International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 9, Issue 2, February 2022)

Wireless Sensor Networks Technology and Simulation Software Overview

AJAY SONKESRIYA
Lecturer
Electronics and Communication Engineering
Kalaniketan polytechnic College Jabalpur MP

Abstract: A new generation of massive-scale sensor networks has been enabled by advancements in wireless networking, microfabrication, and integration of sensors and actuators created using micro-electromechanical system (MEMS) technology and embedded microprocessors. Sensor networks have the potential to connect end users directly to sensor readings and give information that is accurately localised in time and/or location, based on the user's requirements. Simulators that replicate the behaviour of a sensor network on a per-node basis are known as node-level design simulators. Designers can quickly assess performance in terms of timing, power, bandwidth, and scalability using simulation rather than implementing them on real hardware and dealing with physical phenomena. The current communication provides an overview of wireless sensor networks (WSNs), as well as information on WSN components, operating systems (TinyOS, RIOT, Nano-RK, MANTIS, and others), and WSN simulation software (NS2, OMNET++, J-Sim, JiST, GloMoSim, SSFNet, and others). The goal is to give learners/researchers a better grasp of current research concerns in this field and to define the use of specific tools to accomplish design objectives that will help them construct WSN applications.

Keywords: Sensor nodes, Architecture, Operating system, Simulation software, Wireless technology.

I. INTRODUCTION

In the nineteenth century, the phrase "wireless communication" was coined. Marconi made the first radio transmission from the Isle of Wright to Tugboat, 18 miles away, in 1895. Information can be conveyed by electromagnetic waves via the air without the use of wires or electronic conductors in this technology. Wireless communication technology is used in sensing, monitoring, smart phones, computers, Bluetooth technology, and networking at the moment.

The existing Internet is extended deep into the physical environment by sensor networks. The resulting new network is more vast and dynamic than today's TCP/IP network, and it's generating totally new sorts of traffic that aren't found on the Internet today. To efficiently support user-level tasks, information gathered by and delivered on a sensor network specifies conditions of physical environments, such as temperature, humidity, or vibration, and requires powerful query interfaces and search engines.

Compared to typical centralised techniques, networked sensing has numerous advantages. By reducing average distances between sensor and signal source, or destination, dense networks of distributed communicating sensors can enhance signal-to-noise ratio (SNR) [1]. The multi-hop design of the network allows for increased energy efficiency in communications. Furthermore, in-network processing can combine additional relevant data from other sensors during this multi-hop transmission. Improved robustness and scalability are the most significant benefits of networked sensing. Individual sensor node or link failures are less likely in a decentralised sensing system.

Decentralized algorithms are also significantly more scalable in practise, and for some applications, they may be the only method to achieve the massive scales required [2].

WIRELESSSENSORNETWORKARCHITECTURE

Wirelesssensornetwork(WSN)orwirelesssensor&actuator network (WSAN) are spatially distributed sensorsto monitor physical or environmental conditions such astemperature, humidity, fire etc. and to cooperatively passtheir data through the network to the main location. WSNconsist of three main components nodes, gateways and thesoftware. The sensors measure the parameter of interest &transmit their data wirelessly through the gateway to thehost system where the software collects the data, processesitsothatitcanbeanalysed[2,3].

A. Sensor nodeor Mote

The main components of the WSN sensor node is radiomodem, controller, sensor and power supply. The blockdiagramofsensornodeisshowninFig.1.

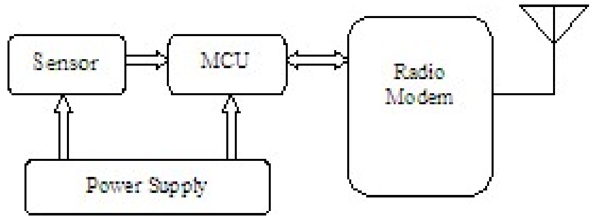

Fig.1.Blockdiagramofsensornode

1. Radiomodem:

They often make use of ISM band which gives free radiospectrumallocationandglobalavailability.WSNusesthreelicencefreecommunicationfrequenciesbands868MHz,915MHzand2.4GHz.

