_{min}** → jitter in the network delay d)
  - Typical precision with SW methods in small networks is worse than 10μs
    - In LANs it is common to achieve 1-5ms precision
  - With special HW support, it is possible to reach 1μs or better
  - **Ludelius and Lynch** showed that the **precision** δ achievable in a network with **N nodes** (with drift-free oscillators) and **ε network delay jitter** is bounded to

$$\delta \leq \varepsilon \left(1 - \frac{1}{N}\right)$$

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Network delay and precision

- In **shared networks** with collisions and back-off/retry mechanisms
  - Typical distribution of the network-induced delay (d)

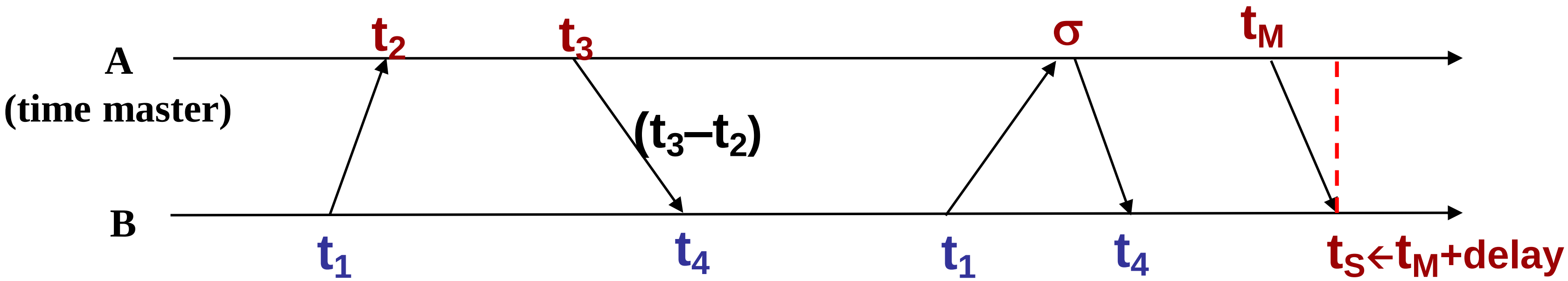


Such distribution motivates **stochastic approaches** to **synchronization** and **precision**

  - **Ludelius and Lynch**'s bound is a deterministic bound
  - **Stochastic approaches** can achieve **unbounded precision**

Measuring the network delay

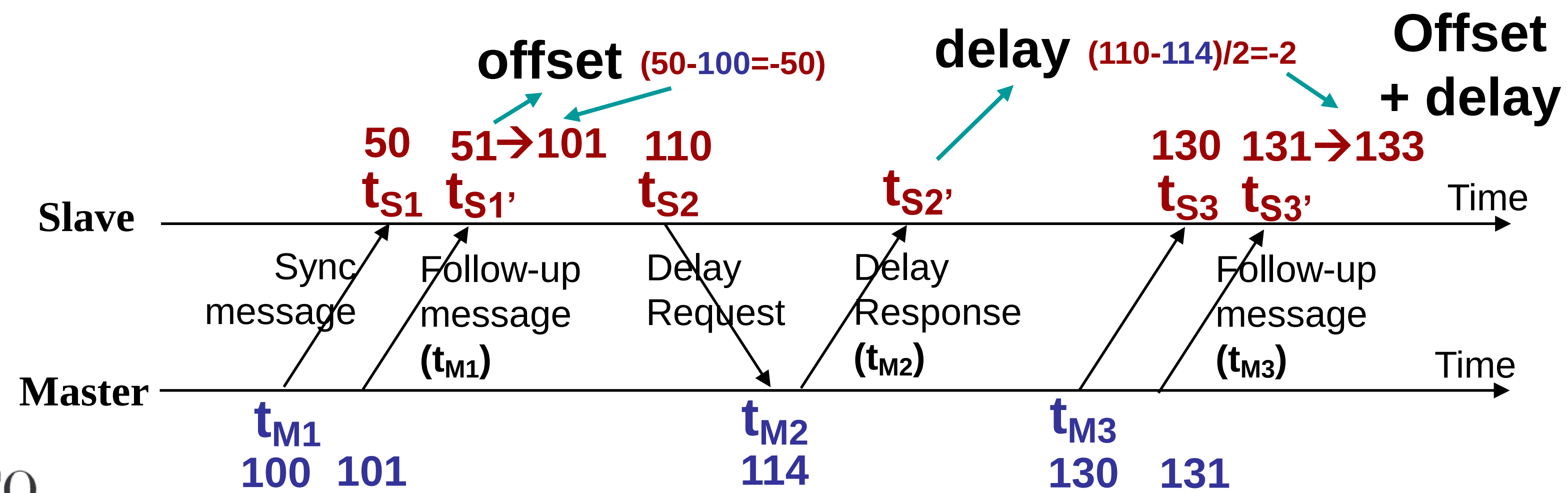
- **Round-trip delay - RTD**
  - estimated from $((t_4 - t_1) - (t_3 - t_2))/2$ on node B
  - used to correct delay at B upon reception of time marks from A



$$delay \approx \frac{RTD}{2} = \left\langle \frac{(t_4 - t_1) - (t_3 - t_2)}{2} \middle| \frac{(t_4 - t_1) - \sigma}{2} \right\rangle$$

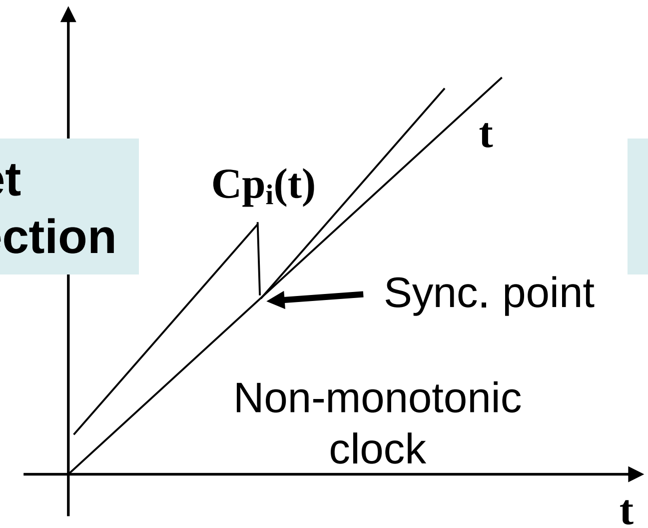
Synchronization with IEEE 1588

- **Follow up messages**
  - ***Timestamps on “end of transmission”***
  - ***Synchronization messages do not carry timestamps***
- **Slave to Master offset:** estimated from $(t_{S1}-t_{M1})$ at $t_{S1'}$
  corrected at $t_{S1'} \leftarrow t_{S1'}-(t_{S1}-t_{M1})$
- **Network induced delay:** estimated from $(t_{S2}-t_{M2})/2$ at $t_{S2'}$
  corrected at $t_{S3'} \leftarrow t_{S3'}-(t_{S3}-t_{M3})-\text{delay}$

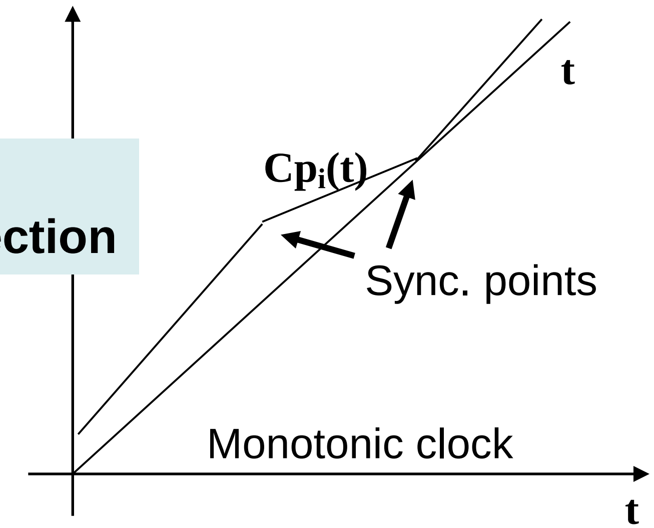


Clock correction

Offset correction



Rate correction



- **Most applications** require **monotonic clocks**

$t_1 < t_2 \Rightarrow Cp_i(t_1) < Cp_i(t_2) \quad \forall_{i,\, t_1,\, t_2}$ **chronoscopic behavior**

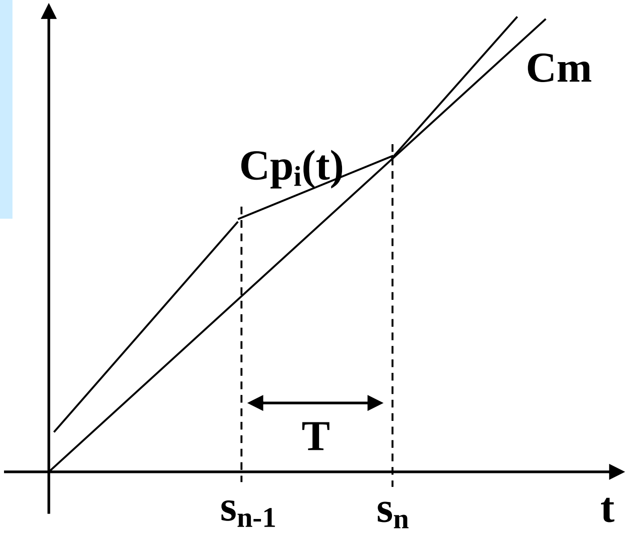
- A **chronoscopic behavior** implies **rate correction**

Correcting the local clock

- **Rate correction with feedback** (*servo-clock*)
  - **Cp_i** = **local clock** (as seen by the local applications)
  - **Cm** = **master clock** (as received in master messages)
    - $\rho_n = 1 - (\mathbf{Cp_i}(s_n) - (\mathbf{Cm}(s_n)+d)) / T$
      - ρ_n is the microticks (rate) correction term
      - T = sync interval, s_n = synchronization point n

– $\mathbf{Cp_i}(t) = \mathbf{Cp_i}(s_n) + \rho_n*(t-s_n)$ $\mathbf{s_n < t < s_{n+1}}$

$$\delta = \xi + \varepsilon$$

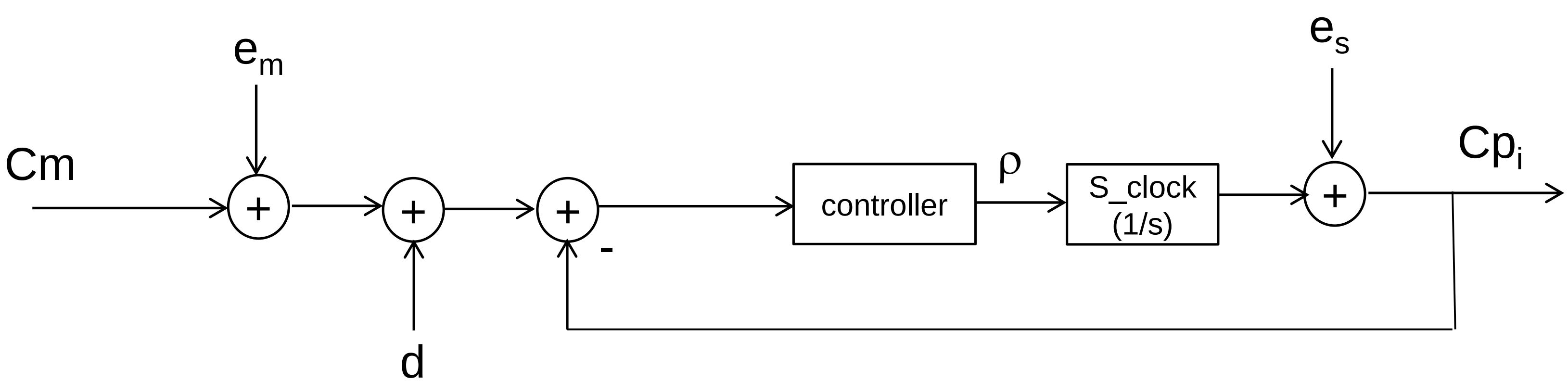


<u>It is common to consider:</u>

- $\rho_{min} < \rho_n < \rho_{max}$ to bound the clock growth
- **k*T** (k>1) instead of **T** to stabilize the local clock (less reactive)

Correcting the local clock

- **Rate correction with feedback** (*servo-clock*)
  - The clock correction as a **feedback control loop**
    - e_m → delays affecting the time stamping in the master
    - e_s → errors affecting the oscillator in the slave
    - d → network delay



Distributed clock synchronization

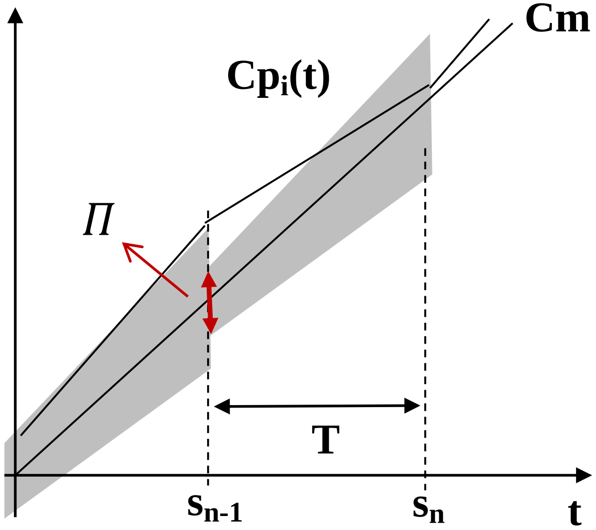
- There is **no master clock**
  - All nodes exchange their clock values among themselves
  - A **virtual reference clock Cm$_i$** is built averaging **all N clocks**

$$Cm_i(s_n) = \frac{1}{N}\sum_{k=1}^{N} Cp_k(s_n)$$ → error divided by N (convergence factor Π)

  - Under certain conditions, all nodes converge to this virtual ref

$$\delta_{dist} \geq \Pi + \xi + \varepsilon \text{ and } \Pi = \frac{\delta_{dist}}{N}$$

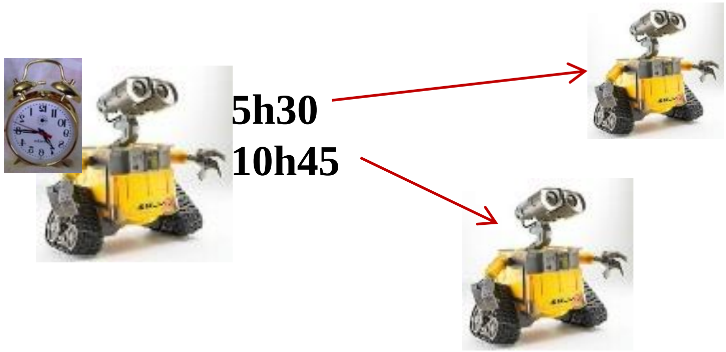
$$\delta_{dist} \geq (\xi + \varepsilon)\frac{N}{N-1}$$



  - Crashed nodes can be removed from the group and do not affect the global clock

Distributed clock synchronization

- **Fault-Tolerant Average (FTA)**
  - The **usual average** is **sensitive to very poor clocks** that diverge a lot from the others, in the presence of network errors
    - Good clocks can end up with less weight than poor clocks
    - In sparsely connected networks, normal average can lead to different nodes computing different virtual references
  - The **usual average** is also **sensitive to Byzantine** errors
    - Nodes that send inconsistent information to different destinations
    - In this case, the whole set of nodes will not converge to a virtual reference

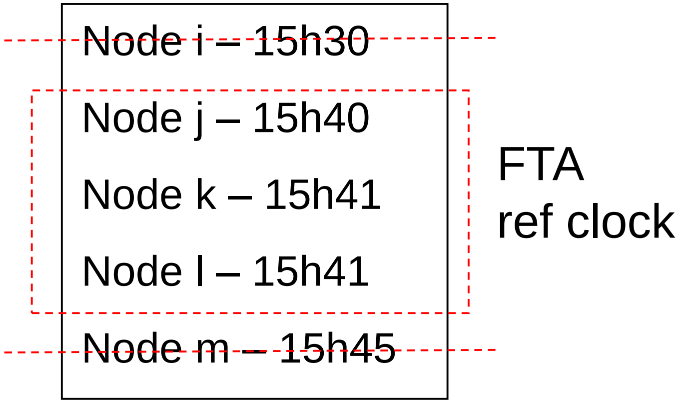


Distributed clock synchronization

- **Fault-Tolerant Average** (FTA)
  - **sort differences** between local clocks and average
  - **eliminate** the clocks with the **k highest** and **k lowest** differences to the average
  - Use the average of the remaining clocks as the **virtual reference**
  - This allows **tolerating k Byzantine** clocks

$$\Pi = k\frac{\delta_{dist}}{N-2k}$$

$$\delta_{dist} \geq (\xi + \varepsilon)\frac{N-2k}{N-3k}$$



Distributed clock synchronization

- **Interactive Consistency**
  - **All nodes:**
    - Send a **vector** with **their view** of all other clocks
    - Build a **local matrix** with **all views of all clocks**
  - Allows immediate detection of Byzantine clocks
    - Can be readily excluded
    - Remaining ones averaged to generate the virtual reference
  - **Tolerant to any Byzantine clocks**
  - Requires more communication

Clock of node 4 received by others

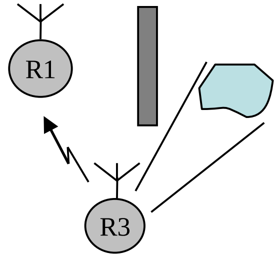
	0	1	2	3	4	5
0	21	20	21	22	19	20
1	31	30	29	30	29	30
2	41	41	40	39	**60**	39
3	50	51	50	50	51	49
4	61	60	59	60	**24**	60
5	71	70	69	70	71	70

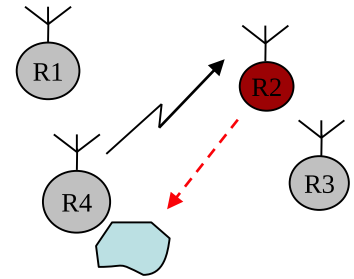
Clocks received by node 4 in one round

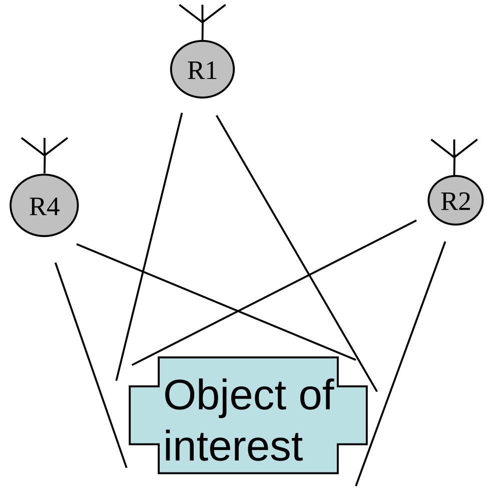
Basics on RF-based localization
Measuring distance

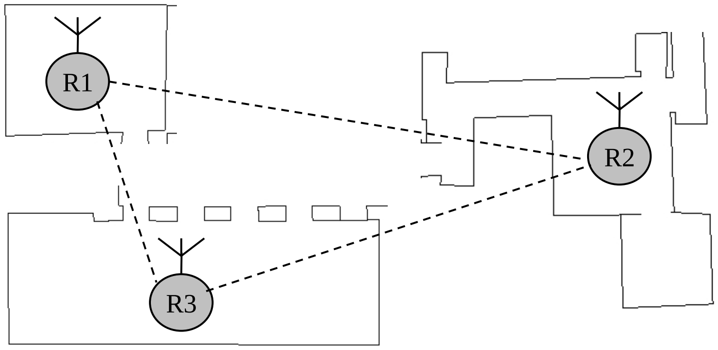
Localization

- **Essential to many collaborative behaviors**
  - Formation control
    - Set plays in team games
  - Joint object transportation
  - Joint SLAM
  - Heterogeneous robots
    - Sharing expensive actuators
      - Demining
  - Area coverage
    - Search and rescue
    - Inspection
    - Cleaning
  - Surveillance
  - …









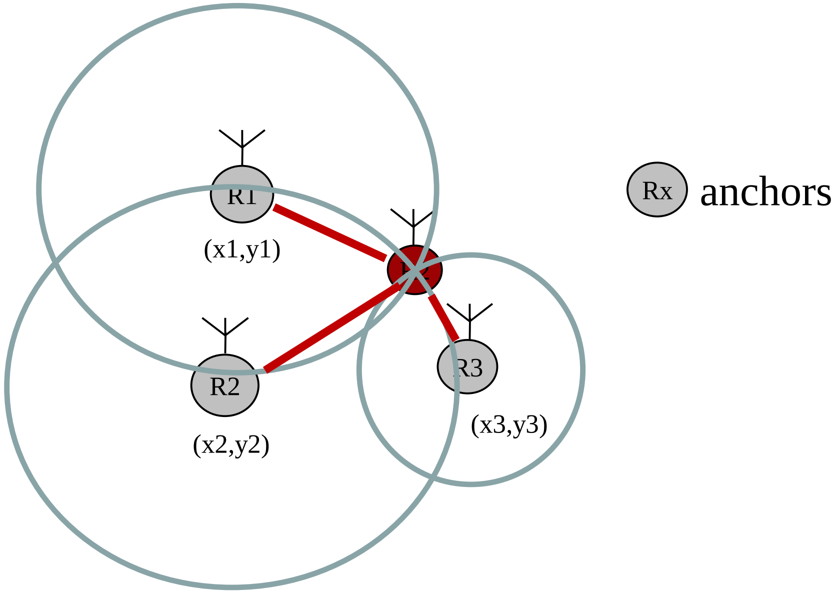
RF-based localization

- **Uses RF communication to measure distance / angle**
  - Attenuation models (RSSI), Time-of-Flight, Angle-of-Arrival, …
  - Makes use of the communication devices
- **Absolute**
  - With respect to **a fixed referential**
  - Requires **infrastructure** (anchors)
    - GPS, Base Stations (GSM), Access Points (WiFi), …
  - Localization of each individual agent in the infrastructure
    - Angulation, lateration, MLE, …
- **Relative**
  - Internal to the team
    - Intra-team distances / angles
  - **Infrastructure-free**

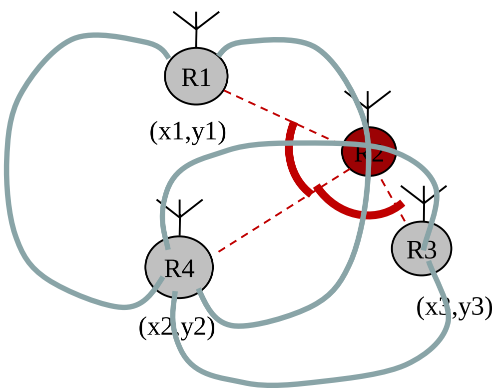
Higher flexibility (no anchors), Adequate to **unstructured environments**

Basic localization methods

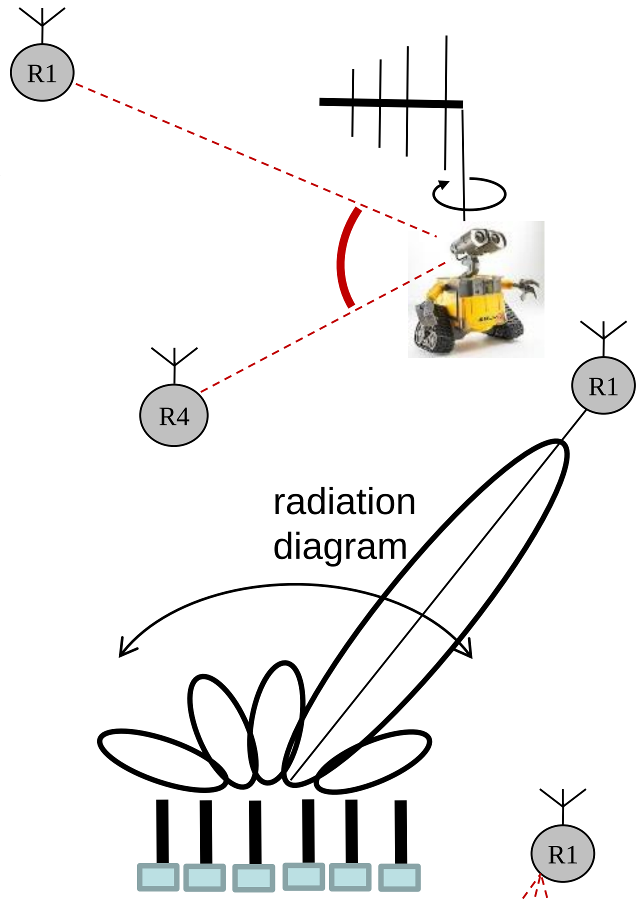
- **Lateration**
  - Measure **distances** to several known points (anchors)



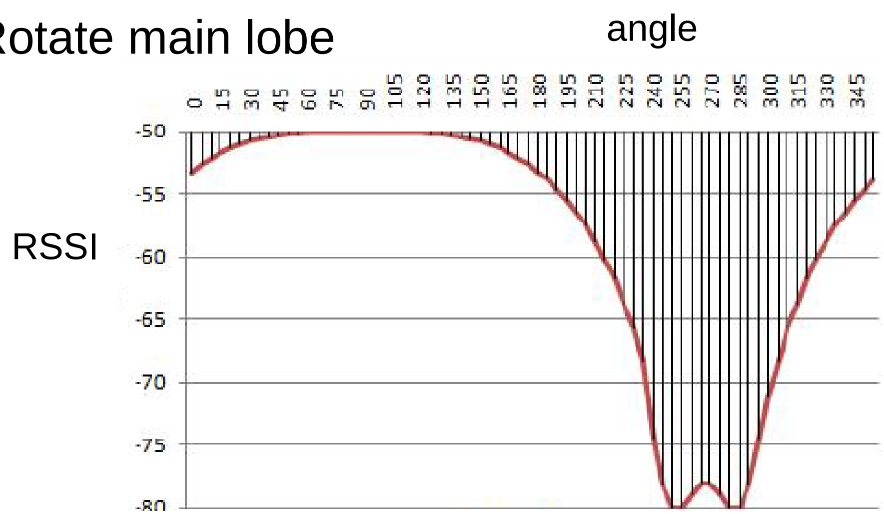
- **Angulation**
  - Measure **angles** to several known points (anchors)



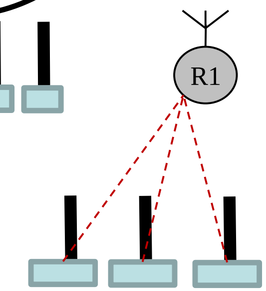
Measuring angles



- **Directional antennas**
  - Rotate antenna or rotate robot with odometry
- **Phased-arrays**
  - Rotate main lobe



- **Time based**
  - **Time Difference of Arrival – TDoA**
    - Differences in receiving time at anchors determines angle of arrival



U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Measuring distances

- **Time based**
  - Explore knowledge of propagation speed
    $d = v \Delta t$
  - High speed (e.g. speed of light) → short intervals…
    - Either use low speed waves (e.g. sound) or wide bandwidth (e.g., UWB, chirps)
  - **Time of Arrival – ToA**
    - Receiver knows when message was sent, computes distance on arrival
  - **Time Difference of Arrival – TDoA**
    - Sender *informs* receiver with fast message RF when slow message US was sent
  - **Time of Flight – ToF**
    - Sender sends a query, receiver loops back. Sender uses roundtrip time
- **Signal attenuation model**
  - Explore the attenuation of RF signals
    $$d = d_0 * 10^{\frac{P_0 - P_d}{10\alpha}}$$
    - Typically uses **Received Signal Strength Indicator**

Time of Flight – ToF

- **Round-trip delay of an electromagnetic wave**

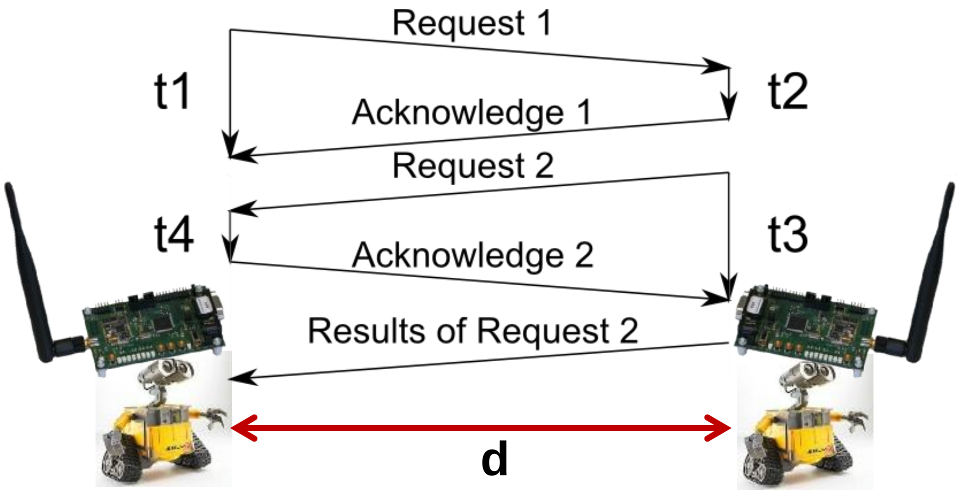
  Requires special communication devices (nanoLoc)

  - **Chirp modulation – CSS** (IEEE 802.15.4a) – 80MHz bandwidth channels
  - Round-trip loopback done in hardware with very high precision (t_2 and t_4)
  - Time interval measured with auto-correlation
  - Two ranging modes: fast ($d = r_1$) and precise ($d = \frac{r_1+r_2}{2}$)

$$r_1 = v \times \frac{(t_1 - t_2)}{2}$$

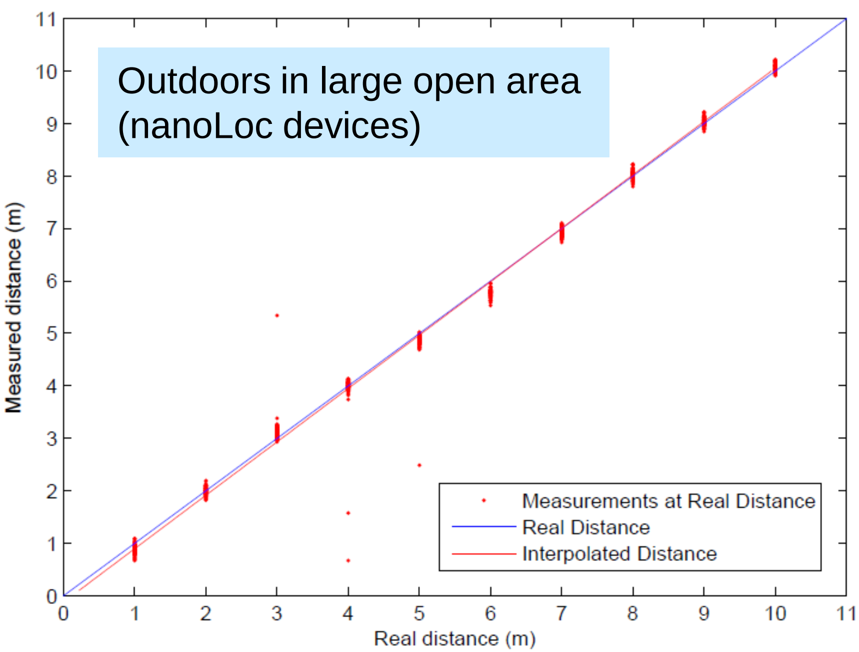
$$r_2 = v \times \frac{(t_3 - t_4)}{2}$$

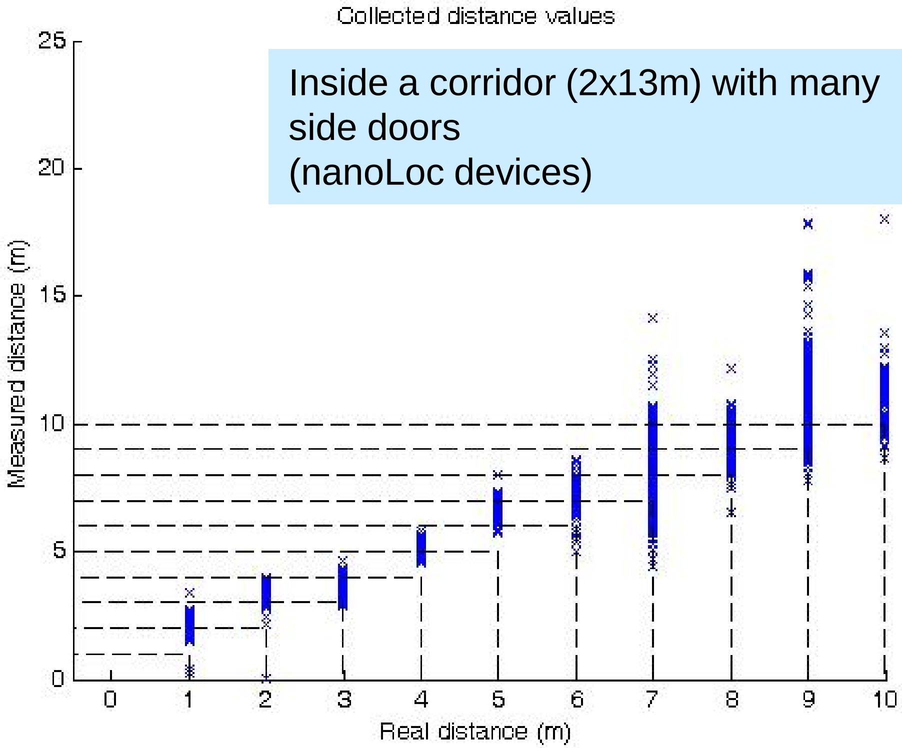
$$\boldsymbol{d = \frac{(r_1 + r_2)}{2}}$$



