**LTE X2 Handover**

**Software Recommended:** NetSim Standard v12.0 (32bit/64bit), Microsoft Visual Studio 2017/2019

**Handover**

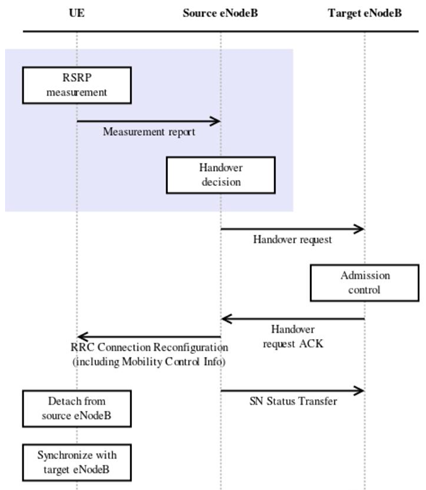
Handover is an important function that maintains seamless connectivity when transitioning  
from one base station to another.

* Due to mobility UEs can move from one place to another.
* Then UE sends the MEASUREMENT REPORT to the S-eNB.
* The S-eNB issues a HANDOVER REQUEST message to the T-eNB.
* The T-eNB checks for resource availability and, if available, reserves the resources  
  and sends back the HANDOVER REQUEST ACKNOWLEDGE message.

**LTE X2 Handover**

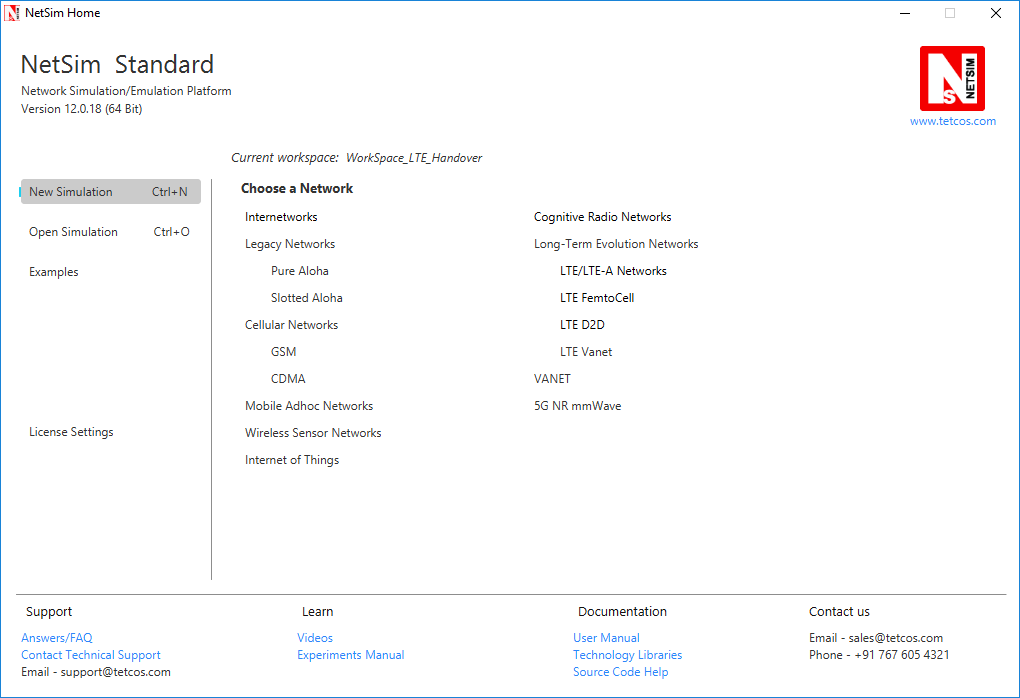
In event-triggered handover procedures, each UE evaluates the Event condition every time a new averaged measurement sample is available .The evaluated condition is the entering condition of Event whether the RSRP/SNR measured from a neighbouring cell becomes an offset better than the RSRP/SNR measured from the serving cell . The offset is represented as hysteresis.

The UE generates a measurement report and transmit it as an RRC message to the serving cell. This report typically contains measurement results of at least the serving cell, but is extendable with measurement results of neighboring cells. Series of steps occurs to carry out handover process which can be seen in the diagram below and also in our table in Result Section.

