UWB 802.15.4 Toolbox MATLAB

Codes provided by MATWORKS

- Channels mode : link
- Localization : link
- Ranging : link
- Waveform generator : link
- Recovery Signals : link
- End-to-End Simulation of HRP : link

Channels mode

Environment Parameterization

Propagation and Path Loss :

- Transmitted Signal
- Distance Dependence
- Frequency Dependence
- Antenna Effects

Power Delay Profile

- Clusterization
- Path Modeling

Small-Scale Fading

Environment Comparison

Channels mode

- How to define and simulate a uwb channel :
  - Creation of a channel environment in LOS or N-LOS with effects of multipath propagation
  - Set it up :
    - frequency band of the UWB signal
    - number and nature of obstacles
    - propagation parameters such as surface reflection coefficients
    - adjust multipath density and attenuation models
- How to model the effects of multipath propagation :
  - where signals reflect off surrounding objects,
  - creating several signal paths that arrive at the receiver at different times.

Channels mode

```
uwbChannelConfig with properties:

                       Type: 'Industrial'
                     HasLOS: 1
          ReferencePathLoss: 56.7000
            PathLossExponent: 1.2000
         ShadowingDeviation: 6
                AntennaLoss: 3
          FrequencyExponent: -1.1030
        AverageNumClusters: 4.7500
         ClusterArrivalRate: 0.0709
  ClusterEnergyDecayConstant: 13.4700
             PathDecaySlope: 0.9260
            PathDecayOffset: 0.6510
   ClusterShadowingDeviation: 4.3200
          NakagamiMeanOffset: 0.3600
          NakagamiMeanSlope: 0
     NakagamiDeviationOffset: 1.1300
      NakagamiDeviationSlope: 0
           FirstPathNakagami: 12.9900
```

```
function obj = setupResidentialEnvironment(obj)
  if obj.HasLOS
    obj.ReferencePathLoss            = 43.9;
    obj.PathLossExponent             = 1.79;
    obj.ShadowingDeviation           = 2.22;
    obj.AntennaLoss                  = 3;
    obj.FrequencyExponent            = 1.12;
    obj.AverageNumClusters           = 3;
    obj.ClusterArrivalRate           = 0.047;
    obj.PathArrivalRate1             = 1.54;
    obj.PathArrivalRate2             = 0.15;
    obj.MixtureProbability           = 0.095;
    obj.ClusterEnergyDecayConstant   = 22.61;
    obj.PathDecaySlope               = 0;
    obj.PathDecayOffset              = 12
    obj.ClusterShadowingDeviation    = 2.
    obj.PDPIncreaseFactor            = na
    obj.PDPDecayFactor               = na
    obj.FirstPathAttenuation         = na
    obj.NakagamiMeanOffset           = 0.
    obj.NakagamiMeanSlope            = 0;
    obj.NakagamiDeviationOffset      = 0.
    obj.NakagamiDeviationSlope       = 0;
    obj.FirstPathNakagami            = na
```

```
else % NLOS
  obj.ReferencePathLoss            = 48.7;
  obj.PathLossExponent             = 4.58;
  obj.ShadowingDeviation           = 3.51;
  obj.AntennaLoss                  = 3;
  obj.FrequencyExponent            = 1.53;
  obj.AverageNumClusters           = 3.5;
  obj.ClusterArrivalRate           = 0.12;
  obj.PathArrivalRate1             = 1.77;
  obj.PathArrivalRate2             = 0.15;
  obj.MixtureProbability           = 0.045;
  obj.ClusterEnergyDecayConstant   = 26.27;
  obj.PathDecaySlope               = 0;
  obj.PathDecayOffset              = 17.5;
  obj.ClusterShadowingDeviation    = 2.93;
  obj.PDPIncreaseFactor            = nan;
  obj.PDPDecayFactor               = nan;
  obj.FirstPathAttenuation         = nan;
  obj.NakagamiMeanOffset           = 0.69;
  obj.NakagamiMeanSlope            = 0;
  obj.NakagamiDeviationOffset      = 0.32;
  obj.NakagamiDeviationSlope       = 0;
  obj.FirstPathNakagami            = nan;
end
```

Localization

One-Way Ranging / Time-Difference of Arrival (OWR/TDOA)