Time of Flight – ToF

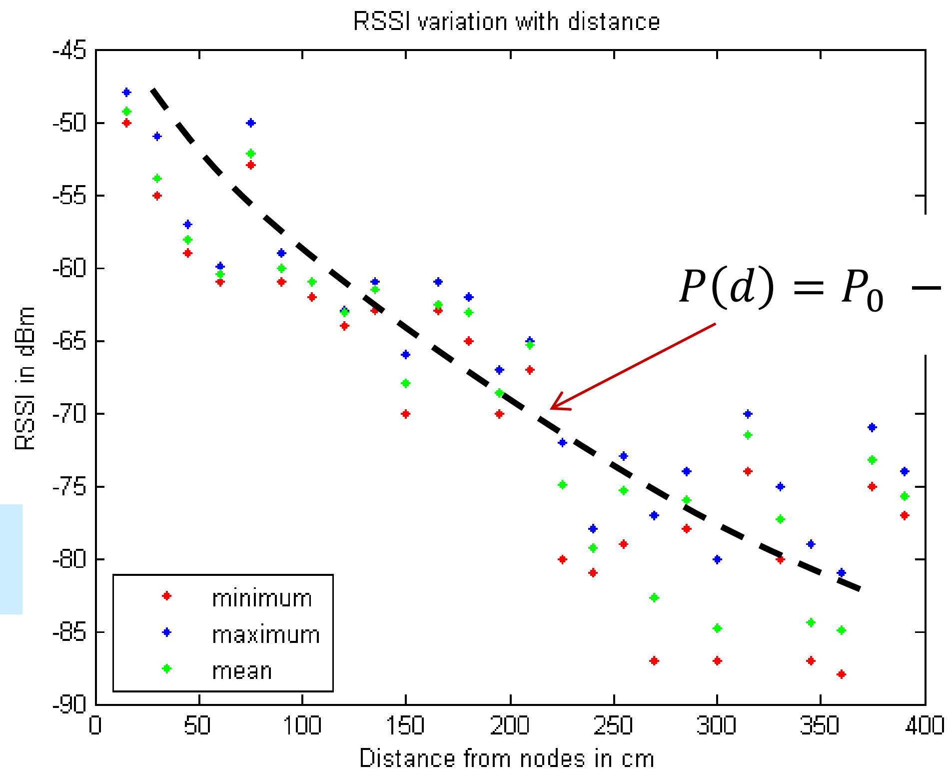
- **Still not perfect**
  - Bias (~1m) + noise
  - Still some noise





Received Signal Strength Indicator – RSSI

- **Attenuation model has complex behavior**
  - Highly dependent on the environment
  - Noisy information, particularly in closed spaces
    - Time varying errors (not stationary process)

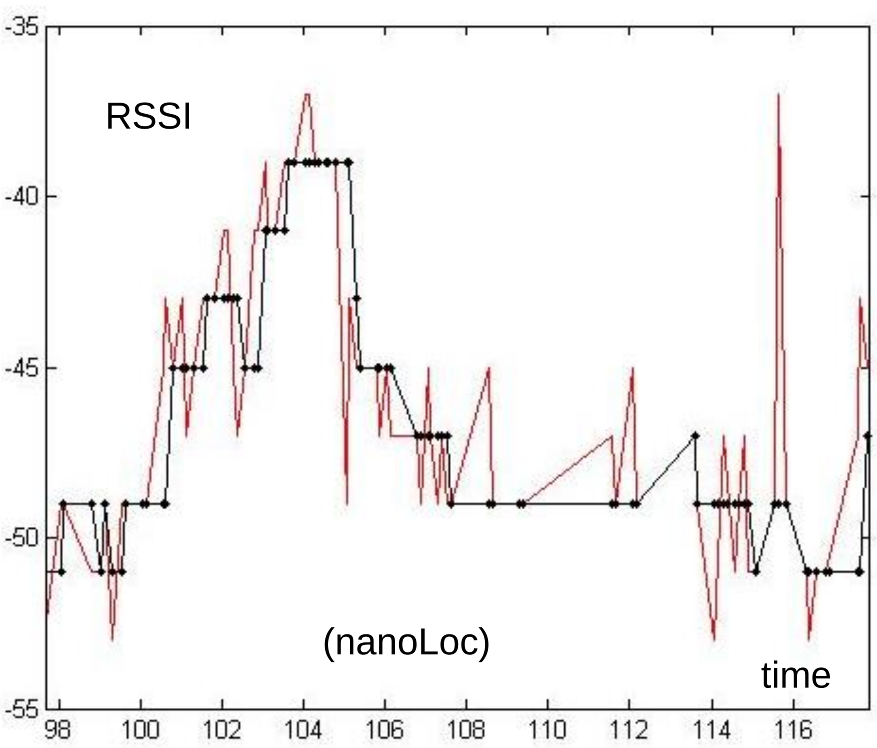


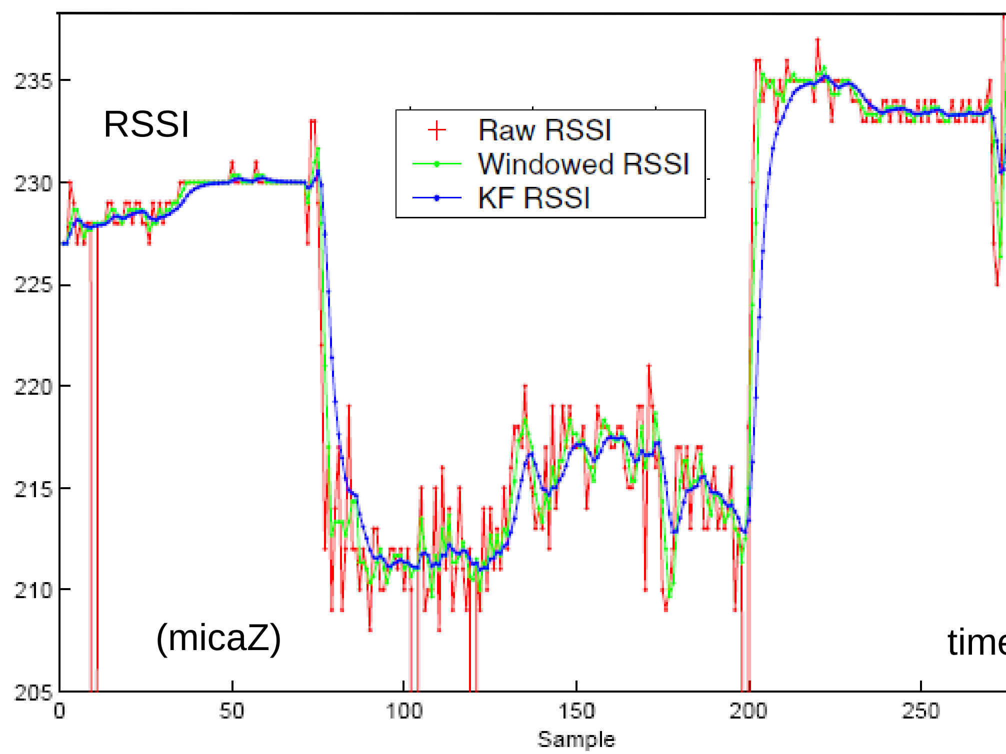
$$P(d) = P_0 - 10\alpha \log \frac{d}{d_0}$$

Inside lab (7x10m)
IEEE 802.15.4 (micaZ)

Improving RSSI quality

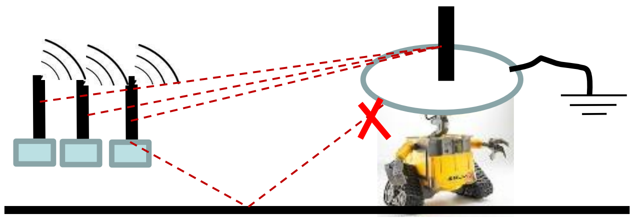
- **Median sliding window**
  - reduces impact of packet losses and outliers, delays signal
- **Kalman filter**
  - Softens but also delays the signal even more



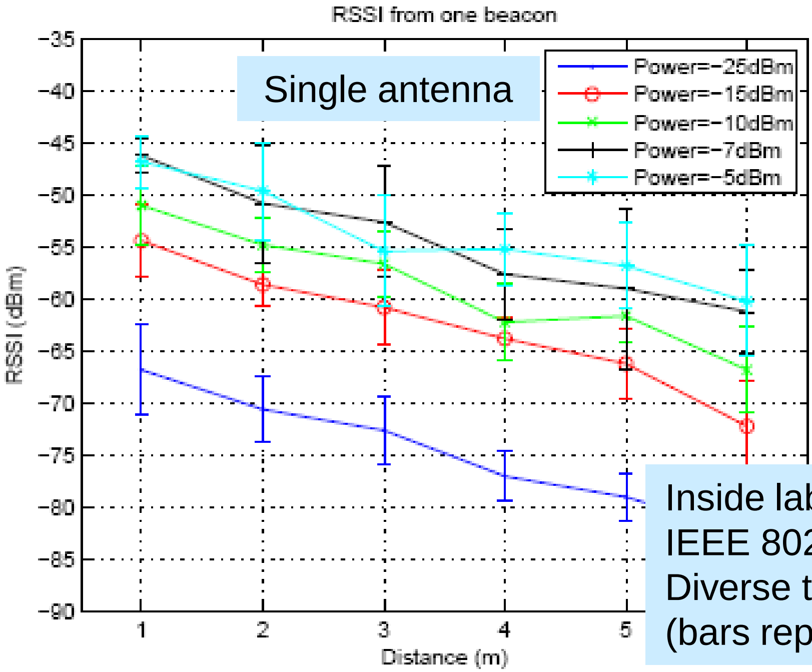


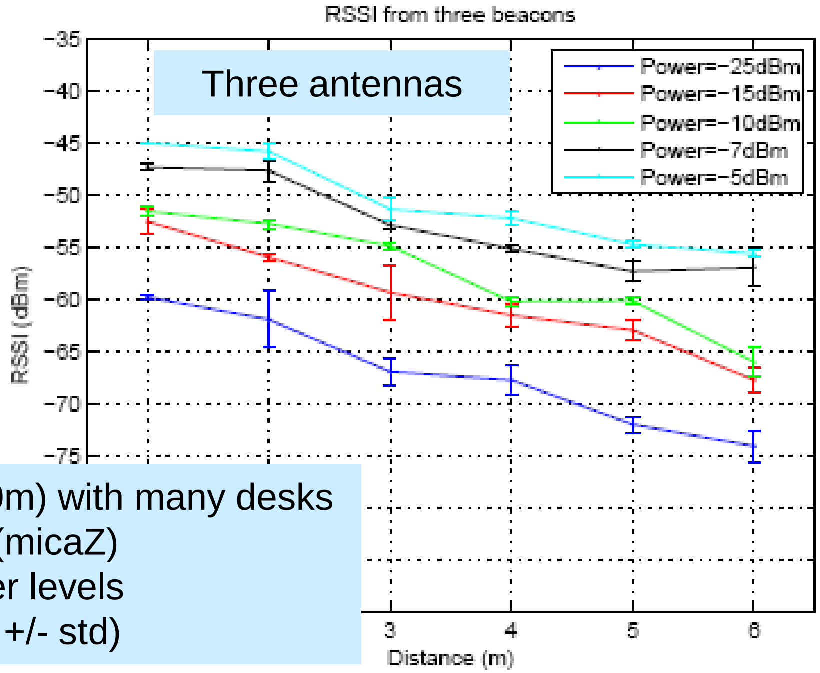
Improving RSSI quality

- Using a **shielding plane**
  - Reduces reflections on the floor



- **Antenna diversity**
  - Either at receiver or sender (create different paths)
  - Standard deviation is divided by the number of antennas





Inside lab (7x10m) with many desks
IEEE 802.15.4 (micaZ)
Diverse tx power levels
(bars represent +/- std)

ToF versus RSSI

ToF

- Plus
  - provides a **real distance**
  - **more precise** for localization
- Cons
  - **Slow measurements**
    - 1 range uses 20ms (nanoLoc)
    - 1 unit ranging 4 other → 80ms
    - 5 units need 400ms (+overheads)

RSSI

- Plus
  - **Fast measurements**

    ~ simultaneously for each message
- Cons
  - **Noisy** measurements
  - **Environment dependent** model
  - **Lower precision**
    - depends on model

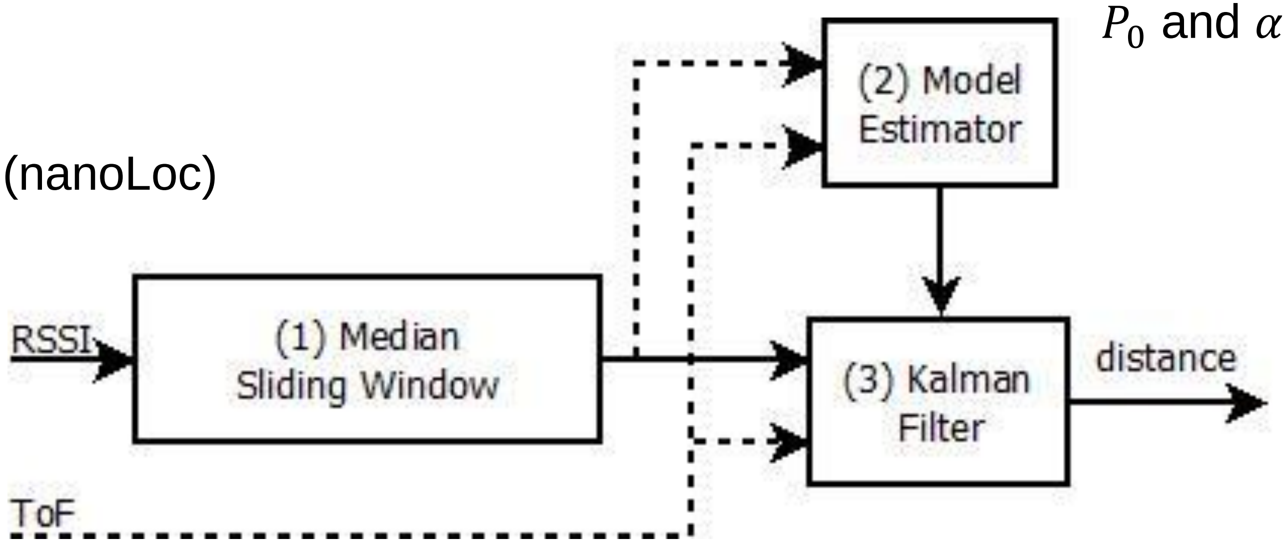
Combining ToF and RSSI

- **RSSI–based system**
  - Good for **fast measurements**
- **Update attenuation model with ToF**
  - From time to time take ToF measurements
  - Update the **attenuation model**

$$d = d_0 * 10^{\frac{\boldsymbol{P_0} - P_d}{10\boldsymbol{\alpha}}}$$

Online identification of P_0 and α

(nanoLoc)



Combining ToF and RSSI

- **Extended Kalman filter**

$$X = \begin{bmatrix} d \\ \dot{d} \end{bmatrix}$$

$$X_k = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \mathrm{X}_{k-1} + \begin{bmatrix} \frac{\Delta t^2}{2} & 0 \\ 0 & \Delta t \end{bmatrix} \omega(k)$$

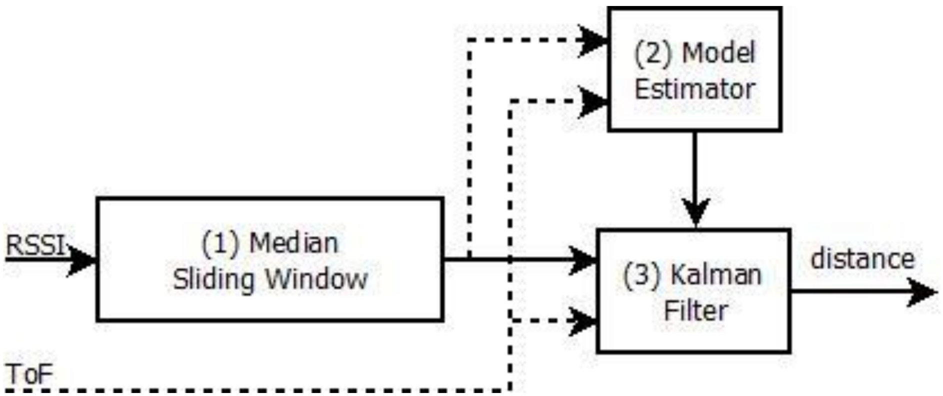
<u>Measurement equations</u>

RSSI and ToF

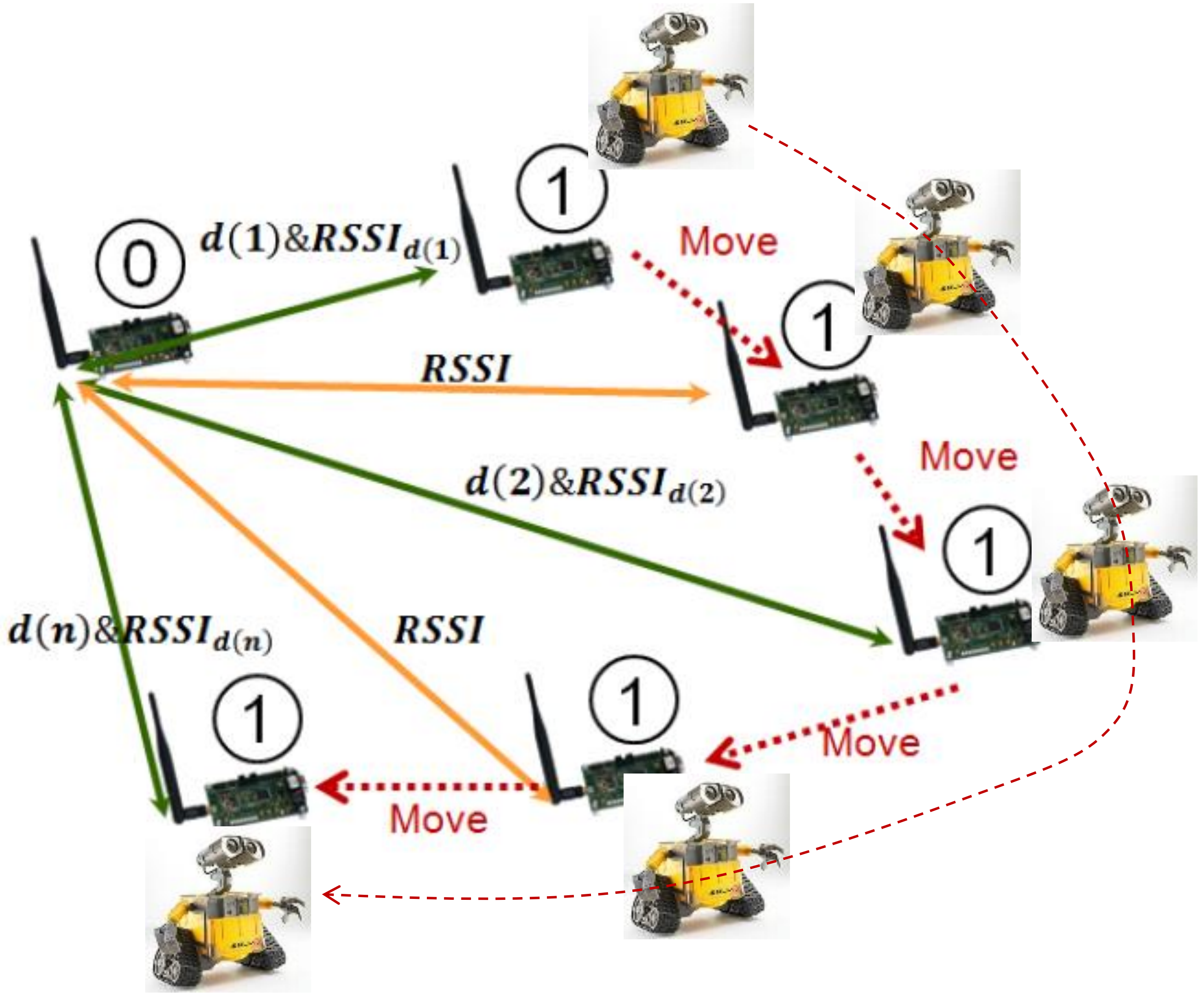
$$\begin{bmatrix} \bar{d} \\ \overline{P_d} \end{bmatrix}_k = \begin{bmatrix} d_k - bias_d \\ P_0 - 10\alpha \log d_k \end{bmatrix} + \nu(k)$$

$$[\overline{P_d}]_k = [P_0 - 10\alpha \log d_k] + \nu(k)$$

RSSI only



Combining ToF and RSSI



Combining ToF and RSSI

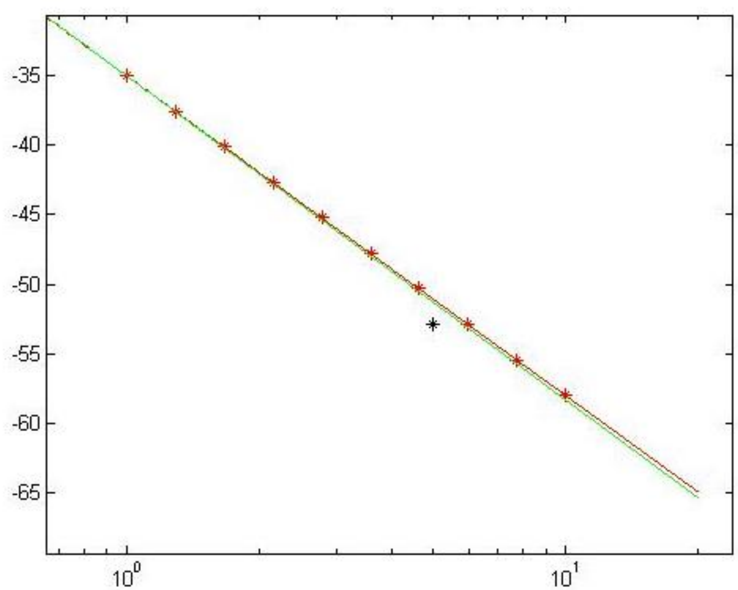
- **Online identification** using **MLE** (Max Likelihood Estimation)
  - Force the new model to match the previous one in a set of n points (s)
  - Integrate the new measurement

n determines the weight of the new measurement

$$\forall_{i=1..n} \quad P_{0,t} - 10\alpha_t \log s_i = P_{0,t-1} - 10\alpha_{t-1} \log s_i$$

$$P_{0,t} - 10\alpha_t \log d = P_d$$

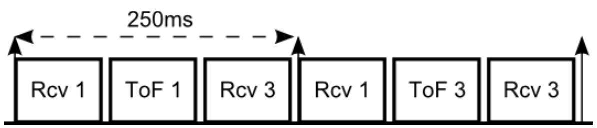
The ***s*** points are log evenly spaced to improve MLE performance

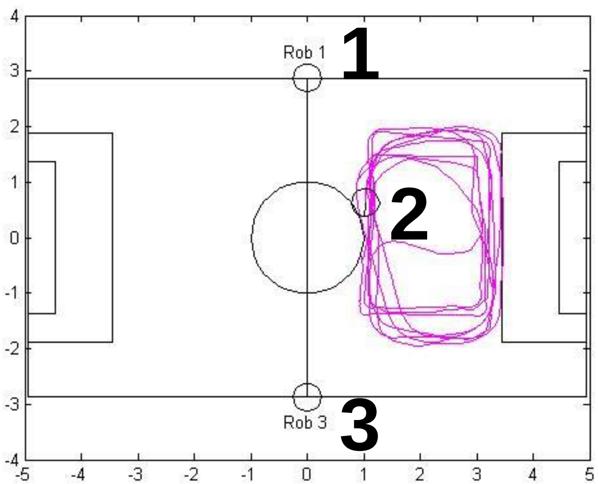


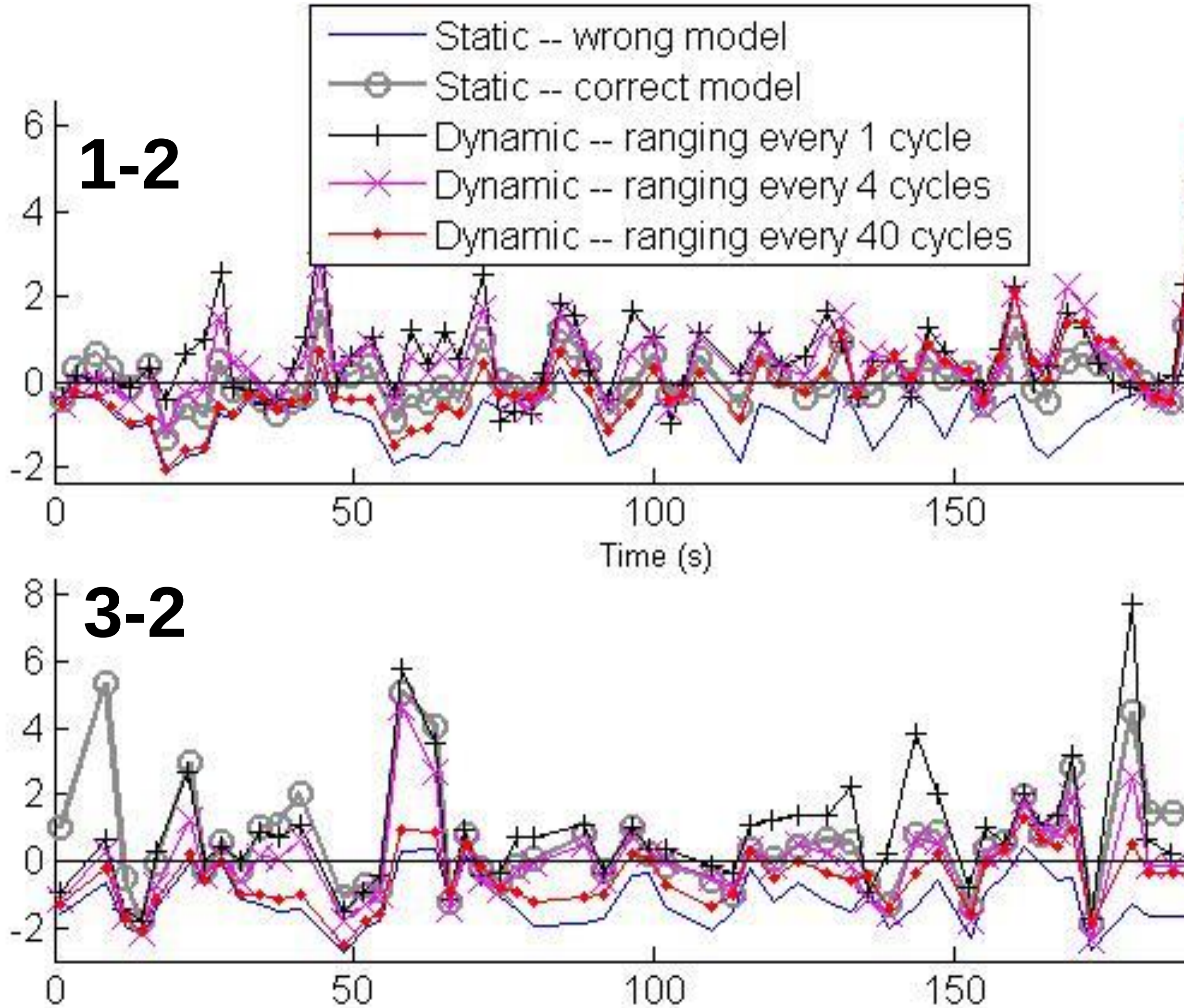
$$\Leftrightarrow \quad \mathbf{A}_t X_t = \mathbf{b}_t \qquad X_t = \begin{bmatrix} P_{0,t} \\ \alpha_t \end{bmatrix}$$

$$\widehat{X_t} = \begin{bmatrix} \widehat{P_{0,t}} \\ \widehat{\alpha_t} \end{bmatrix} = \left(\boldsymbol{A}_t{}^{T} \boldsymbol{A}_t\right)^{-1} \boldsymbol{A}_t{}^{T} \boldsymbol{b}_t$$

Combining ToF and RSSI







Signal strength space

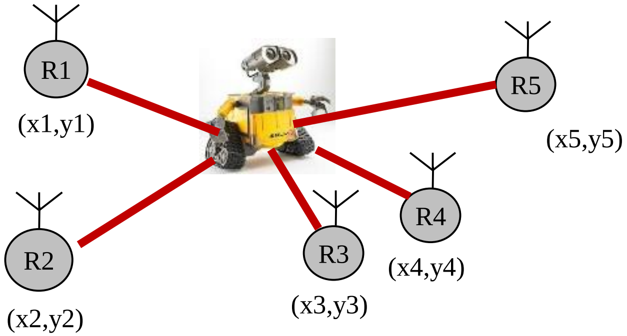
- **Accuracy is not always needed**
  - To **drive a robot** to the **area of a beacon**, or to **near another robot** it is sufficient to have a **course notion** of relative **localization**
  - This covers many practical cases of team **navigation**
    - approach, change side, keep connectivity
  - The robot approaches the desired target stochastically
- **Signal strength distance**
  - Is a direct use of **RSSI as a *distance***
  - Still requires RSSI filtering but does not use the signal attenuation model

$$\tilde{d}(t) = RSSI^{\max} - RSSI(t)$$

RF-based localization and navigation

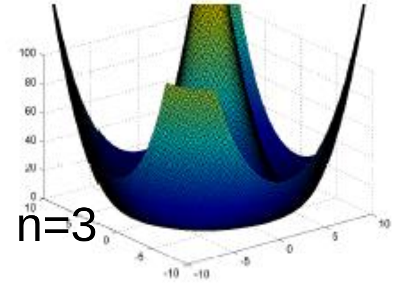
Localize wrt *n* known points

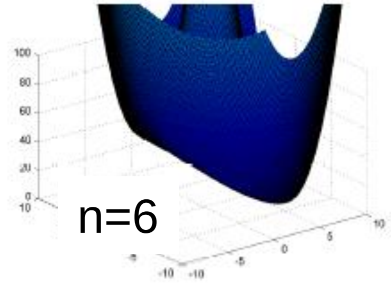
- **Multi–Lateration**
  - Get distances to three or more known points
  - Consolidate info, e.g. MLE
    - N. of points & location have a significant impact on precision



$$F(x,y) = \sum_{i=1}^{n} \left(d_i - \sqrt{(x - x_i)^2 - (y - y_i)^2} \right)^2$$

Least squares error





Estimate point that best matches *n* distances

- Formulate as **Least Squares problem** and **apply MLE**
  - Using **n known points (x_i, y_i)**

$$\begin{cases} (x-x_1)^2-(y-y_1)^2={d_1}^2 \\ (x-x_2)^2-(y-y_2)^2={d_2}^2 \\ \dots \\ (x-x_n)^2-(y-y_n)^2={d_n}^2 \end{cases}$$

$$A=\begin{bmatrix} 2(x_1-x_n) & 2(y_1-y_n) \\ \dots & \dots \\ 2(x_{n-1}-x_n) & 2(y_{n-1}-y_n) \end{bmatrix}_{(n-1)\times 2}$$

$$\Leftrightarrow$$

$$b=\begin{bmatrix} {x_1}^2-{x_n}^2+{y_1}^2-{y_n}^2-{d_1}^2+{d_n}^2 \\ \dots \\ {x_{n-1}}^2-{x_n}^2+{y_{n-1}}^2-{y_n}^2-{d_{n-1}}^2+{d_n}^2 \end{bmatrix}_{(n-1)\times 1}$$

