Jakub MAJ, Krzysztof PIOTROWSKI
IHP – Leibniz-Institut für innovative Mikroelektronik, Frankfurt (Oder), Germany

KANGAROO PROTOCOL AS A SENSOR: MONITORING PARAMETERS OF WIRELESS SENSOR NETWORKS

The current development of Wireless Sensor Networks (WSN) has made them capable of handling large amounts of nodes and data, often using network layer protocols enabling multi-hop transmission. This approach means that the network protocol itself can provide a lot of valuable information. This article presents the multi-hop WSN protocol, which, in addition to data transmission, is also classified as a sensor, providing valuable information about the state of connections in the network, the status of packets, information about neighboring nodes or the paths through which packets are sent.

PROTOKÓŁ KANGAROO JAKO SENSOR: MONITOROWANIE PARAMETRÓW BEZPRZEWODOWYCH SIECI SENSOROWYCH

Obecny rozwój Bezprzewodowych Sieci Sensorowych (WSN) sprawił, że potrafią one obsługiwać duże ilości węzłów oraz danych, często wykorzystując do tego protokoły warstwy sieciowej umożliwiające transmisję wieloskokową. Takie podejście sprawia, iż protokół sieciowy sam w sobie dostarczyć może wiele cennych informacji. Ten artykuł przedstawia wieloskokowy protokół WSN, który oprócz transmisji danych, klasyfikowany jest także jako sensor, dostarczający cennych informacji o stanie połączeń w sieci, statusie pakietów, informacji o sąsiadujących węzłach czy też ścieżkach przez które przesyłane są pakiety.

1. INTRODUCTION

Wireless Sensor Networks (WSNs) [1] represent a pivotal technology for real-time monitoring and data collection in various domains, including environmental sensing, industrial automation, and healthcare. Central to the efficient operation of WSNs is the ability to monitor key network parameters, such as topology, packet status, connection quality, and protocol states. Effective monitoring of these parameters ensures reliable communication, optimization of resource utilization, and promptly addressing performance issues.

Multi-hop communication plays a crucial role in facilitating data transmission over multiple hops between sensor nodes and gateways. Unlike traditional single-hop wireless networks, where each node communicates directly with a central base station, Multi-Hop communication in WSNs allows packets to traverse multiple intermediate nodes before reaching their destination. This approach enhances network coverage, making it well-suited for applications requiring long-range communication without high-power radio modules and operation in challenging environments.

This article considers the communication protocol as one of the sensors, providing valuable information about the network and the nodes within a Multi-Hop Wireless Sensor Networks. While communication is the fundamental aspect of Multi-Hop WSN protocols, facilitating data transmission over multiple hops between sensor nodes and sink nodes, monitoring the network parameters is a distinct function that involves observing and analyzing the behavior and characteristics of the network.

In this paper, approaches and methodologies for monitoring network parameters in Multi-Hop WSNs are explored, highlighting the challenges and opportunities in this domain. The investigation into network parameter monitoring intricacies and the exploration of potential solutions and future directions aim to advance the development of effective monitoring mechanisms for Multi-Hop WSNs, ensuring reliable and efficient operation in dynamic and challenging environments.

2. STATE OF THE ART

Numerous methods exist for real-time monitoring of WSNs, encompassing both hardware and software approaches to observe their behavior and parameters, to collect the information about WSNs.

In [2], the authors introduce a network monitoring method based on hardware sniffers. These devices passively collect information about Multi-hop networks without disrupting the WSN. However, this solution requires an additional device operating within the WSN area to gather information. The study focuses on evaluating the sniffer's capability to gather basic network information.

In [3], the authors propose communication metrics aimed at ensuring Quality of Service (QoS) for WSNs, along with a framework to control WSN performance based on these metrics. The framework offers network specification and planning guidelines, continuous network monitoring, and procedures for network repair. While most parameters are analyzed at the sink, some can be analyzed at individual nodes, thus avoiding unnecessary computation or transmission.

In [4], an efficient WSN monitoring method is presented to detect communication issues without exceeding network traffic limits. In this approach, sensors actively monitor their neighbors, reducing the need for central monitoring. Additionally, emphasis is put on local decision-making, where sensors coordinate with neighbors before triggering alarms, thereby minimizing unnecessary traffic.

3. PROTOCOL AS A COMMUNICATION MODULE

The functionality of network parameter monitoring requires an environment where it can operate and gather information. In the proposed solution, the network parameter monitoring functionality is integrated as part of the Multi-Hop protocol for Wireless Sensor Networks named Kangaroo [5].