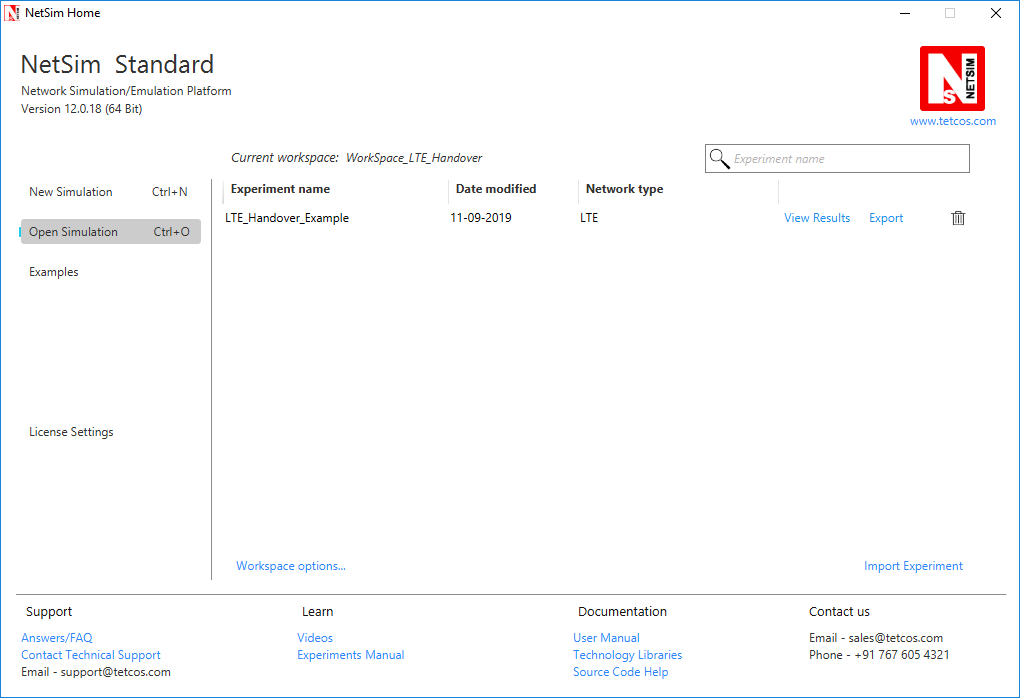


**Code Changes done to obtain SNR logs in NetSim:**

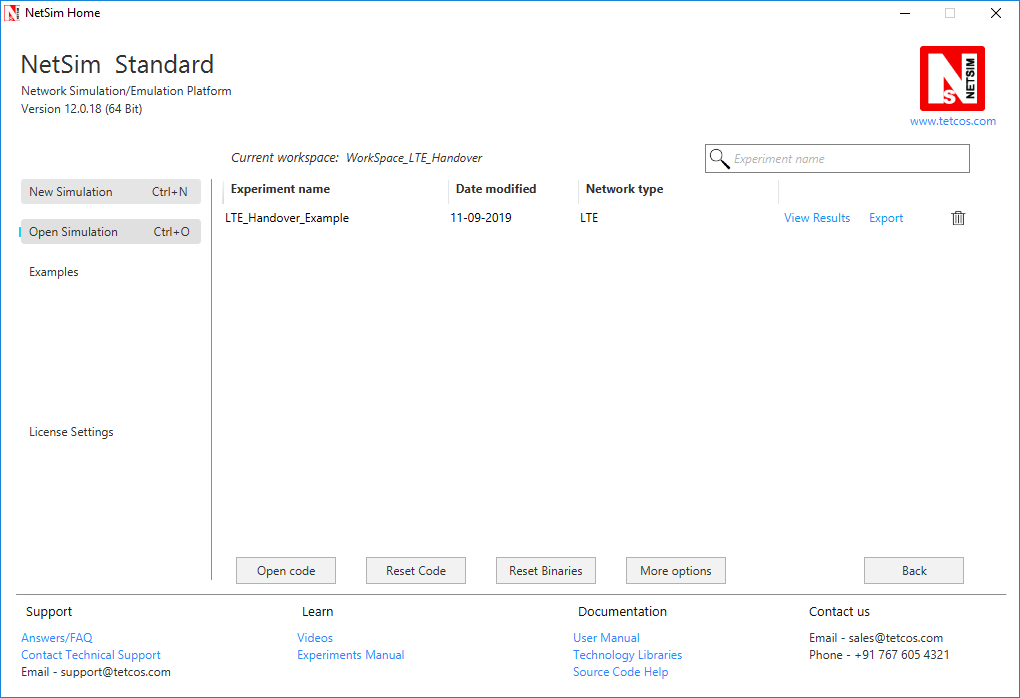
1. In the LTE Project of NetSim source codes modifications are done to perform LTE X2 Handover.
2. After you unzip the downloaded project folder. Open NetSim Home Page click on **Open Simulation** option,



