# 14

# Wireless Sensor Network: At a Glance

A.K. Dwivedi[1] and O.P. Vyas[2]
[1]*School of Studies in Computer Science & Information Technology, Pandit Ravishankar Shukla University, Raipur, C.G.,*
[2]*Indian Institute of Information Technology-Allahabad (IIIT-A), Deoghat, Jhalwa, Allahabad, U.P., India*

## 1. Introduction

Wireless Sensor Network is a technology which has capability to change many of the Information Communication aspects in the upcoming era. From the last decade Wireless Sensor Networks (WSNs) is gaining magnetic attention by the researchers, academician, industry, military and other ones due to large scope of research, technical growth and nature of applications etc. Wireless Sensor Networks (WSNs) employ a large number of miniature disposable autonomous devices known as sensor nodes to form the network without the aid of any established infrastructure. In a Wireless Sensor Network, the individual nodes are capable of sensing the environments, processing the information locally, or sending it to one or more collection points through a wireless link. Day to day applications of WSNs is increasing from domestic use to military use and from ground to space.
The objective of this book chapter is to explore all aspects of WSNs under different modules including these as well in a systematic flow: Sensor nodes, Existing hardware, Sensor node's operating systems, node deployment options, topologies used for WSN, architectures, WSN lifecycle, Resource constraint nature, Applications, Existing experimental tools, Usability & reliability of experimental tools, Routing challenges and Protocol design issues, Major existing protocols, Protocol classifications, Protocols evaluation factors, Theoretical aspects of major Energy Efficient protocols, Security issues, etc.
This chapter contains from very basic to high level technical issues obtained from highly cited research contribution in a concluding manner but presenting whole aspects related to this field.

## 2. Wireless sensor nodes and existing hardware

Wireless sensor nodes are tiny, light weight sensing devices consists of a constrained processing unit, little memory, EEPROM or Flash memory for tiny operating systems and other desired programs, one or more sensors, a limited range transceiver, battery or solar based power unit and optionally a mobility subsystem for mobile sensor nodes (Dwivedi & Vyas, 2010).
Tatiana Bokareva presented a mini hardware survey related to wireless sensor nodes (Tatiana), except this a comprehensive listing of existing wireless sensor nodes is presented

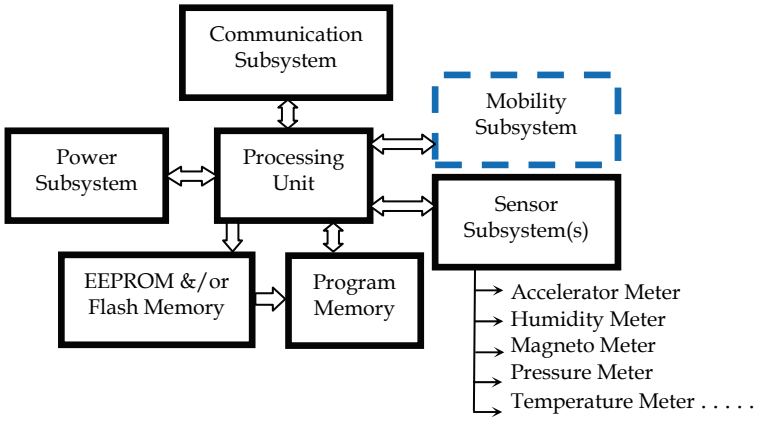



Fig. 1. Block diagram of wireless sensor node

and maintained by Imperial College London (ICL, 2007), Embedded WiSeNts Platform Survey (Embedded WiSeNts, 2006) presents an in-depth survey of five popular wireless sensor nodes (ESB/2, BTnode, uNode, Tmote Sky, and EYES IFXv2), another pretty listing is presented by University of California's Sensor Network Systems Laboratory (Senses, 2005). As well as Sensor Network Museum (SNM, 2010) maintained by TIK computer Engineering and Networks Laboratory, ETH Zurich presents a collection of reference data and links for commonly used wireless sensor nodes and related links. In a research contribution (Manjunath, 2007), technical specifications of some well known wireless sensor nodes are presented in tabular format, as here in its original (Table 1).

Resource footprint (Tatiana; ICL, 2007; Embedded WiSeNts, 2006; Senses, 2005; SNM, 2010; Manjunath, 2007) for various currently available Wireless Sensor nodes provides us a summary that most of the Nodes belongs to within the following configuration:

- 4-bit to 8-bit processor
- 512 Byte to 512 KB RAM (Program and Data Memory)
- 4 KB to 4 MB Flash/External Memory
- 250 Kbps 2.4 GHz IEEE 802.15.4 or Bluetooth 2.0 or 10 Kbps etc. as radio transceiver

On the basis of above mentioned resource footprint it can be concluded that each and every currently available sensor nodes face limited resource problems such as narrow address space and slow clock cycle of micro controller, small program and data memory as well as external memory, low bandwidth and low range of transceivers.

Table 2 presents a wider look on technical aspects of some hardware systems for WSNs, because hardware designing requires a holistic approach for WSNs, looking at all areas of the design space. Expanding the uses of WSNs for various applications, expect more performance for less power out of the hardware platforms. Envision a future of WSNs made up of ultra low power nodes that provide high power computation and can be deployed for decades is possible only with more research effort (Hempstead et al., 2008).

## 3. Operating systems for wireless sensor nodes

WSNs are composed of large numbers of tiny-networked devices that communicate untethered. Operating systems are at the heart of the sensor node architecture. In terms of

Table 1. A summarized list of some popular wireless sensor node (Manjunath, 2007)

| S.N. | Platform | MCU | RAM | Code Memory | RF Transceiver | |
|---|---|---|---|---|---|---|
| 1. | Mica | Atmel ATMega128L | 4KB | 128KB | TR1000 | |
| 2. | Mica2 | Atmel ATMega128L | 4KB | 128KB | CC1000 | 3 |
| 3. | Mica2Dot | Atmel ATMega128L | 4KB | 128KB | CC1000 | 3 |
| 4. | MicaZ | Atmel ATMega128L | 4KB | 128KB | CC2420 | |
| 5. | Cricket | Atmel ATMega128L | 4KB | 128KB | CC1000 | |
| 6. | TelosA | TIMSP430 | 2KB | 60KB | CC2420 | |
| 7. | TelosB | TIMSP430 | 10KB | 48KB | CC2420 | |
| 8. | BTnode3 | Atmel ATMega128L | 64KB | 128KB | Zeevo-BT/CC1000 | 2. |
| 9. | EYES | TIMSP430 | 4KB | 60KB | TR1001 | |
| 10. | Intel mote | ARM7TDMI (Core) | 64KB | 512KB | Zeevo-BT | |
| 11. | Intel mote2 | PXA27x (Core) | 256KB | 32MB | CC2420 | |
| 12. | MANTIS nymph | Atmel ATMega128L | 4KB | 128KB | CC1000 | 315 |
| 13. | XYZ mote | ARM7TDMI (Core) | 32KB | 256KB | CC2420 | |
| 14. | ECR | TIMSP430 | 2KB | 60KB | TR1001 | |
| 15. | ESB | TIMSP430 | 2KB | 60KB | TR1001 | |
| 16. | Smart-Its mote | Atmel ATMega103L | 4KB | 128KB | Ericsson-BT/TR1001 | 2. |
| 17. | Tmote Sky | TIMSP430 | 10KB | 48KB | CC2420 | |
| 18. | TinyNode 584 | TIMSP430 | 10KB | 48KB | Xemics XE1205 | |
| 19. | ZebraNet H/W | TIMSP430 | 2KB | 60KB | 9XStream | |

Table 2. Technical specification for some hardware systems for Wireless Sensor Network (Hempstead et al., 2008)

| SN | System | Arch Style | Data path width | Event driven (Y/N) | Circuit Techniques | Accelerators | Memory (KB) | Process |
|---|---|---|---|---|---|---|---|---|
| 1. | Atmel ATmega128L | GP Off-the-shelf | 8 | N | N | N | 132KB | 350nm |
| 2. | TI MSP430 | GP Off-the-shelf | 16 | N | N | N | 10KB | NA |
| 3. | SNAP/LE | GP RISC | 16 | Y | Asynchronous | Timer, message interface | 8KB | 180nm |
| 4. | BitSNAP | GP RISC Bit-serial datapath | 16 | Y | Asynchronous | Timer, message interface | 8KB | 180nm |
| 5. | Smart Dust | GP RISC | 8 | N | Synchronous - 2 clock | None | 3.125KB | 250nm |
| 6. | Charm | Protocol processor | NA | N | Two power domains | Custom radio stack | 68KB | 130nm |
| 7. | Michigan 1 | GP | 8 | Y | Sub-threshold | None | 0.25KB | 130nm |
| 8. | Michigan 2 | GP | 8 | Y | Sub-threshold | None | 0.3125KB | 130nm |
| 9. | Harvard | Event driven accelerator | 8 | Y | VDD-gating | Timer, filter, message proc | 4KB | 130nm |

