# **Localization in WSN**

Software Recommended: NetSim Standard v11.1(32 bit/ 64 bit), Microsoft Visual Studio 2017/2019

Follow the instructions specified in the following link to clone/download the project folder from GitHub using Visual Studio:

https://tetcos.freshdesk.com/support/solutions/articles/14000099351-how-to-clone-netsim-fileexchange-project-repositories-from-github-

Other tools such as GitHub Desktop, SVN Client, Sourcetree, Git from the command line, or any client you like to clone the Git repository.

**Note**: It is recommended not to download the project as an archive (compressed zip) to avoid incompatibility while importing workspaces into NetSim.

# Secure URL for the GitHub repository:

# https://github.com/NetSim-TETCOS/WSN\_Localization\_v11.1.git

#### Introduction:

Localization is the process of finding the physical or relative location of a sensor node as data and information are useless if the nodes have no idea of their geographical positions. GPS (global positioning system) is the simplest method for localization of nodes, but it becomes very expensive if large number of nodes exists in a given network.

#### **Anchor Nodes:**

Sensor nodes with known location information are called "Anchor nodes". Typically, anchor nodes obtain their location information by using a global positioning system (GPS), or by manually being placed at defined coordinates.

# Unknown Nodes:

Sensor nodes with unknown location information are called "Non-Anchor nodes" or "Unknown nodes". Localization is estimated through communication between localized node and unknown node for determining their geometrical placement or position. Location is determined by means of distance and angle between nodes.

# **Trilateration:**

Location of node is estimated through distance measurement from three nodes. In this concept, intersection of three circles is calculated, which gives a single point which is a position of unknown node.

Use the distance equation. If your unknown point is (x, y) and known points are  $(x_i, y_i)$  which are at distances  $r_i$  from unknown point, then you get three equations:

$$(x - x1)^{2} + (y - y1)^{2} = r1^{2}$$
$$(x - x2)^{2} + (y - y2)^{2} = r2^{2}$$
$$(x - x3)^{2} + (y - y3)^{2} = r3^{2}$$

To calculate the distance between to sensors we have used NetSim API

# DEVICE\_DISTANCE(d1,d2)

Expand out the squares and subtract the second equation from the first and third equation from second, we get

$$2(x^2 - x^1)x + 2(y^2 - y^1)y = r^{1^2} - r^{2^2} + x^{2^2} - x^{1^2} + y^{2^2} - y^{1^2}$$
$$2(x^3 - x^2)x + 2(y^3 - y^2)y = r^{2^2} - r^{3^2} + x^{3^2} - x^{2^2} + y^{3^2} - y^{2^2}$$

This is a system of two equations with two unknowns:

$$Ax + By = C$$
$$Dx + Ey = F$$

The values of x and y is obtained from the below equations:

$$x = (CE - FB) / (EA - BD)$$
$$y = (CD - AF) / (BD - AE)$$

#### Localisation in NetSim:

- 1. To implement Localisation, we have added **Localisation.c** file in Zigbee project. The file contains the following functions:
  - int fn\_NetSim\_localisation()

This function is used to find the anchor nodes based on the highest received powers received at unknown sensors from anchor nodes.

int fn\_NetSim\_trilateration\_method()

This function is used to implement the trilateration method to calculate the position / location of the unknown sensor.

bool IsUnknownNode()

This function is used to check whether the given node is unknown node or not.

bool determine\_anchor\_node()

This function is used to check whether the given node is anchor node or not.

2. Users can give their own unknown node IDs and unknown node count in **Localisation.c** file. NetSim knows all the positions of sensor nodes. Localisation is used to find the position of unknown nodes and then comparing this position with NetSim sensor positions.

802_15_4.c	Mobility.c Localisation.c 🕈 🗙
💁 ZigBee	- (Global Scope)
32	<pre>int fn_NetSim_trilateration_method(int unknown_id, double x_p, double y_p);</pre>
33	<pre>int get_device_id(int an[]); // Will return the device ID per NetSim GUI</pre>
34	
35	<pre>typedef struct stru_anchor_details *ANCHOR_DETAILS;</pre>
36	<pre>typedef struct stru_sensor_details *SENSOR_DETAILS;</pre>
37	
38	int an[100] = { 0 }; // Max number of achor nodes can be set here. Default is 100
39	
40	//Set up unknown node
41	<pre>int unknown_node_count = 2;</pre>
42	<pre>int unknown_node_IDs[10] = { 4 , 7 };</pre>
43	
44	double power = 0;
45	<pre>int unknown_id = 0;</pre>
46	<pre>double recv_power[100][100] = { 0 };</pre>
47	<pre>int neighbour_node[100] = { 0 }, temp[100] = { 0 };</pre>
48	<pre>FILE *local = NULL;</pre>

 Since the unknown nodes are mobile, we have added a call to localisation in fn\_NetSim\_Mobility\_Run() function present in mobility.c file inside Mobility project to calculate the new positions of the unknown node whenever a node moves.