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 9, Issue 2, February 2022)

2. Controller:

The controller performs tasks, processes data and controlsthe functionality of other components in the sensor node.Theanaloguesignalsproducedfromthesensorareconverted into digital signals by the analogue to digitalconverter(ADC)andfedtotheprocessingunit.

3. Sensors:

They are the hardware devices that produce a measurableresponseto achangeinphysicalcondition.They havesmallsizeandaretypicallylargeinnumbersoastomeasuremaximumparameters.

4. Powersupply:

TheWSNaredesignedwithouttheneedofhumanintervention as they are placed in hard to reach location.The battery formsthe heartof the sensorsystemasitdecides the lifespan of the system. Hence battery plays animportant role in ensuring that there is adequate energyavailabletopowerthesystem.

B. Gateway

The Gateway acts as a bridge between the WSN and theothernetwork.TheGatewaycollectstheinformationreceivedfromthemotesinadatabaseandmakesthisinformation available usually via a wireless network [3].Sensornetworksmayinternetworkwithan IPcorenetwork via a number of gateways. A gateway routes userqueriesorcommandstoappropriate nodesinasensornetwork. It also routes sensor data, at times aggregated andsummarized,touserswhohaverequesteditorareexpectedtoutilizetheinformation.

C. Software

A typical Operating System (OS) abstracts the hardwareplatform by providing a set of services for applications,includingfilemanagement,memoryallocation,taskscheduling, peripheral device drivers, and networking. Forembeddedsystems,duetotheirhighlyspecializedapplications and limited resources, their operating systemsmake different trade-offs when providing these services.For example, if there is no file management requirement,then a file system is obviously not needed. If there is nodynamicmemory allocation,thenmemory managementcan be simplified. If prioritization among tasks is critical,then a more elaborate priority scheduling mechanism maybeadded.TinyOS,MATE,aVirtualMachinefortheBerkeley motesandTinyGALSareexamplesofnode-levelprogrammingtools.Differentavailable OSare[4,5]

TinyOs: TinyOS have no file system, supports only staticmemory allocation, implements a simple task model, andprovidesminimaldeviceandnetworkingabstractions.Furthermore,ittakesalanguage-basedapplicationdevelopment approach, so that only the necessary parts oftheoperatingsystemarecompiledwiththeapplication.To a certain extent, each TinyOS application is built into theoperatingsystem.TinyOSorganizescomponentsintolayers. Intuitively, the lower a layer is, the “closer” it is tothe hardware; the higher a layer is, the “closer” it is to theapplication. In addition to the layers, TinyOS has uniquecomponent architecture and provides as a library a set ofsystem software components.A component specificationisindependentofthecomponentimplementation.Althoughmostcomponentsencapsulatesoftwarefunctionalities,someare justthinwrappersaroundhardware. The nesC language is used for programming. Itis an extension of C to support and reflect the design ofTinyOSv1.0and above.Itprovidesasetof languageconstructsandrestrictionstoimplementTinyOScomponentsandapplications.

RIOT:Itisasmall operatingsystem fornetworked,memory-constrained systems with a focus on low-powerwireless Internet of Things (IoT) devices. It is based onmicrokernelarchitecture.RIOTallowsapplicationprogrammingwith C and C++languagesandprovidesmulti-threadingandrealtimeabilities.

OpenTag: It is a DASH7 protocol stack & is designed torun on microcontrollers or a radio system on chip (SoC).Itis written in C language. OpenTag is designed to be lightandcompact,asitistargetedtorunonresource-constrainedmicro-controllers.

Nano-RK:"Nano"impliesthattheRTOSissmall,consuming less power while "RK" is short for “resourcekernel”. It is an open source OS written in C and is a fixed,pre-emptivemultitaskingrealtimeoperatingsystemdesignedforusein WSN.

LiteOs:Itisopensource,writtenin CandUnix-likeoperating system that’s fit on memory constrained sensornodes.LiteOSprovidesafamiliarprogrammingenvironment based on UNIX, threads and C. It follows ahybrid programming model allows both event-drivenandthread-drivenprogramming.

MANTIS: The MultimodAl system for NeTworks of In-situwirelessSensors(MANTIS)providesanewmultithreaded operating system for WSNs. It is an opensource OS which is written in C. It provides automatic pre-emptivetimeslicingforfastprototyping.