Wireless Sensor Networks we need these things in operating system architectures: Extremely small footprint, extremely low system overhead and extremely low power consumption. When designing or selecting operating systems for tiny-networked sensors, our goal is to strip down memory size and system overhead because typical wireless sensor nodes are equipped with a constrained processing unit, little memory, EEPROM or Flash memory, battery or solar based power unit. In a research contribution (Hempstead et al., 2008) and in a technical report (Fröhlich & Wanner, 2008) three classifications of O. S. architectures are described for wireless sensor nodes: Monolithic, Modular/Micro and Virtual Machine.
After evaluating various research contributions specifically devoted to operating systems used for wireless sensor nodes (Fröhlich & Wanner, 2008, Reddy et al., 2007; Dwivedi et al., 2009a; Manjunath, 2007) total 39 operating systems are identified:

| | | | | | |
|---|---|---|---|---|---|
| 1. | TinyOS | 2. | Contiki | 3. | Mantis OS |
| 4. | Microsoft .NET Micro | 5. | YATOS (Yet Another Tiny OS) | 6. | BTnutOS or NutOS |
| 7. | PeerOS | 8. | Embedded Linux | 9. | NanoRK |
| 10. | µCOS | 11. | Squawk VM | 12. | SensorOS |
| 13. | MagnetOS | 14. | CORMOS | 15. | Bertha |
| 16. | kOS | 17. | VMSTAR | 18. | Maté |
| 19. | CVM | 20. | EYES | 21. | SenOS |
| 22. | DCOS | 23. | t-Kernel | 24. | Nano-QPlus |
| 25. | SmartOS | 26. | AVRX | 27. | Pixie |
| 28. | LiteOS | 29. | T2 | 30. | OSSTAR |
| 31. | Jallad | 32. | CustomOS | 33. | GenOS |
| 34. | MoteWorks | 35. | NanoVM | 36. | ParticleVM |
| 37. | KVM | 38. | AmbiCompVM | 39. | SOS |

Table 3. List of operating systems available for Wireless Sensor Nodes

D. Manjunath presents a review of current operating systems for WSNs (Manjunath, 2007) whose aims were to explicate "why sensor operating systems are designed the way they are". This technical report questions every design decision, and provide a detail reasoning for why these decisions.

## 4. Node deployment options in wireless sensor networks

As we know that WSN is deployed to measure environment parameters in Region of Interest (ROI) and to send it to a controller node or base station. In WSNs how nodes will deployed is basically application specific and totally dependent on environment. The node deployment option affects the performance of routing protocol basically in terms of energy consumptions. Basically there are three ways in which tiny sensor nodes can be deployed in a wireless sensor network environment:

- **Regular Deployment** - Sensor nodes can be deployed in a well planned, fixed manner; not necessarily geometric structure, but that is often a convenient assumption. In this type of deployment data is routed through a predefined path.
  **Area of Use**: Medical and health, Industrial sector, Home networks, etc.

- **Random Deployment** – Sensor nodes are scattered over finite area. When the deployment of nodes is not predefined optimal positioning of cluster head becomes a critical issue to enable energy efficient network operation. Random deployment is generally used in rescue operations.
  **Area of Use**: Environmental and Habitual monitoring, etc.
- **Sensor Nodes with Mobility** - Can move to compensate for deployment shortcomings; can be passively moved around by some external force (wind, water, vehicle); can actively seek out "interesting" areas.
  **Area of Use**: Battle field surveillances, Emergency situations (Fire, Volcano, Tsunami), etc.

## 5. Topologies used for wireless sensor networks

Wireless sensor nodes are typically organized in one of three types of network topologies:

- In a **star topology**, each node connects directly to a gateway.
- In a **cluster tree topology**, each node connects to a node higher in the tree and then to the gateway, and data is routed from the lowest node on the tree to the gateway.
- Finally, to offer increased reliability, **mesh networks** feature nodes that can connect to multiple nodes in the system and pass data through the most reliable path available.

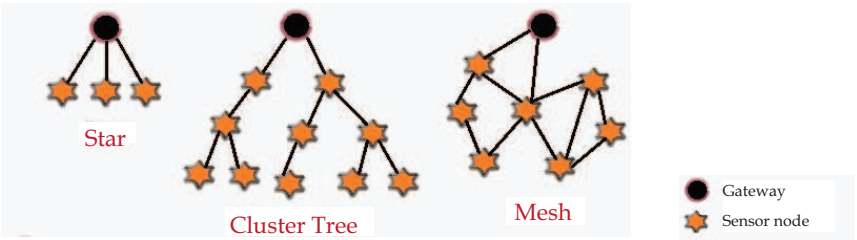



Fig. 2. Topologies used for Wireless Sensor Networks

Three phases related to topology maintenance and changes has been presented in a research contribution (Akyildiz et al., 2002a):

- Pre-deployment and Deployment phase
- Post-deployment phase
- Redeployment of additional nodes phase

## 6. Architectures for wireless sensor networks

In a technical report (Karl & Willig, 2003) Holger Karl and Andreas Willig present views on WSN architectures in the light of principle differences in application scenarios and underlying communication technology. The architecture of WSNs will be drastically different both concerning a single node and the network as a whole. Wide range of sensor node architectures has been presented till today but as a general design principle all of them have targeted the following objectives: energy efficiency, small size and low cost. The architecture for network as a whole is a set of principles that guide where functionality should be implemented along with a set of interfaces, functional units, protocols, and physical hardware that follows those guidelines.

In another research paper (Dulman & Havinga, 2005) the characteristics of wireless sensor networks from an architectural point of view is presented. Since WSNs are designed for specific applications so there is no precise architecture to fit them all but rather a common set of characteristics that can be taken as a starting point. In same paper a data-centric architecture is also presented.
A research paper (NeTS-NOSS, 2007) presents six aspects of architecture for WSN: Design Principles, Functional Architecture, Programming Architecture, Protocol Architecture, System Support Architecture and Physical Architecture. This paper also states that "The situation today in sensor networks is that none of these six levels of network system architecture are 'solved' or even clearly established. The vast majority of the studies fall in the category of protocol architecture".
In a research paper (Vazquez et al., 2009), an architecture for integrating Wireless Sensor Networks into the Internet of Things called "Flexeo" is presented. In another research paper (Schott et al., 2007) a flexible protocol architecture "e-SENSE" for WSNs has been introduced, which is well-suited for capturing the context surrounding service users in order to enable a variety of advanced context-aware applications in beyond 3G mobile communication systems.

## 7. Wireless sensor networks lifecycle

Characteristically, there are four phases in the lifecycle of a wireless sensor network (the implementation phase is omitted because the sensor code is frequently reused). Researchers are usually involved in the planning and deployment phase, while the final customers are more interested in monitoring and control the WSN.


Fig. 3. Wireless sensor network lifecycle

**Planning WSNs** Planning phase usually involves the inspection of the deployment area and the selection of the correct locations to position the sensors in a way that accomplishes the intended goal.

**Deploying WSNs** In the deployment phase, sensor nodes continually send their wireless connection quality and route to the base.

**Monitoring WSNs** In this phase, the user interest is mainly focuses on the values read by network sensors.

**Controlling WSNs** The application can also be used to control WSNs by sending commands to the network. These commands can tell the network devices to stop sending messages, increase the time between messages or even reset the network (restart the Multi-Hop algorithm). In future, WSNs could be controlled via a web interface or a handheld device, being easier to stop and restart the network as needed.

## 8. Resource constraint nature of wireless sensor networks

Wireless Sensor Networks (WSNs) employ a large number of miniature disposable autonomous devices known as sensor nodes to form the network without the aid of any established infrastructure. In a Wireless Sensor Network, the individual nodes are capable of sensing their environments, processing the information locally, or sending it to one or more collection points through a wireless link. Sensor node may fail due to lack of energy, physical damage, communications problem, inactivity (a node becomes suspended), or environmental interference. Resource footprint for various currently available Wireless Sensor nodes is presented in section 2, obtained from (Tatiana; ICL, 2007; Embedded WiSeNts, 2006; Senses, 2005; SNM, 2010; Manjunath, 2007). Here is a table focuses on resource constraint nature of Wireless Sensor Nodes and obviously WSNs:

| Node | CPU | Memory | Radio |
|---|---|---|---|
| **Rene**<br>1999 | ATMEL 8535 | 512 Byte RAM<br>8 KB Flash | 10 Kbps |
| **Mica-2**<br>2001 | ATMEGA 128 | 4 KB RAM<br>128 KB Flash | 76 Kbps |
| **Telos**<br>2004 | Motorola HC 508 | 4 KB RAM | 250 Kbps |
| **Mica-Z**<br>2004 | ATMEGA 128 | 4 KB RAM<br>128 KB Flash | 250 Kbps |
| **BT Node**<br>2001 | ATMEL Mega 128L | 128 KB Flash 4 KB EEPROM<br>4 KB SRAM | Bluetooth |
| **Imote 1.0**<br>2003 | ARM 7TDMI | 64 KB SRAM<br>512 KB Flash | Bluetooth |
| **Stargate**<br>2003 | Intel PXA 255 | 64 KB SRAM | Serial Connection to Sensor Network |
| **Insysnc Cerfoube**<br>2003 | Intel PXA 255 | 32 KB Flash<br>64 KB SRAM | |
| **PC 104** | X86 Processor | 32 KB Flash<br>64 KB SRAM | |

Table 4. Presenting resource constraint nature of some popular wireless sensor nodes

