Advanced Computer Network

PRACTICAL NO 9

027\_Abhishek\_Ojha

**Practical 9:**

**Aim:**

Create Single mobile network.

**Source Code:**

**MobileNetwork.ned**

**package** inet.examples.mobility;

**network** MobileNetwork

{

**parameters**:

**int** numHosts;

**@display**("bgb=600,400");

**submodules**:

host[numHosts]: MobileHost {

**parameters**:

**@display**("p=300,300;r=,,#707070");

}

}

**omnetpp.ini**

[General]

*#scheduler-class = "cRealTimeScheduler" #so that speed appears realistic*

*#debug-on-errors = true*

tkenv-plugin-path = ../../../etc/plugins

sim-time-limit = 10day

\*.numHosts = 2

\*\*.constraintAreaMinX = 0m

\*\*.constraintAreaMinY = 0m

\*\*.constraintAreaMinZ = 0m

\*\*.constraintAreaMaxX = 600m

\*\*.constraintAreaMaxY = 400m

\*\*.constraintAreaMaxZ = 0m

\*\*.updateInterval = 0.1s *# test with 0s too, and let getCurrentPosition update the display string from a test module*

\*\*.debug = **true**

\*\*.mobility.initFromDisplayString = **false**

[Config ANSimMobility]

network = MobileNetwork

\*\*.host\*.mobilityType = "ANSimMobility"

\*\*.host\*.mobility.ansimTrace = **xmldoc**("ansimtrace.xml")

\*\*.host\*.mobility.nodeId = -1 *#means "host module's index"*

[Config BonnMotionMobility1]

network = MobileNetwork

description = "2 hosts"

\*\*.host\*.mobilityType = "BonnMotionMobility"

\*\*.host\*.mobility.traceFile = "bonnmotion\_small.movements"

\*\*.host\*.mobility.nodeId = -1 *#means "host module's index"*

[Config BonnMotionMobility2]

network = MobileNetwork

description = "100 hosts"

\*.numHosts = 100

\*\*.host\*.mobilityType = "BonnMotionMobility"

\*\*.host\*.mobility.traceFile = "bonnmotion\_scenario.movements"

\*\*.host\*.mobility.nodeId = -1 *#means "host module's index"*

[Config ChiangMobility]

network = MobileNetwork

\*.numHosts = 1

\*\*.host\*.mobilityType = "ChiangMobility"

\*\*.host\*.mobility.stateTransitionUpdateInterval = 3s

\*\*.host\*.mobility.speed = 10mps

[Config CircleMobility1]

network = MobileNetwork

\*.numHosts = 3

\*\*.host\*.mobilityType = "CircleMobility"

\*\*.host\*.mobility.cx = 200m

\*\*.host\*.mobility.cy = 200m

\*\*.host\*.mobility.r = 150m

\*\*.host\*.mobility.speed = 40mps

\*\*.host[0].mobility.startAngle = 0deg

\*\*.host[1].mobility.startAngle = 120deg

\*\*.host[2].mobility.startAngle = 240deg

[Config CircleMobility2]

network = MobileNetwork

\*.numHosts = 3

\*\*.host\*.mobilityType = "CircleMobility"

\*\*.host[0].mobility.cx = 100m

\*\*.host[1].mobility.cx = 300m

\*\*.host[2].mobility.cx = 500m

\*\*.host\*.mobility.cy = 200m

\*\*.host\*.mobility.r = 150m

\*\*.host\*.mobility.speed = 40mps

\*\*.host\*.mobility.startAngle = 0deg

[Config ConstSpeedMobility]

network = MobileNetwork

\*\*.host\*.mobilityType = "ConstSpeedMobility"

\*\*.host\*.mobility.initFromDisplayString = **false**

\*\*.host\*.mobility.speed = 50mps

[Config ConstSpeedMobility01]

extends = ConstSpeedMobility

\*\*.updateInterval = 0.1s

[Config ConstSpeedMobility1]

extends = ConstSpeedMobility

\*\*.updateInterval = 1s

[Config ConstSpeedMobility10]

extends = ConstSpeedMobility

\*\*.updateInterval = 10s

[Config ConstSpeedMobility100]

extends = ConstSpeedMobility

\*\*.updateInterval = 100s

[Config ConstSpeedMobility1000]

extends = ConstSpeedMobility

\*\*.updateInterval = 1000s

[Config GaussMarkovMobility]

network = MobileNetwork

\*.numHosts = 1

\*\*.host\*.mobilityType = "GaussMarkovMobility"