Kangaroo is a protocol independent of the physical layer, capable of being reconfigured to operate with both short and long-range physical layers, each with different transmission parameters. By default, it is configured to operate on top of SimpleLink Long Range (SLLR) [6], a physical layer encoding technique. This approach extends network coverage not only through Multi-Hop transmission, but also through Long-Range hop-to-hop transmission. SLLR employs 2-(G)FSK (Gaussian Frequency Shift Keying) modulation and trades data rate for sensitivity gains, thereby increasing transmission range through digital coding, assigning a known bit pattern to each incoming bit to the module using DSSS (Direct Sequence Spread-Spectrum). The bit pattern depends on the chosen DSSS (2, 4, or 8) value.

At the physical layer, the CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) [7] MAC layer has been implemented. CSMA/CA is utilized for both broadcast and unicast transmissions to mitigate the probability of packet collisions between neighboring nodes. It enables nodes to monitor the presence of a carrier signal before initiating transmission and, in the case of a busy communication channel, implements a randomized back-off period before attempting another transmission. To enhance the reliability of packet transmission, unicast packets are confirmed by the recipient with Acknowledgment (ACK) packets, and they can be retransmitted in case of being not confirmed.

The protocol is designed to support a tree-based network architecture. During the *Discovery* phase, the network is constructed into a tree structure by flooding the network with *Discovery* packets, originating from the gateway. Nodes maintain a table of preconfigured number of parents to provide redundancy in case of parent failure. The *Discovery* phase is repeated at regular intervals, but subsequent *Discovery* phases refresh the network rather than rebuilding it from scratch. Refreshing involves updating metrics such as RSSI (Received Signal Strength Indicator), number of hops to the gateway, and link quality (metric based on RSSI and number of hops) of parents and children and allows to replace the worst parents with newly discovered ones. Upon receiving a *Discovery* packet, nodes transmit *NetInfo* packets to the gateway through the parent with the best link quality metric. The *NetInfo* packet serves two purposes: introducing the node to the parent and assigning it as its child, and informing the gateway about the path from the node to the gateway. Nodes may change their primary parent after metric refresh during subsequent *Discovery* phases or after an unsuccessful packet confirmation with an ACK. Following a parent change, the node transmits another *NetInfo* packet to the gateway through the new parent. *Node Self Join* is an additional part of the protocol that allows

nodes to join the network and search for a parent if they fail to receive a *Discovery* packet or become disconnected from the network.

Currently, data transmission focuses on the uplink direction. *Data* packets are transmitted directly to gateways if the node is a child of the gateway, or through other nodes in the network via multi-hop communication. Nodes always transmit *Data* packets to their primary parent, which relays those packets until they reach the gateway. Future plans involve implementing the optimized downlink transmission to enable the transmission of packets with configuration parameters from the gateway to the nodes.

4. PROTOCOL AS A SENSOR

While transmitting data is a fundamental task of the protocol, equally important is its capability to monitor network parameters. In this approach, the protocol functions as both a sensor and an environment to be sensed. It collects information about the network, nodes, and connections.

The Kangaroo protocol can monitor and relay information to the gateway regarding paths chosen during the network build phase. As nodes dynamically establish connections and routes during or after network initialization, the protocol meticulously tracks the path selection process. Kangaroo protocol provides real-time insights into the evolving network topology and route preferences by relaying this information to the gateway. Path information is transmitted using the *NetInfo* packet, initiated by each node in the network upon establishing a connection with the parent node, and forwarded by each subsequent parent. Each node forwards a *NetInfo* packet adding its address to the packet. By keeping the gateway informed about chosen paths, the protocol enables efficient coordination and management of data transmission, ensuring that the gateway remains aware of the network's structure and facilitating data processing monitoring and bottleneck detection.

Once operational, the Kangaroo protocol monitors packets transmission and reception. It tracks the flow of packets, logs successful transmissions, and identifies instances of packet drops. Through packet loss analysis, the protocol discerns the underlying causes of packet drops, whether due to protocol failure, transceiver module failure, or environmental interference, thereby providing information on actions to improve the packet transmission process.

Moreover, the Kangaroo Protocol meticulously monitors the machine state of each layer in the protocol stack. By continuously assessing the operational status of MAC and network layers, Kangaroo detects anomalies and failures, enabling prompt issue identification and swift troubleshooting.

After a failure, combined packet and machine state information are stored together, including packet status, packet ID, transmission or reception status, machine state, and buffer information, as a comprehensive dump of error metrics.