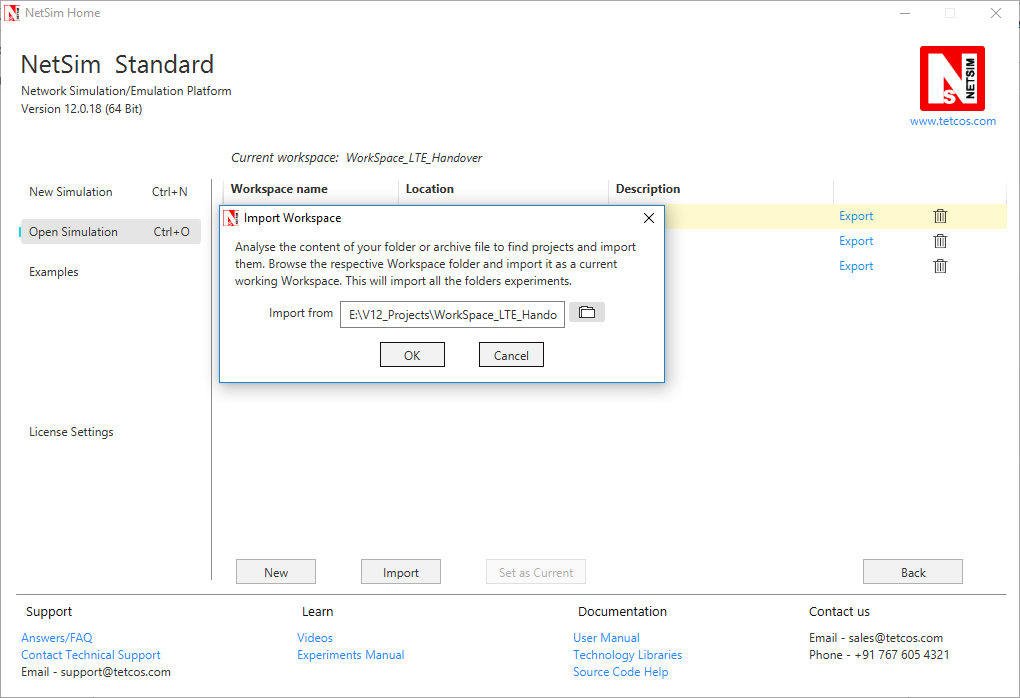
1. Click on **Workspace options**



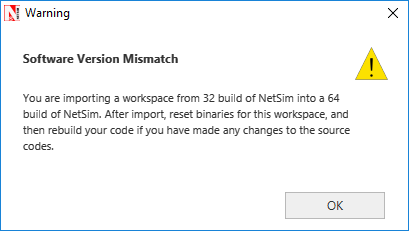
1. Click on **More Options,**



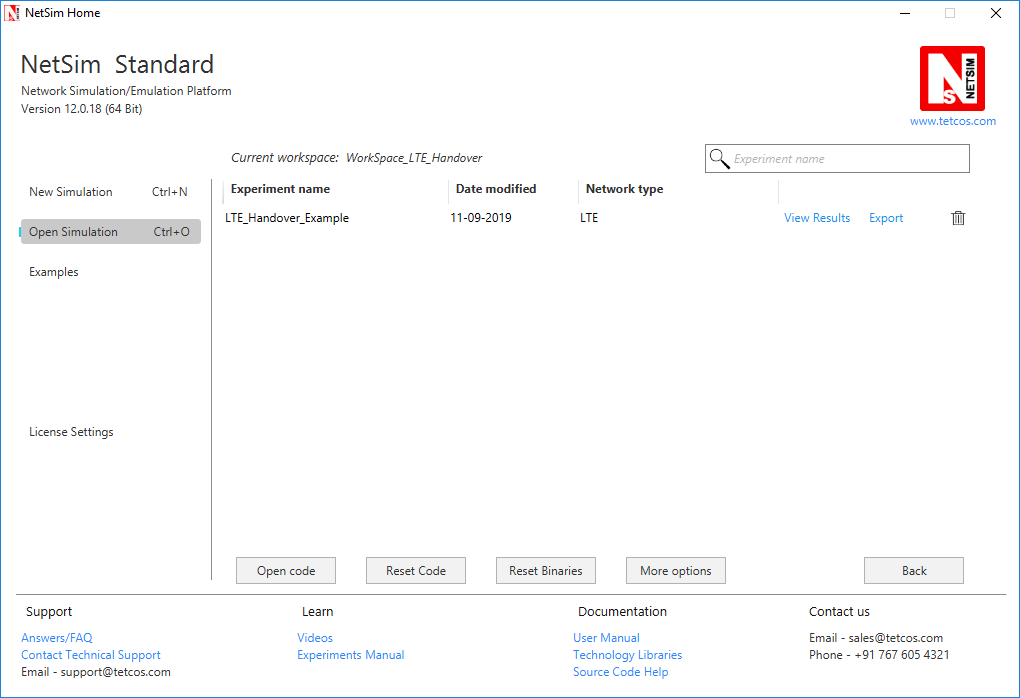
1. Click on **Import**, browse the extracted folder path and go into WorkSpace\_LTE\_Handover folder. Click on Select Folder and then on **OK.**