## 9. Applications of wireless sensor networks

WSNs can be applied in a wide range of areas, such as: habitat monitoring and tracking, disaster relief, emergency rescue operation, home networks, detecting chemical/biological/radiological/nuclear/explosive material, monitoring patents and elderly people, asset and warehouse management, building monitoring and control, fleet monitoring, military battlefield awareness and surveillance, security and surveillance, environmental monitoring, pipeline corrosion monitoring, homeland security, monitoring conditions of buildings and bridges, industrial process monitoring and control, machine health monitoring, healthcare applications, home automation, traffic control, etc.
With the help of research contributions (Biradar et al., 2009; Katiyar et al., 2010) a table is presented here, which systematically summarized some applications for different areas:

| Area | Applications |
|---|---|
| Military | • Military situation awareness.<br>• Sensing intruders on basis.<br>• Detection of enemy unit movements on land and sea.<br>• Battle field surveillances |
| Emergency situations | • Disaster management.<br>• Fire/water detectors.<br>• Hazardous chemical level and fires. |
| Physical world | • Environmental monitoring of water and soil.<br>• Habitual monitoring.<br>• Observation of biological and artificial systems.<br>• Marginal Farming. |
| Medical and health | • Sensors for blood flow, respiratory rate, ECG (electrocardiogram), pulse oxymeter, blood pressure and oxygen measurement.<br>• Monitoring people's location and health condition. |
| Industry | • Factory process control and industrial automation.<br>• Monitoring and control of industrial equipment.<br>• Machine health monitoring. |
| Home networks | • Home appliances, location awareness (blue tooth).<br>• Person locator. |
| Automotive | • Tire pressure monitoring.<br>• Active mobility.<br>• Coordinated vehicle tracking. |
| Area monitoring | • Detecting enemy intrusion<br>• Geo-fencing of gas or oil pipelines.<br>• Detecting the presence of vehicles. |
| Environmental monitoring | • Air pollution monitoring.<br>• Forest fires detection.<br>• Greenhouse monitoring.<br>• Landslide detection.<br>• Volcano monitoring.<br>• Flood detection. |
| Water/Wastewater monitoring | • Landfill ground well level monitoring and pump counter.<br>• Groundwater arsenic contamination assessment.<br>• Measuring water quality. |
| Cognitive sensing | • Bio-inspired sensing.<br>• Swarm intelligence.<br>• Quorum sensing. |
| Underwater acoustic sensor systems | • Oceanographic data collection.<br>• Pollution monitoring.<br>• Disaster prevention.<br>• Assisted navigation.<br>• Tactical surveillance. |
| Traffic Management & Monitoring | • Traffic congestion control.<br>• Road Surface Condition Monitoring (BusNet in Sri Lanka). |

Table 5. Some applications of WSNs in different areas

Deploying nodes in an unattended environment will provide more possibilities for the exploration of new applications. WSNs will be ubiquitous in the near future, due to new opportunities for the interaction between humans and their physical world also WSNs are expected to contribute significantly to pervasive computing.

## 10. Existing standards for wireless sensor networks

WSNs fascinate a number of standardization bodies to develop standards, due to a smaller amount of standards exists for WSNs in comparison to other wireless networks. A number of standards are currently under development or ratified for WSNs. Some standardization bodies working in the specific field of WSNs to setup standards, such as:

| **Standardization body** | **Specific work area for WSN** |
|---|---|
| Institute of Electrical and Electronics Engineers | Physical layer and MAC sub layer of Data link layer. |
| Internet Engineering Task Force | Data link layer and all above layers of WSN protocol stack. |
| International Society of Automation | All layers of WSN protocol stack |
| DASH7 Alliance | Promotes the use of the ISO 18000-7 standard for wireless sensor networks. |

Table 6. Some main Standardization bodies and their specific work area

Apart from these several non-standard, proprietary mechanisms and specifications also exist. The most commonly used predominant standards in WSNs include:

| | |
|---|---|
| IEEE 802.15.4 | Standard for low-rate, wireless personal area networks, defines the "physical layer" and the "medium access layer". |
| Zigbee | ZigBee builds upon the 802.15.4 standard to define application profiles that can be shared among different manufacturers. |
| IEEE 802.11 | Standards efforts for low-power Wi-Fi. |
| IEEE 1451 | The objective of this standard is to make it easier for different manufacturers to develop smart sensors and to interface those devices to networks. |
| ISA100 | Addresses wireless manufacturing and control systems in the areas of the: Environment, Technology and life cycle, and Application of Wireless technology. |
| 6LoWPAN | IPv6 over low-power wireless networks, defines an adaptation layer for sending IPv6 packets over IEEE 802.15.4. |
| uIPv6 | uIPv6 is the world's smallest certified open source IPv6 stack provides TCP/IP connectivity to tiny embedded 8-bit micro controllers for low-cost networked device such as sensors and actuators with maintained interoperability and RFC standards compliance. |

Table 7. Predominant standards in field of WSNs

## 11. Existing experimental tools for wireless sensor networks

Research activities in the area of Wireless Sensor Networks (WSNs) need expositive performance statistics about scenario, systems, protocols, gathered data, applications and many more. There are various experimental tools for fulfilling these requirements, someone are in practical use while other one are in literatures. In this part of chapter a glance on currently available simulation tools/frameworks, emulators, visualization tools, testbeds, debuggers, code-updaters and network monitoring tools used for wireless sensor networks is presented (Dwivedi & Vyas, 2011).

### 11.1 Simulator/simulation framework

A simulator is a software that imitates selected parts of the behavior of the real world. Depending on the intended usage of the simulator, different parts of the real-world system are modeled and imitated. The parts that are modeled can also be of varying abstraction level. A wireless sensor network simulator imitates the wireless media and the constraints nodes in the network. Some sensor network simulators have a detailed model of the wireless media including effects of obstacles between nodes, while other simulators have a more abstract model.

**Type of simulation**

Simulators either run as in an asynchronous mode, event triggered mode, or in synchronous mode, where events happen in parallel in fixed time slots (DCG's Sinalgo, 2009):

- *Synchronous simulation*
  The synchronous simulation is purely based on rounds.
- *Asynchronous Simulation*
  The asynchronous simulation is purely event based.

**Categorization of simulators**

A large number of sensor network simulators have been proposed by researchers. In a research contribution WSN Simulators are categorized (Eriksson, 2009) as:

- *Generic Network Simulators*
- *Code Level Simulators*
- *Firmware Level Simulators*

In another research contribution (Shu et al., 2009), simulators have been classified into the following three major categories based on complexity:

- *Algorithm Level Simulators*
- *Packet Level Simulators*
- *Instruction Level Simulators*

Several simulators exist that are either adjusted or developed specifically for wireless sensor networks. Here is a table presenting **63** simulators/simulation frameworks (Table 8).

### 11.2 Emulator or emulation environment

As a networked embedded system, a WSN application involves sensor node hardware, its drivers, operating systems, and networking protocols. As a result, the performance of the WSN application depends on all of these factors in addition to its implementation. An emulator is a special type of simulator whose aims is to enable realistic performance evaluation for WSN applications. Emulation environment or emulators are good choice, in

| | | | | | |
|---|---|---|---|---|---|
| 1. | Network Simulator (NS) | 2. | Mannasim (NS-2 Extension for WSNs) | 3. | DiSenS (Distributed SENsor network Simulation) |
| 4. | (J) Prowler | 5. | LecsSim | 6. | WISDOM |
| 7. | TOSSIM | 8. | OPNET | 9. | Sinalgo |
| 10. | TOSSF | 11. | SENS | 12. | SENSORIA |
| 13. | PowerTOSSIMz | 14. | EmStar/Em* | 15. | Capricorn |
| 16. | ATEMU | 17. | EmTOS | 18. | SIDnet-SWANS |
| 19. | COOJA | 20. | SenQ | 21. | Stargate Simulator (starsim) |
| 22. | GloMoSim (Global Mobile Information Systems Simulation) | 23. | H-MAS (Heterogeneous Mobile Ad-hoc Sensor-Network Simulation Environment) | 24. | JiST/SWANS (Java in Simulation Time/ Scalable Wireless Ad hoc Network Simulator) |
| 25. | QualNet | 26. | SensorSim | 27. | SNSim |
| 28. | SENSE | 29. | Shawn | 30. | SNIPER-WSNSim |
| 31. | VisualSENSE | 32. | NetTopo | 33. | SNAP |
| 34. | AlgoSenSim | 35. | Atarraya | 36. | SimPy |
| 37. | Georgia Tech Network Simulator (GTNetS) | 38. | SSFNet (Scalable Simulation Framework) | 39. | Mule |
| 40. | OMNet++ | 41. | WiseNet | 42. | CaVi |
| 43. | Castalia | 44. | SimGate | 45. | Ptolemy |
| 46. | J-Sim (formerly JavaSim) | 47. | SimSync | 48. | Maple |
| 49. | Mote simulator (motesim) | 50. | SNetSim | 51. | WISENES (WIreless SEnsor NEtwork Simulator) |
| 52. | JiST/SWANS++ | 53. | SensorMaker | 54. | WSNet-Worldsens and WSim |
| 55. | Avrora | 56. | TRMSim-WSN | 57. | LSU SensorSimulator |
| 58. | Sidh | 59. | *PAWiS* | 60. | WSNGE |
| 61. | Prowler | 62. | OLIMPO | 63. | TikTak |

Table 8. Simulator/simulation frameworks specifically designed for WSNs

which WSN applications can be directly run for testing, debugging, and performance evaluation. Additionally, studies on the lower layers (e.g., hardware drivers, OS, and networking) as well as cross-layer techniques can also be done in this environment by plugging the target modules into the emulator. Here is a table which presents **14** emulators:

| | | | | | |
|---|---|---|---|---|---|
| 1. | VMNET | 2. | Freemote | 3. | UbiSec&Sens |
| 4. | ATEMU | 5. | EmPro | 6. | Emuli |
| 7. | Emstar | 8. | NetTopo | 9. | MSPSim |
| 10. | TOSSIM | 11. | OCTAVEX | 12. | MEADOWS |
| 13. | AvroraZ/Avrora | 14. | SENSE | | |

