L'ÉLECTROMAGNÉTISME, 150-1
UNE SCIENCE EN PLEINE ACTION !

PyLayers : Un outil open source pour la simulation de la propagation en mobilité à l'intérieur de bâtiments orienté localisation

PyLayers : An open source Indoor Propagation Simulator in Mobility for Localization Applications

Nicolas Amiot, Mohamed Laaraiedh, Bernard Uguen

IETR UMR CNRS 6164, Université de Rennes 1
nicolas.amiot@univ-rennes1.fr
mohamed.laaraiedh@univ-rennes1.fr
bernard.uguen@univ-rennes1.fr

Simulateur, Propagation, Localisation, Mobilité, Théorie des graphes
Simulator, Propagation, Localization, Mobility, Graph theory

Résumé
PyLayers is a new open source radio simulator built to tackle indoor localization problem. It has been designed to simulate dynamic scenarios including the realistic movement pedestrian agent inside a building, the transmission channel synthesis for multiple radio access technologies and the position estimation relying on location-dependent parameters originated from the simulated physical layer.
The radio channel is synthesized by using a graph-based ray tracing method especially design for mobility and conveniently reuse static geometrical information from the layout. Channel impulse response (CIR) can be synthesized, and various location dependent parameters, such as received power or time of arrival, can be deduced. The pedestrian mobility is modelled with a virtual forces approach. The simulated data can be directly used with one of the built-in localization algorithms or be exported to various standards extensions including Matlab. For further information please visit our website http://www.pylayers.org.

Introduction

If the global navigation satellite systems have now merely resolved the outdoor position estimation problems, it is still an open question for indoor environment. Consequently, many indoor positioning algorithms are regularly proposed to tackle this localization problem, but their comparison is not ease because they are generally validated on specific scenarios. One approach to would be to perform that validation on a common radio simulator which could both take into account the complexity of the encountered environment, and offer a framework for comparison.
Radio simulators [1] are not generally built to fit with localization purpose because they don't provide location dependent parameters (LDPs) as a standard output. In addition, the pedestrian mobility is generally not in the scope of those dedicated simulator, and therefore use of an external tool is required.
The aim of PyLayers is to provide a simulator fitted for indoor localization, with both a realistic mobility model and the accurate LDPs computation. The simulator is an open source project coded with Python and its standard scientific high level libraries: Numpy, Scipy, NetworkX, Matplotlib and SimPy, but it also ensures standard Matlab export.
In the following the simulator processing is described in section 1, and some examples of Pylayers outputs are described and shown in section 2.

1. A Multi-Layers architecture

PyLayers is a dynamic simulator for indoor propagation and localization based on graphs description.
The simulator is based on four layers: layout, mechanical, network and localization, which describe a specific behaviour of the simulator. From the implementation side, each layer is an independent process which can share data with the others, e.g. the positions computed at the mechanical layer can be used by the network layer.

1.1. Layout Layer

The layout layer is used to describe the simulation environment in terms of outlines, constitutive materials of walls, radio propagation. This description is made with the help several graphs. The main graphs used for the simulation are: The structure graph G_s (Figure 1): describe the outlines and the constitute materials of the walls, the graph of rooms G_r (Figure 2): describe the possible paths for pedestrian movement, the graph of interactions G_i (Figure 3): describe all the possible electromagnetic interactions between the nodes of G_s .

1.2. Mechanical Layer

The Mechanical Layer describes the movement of pedestrian agent into the layout. Usually, the movement of agents is modelled with a levy flight model or by a Markov chain [2]. This model is useful to describe mobility at a large scale, but less at a smaller scale e.g. how to move into a room avoiding the walls. In PyLayers, the mobility has been considered at those both scales, as well as the agents are able to both choose their destination and move by taking into account the layout environment.

The choice of the destination is made with the help of G_r . First, each agent randomly chose a target room. Then, the path to reach the targeted room t from the current room i is obtained with the help of G_r . By construction, rooms t and i respectively corresponds to nodes v_r^i and v_r^t of G_r .The path from i to t is obtained by computing a Djikstra shortest path algorithm between v_r^i and v_r^t .