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



1. Go to home page, Click on **Open Simulation** 🡪 **Workspace options** 🡪 **Open code**



1. In file LTE Phy.c the following changes(highlighted in red) were made:

**#include "main.h"**

**#include "LTE.h"**

**#define devid(id) fn\_NetSim\_GetDeviceIdByConfigId(id)**

**NETSIM\_ID fn\_NetSim\_LTE\_FindNearesteNB(NETSIM\_ID nDeviceId);**

**FILE\* fp;**

**char snrlog[100];**

**int fn\_NetSim\_LTE\_CalculateReceivedPower()**

**{**

**NETSIM\_ID i;**

**//fp = fopen("LTE\_UE\_SNR.txt", "w+");**

**for(i=0;i<NETWORK->nDeviceCount;i++)**

**{**

**…………………………………………………………**

**if(ber<TARGET\_BER)**

**break;**

**else**

**{**

**info->DLInfo[j].nCQIIndex--;**

**info->ULInfo[j].nCQIIndex--;**

**}**

**}**

**sprintf(snrlog, "LTE\_UE\_SNR\_%d.csv", devid(info->nUEId));**

**fp = fopen(snrlog, "w+");**

**int arr[512],k=0,c;**

**if (fp)**

**{**

**for (c = 0; c < NETWORK->nDeviceCount; c++)**

**{**

**if (NETWORK->ppstruDeviceList[c]->nDeviceType == eNB)**

**{**

**arr[k] = NETWORK->ppstruDeviceList[c]->nDeviceId;**

**k++;**

**}**

**}**

**fprintf(fp, "UE\_ID,Time");**

**for (c = 0; c < k; c++)**

**{**

**fprintf(fp, ",SNR\_ENB\_ID\_%d",arr[c]);**

**}**

**fclose(fp);**

**}**

}

**info=(LTE\_ASSOCIATEUE\_INFO\*)LIST\_NEXT(info);**

**}**

**}**

**}**

**return 1;**

**}**

1. Now in NAS.c change the initial function as following

**#define MEASUREMENT\_REPORT\_SIZE 184/8.0**

**#define HO\_REQUEST\_SIZE 288/8.0**

**#define HO\_CONFIRM\_SIZE 112/8.0**

**#define HANDOVER\_DIFF      3 //db**

**#define devid(id) fn\_NetSim\_GetDeviceIdByConfigId(id)**

**int fn\_NetSim\_LTE\_InitHandover(NETSIM\_ID ueId,NETSIM\_ID nENBId)**

**{**

**//Prepare the measurement report**

**NetSim\_PACKET\* packet;**