#### Steps:

٠

After downloading the project folder using the GitHub URL, Open NetSim Home Page click on **Open Simulation** option,

NetSim Home			- 🗆 X
NetSim Standard Network Simulation/Emulation Platf Version 11.1.11 (32 Bit)	l		www.tetcos.com
New Simulation Ctrl+N	Choose a Network		
	Internetworks	Wireless Sensor Networks	
Open Simulation Ctrl+O	Legacy Networks	Internet Of Things	
Examples	Pure Aloha	Cognitive Radio Networks	
	Slotted Aloha	Long-Term Evolution Networks	
	Cellular Networks	LTE/LTE-A Networks	
	GSM	LTE FemtoCell	
	CDMA	LTE D2D	
	Mobile Adhoc Networks	LTE Vanet	
License Settings	Single MANET	VANET	
	Multiple MANETs		
Support	Learn	Documentation	Contact us
Answers/FAQ Contact Technical Support	Videos Experiments Manual	User Manual Technology Libraries	Email - sales@tetcos.com Phone - +91 767 605 4321
Email - support@tetcos.com		Source Code Help	

Click on Workspace options

NetSim Home								-	
NetSim Sta Network Simulation/Emu Version 11.1.11 (32 Bit)	ndard Ilation Platfo	urm						ww	W.tetcos.com
		Current workspace: Wo	rkSpace_Localizatio	on_WSN		C Experiment	t name		
New Simulation	Ctrl+N	Experiment name	C	Date modified	Network type				
Open Simulation	Ctrl+O	config_file	21	-03-2019	Wsn	V	iew Results	Export	Ŵ
Examples									
License Settings									
		Workspace options						Import Expe	eriment
Support		Learn			Documentation		Contact	us	
Answers/FAQ Contact Technical Suppo Email - support@tetcos.c	ort com	Videos Experiments	Manual		User Manual Technology Libraries Source Code Help		Email - sa Phone - +	les@tetcos.c 91 767 605 4	om 4321

# Click on More Options,

•

٠

🚺 NetSim Home					- 🗆 X
NetSim Standarc Network Simulation/Emulation Plat Version 11.1.11 (32 Bit)	form				www.tetcos.com
	Current workspace: WorkSpace_Localiz	ration_WSN		C Experiment name	
New Simulation Ctrl+N	Experiment name	Date modified	Network type		
Open Simulation Ctrl+O	config_file	21-03-2019	Wsn	View Results	Export 🔟
Examples					
License Settings					
	Open code Reset Code	e Reset Binarie	More optio	ins	Back
Support	Learn		Documentation	Contact (	IS
Answers/FAQ Contact Technical Support Email - support@tetcos.com	Videos Experiments Manual		User Manual Technology Libraries Source Code Help	Email - sa Phone - +	les@tetcos.com 91 767 605 4321

Click on **Import**, browse the extracted folder path and go into the WorkSpace\_Localization\_WSN directory. Click on the Select folder button and then on **OK**.

NetSim Home			- 🗆 X
NetSim Standard Network Simulation/Emulation Plat Version 11.1.11 (32 Bit)	d		www.tetcos.com
	Current workspace: WorkSpace_Localization	n_WSN	
New Simulation Ctrl+N	Workspac	×	
	WorkSpace Analyse the content of your fol	der or archive file to find projects and import	Export 🕅
Open Simulation Ctrl+O	WorkSpace, working Workspace. This will in	nport all the folders experiments.	Export 🔟
Examples	WorkSpace, Import from Elv11.1 F	Project/WSN Localization v11,	Export 🕅
License Settings	WorkSpace	OK Cancel	Export III
Support	Learn	Documentation	Contact us
Answers/FAQ Contact Technical Support Email - support@tetcos.com	Videos Experiments Manual	User Manual Technology Libraries Source Code Help	Email - sales@tetcos.com Phone - +91 767 605 4321

• While importing the workspace, if the following warning message indicating Software Version Mismatch is displayed, you can ignore it and proceed.

N	Warning	×
	Software Version Mismatch	1
	You are importing a workspace from 64 build of NetSim into a 32 build of NetSim. After import, reset binaries for this workspace, and then rebuild your code if you have made any changes to the source codes.	
	ОК	

Go to home page, Click on Open Simulation  $\rightarrow$  Workspace options  $\rightarrow$  Open code

