Distributed sensor calibration and least–square parameter identification in WSNs using consensus algorithms

Saverio Bolognani, Simone Del Favero,
Luca Schenato, Damiano Varagnolo

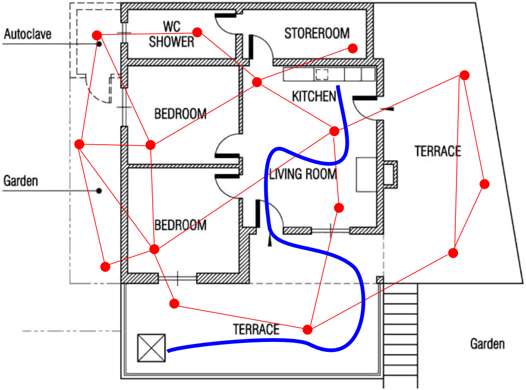
UNIPD

September 25th, 2008

Motivation of the work

Aim of the work:

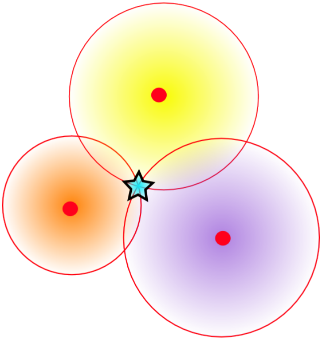
To address some modeling and algorithmic issues of localization and target tracking in wireless sensors networks

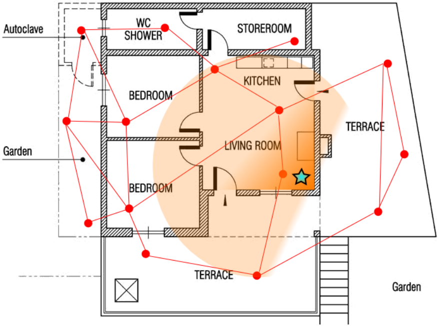


2 approaches to target tracking:

Map Based

Most likely location that matches with pre-learned maps.



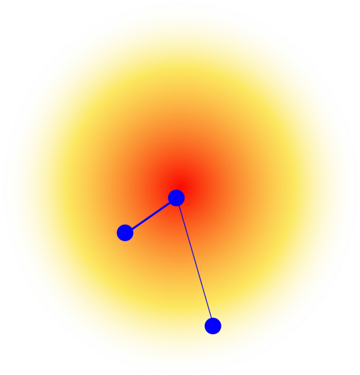


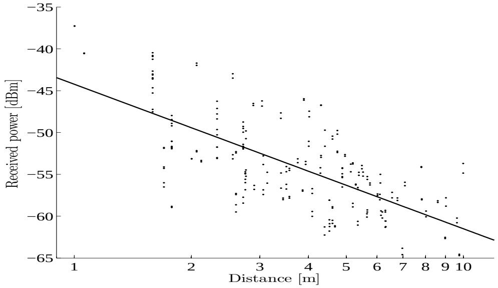
Range based

Triangulation (similarly to GPS)

Localization and tracking, the idea:

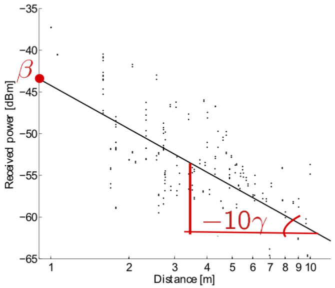
- Nodes measure the radio signal strength of the received packet
- It depends on the distance tx-rx.



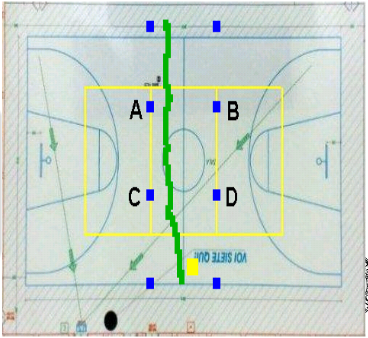


- From the distance it can be estimated the position of the nodes

2 issues:



- Wireless channel parameters identification is needed.
  If possible has to be done in-situ by a distributed algorithm.



- Uncalibrated Sensors. Offset in the Radio Signal Strength measure.
  A distributed calibration algorithm is needed.