**LTE\_MAC\_PACKET\* macPacket;**

**LTE\_PHY\_PACKET\* phyPacket;**

**LTE\_MEASUREMENT\_REPORT\* report=NULL;**

**NETSIM\_ID i;**

**FILE\* fp = NULL;**

**char snrlog[100];**

**sprintf(snrlog, "LTE\_UE\_SNR\_%d.csv", fn\_NetSim\_GetDeviceIdByConfigId(ueId));**

**fp = fopen(snrlog, "a+");**

**fprintf(fp, "\n%d,%lf", fn\_NetSim\_GetDeviceIdByConfigId(ueId), pstruEventDetails->dEventTime);**

**for(i=0;i<NETWORK->nDeviceCount;i++)**

**{**

**//fprintf(fp, "\n%d,%lf", devid(info->nUEId), pstruEventDetails->dEventTime);**

**if(DEVICE\_TYPE(i+1) == eNB)**

**{**

**unsigned int j;**

**LTE\_MEASUREMENT\_REPORT\* temp=MEASUREMENT\_REPORT\_ALLOC();**

**LTE\_ASSOCIATEUE\_INFO\* info = UEINFO\_ALLOC();**

**LTE\_ENB\_PHY\* enbPhy=(LTE\_ENB\_PHY\*)DEVICE\_PHYVAR(i+1,1);**

**info->nUEId=ueId;**

**info->nUEInterface=1;**

**temp->nENBId=i+1;**

**temp->nUEId=ueId;**

**temp->carrier\_count = enbPhy->ca\_count;**

**for(j=0;j<enbPhy->ca\_count;j++)**

**{**

**fn\_NetSim\_LTE\_CalculateRxPower(i+1,1,info,j);**

**fn\_NetSim\_LTE\_CalculateSNR(i+1,1,info,j);**

**fn\_NetSim\_LTE\_GetCQIIndex(i+1,1,info,j);**

**fn\_NetSim\_LTE\_GetMCS\_TBS\_Index(info,j);**

**while(info->DLInfo[j].nCQIIndex>1 && info->ULInfo[j].nCQIIndex>1)**

**{**

**double ber;**

**fn\_NetSim\_LTE\_GetMCS\_TBS\_Index(info,j);**

**ber = fn\_NetSim\_LTE\_CalculateBER(0,info->DLInfo[j].MCSIndex,info->DLInfo[j].dSNR);**

**if(ber<TARGET\_BER)**

**break;**

**else**

**{**

**info->DLInfo[j].nCQIIndex--;**

**info->ULInfo[j].nCQIIndex--;**

**}**

**}**

**temp->nCQIIndex\_DL[j]=info->DLInfo[j].nCQIIndex;**

**temp->dSNR\_DL[j]=info->DLInfo[j].dSNR;**

**//dETime = pstruEventDetails->dEventTime;**

**if (fp)**

**{**

**if (j == 0)**

**{**

**fprintf(fp, ",%lf", info->DLInfo[j].dSNR);**

**}**

**}**

**}**