•

802_15_4.c	Mobility.c Localisation.c + ×		
🔊 ZigBee	Global Scope)	Solution Explorer	* ⊡ ×
32	<pre>int fn_NetSim_trilateration_method(int unknown_id, double x_p, double y_p);</pre>	○○☆ ७·ち @ @ ◇,	= 4
33	<pre>int get device id(int an[]); // Will return the device ID per NetSim GUI</pre>	Search Solution Explorer (Ctrl+:)	- م
34		Solution 'NetSim' (2 projects)	
35	<pre>typedef struct stru_anchor_details *ANCHOR_DETAILS;</pre>	▶ Solution Account (a projectly)	
36	<pre>typedef struct stru_sensor_details *SENSOR_DETAILS;</pre>	▲ 🔄 ZigBee	
37		References	
38	<pre>int an[100] = { 0 }; // Max number of achor nodes can be set here. Default is 100</pre>	External Dependencies	
39		▶ ++ 802_15_4.c	
40	//Set up unknown node	P [6] 802_15_4.h	
41	<pre>int unknown_node_count = 2;</pre>	Agentwiddel.c	
42	<pre>int unknown_node_IDs[10] = { 4 , 7 };</pre>	b the BERCalculation c	
43		++ CCA.c	
44	double power = 0;	** ChangeRadioState.c	
45	<pre>int unknown_id = 0;</pre>	*+ ErrorPathLoss.c	
46	<pre>double recv_power[100][100] = { 0 };</pre>	A focalisation.c	
47	<pre>int neighbour_node[100] = { 0 }, temp[100] = { 0 };</pre>	*+ LocateBackoffBoundary.c	
48	<pre>FILE *local = NULL;</pre>	** Sensor.c	
49		P ** ShadowLoss.c	
50		b the SlottedCSMACA.c	
51	//Unknown Node Checking	++ Superframe.c	
52	Ebool IsUnknownNode(NETSIM ID devId)	++ UnslottedCSMACA.c	
53	{		

• Right click on Solution in Solution Explorer and select rebuild solution.



• Upon rebuilding, **libZigbee.dll** and **libMobility.dll** will automatically get updated in NetSim binary folder.

**Note:** Based on whether you are using NetSim 32 bit or 64 bit setup you can configure Visual studio to build 32 bit or 64 bit DII files respectively as shown below:



 Go to NetSim home page, click on Open Simulation, Click on Localization\_in\_WSN\_Example.

🚺 NetSim Home						
NetSim St Network Simulation/E Version 11.1.11 (32 Bi	andard	<b>j</b> orm				www.tetcos.com
		Current workspace: WorkSpace_Lo	calization_WSN		C Experiment name	
New Simulation	Ctrl+N	Experiment name	Date modified	Network type		
Open Simulation	Ctrl+O	Localization_in_WSN_Example	22-03-2019	Wsn	View Results	Export 🔟
Examples License Settings						
Current		Workspace options		Desurveitation	Control of	Import Experiment
Support Answers/FAQ Contact Technical Sup Email - support@tetco	oport os.com	Learn Videos Experiments Manual		Documentation User Manual Technology Libraries Source Code Help	Contact Email - sa Phone	JS les@tetcos.com -91 767 605 4321

- Run simulation after the network scenario gets loaded.
- After simulation, localisation.txt file will get created in the **bin** folder of NetSim. Right click on the NetSim shortcut icon in your desktop and select Open file location to go to NetSim bin folder. The localisation.txt file logs the unknown node IDs, received powers from all anchor nodes to unknown nodes, anchor node IDs based on highest received powers and the position or coordinates of the unknown nodes with variation in time as shown below.

```
Iocalisation - Notepad
File Edit Format View Help
From 1 to 7 is: -74.8270943312 dbm
From 2 to 7 is: -79.6770789223 dbm
From 3 to 7 is: -78.1200666900 dbm
From 5 to 7 is: -82.7459227414 dbm
From 6 to 7 is: -84.5704176688 dbm
From 8 to 7 is: -86.3642750408 dbm
From 9 to 7 is: -77.0410805355 dbm
Unknown node = 4
Anchor nodes = 1, 2, 9,
The position of Unknown node 4 at time 23000000.000000\mu s = 79, 115
Unknown node = 7
Anchor nodes = 1, 9, 3,
The position of Unknown node 7 at time 23000000.000000\mus = 162, 116
Unknown nodes
4
7
Received powers
From 1 to 4 is: -69.9012340460 dbm
From 2 to 4 is: -78.7497043392 dbm
From 3 to 4 is: -79.6973643345 dbm
From 5 to 4 is: -83.1560378879 dbm
From 6 to 4 is: -85.3201536937 dbm
From 8 to 4 is: -87.1334200371 dbm
From 9 to 4 is: -79.4031897975 dbm
From 1 to 7 is: -78.8012301739 dbm
From 2 to 7 is: -75.7863125093 dbm
From 3 to 7 is: -69.0526329446 dbm
From 5 to 7 is: -75.9135394712 dbm
From 6 to 7 is: -80.3219905678 dbm
From 8 to 7 is: -83.6881019067 dbm
From 9 to 7 is: -81.8633293994 dbm
Unknown node = 4
Anchor nodes = 1, 2, 9,
The position of Unknown node 4 at time 24000000.000000\mu s = 79, 115
Unknown node = 7
Anchor nodes = 3, 2, 5,
The position of Unknown node 7 at time 24000000.000000\mus = 162, 116
```