\*\*.host\*.mobility.alpha = 0.9

\*\*.host\*.mobility.speed = 10mps

\*\*.host\*.mobility.angle = 0deg

\*\*.host\*.mobility.variance = 40

\*\*.host\*.mobility.margin = 30m

[Config LinearMobility]

network = MobileNetwork

\*\*.host\*.mobilityType = "LinearMobility"

\*\*.host\*.mobility.initFromDisplayString = **false**

\*\*.host\*.mobility.speed = 50mps

\*\*.host\*.mobility.angle = 30deg *# degrees*

*#\*\*.host\*.mobility.acceleration = -0.5*

[Config LinearMobility01]

extends = LinearMobility

\*\*.updateInterval = 0.1s

[Config LinearMobility1]

extends = LinearMobility

\*\*.updateInterval = 1s

[Config LinearMobility10]

extends = LinearMobility

\*\*.updateInterval = 10s

[Config LinearMobility100]

extends = LinearMobility

\*\*.updateInterval = 100s

[Config LinearMobility1000]

extends = LinearMobility

\*\*.updateInterval = 1000s

[Config LinearMobility\_accdown]

extends = LinearMobility

\*\*.updateInterval = 0.1s

\*\*.host\*.mobility.acceleration = -1.0 *# m/s2*

[Config LinearMobility\_accup]

extends = LinearMobility

\*\*.updateInterval = 0.1s

\*\*.host\*.mobility.speed = 0mps

\*\*.host\*.mobility.acceleration = 1.0 *# m/s2*

[Config MassMobility]

network = MobileNetwork

\*.numHosts = 5

\*\*.host\*.mobilityType = "MassMobility"

\*\*.host\*.mobility.initFromDisplayString = **false**

\*\*.host\*.mobility.changeInterval = truncnormal(2s, 0.5s)

\*\*.host\*.mobility.changeAngleBy = normal(0deg, 30deg)

\*\*.host\*.mobility.speed = truncnormal(15mps, 5mps)

[Config MassMobilityWithScenario]

network = MobileNetworkWithScenario

\*.numHosts = 5

\*\*.host\*.mobilityType = "MassMobility"

\*\*.host\*.mobility.initFromDisplayString = **false**

\*\*.host\*.mobility.changeInterval = truncnormal(2s, 0.5s)

\*\*.host\*.mobility.changeAngleBy = normal(0deg, 30deg)

\*\*.host\*.mobility.speed = truncnormal(15mps, 5mps)

\*\*.scenarioManager.script = **xmldoc**("scenario.xml")

[Config MoBANMobility1]

network = MoBANNetwork

\*\*.constraintAreaMaxX = 1000m

\*\*.constraintAreaMaxY = 1000m

\*\*.constraintAreaMaxZ = 1000m

\*\*.numNodes = 12

\*\*.numMoBAN = 1

\*\*.coordinator[\*].postureSpecFile = **xmldoc**("postures1.xml")

\*\*.coordinator[\*].configFile = **xmldoc**("configMoBAN1.xml")

\*\*.coordinator[\*].useMobilityPattern = **false**

\*\*.coordinator[0].mobilityPatternFile = "MoBAN\_Pattern\_in0.txt"

\*\*.node[\*].mobilityType = "MoBANLocal"

\*\*.node[\*].mobility.coordinatorIndex = 0

[Config MoBANMobility2]

network = MoBANNetwork

\*\*.constraintAreaMaxX = 1000m

\*\*.constraintAreaMaxY = 1000m

\*\*.constraintAreaMaxZ = 1000m

\*\*.numNodes = 44

\*\*.numMoBAN = 2

\*\*.coordinator[\*].postureSpecFile = **xmldoc**("postures1.xml")