RSSI, a key metric for connection quality, is continuously monitored by the Kangaroo Protocol. It assesses connection strength and reliability between parent and child nodes by reading RSSI values of received packets. Nodes provide information about the last RSSI of packets received from parents and the last and average RSSI from children. Additionally, Kangaroo calculates link quality, a metric combining RSSI and hop count in the right proportion, responsible for the selection and possible change of the parent node. Additionally, each node stores the number of currently stored parents and children in both tables, a piece of information provided by the protocol along with RSSI information.

As a future plan, the protocol assumes the creation of dynamic transmission power adjustment functionality, making it valuable to provide current transmission power values. This information would be crucial in case of dynamic transmission power adjustment failure and setting too low transmission power values, rendering nodes unable to reach neighboring nodes.

Collecting all this information aids in solving potential protocol operation problems, primarily by signaling weak points within the protocol or the network that require replacement or strengthening.

Collected information can be provided by nodes in two ways. The first method involves the transmission of error reports once at a predefined time (default: once a day) using *Data* packets of the Kangaroo protocol itself. The second method involves reading selected information using the BLE (Bluetooth Low Energy) protocol, provided the microcontroller and radio module allows for a multi-protocol solution. By default, the protocol operates on the Mars Node board based on the CC1352R MCU, enabling the use of multi-protocol solutions through DMM (Dynamic Multi-protocol Manager) [8]. This solution allows for the reading of any collected information through, e.g. a smartphone connected via BLE to the Mars Node board that shares the radio module between the Kangaroo Protocol and BLE with the use of DMM. There are also plans to attach the Kangaroo at the tinyDSM [9] Data Interface and to include the protocol in the management control on the node and within the network.

5. CONCLUSIONS

This paper has delved into the domain of network parameter monitoring within Multi-Hop Wireless Sensor Networks (WSNs), shedding light on its pivotal role in ensuring robust and efficient communication. Through the exploration of the Kangaroo protocol, we've uncovered a versatile solution capable of not only facilitating data transmission but also serving as a sensor, actively collecting and analyzing critical network insights.

The primary function of the Kangaroo protocol is to allow the transmission of data from end nodes to the gateway through Multi-Hop transmission. It incorporates mechanisms to ensure redundancy and transmission reliability, including the implementation of a robust MAC layer, the presence of multiple parent nodes and dynamic choosing of those, and a self-connection mechanism to the network.

By integrating comprehensive monitoring functionalities, Kangaroo enables real-time assessment of network topology, packet transmission, and machine states, thereby empowering actionable insights to address potential issues promptly, find weaknesses, and streamline the improvement process.

Looking ahead, the proposed enhancements such as downlink transmission implementation and dynamic transmission power adjustment underscore the protocol's commitment to evolving alongside the demands of modern WSN applications. In essence, this paper not only highlights the importance of effective network parameter monitoring but also presents Kangaroo as a promising avenue for advancing the reliability and efficiency of Multi-Hop WSNs in dynamic environments.

REFERENCES

1. Fu, Z., Al-Shareeda M., Ali M., Manickam S., Karuppayah S.: (2023). Wireless sensor networks in the internet of things: review, techniques, challenges, and future directions, Indonesian Journal of Electrical Engineering and Computer Science, 2023
2. Guo X., Gao T., Dong C., Cao K., Nan Y., Yu F.: A Real-time Network Monitoring Technique for Wireless Sensor Networks, 2022 IEEE 12th International Conference on Electronics Information and Emergency Communication (ICEIEC), Beijing, China, 2022.
3. Pereira V., Silva J. S., Monteiro E.: A framework for Wireless Sensor Networks performance monitoring, 2012 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), San Francisco, USA, 2012.
4. Chihfan H., Mingyan L.: Self-monitoring of wireless sensor networks, Computer Communications, 2006.
5. Maj J., Piotrowski K., Michta E.: Kangaroo: Multi-Hop protocol stack for Smart City sensor networks, Uniwersytet Zielonogórski, Instytut Metrologii, Elektroniki i Informatyki, Zielona Góra, Poland, 2022.
6. Hellan S.: CC13xx Long Range Modes Application Report, 2018.
7. Niu L., Liu D.: Performance Evaluation of Unslotted CSMA/CA Algorithm in Wireless Sensor Networks, 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, China, 2020.
8. K. Moorthy: Application Brief: Dynamic Multi-Protocol Manager (DMM), 2018.
9. K. Piotrowski, P. Langendoerfer, and S. Peter, "tinyDSM: A highly reliable cooperative data storage for Wireless Sensor Networks," in Collaborative Technologies and Systems, International Symposium on, 2009.