Contiki: It is anevent drivenoperating system. It makesthetaskofprogrammingthesensornetworkcloselyresemble programming a PC.It is an open source OS.Contikiprovidesmultitaskingandabuilt-in InternetProtocol Suite (TCP/IP stack) with a full Graphical UserInterface(GUI)features.

II. WSNSIMULATIONSOFTWARE’S

Node-leveldesignmethodologies are usually associatedwithsimulators that simulate the behaviour of a sensornetwork on a per-node basis. Using simulation, designerscanquicklystudytheperformanceintermsoftiming,power, bandwidth, and scalability of algorithms withoutimplementing them on actual hardware and dealing withthe actual physical phenomena.A node in a simulator actsasasoftwareexecutionplatform,asensorhost,aswellasacommunicationterminal.Inorderfordesignerstofocus ontheapplicationlevelcode,anodemodeltypicallyprovidesorsimulatesacommunicationprotocolstack,sensor behaviours e.g., sensing noise and operating systemservices. If the nodes are mobile, then the positions andmotion properties of the nodes need to bemodelled. Ifenergycharacteristicsarepartofthedesignconsiderations, then the power consumption of the nodesneeds to be modelled. Popular simulation software's aredescribedbelow[6].

NS-2- The Network Simulator-2 (NS-2) is an open-sourcenetwork simulator that was originally designed for wired,IPnetworks.Extensionshavebeenmadetosimulatewireless /mobile networks (e.g., 802.11 MAC and TDMAMAC)andmorerecentlysensornetworks.Whiletheorigin alNS-2only supportslogicaladdressesforeachnode, the wireless/mobile extension of it introduces thenotionof node locations and a simple wireless channelmodel. This is not a trivial extension, since once the nodesmove, the simulator needs to check for each physical layereventwhetherthedestinationnodeiswithinthecommunicatio nrange.Foralargenetwork,thissignificantly slows downthe simulationspeed[7]. ThemainmenuofNS-2isshowninFig.2.

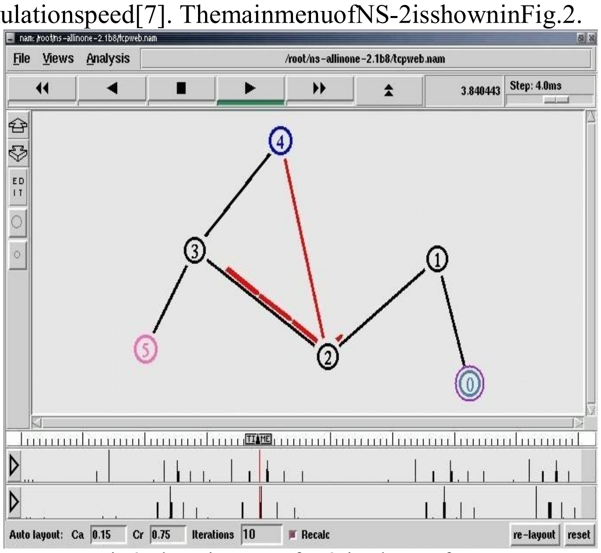

Fig.2.Themain menu ofNS2simulator software

The main functionality of NS-2 is implemented in C++,while the dynamics of the simulation (e.g., time-dependentapplicationcharacteristics)iscontrolledby Tclscripts.Basic components in NS-2 are the layers in the protocolstack. They implement the handlers interface, indicatingthat they handle events. Events are communication packetsthatarepassedbetweenconsecutivelayerswithinonenode, or between the same layers across nodes. The mainadvantage of NS-2is its richlibraries of protocolsfornearlyallnetworklayersandformanyroutingmechanisms.

OMNET++-
ItisamodulardiscreteeventsimulatorimplementedinC++.Getting startedwithitisquitesimple, due to its clean design. OMNET++ also provides apowerfulGUIlibraryfor3-D virtualization,animationand tracing and debugging support. OMNeT++ has been usedinnumerousdomainsfromqueuingnetworksimulationsto wireless and ad-hoc network simulations, from businessprocess simulation to peer-to-peer network, optical switchand storage area network simulations. OMNET++ openingmenuisshowninFig.3[8].