Contribution of this work

- Experimental validation of the wireless channel model proposed
- Development of a distributed, consensus based, algorithm for
    - Sensor Calibration
    - Wireless Channel Least Square Parameter Identification
- Testing the proposed algorithms on real data

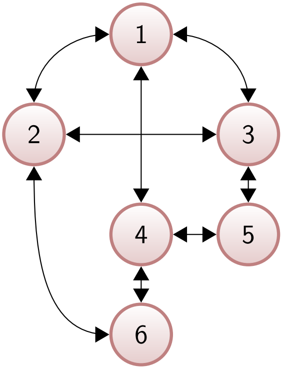
S. Bolognani, S. Del Favero, L. Schenato, and D. Varagnolo, *Distributed sensor calibration and least-square parameter identification in WSNs using consensus algorithms,* Allerton '08, Illinois, 2008.

What is Consensus?

The solution we proposed is based on Consensus Algorithms.
What is Consensus?

Network with

- N agents
- a communication graph
- a stored variable for each node.



Consensus Algorithm

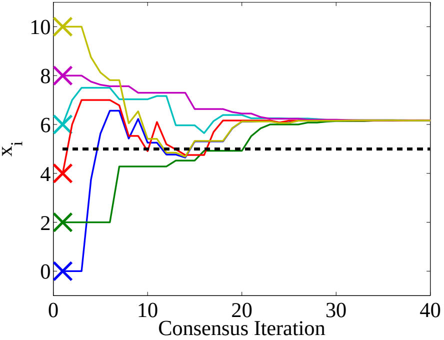
A distributed algorithm (node states update based on the neighbors states only) which makes the nodes to reach consensus on their state.

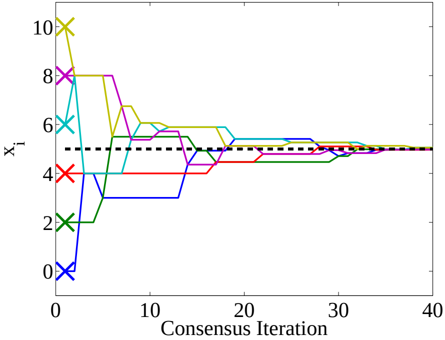
Linear consensus algorithms

in the form

$x_i^+ = x_i + \sum_{j \in \mathcal{V}(i)} p_{ij}(x_j - x_i)$

are proved to asymptotically achieve consensus.



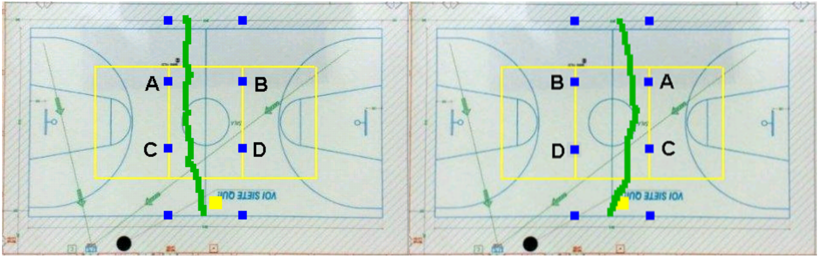


Average consensus

can be achieved by linear consensus algorithms, but this requires the communications between nodes to be “symmetric”.

Offset effects

Reception offset is particularly harmful for localization applications, Experiment inside a basketball court.[1]



[1]Courtesy of ST Microelectronics,
I. Solida, "Localization services for IEEE802.15.4/Zigbee devices. Mobile node tracking (in Italian)", Master Thesis, Department of information Engineering, University of Padua, 2007

Sensor Calibration

Ideally:

- Estimate o_i: $\hat{o}_i$
- Use $\hat{o}_i$ to compensate the offset: $o_i - \hat{o}_i = 0$

Remember the previous example

What we propose is:

$$o_i - \hat{o}_i = \alpha \qquad \alpha \cong 0 \qquad \text{equal for all nodes}$$

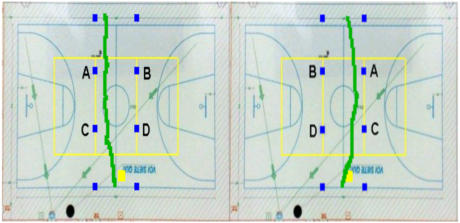
All nodes overestimate or underestimate the distance similarly.
The channel parameter estimation will deal with this common offset.