Table 9. Emulators specifically designed for WSNs

### 11.3 WSN data visualization tools

With the increase in applications for sensor networks, data manipulation and representation have become a crucial component of sensor networks. The data gathered from WSNs is usually saved in the form of numerical form in a central base station. There are many programs that facilitate the viewing of these large amounts of data. These special programs are called data visualization tool for WSNs. Visualization tools can support different data types, and visualize the information using a flexible multi-layer mechanism that renders the information on a visual canvas. Here is a table presenting **19** data visualization tools (Parbat et al., 2010) that are especially designed and developed for WSNs applications:

| | | | | | |
|---|---|---|---|---|---|
| 1. | SpyGlass | 2. | TOSGUI | 3. | *Oscilloscope* |
| 4. | MoteView | 5. | MSR Sense | 6. | GSN |
| 7. | TinyViz | 8. | Trawler | 9. | WiseObserver |
| 10. | XbowNet | 11. | SNAMP | 12. | SenseView |
| 13. | MonSense | 14. | Surge Network Viewer | 15. | *MeshNetics WSN Monitor* |
| 16. | NetTopo | 17. | Mica Graph Viewer | 18. | MARWIS |
| 19. | Octopus | | | | |

Table 10. Data visualization tools specifically designed for WSNs

| | | | | | |
|---|---|---|---|---|---|
| 1. | Motelab | 2. | NetEye | 3. | Sharesense |
| 4. | NESC-Testbed | 5. | INDRIYA | 6. | Trio |
| 7. | WUSTL | 8. | CLARITY | 9. | sMote |
| 10. | CitySense | 11. | GNOMES | 12. | CTI-WSN Testbed |
| 13. | Kansei | 14. | WSNTB | 15. | FEEIT WSN Testbed |
| 16. | MISTLAB | 17. | TWIST | 18. | Roulette |
| 19. | Orbitlab | 20. | X-sensor | 21. | BigNet |
| 22. | Emulab | 23. | ENL Sensor Network Testbed | 24. | UCR Wireless Networking Research Testbed |
| 25. | WISEBED (Wireless Sensor Network Testbeds) | 26. | Imote2 Sensor Network Testbed | 27. | SWOON (Secure Wireless Overlay Observation Network) |
| 28. | REALnet | 29. | PICSENSE | 30. | WHYNET |
| 31. | KonTest | 32. | SOWNet | 33. | CENS-Testbed |
| 34. | SANDbed | 35. | IP-WSN Testbed | 36. | SCADDS WSN Testbeds |
| 37. | BANAID | 38. | SenseNet | 39. | Crossbow WSN Testbed |
| 40. | Motescope | 41. | Omega | 42. | GaTech Testbed |
| 43. | Tutornet: A Tiered Wireless Sensor Network Testbed | 44. | CENSE (A Century of Sensor nodes) | 45. | Intel Research Berkeley's 150-mote SensorNet Testbed |
| 46. | WINTeR (Wireless Industrial Sensor Network Testbed for Radio-Harsh Environments) | 47. | FireSenseTB: A wireless sensor networks testbed for forest fire detection (Kosucu et al., 2009) | | |

Table 11. Testbeds used for experimental usage specifically for WSNs

### 11.4 Testbeds for WSN

To achieve high-fidelity in WSN experiments use of testbed is very productive. Testbeds are an environment that provides support to measure number of physical parameters in controlled and reliable environment. This environment contains the hardware, instrumentations, simulators, various software and other support elements needed to conduct a test. Generally, testbeds allow for rigorous, transparent and replicable testing. By providing the realistic environments for testing the experiments, the testbeds bridge the gap between the simulation and deployment of real devices. The testbeds thus deployed can improve the speed of innovation and productive research. Here is a table presenting **47** testbeds, used for experimental purposes in various universities, colleges, research institutions or by individuals (Table 11).

### 11.5 Debugging tools/services/concepts

Due to extreme resource constraints nature, deployment in harsh and unattended environments, lack of run-time support tools and limited visibility into the root causes of system and application level faults make WSNs notoriously difficult to debug. Currently, most debugging systems in WSNs are aimed at diagnosing specific faults, such as detection of crashed nodes, sensor faults, or identifying faulty behavior in nodes. There are few debugging solutions for WSNs available, with a fairly wide range of goals and feature sets. Debuggers for WSNs have been categorized (Tavakoli, 2007) into three distinct categories: source-level debuggers, query-oriented debuggers, and decision-tree debuggers. Here is a table presenting **26** debuggers/debugging concepts/debugging concepts:

| | | | | | |
|---|---|---|---|---|---|
| 1. | Clairvoyant | 2. | $S_2$DB | 3. | ActorNet |
| 4. | Dustminer | 5. | Envirolog | 6. | ANDES |
| 7. | Sympathy | 8. | NodeMD | 9. | EvAnT |
| 10. | FIND | 11. | StackGaurd | 12. | KleeNet |
| 13. | Passive Distributed Assertions (PDA) | 14. | Storage-centric method for Debugging | 15. | Model-based diagnosis for WSNs |
| 16. | Chowkidar | 17. | Marionette | 18. | Post-Deployment Performance Debugging (PD2) |
| 19. | Nucleus-NMS | 20. | REDFLAG | 21. | Declarative Tracepoints |
| 22. | Debugging WSNs Using Mobile Actors | 23. | Monitored External Global State (MEGS) | 24. | SNTS: Sensor Network Troubleshooting Suite |
| 25. | Wringer | 26. | MDB | | |

Table 12. Debugging tools/services/concepts specifically useful for WSNs

### 11.6 Code-updation/reprogramming tool

Large scale WSNs may be deployed for long periods of time during which the requirements from the network or the environment in which the nodes are deployed may change. This may necessitate modifying the executing application or re-tasking the existing application with different sets of parameters, which will collectively refer to as code-updation/reprogramming. The relevant forms of code-updation/reprogramming are (Panta et al., 2009):

- *Remote Multi-hop Reprogramming*
- *Incremental Reprogramming*

Incremental Reprogramming poses several challenges. A class of operating systems, including the widely used TinyOS, does not support dynamic linking of software components on a node. SOS and Contiki, do support dynamic linking, however, their reprogramming support also does not handle changes to the kernel modules. Here is a table presenting **10** code-updaters/reprogramming (Table 13).

| | | | | | |
|---|---|---|---|---|---|
| 1. | Trickle | 2. | Deluge | 3. | Hermes |
| 4. | FlexCup | 5. | Stream | 6. | FIGARO |
| 7. | Zephyr | 8. | MNP (Multi-hop network reprogramming) | 9. | Multihop Over-the-Air Programming (MOAP) |
| 10. | MARWIS (Management ARchitecture for WIreless Sensor Networks) | | | | |

Table 13. Code-updaters/Reprogramming tools specifically designed for WSNs

### 11.7 Network monitoring tools

WSNs are typically composed of low cost tiny hardware devices and tend to be unreliable, with failures a common phenomenon. Accurate knowledge of network health status, including nodes and links of each type, is critical for correctly configuring applications on really deployed WSN and/or WSN testbeds and for interpreting the data collected from them. Here is a table presenting **8** networks monitoring:

| | | | | | |
|---|---|---|---|---|---|
| 1. | Memento | 2. | Sympathy | 3. | LiveNet |
| 4. | NUCLEUS | 5. | HERMES | 6. | Chowkidar |
| 7. | DiMo | 8. | MARWIS (Management Architecture for heterogeneous Wireless Sensor Networks) | | |

Table 14. Network monitoring tools specifically designed for WSNs

## 12. Usability & reliability of experimental tools

The statistics gathered from experimental tools can be realistic and convenient, but due to cost of large number of sensors most researches in wireless sensor networks area is performed by using these experimental tools in various universities, institutes, and research centers before implementing real one. These experimental tools provide the better option for studying the behavior of WSNs before and after implementing the physical one.

Simulators are commonly used for rapid prototyping and also used for the evaluation of new network protocols and algorithms as well as enable repeatability because they are independent of the physical world and its impact on the objects. Moreover, simulations enable nonintrusive debugging at the desired level of detail. In a research contribution various factors have been presented that influences simulation results (Dwivedi et al., 2010).

For successful WSN development cooperation not only between test-beds and simulators but also between simulators is required, however, simulators are usually not designed with cooperation in mind (Li et al., 2010).