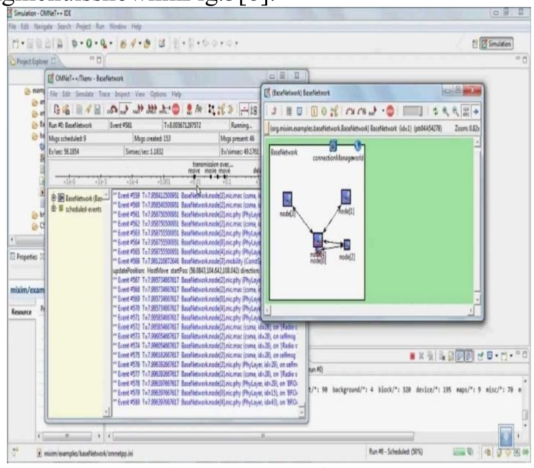

Fig.3.TheopeningmenuofOMNET++

J-Sim-Acomponent-basedsimulationenvironmentdeveloped entirely in Java. The main benefit of J-sim is itsconsiderable list of supported protocols, including a WSNsimulationframeworkwithaverydetailedmodelofWSNs and an implementation of localization, routing anddata diffusion WSN algorithms [9]. J-Sim main menu isshowninFig. 4.

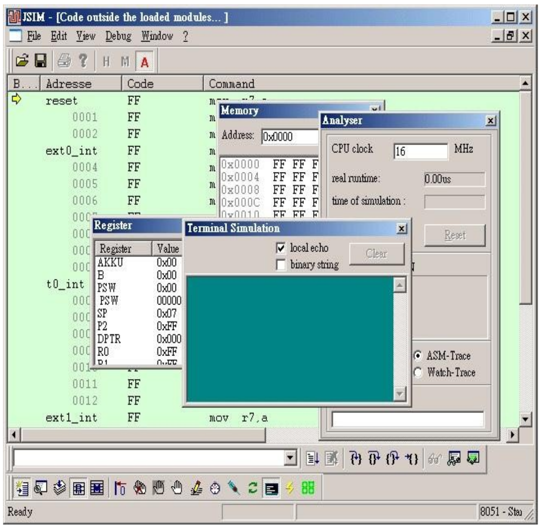

Fig.4.J-Simmainmenuwith differentwindows

NCTUns2.0- It is discrete event simulator whose engine isembedded in the kernel of a UNIX machine. The actualnetworklayerpacketsaretunnelledthroughvirtualinterfaces thatsimulatelowerlayersandphysicaldevices [10]anditsscreen shotisshowninFig.5.

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 9, Issue 2, February 2022)

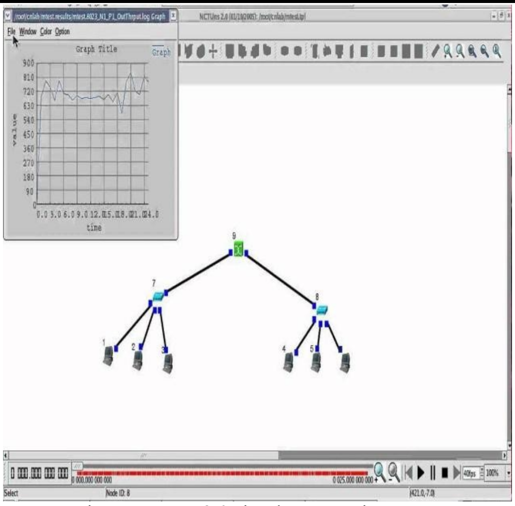

Fig.5.NCTUns2.0 simulator openingmenu

JiST/SWANS-JavainSimulationTime(JiST)andScalable Wireless Ad-hoc Network Simulator (SWANS)are discrete event simulation framework that embeds thesimulationengineintheJavabyte-code.Modelsareimplemented in Java and compiled. Then the byte-codesare rewritten to introduce simulation semantics. Javis is apacketflow and networkanimatorforJiST[11].Javissimulatoropeningmenuis showninFig.6.