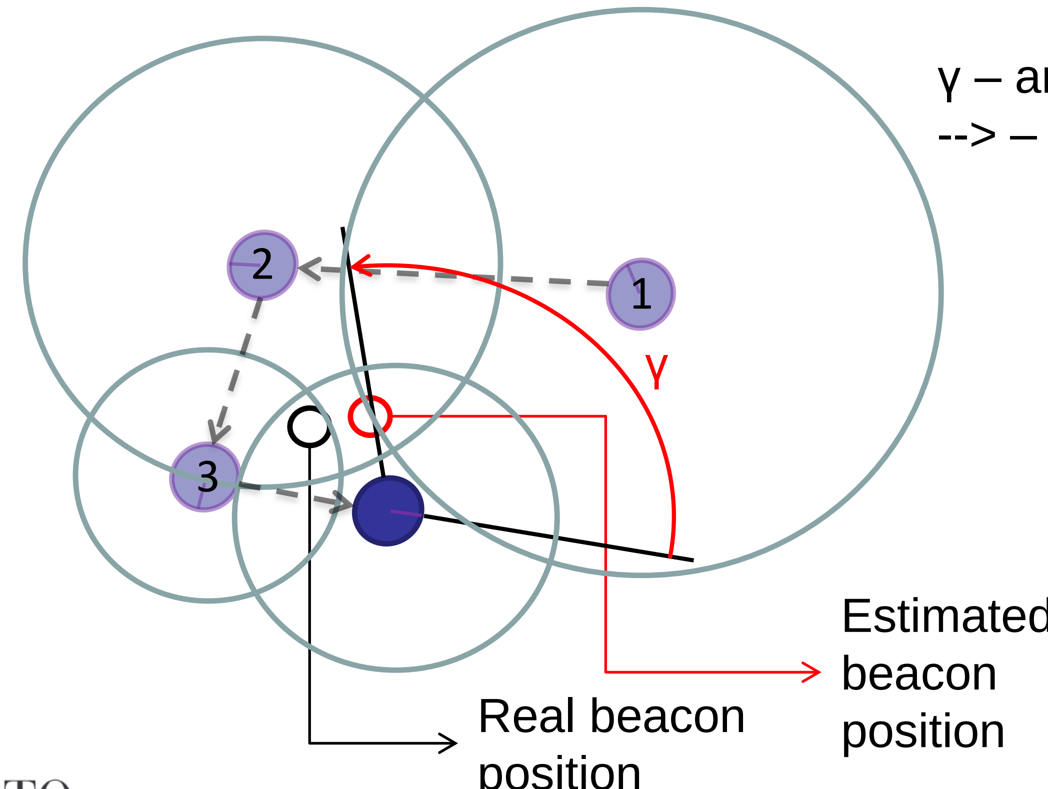
$$AX=b$$

$X=\begin{bmatrix} x \\ y \end{bmatrix}$ unknown point

$$\hat{X}=(A^TA)^{-1}A^Tb$$

Moving to localize a source

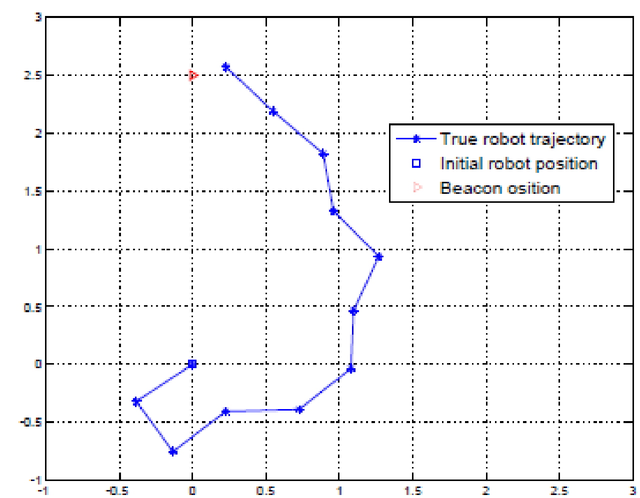
- **Finding other team mates / beacons**
  - Carry out a sequence of movements and distance measurements
  - Requires odometry for relative displacements, good use case for MLE



γ – angle robot should turn

--> – path

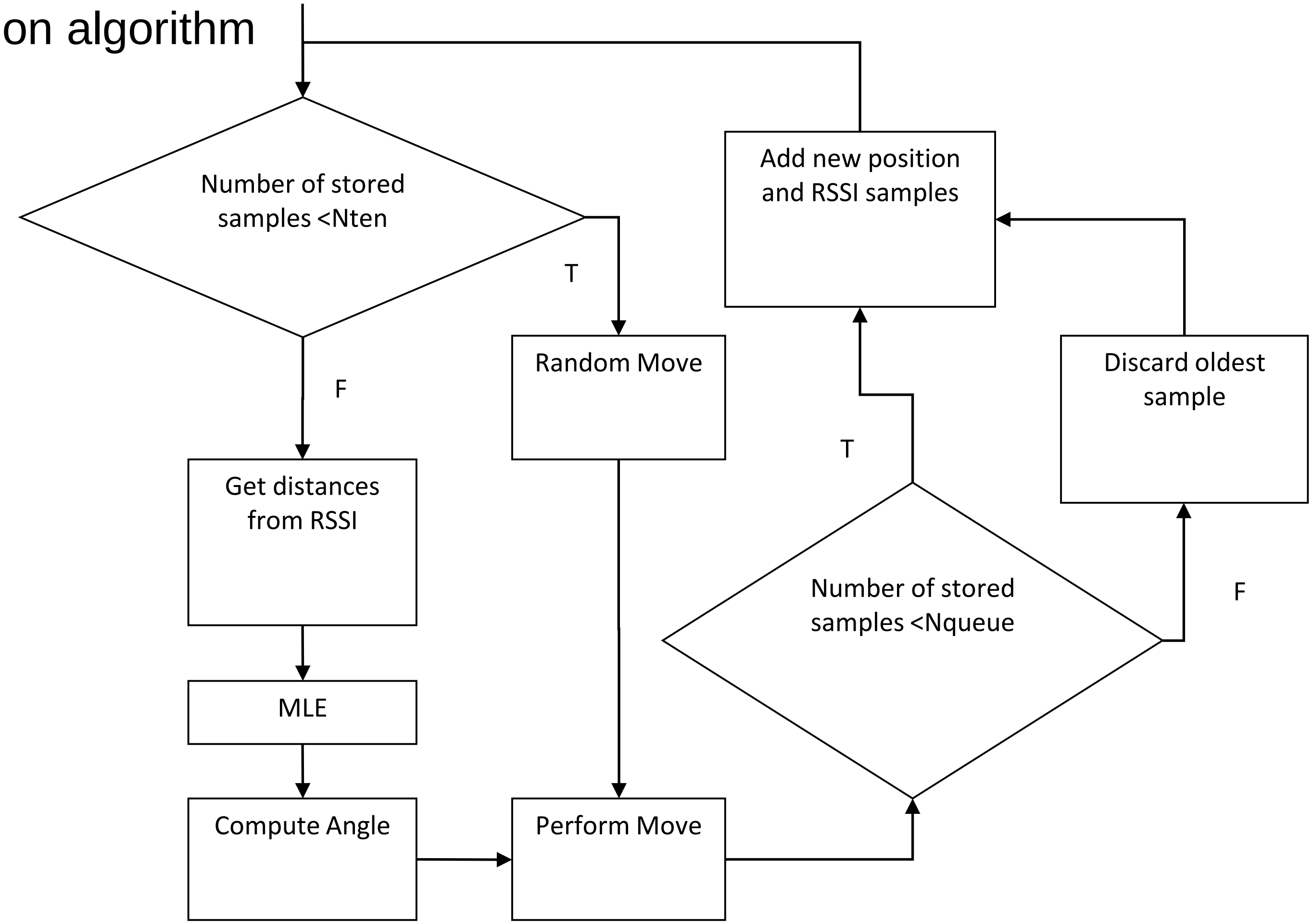
Previous positions are used as *anchors*

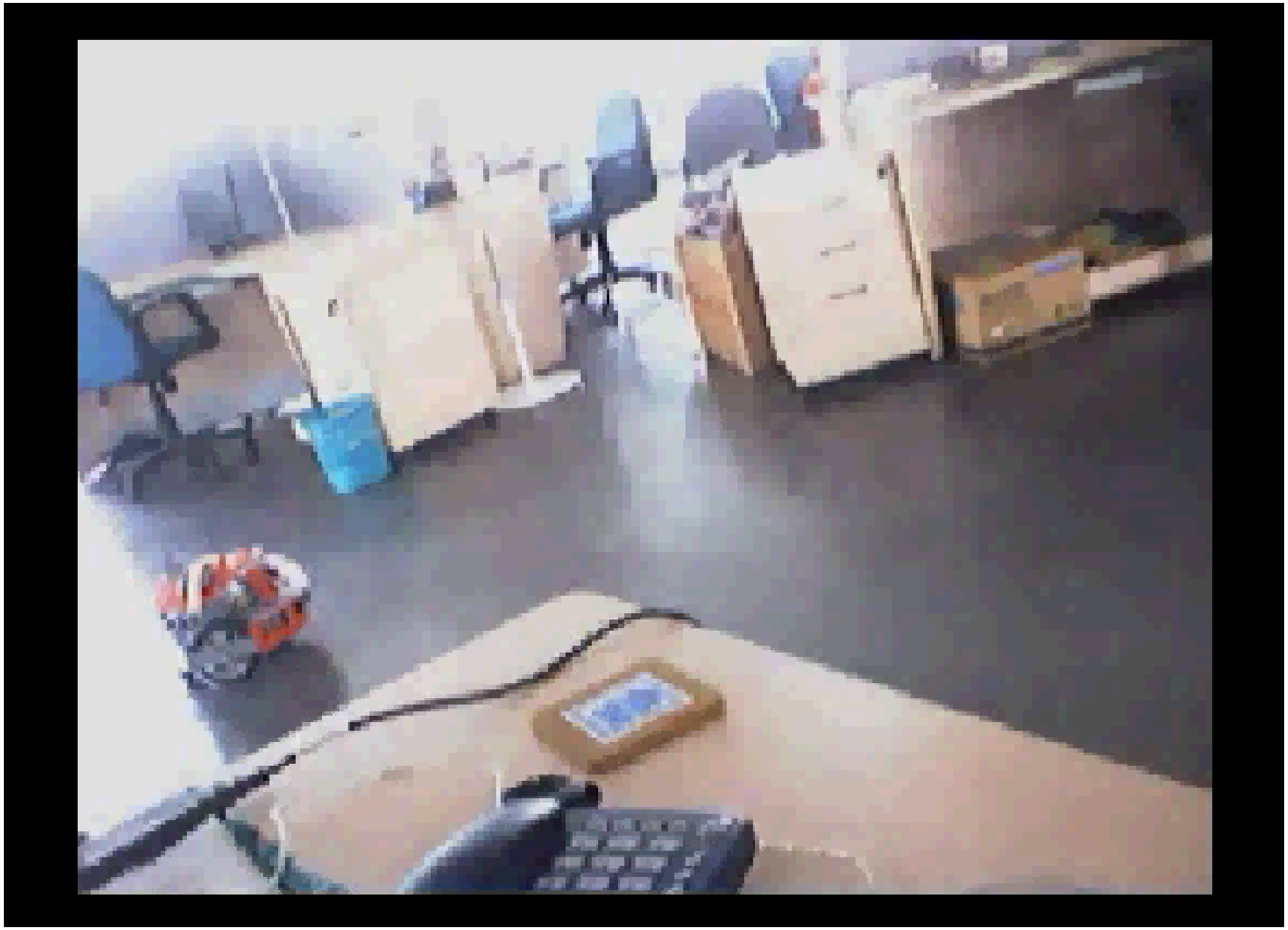


(Luis Oliveira, MSc dissertation, Univ Aveiro, 2009)

Moving to localize a source

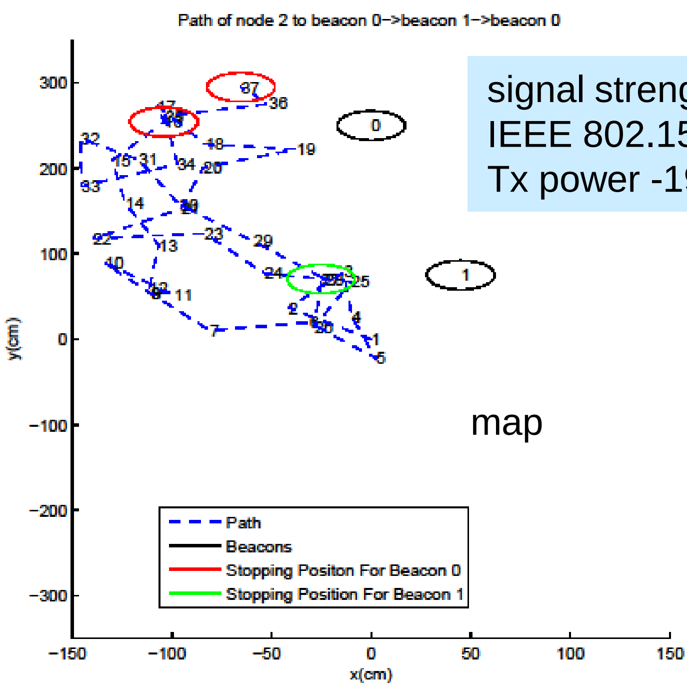
- **Finding other team mates / beacons**
  - Navigation algorithm





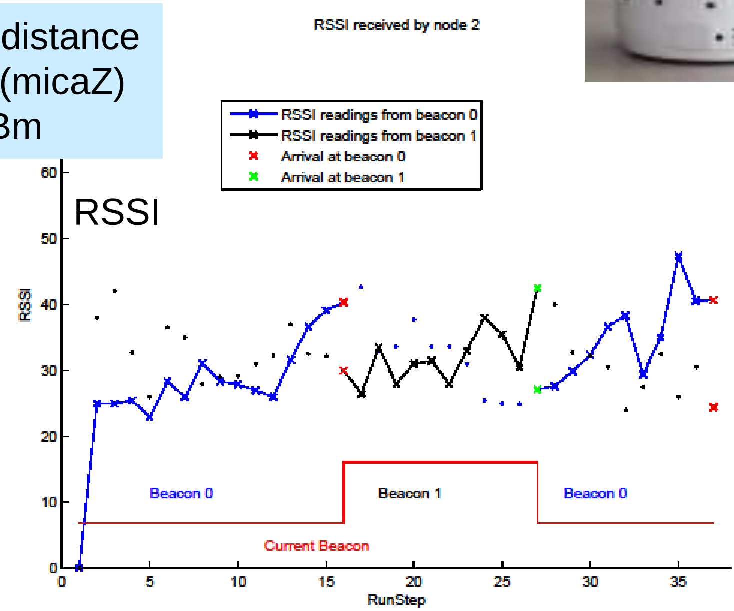
Following a path of sources

- **Simple and flexible navigation through a beacons path**
  - Make a robot follow a beacon. When reached, move to the next
    - Stopping condition based on RSSI threshold
  - Easy deployment (drop RF beacons)
    - Suitable for surveillance (RoboVigil)



map

signal strength distance
IEEE 802.15.4 (micaZ)
Tx power -19dBm



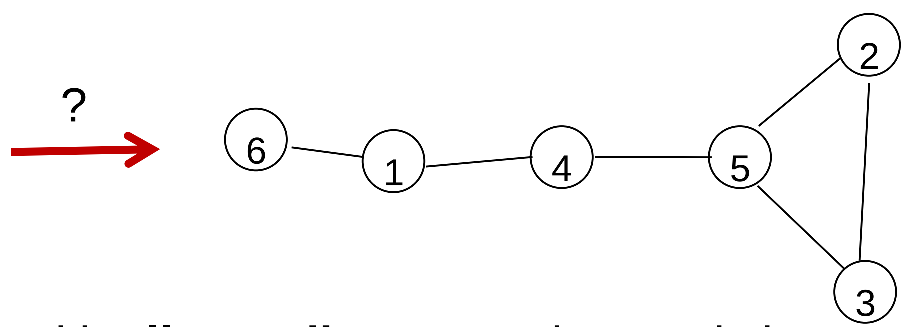
RSSI

Find *n* points that best match *m* distances

- **Topology reconstruction** problem
  - From **pair-wise distances** between team members find a compatible set of **relative positions** in space
  - Many **ambiguities** inherent to relative localization
    - Agnostic to “rigid” transformations (translation, reflection, rotation)
    - Certain ambiguities **can be resolved with movements**! (see later on)
    - Certain ambiguities **cannot be resolved with relative measurements**, only
      - e.g., when the connectivity matrix is **incomplete** and/or **inconsistent**

$M^3(t)$

	1	2	3	4	5	6
1	0	-	-	10	-	7
2	-	0	12	-	8	-
3	-	12	0	-	11	-
4	10	-	-	0	11	-
5	-	8	11	11	0	-
6	7	-	-	-	-	0



With **all true distances**, the matrix is **symmetrical and complete →** the topology can be univocally recovered

Find *n* points that best match *m* distances

- Applying **Multi-Dimensional Scaling (MDS)**
  - Technique from multivariate analysis to visualize data in a given space

    From an ***n* x *n* (dis)similarities matrix** finds **n points in an m-dimensional Euclidean space** so that **their pair-wise distances** are compatible with the (dis)similarities given

    $X = \left[x_{ij}\right]_{n\times m}$ n points in the m-dimensional Euclidean space

    $d_{ij}(X) = \left(\sum_{a=1}^{m}\left(x_{ia} - x_{ja}\right)^2\right)^{\frac{1}{2}}$ pair-wise distances in X

    δ_{ij} measured pair-wise distances, weighted by w_{ij}

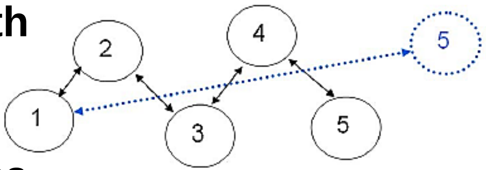
Problem: find X that minimizes $\sigma(X)$

$$\sigma(X) = \sum_{i<j} w_{ij}\left(d_{ij}(X) - \delta_{ij}\right)^2$$

MDS provides a closed formula solution

Find *n* points that best match *m* distances

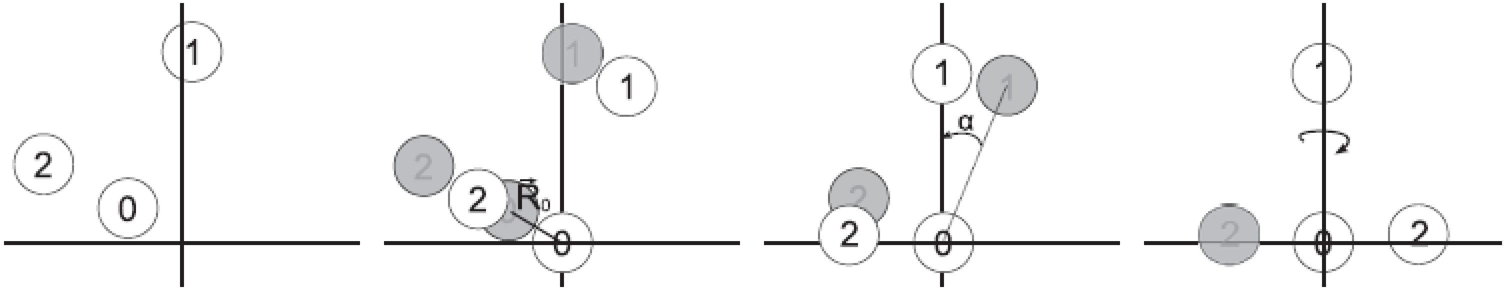
- When the **measurements** are **noisy** and **incomplete**
  - Message losses, errors…
  - Apply **MDS variants**: Ordinal MDS, weighted MDS, iterative MDS
  - Alternatively, we can **complete and adjust** the connectivity matrix
    - To make it fulfil the requirements of classic MDS
  - Filling in distances for unconnected nodes (Floyd-Warshall algorithm)
    - **Sum the distances along the shortest path**
  - **Force symmetry**
    - e.g., average of the upper and lower triangles



	1	2	3	4	5	6
1	0	-	-	10	-	7
2	-	0	12	-	8	-
3	-	12	0	-	11	-
4	10	-	-	0	11	-
5	-	8	11	11	0	-
6	7	-	-	-	-	0

Obtaining a smoothly varying topology

- MDS can cause **sudden “jumps”** due to errors in positions
  - Tie **resulting positions** to pre-defined **referential / orientation**
    - **Node 0** at the **origin** (logical IDs)
    - **Node 1** on the **positive side of YY**
    - **Node 2** on the **right half plane**



$R = [r_{ij}]_{n\times 2}$ coordinates given by MDS

$S = [s_{ij}]_{n\times 2}$ adjusted coordinates

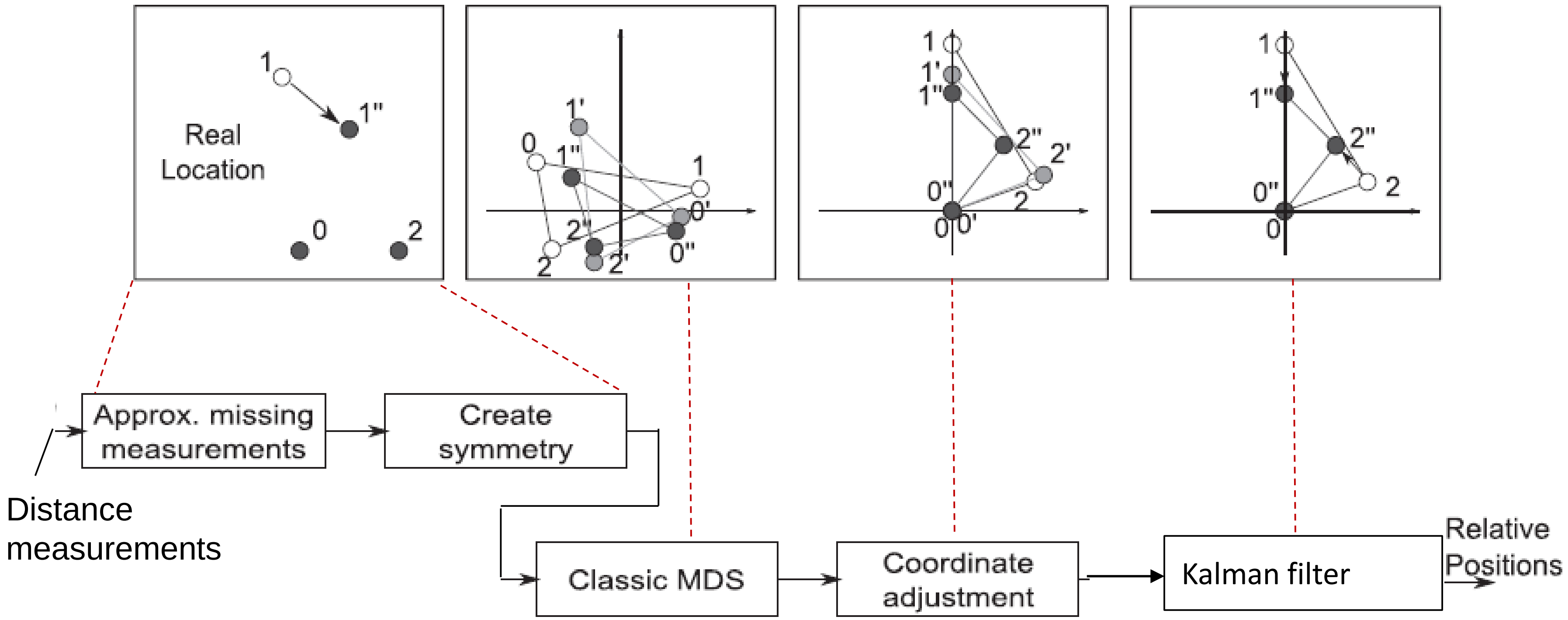
$$Y = (R - r_0) \times \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix}$$

$$S = \begin{cases} Y, & y_2 \text{ is on rhp} \\ Y \times \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}, & \text{otherwise} \end{cases}$$

U.PORTO FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

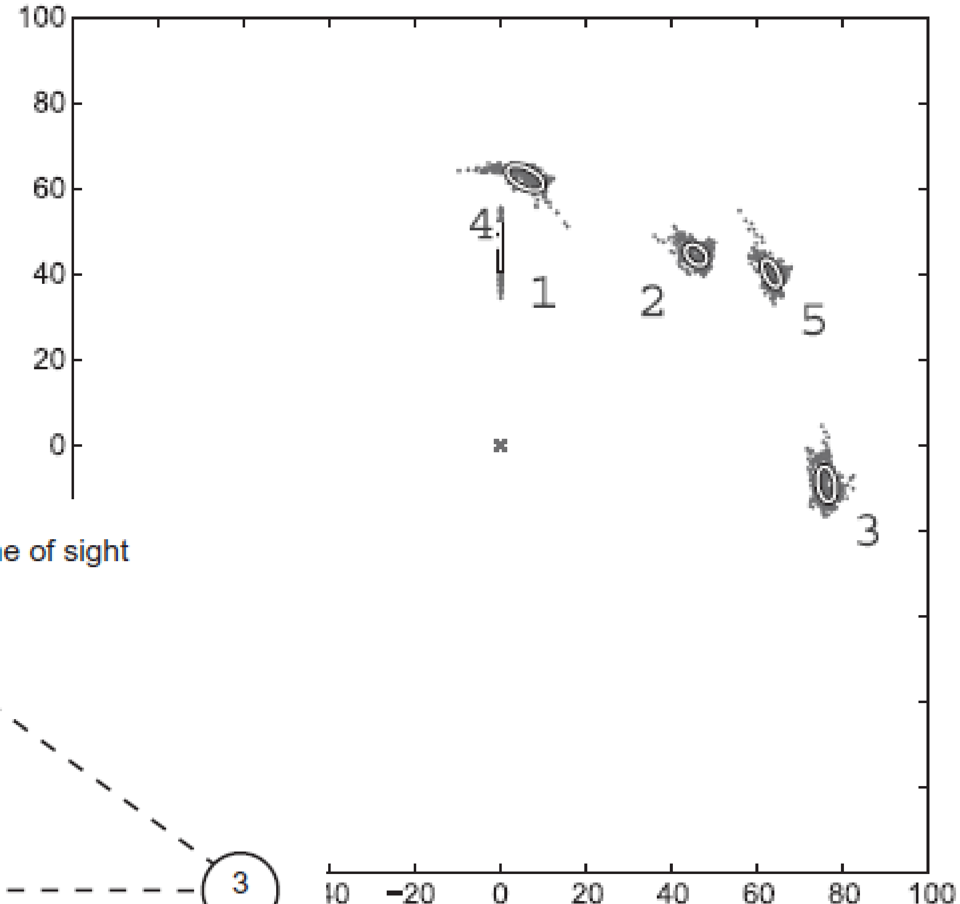
Obtaining a smoothly varying topology

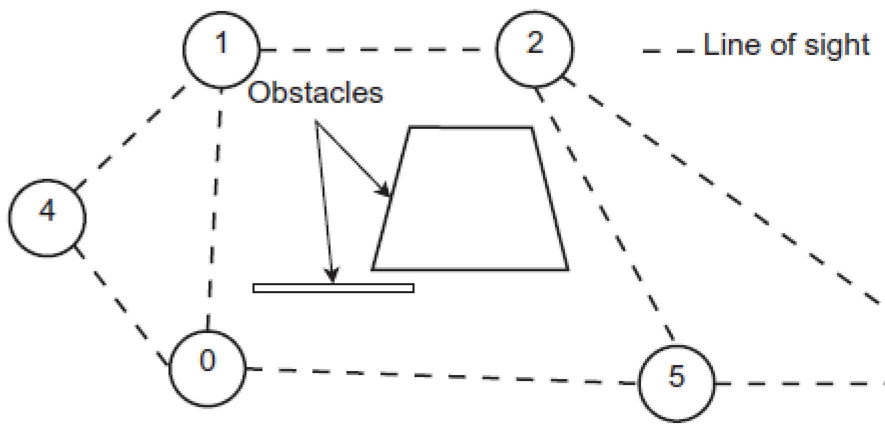
- **Movements** can be further **smoothed** applying a **Kalman filter**
  - Team localization process



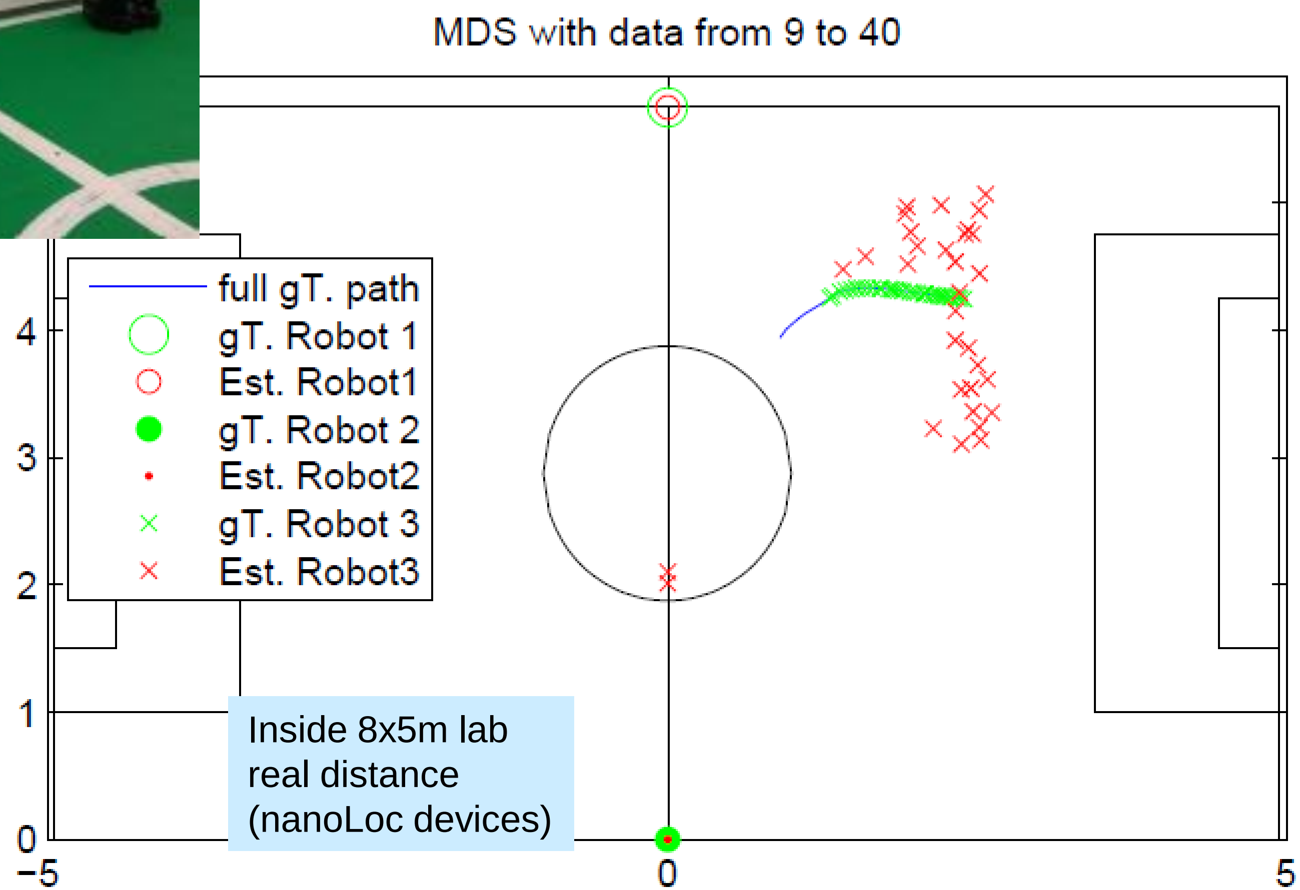
Estimating a topology with MDS

Inside 7x7m lab
signal strength distance
IEEE 802.15.4 (micaZ)
Tx power -10dBm





Estimating the position of a moving node



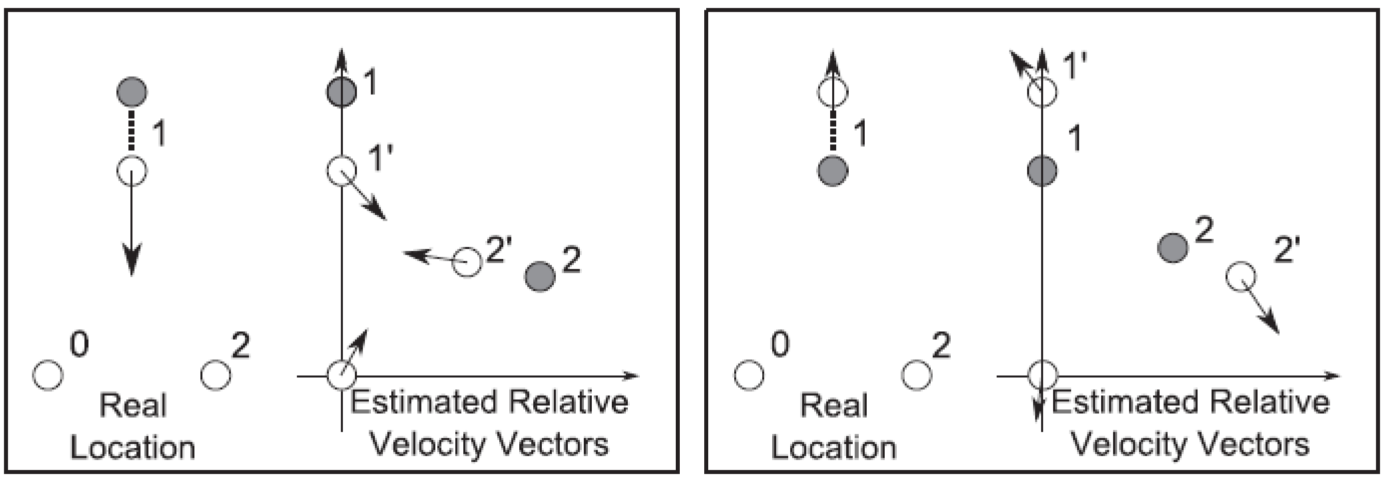
Inside 8x5m lab
real distance
(nanoLoc devices)

(Carmelo Di Franco, MSc dissertation, SSSUP, 2013)

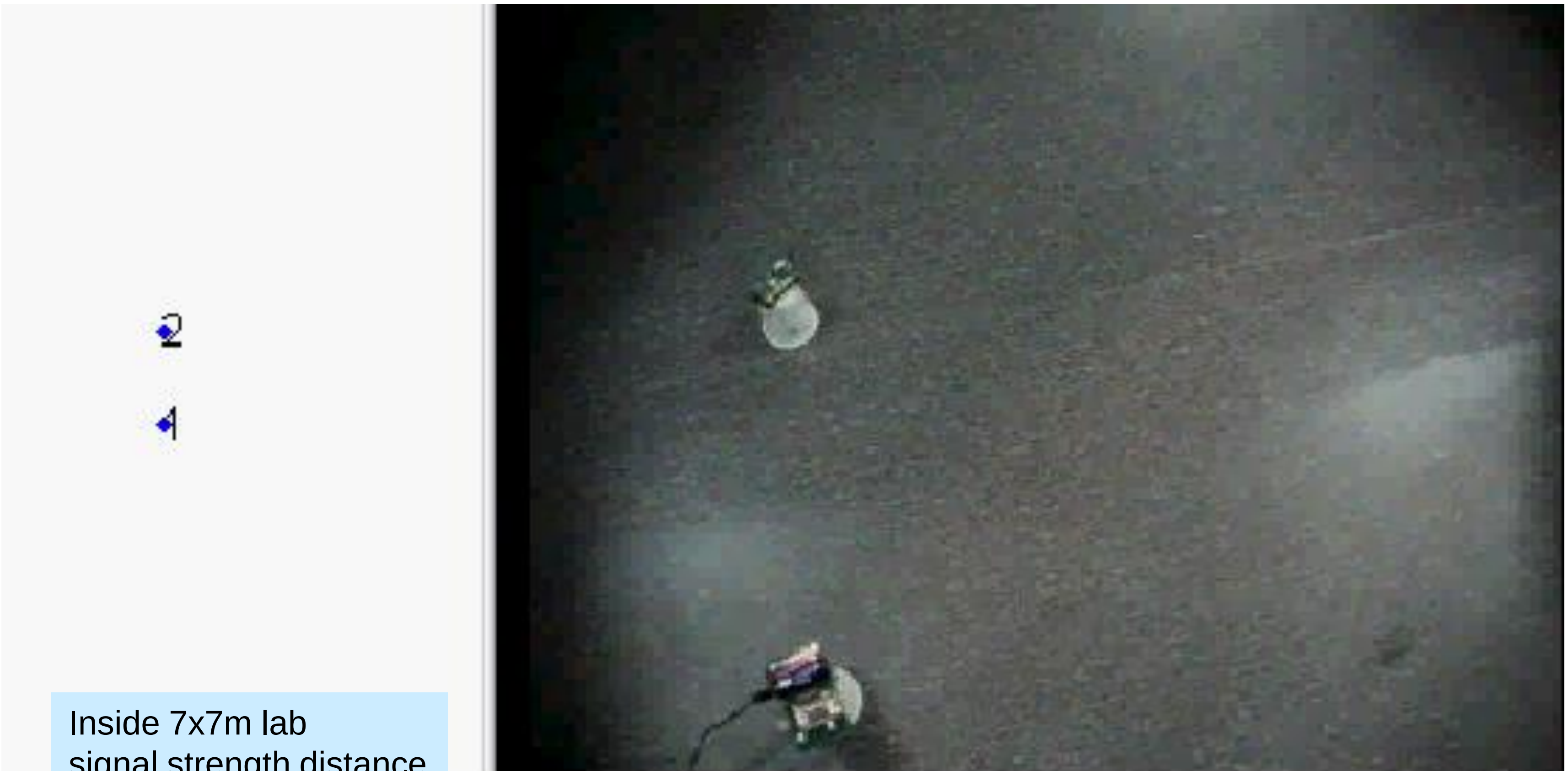
Estimating velocity from topology

- Identifies **approximating** and **moving away** nodes

$$v_i = \frac{1}{T_t} \sum_{j=1..n \neq i} \left(d_{ij} - d_{ij}^{-} \right) \frac{s_i - s_j}{\left| s_i - s_j \right|}$$



Estimating velocity from topology



Inside 7x7m lab
signal strength distance
IEEE 802.15.4 (micaZ)
Tx power -10dBm

Global team applications

Dynamic Target Tracking with Integration of Communication and Coverage using Mobile Sensors

Hongbin Li, Luis Almeida, Youxian Sun.