## 13. Routing challenges & protocol design issues in WSNs

Routing in WSNs is very challenging due to unique inherent characteristics (energy efficiency and awareness, connection maintenance, minimum resource usage limitation, low

Table 15. Protocol classifications and sub-classifications for WSNs

| SN | Main Category | Sub Categories |
|---|---|---|
| 1. | Classification based on Network Structure (Al-Karaki & Kamal, 2004) | • *Flat-based or Data Centric routing:* In flat-based routing algorithm, all nodes play mainly apply flood based data transferring.<br>• *Hierarchical-based or Cluster based routing:* Hierarchical protocols aim at clusteri cluster heads can do some aggregation and reduction of data in order to save e routing is mainly two-layer routing where one layer is used to select cluster he routing.<br>• *Location-based routing:* Location-based protocols utilize the position information desired regions rather than the whole network. |
| 2. | Classification based on Protocol Operation (Al-Karaki & Kamal, 2004) | • *Multipath-based routing:* This type of routing protocols uses multiple paths inste order to enhance network performance.<br>• *Query-based routing:* In this type of routing protocol destination nodes propaga (sensing task) from a node through the network, and a node with this data sen the query back to the node that initiated the query.<br>• *Negotiation-based routing:* These protocols use high-level data descriptors in orc redundant data transmissions through negotiation. Communication decisions the resources available to them.<br>• *QoS-based routing:* In QoS-based routing protocols, the network has to balance consumption and data quality. In particular, the network has to satisfy certain energy, bandwidth, etc.) when delivering data to the base station.<br>• *Non-coherent & Coherent data-processing based routing:* In non-coherent data proc will locally process the raw data before it is sent to other nodes for further proc |
| 3. | Classification based on Packet Destination (Karl & Willig, 2006) | • *Gossiping and agent-based unicast forwarding:* These schemas are an attempt of w tables in order to minimize the overflow needed to build the tables, as much a stages in which the tables were not built yet.<br>• *Energy-efficient unicast:* These techniques analyze the network nodes distributio transmitting over the link between two nodes and select an algorithm to calcul<br>• *Broadcast and multicast:* Many nodes must collect or distribute the information network (broadcast). In a similar way, sometimes it is necessary to distribute d previously known nodes. This process is called multicast.<br>• *Geographic routing:* This kind of routing appeared due to two main motivations randomly to every node in a given region is called geo-casting; (2) the destinat specified geographically or relatively (with a location service).<br>• *Mobile nodes:* These aspects with motion ability should be considered for wirel mobile sensor nodes, mobile base station, mobile sensed phenomenon or comb |
| 4. | Crossbow (Xbow) classification (Olivares et al., 2007) | • *Basic routing (with normal or improved variants)*<br>• *Reliable routing*<br>• *Low Power routing*<br>• *XMesh routing* |

Table 15. (continues) Protocol classifications and sub-classifications for WSNs

| SN | Main Category | Sub Categories |
|---|---|---|
| 5. | Classification based on State (Eriksson, 2009) | • *Stateful Ad Hoc routing:* Stateful ad hoc routing protocols require node to maint information that is collected using the routing protocol (e.g., through route rec reversing paths taken by the query).<br>• *Stateless Geometric Ad Hoc routing:* These kinds of protocols only track the posit and select among them a neighbor that is likely to be closer to the destination. |
| 6. | Classification based on Epidemic behavior (Akdere et al., 2006) | • *Pull based epidemic algorithm:* A node asks a selected neighbor for new informat new information only if the neighbor has new information.<br>• *Push based epidemic algorithm:* A node with new information sends the informat neighbor.<br>• *Pull-push based epidemic algorithm:* This algorithm is a combination of two mod |
| 7. | Classification based on Sensor Node Architecture (Al-Karaki & Kamal, 2004) | • *Protocols operating on flat topology (WSN consisting Homogeneous nodes)*<br>• *Protocols operating on hierarchical topology(WSN consisting Heterogeneous nodes).* |
| 8. | Classification based on Protocol's initialization point (Biradar et al., 2009) | • *Source-initiated (Src-initiated):* A source-initiated protocol sets up the routing pa the source node, and starting from the source node. Here source advertises the initiates the data delivery.<br>• *Destination-initiated (Dst-initiated):* A destination initiated protocol, on the othe setup from a destination node. |
| 9. | Classification based on how the source finds the destination (Biradar et al., 2009) | • *Proactive:* A proactive protocol sets up routing paths and states before there is traffic. Paths are maintained even there is no traffic flow at that time. This app applications having fixed nodes<br>• *Reactive:* In reactive routing protocol, routing actions are triggered when there disseminated to other nodes. Here paths are setup on demand when queries a approach is best suited for applications mobile nodes<br>• *Hybrid:* This approach combines both techniques. |
| 10. | Classification based on the basis of how to reduce useful energy consumption (Younis & Fahmy, 2004) | • Protocols that control the transmission power level at each node by increasing keeping the network connected.<br>• Protocols that make routing decisions based on power optimization goals.<br>• Protocols that control the network topology by determining which nodes shou network operation (be awake) and which should not (remain asleep). |
| 11. | Cooperative routing (Castillo et al., 2007) | • In this approach, sensor nodes send data to a central node that join the data to of energy consumption. |

to WSNs these design challenges are identified (Dwivedi et al., 2009a; Eriksson, 2009; Al-Karaki & Kamal, 2004; Karl & Willig, 2006; Akyildiz et al., 2002b; Akkaya & Younis, 2005; Wachs et al., 2007).

- Due to the relatively large number of sensor nodes, it is not possible to build a global addressing scheme for the deployment of a large number of sensor nodes as the overhead of ID maintenance is high. Thus, traditional IP based protocols may not be applied to WSNs.
- In contrast to typical communication networks, almost all applications of sensor networks require the flow of sensed data from multiple sources to a particular Base Station.
- Sensor nodes are tightly constrained in terms of energy, processing, and storage capacities. Thus, they require careful resource management.
- In most application scenarios, nodes in WSNs are generally stationary after deployment except for, may be, a few mobile nodes.
- Sensor networks are application specific, i.e., design requirements of a sensor network change with application.
- Position awareness of sensor nodes is important since data collection is normally based on the location.
- Finally, data collected by various sensors in WSNs is typically based on common phenomena; hence there is a high probability that this data has some redundancy.

Visibility (Wachs et al., 2007) is a new metric for WSNs protocol design. The objective of this visibility metric is that "Minimize the energy cost of diagnosing the cause of a failure or behavior".

## 14. Major existing protocols for wireless sensor networks

A lot of protocols has been proposed in various research contributions, some of them are as follows: Rumor, DSR, SER (Stream Enabled Routing), AODV, SPIN (Sensor Protocols for Information via Negotiation) (SPIN-PP, SPIN-EC, SPIN-BC, SPIN-RL), GRAB, Direct Diffusion, GAF, SEER (Simple Energy-Efficient Routing), GBR, ARPEES, TIDD, TEEN, CADR, APTEEN, ACQUIRE, CEDAR, COUGAR, SAR, TinyAODV, PEQ (Periodic Event-driven and Query-based), GEAR, HPEQ (Hierarchical PEQ), MECN, CPEQ (Cluster PEQ), SMFCN, HEAP (Hierarchical Energy Aware Protocol for Routing & Aggregation in Sensor Networks), GF, PEGASIS (Power Efficient Gathering in Sensor Information System), GF-RSST, HPEGASIS (Hierarchical PEGASIS), LEACH, etc.

Some good research contributions (Al-Karaki & Kamal, 2004; Wachs et al., 2007) presents survey on existing WSN Protocols, whereas some other good one are dedicated to comparison, classification and other aspects of WSN Protocols (Dwivedi & Vyas, 2010; Biradar et al., 2009; Al-Karaki & Kamal, 2004; Wachs et al., 2007; Castillo et al., 2007).

## 15. Existing protocol classifications for wireless sensor networks

A careful attention is needed while selecting or proposing a new routing protocols for wireless sensor networks because WSNs are challenging due to the inherent characteristics such as energy efficiency and awareness, connection maintenance, minimum resource usage limitation, low latency, load balancing in terms of energy used by sensor nodes, etc. Various classifications for WSNs are presented in different literatures, at a glance these are (Table 15).

## 16. Protocol evaluation factors

These are the some parameters on which routing protocols must be evaluated during designing new one:

| Evaluation Parameter | Description |
|---|---|
| Power Usage | Sensor node's lifetime is clearly dependent on its power source, thus useful power usage must be which involves: transmitting/receiving data, processing query requests, forwarding queries/data to neighboring nodes. |
| Data Aggregation | Substantial energy savings and traffic optimization can be obtained through data aggregation. |
| Scalability | The possibility to enlarge and reduce the network. |
| Reliability or Fault Tolerance | Fault tolerance is the ability to sustain WSN functionalities without any interruption due to node failures. |
| Latency (delay) and Overhead | Multi-hop relays and data aggregation cause data latency, these important factors influences routing protocol design. |
| Data Delivery Model | Data delivery model (Continuous, Event-driven, Query-driven , Hybrid) (Ahvar & Fathy, 2010) determines when the data collected by the sensor node has to be delivered. |
| Quality of Service (QoS) | Quality service required by the application, involves: length of life time, data reliability, energy efficiency, location-awareness, collaborative-processing, etc. QoS factors will affect the selection of routing protocols for a particular application. |
| Security | Security concerns needs special attention in current era where data stealing and data diddling becomes major issue. |
| Node Deployment option | Node deployment option affects the performance of routing protocol basically in terms of energy consumptions. |
| Topology | Topology of a WSN affects many of its characteristics like; latency, capacity, and robustness. As well as, the complexity of data routing and processing depends on the network topology. |
| Sensor Density and Network Size | Sensor density of nodes affects the degree of coverage area of interest whereas networks size affects reliability, accuracy, and data processing algorithms. |
| Environment or Scenario | A critical parameter, because node and network lifetime is directly dependent on it. |
| Byte Overhead (Saaranen & Pomalaza-Ráez, 2004) | Byte overhead means the total number of bytes in the routing control messages needed to find a route to the sink. For flooding, byte overhead means the total number of bytes in the extra messages flooded throughout the network. In both cases the bytes in the data packets transmitted by nodes along the route from the originating node to the sink node are not counted as overhead. |

Table 16. WSN Protocol evaluation factors

Except these there are exist some common performance metrics, including latency, throughput, success rates, energy consumption and energy load, that must be calculated for the evaluation of routing algorithms.