The movement of the agent through the layout is modelled using magnetic forces. The magnetic force model aims to obtain a resulting acceleration vector a to drive the agents through the layout. Hence, agents and the layout environment are assimilated to positive poles unlike target destination is assimilated to negative poles.

Each agent is under the influence of several magnetic forces which either attracts the agent to its target node or which repulse he agent from the walls or from the other agents. According to [3], the attractive magnetic force from i to t $\mathbf{F}_{it}^k$ on agent k , is proportional to the Coulomb's law:

$$\mathbf{F}_{it}^k \propto \frac{q_k q_{it}}{\left\|\mathbf{R}_{k,it}\right\|^2}\hat{\mathbf{R}}_{k,it}$$

with q_k and q_{it} are the intensities of the magnetic load of the agent and the intermediate target respectively, $\mathbf{R}_{k,it}$ the vector from the agent k to the targeted node it . The repulsive magnetic forces $\mathbf{F}_r$ avoid agent penetrating the walls is given by:

$$\mathbf{F}_r(d_{k,w}) = \frac{\alpha}{\left\|\mathbf{d}_{k,w}\right\|^2}\hat{\mathbf{d}}_{k,w}$$

with $\mathbf{d}_{k,w}$ the distance from the agent k to a wall w and α a parameter to adjust the repulsion level. Identically, we can define a repulsion between agent by replacing $\mathbf{d}_{k,w}$ by $\mathbf{d}_{k,l}$ the distance between the two agents k and l .

Finally the acceleration vector $\mathbf{a}_k$ of agent k can be written:

$$\mathbf{a}_k = \mathbf{F}_{it}^k + \sum_{w}^{W}\mathbf{F}_r(\mathbf{d}_{k,w}) + \sum_{l\neq k}^{K}\mathbf{F}_r(\mathbf{d}_{k,l}),$$

with W the number of walls in the vicinity of agent k .

1.3. Network Layer

The network layer is described with the help of a network graph G_n . The nodes of G_n are the moving agents and static access points. In order to address multiple radio access technology (RAT), G_n is build as a multi-graph. Hence, two nodes are linked with as many edges RAT are simulated. On the edges are computed LDPs: The received power (Pr) and the time of arrival (TOA) for ultra wide band RAT.

Those LDPs are computed with an enhanced COST-231 multi-wall approach [4] which takes into account the materials and the thickness of the walls to determine both the energy of the first path and the excess time of flight due to the wall crossing. Alternatively, a more accurate LDPs estimation can be obtained with the complete ray-tracing CIR computation.

1.4. Localization Layer

In the localization layer, each agent estimate its position with the help of LDPs available on the edges of G_n . Two types of methods are implemented to estimate the position: Algebraic based methods [5] and geometric based method [6]. In algebraic methods, LDPs are used to feed maximum likelihood or weighted least square estimators and thus estimate a position. In the geometric based method, LDPs are used to build bounded regions of space called constraints. The geometrical intersection of those constraints defines a smaller bounded region which contains the sought position.

2. Simulator outputs

In order to easily exploit the simulated data, the proposed simulator provides several build-in tools, export option, and extra features. During processing, a time stamped log file can be created. It includes the true position of the agents, the estimated position of the method proposed by the localization layer, and all the LDP between all nodes on each RAT. Those valuables information can be directly exploited by included post processing tools, or can be exported into several file format including *csv* and *Matlab*.
As explained in section 1.3, LDPs can be obtained either by an enhanced multi-wall approach, or by a graph based ray-tracing tool. The multi-wall approach allows performing a fast coverage of the received power (Figure 5) and of the excess time of flight (Figure 6) of the first path. For the received power map, information of areas under the receiver sensitivity and area under the noise floor are also provided. The map of excess time of flight can be used to build a ToA bias model based on a specific layout. For both figures, the coverage of 80x40 links only takes few seconds.
In order to obtain more realistic LDPs, the use of the graph based ray-tracing tool is required. Figure 4 compares channel impulse response of measurements from [7] and the simulated data, for a given link. From that figure, it may be notice that both the first path is correctly obtained, and the most important paths due to the layout geometry are predicted, with a sufficient accuracy to be relevantly exploited by a multi-path localization model. The accuracy of the model has been compared in [8].
Finally those LDPs are used to feed the localization algorithm describe in section 1.4. Figure 7 is an example of position estimation using the provided algorithms on a simulated scenario.