Sensor Calibration

Ideally:

- Estimate o_i: $\hat{o}_i$
- Use $\hat{o}_i$ to compensate the offset: $o_i - \hat{o}_i = 0$

Remember the previous example



What we propose is:

$$o_i - \hat{o}_i = \alpha \qquad \alpha \cong 0 \qquad \text{equal for all nodes}$$

All nodes overestimate or underestimate the distance similarly.
The channel parameter estimation will deal with this common offset.

Sensor Calibration as a Consensus Problem

Asking

$$o_i - \hat{o}_i = \alpha \text{ equal for all nodes}$$

means to cast sensor calibration into a consensus problem

Consider the consensus algorithm

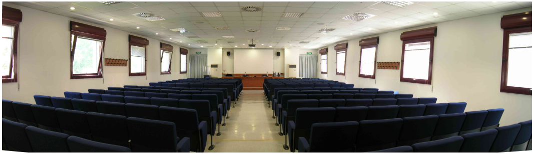
$$(o_i - \hat{o}_i)^+ = (o_i - \hat{o}_i) + \sum_{j \in \mathcal{V}(i)} p_{ij}((o_j - \hat{o}_j) - (o_i - \hat{o}_i))$$

That leads to

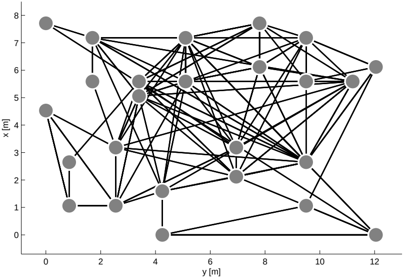
$$\hat{o}_i^+ = \hat{o}_i - \sum_{j \in \mathcal{V}(i)} p_{ij}(\bar{P}_{rx}^{ji} - \bar{P}_{rx}^{ij} + \hat{o}_j - \hat{o}_i)$$

Experimental Testbed

25 TMote-Sky nodes with Chipcon CC2420 RF Transceiver randomly placed inside a single conference room:



Network topology and nodes displacement:

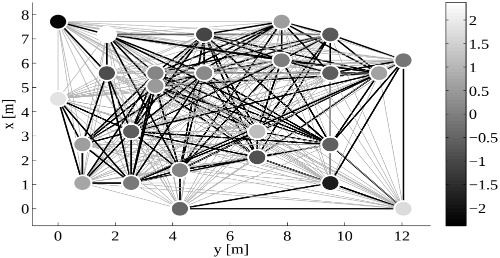


Courtesy of Zanca and Zorzi, "Measurement on CC2420 radio chipset", Department of Information Engineering, University of Padua, Italy, Tech. Rep.,2008.

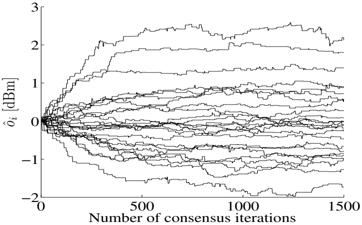
Experimental Results

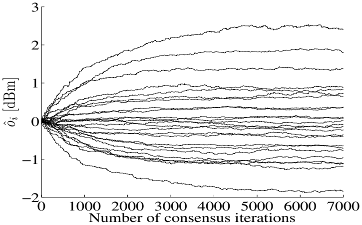
Links divided in 2 categories:

- Training links (black)
- Validation links (gray)



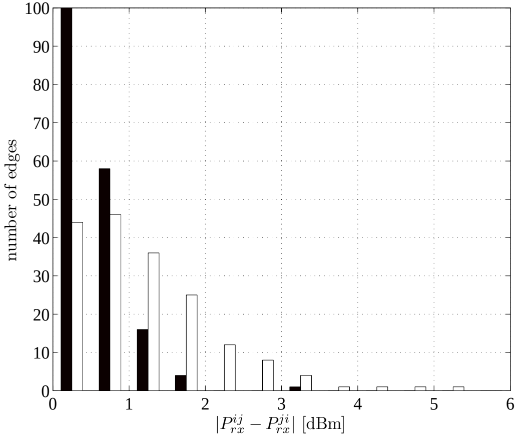
Estimate time evolution





Experimental Results: Asymmetric error before and after offset correction

$$\Delta \bar{P}^{ij} = \bar{P}^{ij} - \bar{P}^{ji} = o_i - o_j$$



	Before	After
< 1	50%	88 %
> 2dB	35%	0.6 %

Effects of systematic errors when estimating distances

$1dB \longmapsto \cong 2m \pm 0.28m$.
$6dB \longmapsto$ uncertainty for $0.9m$ to $4.4m$ for an actual distance of $2m$.