\*\*.coordinator[\*].configFile = **xmldoc**("configMoBAN2.xml")

\*\*.coordinator[\*].useMobilityPattern = **false**

\*\*.coordinator[\*].mobilityPatternFile = ""

\*\*.node[0..19].mobilityType = "ConstSpeedMobility"

\*\*.node[0..19].mobility.speed = 30mps

\*\*.node[20..43].mobilityType = "MoBANLocal"

\*\*.node[20..31].mobility.coordinatorIndex = 0

\*\*.node[32..43].mobility.coordinatorIndex = 1

\*\*.node[0].mobility.initialX = 5m

\*\*.node[0].mobility.initialY = 5m

\*\*.node[0].mobility.initialZ = 4m

\*\*.node[1].mobility.initialX = 12m

\*\*.node[1].mobility.initialY = 10m

\*\*.node[1].mobility.initialZ = 4m

\*\*.node[2].mobility.initialX = 20m

\*\*.node[2].mobility.initialY = 5m

\*\*.node[2].mobility.initialZ = 4m

\*\*.node[3].mobility.initialX = 30m

\*\*.node[3].mobility.initialY = 8m

\*\*.node[3].mobility.initialZ = 4m

\*\*.node[4].mobility.initialX = 40m

\*\*.node[4].mobility.initialY = 3m

\*\*.node[4].mobility.initialZ = 4m

\*\*.node[5].mobility.initialX = 6m

\*\*.node[5].mobility.initialY = 18m

\*\*.node[5].mobility.initialZ = 4m

\*\*.node[6].mobility.initialX = 15m

\*\*.node[6].mobility.initialY = 15m

\*\*.node[6].mobility.initialZ = 4m

\*\*.node[7].mobility.initialX = 20m

\*\*.node[7].mobility.initialY = 8m

\*\*.node[7].mobility.initialZ = 4m

\*\*.node[8].mobility.initialX = 35m

\*\*.node[8].mobility.initialY = 20m

\*\*.node[8].mobility.initialZ = 4m

\*\*.node[9].mobility.initialX = 45m

\*\*.node[9].mobility.initialY = 15m

\*\*.node[9].mobility.initialZ = 4m

\*\*.node[10].mobility.initialX = 40m

\*\*.node[10].mobility.initialY = 25m

\*\*.node[10].mobility.initialZ = 4m

\*\*.node[11].mobility.initialX = 16m

\*\*.node[11].mobility.initialY = 25m

\*\*.node[11].mobility.initialZ = 4m

\*\*.node[12].mobility.initialX = 24m

\*\*.node[12].mobility.initialY = 35m

\*\*.node[12].mobility.initialZ = 4m

\*\*.node[13].mobility.initialX = 35m

\*\*.node[13].mobility.initialY = 32m

\*\*.node[13].mobility.initialZ = 4m

\*\*.node[14].mobility.initialX = 35m

\*\*.node[14].mobility.initialY = 28m

\*\*.node[14].mobility.initialZ = 4m

\*\*.node[15].mobility.initialX = 45m

\*\*.node[15].mobility.initialY = 40m

\*\*.node[15].mobility.initialZ = 4m

\*\*.node[16].mobility.initialX = 2m

\*\*.node[16].mobility.initialY = 45m

\*\*.node[16].mobility.initialZ = 4m

\*\*.node[17].mobility.initialX = 10m

\*\*.node[17].mobility.initialY = 40m

\*\*.node[17].mobility.initialZ = 4m

\*\*.node[18].mobility.initialX = 23m

\*\*.node[18].mobility.initialY = 45m

\*\*.node[18].mobility.initialZ = 4m

\*\*.node[19].mobility.initialX = 37m

\*\*.node[19].mobility.initialY = 43m

\*\*.node[19].mobility.initialZ = 4m

[Config RandomWPMobility]

network = MobileNetwork

\*.numHosts = 5

\*\*.host\*.mobilityType = "RandomWPMobility"

\*\*.host\*.mobility.initFromDisplayString = **false**

\*\*.host[0].mobility.speed = 10\*uniform(20mps,50mps)