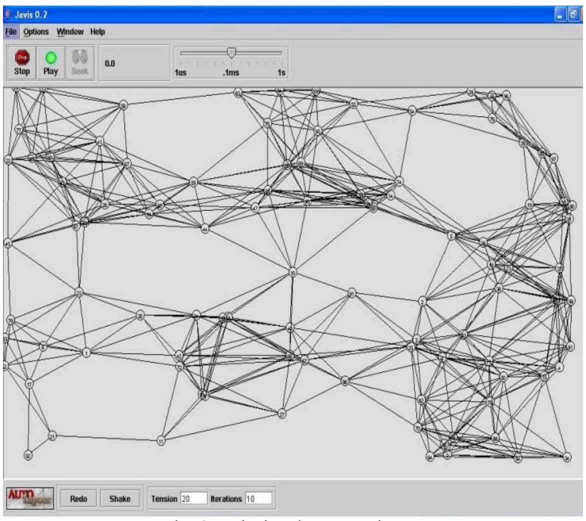

Fig.6.Javissimulatoropeningmenu

SSFNet- It is a set of Java network models built over theScalable Simulation Framework (SSF) for modeling andsimulation of Internet protocols and networks at and abovethe IP packet level. SSF is a specification of a commonAPIforsimulationthatassuresportabilitybetweencompliantsimulators.SSFNetmodelsare self-configuring i.e.eachSSFNetclassinstancecanautonomously configure itself by querying a configurationdatabase, which may be locally resident or available overthe Web [12]. SSFNet simulator execution screen shot isshowninFig.7.

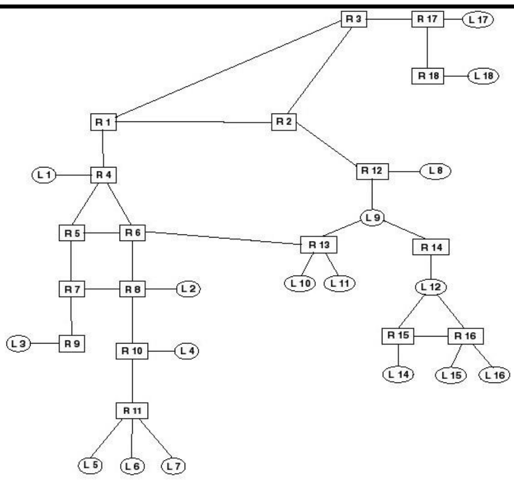

Fig.7.SSFetsimulatorExecutionscreenshot

GloMoSim-Simulationenvironmentforwirelessnetworksbuiltwithwith Parsec.Parsecisasimulationlanguage derived from C that adds semantics for creatingsimulationentitiesandmessagecommunicationonavariety of parallel architectures[13].GloMoSimscreenshotisshowninFig.8.

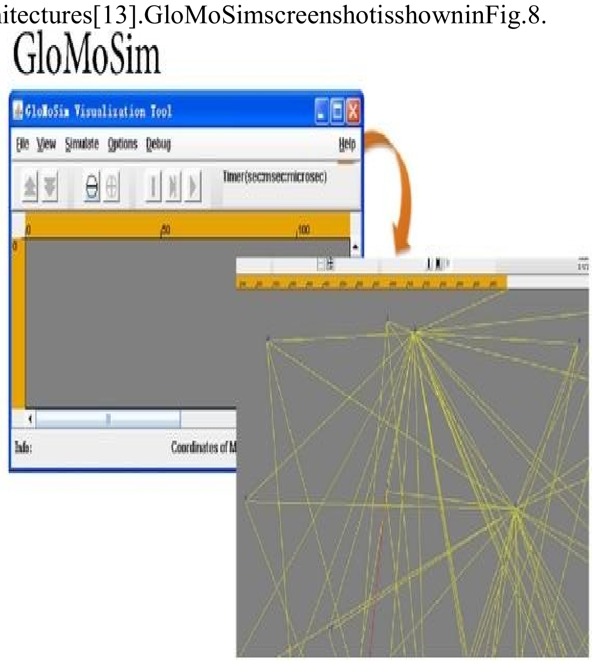

Fig.8.GloMoSimsimulator openingmenu

PtolemyII-ItcontainsJavapackagesthatsupportdifferent models of simulation paradigms (e.g. continuostime, dataflow, and discrete-event). It also addresses themodeling, simulation and design of concurrent, real-time,embedded systems. A major problem area being addressedistheuseofheterogeneousmixturesofmodelsofcomputation.Theprojectisnamedafter ClaudiusPtolemaeus,the2ndcenturyGreekastronomer,mathematician, and geographer [14]. The opening menu ofPtolemyIIisshowninFig.9.

