Adaptive Routing Scheme for Energy Harvesting in WSN Based IOT Applications

Mr. K. SIVAKUMAR
Research Scholar, Department of Computer Science,
Rathinam College of Arts & Science,
Coimbatore.

Dr. V. VASANTHI
Assistant Professor, Department of Computer Science,
Rathinam College of Arts & Science,
Coimbatore.

***Abstract*: -** IoT or Internet of Things is a phrase coined to define a system of interrelated physical devices embedded with sensors and network connectivity that enable them to collect, process and exchange data. "Things" in IoT can refer to a wide variety of devices like heart monitoring implants, weather monitoring sensors, biochip transponders, home appliances connected to the internet, automobiles with sensors or any other object with an IP address assigned to it and the ability to transfer data over a network. The convergence of wireless technologies, micro-electromechanical systems and the internet has broken down the wall between operational technology and information technology allowing unstructured data generated by machines to be transmitted over a network and then analyzed.

The major challenge of IoT is energy management, because all the devices, network and applications are operated based on energy. Routing is one of the main problems in WSNs and many solutions have been developed to address this problem. Ensuring efficient routing faces many challenges due to both wireless communication effects and the peculiarities of sensor networks. In this paper, a new approach of routing scheme is used to avoid the energy wastage and it is consumed and harvested by the proposed technique.

Keywords: - Internet, Things, Routing, Sensor Network, Wireless, Adoptive.

*****

1. Concept Of Iot

The term "Internet of Things" (IoT) was first used in 1999 by British technology pioneer Kevin Ashton to describe a system in which objects in the physical world could be connected to the Internet by sensors. Ashton coined the term to illustrate the power of connecting Radio-Frequency Identification (RFID) used in corporate supply chains to the Internet in order to count and track goods without the need for human intervention. Today, the Internet of Things has become a popular term for describing scenarios in which Internet connectivity and computing capability extend to a variety of objects, devices, sensors, and everyday items.

While the term "Internet of Things" is relatively new, the concept of combining computers and networks to monitor and control devices has been around for decades. By the late 1970s, for example, systems for remotely monitoring meters on the electrical grid via telephone lines were already in commercial use. In the 1990s, advances in wireless technology allowed "machine–to–machine" (M2M) enterprise and industrial solutions for equipment monitoring and operation to become widespread. Many of these early M2M solutions, however, were based on closed purpose–built networks and proprietary or industry–specific standards,15 rather than on Internet Protocol (IP)–based networks and Internet standards.

IoT should have the capability to connect and transfer data among billions and trillions of devices. For this to happen seamlessly, it is critical to have a layered architecture in place. The architecture should be highly scalable and flexible to accommodate the wide gamut of components and technologies that form a part of the IoT ecosystem.

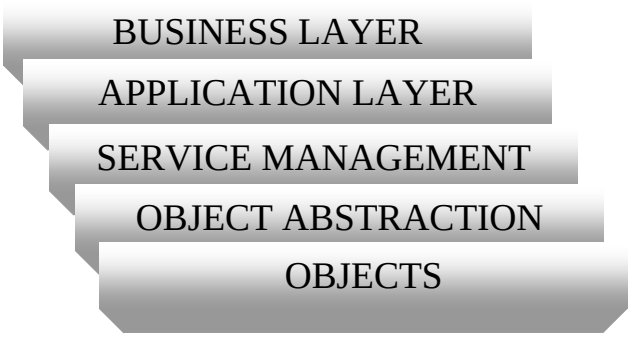


Figure 1: - Layered architecture of IoT

Objects layer, also known as devices layer, comprises the physical devices that are used to collect and process information from the IoT ecosystem. Physical devices include different types of sensors such as those that are typically based on micro-electromechanical systems (MEMS) technology. Sensors could be optical sensors, light sensors, gesture and proximity sensors, touch and fingerprint sensors, pressure sensors, and more.

Object abstraction layer transfers data that are collected from objects to service management layer using

secure transmission channels. Data transmission can happen using any of the following technologies like, RFID, 3G, GSM, UMTS, Wi-Fi, Bluetooth low energy, Infrared, ZigBee. Specialized processes for handling functions such as cloud computing and data management are also present in this layer.