- Configure Network
- Configure Blinks / MAC and PHY layers

Dertermine hyperbolic surfaces and intersections for each nodes

Calculate localization error for each answer

Localization : Chronological Operation of the Code

- **Simulation Configuration**:
  - **numDevices** and **numNodes**: Define the number of devices and nodes.
  - **deviceLoc** and **nodeLoc**: Coordinates of the devices and nodes on a 100x100 plane.
- **Calculating Distances and Time of Flight (TOF)**:
  - Distances between each node and each device are calculated.
  - Time of flight (TOF) is calculated by dividing the distances by the speed of light.
- **Generating the Transmission Signal**:
  - A simple pulse signal is created, and the calculated delays are applied to simulate the signal reception at each node.
- **Reading Data**:
  - The code reads a data file to extract the initiator address.
- **Configuring MAC and PHY Frames**:
  - MAC frames are configured with appropriate parameters.
  - The PHY waveform is generated using the MAC frame configurations.
- **Simulating Propagation and Preamble Detection**:
  - Each node applies the propagation delay to the received signal.
  - The position of the preambles is detected for each node, and arrival times (TDOA) are calculated.
- **Calculating Hyperbolic Surfaces and Intersections**:
  - For each pair of nodes, the time difference of arrival is used to calculate hyperbolic surfaces.
  - The intersections of these surfaces provide position estimates for the device.
- **Displaying Results**:
  - Estimated positions are plotted to visualize intersections, and localization errors are calculated.

Localization

```
%==============================================================================
% CONFIGURATION
% Numbers
numDevices = 2;
numNodes = 3;
% Initiator(s)
deviceLoc = [50 50;
             25 25;
             98 71];  % place devices at given locations
% Receptor(s)
nodeLoc = [40 41;
           62 83;
           87 24;
           10 12;
           21 21;
           45 29;
           84 35];
TDOA = nan(numNodes);
helperShowLocations(deviceLoc, nodeLoc);
```

Calculate the actual distance and time of flight (TOF) between nodes and the device.

```
actualDistances = pdist2(nodeLoc, deviceLoc)
```

```
actualDistances = 7×3

   13.4536   21.9317   65.2993
   35.1141   68.7968   37.9473
   45.2217   62.0081   48.2701
   55.1725   19.8494  105.9481
   41.0122    5.6569   91.8096
   21.5870   20.3961   67.6240
   37.1618   59.8415   38.6264
```

```
c = physconst('LightSpeed')% speed of light (m/s)
```

```
c = 299792458
```

```
actualTOF = actualDistances/c
```