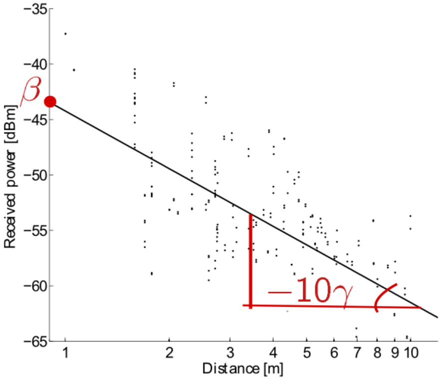
Least Square Parameter Identification

Another important problem:

Accurately identify the wireless channel parameters β and γ.

In fact:

- Parameters extremely environment dependent
- $\gamma \in [1, 6]$
- Environment change hourly or daily



Least Square Identification of Wireless Channel Parameters

Easiest Wireless Channel Model

$$\bar{P}_{rx}^{ij} + \hat{o}_i = \beta - \gamma 10 \log_{10}(d_{ij}) + w_i$$

$$\underbrace{\bar{P}_{rx}^{ij} + \hat{o}_i}_{b_{ij}} = \underbrace{[1 \quad -10 \log_{10}(d_{ij})]}_{a_{ij}^T} \underbrace{\begin{bmatrix} \beta \\ \gamma \end{bmatrix}}_{\theta} + w_i$$

What one would do is to apply least square estimation of the whole set of measurements to obtain $\hat{\beta}$ and $\hat{\gamma}$.

Least Square Identification of Wireless Channel Parameters

Easiest Wireless Channel Model

$$\bar{P}_{rx}^{ij} + \hat{o}_i = \beta - \gamma 10 \log_{10}(d_{ij}) + w_i$$

$$\underbrace{\bar{P}_{rx}^{ij} + \hat{o}_i}_{b_{ij}} = \underbrace{[1 \quad -10 \log_{10}(d_{ij})]}_{a_{ij}^T} \underbrace{\begin{bmatrix} \beta \\ \gamma \end{bmatrix}}_{\theta} + w_i$$

What one would do is to apply least square estimation of the whole set of measurements to obtain $\hat{\beta}$ and $\hat{\gamma}$.

Least Square as a Consensus Problem

Globally, the sensor network collected

M couples measure-regressors: $(a_1, b_1), \dots (a_M, b_M)$.

Let us call

$$A = [a_1, \dots, a_M]^T \text{ and } b = [b_1, \dots, b_M].$$
$$b = A\theta + w$$

The least square estimate of θ,

given the measurements b is

$$\hat{\theta} = \arg\min_{\theta} ||A\theta - b|| = (A^T A)^{-1} A^T b$$

$$\hat{\theta} = \arg\min_{\theta} ||A\theta - b|| = (A^T A)^{-1} A^T b = \Big(\sum_{i \in \mathcal{N}} a_i a_i^T\Big)^{-1} \Big(\sum_{i \in \mathcal{N}} a_i b_i\Big)$$

Consensus Solution:

- Initialize every node with $x_i(0) = a_i a_i^T$ and $y_i(0) = a_i b_i$
- Run any Average Consensus Algorithm
- $x_i(t) \longrightarrow \frac{1}{N} \sum_{i \in \mathcal{N}} a_i a_i^T$
- $y_i(t) \longrightarrow \frac{1}{N} \sum_{i \in \mathcal{N}} a_i b_i$
- Each node can compute then

$$\hat{\theta}_i(t) = x_i^{-1}(t) y_i(t)$$

$$\hat{\theta}_i(t) \rightarrow \Big(\sum_{i \in \mathcal{N}} a_i a_i^T\Big)^{-1} \Big(\sum_{i \in \mathcal{N}} a_i b_i\Big)$$

$$\hat{\theta} = \arg\min_{\theta} ||A\theta - b|| = (A^T A)^{-1} A^T b = \left(\frac{1}{N} \sum_{i \in \mathcal{N}} a_i a_i^T\right)^{-1} \left(\frac{1}{N} \sum_{i \in \mathcal{N}} a_i b_i\right)$$