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 9, Issue 2, February 2022)

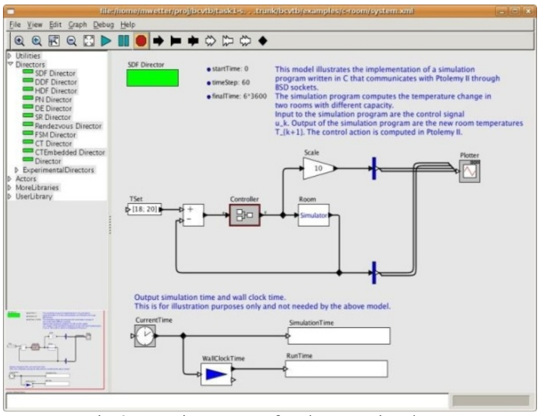

Fig.9.Openingmenu ofPtolemyII simulator

III. DIFFERENTWIRELESSTECHNOLOGIES

To Use wireless technology for different applications, theInstitute of Electrical and Electronic Engineers (IEEE) hasproposed standards and protocols according to its need.Table1listthecomparisonofdifferentstandards [15].

Table1.Comparisonofdifferentcommunicationstandards

Standard	Bluetooth	Infrared	Wi-Fi	ZigBee
Governingbody	luetoothSIG	Infrared DataAssociation	Wifi Alliance	**Zigbee Alliance**
IEEE Specification	802.15.1	802.11	802.11 a/b/g	**802.15.4**
FrequencyBand	2.4GHz	875nm+-	2.4GHz, 5 MHz	**868/915MHz, 2.4 GHz**
Standard range	1-100m	0.2-1m	100m	**10-100m**
PowerConsumption (Days)	1 to7	<200	1 to5	**100to >1000**
NumberofRF channels	79	50	14	**1/10,16**
Datatransfer rate	3Mbits/s	4Mbits/s	4Mbits/s	**50Kbits/s**
Maxno. Ofnodes	8	2	2007	**>65000**
Modulation Type	GFSK	Pulse	BPSK,QPSK,M-QAM,CCM	**BPSK(+ASk), O-QPSK**
pplications	**Cable replacement**	**Cablereplacement**	**Web,E-mailvideo**	**Monitering&control**

IV. APPLICATIONS

WSN are used in various applications. The list of differentareas and the possible parameters measurement are givenbelow[16].

A. Military or Border surveillance applications: ExamplesareVehicledetection,Weapons,Chemicalsensingetc.
B. EnvironmentalApplications:Examplesaretrackingthemovementsandpatternsofinsects,birdsorsmallanimals etc.
C. Healthcare applications: Examples are Blood pressure,Heartbeat,Stress,Bodytemperature,Sleep,Brainactivities etc.
D. HomeIntelligence:ExamplesareHVAC,Lightingcontrolssystem,Security,Gasleakdetection,Energysavingetc.
E. Automobile Industry: Examples are Acceleration, Fuelconsumption, Tirepressure, acknowledgmentofillumination failures (turn lights, brake lights, front lights,andregisterplatelights) etc.
F. PrecisionAgriculture:ExamplesareWaterlevel,Temperature,Humidity,SoilMoisture,pHlevel,Windflow,LightIntensityetc.
G. StructuralMonitoring:ExamplesareWindandWeather,Traffic,Deck,Pylons,Ground,Prestressing,Straycablesetc.
H. Industrial Process Control: Examples are Temperature,Steampressure,Liquidlevels,Flow,Viscosityetc.
I. Environmental Conditions Monitoring: Examples areEarthquakesVolcano,Tsunami, Fire, Flood,Pollutionetc.
J. Oil and Gas industry: Examples are Oil bunkering andtheft,pipelinevandalizationetc.