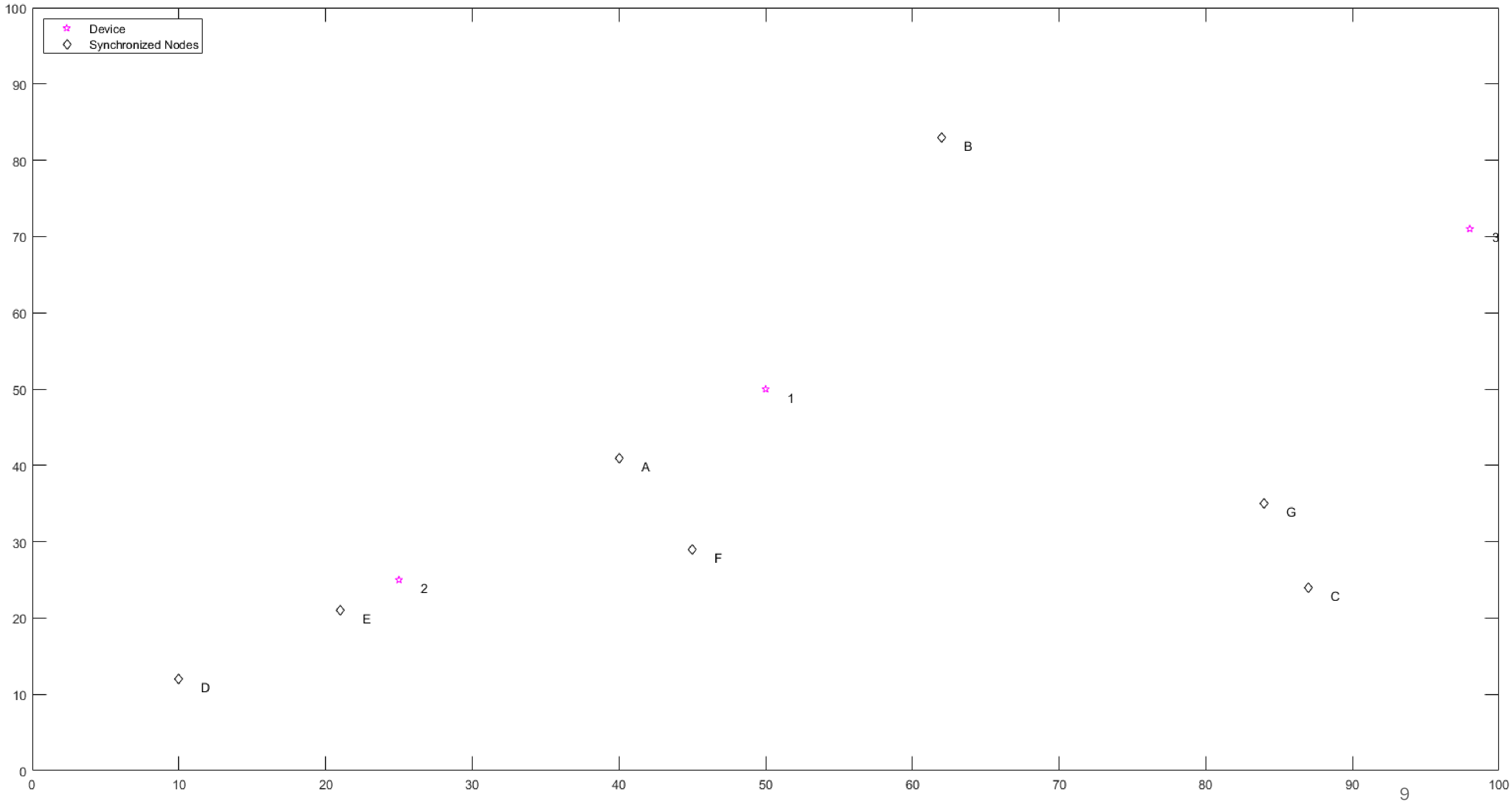
```
actualTOF = 7×3
10^-6 ×
    0.0449    0.0732    0.2178
    0.1171    0.2295    0.1266
    0.1508    0.2068    0.1610
    0.1840    0.0662    0.3534
    0.1368    0.0189    0.3062
    0.0720    0.0680    0.2256
    0.1240    0.1996    0.1288
```

```
sampleRate = 1e9;

samplesToDelay = actualTOF * sampleRate
```

```
samplesToDelay = 7×3

   44.8765   73.1563  217.8151
  117.1280  229.4814  126.5787
  150.8433  206.8366  161.0116
  184.0355   66.2106  353.4048
  136.8020   18.8692  306.2438
   72.0066   68.0340  225.5693
  123.9584  199.6096  128.8439
```


Device
Synchronized Nodes
A
B
C
D
E
F
G
1
2
3
0
10
20
30
40
50
60
70
80
90
100

Localization : Configure Blinks / MAC and PHY layers

```
numBlinks = 1; % like a pulse

% MAC layer:
payload = '00';
cfg = lrwpan.MACFrameConfig( ...
    FrameType='Data', ...
    DestinationAddressing='Short address', ...
    SourceAddressing='Short address', ...
    SourcePANIdentifier='ABCD', ...
    SourceAddress=initiatorAddr)
```