The service management layer acts as middleware for the IoT ecosystem. This layer pairs specific services to its requester based on addresses and names. This layer provides flexibility to the IoT programmers to work on different types of heterogeneous objects irrespective of their platforms. This layer also processes the data that are received from the object abstraction layer. After data processing, necessary decisions are taken about the delivery of required services, which are then done over network wire protocols.

Application layer provides the diverse kinds of services requested by the customer. The type of service requested by the customer depends on the specific use case that is adopted by the customer. ***Business layer*** performs the overall management of all IoT activities and services. This layer uses the data that are received from the network layer to build various components such as business models, graphs, and flowcharts. This layer also has the responsibility to design, analyze, implement, evaluate, and monitor the requirements of the IoT system. This layer has the capability to use big data analysis to support decision-making activities. This layer also performs a comparison of obtained versus expected outputs to enhance the quality of services.

2. Iot Communication Models

Networking and communications models for smart objects include those where exchanged data does not traverse the Internet or an IP-based network.

From an operational perspective, it is useful to think about how IoT devices connect and communicate in terms of their technical communication models. The Internet Architecture Board (IAB) released a guiding architectural document for networking of smart objects (RFC 7452),39 which outlines a framework of four common communication models used by IoT devices. The discussion below presents this framework and explains key characteristics of each model in the framework.

2.1. Device –to-Device Communication

The device-to-device communication model represents two or more devices that directly connect and communicate between one another, rather than through an intermediary application server. These devices communicate over many types of networks, including IP networks or the Internet. Often, however these devices use protocols like Bluetooth, Z-Wave, and ZigBee to establish direct device-to-device communications

These device-to-device networks allow devices that adhere to a particular communication protocol to communicate and exchange messages to achieve their function. This communication model is commonly used in applications like home automation systems, which typically use small data packets of information to communicate between devices with relatively low data rate requirements.

2.2. Device-to-Cloud Communications

In a device-to-cloud communication model, the IoT device connects directly to an Internet cloud service like an application service provider to exchange data and control message traffic. This approach frequently takes advantage of existing communications mechanisms like traditional wired Ethernet or Wi-Fi connections to establish a connection between the device and the IP network, which ultimately connects to the cloud service.

2.3. Device-to-Gateway Model

In the device-to-gateway model, or more typically, the device-to-application-layer gateway (ALG) model, the IoT device connects through an ALG service as a conduit to reach a cloud service. In simpler terms, this means that there is application software operating on a local gateway device, which acts as an intermediary between the device and the cloud service and provides security and other functionality such as data or protocol translation.

The other form of this device-to-gateway model is the emergence of “hub” devices in home automation applications. These are devices that serve as a local gateway between individual IoT devices and a cloud service, but they can also bridge the interoperability gap between devices themselves.

2.4. Back-end Data-Sharing Model

The back-end data-sharing model refers to a communication architecture that enables users to export and analyze smart object data from a cloud service in combination with data from other sources. This architecture supports “the [user’s] desire for granting access to the uploaded sensor data to third parties”. This approach is an extension of the single device-to-cloud communication model, which can lead to data silos where “IoT devices upload data only to a single application service provider’’. A back-end sharing architecture allows the data collected from single IoT device data streams to be aggregated and analyzed.

The back-end data-sharing model suggests a federated cloud services approach or cloud applications

programmer interfaces (APIs) are needed to achieve interoperability of smart device data hosted in the cloud.

The four basic communication models demonstrate the underlying design strategies used to allow IoT devices to communicate. Aside from some technical considerations, the use of these models is largely influenced by the open versus proprietary nature of the IoT devices being networked. And in the case of the device-to-gateway model, its primary feature is its ability to overcome proprietary device restrictions in connecting IoT devices. This means that device interoperability and open standards are key considerations in the design and development of internetworked IoT systems. From a general user perspective, these communication models help illustrate the ability of networked devices to add value to the end user. By enabling the user to achieve better access to an IoT device and its data, the overall value of the device is amplified.

3. Iot Architectures

The building blocks of IoT are sensory devices, remote service invocation, communication networks, and context-aware processing of events; these have been around for many years. However, what IoT tries to picture is a unified network of smart objects and human beings responsible for operating them (if needed), who are capable of universally and ubiquitously communicating with each other.