IECON 2009 – IEEE Conf. on Industrial Electronics. Porto, Portugal. Nov 3-5, 2009.

Dynamic targets tracking

- **Develop decentralized motion coordination**
  - For a set of **N** autonomous robots (mobile sensors)
  - To **track non-cooperative dynamic targets**
- **Use multi-objective optimization to**
  - Maitain a good **sensing capability**
  - Keep the mobile nodes **connected**
  - Assure a good **area coverage**

Information sharing based on RTDB

$$J = J_{Cov} + J_{Com} + J_{Sense}$$

Global cost function

Individual cost factors

- **Sensing cost**

Fused estimate error covariance

Sensing covarianves in polar coordinates

$$J_{Sense} = det(\mathbf{P}_{fused}) = det(\sum_{i}^{M} \mathbf{R}_i^{-1})^{-1} \qquad \overline{\mathbf{R}}_i \sim \begin{bmatrix} (\sigma^i_{range})^2 & 0 \\ 0 & (\sigma^i_{bearing})^2 \end{bmatrix}$$

- **Connectivity cost**

$$J_{Com} = \sum_{i=1, j=i+1}^{i=N'-1, j=N'} \frac{1}{SNR_{ij}} \qquad SNR_{ij} = \frac{\kappa}{d_{ij}^2}$$

Weighting factor

Approximation of the communications signal to noise ratio

- **Area coverage cost**

$$J_{Cov} = -\frac{A_{1-covered}}{A_{tot}} \qquad J_{Cov,ij} = -\frac{A_i + A_j - A_{i\cap j}}{A} \qquad A_{i\cap j} = F(d_{ij}, r_i, r_j)$$

Percentage of area covered by at least 1 sensor

Approximation of the intersection area

Decentralized control law

- **Control actions using gradient descent**

$$\mathbf{u}_{Sense,i}(x_i, y_i) = T_i^T \mathbf{u}_{Sense,i}(r_i, \theta_i) \qquad \mathbf{u}_{Sense,i}(r_i, \theta_i) = \left[(\frac{\partial J_{sense}}{\partial r_i}), \quad \frac{1}{r_i}(\frac{\partial J_{sense}}{\partial \theta_i}) \right]$$

Control action in cartesian coordinates

$$\mathbf{u}_{Com,i} = (\frac{\partial J_{Com,i}}{\partial x_i}, \frac{\partial J_{Com,i}}{\partial y_i}) = \frac{2}{\kappa} \sum_{j=1,j\neq i}^{j=N} \left(x_i - x_j, \quad y_i - y_j\right)$$

Approximation of the intersection area

$$\mathbf{u}_{Cov,i} = (\frac{\partial J_{Cov,i}}{\partial x_i}, \frac{\partial J_{Cov,i}}{\partial y_i}) = \sum_{j=1,,j\neq i}^{j=M} G(d_{ij}, r_i, r_j)\left(x_i - x_j, y_i - y_j\right)$$

- **Final control law**

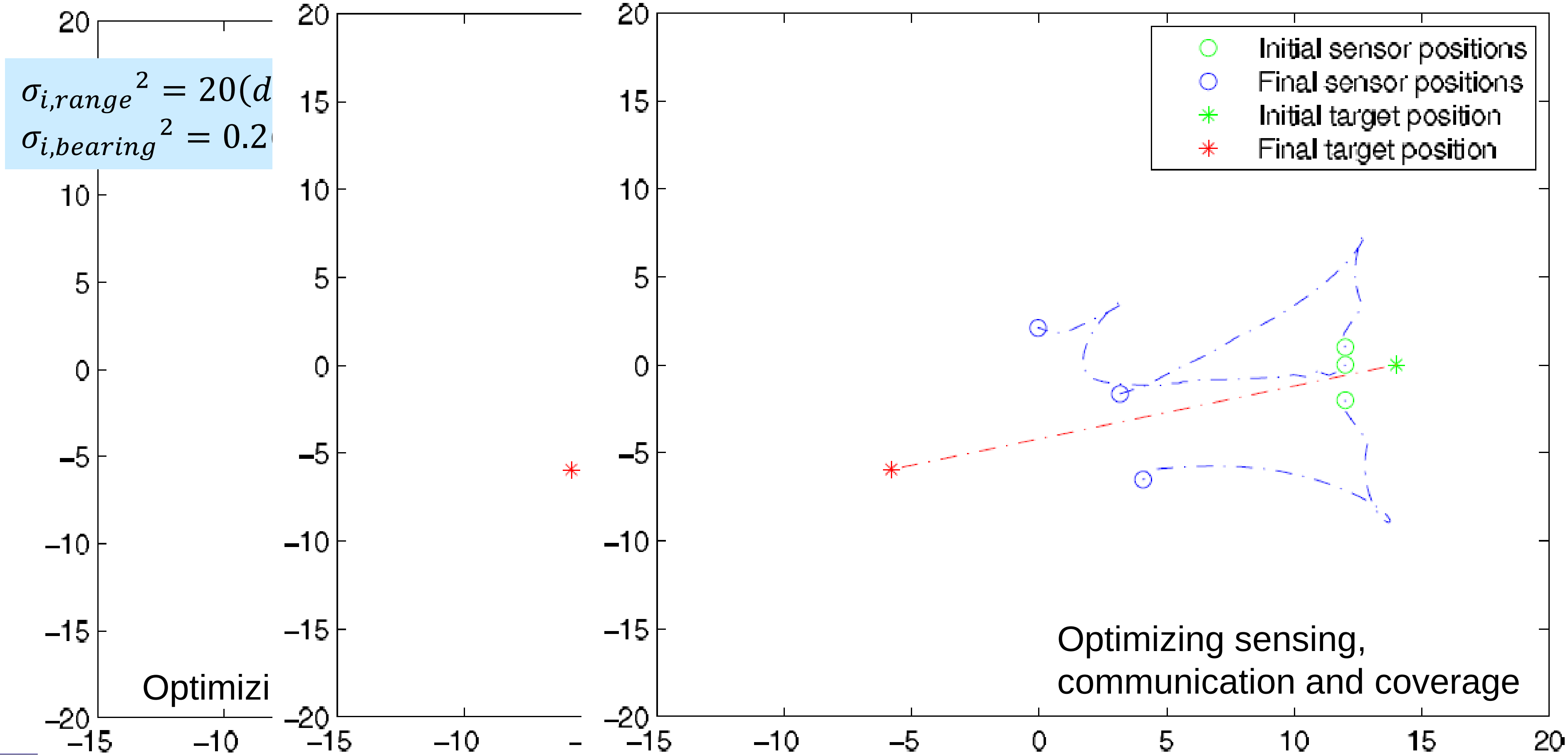
$$u_i = u_{Cov,i} + u_{Com,i} + u_{Sense,i}$$

$$G(d_{ij}, r_i, r_j) = \begin{cases} \varphi d_{ij}^{-2}, & r_i + r_j > d_{ij}, \\ 0, & \text{otherwise.} \end{cases}$$

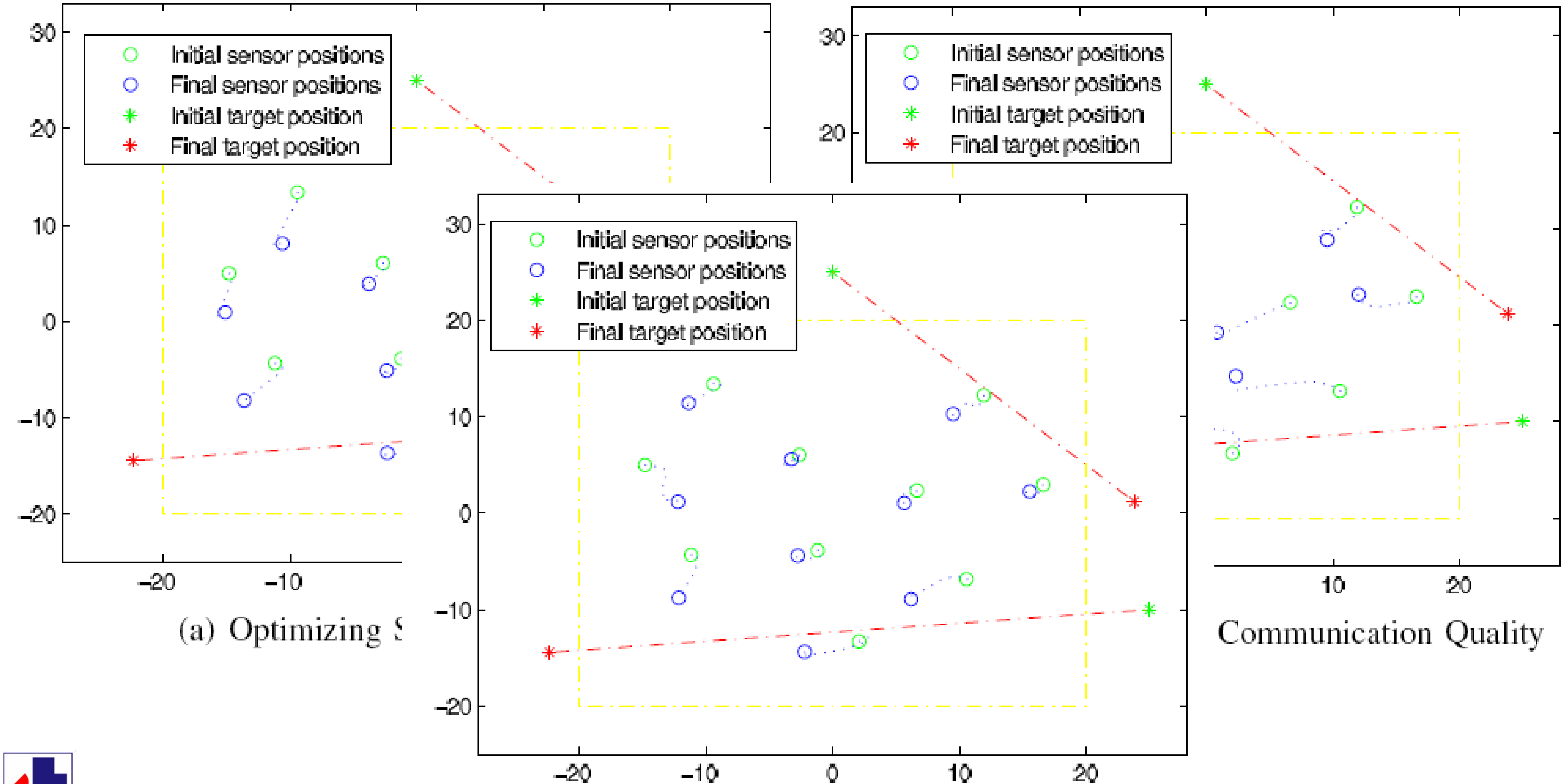
Weighting factor

U.PORTO FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Simulating single target tracking



Simulating multiple target tracking



(a) Optimizing S

(c) Optimizing Sensing, Communication and Coverage Quality

Simple experiment with J_{sense}

Formation Control Driven by Cooperative Object Tracking

Pedro U. Lima, Aamir Ahmad, Andre Dias, Andre G. S. Conceição, António Paulo Moreira, Eduardo Silva, Luis Almeida, Luis Oliveira, Tiago P. Nascimento

PTDC/EEA-CRO/100692/2008 – FCT national project

- P. Lima, A. Ahmad, A. Dias, A. G. S. Conceição, A.P. Moreira, E. Silva, L. Almeida, L. Oliveira, T. P. Nascimento. Formation Control Driven by Cooperative Object Tracking. Robotics and Autonomous Systems (ISSN: 0921-8890), 63(P1):68-79. Elsevier, Jan 2015.

Cooperative sensing

- **Maximize cooperative perception with formation control**
  - Maintain dynamic adjustable performance
  - Function of the quality of the target perception
- **Control module**
  - **distributed** non-linear **model predictive** controller
- **Estimator module**
  - fuses local estimates of the target state,
  - obtained by a **particle filter** at each robot
- **Team integration**
  - Supported on the RTDB

Control module

- Distributed non-linear model predictive controller
- Iterative approach involving
  - **Optimizer**
    - **Online numeric minimization** method
      - Resilient propagation → quick convergence
    - Generates **control signals**
  - **Predictor**
    - Predicts **state evolution** based on current state
    - N_p prediction horizon

Control module

- **DNMPC cost function**
  - For the nth robot (R_n)

$$J(\mathrm{N}_1,\mathrm{N}_2,\mathrm{N}_c) = \sum_{i=\mathrm{N}_1}^{\mathrm{N}_2} \lambda_a |{}^i\mathcal{C}^{\perp}|$$

$$+ \sum_{i=\mathrm{N}_1}^{\mathrm{N}_2} \lambda_0 |(D_{\mathrm{val}} - ||{}^iP_{\mathrm{t}}^{\mathrm{R}_n}||)| \quad + \sum_{i=\mathrm{N}_1}^{\mathrm{N}_2} \lambda_1 |\delta(\theta_n, {}^i\theta_{\mathrm{t}}^{\mathrm{R}_n})|$$

$$+ \sum_{i=\mathrm{N}_1}^{\mathrm{N}_2} \lambda_2 |P_{\mathrm{val}} + ({}^i\tilde{P}_{\mathrm{t}}^{\mathrm{R}_n} \cdot {}^i\tilde{V}_t)| \quad + \sum_{i=\mathrm{N}_1}^{\mathrm{N}_2} \sum_{j=1}^{\mathrm{N}} \lambda_3 \max(1 - \frac{||{}^iP_{\mathrm{R}_n}^{\mathrm{R}_j}||}{K_0}, 0)$$

$$+ \sum_{i=\mathrm{N}_1}^{\mathrm{N}_2} \sum_{k=0}^{\mathrm{N}_b} \lambda_4 \max(1 - \frac{||{}^iP_{\mathrm{R}_n}^{\mathrm{O}_k}||}{K_0}, 0) \quad + \sum_{i=1}^{\mathrm{N}_c} \lambda_5 |\Delta U^i|$$

- det. of the uncertainty matrix
- Keeps Rn at distance Dval from target
- Keeps Rn oriented towards the target
- Desired alignment Pval with target
- Variations in the control signal
- Keeps Rn away (K_0) from Rj and Ok

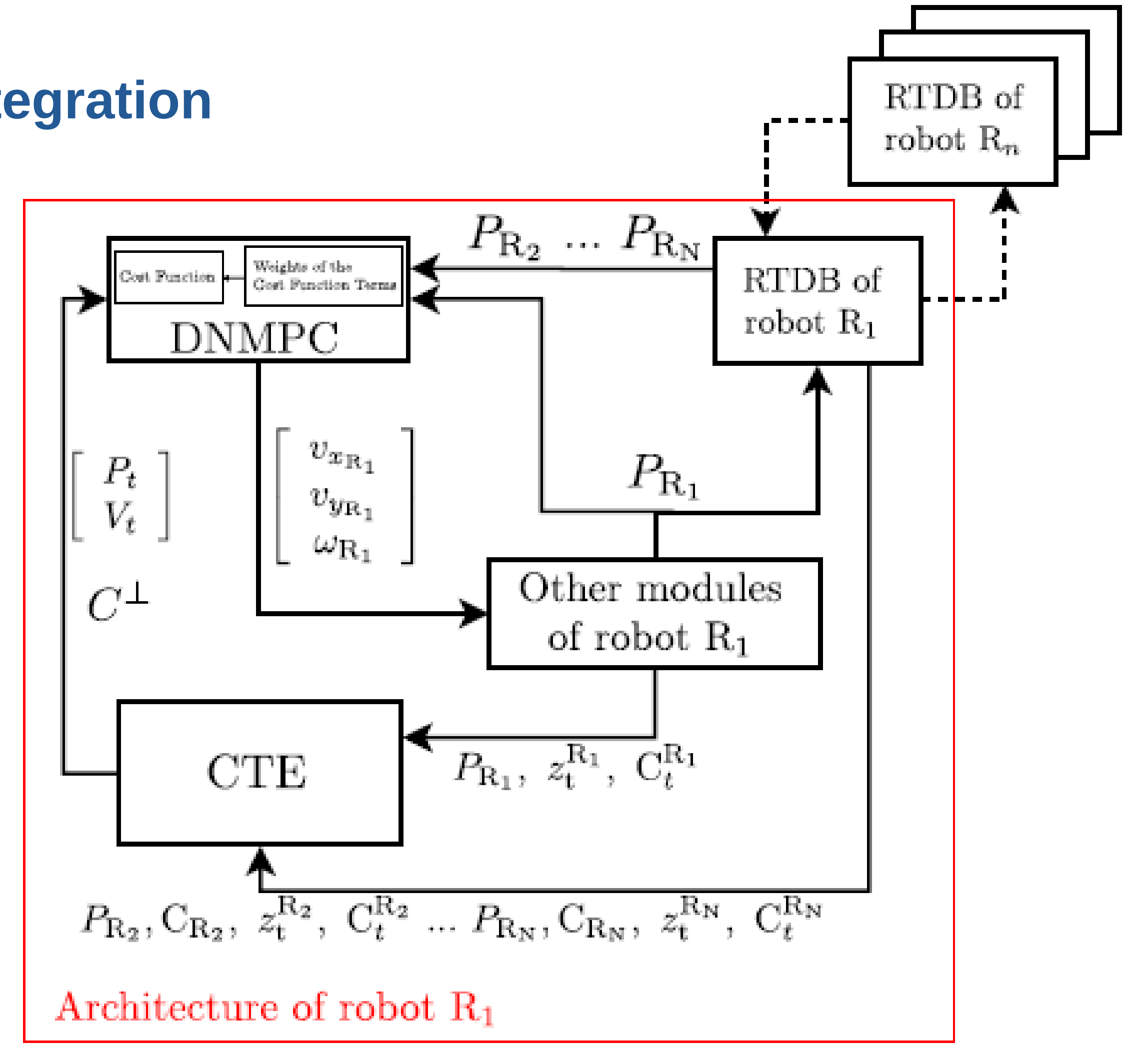
U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Estimator module

- Cooperative Target Estimator (CTE)
  - **Particle Filter** with enhanced fusion step
  - Exchanges:
    - **Target observation** measurement and respective **confidence**
    - **Self-localization confidence**
  - Builds own **observation management pool – OMP**
    - Matrix with observations from all team members

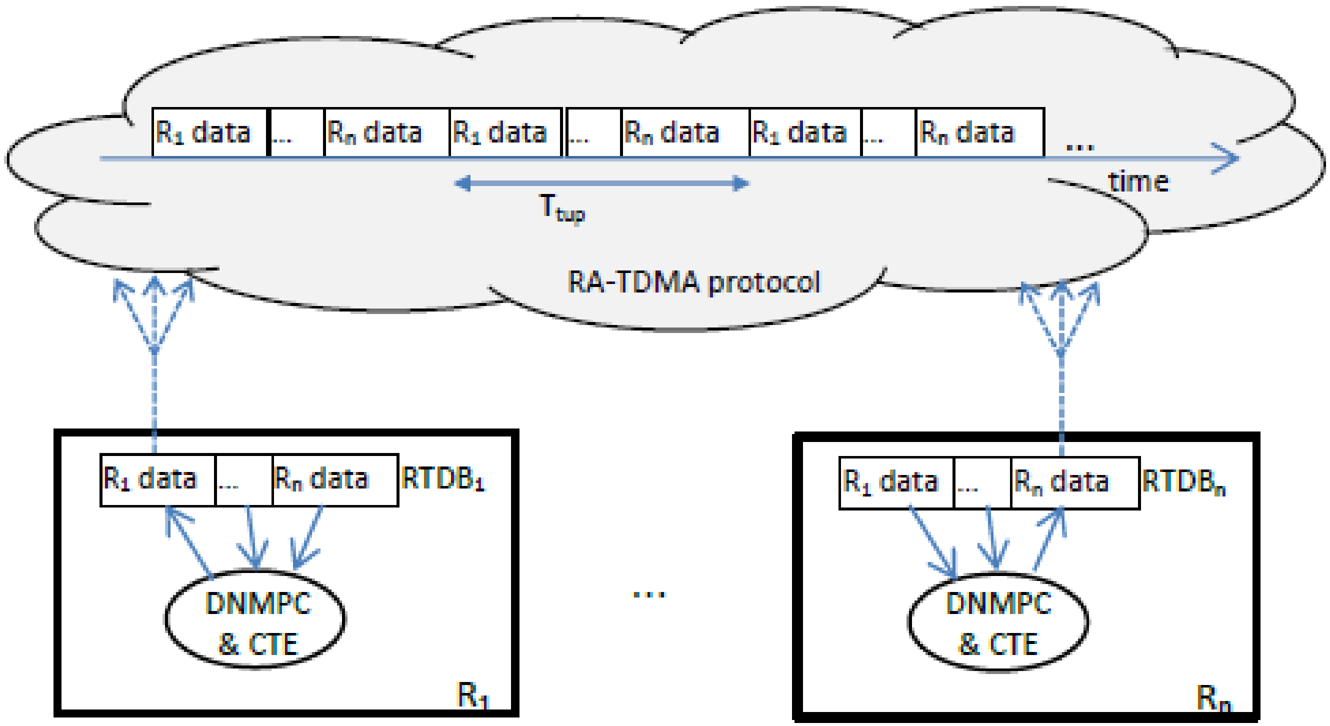
Modules integration

- **Functional integration**



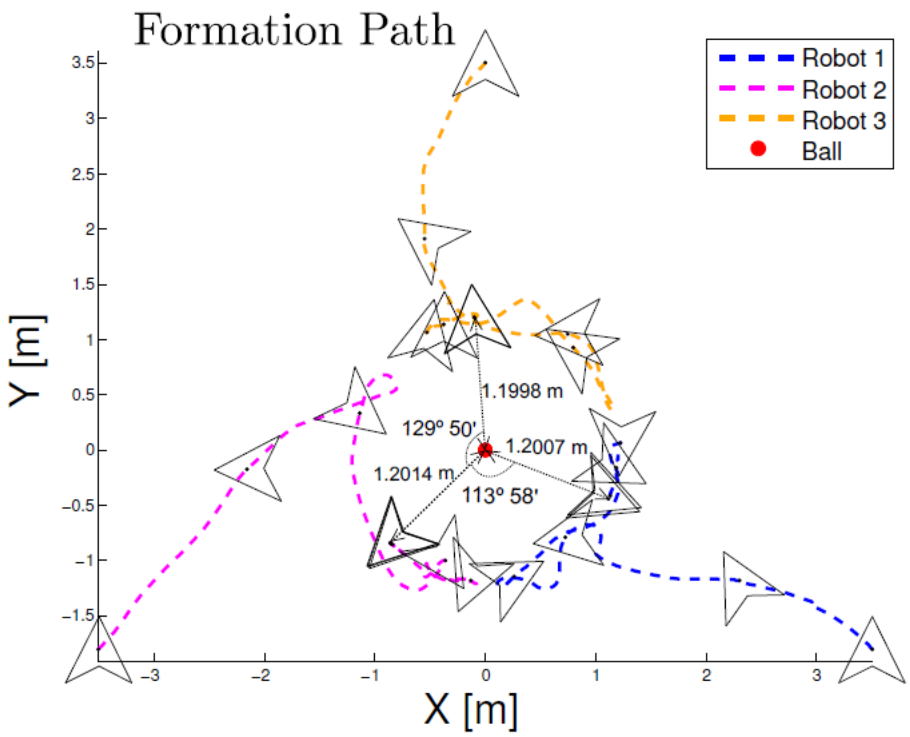
Modules integration

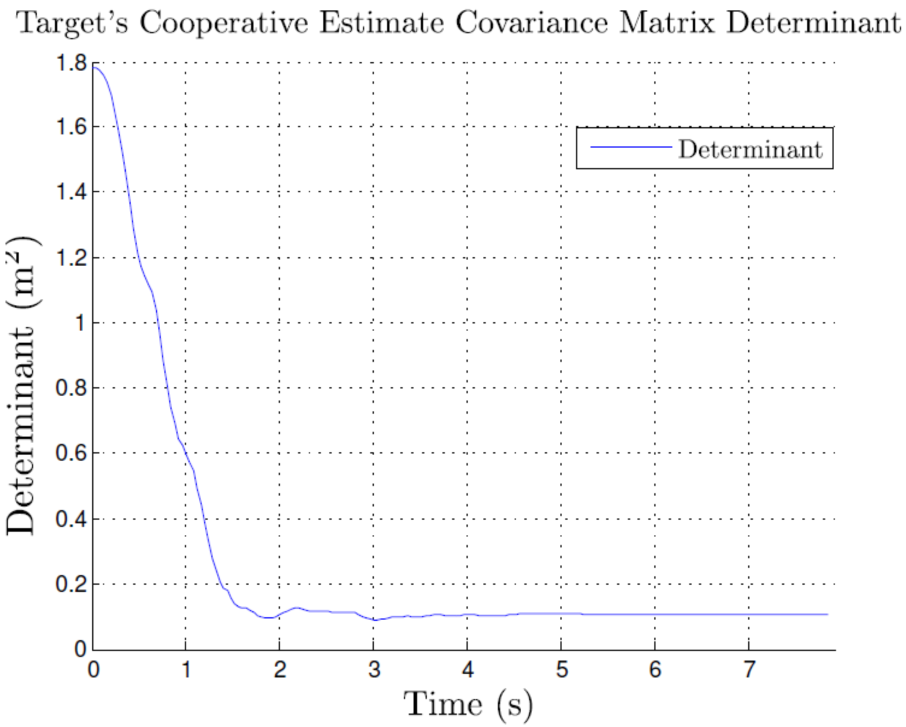
- ## Information integration



Results

- Experiments with 3 5DPO MSL RoboCup robots





Results

- Experiments with 3 5DPO MSL RoboCup robots

$$\mathcal{C}^{\perp}_{\text{5dpo}} = \begin{bmatrix} K_1 d^2 & 0 \\ 0 & K_2 d \end{bmatrix}$$

5dpo - 2 robots case	
Situation	$\lvert \mathcal{C}^{\perp}_{\mathbf{5dpo}} \rvert$
Only Target Covariance	0.2252
Only Mates	0.3145
All terms	0.2201

5dpo - 3 robots case	
Situation	$\lvert \mathcal{C}^{\perp}_{\mathbf{5dpo}} \rvert$
Only Target Covariance	0.1017
Only Mates	0.1095
All terms	0.1074

Impact of communications in multi-robot consensus

Bernardo Ordoñez, "Estratégia de controle cooperativo baseado em consenso para um grupo multi-veículos" (in Portuguese), PhD Thesis, Universidade Federal de Santa Catarina, Florianópolis, Brazil, May 2012

B. Ordoñez, U. F. Moreno, J. Cerqueira, L. Almeida. Generation of trajectories using predictive control for tracking consensus with sensing and connectivity constraint. In Cooperative Robots and Sensor Networks, Anis Koubaa, Abdelmajid Khelil (Eds.), Springer, Series on Studies in Computational Intelligence, Vol. 507:19-37. ISBN 978-3-642-39300-6, 2014. (DOI:10.1007/978-3-642-39301-3)

Objective

- ***"Design decentralized control laws that generate a consensus trajectory "***

- **Limited knowledge** and **limited information sharing**, mainly caused by **physical distribution**
- **Communication channels** with **transient faults**
- Information on the ***rendez-vous*** point available to **part of the group, only**

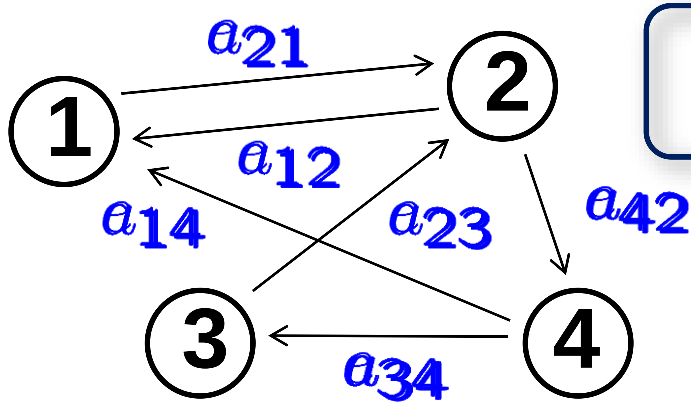
Strategy

- **Cooperation strategy** based on
  reference point consensus with predictive control
  - Predictive control → sliding window
  - Preserve connectivity even with transient faults
  - Extend the problem to cover the sensing angle and formation when arriving at the rendez-vous point
- **Use a sincronization protocol when exchanging messages**
  - Reducing packet losses due to collisions within the team
  - Access the impact of collisions in the consensus performance

Consensus theory

- **Cooperative control**
  - Achieve a common target by several autonomous agents
- **Information state**
  - Agents localization, speed and bearing
- **Consensus should be achieved in a decentralized fashion**
  - Interactions among neighbor agents is possible
- **Coordination of the agents motion**
  - Based on sharing their own states

Consensus theory



$$G_n = (\nu_n, \varepsilon_n)$$

- **Graph nodes:** $\nu_n = \{v_1, v_2, v_3, v_4\}$ Case of n=4 agents
- **Graph edges (links):** $\varepsilon_n \subseteq \nu_n \times \nu_n \mid e_{ij} = (v_i, v_j).$
- **Adjacency matrix:**

$$\begin{bmatrix} 0 & & 0 & \\ & 0 & & 0 \\ 0 & 0 & 0 & \\ 0 & & 0 & 0 \end{bmatrix}$$

Represents the network topology

Consensus theory

- **Approach**
  - Impose **similar dynamics** to the information states so that these converge to a **common value**

$$\dot{\xi}_i(t) = -\sum_{j=1}^{n} a_{ij} \left(\xi_i(t) - \xi_j(t)\right)$$

where a_{ij} is the corresponding element of the adjacency matrix and $\xi_{i,j} \in \mathbb{R}^m$ is the corresponding agent information state.