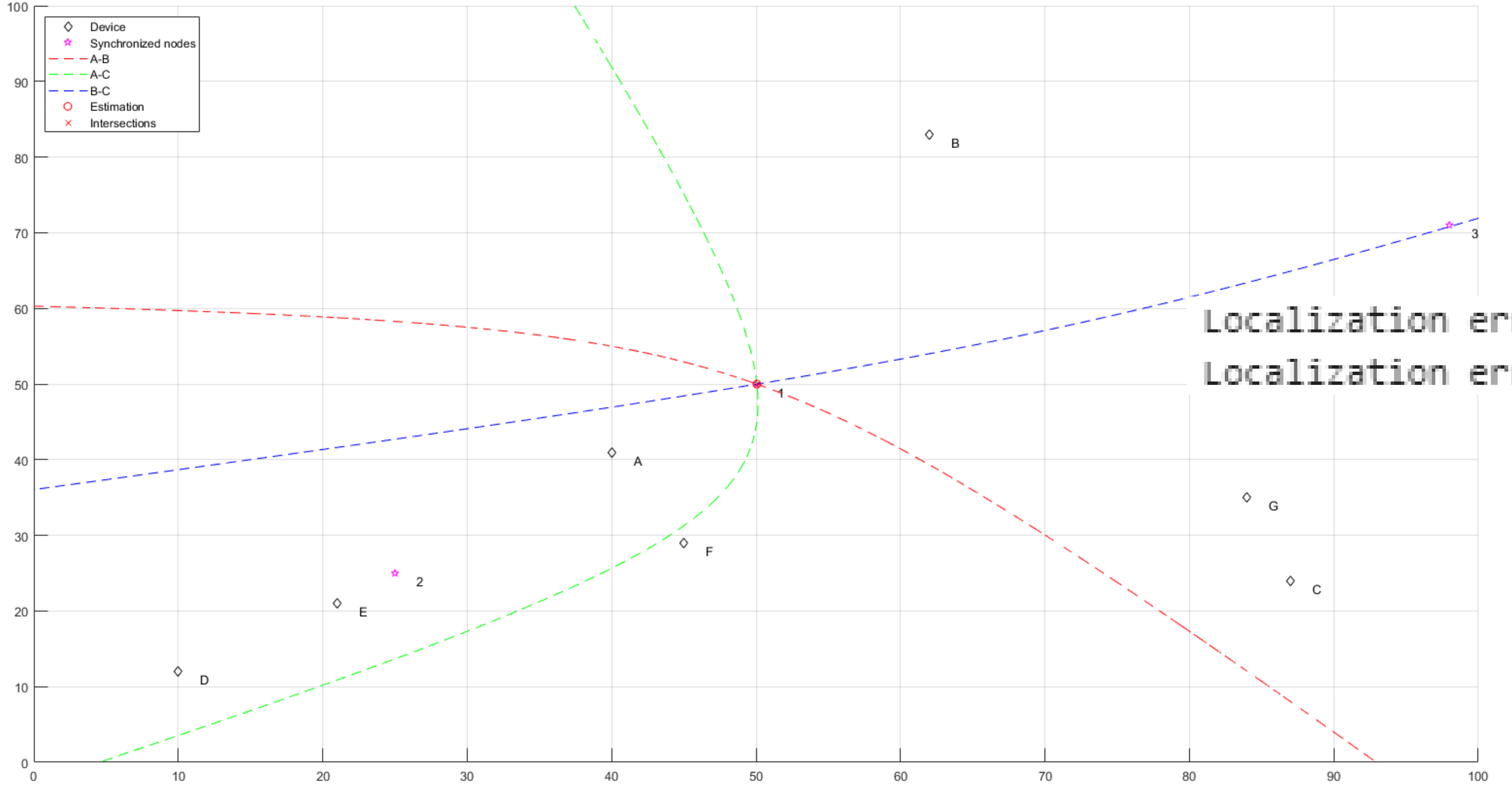
MAC Config

```
% PHY layer:
% Ensure the Ranging field is enabled.
% Also set the proper PSDU length.
blinkPHYConfig = lrwpanHRPConfig( ...
    Mode='HPRF', ...
    STSPacketConfiguration=1, ... % TO CHECK
    PSDULength=length(blinkMAC)/8, ...
    Ranging=true);
blinkPHY = lrwpanWaveformGenerator( ...
    blinkMAC, ...
    blinkPHYConfig);

% cfg4a = lrwpanHRPConfig( ...
%     Mode='802.15.4a', ...
%     Channel=9, ...            % High-band mandatory chan for code index 3
%     MeanPRF=15.6, ...         % 8 candidate bursts
%     DataRate=27.24, ...       % 1 chip per burst (PHR at 850 kbps max)
%     CodeIndex=3, ...          % 3rd code with length 31
%     STSPacketConfiguration=1, ...
%     Ranging=true,...
%     PreambleMeanPRF=4.03, ... % PreambleSpreadingFactor = 64
%     PSDULength=length(psdu)/8);
%
% wave4a = lrwpanWaveformGenerator( ...
%     psdu, ...
%     cfg4a);
```