## 17. Theoretical aspects of major energy efficient protocols

A classification on energy efficient/aware routing protocols is available in a research contribution (Ahvar & Fathy, 2010) which classified this type of protocols into: Energy Saver and Energy Manager. Energy saver protocols decrease energy consumption totally because most of them try to find the shortest path between source and destination to reduce energy consumption. The objective of energy manager protocols is to balance energy consumption in networks to avoid network partitioning. In first approach finding best route is totally based on energy balancing consideration, it may lead to long path with high delay and decreases network lifetime whereas in later approach finding best route only with the shortest distance consideration may lead to network partitioning. A lot of researches were conducted on the energy efficiency/awareness issue, some are presented here (Table 17)

## 18. Security issues in wireless sensor networks

In a survey paper (Dwivedi et al., 2009b) different classes of adversaries, and considers security goals in each scenario (indoor and outdoor) of WSNs, including: sensor nodes, networks of sensor nodes, operating systems, applications, middleware, and internet, are presented. This paper also presents valuable, in-depth recommendations of how to design and implement a security strategy for WSN. A procedure for protecting systems makes sure that the facility is physically secure, provides a recovery/restart capability, and has access to backup files establishing a priority sequence, one would probably want to start from within the firm and work out. Threats and their usual defenses are illustrated in (Figure 4)

Most WSN routing protocols are quite simple thus sometimes even more susceptible to attacks. Most network layer attacks against sensor networks falls under one of the following categories: Selective forwarding, Sinkhole attacks, Sybil attacks, Wormholes, HELLO flood attacks, Spoofed/Altered/Replayed routing information, Acknowledgement spoofing.

Some security issues that must need attention in wireless sensor networks, are as follows: Secure routing, Secure discovery and verification of location, Key establishment and trust setup, Attacks against sensor nodes, Secure group management, and Secure data aggregation.

In the ideal world, a secure routing protocol should guarantee the integrity, authenticity, and availability of messages in the presence of adversaries of arbitrary power. Every eligible receiver should receive all messages intended for it and be able to verify the integrity of every message as well as the identity of the sender. Several countermeasures and design considerations are also proposed in a research contribution (Karlof & Wagner, 2003).

Some mechanisms for authentication and security are based on public key cryptography. Public key cryptography is too expensive for sensor nodes. Security protocols for sensors networks must rely exclusively on efficient symmetric key cryptography. These protocols are too expensive in terms of node state and packet overhead and are designed to find and establish routes between any pair of nodes - a mode of communication not prevalent in sensor networks. Tackling with natural and manmade disasters is only possible with proper planning.

Table 17. Major theoretical aspects of some major energy efficient protocols for WSNs

| S.N. | Energy Efficient Protocol | Major Theoretical Aspects |
|---|---|---|
| 1. | TEEN (Threshold sensitive Energy Efficient sensor Network protocol) (Manjeshwar & Agarwal, 2001) | - First protocol for reactive networks with enhanced efficiency.<br>- Time critical data reaches the user almost instantaneously. Eminently w sensing applications.<br>- Message transmission consumes much more energy than data sensing. S continuously, the energy consumption in this scheme can potentially be network, because data transmission is done less frequently.<br>- The soft threshold can be varied, depending on the criticality of the sens application.<br>- A smaller value of the soft threshold gives a more accurate picture of th increased energy consumption. Thus, the user can control the trade-off b accuracy.<br>- At every cluster change time, the attributes are broadcast afresh and so, required.<br>- The main drawback of this scheme is that, if the thresholds are not reach communicate; the user will not get any data from the network at all and all the nodes die. Thus, this scheme is not well suited for applications w on a regular basis.<br>- Another possible problem with this scheme is that a practical implemen there are no collisions in the cluster. |
| 2. | APTEEN (Adaptive Periodic Threshold-sensitive Energy Efficient Sensor Network Protocol) (Manjeshwar & Agarwal, 2002) | - A Protocol for Hybrid network (inherit best characteristics of both proac<br>- To provide periodic data collection as well as near real-time warnings a<br>- By sending periodic data, it gives the user a complete picture of the netw immediately to drastic changes, thus making it responsive to time critic both proactive and reactive policies.<br>- It offers a flexibility of allowing the user to set the time interval (TC) and attributes.<br>- Energy consumption can be controlled by the count time and the thresh<br>- The hybrid network can emulate a proactive network or a reactive netw time and the threshold values.<br>- The main drawback of this scheme is the additional complexity required functions and the count time. However, this is a reasonable trade-off an and versatility. |
| 3. | HEED (Hybrid Energy-Efficient Distributed clustering) (Younis & Fahmy, 2004) | - An energy-efficient clustering protocol, using residual energy as primar topology features (e.g. node degree, distances to neighbors) as secondar<br>- Here all nodes are assumed to be homogenous nodes (with same initial<br>- It extends the basic scheme of LEACH.<br>- The clustering process is divided into a number of iterations, as well as not covered by any cluster head doubles their probability of becoming a<br>- Since it enable every node to independently and probabilistically decide network, thus cannot guaranteed optimal elected set of cluster heads. |

Table 17. (continues) Major theoretical aspects of some major energy efficient protocols for WSNs

| S.N. | Energy Efficient Protocol | Major Theoretical Aspects |
|---|---|---|
| 4. | H-HEED (Heterogeneous - HEED) (Kour & Sharma, 2010) | - A protocol for heterogeneous WSN.<br>- Cluster head selection is primarily based on the residual energy of each consumed per bit for sensing, processing, and communication is typical energy can be estimated.<br>- Intra cluster communication cost is considered as the secondary parame that a node might fall within the range of more than one cluster head.<br>- Different level of heterogeneity is introduced: 2-level, 3-level and multi- energy.<br>- In 2-level H-HEED, two types of sensor nodes, i.e., the advanced nodes<br>- In 3-level H-HEED, three types of sensor nodes, i.e. the super nodes, ad are used.<br>- In this heterogeneous approach all the sensor nodes are having differen die randomly.<br>- Multi-level H-HEED prolongs lifetime and shows better performance th HEED protocol. |
| 5. | Reactive Energy Decision Routing Protocol (REDRP) (Ying-Hong et al., 2006) | - To solve the problem of limited energy, the loading of nodes have to be<br>- If the energy consumption can be shared averagely by most nodes, then will be enlarged.<br>- This protocol will create the routes in reactive routing method to transm<br>- It uses the residual energy of nodes as the routing decision for energy-a |
| 6. | PEGASIS (Power-Efficient Gathering in Sensor Information Systems) (Lindsey & Raghavendra, 2002) | - A near optimal chain-based protocol and an enhanced descendant of LE<br>- It has two main objectives: increases the lifetime of each node by using allow only local coordination between nodes that are close together so t communication is reduced.<br>- Nodes route data destined ultimately for the base station through intern<br>- In determining the routes only consider the energy of the transmitter an of the receivers.<br>- It assumes that each sensor node can be able to communicate with the b maintain a complete database about the location of all other nodes in th<br>- The method of which the node locations are obtained is not outlined.<br>- It also assumes that all sensor nodes have the same level of energy and time. |
| 7. | Hierarchical-PEGASIS (Savvides et al., 2001) | - Its objective is to decrease the delay incurred for packets during transm<br>- In its concept only spatially separated nodes are allowed to transmit at t<br>- This chain-based protocol with CDMA capable nodes, constructs a chai hierarchy, and each selected node in a particular level transmits data to the hierarchy, that ensures data transmitting in parallel and reduces the<br>- Results shows that this hierarchical extension of PEGASIS performs bet scheme by a factor of about 60. |
| 8. | SHPER (Scaling Hierarchical Power Efficient Routing) (Kandris et al., 2009) | - Enhanced integration of a hierarchical reactive routing protocol.<br>- It supposes the coexistence of a base station and a set of homogeneous s distributed within a delimited area of interest.<br>- Consists of two phases: the initialization phase and the steady state pha<br>- Hard and soft thresholds are utilized in the SHPER protocol as with TE<br>- Best suited in real life applications where imbalance in energy distributi<br>- Network scalability is retained because it adopts both multi-hop routing |