- **Consensus is achieved** if $\forall \xi_{i,j}(0)$ and $i, j = 1..n$

$$lim_{t\to\infty} \left|\xi_i(t) - \xi_j(t)\right| = 0$$

Consensus theory

- ## Approach
  - Impose **similar dynamics** to the information states so that these converge to a **common value** and **that value is the rendez-vous point**

$$\dot{\xi}_i(t) = -\sum_{j=1}^{n} a_{ij}\left(\xi_i(t) - \xi_j(t)\right) + a_{i(j+1)}\left(\xi_i(t) - \xi_r(t)\right)$$

where $\xi_r \in \mathbb{R}^m$ is the **desired state at the rendez-vous point**

  - **Consensus is achieved** if $\forall \xi_{i,j}(0)$ and $i, j = 1..n$

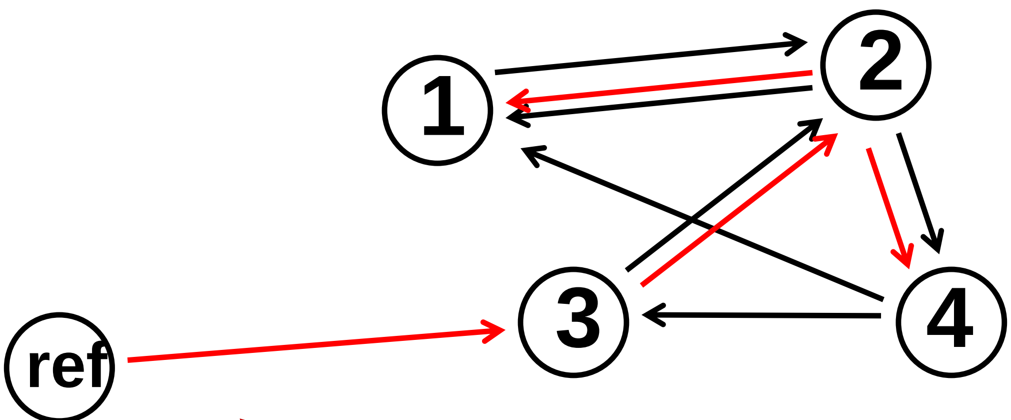
$$lim_{t\to\infty} |\xi_i(t) - \xi_r(t)| = 0$$

Reference works as another agent

Consensus theory

- **Adding the reference to the adjacency matrix**
  - Build a **spanning tree** $G_n(t)$ over the adjacency matrix considering the **reference as the root node** $G_{n+1}(t)$

$$G_{n+1} = (\nu_{n+1}, \varepsilon_{n+1})$$



$$\begin{bmatrix} 0 & a_{12} & 0 & a_{14} & 0 \\ a_{21} & 0 & a_{23} & 0 & 0 \\ 0 & 0 & 0 & a_{34} & a_{35} \\ 0 & a_{42} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

The root node just send information
(not influenced by the remainig nodes)

Consensus theory

- **Information state** with **1st order dynamics** $\dot{\xi}_i = u_i$

  where $u_i \in \mathbb{R}^m, i = 1..n$, is the control input.

- A **basic consensus protocol** can be achieved with
  - $\forall i = 1..n$

$$u_i(t) = -\sum_{j=1}^{n} a_{ij} \left(\xi_i(t) - \xi_j(t)\right)$$

- In a discrete form

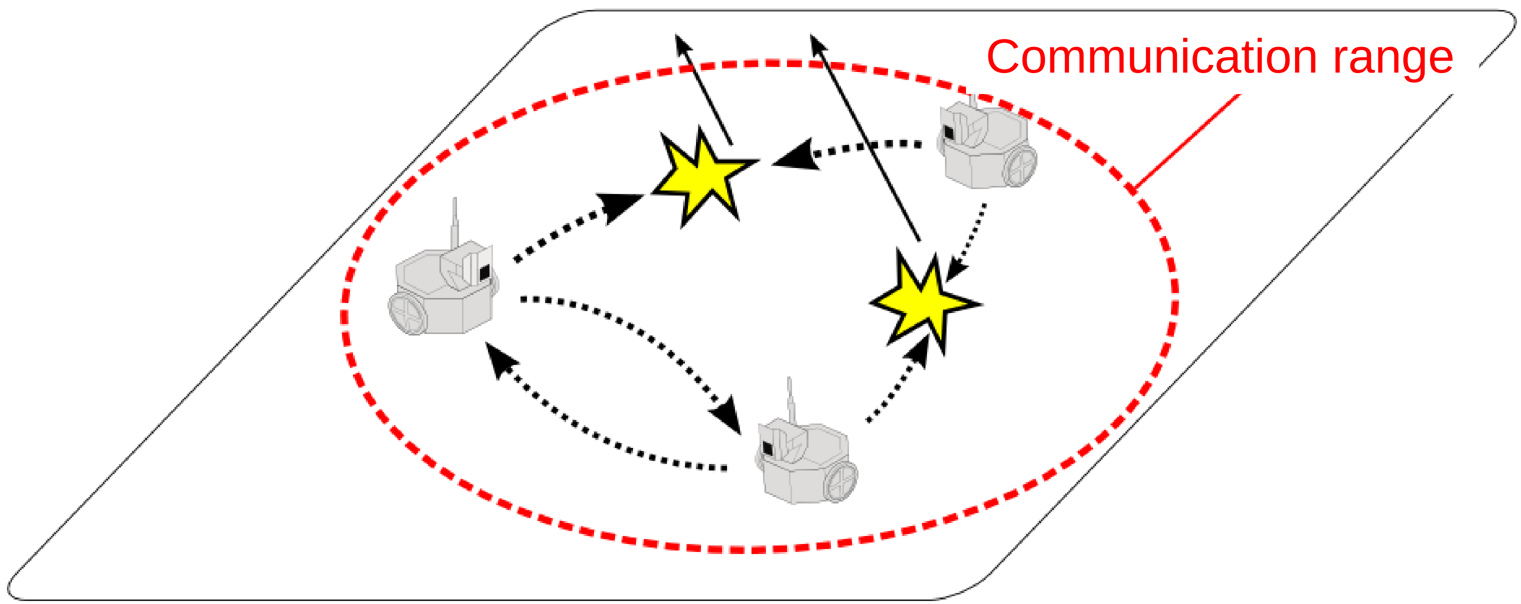
$$\xi_i[k+1] = \xi_i[k] + \Delta_k \; u_i[k]$$

  where k is the sampling index and Δk is the integration step

- The **information state** includes the **agents trajectories**

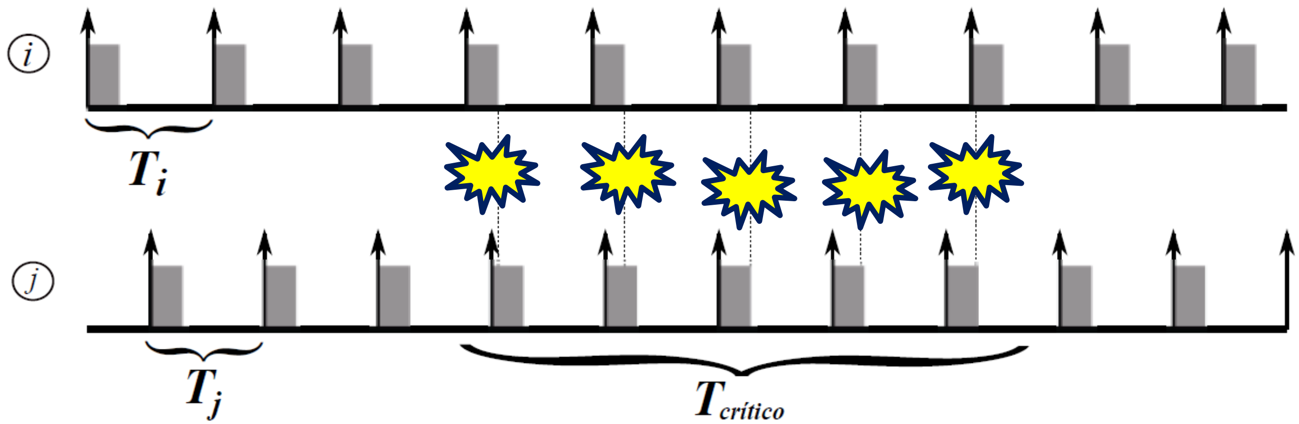
Impact of the communications

- **Wireless communications are prone to**
  - Errors from interferences and collisions
- **Collisions can occur due to lack of synchronization**



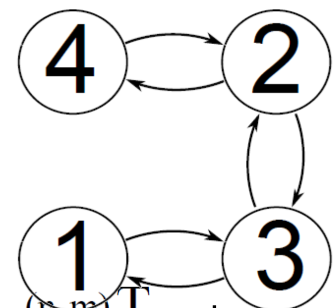
Impact of the communications

- All team agents typically **transmit their states periodically** and with **similar period**
  - **Without synchronization**, clock drifts in the different agents can lead to **phase rotation** of their transmissions leading to **periods of high collision probability → critical periods**
- **Critical periods**
  - strong degradation of the communication capabilities

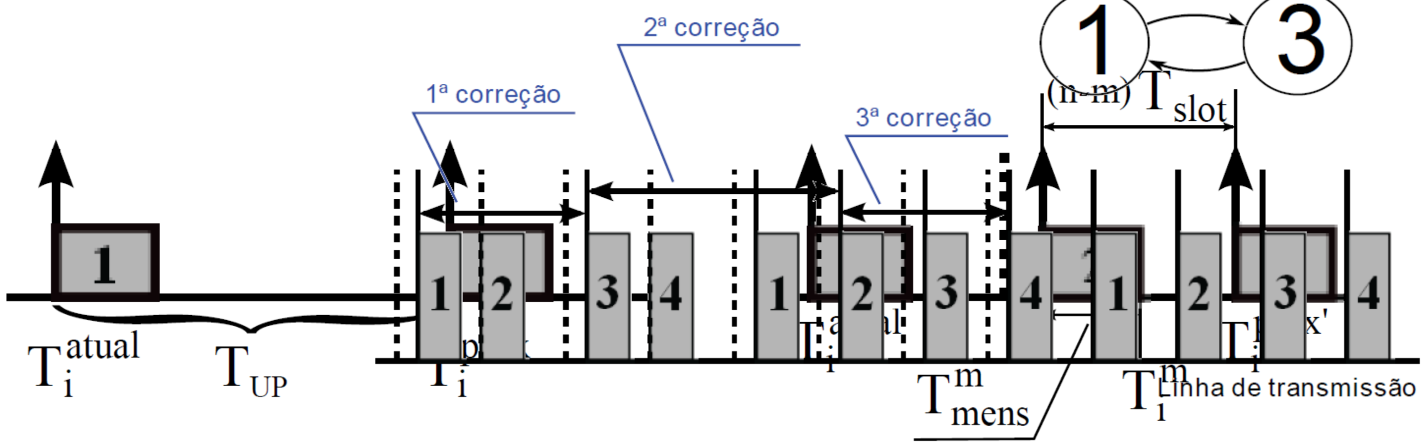


Synchronization approach

- **Synchronize** the agents transmissions so that they **transmit in a round**, in a **TDMA** fashion
  - Synchronization achieved by **listening** to the other transmissions in the team and **follow any delay** they suffer within a certain bound
  - Does **not need clock synchronization** and nodes maintain the **relative intervals** in between their transmissions



Reconfigurable and Adaptive TDMA



Modeling collisions

- **Probabiility of successful transmission depends on phase**
  - Modeled in the adjacency matrix with probability of connection
  - **Normal phase**, high probability (low error rate)

$$a_{ij}(p_i) : \Omega_i \rightarrow \Re \quad \forall\, a_{ij} \in \varepsilon \mid \Omega_i = [80\%, 95\%]$$

  - **Critical period**, low probability (high error rate)

$$a_{ij}(p_{crit}) : \Omega_{crit} \rightarrow \Re \quad \forall\, a_{ij} \in \varepsilon \mid \Omega_{crit} = 30\%$$

  - **Definition of the critical period**

$$\left\| T_i(t) - T_j(t) \right\| \leq T^m_{mens}$$

where T^m_{mens} is the message trnasmission duration

Simulation results

- **5 agents** with ***slightly* different period**

$$T_{up}^{i} = [50,1; 50; 50,1; 49,9; 49,9] \ ms$$

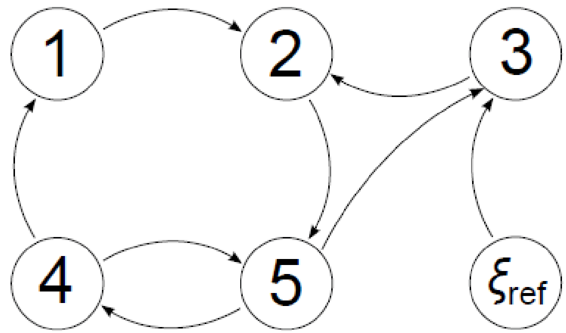
- **Objectives**
  - Observe **impact** on the **state information propagation**
  - Observe **relationship** between **network connections density** and the **speed of convergence** to consensus

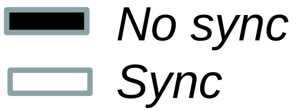
- **Criterion for convergence**

$$\|\xi_i^{(x,y)} - \xi_j^{(x,y)}\|_2 \leq \psi \qquad \psi = 0,1.$$

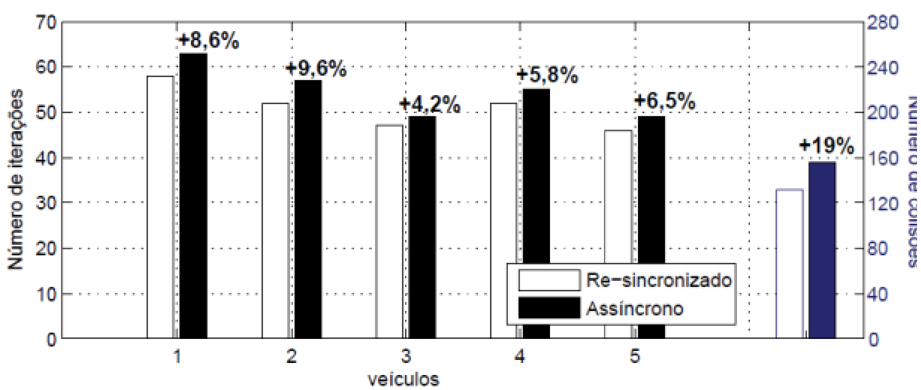
Siimulation results

- **1000 simulation runs**
- **Network topology**

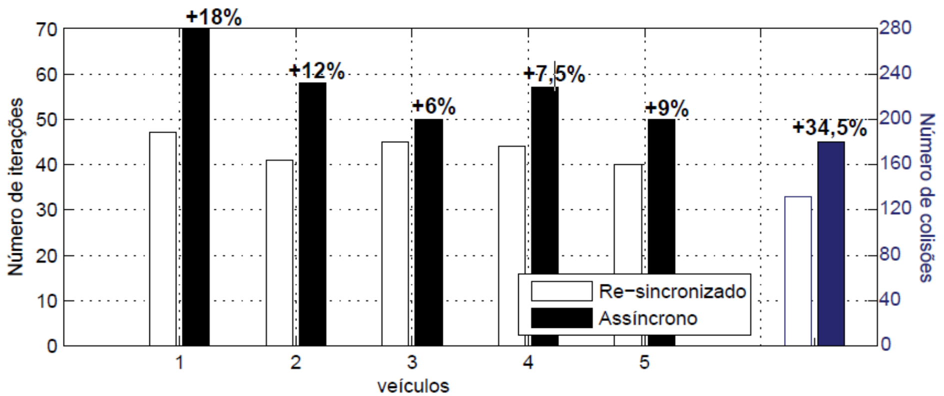




T^{m}_{mens} = 1ms



T^{m}_{mens} = 2ms



Number of steps to achieve consensus

RT-WMP
Real-time – Wireless Multi-hop Protocol

(currently offered as a ROS package)

D. Tardioli, '"Real-Time Communication in Wireless ad-hoc networks. The RT-WMP protocol"', PhD Thesis, Universidad de Zaragoza, October, 2010

RT-WMP basics

- **Based on the RT-EP**
  - Real-time communication for shared Ethernet
- **Works over the IEEE 802.11 protocol**
- **Token Passing**
  - For priority agreement and topology tracking
- **Works in three time-bounded phases**
  - Priority Arbitration (PAP)
  - Authorization Transmission (ATP)
  - Message Transmission (MTP)
- **Routing is based on Link Quality amongst nodes**

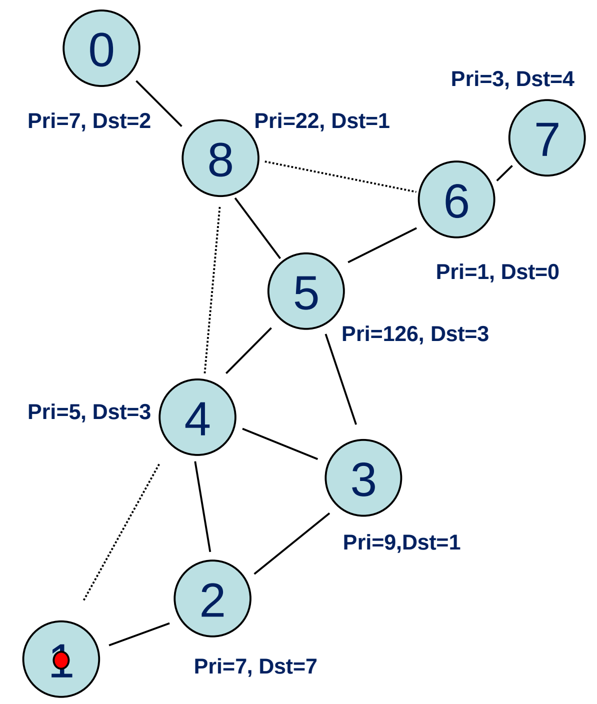
Protocol Definition

The Priority Arbitration Phase

- Token reaches all of the nodes (2n-3 hops in the WC)
- priority message in the ...eue is selected
- based on the Link
- ...QM contains the Link ... amongst nodes

Token

Res	TYPE	FRM	TO	MP	MPI	LA	nstat	LQM
Res	1	8	0	126	5	0		LQM



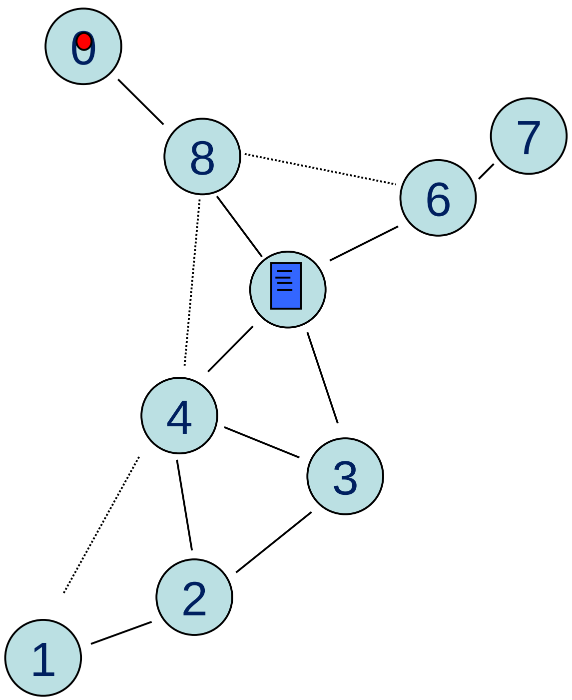
	p0	p1	p2	p3	p4	p5	p6	p7	p8
p0	-	0	0	0	0	0	0	0	70
p1	0	-	80	0	74	0	0	0	0
p2	0	80	-	68	72	-	0	0	0
p3	0	0	68	-	88	56	0	0	0
p4	0	74	72	88	-	56	0	0	0
p5	0	0	0	56	56	-	89	0	73
p6	0	0	0	0	0	89	-	90	0
p7	0	0	0	0	0	0	90	-	0
p8	70	0	0	0	0	76	0	0	-

The Link Quality Matrix

Protocol Definition

AUT

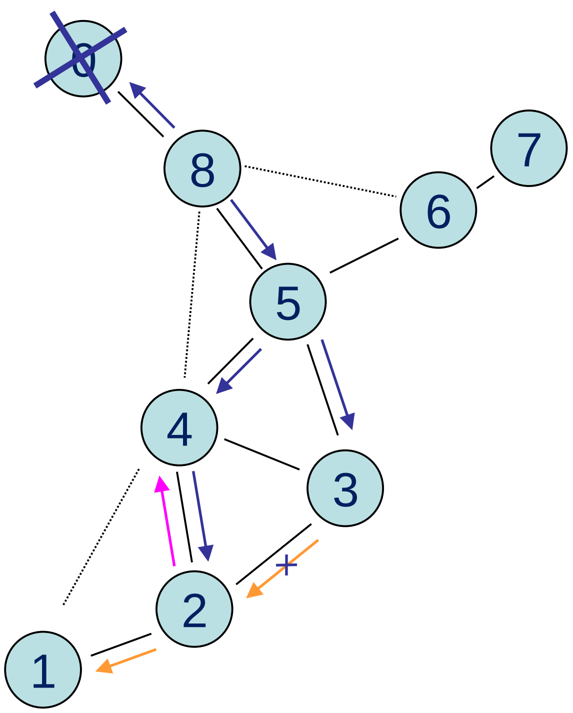
Res	TYPE	FRM	TO	SRC	DST
Res	3	8	5	0	5



- ## Authorization transmission phase
  - Last node calculate the best path to the authorized node
    - Dijkstra algorithm over a matrix derived by the LQM
      (n-1 hops in the WC)
  - Sends it the authorization
- ## Message transmission phase
  - Authorized node calculate the best path to the destination node
    - Dijkstra algorithm over a matrix derived by the LQM
      (n-1 hops in the WC)
  - Sends it the message

Error Recovery Mechanisms

- **Implicit acknowledge**
- **Two causes of error**
  - Node falls/disappears
    - Timeout
    - Does not jeopardize real time behavior
  - Communication error
    - Token Duplication
    - Serial field

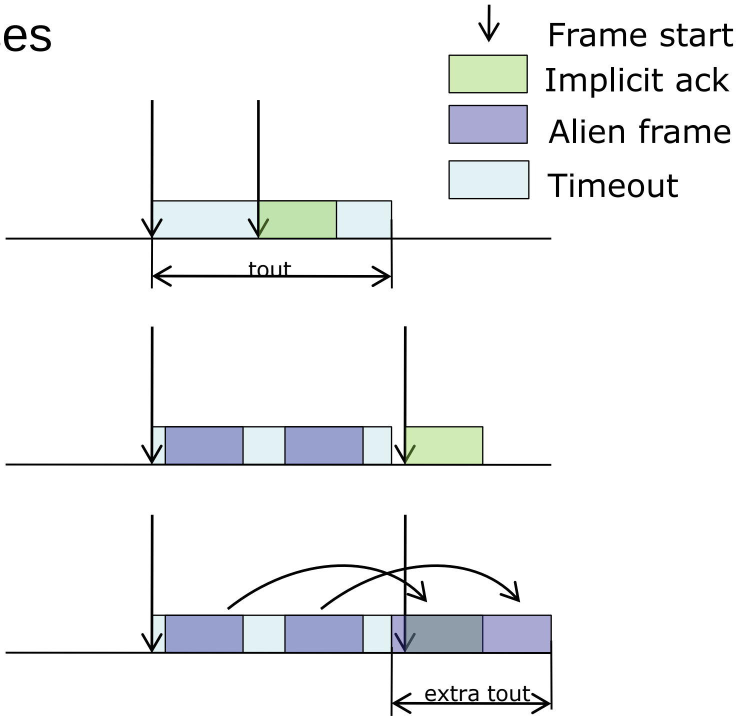


Implementation details

- **Implemented under**
  - Linux Operating System (2.6)
    - User space and Kernel space
    - Needs modified NIC Driver
  - MaRTE OS
- **ANSI C code**
- **One TX and one RX queues**
  - Multiple queues in current implementation
- **Works over 802.11a/b/g devices**

RT-WMP – tolerating alien traffic

- **All 3 phases are bounded**
- **Includes mechanisms based on timeouts**
  - These are **not compatible with alien traffic**!
  - Thus, alien traffic causes high packet losses
- **Mechanism**
  - **Extend all timeouts** by the time taken by alien frames
  - Increases latency but avoids extra packet losses
- Provides the **desired graceful degradation**



RT-WMP – routing on demand

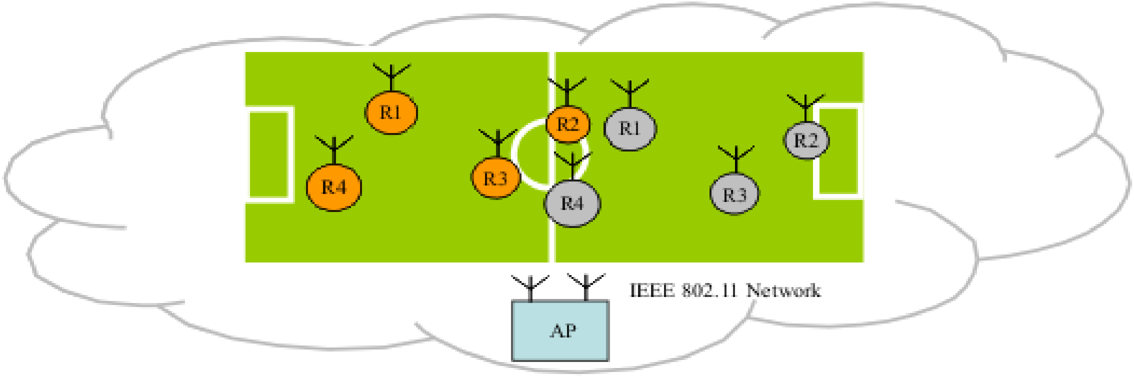
Quick look into RoboCup Middle-Size League
A communications perspective

Motivation

- **Wireless** communication is **fundamental** for **multi-robot systems**
- However wireless medium is **open** and may suffer of **strong interference**
- Moreover **adequate use** of the network requires **specialized knowledge**
- **RoboCup MSL has always been suffering from wireless related problems**

MSL Rules

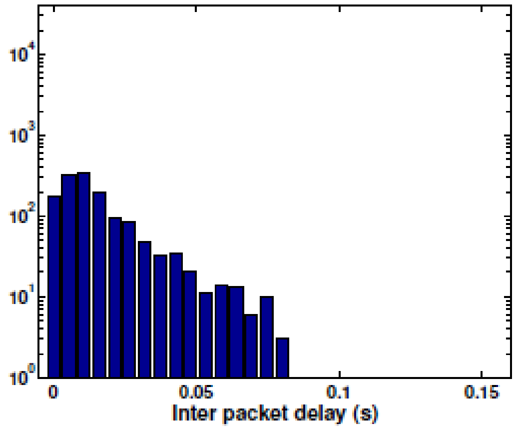
- IEEE802.11a/b technology
- **Infrastructure mode** (through Access Point)
- **Single a + single b** channels shared by both teams
- **Only unicasts/multicasts** (broadcasts are forbidden)
- **IPv4** addressing within predefined networks
- **Teams cannot use more than 2.2Mbit/s**
  - Removed recently



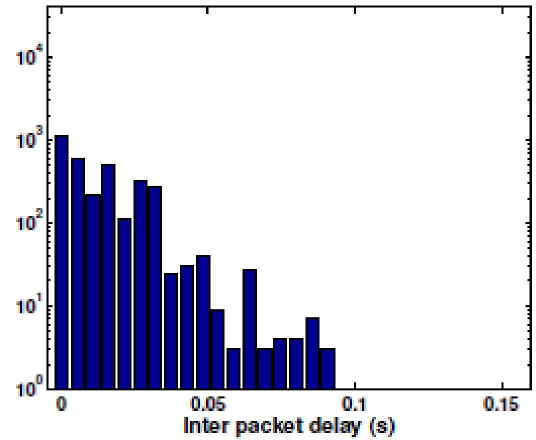
Logs from RoboCup 2008 – Suzhou, China

- **Log station:**
  - Laptop with built in wireless network card in monitor mode
  - IEEE802.11a
  - *Wireshark* software
- **Random games from the 3th round-robin**
  - 6 teams monitored
- **Logs duration ≈1minute**
  - Inter packet delays from the same team
  - Packet size