When talking about a distributed environment, interconnectivity among entities is a critical requirement, and IoT is a good example. A holistic system architecture for IoT needs to guarantee flawless operation of its components (reliability is considered as the most import design factor in IoT) and link the physical and virtual realms together. To achieve this, careful consideration is needed in designing failure recovery and scalability. Additionally, since mobility and dynamic change of location has become an integral part of IoT systems with the widespread use of smart phones, state-of-the-art architectures need to have a certain level of adaptability to properly handle dynamic interactions within the whole ecosystem.

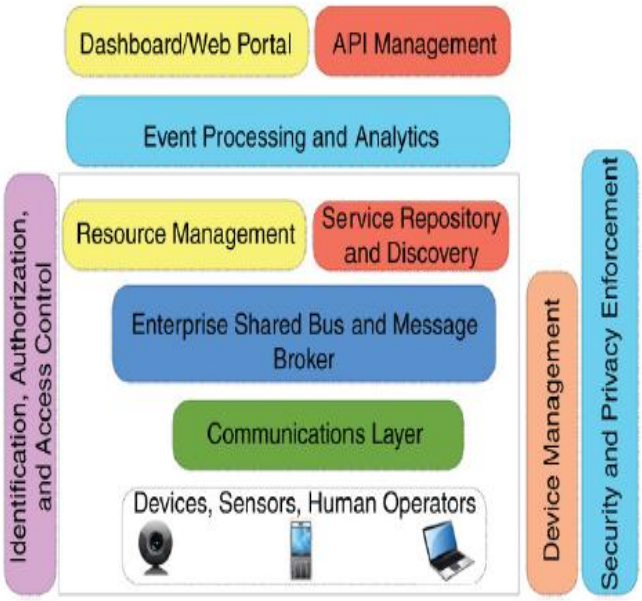


Figure 2: - Architecture of IoT

Different service and presentation layers are shown in this architecture. Service layers include event processing and analytics, resource management and service discovery, as well as message aggregation and Enterprise Service Bus (ESB) services built on top of communication and physical layers. API management, which is essential for defining and sharing system services and web-based dashboards (or equivalent Smartphone applications) for managing and accessing these APIs, are also included in the architecture.

Due to the importance of device management, security and privacy enforcement in different layers, and the ability to uniquely identify objects and control their access level, these components are prestressed independently in this architecture.

3.1. SOA Based Architecture

In IoT, service-oriented architecture (SOA) might be imperative for the service providers and users. SOA ensures the interoperability among the heterogeneous devices. SOA consisting of four layers, with distinguished functionalities as follows:

- Sensing layer is integrated with available hardware objects to sense the status of things
- Network layer is the infrastructure to support over wireless or wired connections among things.
- Service layer is to create and manage services required by users or applications
- Interfaces layer consists of the interaction methods with users or applications.

Generally, in such architecture a complex system is divided into subsystems that are loosely coupled and can be reused later (modular decomposability feature), hence providing an easy way to maintain the whole system by taking care of its individual components. This can ensure that in the case of a component failure the rest of the system (components) can still operate normally.

SOA has been intensively used in WSN, due to its appropriate level of abstraction and advantages pertaining to its modular design.

3.2. API-Oriented Architecture

APIs for IoT applications helps the service provider attract more customers while focusing on the functionality of their products rather than on presentation. In addition, it is easier to enable multitenancy by the security features of modern Web APIs such as OAuth, APIs which indeed are capable of boosting an organization's service exposition and commercialization. It also provides more efficient service monitoring and pricing tools than previous service-oriented approaches.

3.3. Communication Protocols

Seamless connectivity is a key requirement for IoT. Network-communication speed, reliability, and connection durability will impact the overall IoT experience.