**LIST\_FREE((void\*\*)&info,info);**

**LIST\_ADD\_LAST((void\*\*)&report,temp);**

**}**

**}**

**fclose(fp);**

**if(report)**

**{**

**packet=fn\_NetSim\_LTE\_CreateCtrlPacket(pstruEventDetails->dEventTime,**

**LTEPacket\_MeasurementReport,**

**nENBId,**

**ueId,**

**nENBId,**

**MEASUREMENT\_REPORT\_SIZE);**

**macPacket=calloc(1,sizeof\* macPacket);**

**macPacket->logicalChannel=LogicalChannel\_CCCH;**

**macPacket->MessageType=LTEPacket\_MeasurementReport;**

**macPacket->MessageVar=report;**

**macPacket->transportChannel=TransportChannel\_RACH;**

**phyPacket=PACKET\_PHYPROTOCOLDATA(packet);**

**phyPacket->physicalChannel=PhysicalChannel\_PRACH;**

**packet->pstruMacData->Packet\_MACProtocol=macPacket;**

**packet->pstruPhyData->Packet\_PhyData=phyPacket;**

**//Add physical out event**

**pstruEventDetails->nDeviceId=ueId;**

**pstruEventDetails->nDeviceType=UE;**

**pstruEventDetails->nInterfaceId=1;**

**pstruEventDetails->nProtocolId=MAC\_PROTOCOL\_LTE;**

**pstruEventDetails->dPacketSize=MEASUREMENT\_REPORT\_SIZE;**

**pstruEventDetails->nApplicationId=0;**

**pstruEventDetails->nEventType=PHYSICAL\_OUT\_EVENT;**

**pstruEventDetails->nPacketId=0;**

**pstruEventDetails->nSegmentId=0;**

**pstruEventDetails->nSubEventType=0;**

**pstruEventDetails->pPacket=packet;**

**pstruEventDetails->szOtherDetails=NULL;**

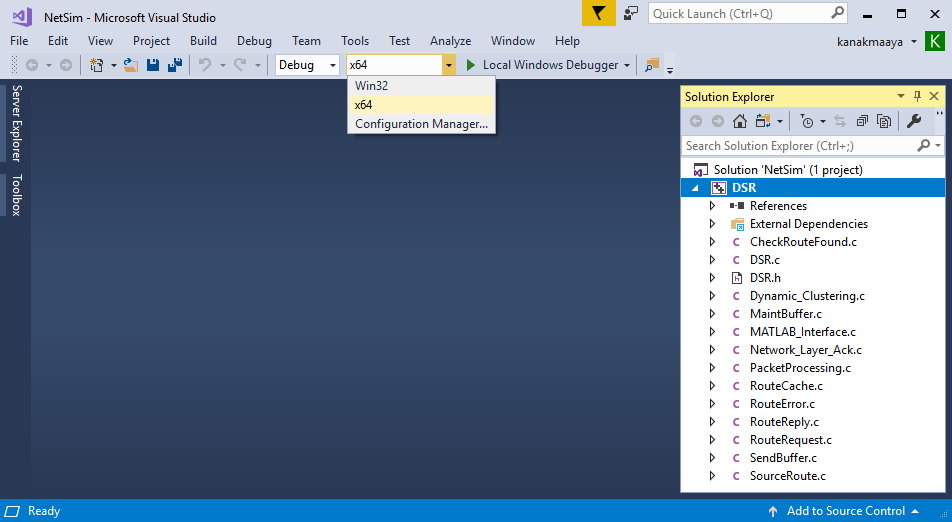
**fnpAddEvent(pstruEventDetails);**

**}**

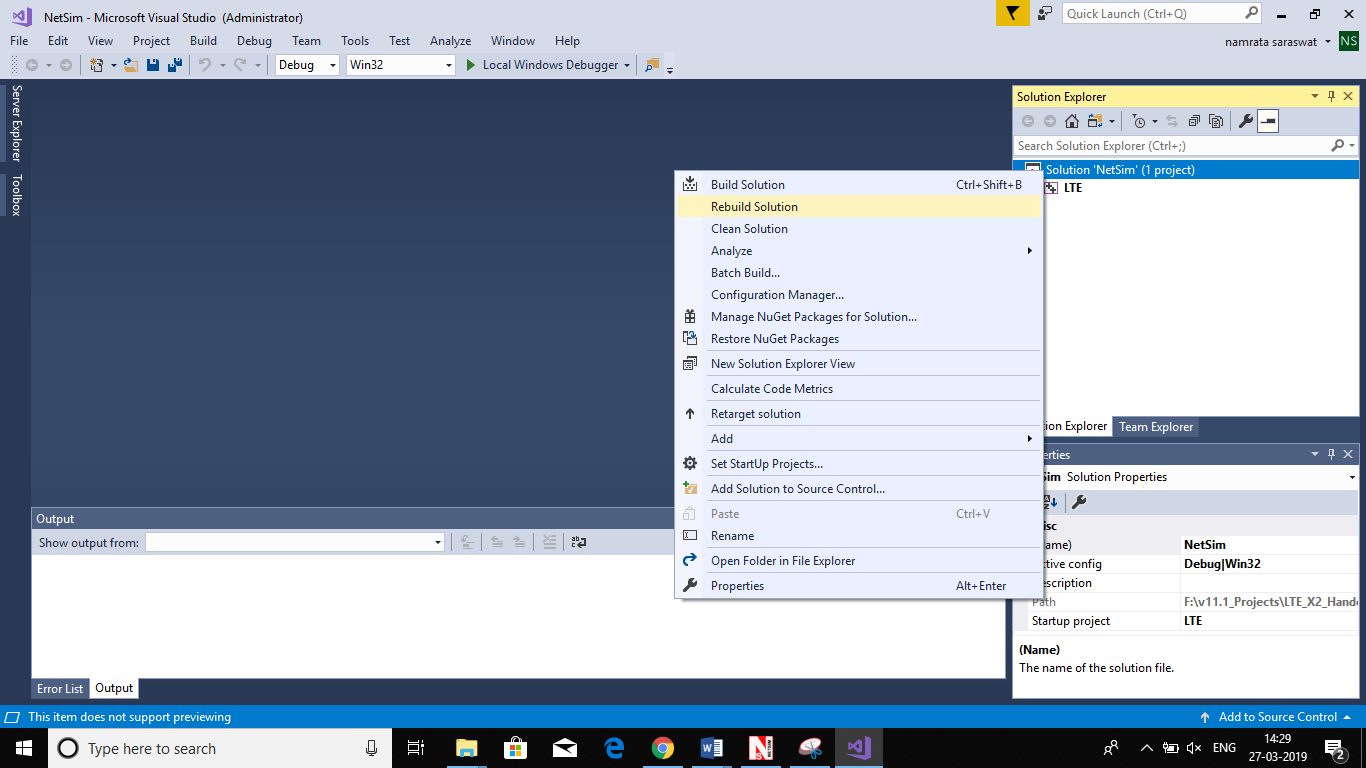
**return 0;**

**}**

1. 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 Dll files respectively as shown below:



1. Right click on the LTE project in the solution explorer and select rebuild. Upon rebuilding, lib**LTE.dll** will automatically get updated in the respective bin folders of the current workspace.