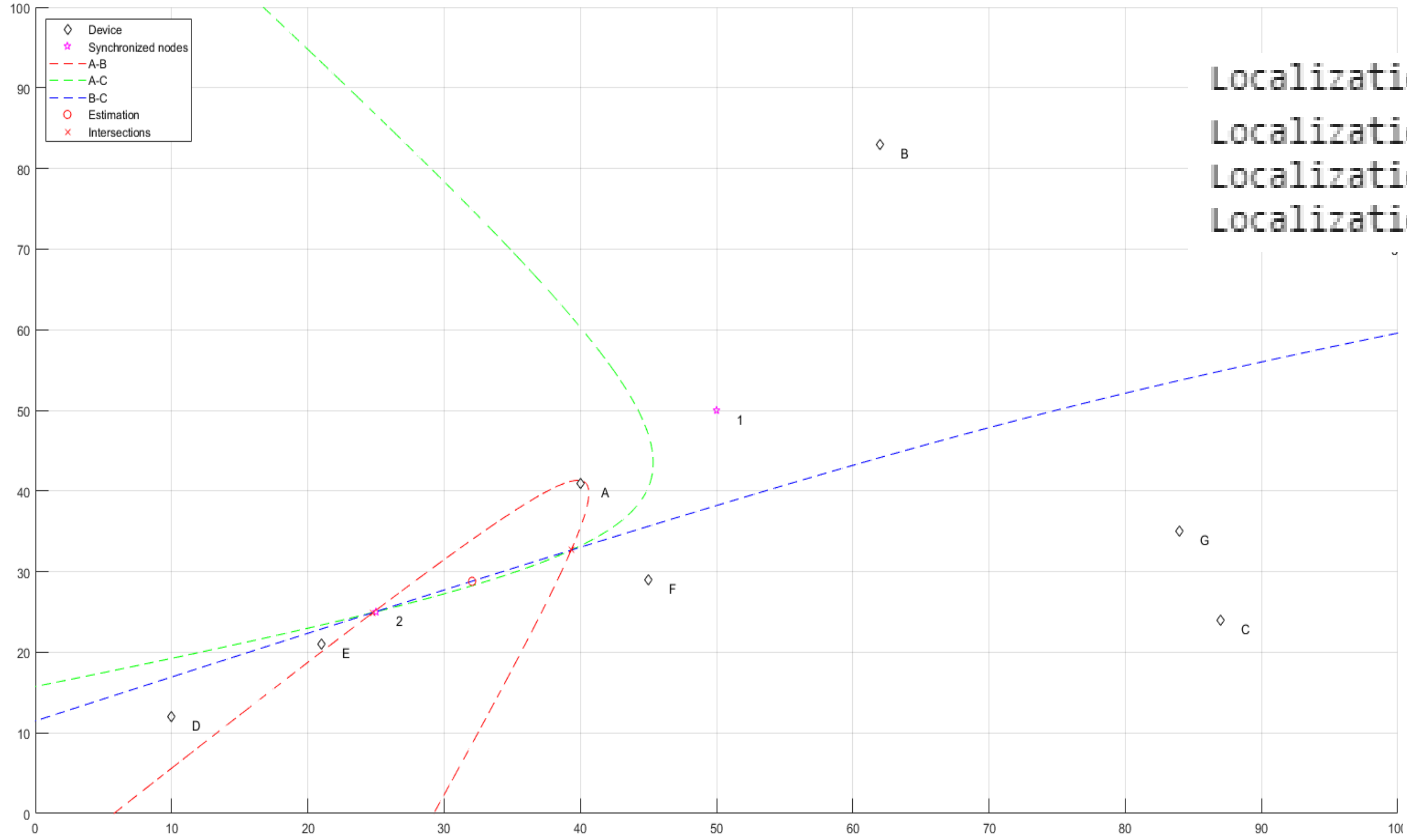
PHY Config

Localization : TDOA Results



Localization error for device 1:
Localization error #1 = 0.060 m.