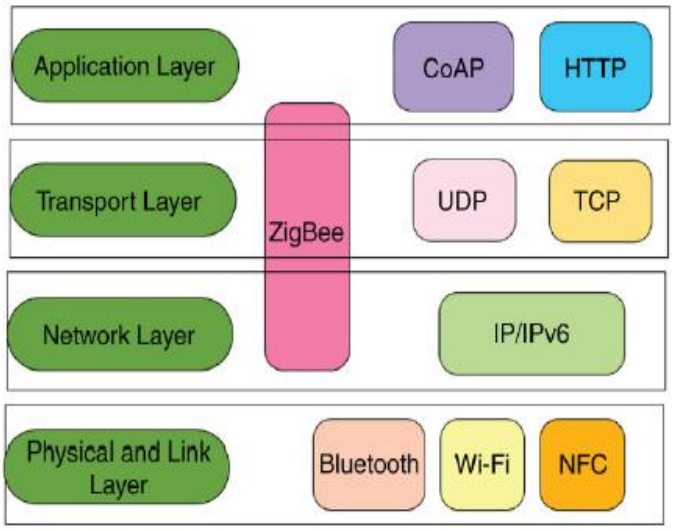


Figure 3: - IoT Communication Protocols

3.3.1. Network Layer

Based on the device's specification (memory, CPU, storage, battery life), the communication means and protocols vary. However, the commonly used communication protocols and standards are listed below:

• RFID
• IEEE 802.11
• Low-power Wireless Personal Area Networks (6LoWPAN) standards by IEFT
• M2M protocols such as MQTT and CoAP
• IP layer technologies, such as IPv4, IPv6, etc.

3.3.2. Transport and Application Layer

Segmentation and poor coherency level, which are results of pushes from individual companies to maximize their market share and revenue, has made developing IoT applications cumbersome. Universal applications that require one-time coding and can be executed on multiple devices are the most efficient.

Protocols in IoT can be classified into three categories:

- general-purpose protocols like IP and SNMP that have been around for many years and are vastly used to manage, monitor, configure network devices, and establish communication links;
- lightweight protocols such as CoAP that have been developed to meet the requirements of constrained devices with tiny hardware and limited resources;
- device- or vendor-specific protocols and APIs that usually require a certain build environment and toolset.

Selecting the right protocols at the development phase can be challenging and complex, as factors such as future support, ease of implementation, and universal accessibility have to be considered.

Additionally, thinking of other aspects that will affect the final deployment and execution, like required level of security and performance, will add to the sophistication of the protocol-selection stage.

4. Existing System & Its Difficulties

The nodes have wireless connectivity and harvest energy from ambient energy sources. The data sink is powered by an unlimited energy supply. In this model, the nodes can be either a sensor or a router. As a sensor node, it generates a data packet to transmit to the sink, and as a router it forwards the packet to the sink via the links that connect sensors and routers.

A sensor can operate as a router to assist other sensors in forwarding packets to the sink. In this work, we consider three typical renewable energy sources, such as: solar, vibration (e.g, moving vehicles) and RF radiation. All nodes can harvest energy from one of these sources with different arrival energy harvesting rates.

To manage the incoming energy, we consider the harvest store-use protocol that allows a node to store electricity energy. If the harvested energy is higher than the node's energy consumption, the excess energy will be stored for later use.

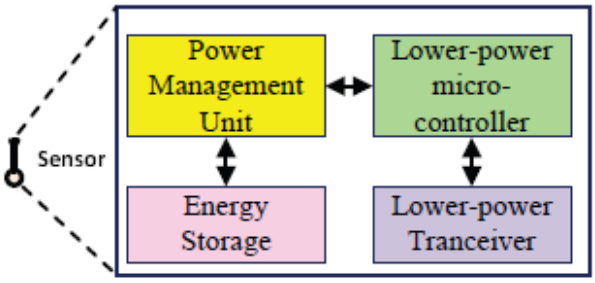


Figure 4: - Existing System Node Configuration

In order to design an effective routing protocol, it is necessary to determine the energy consumed by each node to process a packet. This energy consists of the energy required to transmit, receive or forward the packet on the selected path. In addition, the node has to expend energy to listen for an arrival packet or wait for an incoming event. In the IEEE 802.15.4-based WSNs, the media access control (MAC) sub layer will control nodes to enter into these above operating modes.

The energy harvesting activity can be treated as a stochastic process due to the random nature of ambient energy sources. Assume that the energy harvesting process is independent of the four operating modes (e.g., transmitting, receiving, idle-listening and sleeping) of the nodes. To improve the knowledge of the arrival energy in the harvesting process, it is necessary to develop an energy prediction model. In this work, we adopt the prediction model based on a standard Kalman filter (KF) [18]. The Kalman filter is a recursive algorithm that uses only the estimated state from the previous time step and the current measurement are needed to compute the estimate for the current state. It minimizes the mean square of the estimation error under white noise.