Table 17. (continues) Major theoretical aspects of some major energy efficient protocols for WSNs

| S.N. | Energy Efficient Protocol | Major Theoretical Aspects |
|---|---|---|
| 9. | TREnD (Timely, Reliable, Energy-efficient and Dynamic) (Marco et al., 2010) | - A novel cross-layer WSN protocol for control applications.<br>- The routing algorithm of TREnD is hierarchically subdivided into two p clusters level and a dynamical routing algorithm at node level. This is su hybrid TDMA/CSMA solution.<br>- The protocol parameters are adapted by an optimization problem, whos network energy consumption, and the constraints are the reliability and<br>- It uses a simple algorithm that allows the network to meet the reliability for energy consumption.<br>- It is best fit for industrial environments. |
| 10. | LEACH (Low Energy Adaptive Clustering Hierarchy) (Heinzelman et al., 2000) | - A most popular cluster-based protocol, which includes distributed clust<br>- The idea is to form clusters of the sensor nodes based on the received sig cluster heads as routers to the sink.<br>- It randomly selects a few sensor nodes as cluster-heads and rotates this energy load among the sensors in the network.<br>- Its operation is separated into two phases: setup phase where clusters ar selected and steady state phase where the actual data transfer to the bas<br>- It uses a TDMA/CDMA MAC to reduce inter-cluster and intra-cluster c<br>- Optimal number of cluster heads is estimated to be 5% of the total numb<br>- This protocol is most appropriate for the applications when there is a ne |
| 11. | SEER (Simple Energy Efficient Routing) (Hancke & Leuschner, 2007) | - A protocol that considers energy saving and balancing, with poor idea a<br>- Once the network has been deployed in the area where it is to operate, t packet.<br>- Each node in the network is assumed to have a unique address within th<br>- When a node observes new data, it initiates the process of routing. Two sent: normal data and critical data.<br>- When nodes receive a data message they update the remaining energy v the neighbor that sent the message. Nodes that forward data messages f for minor differences.<br>- If node's remaining energy falls below a certain threshold, it transmits a neighbors to inform them of its energy level.<br>- The sink node periodically sends a broadcast message through the netw neighbors that joined the network to neighbor tables and remove neighb neighbor tables.<br>- Nodes also update remaining energy values stored in the neighbor table |
| 12. | BEAR (Balanced Energy-Aware Routing) (Ahvar & Fathy, 2010) | - An extended version of SEER protocol with some visible difference spec procedure that saves and balance energy consumption in WSNs.<br>- Finds optimal route in energy level and hop count both.<br>- Routing decisions in BEAR are based on the distance to the base-station energy level of nodes on the path towards the base station.<br>- BEAR is better than the SEER protocol in energy managing, due to the f along a balanced path. |

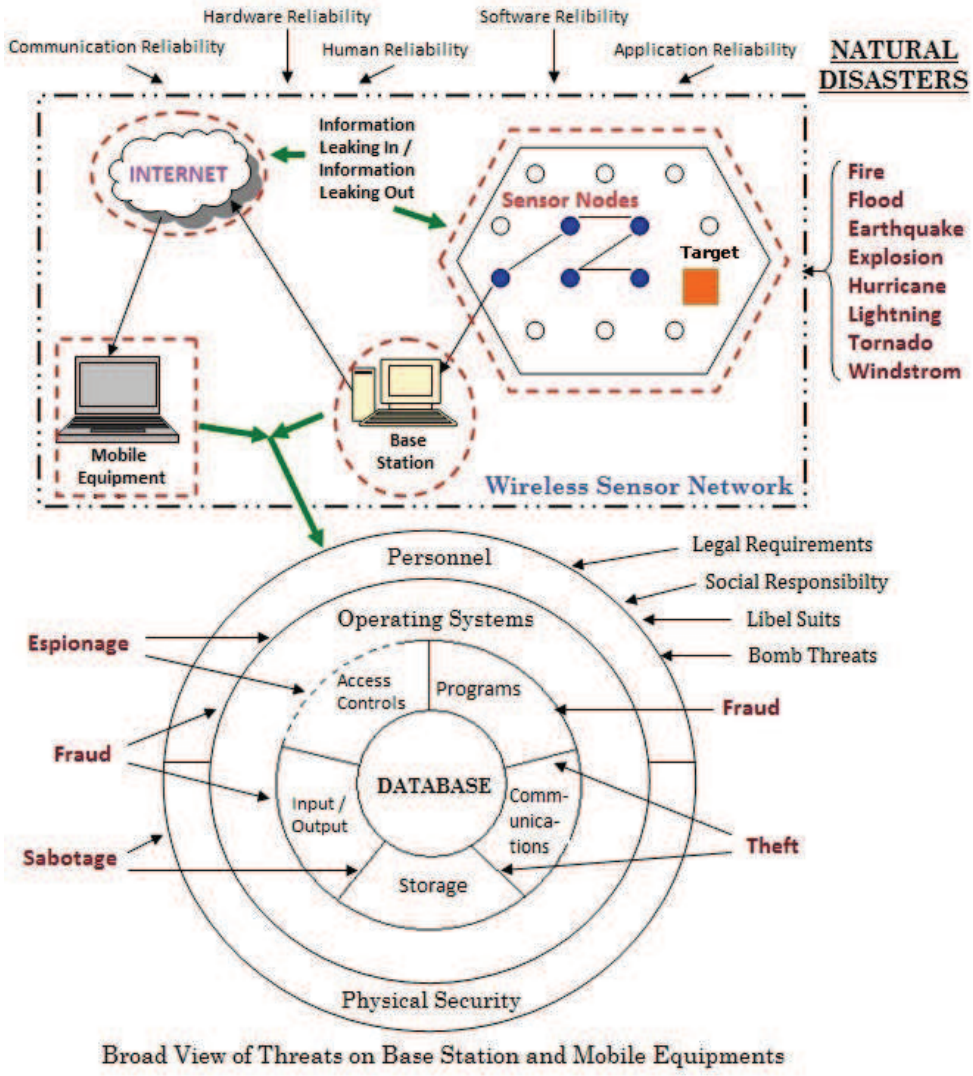



Fig. 4. Security threats and their usual defenses in Wireless Sensor Networks (Dwivedi et al., 2009b)

## 19. Reference

Dwivedi, A. K. & Vyas, O.P. (2010). Network Layer Protocols for Wireless Sensor Networks: Existing Classifications and Design Challenges, *International Journal of Computer Applications (IJCA)*, Vol.8, No.12, Article 6, pp. 30-34.

Tatiana, M. (2010). Mini Hardware Survey. Available from http://www.cse.unsw.edu.au/~sensar/hardware/hardware_survey.html

Imperial college London, U.K. (2007). Body Sensor Networks. Available from http://ubimon.doc.ic.ac.uk/bsn/index.php?m=206

Embedded WiSeNts Platform Survey (2006). Available from http://www.embedded-wisents.org/studies/survey_wp2.html

Senses (2005). Available from http://senses.cs.ucdavis.edu/resources.html

The Sensor Network Museum (2010). Available from http://www.snm.ethz.ch/Main/Homepage

Hempstead, M.; Lyons, M.J.; Brooks D. & Wei, G.-Y. (2008). Survey of Hardware Systems for Wireless Sensor Networks, *Journal of Low Power Electronics*, Vol.4, pp. 1-10.

Fröhlich, A.A. & Wanner L.F. (2008). Operating System Support for Wireless Sensor Networks, *Journal of Computer Science*, Vol.4, No.4, pp. 272-281.

Reddy, A.M.; Kumar, V.A.V.U.P.; Janakiram, D, & Kumar, G.A. (2007). Operating Systems for Wireless Sensor Networks: A Survey, *Technical Report*, IIT Madras, Chennai, India.

Dwivedi, A. K.; Tiwari, M.K. & Vyas, O.P. (2009). Operating Systems for Tiny Networked Sensors: A Survey, *International Journal of Recent Trends in Engineering*, Vol.1, No.2, pp. 152-157.

Manjunath, D. (2007). A Review of Current Operating systems for Wireless Sensor Networks, *Technical Report*, Department of ECE, Indian Institute of Science, Bangalore, INDIA.

Akyildiz, I.; Su, W.; Sankarasubramaniam, Y. & Cayirci, E. (2002). A survey on Sensor Networks, *IEEE Communications Magazine*, Vol.40, Issue:8, pp. 102-114.

Karl, H. & Willig, A. (2003). A Short Survey of Wireless Sensor Networks, *TKN Technical Report TKN-03-018*, Technical University Berlin, Germany.

Dulman, S. & Havinga, P. (2005). Architectures for Wireless Sensor Networks, *Proceedings of the IEEE ISSNIP 2005*, pp. 31-38.

NeTS-NOSS: Creating an Architecture for Wireless Sensor Networks, (2007). Available from http://snap.cs.berkeley.edu/documents/architecture.pdf

Vazquez, J.; Almeida, A.; Doamo, I.; Laiseca, X. & Orduña, P. (2009). Flexeo: An Architecture for Integrating Wireless Sensor Networks into the Internet of Things, *Proceedings of 3rd Symposium of Ubiquitous Computing and Ambient Intelligence 2008*, Springer Berlin/Heidelberg, Vol.51, pp. 219-228, 2009.

Schott, W.; Gluhak, A.; Presser, M.; Hunkeler U. & Tafazolli, R. (2007). e-SENSE Protocol Stack Architecture for Wireless Sensor Networks. *Proceedings of 16th IST Mobile and Wireless Communication Summit*, pp. 1-5.

Biradar, R.V.; Patil, V.C.; Sawant, S.R. & Mudholkar, R.R. (2009). Classification and Comparison of Routing Protocols in Wireless Sensor Networks, *Special Issue on Ubiquitous Computing Security Systems, UbiCC Journal*, Vol.4, pp. 704-711.

Katiyar, V.; Chand, N. & Chauhan, N. (2010). Recent advances and future trends in Wireless Sensor Networks, *International Journal of Applied Engineering Research*, Vol.1, No.3, pp. 330-342, ISSN 0976-4259.

Distributed Computing Group's Sinalgo-Simulator for Network Algorithms. (2009). Available from http://disco.ethz.ch/projects/sinalgo/tutorial/tuti.html

Eriksson, J. (2009). Detailed Simulation of Heterogeneous Wireless Sensor Networks, *Dissertation for Licentiate of Philosophy in Computer Science*, Uppsala University, Sweden, ISSN 1404-5117.

Shu, L.; Hauswirth, M.; Zhang, Y.; Mao, S.; Xiong N. & Chen, J. (2009). NetTopo: A Framework of Simulation and Visualization for Wireless Sensor Networks, *Proceedings of the ACM/Springer Mobile Networks and Applications*.