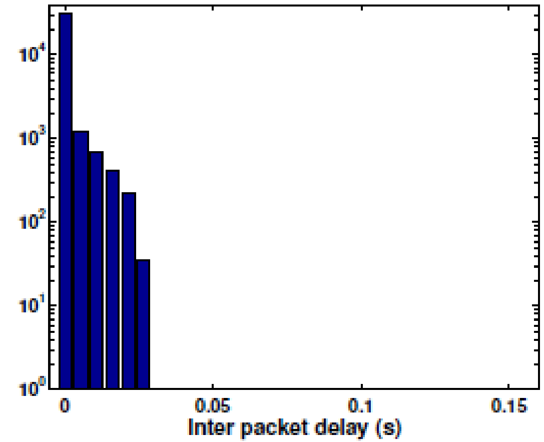
Inter-packet delays



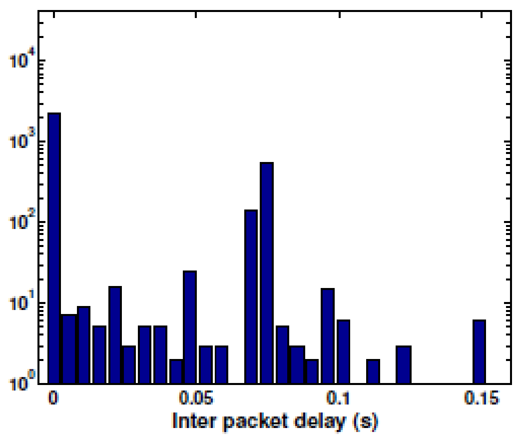
a) Team 1



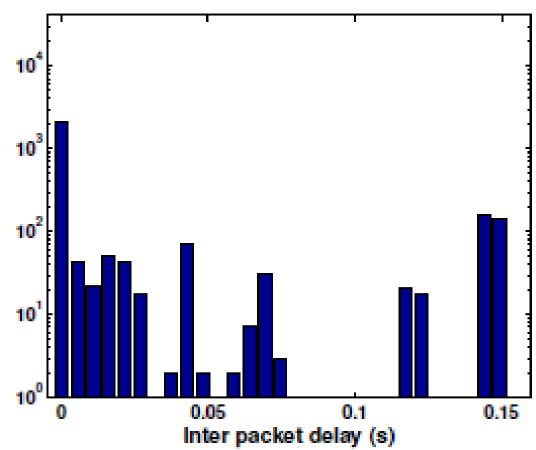
b) Team 2



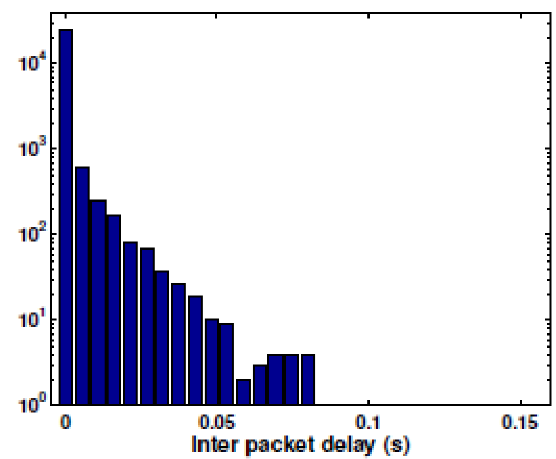
e) Team 5



c) Team 3

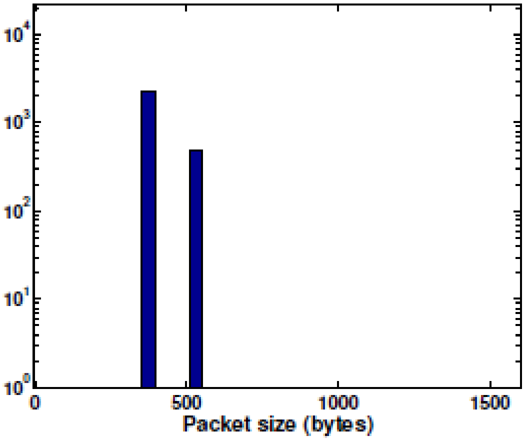


d) Team 4

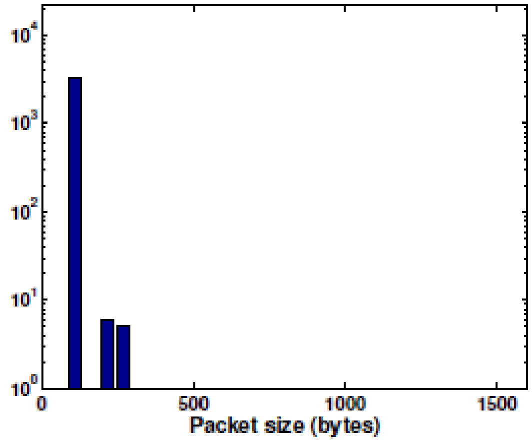


f) Team 6

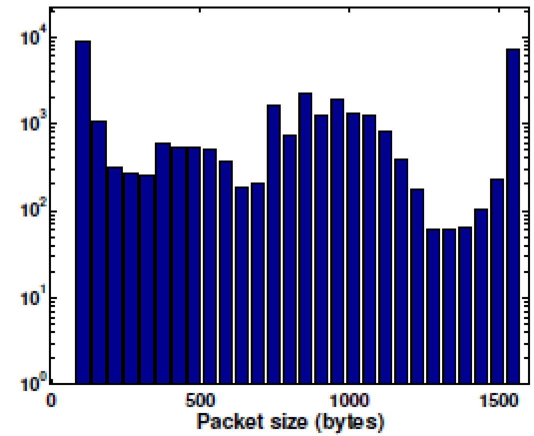
Packet sizes



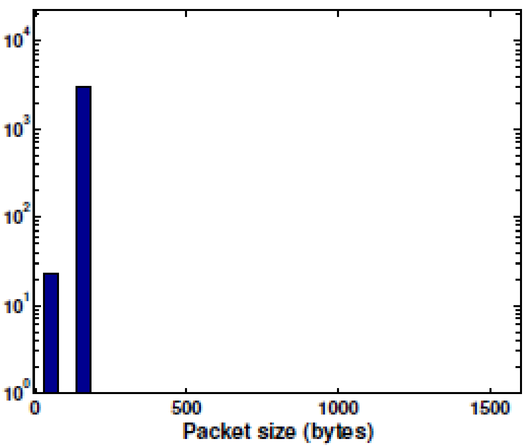
a) Team 1



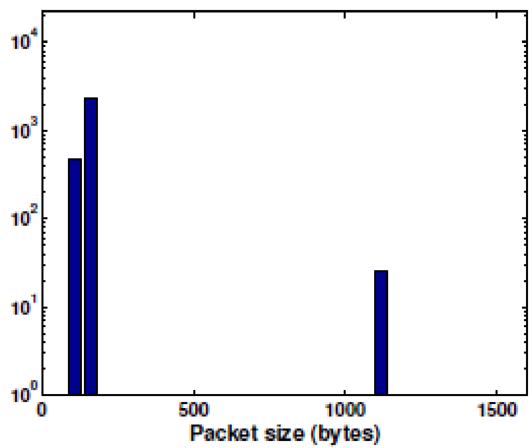
b) Team 2



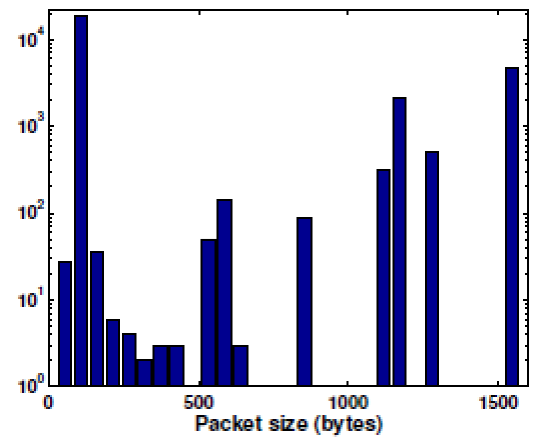
e) Team 5



c) Team 3



d) Team 4



f) Team 6

Summary of measurements

		Team 1	Team 2	Team 3	Team 4	Team 5	Team 6
Inter Packet (ms)	*avr*	17.74	15.20	20.03	21.72	1.74	1.90
	std	17.63	14.65	33.23	48.16	3.62	4.44
Packet Size (Bytes)	*avr*	412.87	139.68	160.51	187.67	787.40	497.81
	std	73.66	8.03	5.59	93.77	549.09	598.36
Burst Size (# 1.5kB pk)		–	–	–	–	6	12
Total kBytes		1158	460	480	517	26154	13072
% of max		4.43	1.75	1.84	1.98	100.00	49.98
Bandwidth utilization	802.11a	1.1%	0.4%	0.5%	0.6%	25%	13%
	802.11b	5.5%	2.0%	2.5%	3.0%	125%	65%

Summary of measurements

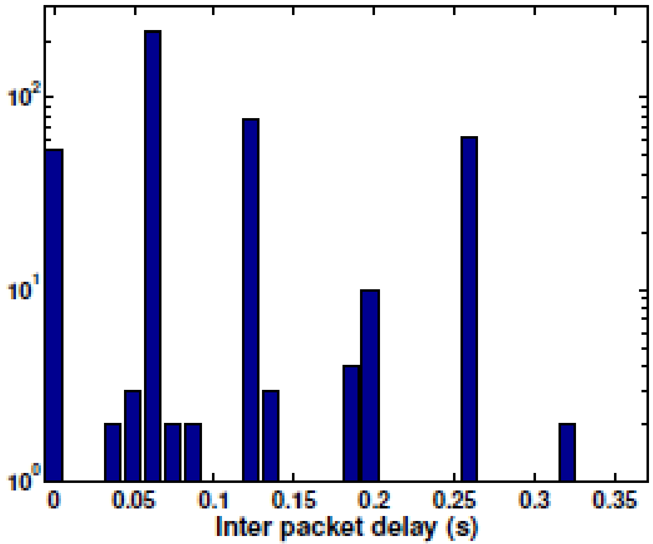
- **Wide** variability of **packet sizes**
- Some **long bursts** were observed in some teams
- **Large use** of the **bandwidth**
  - That would strongly overload the 802.11b mode
- **Very short inter-packet** intervals
- **Two of the observed teams would not comply with the rule of limiting bandwidth**
- **Limiting bandwidth is not enough**
  - Beyond bandwidth is it important to restrict **consecutive channel use**

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

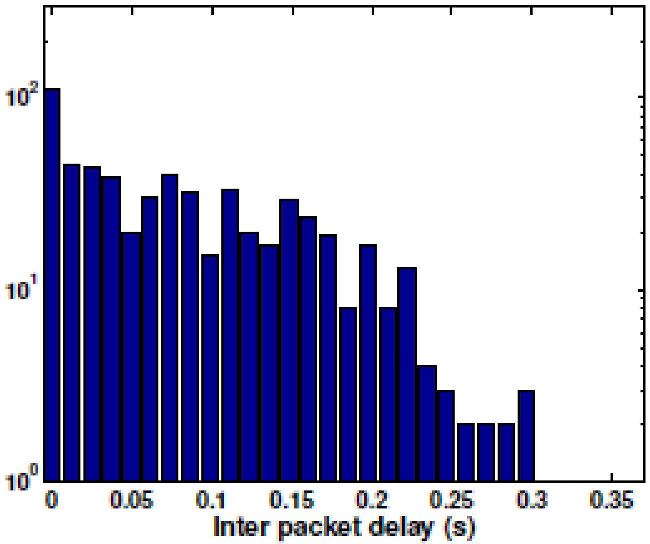
Problems

- **Infrastructure configuration**
  - Regular wireless Internet access network in the venue
- **Team communications configuration**
  - Teams using own AP or connections in Ad-Hoc mode
  - Bursts or non-IP traffic (sometimes, even malformed frames)
- **Lack of policing**
  - No one verifies the correct application of rules
- **Channel overuse by teams**
  - High bandwidth utilization means:
    - Large packet transmission delays
    - Increase of packet losses due to collisions and channel saturation

Impact of different communications patterns



a) Robot 1 of team 2 against team 1



b) Robot 1 of team 2 against team 6

Misconceptions

- ***No need for restricting teams transmissions***
  - But bandwidth is limited !
- ***Larger bandwidth solves the problem***
  - Only for a while, since teams will then transmit more
- ***Use technology with QoS support***
  - Which team would you give higher priority?
- ***No need for technical verifications***
  - Non-compliance will only be detected in the games !

Best practices for the teams

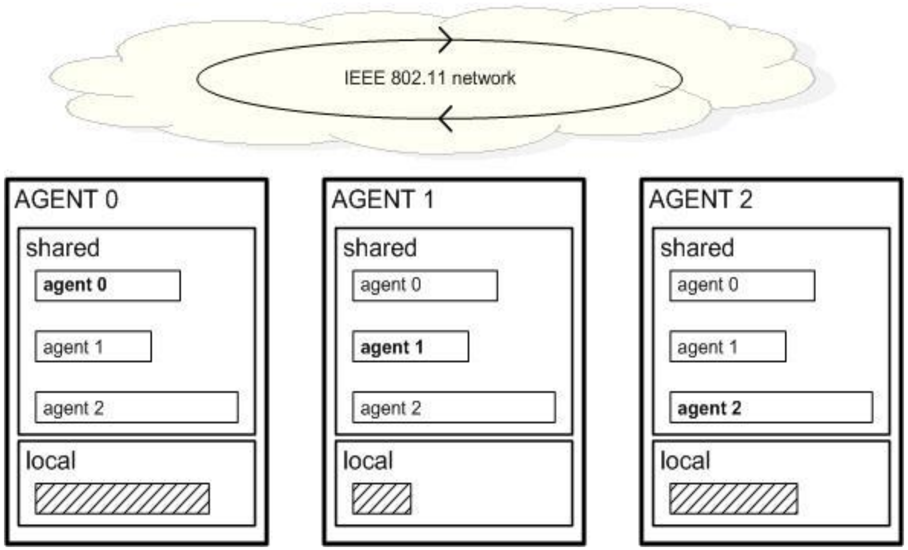
- **Low bandwidth** cooperation approaches that can work with the exchange of **small amounts of data**
- Use **periodic transmission** pattern in general, with small randomness
- Verify the **compliance of wireless** communications with the **rules before** the actual **competitions**
- **Do not use APs** that are **not from the organization**
- **Do not transmit wireless** traffic during competitions while in the neighborhood of the fields

Best practices for the organization

- Adequate **planning of APs** and **channels**
- **Switch off** any **pre-installed WLAN** for general Internet access in the venue
- Enforce **technical verifications** of the wireless communications
- **Traffic policing** using a network monitor
- Use a specific **network analyzer**, capable of providing information on the physical channel status

Proposed middleware solution

- **Real-Time Database that fulfills the best practices**
  - Follows **distributed shared memory** model
  - Each robot has a **local copy** of all relevant data in the team
  - Copies of **remote data** are **updated automatically** in the background
  - **Transparent** data location
  - **Fair communication** protocol (limited bandwidth and burstiness)



Proposed communication protocol

- **The load in wireless network cannot be totally controlled → we can control the team traffic, only**
- **Use Time Division Multiple Access**
  - Transmissions using **multicast** (no retransmissions)
  - Robots transmit in **dedicated slots** in a round
    - Team transmissions separated as much as possible
  - Round period: T_{tup} **– team update period** → sets the protocol reactivity according to real-time constrains
  - **Highly permeable** to other traffic → **fair protocol** ☺

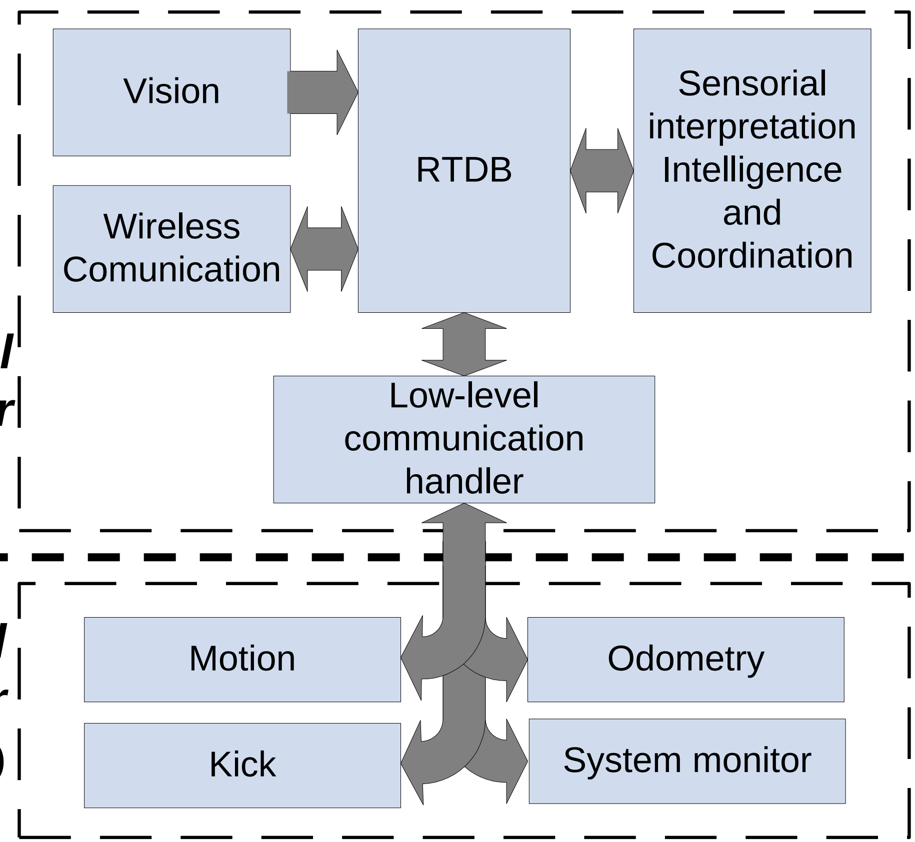
Reconfigurable and Adaptive TDMA protocol

The RoboCup Middle-Size League
CAMBADA robotic soccer team

CAMBADA
A RoboCup MSL soccer team

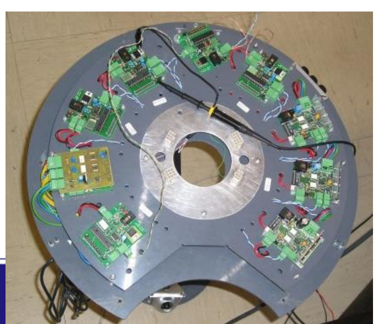
http://www.ieeta.pt/atri/cambada/

Robots functional architecture



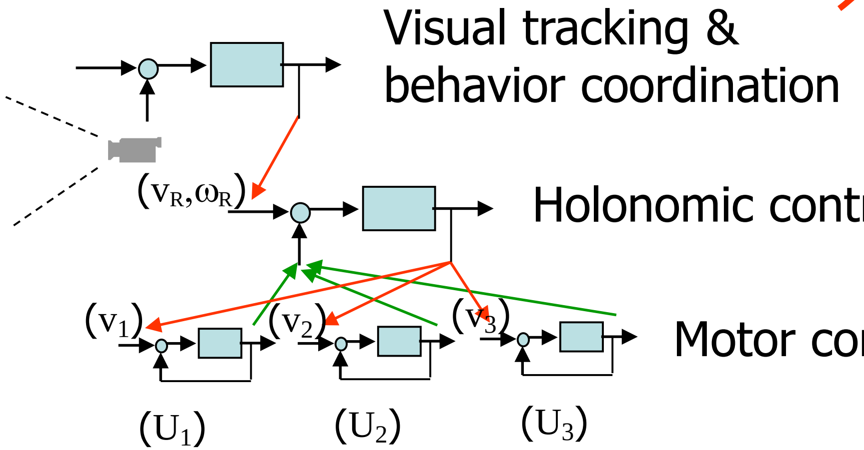
Higher-level Coordination layer

Lower-level sensing&actuation layer
(**CAN**-based distrib. system)

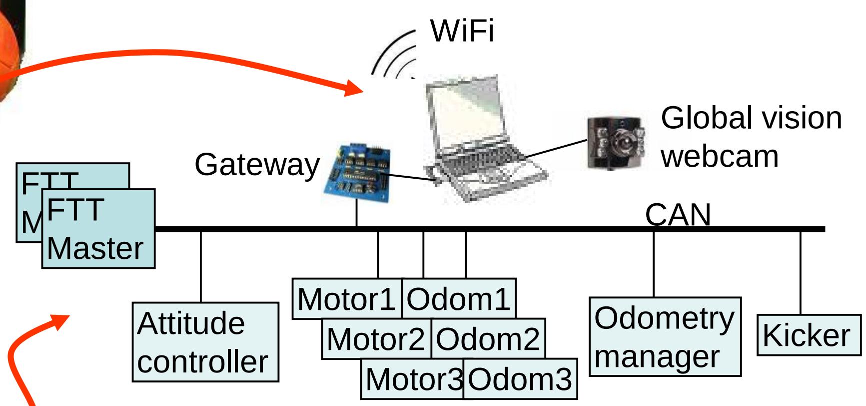


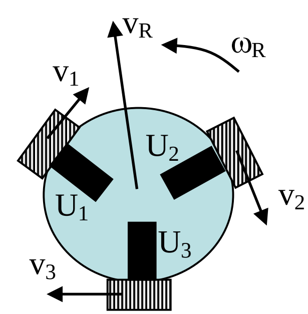
Robots architecture

Control architecture



Hardware architecture





J. L. Azevedo, B. Cunha, L. Almeida. Hierarchical Distributed Architectures for Autonomous Mobile Robots: a Case Study. ETFA 2007. Patras, Greece, September 2007.

Sharing and integrating robots information

- **Use of the channel also influenced by how the data is exchanged**
  - Client-server, Publisher-Subscriber, Blackboard…
- We use a **shared memory model** with a **proxy** in each team member
  - Just **one channel transaction** per round per team member
  - Homogeneous computing architectures
  - Each data item has **age information**
  - **Remote** data used **as local**

The Real-Time DataBase concept (RTDB)

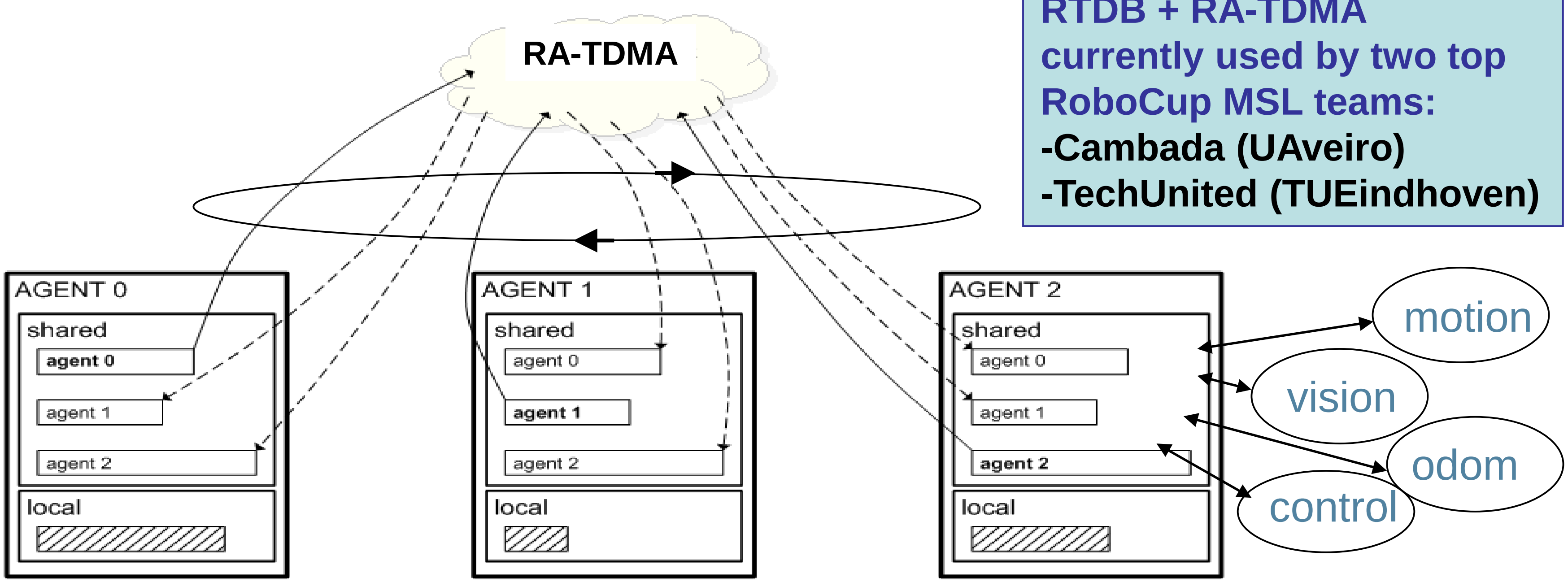
L. Almeida, F. Santos, L. Oliveira. Structuring Communications for Mobile Cyber-Physical Systems. In Management of Cyber Physical Objects in the Future Internet of Things: Methods, Architectures and Applications. A. Guerrieri, V. Loscri, A. Rovella, G. Fortino (Eds), Springer, Series on Internet of Things, Vol. 1: ISBN 978-3-319-26867-5, 2016. (DOI: 10.1007/978-3-319-26869-9)

The RTDB middleware

Source code available at:
www.bitbucket.org/fredericosantos/rtdb/

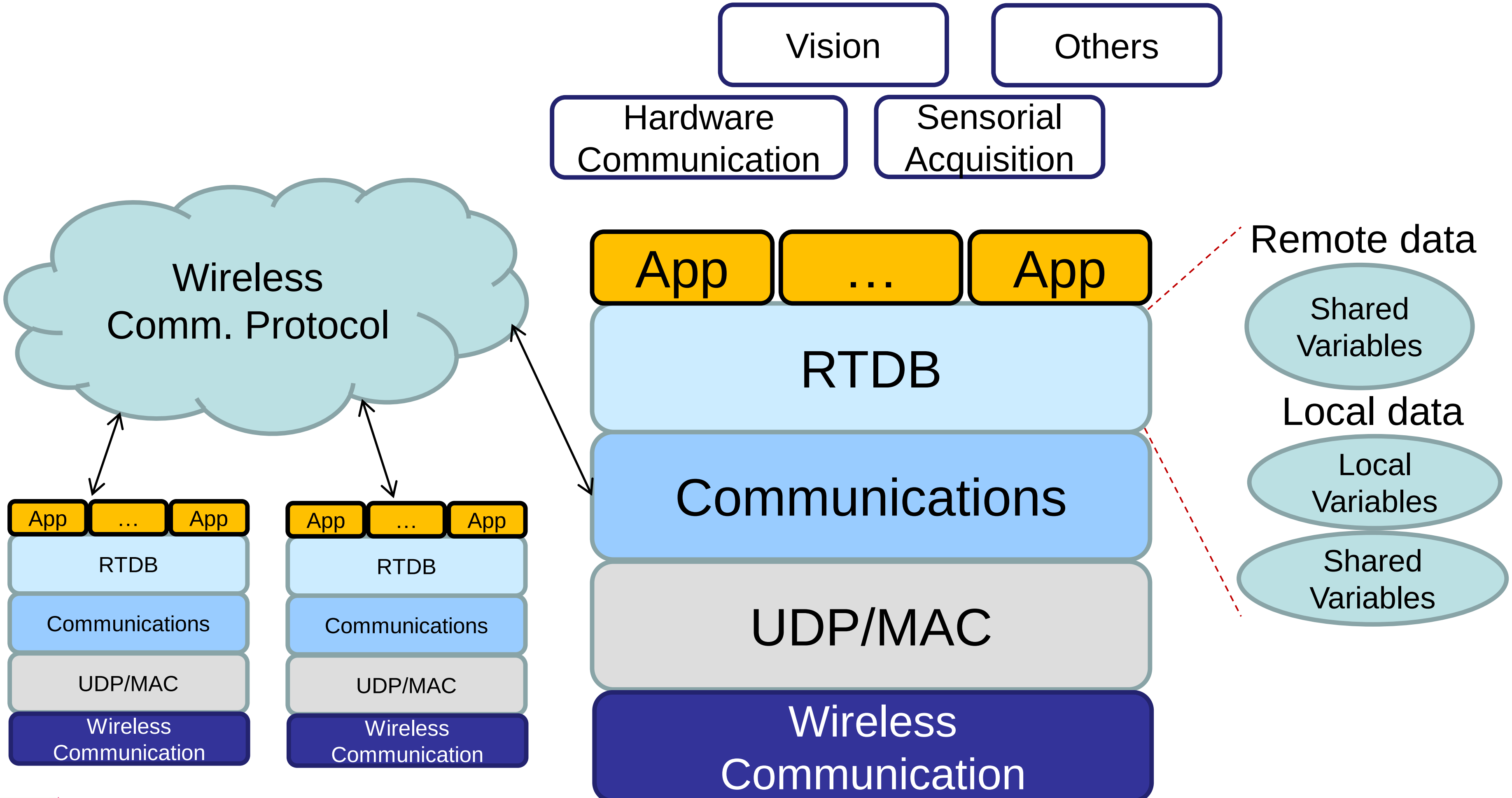
- **Nodes share data** with a **Real-Time DataBase** that holds
  - **Local sensor/state** data gathered from local processes
  - **Images of remote data** updated transparently with RA-TDMA

RTDB + RA-TDMA currently used by two top RoboCup MSL teams:
-Cambada (UAveiro)
-TechUnited (TUEindhoven)



F. Santos, L. Almeida, P. Pedreiras, L. S. Lopes. A real-time distributed software infrastructure for cooperating mobile autonomous robots. ICAR 2009. Munich, Germany, 24-26 June 2009.

Middleware / communications stack



RTDB configuration

- **Meta-model**
  - To generate a **configuration file** with the team data model
    - Currently this file is static and saved in all team members

```
AGENTS = id_ag [, id_ag , ...] [;]

ITEM id_it { datatype = type; [headerfile = <<filename>>]; [period = <<number>>]; }

SCHEMA id_sc { [shared = <<id_it>> [ , <<id_it>>, ...] ; ]
               [local = <<id_it>> [, <<id_it>>, ...] ; ]

ASSIGNMENT { schema = <<id_sc>>; agents = <<id_ag>>, ... ; }
```

RTDB configuration

```
AGENTS = robot1 , robot2 , base ;

ITEM image {datatype = struct image ; headerfile = image.h ; }
ITEM position { datatype = struct pos ; headerfile = pos.h ; period = 1 ; }
ITEM obstacles { datatype = struct obstacles ; headerfile = obstacles.h ;
                 period = 1 ; }
ITEM fuse_data {datatype = struct fuse ; headerfile = fuse.h ; period = 1; }

SCHEMA robot { shared = position, obstacles ;
               local = image ; }
SCHEMA base_st { shared = fuse_data ; }

ASSIGNMENT { schema = robot ; agents = robot1 , robot2 ; }
ASSIGNMENT { schema = base_st ; agents = base ; }
```

RTDB Applications Programming Interface

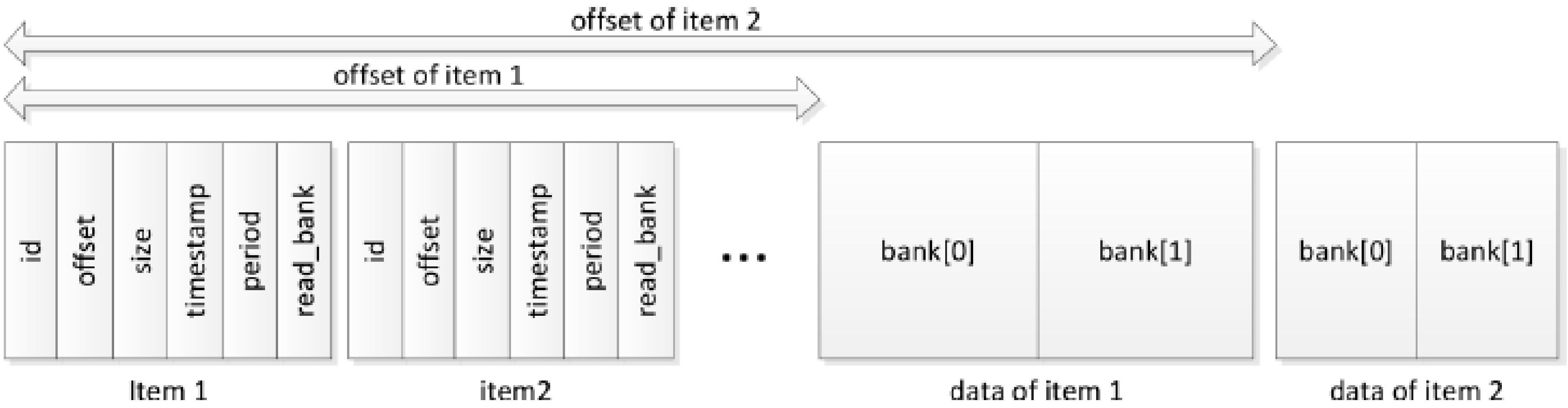
```
public:
  int DB_init (void)
  void DB_free (void)
  int DB_put (int id_it, void *data)
  int DB_get (int id_ag, int id_it, void *data)

protected:
  int DB_comm_init(RTDBconf_var *rec)
  int DB_comm_put(int id_ag, int id_it, int size, void *data, int age);
```

RTDB access from the node side

RTDB access from the network side

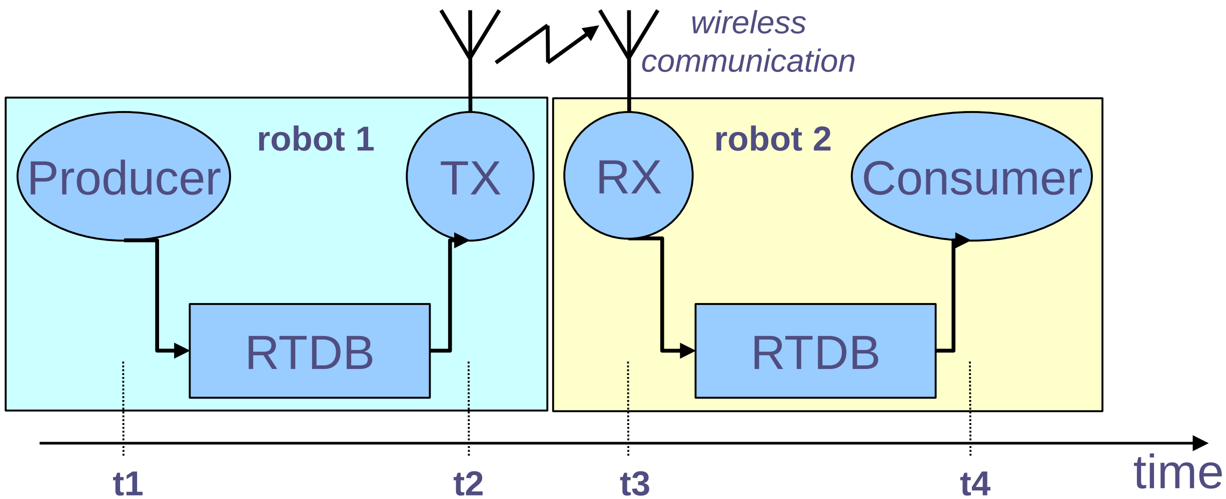
RTDB internal data structure




```
typedef struct {
  int id;
  int offset;
  int size;
  int period;
  struct timeval timestamp[2];
  int read_bank;
} TRec;
```

RTDB age of each item

- **No global clock synchronization → use age (relative)**



(tx – local timestamps)

t1 – Robot 1 produces and writes data into the RTDB

t2 – Communication protocol fetches data and also sends age (*t2 – t1)*

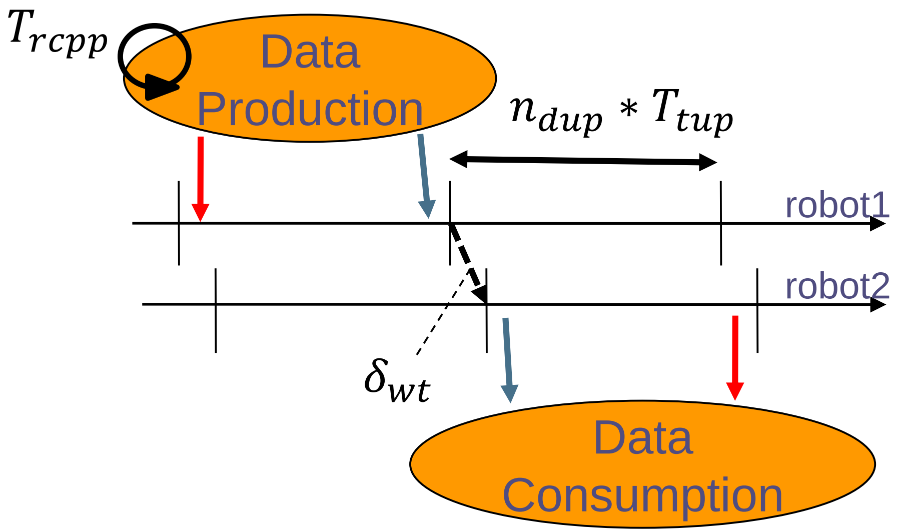
t3 – Robot 2 writes robot 1 data to RTDB and subtracts (*t2 – t1*) from *t3*

t4 – Consumer reads data and computes total data age

age = t4 – (t3 – (t2 – t1)) + wireless_communication_delay

RTDB age of each item

- **Maximum age (worst-case)**



Data produced just after a communication broadcast

Data consumed just before a new reception from robot1, with updated info

$$\delta_{wt} \leq \textbf{age} \leq T_{rcpp} + \delta_{wt} + (n_{dup} * T_{tup})$$