The main disadvantages of the existing system are summarized as follows:

- We jointly address the issues of EE and QoS for IoT applications by developing an energy-harvesting aware routing protocol that is operated at the network layer of IEEE 802.15.4-based networks. The proposed algorithm can adapt to the varying traffic load from the IoT applications, the residual energy and the arrival harvesting energy at sensor nodes,
- We propose an energy prediction model for the arrival harvested energy at the sensor nodes. The stochastic characteristics of the ambient energy sources are taken into account in the model,
- We introduce a new parameter termed as *'extra backoff'*, which can be integrated into the proposed routing algorithm. Based on a combination of the *'extra backoff'* and the energy prediction process, we define the cost metric which can be used to build the routing table and to select the best routes for packet forwarding.

5. Proposed System & Its Contribution

An adaptive scheduling algorithm is evaluated. The work combines the topology and routing improvements with management, synchronization and scheduling techniques.

In an appropriate routing scheme, a lot of random depletion schemes of wireless sensors are reduced and partitioned into grid layouts. In [9] the authors propose a direct grid topology from the source node to the sink node. The sensor network is divided in grid subnets where the transmitter node is selected according to its dependencies on a certain cost parameter that includes the distance to the location of the ideal grid node and the residual power. In [10] a non-uniform grid-based coordinated routing design is presented. Here are used different types of partitioned square shaped grids that divide the sensor network. A load balancing with respect to the residual energy is also implemented.

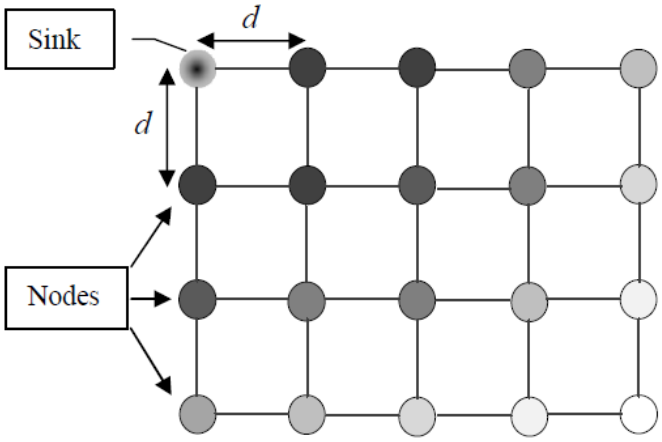
Some studies are made on a classic grid topology where the nodes are arranged on an array layout and all of them participate to route data. In [11] a comparison between four routing algorithms for grid topologies is presented. A similar topology is proposed in this paper but the data is transmitted accordingly with the residual energy of the nodes.

The routes are structured on an adaptive format, considering the leveling of energy spent. It is also studied the influence on the network lifetime of the sink position in a grid WSN. In addition, by taking into account a certain degree of spatial and temporal correlation, a data aggregation technique was proposed in order to increase the network lifetime.

We are interested to study a deployment of the wireless sensor nodes in a situation where the nodes respect the places of an array. They are placed manually at certain locations where the distance between two neighboring nodes is the same (d). The sensor network has a MXN dimension and is similar with the one presented in Fig. 1. Each sensor is identified by its bi-dimensional coordinates, (i, j), where i represents the horizontal index of the sensor with values between 0, M-1 and j represents the vertical index of the sensor taking values between 0, N-1.

For the simplicity of the presentation we choose to select the network sink at the point (0,0). The sink also acts like a sensor and it has unlimited energy. Each node placed in the interior of the grid has 4 neighbors: two high neighbors, node $(i, j + 1)$ and node $(i + 1, j)$, and two low neighbors, node $(i - 1, j)$ and node $(i, j – 1)$. The nodes located on the edge of the grid can have two or three neighbors. A node can transmit only through the smallest

paths, to his low neighbors (Fig. 5). This way the nodes closer to the sink are more used because they transmit all the data from behind.



In a grid WSN the nodes closer to the sink spent the most amount of energy. It is considered that the network lifetime is the same with the lifetime of the first node that dies. Assuming ideal conditions where all the data packets have equal sizes and the transmission is without error, in order to balance the data traffic and to maximize the lifetime of the network, the nodes that have two options in transmitting will choose alternate destination.

