Performance Evaluation and Optimization of Neighbor Discovery Implementation Over Contiki OS*

Mohamed A. M. Seliem
The Web of Objects Project
Cairo University
Giza, Egypt 12613
Mseliem11@gmail.com

Khaled M. F. Elsayed, Ahmed Khattab
Dept. of Electronics and Communications Engineering
Cairo University
Giza, Egypt 12613
{khaled, akhattab}@ieee.org

***Abstract*— IPv6 Neighbor Discovery (ND) based on RFC 4861 is not designed for non-transitive wireless links. Its heavy use of multicast transmission makes it inefficient and sometimes impractical for IPv6 over Low power Wireless Personal Area Networks (6LoWPAN). Recently, further optimizations have been proposed by Internet Engineering Task Force (IETF) to make ND more suitable for 6LoWPAN. In this paper, we provide an implementation of the most prominent features of the new optimized ND protocol based on RFC 6775 over the Contiki operating system (OS). First, we evaluate the performance of the basic non-optimized IPv6 ND protocol, and analyze its originally implemented functions to set up a good foundation for our implementation and to maintain compatibility. Then, we implement the new optimized Router Solicitation (RS) and Router Advertisement (RA) messaging scheme that reduces the effect of multicasting and unfavorable periodic RA messages. Our results show that the optimized RFC 6775 ND protocol reduces the number of the exchange radio messages in the network by 60-80%. Such optimization alleviates network congestion and saves more power.**

***Index Terms*—neighbor discovery; Contiki OS; performance evaluation; RFC 6775.**

I. Introduction

The Internet of Things (IoT) is the network of physical objects that contain embedded technology to communicate and sense or interact with their internal states or the external environment. According to Gartner, there will be nearly 26 billion devices on the Internet of Things by 2020 [1].

Internet Protocol version 6 (IPv6) would be used to communicate with devices attached to virtually all human-made objects because of the extremely large address space of the IPv6 protocol, and considering the importance of IPv6 as a building block of IoT Network. It is necessary to optimize protocols used with Internet Protocol Version 6 (IPv6) to minimize its overheads and ensure efficiency in large-scale networks.

Neighbor Discovery (ND) protocol is one of the main protocols used with IPv6 to perform functions analogous to the Address Resolution Protocol (ARP), Internet Control Message Protocol (ICMP) Router Discovery, and Router Redirect protocols developed for IPv4. However, it provides many improvements over its IPv4 counterparts.

Neighbor Discovery is based on the exchange of ICMPv6 messages. These messages are used by the ND protocol to achieve its various functions discussed above. The multicast transmission scheme used in the basic IPv6 ND, defined in RFC4861 [2], is not desirable for low-power and lossy wireless networks such as 6LoWPAN). Recently, an optimized version of ND has been proposed in RFC 6775 [3].

In this paper, we implement and evaluate the performance of the most prominent features of the new ND optimization introduced in RFC 6775 [3]. First, we analyze the new optimizations and divide them into separate well-defined components to simplify the implementation. These components are: 1) the use of host-initiated interactions to allow for sleeping hosts, 2) the elimination of multicast-periodic Router Advertisement (RA) messages, 3) the introduction of a host address registration feature using a new option in the unicast Neighbor Solicitation (NS) and Neighbor Advertisement (NA) messages, and 4) a new Neighbor Discovery option to distribute 6LoWPAN header compression context to hosts, and multi-hop distribution of prefix and 6LoWPAN header compression context. Second, we present a first of its kind implementation of the optimized RFC 6775 ND protocol. Our implementation focuses on the most important and beneficial optimizations which are the host-initiated interactions, and the elimination of the effect of multicast-periodic RA messages. Finally, we perform a set of experiments to evaluate the reduction in the number of RS/RA messages in our implementation of the optimized ND. For low number of hosts, the optimized ND sends around 80% less RS/RA messages than the basic IPv6 ND protocol. As the number of hosts increase, this percentage is approximately 60%.

The rest of the paper is organized as follows. SectionII provides an overview of the basic IPv6 ND protocol and its implementation. Section III describes the implementation details of optimized ND protocol. We present the performance evaluation results in Section IV and conclude in Section V.

* This work is part of the ITEA2 Web of Objects Project funded in Egypt by the Egyptian Ministry of Communication and Information Technology.

II. NEGHIBOUR DISCOVERY PROTOCOL OVERVIEW

Neighbor Discovery (ND) provides solutions for a wide set of problems related to the interaction between the nodes attached to the same link. ND defines mechanisms for addressing the following challenges: router discovery, prefix discovery, parameter discovery, addresses auto-configuration, address resolution, neighbor unreachability detection, and duplicate address detection. ND functions are designed to allow hosts to ascertain the ownership of an address or the mapping between link-layer and IP-layer addresses. Nodes use ND to determine the link-layer addresses for neighbors residing on the attached links to (1) find neighboring routers that are willing to forward packets on their behalf, (2) actively keep track of which neighbors are reachable and which are not, and (3) detect changes in link-layer addresses. When a router or a path to a router fails, a host actively searches for a functioning alternative route. Then, the ND protocol quickly purges the cached values that become invalid.