Best-Case Worst-Case

Wireless communications

Some wireless specifics

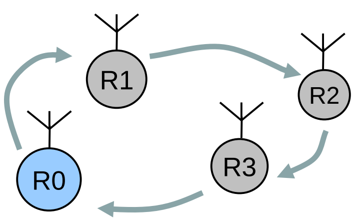
- Open medium, uncontrolled environment / load, non-stationary interference...
  - **Real-time** properties have **low coverage**
- Fading
  - **Connectivity** among the team **not guaranteed**

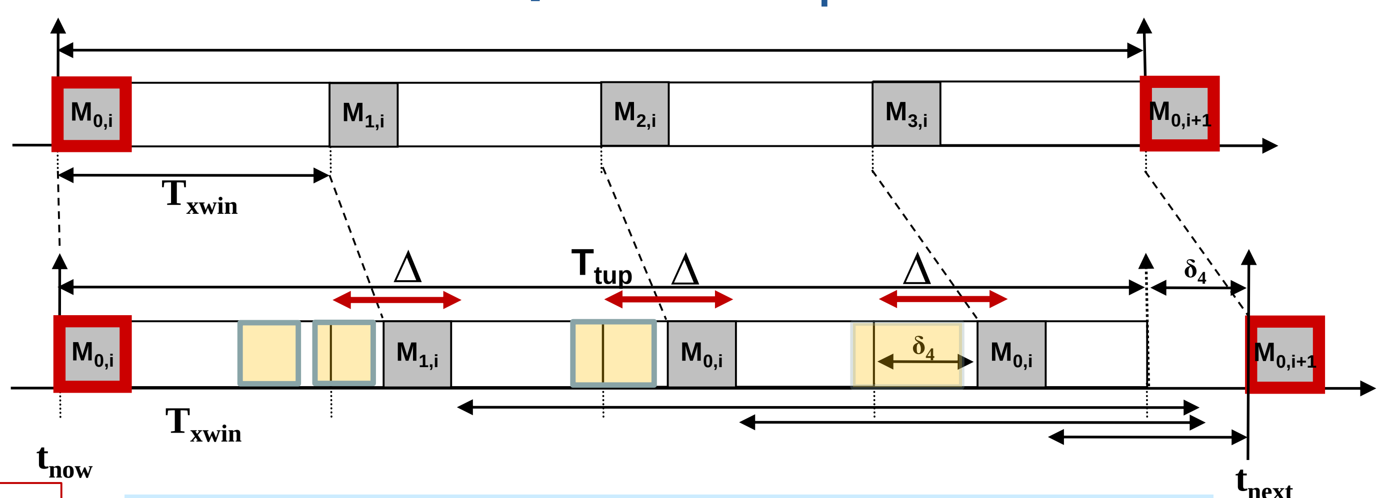
Our claim

- Robots **transmission pattern** is typically **periodic**
- **Automatically synchronizing** transmissions reduces chances of collision within the team
  - Improved performance in **latency** and **packets lost**

Adaptive TDMA

- **TDMA (+CSMA/CA) with synchronization on receptions**
  - no need for clock sync
- **Phase of round shifted to avoid external interference**
  - Maximizes separation between transmissions in the team
- **Time constraints → round period T_{tup}**





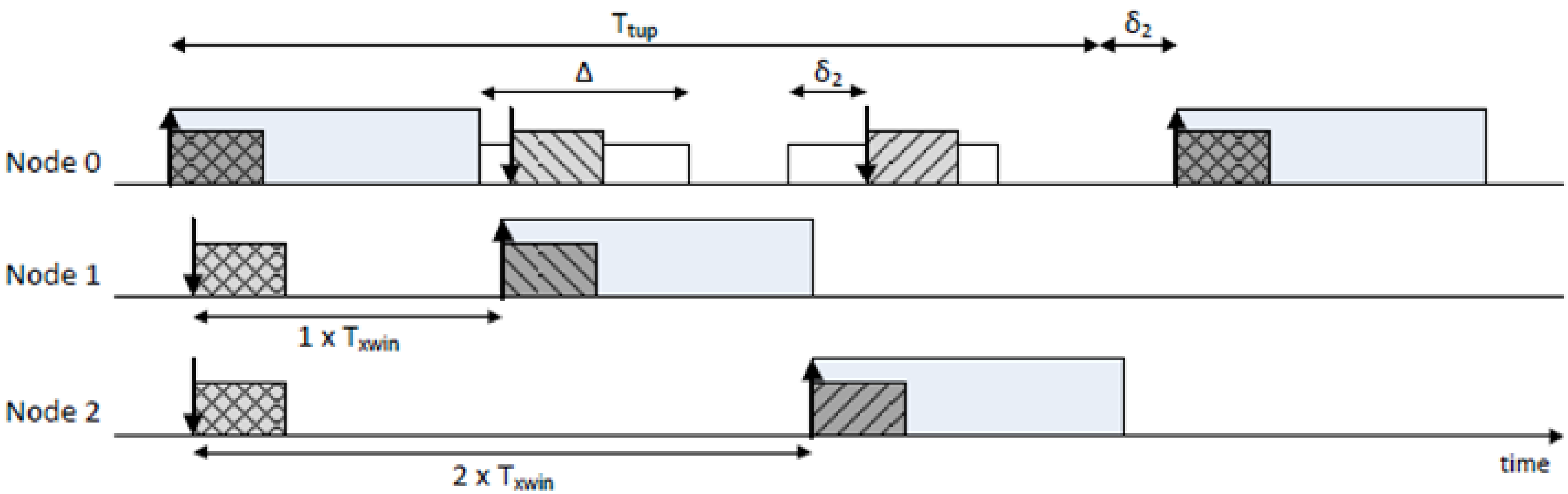
Vision of node 0

If $0<\delta_k<\Delta$ then $t_{next} = t_{now}+T_{tup}+max_k(\delta_k)$

round period will be within $[\ T_{tup}\ ,\ T_{tup}+\Delta\ ]$

Adaptive TDMA

- **Node with lowest ID is the round reference**
  - All others synchronize to its transmissions using an adequate offset
  - Avoids cliques and becomes resilient to AP shaping



$$t_{0,next} = t_{0,now} + T_{tup} + \max_{\substack{j=1..N, j \neq i \\ \delta_j \leq \Delta}} \delta_j$$

$$t_{i,next} = \begin{cases} t_{i,now} + T_{tup} \\ t_{0,now} + i * T_{xwin} \end{cases} \forall_{i=1..N}$$

Configuring the protocol

- **Bandwidth stability equation**
  - T_{tup} – Team update period → sets the reactivity of the protocol
    - Determines the real time properties
  - ρ – bit rate (should be fixed for protocol analyzability, e.g., 24Mbit/s)
  - D_i – Amount of data transmitted by node i
  - L – Bandwidth reserved for external traffic

$$T_{tup} > \frac{L\left(T_{tup} - \frac{\sum_{i=1}^{N} D_i}{\rho}\right)}{\rho} > \frac{\sum_{i=1}^{N} D_i \left(1 - \frac{L}{\rho}\right)}{\rho - L}$$

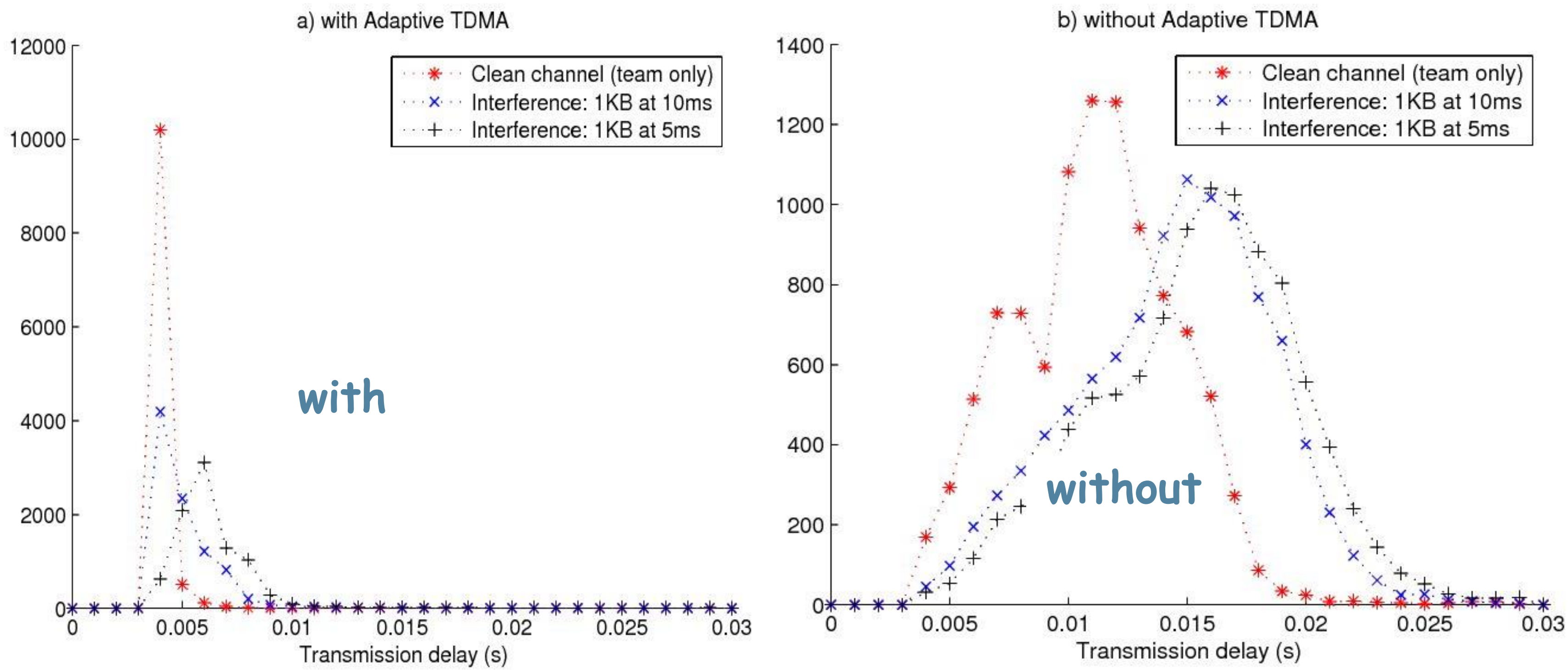
Configuring the protocol

- **Protocol stretchability (Δ)**
  - Determines
    - Tolerable delays keeping synchronization – Δ within T_{xwin}
    - The effective maximum TDMA round period – $T_{tup}+\Delta$
- **Stretchability coefficient (ε)**
  - Determines the protocol stretchability in relative terms

$$\Delta = T_{xwin} * \varepsilon \qquad 0 < \varepsilon \leq 1$$

Adaptive TDMA

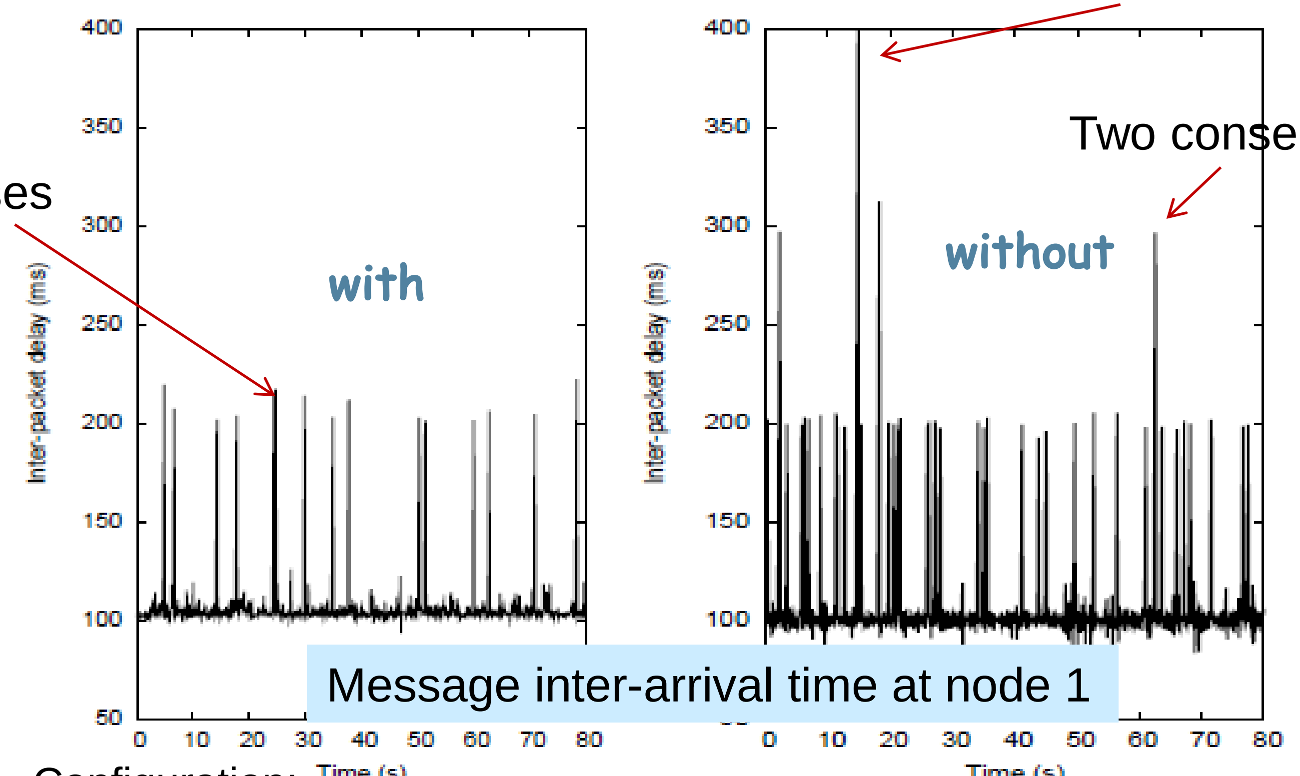
- **Positive impact** verified in practice under intense communication
  - **Network delay**



<u>Configuration</u>:
IEEE 802.11b, managed; 379 bytes packet size; 4 robots; T_{tup} = 50ms
All robots start communication at the same time using an external trigger signal

Adaptive TDMA

- **Positive impact** verified in practice under intense communication
  - **Packet losses**

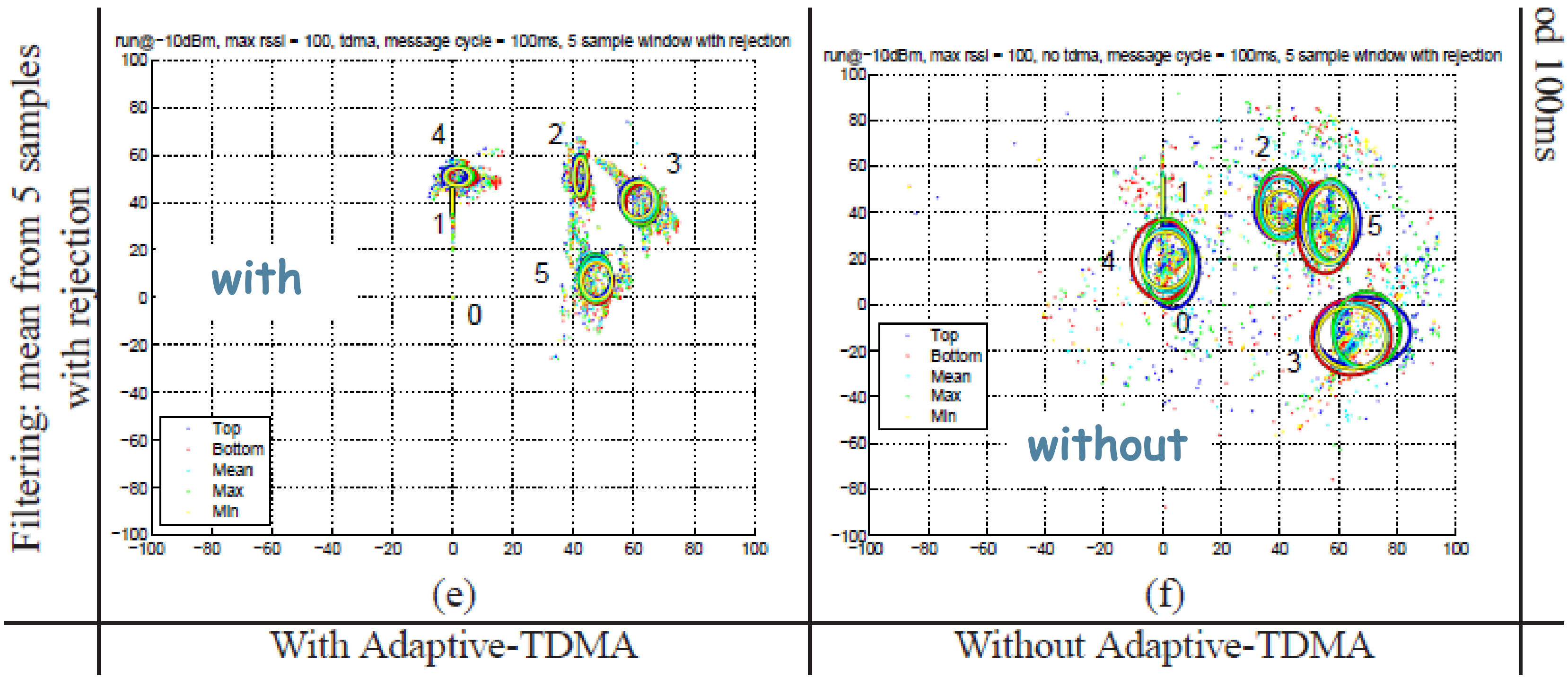


Configuration:
IEEE 802.11b, managed; 379 bytes packet size; 4 robots; T_{tup} = 100ms
All robots start communication at the same time using an external trigger signal

Adaptive TDMA

- **Positive impact** verified in practice under intense communication
  - **Application performance (localization)**

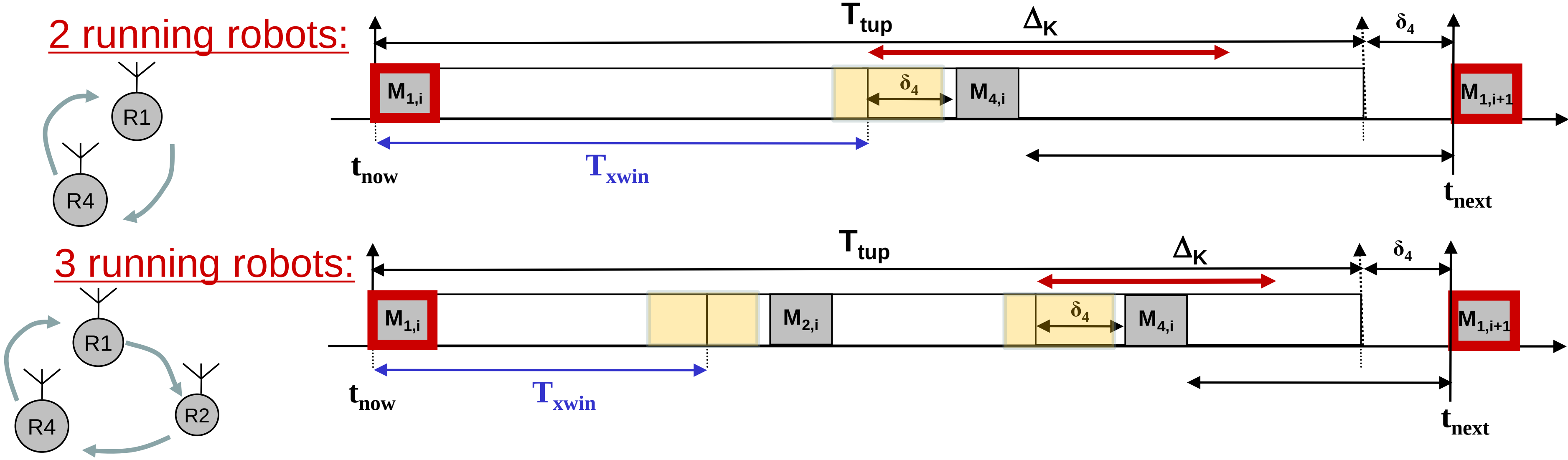
Ovals represent std of the position estimates



Configuration:
IEEE 802.15.4, beaconless; 64 bytes packet size; 4 robots; T_{tup} = 100ms
Ordinary operational conditions (not worst-case trigger)

Reconfigurable & Adaptive TDMA

- **Robots join and leave dynamically**
  - crash, maintenance, movements...
- **Slots are created / destroyed dynamically)**
  - **Fully distributed – virtually configuration-free**



The round (RT requirements) is kept but the separation between team member transmissions is maximized

F. Santos, L. Almeida, L. S. Lopes. Self-configuration of an Adaptive TDMA wireless communication protocol for teams of mobile robots. ETFA 2008. Hamburg, Germany, 15-18 September 2008.

Round structure management

- **Logical Ids**
  - Needed for dynamic definition of reference 0 and slots assignment
- **Dynamic membership**
  - Number of running robots (in the team) keeps varying – **K** ≤ N
  - Slot width varies – $T_{xwin,K}$
  - Slot validity window varies – Δ_K

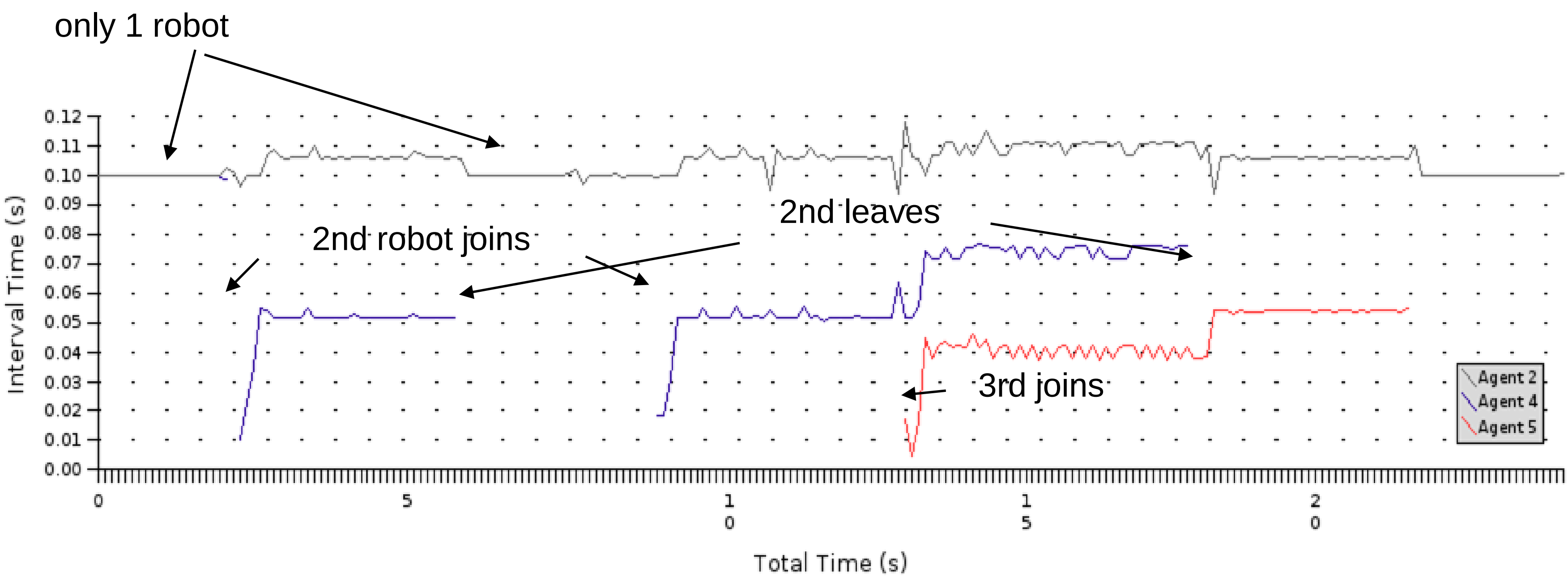
Phys	Memb	Logic
0	--	
1	OK	0
2	--	
3	OK	1
4	--	
5	--	
6	OK	2

$$T_{xwin,K} = \frac{T_{tup}}{K} \qquad K \leq N$$

$$\Delta_K = T_{xwin,K} * \varepsilon \qquad 0 < \varepsilon \leq 1$$

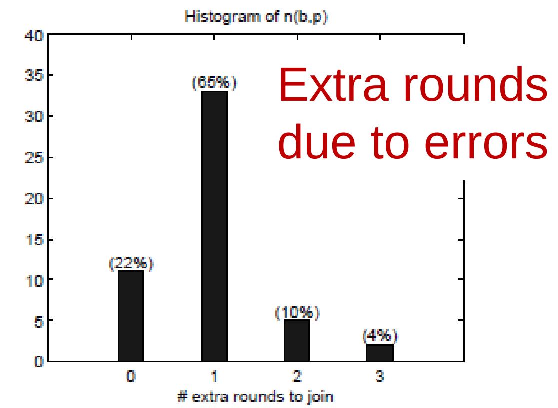
Reconfigurable & Adaptive TDMA

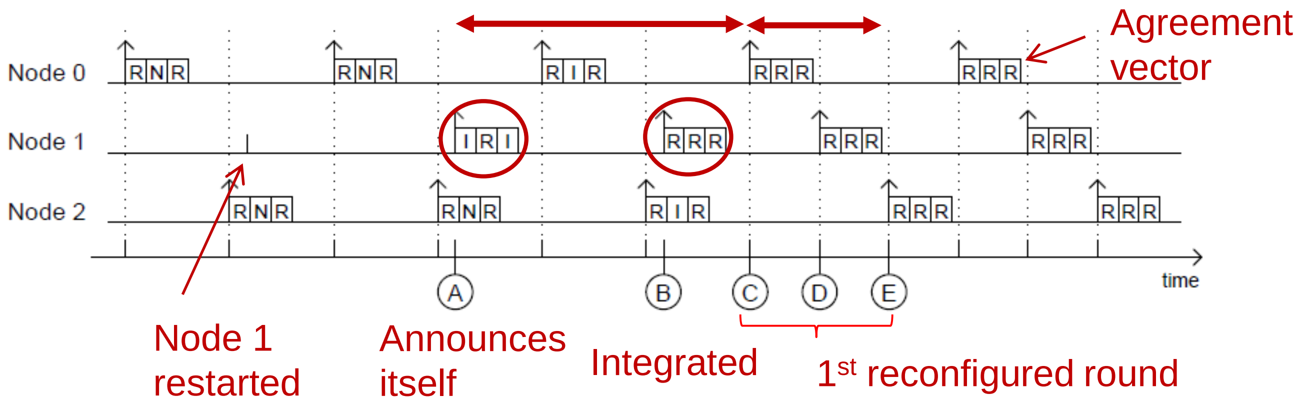
- **Message arrival times wrt node 0**



The joining process

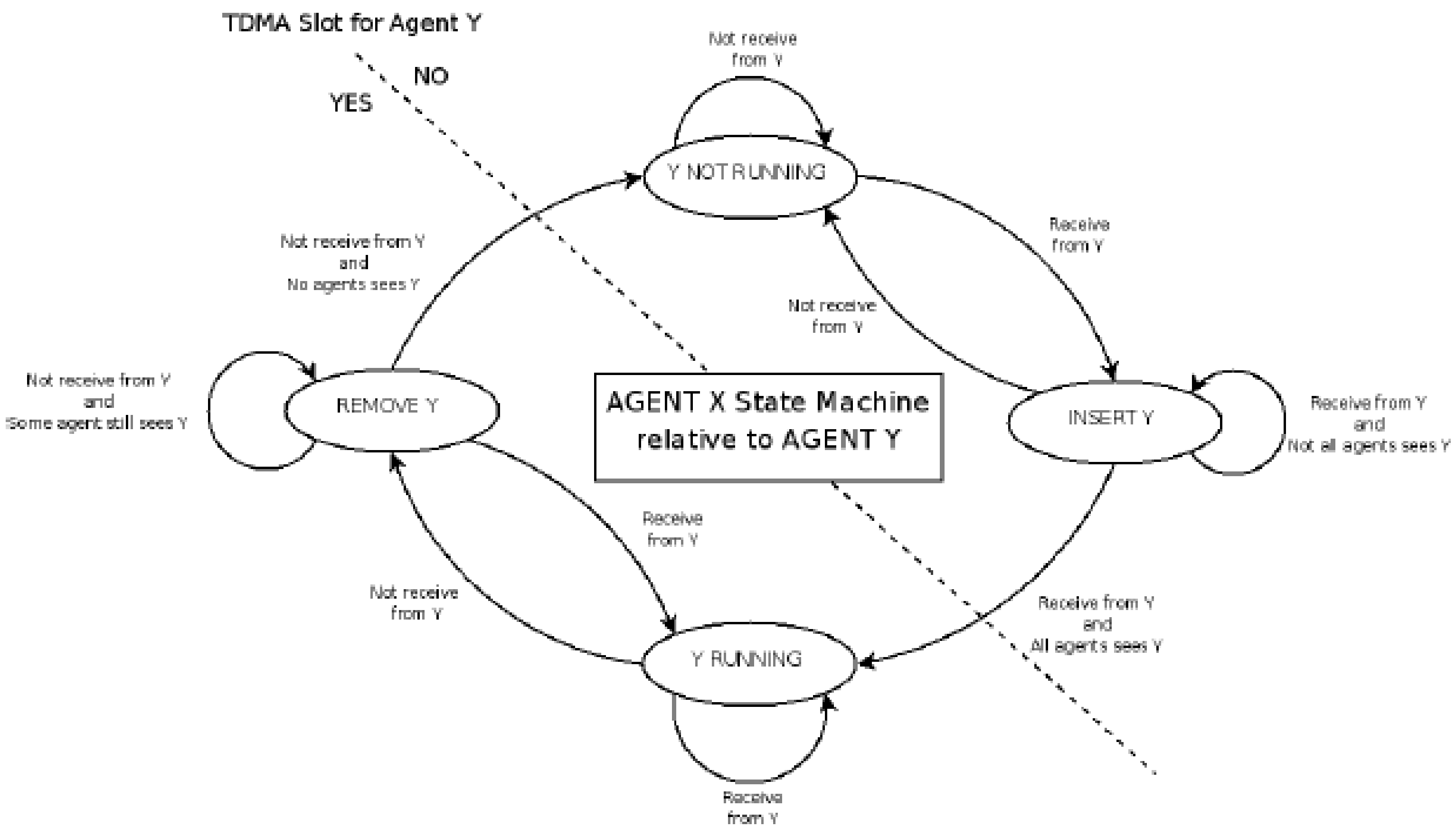
- **Using an AP** simplifies **team membership** definition and speeds up the agreement process **for reconfigurations**
  - **Topology** becomes virtually **fixed**
  - **Agreement** (A-C) takes **[1 2] rounds**
  - **Resynchronization** (C-E) takes **[0 1] round**