V. CONCLUSION

Wireless networking advancements, as well as the integration of MEMS-based sensors and actuators with embedded microprocessors, have enabled a new generation of massive-scale sensor networks suited for a variety of commercial and military applications. Sensor networks have the potential to connect end users directly to sensor measurements and deliver information that is precisely localised in time and/or location to meet the needs of the users. There are various different types of OS for WSNs, each with its own set of features, benefits, and drawbacks. Because of the limitation of resources, WSN operating systems are typically less complex. TinyOS was the first operating system designed specifically for WSNs. TinyOS differs from other operating systems in that it uses event-driven programming, whereas others use multithreading.
Simulators that replicate the behaviour of a sensor network on a per-node basis are known as node-level design simulators. Designers can quickly assess performance in terms of timing, power, bandwidth, and scalability using simulation rather than implementing them on real hardware and dealing with physical phenomena. The overview includes suggestions for choosing a suitable simulation model for a WSN as well as a detailed discussion of the most commonly used tools. All of the packages include a graphical user interface.

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 9, Issue 2, February 2022)

interface. The OMNET++, NCTUns2.0, J-Sim, and Ptolemy II GUI libraries provide advanced animation, tracing, and debugging capabilities. Simulators that run in parallel should perform and scale better than those that run sequentially. When considering the new environment and the energy components, modelling issues occur. They also jeopardise scalability and precision. For a better understanding and characterization of sensor networks and their accompanying simulators, a thorough examination of these challenges is required. The ZigBee standard, developed by the ZigBee Alliance, is used to create WSNs that require high reliability, low cost, and low power for monitoring and control applications.

Military, environmental applications, healthcare, home intelligence, automobile industry, precision agriculture, structure monitoring, industrial process control, environmental condition monitoring, oil and gas industry, and others are among the different application fields of WSN.

REFERENCES

[1] Wireless Sensor Networks: An Information Processing ApproachByFeng Zhao,LeonidasGuibas.

[2] WirelessSensorNetworks:ConceptsandComponents(©SpringerInternational Publishing Switzerland 2014 5 J. Cecílio, P.Furtado, Wireless Sensors in Heterogeneous Networked Systems,ComputerCommunicationsandNetworks,DOI10.1007/978-3-319-09280-5_2.

[3] On the design of a flexible gateway for Wireless Sensor NetworksMarcoZENNARO,HervéNTAREMEandAntoineBagula.

[4] OperatingSystemsforWirelessSensorNetworks:ASurveyTechnical Report AdiMallikarjuna Reddy V AVU Phani Kumar, DJanakiram, andGAshokKumar

[5] OperatingSystemsforWirelessSensorNetworks:ASurveyM.O.FarooqandThomasKunzSensors(Basel).2011;11(6):5900–5930.publishonline2011May31doi:10.3390/s110605900

[6] Simulation Tools for Wireless Sensor Networks E. Egea-López, J.Vales-Alonso,A.S.Martínez-Sala,P.Pavón-Mariño,J.García-HaroSummer SimulationMulticonference-SPECTS2005

[7] TheNetwork Simulator,NS–2[Online]http://www.isi.edu/nsnam/ns/

[8] OMNET++ discrete event simulator. [Online]. Available:http://www.omnetpp.org

[9] J-Sim[Online].Available:http://www.j-sim.org

[10] NCTUns 2.0 Network Simulator and Emulator. [Online]. Available:http://nsl.csie.nctu.edu.tw/nctuns.html

[11] R. Barr, Z. J. Haas, R. van Renesse, "JiST: Embedding SimulationTime into a Virtual Machine." In Proc. 5th EUROSIM Congress onModelingandSimulation, Paris, France, September2004

[12] ScalableSimulationFramework(SSF).[Online].Available:http://www.ssfnet.org

[13] Global MobileInformationSystems SimulationLibrary(GloMoSim).[Online].Available:http://pcl.cs.ucla.edu/projects/glomosim/

[14] Ptolemy II. Heterogeneousmodel and design. [Online].Available:http://ptolemy.eecs.berkeley.edu/ptolemyIIWeid,P.Oberson,and N.Gisin,"HighresolutionfiberdistributedmeasurementswithcoherentOFDR,"in Proc. ECOC'00,2000,paper11.3.4, p.109.

[15] ComparativeAnalysisofdifferentwireless Techologies,NeerajChhabra International Journal of Scientific Research in NetworkSecurity&Communication.Vol-1,Issue-5,ISSN:2321-3256

[16] WSN Practical adaptations RitikaSobti, NehaBassi, InternationalJournalof Engineering and ComputerScience ISSN: 2319-7242Volume4 Issue9 Sep2015,PageNo.14443-14448