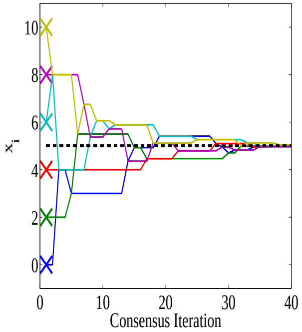
Consensus Solution:

- Initialize every node with $x_i(0) = a_i a_i^T$ and $y_i(0) = a_i b_i$
- Run any Average Consensus Algorithm
- $x_i(t) \longrightarrow \frac{1}{N} \sum_{i \in \mathcal{N}} a_i a_i^T$
- $y_i(t) \longrightarrow \frac{1}{N} \sum_{i \in \mathcal{N}} a_i b_i$
- Each node can compute then

$$\hat{\theta}_i(t) = x_i^{-1}(t) y_i(t)$$

$$\hat{\theta}_i(t) \to \left(\sum_{i \in \mathcal{N}} a_i a_i^T\right)^{-1} \left(\sum_{i \in \mathcal{N}} a_i b_i\right)$$

$$\hat{\theta} = \arg\min_{\theta} ||A\theta - b|| = (A^T A)^{-1} A^T b = \left(\frac{1}{N}\sum_{i\in\mathcal{N}} a_i a_i^T\right)^{-1}\left(\frac{1}{N}\sum_{i\in\mathcal{N}} a_i b_i\right)$$



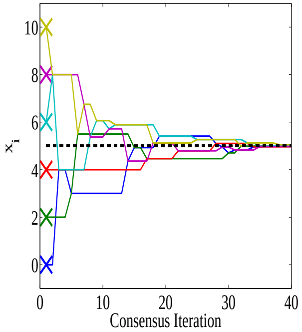
Consensus Solution:

- Initialize every node with $x_i(0) = a_i a_i^T$ and $y_i(0) = a_i b_i$
- Run any Average Consensus Algorithm
- $x_i(t) \longrightarrow \frac{1}{N}\sum_{i\in\mathcal{N}} a_i a_i^T$
- $y_i(t) \longrightarrow \frac{1}{N}\sum_{i\in\mathcal{N}} a_i b_i$
- Each node can compute then

$$\hat{\theta}_i(t) = x_i^{-1}(t) y_i(t)$$

$$\hat{\theta}_i(t) \rightarrow \left(\sum_{i\in\mathcal{N}} a_i a_i^T\right)^{-1}\left(\sum_{i\in\mathcal{N}} a_i b_i\right)$$

$$\hat{\theta} = \arg\min_{\theta} ||A\theta - b|| = (A^T A)^{-1} A^T b = \left(\frac{1}{N} \sum_{i \in \mathcal{N}} a_i a_i^T\right)^{-1} \left(\frac{1}{N} \sum_{i \in \mathcal{N}} a_i b_i\right)$$



Consensus Solution:

- Initialize every node with $x_i(0) = a_i a_i^T$ and $y_i(0) = a_i b_i$
- Run any Average Consensus Algorithm
- $x_i(t) \longrightarrow \frac{1}{N} \sum_{i \in \mathcal{N}} a_i a_i^T$
- $y_i(t) \longrightarrow \frac{1}{N} \sum_{i \in \mathcal{N}} a_i b_i$
- Each node can compute then

$$\hat{\theta}_i(t) = x_i^{-1}(t) y_i(t)$$

$$\hat{\theta}_i(t) \rightarrow \left(\sum_{i \in \mathcal{N}} a_i a_i^T\right)^{-1} \left(\sum_{i \in \mathcal{N}} a_i b_i\right)$$

Conclusions

Main Message:

Many problems can be casted into a consensus problem and hence be solved distributedly

In this work

We considered 2 problems of relevance for localization and tracking:

- Offset removal
- Wireless channel least square parameter identification

Some open issues:

- What kind of problem can be casted into a consensus one?
- How to do such a casting?