3. Conclusion

This paper has presented an overview of PyLayers, an open source dynamic radio simulator for indoor localization. The simulator aims to provide realistic data in terms of mobility and radio propagation. For that purpose, a realistic mobility model based on magnetic force has been implemented, and accurate LDP are provided using either a enhanced multi-wall model or a graph-based ray-tracing. Several visualization tools and the simulated data has been presented. A further work consists in to take into account MAC layer mechanism to increase the realism in terms of network load. Finally, we invite any interested person to use, test or develop any part of the project by forking the following GitHub repository: https://github.com/pylayers/pylayers.

Références bibliographiques

1 Holt, T.; Pahlavan, K.; Lee, J.F.; , "A graphical indoor radio channel simulator using 2D ray tracing," Personal, Indoor and Mobile Radio Communications, 1992. Proceedings, PIMRC '92., Third IEEE International Symposium on , vol., no., pp.411-416, 19-21 Oct 1992

2 Nielsen, J.J., Olsen, R.L., Madsen, T.K., and Schwefel, H.: On the impact of information delay on location-based relaying: A Markov modeling approach, Wireless Communications and Networking Conference (WCNC), 2012 IEEE, volume , 3045–3050, april 2012

3- Teknomo, Kardi: Microscopic Pedestrian Flow Characteristics: Development of an Image Processing Data Collection and Simulation Model, Tohoku University, 2002

4- Damosso, E., Correia, L.M., European Commission. DGX III "Telecommunications, Information Market, and of Research.", Exploitation: COST Action 231: Digital Mobile Radio Towards Future Generation Systems : Final Report, European Commission, 1999

5- Laaraiedh, Mohamed, Yu, Lei, Avrillon, Stephane, and Uguen, Bernard: Comparison of Hybrid Localization Schemes using RSSI, TOA, and TDOA, Wireless Conference 2011 - Sustainable Wireless Technologies (European Wireless), 11th European, volume , 1–5, april 2011

6- Amiot, N., Laaraiedh, M., and Uguen, B.: Evaluation of a geometric positioning algorithm for hybrid wireless networks, Software, Telecommunications and Computer Networks (SoftCOM), 2012 20th International Conference on, volume , 1–5, sept. 2012

7- ICT-WHERE-Project, "Deliverable 4.1: Measurements of location dependent channel features," tech. rep., October 2008.

8- Uguen, B., Amiot, N., and Laaraiedh, M.: Exploiting the graph description of indoor layout for ray persistency modeling in moving channel, Antennas and Propagation (EUCAP), 2012 6th European Conference on, volume , 30–34, march 2012