In order to implement and evaluate the gains of the optimized neighbor discovery protocol (RFC 6775) [3], we first investigate the currently implemented RFC 4861 ND protocol[2]. In this section, we present an overview of such a basic ND protocol and its components, how those components are related and how they interact with each other in the current ND implementation in Contiki operating system [6][7]. Then, we highlight the main problems in such an implementation that must be handled before implementing of the optimized ND protocol.

A. The Basic RFC 4861 ND Protocol

The basic neighbor discovery protocol defines five different ICMPv6 packet types:

- Router Solicitation (RS): When an interface becomes enabled, hosts may send out router solicitations that request the routers to generate router advertisements immediately rather than at their next scheduled time.
- Router Advertisement (RA): The routers advertise their presence together with various link and Internet parameters either periodically, or in response to a router solicitation message.
- Neighbor Solicitation (NS): Such messages are sent by a node to determine the link-layer address of a neighbor, or to verify that a neighbor is still reachable via a cached link-layer address.
- Neighbor Advertisement (NA): A response to a neighbor solicitation message. A node may also send unsolicited NA message to announce a link-layer address change.
- Redirect: This message is used by routers to inform hosts of a better first hop for a destination.

Contiki2.6 operating system implements the ND protocol of the Internet Engineering Task Force (IETF) [2]. This implementation is based on the following modules:

- **Uip_nd6 module** that implements the different actions needed to handle the various ND messages listed above, and contains the variable definitions, function declarations and protocol constants of the ND protocol.
- **Uip_ds6 module** that contains the variables definitions, function declarations and network interface and stateless auto-configuration [4] that is used to handle the IPv6 data. It also comprises parts of the neighbor discovery and auto configuration state machines.

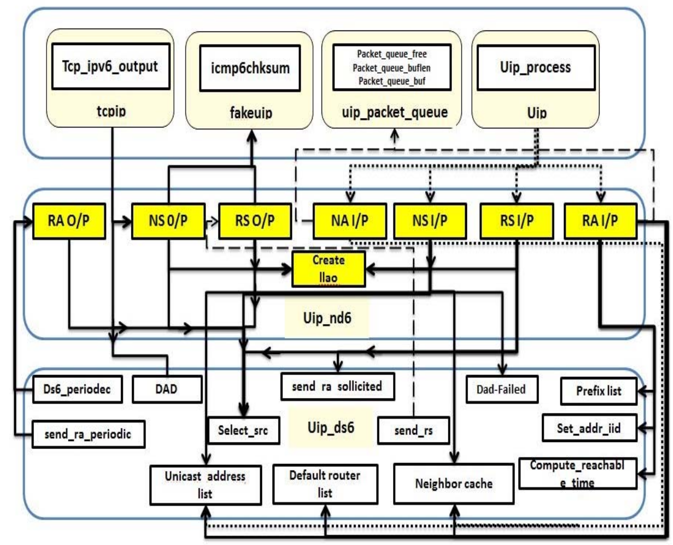

Figure 1: Neighbor discovery components scheme.

Figure 1 shows the relationship between the different neighbor discovery implementation functions included in the **Uip_nd6** and **Uip_ds6 modules** and the related modules such as the **Uip_Tcpip, and Uip_packet_queue modules** which contain the data structure used by the node and affected by ND messages.

B. Asessment of the Basic RFC 4861 ND Protocol

We thoroughly analyzed the basic ND protocol RFC 4861 [2] as implemented in Contiki 2.6. We identified two main issues in this implementation that affects the performance of the ND message exchange and operation.

1) RPL and Neighbor Discovery Contradiction

The Routing Protocol for Lossy networks (RPL) [4] is adopted as the default routing mechanism in Contiki. RPL exchanges ICMPv6 messages (e.g. DAG Information Solicitation (DIS), DAG Information Object (DIO), and DAG Advertisement Object (DAO)) between nodes to construct the routing table for each router node and construct a Data Acyclic Graph (DAG) for the entire network from a network layer point of view. Meanwhile, the ND protocol exchanges ICMPv6 messages between hosts and routers at the link layer; we found that RPL messages duplicate ND protocol messages in some of their function. In order to clearly demonstrate the ND protocol behavior, and focus on the link layer only, we deactivate RPL and all its related functions in order to closely monitor the ND function performance and show all types of messages exchange between client and server.