\*\*.host\*.mobility.speed = uniform(20mps,50mps)

\*\*.host\*.mobility.waitTime = uniform(3s,8s)

[Config RectangleMobility]

network = MobileNetwork

\*\*.host\*.mobilityType = "RectangleMobility"

\*\*.host\*.mobility.constraintAreaMinX = 100m

\*\*.host\*.mobility.constraintAreaMinY = 100m

\*\*.host\*.mobility.constraintAreaMaxX = 500m

\*\*.host\*.mobility.constraintAreaMaxY = 300m

*#\*\*.host\*.mobility.x1 = 100*

*#\*\*.host\*.mobility.y1 = 100*

*#\*\*.host\*.mobility.x2 = 500*

*#\*\*.host\*.mobility.y2 = 300*

\*\*.host[0].mobility.startPos = 0

\*\*.host[1].mobility.startPos = 2.5

\*\*.host[0].mobility.speed = 20mps

\*\*.host[1].mobility.speed = -10mps

[Config StaticGridMobility]

network = MobileNetwork

\*.numHosts = 20

\*\*.host\*.mobilityType = "StaticGridMobility"

\*\*.host\*.mobility.marginX = 100m

\*\*.host\*.mobility.marginY = 100m

\*\*.host\*.mobility.numHosts = 20

[Config StationaryMobility]

network = MobileNetwork

\*.numHosts = 3

\*\*.host\*.mobilityType = "StationaryMobility"

*# place it at a fixed position:*

\*\*.host[0].mobility.initialX = 50m

\*\*.host[0].mobility.initialY = 200m

\*\*.host[0].mobility.initFromDisplayString = **false**

*# the second node is using the display string position (or placed randomly if position is not present in display string)*

\*\*.host[1].mobility.initFromDisplayString = **true**

*# place it at a random position:*

\*\*.host[2].mobility.initFromDisplayString = **false**

[Config TractorMobility]

network = MobileNetwork

\*.numHosts = 1

\*\*.host\*.mobilityType = "TractorMobility"

\*\*.host\*.mobility.x1 = 100m

\*\*.host\*.mobility.y1 = 100m

\*\*.host\*.mobility.x2 = 500m

\*\*.host\*.mobility.y2 = 300m

\*\*.host\*.mobility.rowCount = 4

\*\*.host\*.mobility.speed = 50mps

[Config TurtleMobility1]

network = MobileNetwork

description = "square"

\*.numHosts = 1

\*\*.host\*.mobilityType = "TurtleMobility"

\*\*.host\*.mobility.turtleScript = **xmldoc**("turtle.xml", "movements//movement[@id='1']")

[Config TurtleMobility2]

network = MobileNetwork

description = "two squares"

\*.numHosts = 1

\*\*.host\*.mobilityType = "TurtleMobility"

\*\*.host\*.mobility.turtleScript = **xmldoc**("turtle.xml", "movements//movement[@id='2']")

[Config TurtleMobility3]

network = MobileNetwork

description = "random waypoint"

\*.numHosts = 2

\*\*.host\*.mobilityType = "TurtleMobility"

\*\*.host\*.mobility.turtleScript = **xmldoc**("turtle.xml", "movements//movement[@id='3']")

[Config TurtleMobility4]

network = MobileNetwork

description = "mass+reflect"

\*.numHosts = 2

\*\*.host\*.mobilityType = "TurtleMobility"

\*\*.host\*.mobility.turtleScript = **xmldoc**("turtle.xml", "movements//movement[@id='4']")

[Config TurtleMobility5]

network = MobileNetwork

description = "mass+wrap"

\*.numHosts = 2

\*\*.host\*.mobilityType = "TurtleMobility"

\*\*.host\*.mobility.turtleScript = **xmldoc**("turtle.xml", "movements//movement[@id='5']")

[Config TurtleMobility6]

network = MobileNetwork

description = "mass+placerandomly"

\*.numHosts = 2

\*\*.host\*.mobilityType = "TurtleMobility"

\*\*.host\*.mobility.turtleScript = **xmldoc**("turtle.xml", "movements//movement[@id='6']")

**Output:**