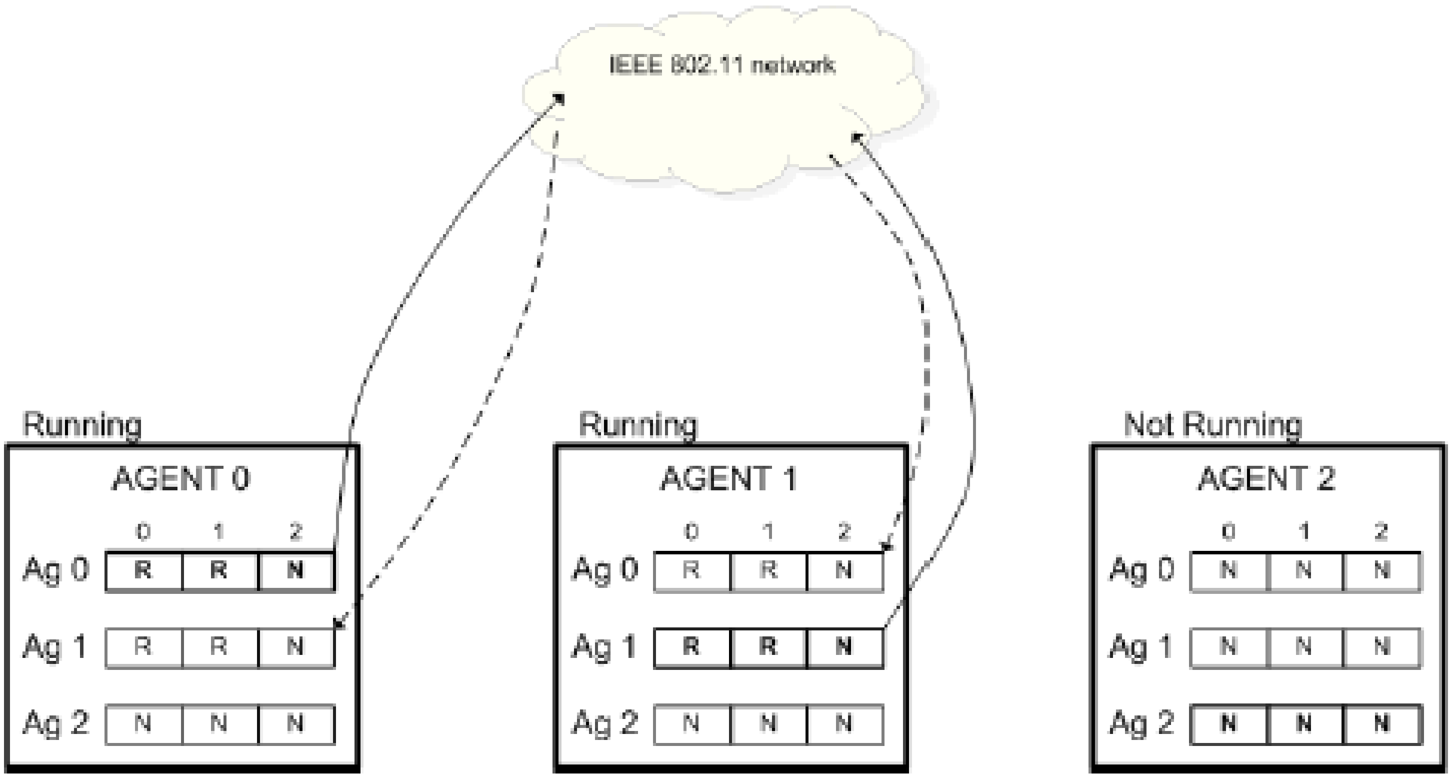
Round structure management

- State machine of agent X with respect to each agent Y



Round structure management

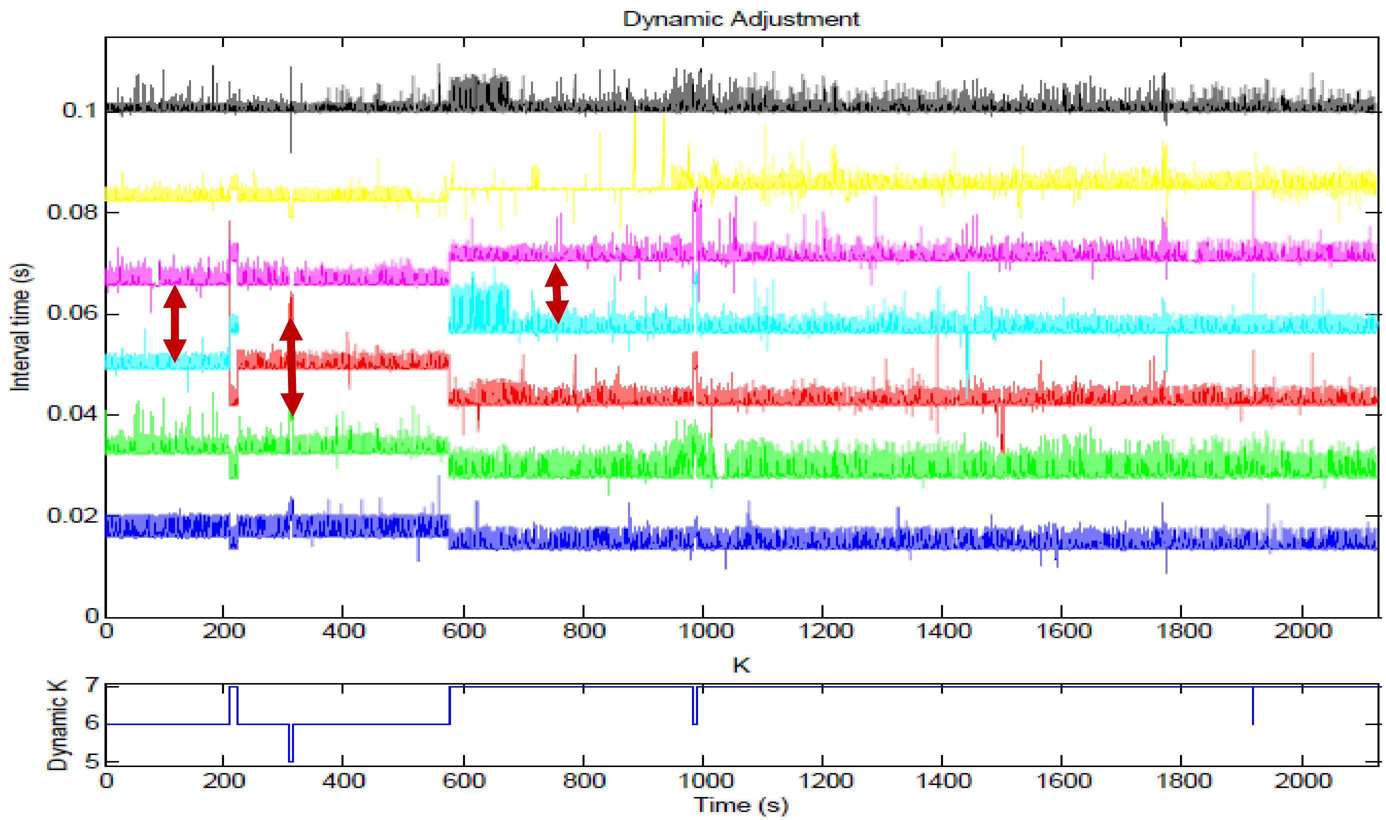
- **Agreement on the current round structure**
  - Dissemination of **membership vectors**



Membership and round structure

Offset of Tx in the round

Team members



FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

On the use of the protocol

- **Adequate to disseminate state information**
  - On the contrary, implies extra delays on event transmission
  - **Events** should be sent as **external traffic**, outside the protocol control
- **Typical behaviors**
  - Collaborative **ball tracking**
  - **Formation** control
    - Team entrance in and departure from field
    - Set-plays (tactics) enforcement
  - Collaborative **sensing** for strategic reasoning
    - At the coach level

Source code available at:
www.bitbucket.org/fredericosantos/rtdb/

Recent additions to the protocol

- **Asymmetric bandwidth requirements**
  - Some robots may have more information to share than others
  - Requirements are known by all since they are currently static
  - Division of slots is done proportionally to the width of the respective data
- **Asymmetric reactivity requirements**
  - The round sets the reactivity of the communications (dead interval)
  - Robots that need to transmit more often can ask for an extra slots
  - Extra slots are given as if it was another (virtual) robot joining
- **Generic OS protocol interface**
  - Allow other general Internet applications to use the protocol transmitting more frames with the remainder of the slot

Ad-hoc
Reconfigurable and Adaptive TDMA protocol (RA-TDMA+)

(WiFi and IEEE 802.15.4)

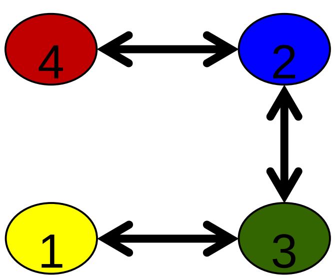
Search and Rescue scenarios

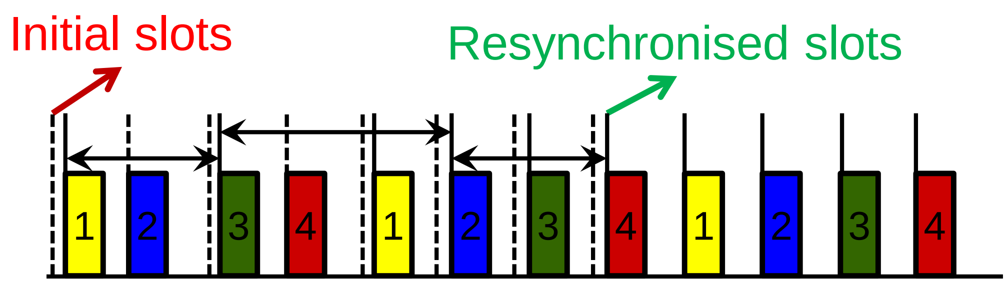
- **Multi-hop topology is more favorable**
  - Higher flexibility in area coverage and formation control
- **but**
  - New robots can **join in a "corner"** of the network
  - There can be **localized noise** affecting a few robots, only

Ad-hoc synchronization

- **Slot synchronization is localized**
  - Robots synchronize with their **neighbours**
- **Synchronization is propagated through the network**
  - Even if robots do **not communicate directly** they synchronize
  - Time necessary to reach **synchronization depends** on **link density**

And does it converge?

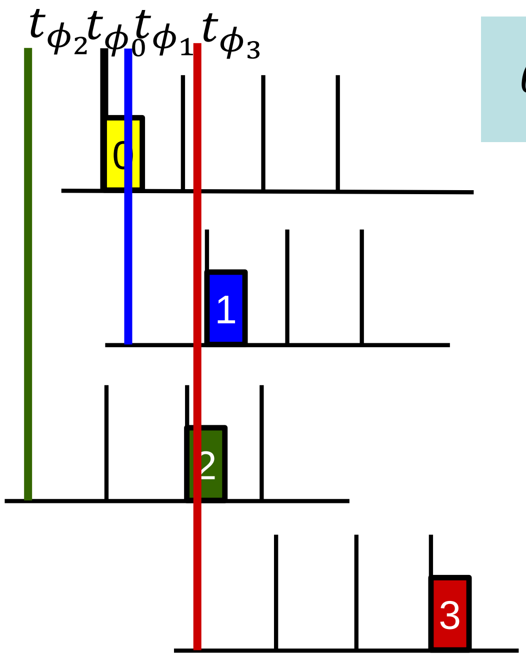




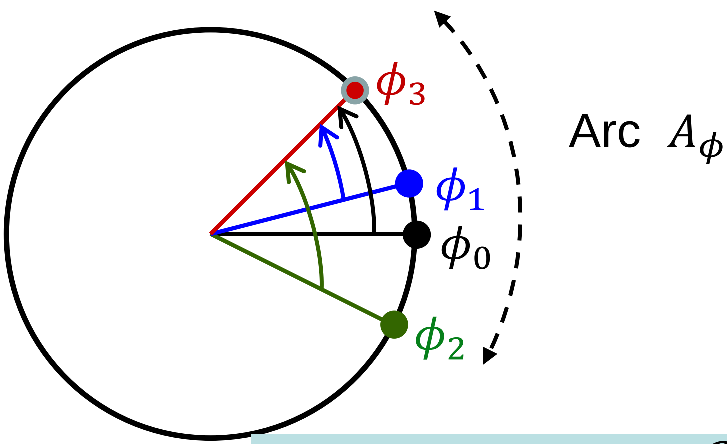
$$t_{i,k+1} = t_{i,k} + T_{tup} + \max_{j \in \aleph_i : \delta_j < \Delta} \delta_j$$

round *k*
robot *i*
neighbour *j*
Δ max delay allowed

Expressing delays as angles → phases



$$\theta(t) = t * \frac{2\pi}{T_{up}}$$



$$\phi = \left(t_\phi + k * T_{up}\right)\frac{2\pi}{T_{up}}$$

$$\delta_i = \left(\max\left(\phi_{j\epsilon\aleph_i}\right) - \phi_i\right)\frac{T_{up}}{2\pi}$$

$$\phi_i \leftarrow \phi_i + \min_{j\epsilon\aleph_i}\left(\delta_j, \Delta\right)\frac{2\pi}{T_{up}}$$

When all robots "see" the same round (synchronized), then they all have the same t_ϕ ou seja Arc A_ϕ =0

If Arc ≥ π then A_ϕ → 0 (means robots converge!)

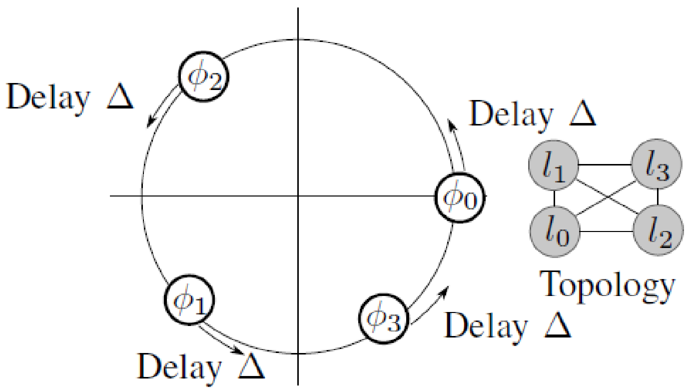
U.PORTO

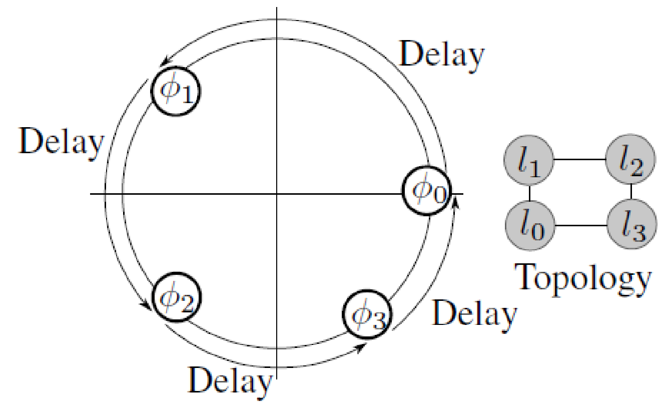
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Anomalies when Arc ≥ π

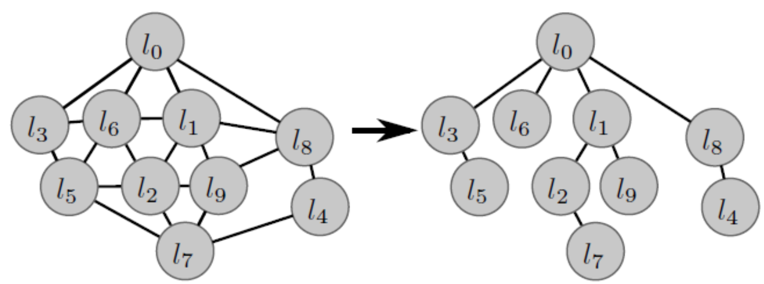
Situations with loops

Robots do not converge to the same round…



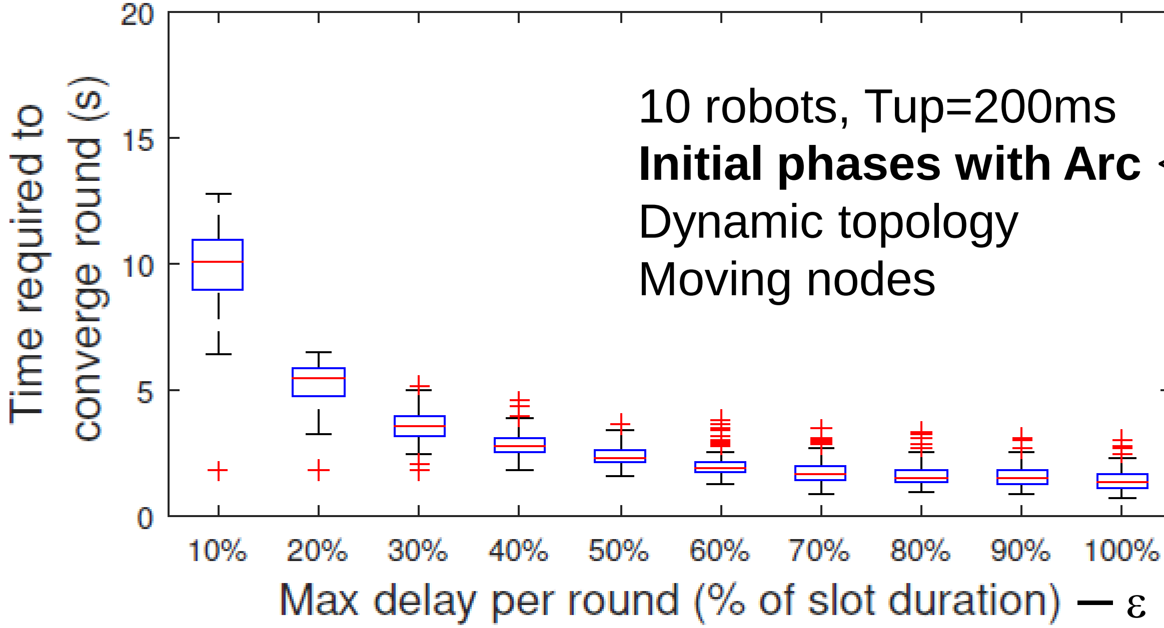


Solution: Spanning-tree



Enforces convergence but takes more time (information propagates slowlier)

Impact of Δ in the convergence time

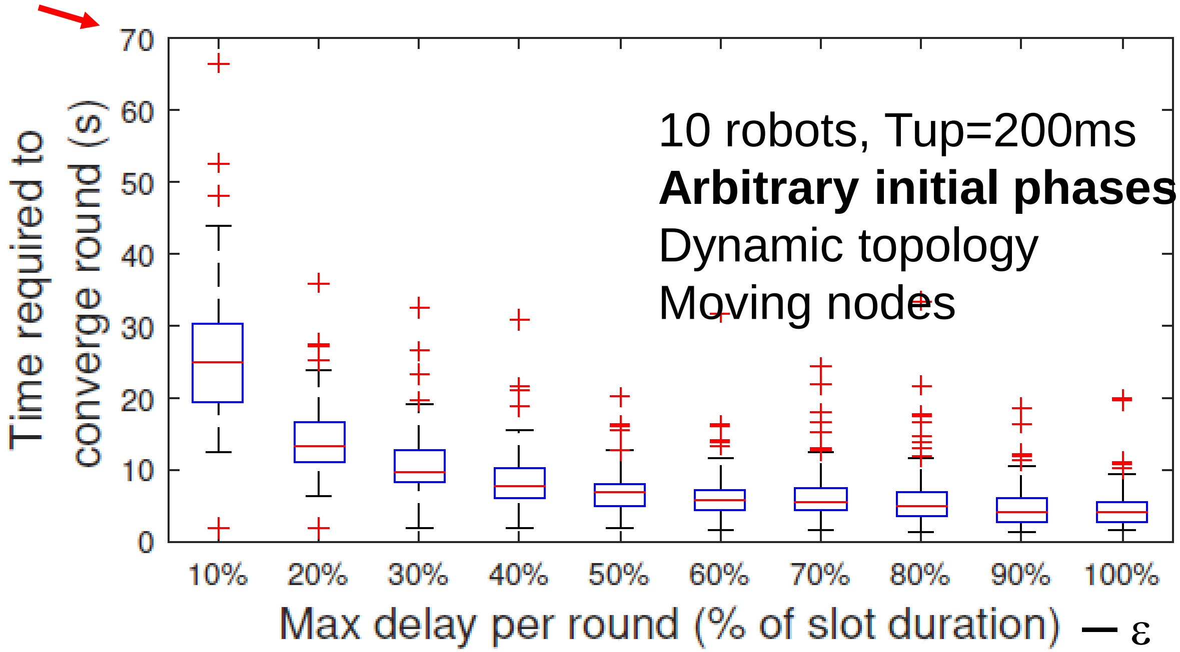


Smaller Δ (ε)
Smaller phase adjustments
Slower convergence

Larger Δ (ε)
Larger phase adjustments
Faster convergence

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Impact of Δ in the convergence time



Smaller Δ (ε)
Smaller phase adjustments
Slower convergence

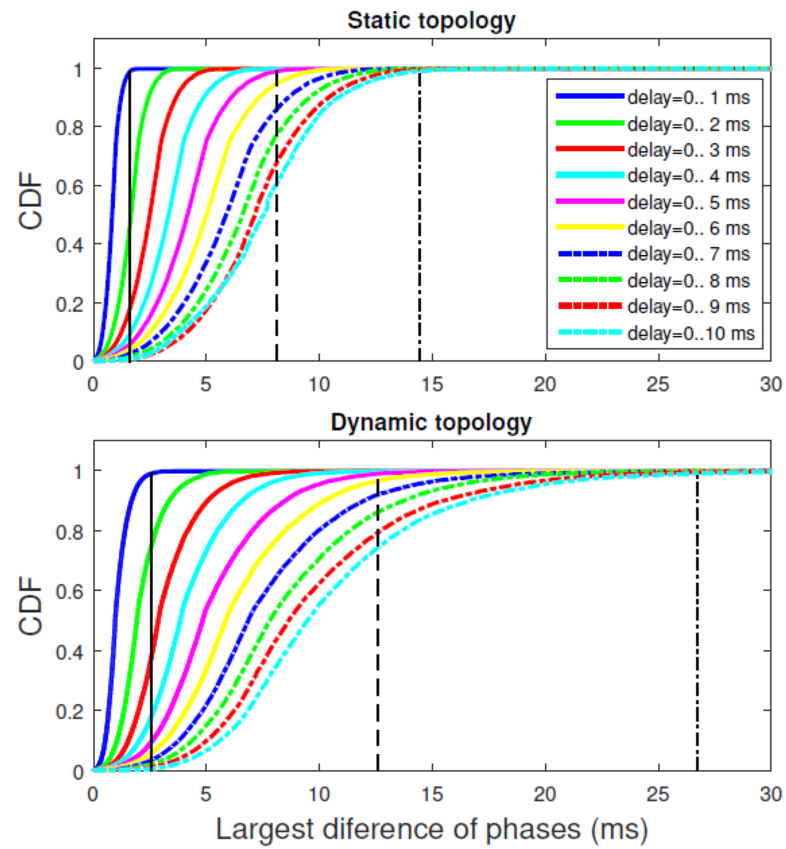
Larger Δ (ε)
Larger phase adjustments
Faster convergence

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Impact of mobility and network delays

10 robots, initially in phase
Tup=200ms
Δ=8ms (ε=40%)
Different levels of interference

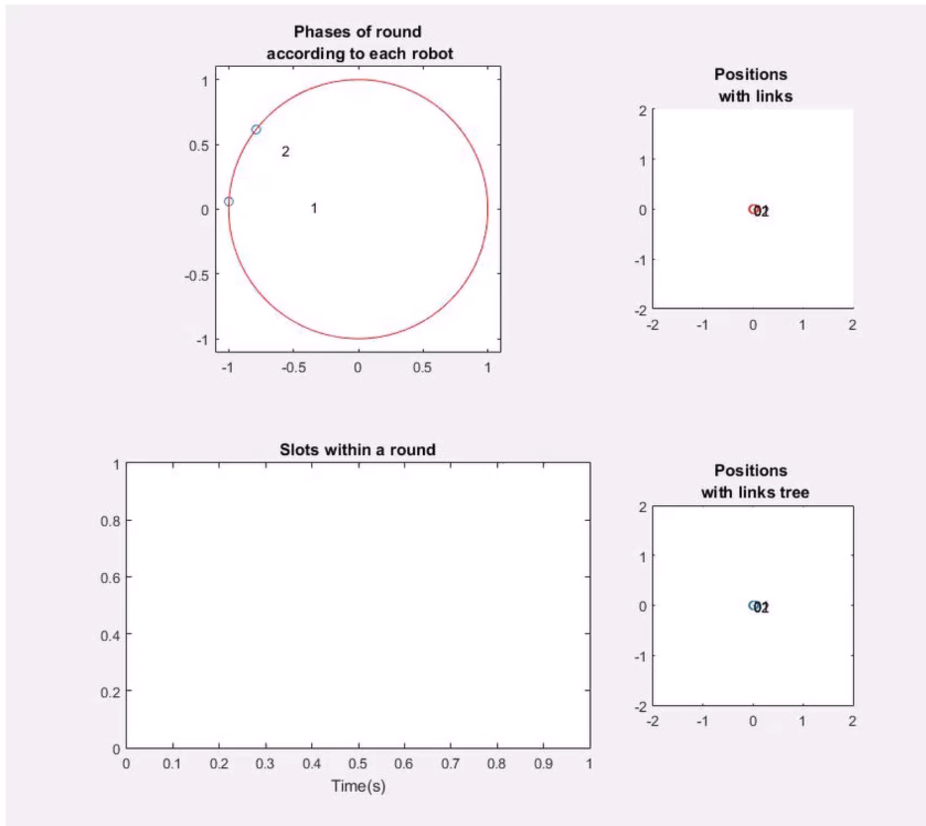
Synchronization kept Arc < π
Arc = π → 100ms difference
Dynamic topology increases Arc



99 percentiles

Synchronization in phase and time

Obs.:
Joining and leaving robots use topology tracking
(coming next)

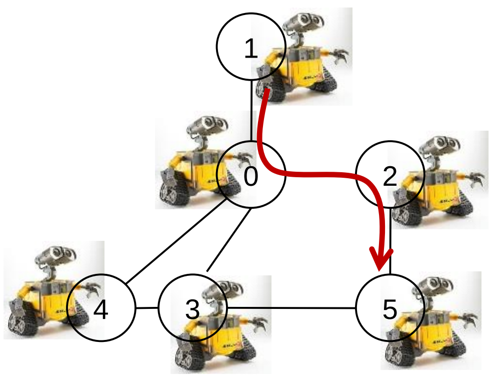


Physical topology with moving robots

Forced spanning-tree topology

Membership? Topology tracking?

- **Build and maintain an adjacency matrix**
  - Sense **neighborhood** and disseminated (**flooding**) in the **synchronization message**
  - **Merge matrices** with neighbors
  - Enables on the fly **(proactive) routing**



	0	1	2	3	4	5
0	0	1	1	1	1	0
1	1	0	0	0	0	0
2	1	0	0	0	0	1
3	1	0	0	0	1	1
4	1	0	0	1	0	0
5	0	0	1	1	0	0

Vision of node 0

Neighbor

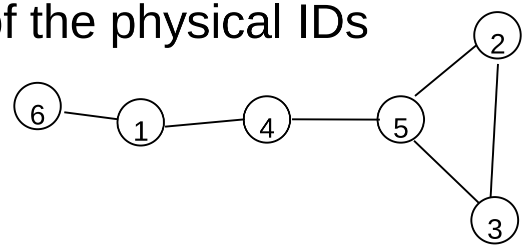
Not neighbor

Who receives from node 0

Luis Oliveira, Luis Almeida, and Frederico Santos. A Loose Synchronisation Protocol for Managing RF Ranging in Mobile Ad-Hoc Networks. RoboCup Symposium 2011, Istanbul, Turkey. July 11, 2011 (Lecture Notes in Computer Science LNCS 7416, Springer 2012).

Converging to a global adjacency matrix

- **Detecting omissions**
  - The **corresponding bit** in the local vision (its line) in the receiver matrix is **reset**
  - A column with all 0s means that **node is disconnected** from the team
- **Uses sequence numbers per line**
  - **Update** the lines with **higher sequence number** (seq num included)
  - **Increment** just before transmission and send together
  - **Erase lines** that have not been updated for a some time
- **Slot allocation**
  - Based on the order of the physical IDs



$S^3(t)$

1	5
2	2
3	3
4	5
5	5
6	5

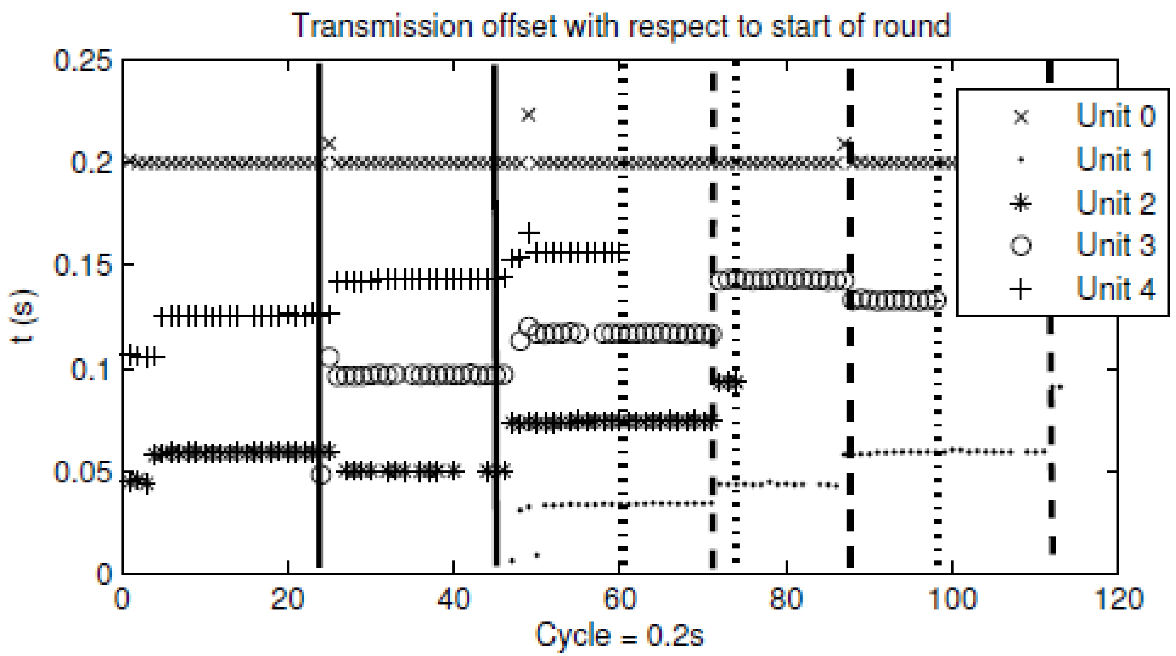
$M^3(t)$

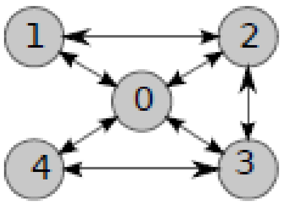
	1	2	3	4	5	6
1	0	0	0	1	0	1
2	0	0	1	0	1	0
3	0	1	0	0	1	0
4	1	0	0	0	1	0
5	0	1	1	1	0	0
6	1	0	0	0	0	0

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Operation example

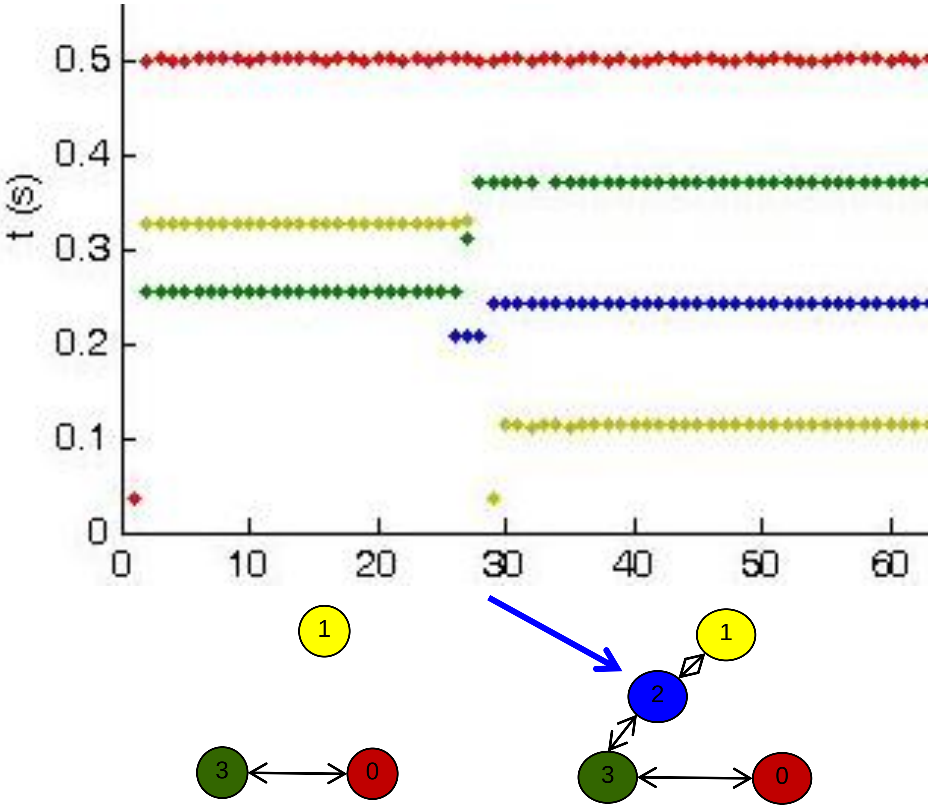
- Offset of packet arrivals relative to node 0





Joining cliques

- Offset of packet arrivals relative to node 0



Wrapping up – Global conclusion

Conclusion

- Cooperation among robots requires
  - **wireless communication, synchronization and relative localization**
- Interference, errors, multi-path fading, attenuation …
  - Lead to poor coverage of real-time assumptions
  - **Adaptive and reconfigurable** mechanisms are particularly suited to provide **graceful degradation**
- Synchronization to reduce collisions is worthwhile
  - Particularly, for periodic traffic and **high medium utilization**
- The RSSI can provide support to team-level functionality
  - Such as **relative localization and navigation**
  - Combination with **ToF** seems promising

Conclusion

- **A few open issues ...**
  - Using an **AP** (possibly mobile) versus **fully ad-hoc** mode
  - Switching **synchronization on/off** depending on **medium load**
  - Faster and efficient **topology tracking** to
    - Cope with higher mobility
    - Support a better combination of reactive and proactive routing
  - **Better processing of the RSSI** to improve its usability
  - Applicability of **new RF-ranging** devices
  - **Team coordination** methods that...
    - Cope with **limitations of the wireless** communication
    - **Manage team connectivity** (maybe not needed permanently)
      - Manage clustering in these dynamic networks
    - **Optimize the global use of the team resources**
      - energy, computing, specific subsystems, ...

Suggestion!
A simulator for team coordination

- **CyberRescue@RTSS2006-9**
  - **http://robot.unipv.it/cyberrescue-RTSS09/**
- ***Control a team of 5 robots with ad-hoc communication capabilities to reach the victim in the least time***