7. REFERENCES

[1] Abdelmalik Bachir, Mischa Dohler, Thomas Watteyne, and Kin Kwan Leung. MAC Essentials for Wireless Sensor Networks. IEEE Commununications Surveys & Tutorials, 12(2):222–248, 2010.

[2] Alessio Di Mauro, Davide Papini, Vigo Roberto, and Nicola Dragoni. Introducinga. the Cyber-Physical Attacker to Energy- Harvesting Wireless Sensor Networks. Journal of Networking Technology, 3:139–148, 2012.

[3] Xenofon Fafoutis and Nicola Dragoni. ODMAC: An On-Demand MAC Protocol for Energy Harvesting - Wireless Sensor Networks. In Proceedings of the 8th ACM Symposium on Performance Evaluation of Wireless Ad-Hoc, Sensor, and Ubiquitous Networks (PE-WASUN), pages 49–56. ACM, 2011.

[4] Bhattacharya S, Saifullah A, Lu C, Roman GC. Multi-application deployment in shared sensor networks based on quality of monitoring. In: Proceedings of the 16th IEEE real-time and embedded technology and applications symposium (RTAS'10). Stockholm, Sweden, 2010.

[5] Crossman MA, Liu H. Study of IoT with authentication testbed. In: Proceedings of the 2015 IEEE international symposium on homeland and security (HLS). Waltham, USA, 2015.

[6] Jianguo X, Gang X, Mengmeng Y. Monitoring system design and implementation based on the Internet of Things. In: Proceedings of the 2013 fourth international conference on digital manufacturing and automation. Qingdao, China, 2013.

[7] Barnaghi, P.,Wang,W., Henson, C., and Taylor, K., "Semantics for the Internet of Things: Early Progress and Back to the Future," International Journal on Semantic Web and Information Systems, vol. 8, No. 1, 2012.

[8] YOUNIS., K. A. A. M. 2005. A survey on routing protocols for wireless sensor networks. The Journal of Ad Hoc Networks, 3, 325-349.

[9] PARK, W. L., AND D.-H. CHO. 2012. Fair clustering for energy efficiency in a cooperative wireless sensor network. In 75th IEEE Conference on Vehicular Technology.a. SHAOQING, W. & JINGNAN, N. 2010. Energy efficiency optimization of cooperative communication in wireless

[10] NOURCHENE, B., LAMIA, C. & LOTFI, K. 2011. A Comprehensive Overview of Wireless Body Area Networks WBAN. Int. J. E-Health Med. Commun., 1-30.

[11] MUSZNICKI, B., TOMCZAK, M. & ZWIERZYKOWSKI, P. Dijkstra-based Localized Multicast Topology management in Wireless Sensor Networks. 8th IEEE, IET International Symposium on Communication Systems, Networks and Digital Signal Processing, 2012. IEEE.

[12] D. Lucani, M. Medard, and M. Stojanovic. Underwater acoustic networks: channel models and network coding based lower bound to transmission power for multicast. IEEE Journal on Selected Areas in Communications, 26(9):1708–1719, December 2008.

[13] Stefan, R. (2009) Theory and Practice of Geographic Routing. In: Liu, H., Chu, X.W. and Leung, Y.W., Eds., Ad Hoc and Sensor Wireless Networks: Architectures, Algorithms and Protocols, Bentham Science, Sharjah, 69-88.

[14] HEINZELMAN, W., KULIK, J. & BALAKRISHNAN, H. Adaptive Protocols for Information Dissemination in Wireless Sensor Networks. 5th ACM/IEEE Mobicom Conference August 1999a Seattle, WA. 174- 85.

[15] J. Gutierrez, J. F. Villa-Medina, A. Nieto-Garibay, andM. A´ . Porta-Ga´ndara, "Automated irrigation system using a wireless sensor network and gprs module," Instrumentation and Measurement, IEEE Transactions on, vol. 63, no. 1, pp. 166– 176, 2014.

[16] X. Liu, "A survey on clustering routing protocols in wireless sensor networks," Sensors, vol. 12, no. 8, pp. 11113–11153, 2012.

[17] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," Computer networks, vol. 54, no. 15, pp. 2787–2805, 2010.

[18] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (iot): A vision, architectural elements, and future directions," Future Generation Computer Systems, vol. 29, no. 7, pp. 1645–1660, 2013.