Models for integrating wireless sensor networks into the Internet of Things

ABSTRACT

Ubiquitous sensing and unique characteristics of wireless sensor networks (WSNs) have led to an increase in application areas such as smart parking, environmental monitoring, automotive industries and sports. In recent years, WSNs have gained more significance as the foundation infrastructure for the Internet of Things (IoT), which has greatly increased the number of connected objects with instantaneous communication and data processing. However, designing energy-efficient models for integrating WSNs into IoT is a challenging issue due to scalability and interoperability of IoT, and previous approaches designed for WSNs cannot be applied directly. This study proposes two energy-efficient models for WSNs in the IoT environment: a service-aware clustering model where individual sensor nodes are assigned roles based on their service delivery; and an energy-aware clustering model. Performance evaluation shows better energy efficiency, end-to-end delay and network load balance of the proposed models for integrating wireless sensor networks into the IoT protocol compared with low-energy adaptive clustering hierarchy centralized protocol and fuzzy C-means clustering protocol.

EXISTING SYSTEM

In recent years, the Internet has increased to connect smart objects globally. A number of other objects related to the idea of connecting anything to anything have emerged, including the RFID, machine-to-machine (M2M), ubiquitous computing and the IoT. IoT is a recent communication paradigm based on the fact that there will be much more objects than human beings connected to the Internet. Thus, objects such as M2M will be able to communicate independently without any human interaction. It promises to revolutionize the global economy as well as the way we interact with one another. Researchers are currently studying the approaches for the interactions between environment and human beings, machine and human being, ubiquitous computing as well as wearable computing. Smart grids are one of the applications where different smart objects of the grid, which include smart appliances, smart meters, renewable energy resources and other devices, are connected and communicate in order to minimize energy consumption in the grid. Smart grids are diverse by nature as they feed power to different consumers such as homes, industries, commercial buildings and so on. They use different technologies such as Wi-Fi, WSNs, M2M, Zigbee, IP multimedia subsystem and 6LoWPAN . One major requirement of an IoT is that the objects in the network must be able to communicate with one another. In designing the architecture of the IoT, the interoperability, reliability, scalability and compatibility among different objects and their models should be taken into consideration. The reason is that objects that are joining the Internet may need to interact with others in real-time mode . A good IoT model should be adaptive to make things communicate with other objects dynamically. According to Shen *et al.*, a rich sensing stack for IoT devices is required. The authors designed a sensing stack that can improve the execution of context aware application on IoT devices. The stack enables applications to reuse stages of the sensing pipeline reducing the developed effort. Low-energy adaptive clustering hierarchy (LEACH) centralized (LEACH-C) protocol uses a centralized clustering algorithm and the same steady-state protocol as LEACH protocol. During cluster formation, each SN forwards information about its current location and residual energy to a base station (BS). The BS utilises its global information of the network to produce better clusters that require less energy for data transmission. This protocol performs better than LEACH protocol in terms of saving energy. However, at the initialisation of every round, each SN has to report its residual energy to the BS resulting to extra energy consumption. Hoang *et al.* proposed energy-efficient cluster-based protocol called fuzzy C-means clustering protocol (FCMCP) for WSNs. The authors use the fuzzy C-means (FCM) algorithm to form the clusters. The BS calculates and assigns SNs to clusters based on their geographical location. Cluster head (CH) is assigned to the SN having the largest residual energy within the cluster. The proposed protocol extends network lifetime.

DRAWBACKS

- It consumes more energy.
- It has shortest network lifetime.

PROPOSED SYSTEM

Ubiquitous sensing and unique characteristics of wireless sensor networks (WSNs) have led to an increase in application areas such as smart parking, environmental monitoring, automotive industries and sports. In recent years, WSNs have gained more significance as the foundation infrastructure for the Internet of Things (IoT), which has greatly increased the number of connected objects with instantaneous communication and data processing. However, designing energy-efficient models for integrating WSNs into IoT is a challenging issue due to scalability and interoperability of IoT, and previous approaches designed for WSNs cannot be applied directly. This study proposes two energy-efficient models for WSNs in the IoT environment: a service-aware clustering model where individual sensor nodes are assigned roles based on their service delivery; and an energy-aware clustering model. Performance evaluation shows better energy efficiency, end-to-end delay and network load balance of the proposed models for integrating wireless sensor networks into the IoT protocol compared with low-energy adaptive clustering hierarchy centralized protocol and fuzzy C-means clustering protocol.

ADVANTAGES

- It saves more energy.
- It reduces data packets delay.
- It has the longest network lifetime compared with LEACH-C and FCMCP protocols.

SYSTEM REQUIREMENTS

H/W System Configuration:-

- Processor - Pentium –IV
- RAM - 4 GB (min)
- Hard Disk - 20 GB
- Key Board - Standard Windows Keyboard
- Mouse - Two or Three Button Mouse

- Monitor - SVGA

S/W System Configuration:-

- Operating System : Windows 7 or 8 32 bit
- Application Server : Tomcat5.0/6.X
- Programming Language : Java
- Java Version : JDK 1.6 and above