Localization : TDOA Results



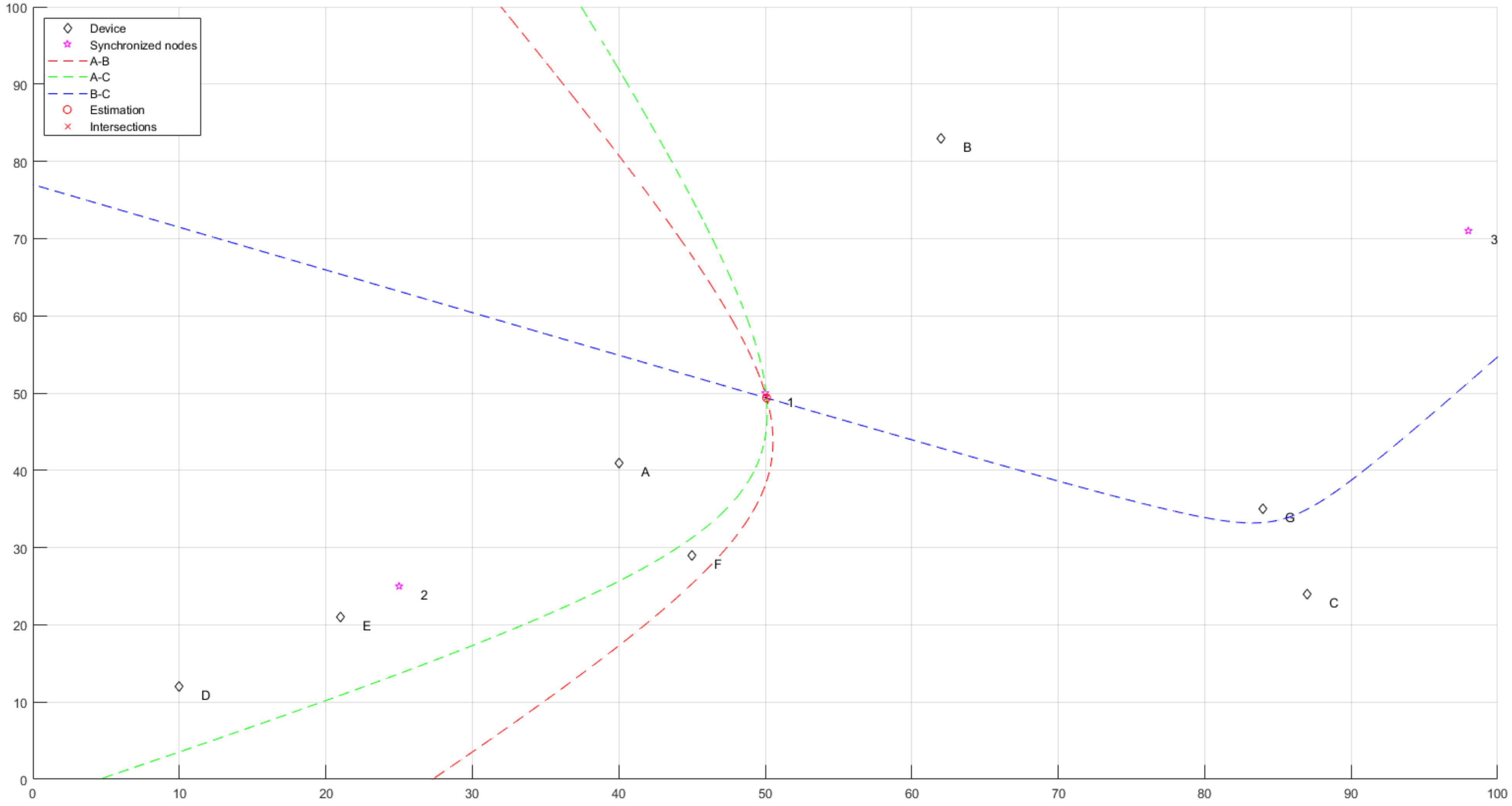
Localization error for device 2:

Localization error #1 = 0.226 m.

Localization error #2 = 16.249 m.

Localization error #3 = 0.226 m.

Localization : TDOA Results



Ranging

Single-Sided Two-Way Ranging (SS-TWR)

Transmitted Frame

- Transmission from Initiator
- Wireless channel : filter the transmission frame through an AWGN channel and add propagation delay
- Reception at Responder

Response Frame

- Transmission from Responder
- Wireless Channel
- Reception at Initiator

Range Estimation