**Steps to be done in NetSim scenario to Create LTE X2 Handover**

**Configuration**

Grid Length: 5000m

Distance between ENB: 5Km

Distance between UE: 5Km

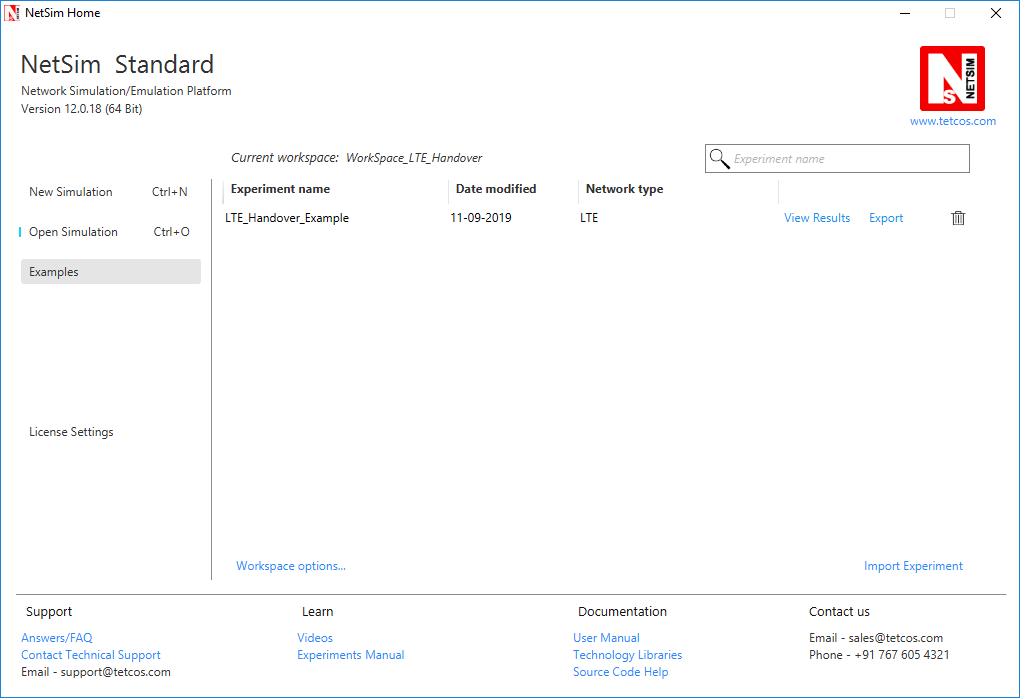
Properties UE-ENB Link: Default

Simulation Time = 100 Sec

Mobility model = File based mobility for UE6, No Mobility for UE 7

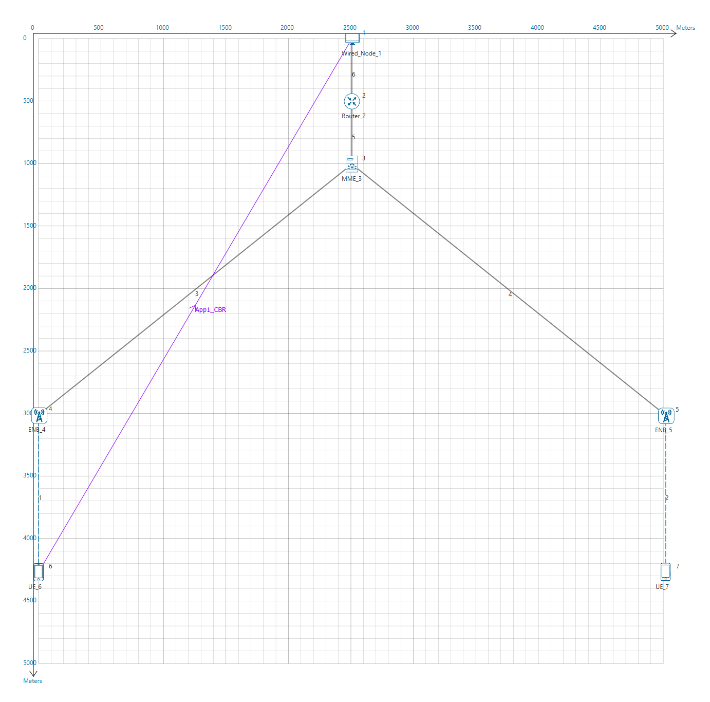
A sample **Configuration.netsim** file is provided, with all the above settings which can be directly loaded in NetSim

1. Go to NetSim home page, click on **Open Simulation**, Click on **LTE\_Handover\_Example.**

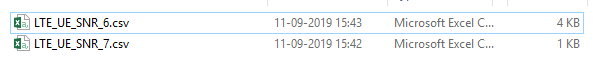


The **mobility.txt** file present in the WorkSpace\_LTE\_Handover -> LTE\_Handover\_Example folder.

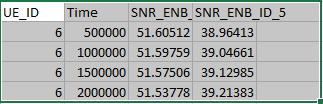
**Network Scenario:**



1. Simulate the scenario in Netsim and you will get two .csv files in bin folder of Netsim corresponding to each UE’s involved in the simulaiton. Open any .csv file of the UE for which File Based Mobility was configured. You will observe the columns containing SNR measured by each ENB with respect to that particular UE.



1. With the help of Excel tools create a graph between SNR measured by each ENB with respect to time period. Properly set the gaps on x-axis and y-axis so you will get a clear graph.