Parbat, B.; Dwivedi A.K. & Vyas, O.P. (2010). Data Visualization Tools for WSNs: A Glimpse, *International Journal of Computer Applications*, Vol.2, No.1, pp.14-20, ISSN 0975-8887.

Kosucu, B.; Irgan, K.; Kucuk, G.; & Baydere, S. (2009). FireSenseTB: A Wireless Sensor Networks Testbed for Forest Fire Detection, *Proceedings of the* IWCMC.

Tavakoli, A. (2007). Wringer: A Debugging and Monitoring Framework for Wireless Sensor Networks, *Proceedings of the ACM SenSys Doctoral Colloquium*.

Panta, R.K.; Bagchi, S. & Midkiff, S.P. (2009). Zephyr: Efficient Incremental Reprogramming of Sensor Nodes using Function Call Indirections and Difference Computation, *Proceedings of the USENIX*. Available from http://www.usenix.org/events/usenix09/tech/full_papers/panta/panta_html

Dwivedi, A.K.; Patle, V.K. & Vyas, O.P. (2010) Investigation on Effectiveness of Simulation Results for Wireless Sensor Networks, *CCIS*, Springer-Verlag Berlin Heidelberg, Vol.70, pp. 202-208.

Li, Q.; Österlind, F.; Voigt, T.; Fischer, S. & Pfisterer, D. (2010). Making Wireless Sensor Network Simulators Cooperate, *Proceedings of the PE-WASUN'10*, Bodrum, Turkey, pp. 95-98.

Al-Karaki, J.N. and Kamal, A.E. (2004). Routing Techniques in Wireles Sensor Networks: A Survey, *IEEE Wireless Communications (J)*, pp. 06-28.

Karl, H. and Willig, A. (2006). Protocols and Architectures for Wireless Sensor Networks, *Editorial John Wiley & Sons Ltd.*, ISBN 13978-0-470-09510-2.

Akyildiz, I.F.; Su, W.; Sankarasubramaniam, Y. & Cayirci, E. (2002). Wireless Sensor Networks: A Survey, *Computer Networks (Elsevier) (J)*, Vol.38, pp.393-422.

Akkaya, K. and Younis M. (2005). A Survey on Routing Protocols for Wireless Sensor Networks, *Ad Hoc Network (Elsevier) (J)*, Vol.3, pp. 325-349.

Wachs, M.; Choi; Jung, J. II; Lee, W.; Srinivasan, K.; Chen, Z.; Jain, M. and Levis, P. (2007). Visibility: A New Metric for Protocol Design. *Proceedings of ACM SenSys*.

Olivares, T.; Tirado, P.J.; Royo, F.; Castillo, J.C. & Orozoco-Barbosa, L.: (2007). IntellBuilding: A Wireless Sensor Network for Intelligent Buldings. Poster. *Proceedings of 4th European Conference on Wireless Sensor networks (EWSN)*.

Akdere, M.; Bilgin, C.C.; Gerdaneri, O.; Korpeoglu, I.; Ulusoy, Ö. and Cetintemel, U. (2006). A Comparison of Epidemic Algorithms in Wireless Sensor Networks. *Elsevier Journal of Computer Communications*, Vol.29, pp. 2450-2457.

Younis, O. & Fahmy, S. (2004). HEED: A Hybrid, Energy-Efficient, Distributed Clustering Approach for Ad Hoc Sensor Networks. *IEEE transactions on Mobile Computing*, Vol.3, pp. 366-379.

Castillo, J.C.; Olivares, T. & Orozco-Barbosa, L. (2007). Routing Protocols for Wireless Sensor Networks-Based Network. *Technical Report*, Albacete Research Institute of Informatics, University of Castilla, SPAIN.

Tilak, S.; Abu-Ghazaleh, N.B. & Heinzelman, W. (2002). A Taxonomy of Wireless Microsensor Network Models, *ACM SIGMOBILE Mobile Computing and Communications Review*, Vol.6, Issue 2, pp. 28–36.

Saaranen, A. & Pomalaza-Ráez, C.A. (2004). Comparison of Reactive Routing and Flooding in Wireless Sensor Networks, *Proceedings of Nordic Radio Symposium*, Oulu, Finland.

Ahvar, E. & Fathy, M. (2010). BEAR: A Balanced Energy-Aware Routing Protocol for Wireless Sensor Networks. *Wireless Sensor Network*, Vol.2, pp. 793-800.

Manjeshwar, A. & Agarwal, D.P. (2001). TEEN: a Routing Protocol for Enhanced Efficiency in Wireless Sensor Networks. *Proceedings of 1st International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing*.

Manjeshwar, A. & Agarwal, D.P. (2002). APTEEN: A Hybrid Protocol for Efficient Routing and Comprehensive Information Retrieval in Wireless Sensor Networks. *Proceedings of International Parallel and Distributed Processing Symposium (IPDPS)*, pp. 195-202.

Kour, H. & Sharma, A.K. (2010). Hybrid Energy Efficient Distributed Protocol for Heterogeneous Wireless Sensor Network. *International Journal of Computer Applications*, Vol.4, pp. 1-5.

Ying-Hong, W.; Yi-Chien, L.; Ping-Fang, F. & Chih-Hsiao, T. (2006). REDRP: Reactive Energy Decisive Routing Protocol for Wireless Sensor Networks. *Ubiquitous Intelligence and Computing, LNCS*, Vol.4159, pp. 527-535, Springer Berlin/Heidelberg.

Lindsey, S. & Raghavendra, C. (2002). PEGASIS: Power-Efficient Gathering in Sensor Information Systems, *Proceedings of IEEE Aerospace Conference*, Vol.3, pp. 1125-1130.

Savvides, A.; Han, C-C & Srivastava, M. (2001). Dynamic Fine-grained localization in Ad-Hoc Networks of Sensors. *Proceedings of 7th ACM Annual International Conference on Mobile Computing and Networking (MobiCom)*, pp. 166-179.

Kandris, D.; Tsioumas, P.; Tzes, A.; Nikolakopoulos, G. & Vergados, D.D. (2009). Power Conservation through Energy Efficient Routing in Wireless Sensor Networks. *Sensors*, 9, pp. 7320-7342, ISSN 1424-8220.

Marco, P.D.; Park, P.; Fischione, C. & Johansson, K.H. (2010). TREnD: a Timely, Reliable, Energy-efficient and Dynamic WSN Protocol for Control Applications. *Proceedings of Information Communication Conference*.

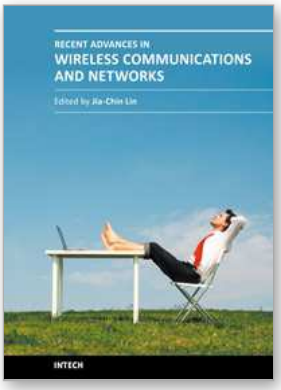
Heinzelman, W.; Chandrakasan, A. & Balakrishnan, H. (2000). Energy-Efficient Communication Protocol for Wireless Microsensor Networks. *Proceedings of 33rd Hawaii International Conference on System Sciences (HICSS '00)*.

Hancke, G.P. & Leuschner, C.J. (2007). SEER: A Simple Energy Efficient Routing Protocol for Wireless Sensor Networks, *South African Computer Journal*, Vol.39, pp.17-24.

Dwivedi, A.K.; Tiwari, M.K. & Vyas, O.P. (2009). A Review of Security in Wireless Sensor networks for Indoor Application Scenario: Prospects and Challenges, *Proceedings of National Conference on Wireless Communication and Networking (WINCON)*, pp. 138-148.

Karlof, C. & Wagner, D. (2003). Secure routing in wireless sensor networks: Attacks and countermeasures. *Proceedings of 1st IEEE International Workshop on Sensor Network Protocols and Applications*.

Dwivedi, A.K. & Vyas, O.P. (2011). An Exploratory Study of Experimental Tools for Wireless Sensor Networks. *Wireless Sensor Network*,Vol. 3, ISSN 1945-3078 (Print), 1945-3086 (Online). Available from http://www.scirp.org/journal/wsn




**Recent Advances in Wireless Communications and Networks**

Edited by Prof. Jia-Chin Lin

ISBN 978-953-307-274-6
Hard cover, 454 pages
**Publisher** InTech
**Published online** 23, August, 2011
**Published in print edition** August, 2011

This book focuses on the current hottest issues from the lowest layers to the upper layers of wireless communication networks and provides â€œreal-timeâ€ research progress on these issues. The authors have made every effort to systematically organize the information on these topics to make it easily accessible to readers of any level. This book also maintains the balance between current research results and their theoretical support. In this book, a variety of novel techniques in wireless communications and networks are investigated. The authors attempt to present these topics in detail. Insightful and reader-friendly descriptions are presented to nourish readers of any level, from practicing and knowledgeable communication engineers to beginning or professional researchers. All interested readers can easily find noteworthy materials in much greater detail than in previous publications and in the references cited in these chapters.

**How to reference**

In order to correctly reference this scholarly work, feel free to copy and paste the following:

A.K. Dwivedi and O.P. Vyas (2011). Wireless Sensor Network: At a Glance, Recent Advances in Wireless Communications and Networks, Prof. Jia-Chin Lin (Ed.), ISBN: 978-953-307-274-6, InTech, Available from: http://www.intechopen.com/books/recent-advances-in-wireless-communications-and-networks/wireless-sensor-network-at-a-glance

INTECH
open science | open minds

**InTech Europe**
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

**InTech China**
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821