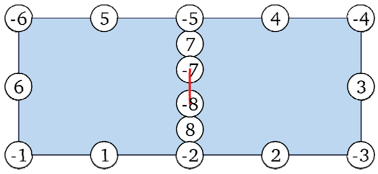

Figure 1 Structural Graph G_s

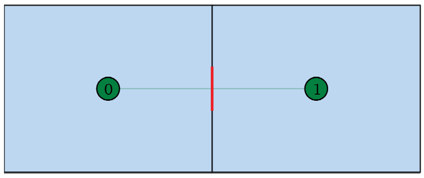

Figure 2 Graph of Room G_r

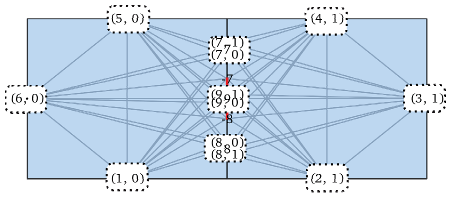

Figure 3 Interactions Graph G_i

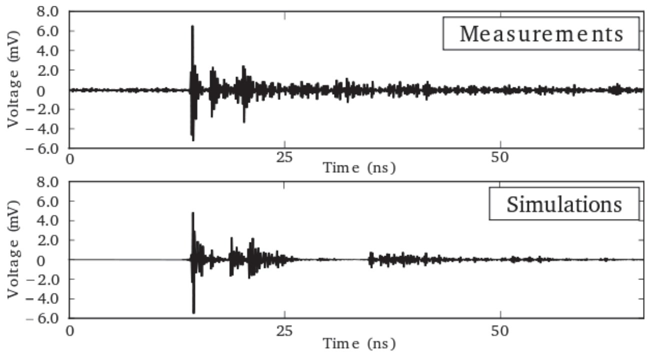

Figure 4 Example of measured and simulated CIRs.

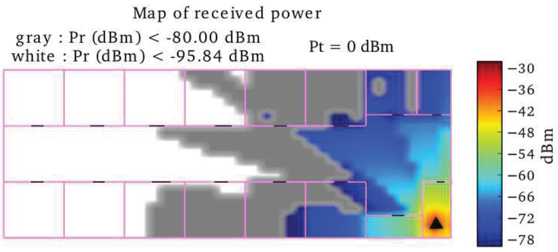

Figure 5 Map of received power. The black triangle is the transmitter position. The white area represents power in under the noise floor, the grey area represents power under receiver sensitivity, elsewhere is the received power in dBm

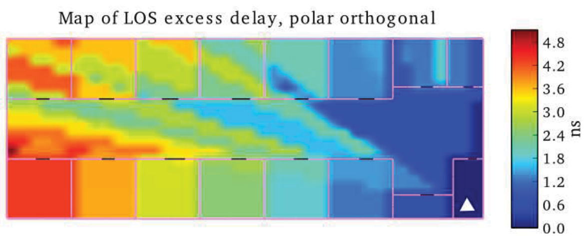

Figure 6 Map of LOS excess delay. The white triangle is the transmitter position. The degraded colors represent the delay in ns introduced by the walls in addition of the LOS delay.

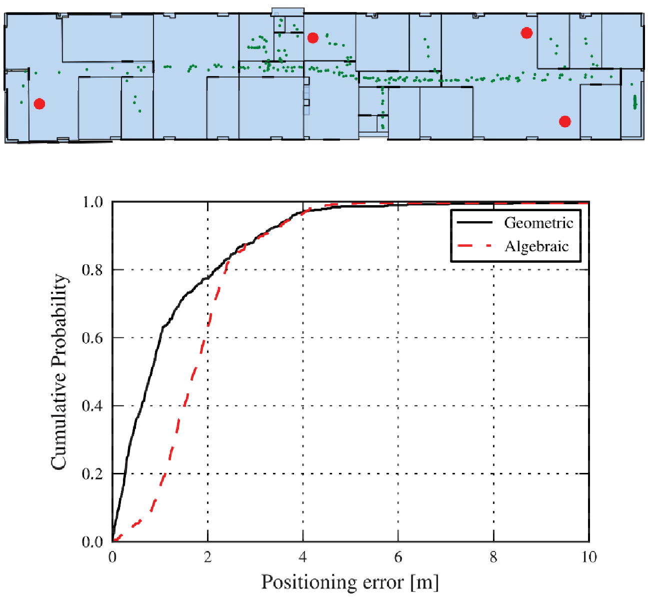

Figure 7 CDF comparison of positioning error between geometric and algebraic weighted least square method. The considered scenario is shown on the the top where green dots are the different agent position and the red dots, the 4 anchors nodes.