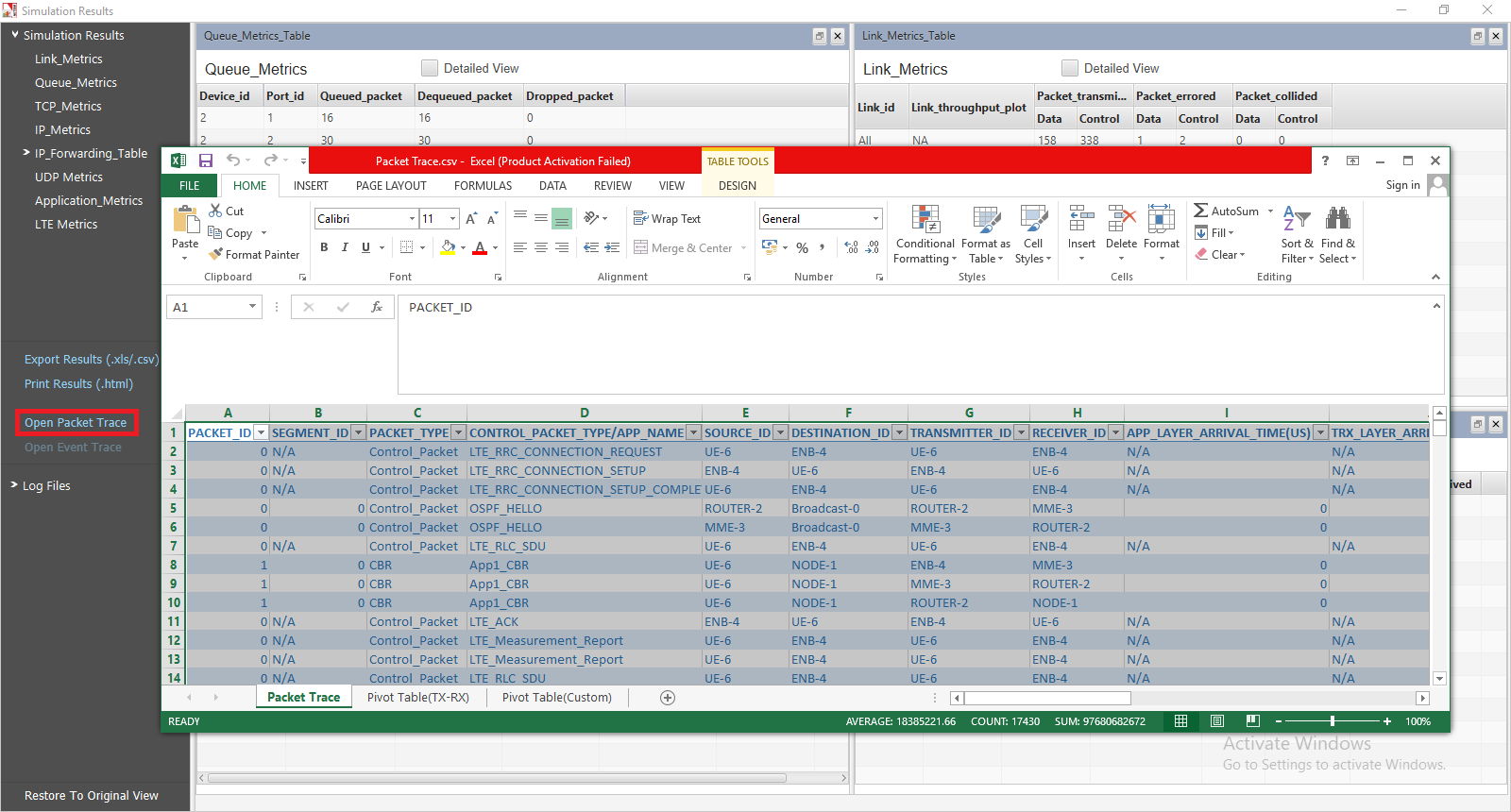


1. Handover occurs when SNR difference between two ENBs is equal to 3 db. So for this scenario find in the graph the time value when SNR\_ENB\_5-SNR\_ENB\_4 measured is equal to 3 db.
2. This is the time value at which handover occurs.

**Result**

Handover occurs when difference between SNR measured by two ENB is equal to 3 db. In below chart Handover occurs when Difference value is equal to 3 on y axis which is between 30.5 sec and 31 sec.

The packet trace file can be accessed from the results window to understand the various packets involved in the handover process.



As UE moves from one position to another it sends measurement report to each ENB in range. As it moves SNR received by each ENB keeps on changing based on distance between ENB and UE. If the difference between SNR received by new ENB to that of old ENB to which it is connected gets greater than 3 decibel than at that point handover occurs.

Related Article links:

[How to Implement Time to Trigger (TTT) used in LTE Handover Modelling?](https://tetcos.freshdesk.com/support/solutions/articles/14000084404-how-to-implement-time-to-trigger-ttt-used-in-lte-handover-modeling-)

[How to vary the Handover Margin in LTE?](https://tetcos.freshdesk.com/support/solutions/articles/14000084401-how-to-vary-the-handover-margin-in-lte-)