2) Prefix Cache Access

Prefix discovery is the process through which hosts learn the ranges of IP addresses that reside on-link and can be reached directly without going through a router. Given that we deactivate RPL protocol, we enforce the nodes to obtain the prefix information by enabling the routers to access the prefix cache contrary to typical router behavior in RPL environment.

III. IMPLEMENTATION OF ND PROTOCOL OPTIMIZATION

Neighbor Discovery optimization introduced in RFC 6775[3] targets the evolved host-to router interaction that allows for sleeping nodes and avoids using multicast ND messages except when necessary. The protocol is designed such that the host to router interaction is not affected by the configuration of the 6LoWPAN Figure 2 summarizes of the required algorithm of the new optimized implementation functions. The main features in these optimizations are:

- RS/RA messages optimizations: This part of optimization targets minimizing the overhead by avoiding the use of multicast flooding, and reducing the use of link-scope multicast messages. Host-initiated refresh of RA information by RS message transmission. This removes the need for periodic or unsolicited RA messages from routers to hosts.
- NS/NA messages optimizations: This part optimizes the interfaces between hosts and their default routers and provides support for sleeping hosts. It implements new address registration mechanism using a new Address Registration Option (ARO) between hosts and routers, this removes the need for routers to use multicast NS to find hosts and supports sleeping hosts.

Also there are parts of this optimization that can be substitutable by a routing protocol such as: (1) Multi-hop distribution of prefix and 6LoWPAN header compression context, (2) Multi-hop Duplicate Address Detection. A routing protocol can provide a good alternative for such mechanisms. Thus, if substitution is intended then multi-hop prefix distribution (the ABRO) and the 6LoWPAN Context Option (6CO) for distributing header compression must be substituted [3] (section 1.4).

A. RA/RS Messages Optimization

Periodic RA messages add overhead to the network especially when the network is lossy by nature like 6LoWPAN networks. RFC 6675[3] suggests that allowing the host to start the request for this information will lead to a significant decrease in the number of RA messages in the network. However, it will lead to an increase in the number of RS messages. In our implementation, routers send solicited RA messages only. According to RFC 6775 sections 5.2 and 5.3 [3], nodes send RS messages to solicit new RA message to update its network information in one of the following cases: (1) interface initialization, (2) when the default router list is empty, or (3) when one of the node's default routers becomes unreachable.

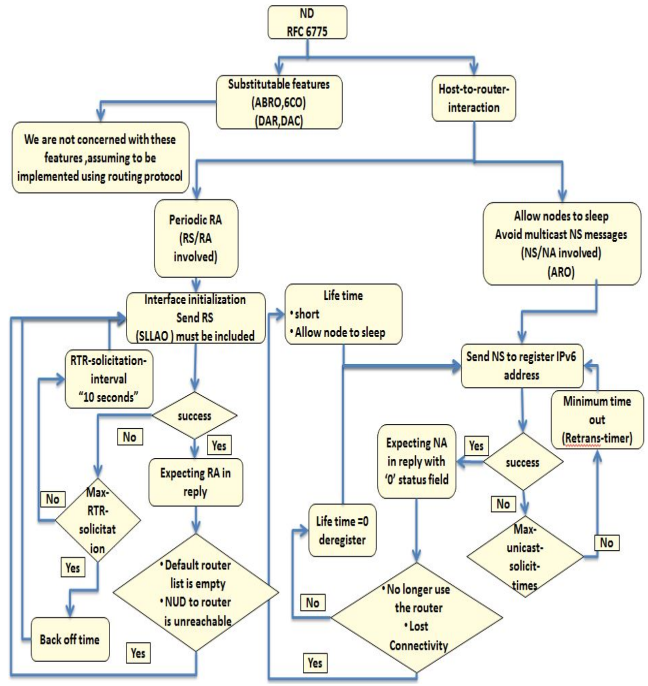

Figure 2: Optimized ND protocol tasks extracted from RFC 6775.

B. Optimized RA/RS Messages Implementation

In order to implement the RFC 6675 RS/RA optimization, we modify the UIP_nd6 and UIP_ds6modules as follow:

1) UIP_nd6 Module

We is add a new protocol constant (MAX_RTR_SOLICITATION_INTERVAL=60s) to truncate the increase in the retransmission timer of an RS message. We modify (RTR_SOLICITATION_INTERVAL=10s) to increase the interval between transmitted RS messages. We modify the function that handles the RA message transmission in order to be able to transmit unicast messages. Note that there is no problem to obtain the destination unicast address as we added the source link local address optional header (SLLAO) in the received RS message.

2) UIP_ds6 Module:

Our changes in this module target the following functions:

- **Uip_ds6_periodic:** This function periodically checks the node's data structure (e.g. neighbor cache, prefix list, default router list, etc.), and triggers the transmission periodic RA message. We are disabling this periodic function and all its effects and interactions.
- **Uip_ds6_inti:** This function initializes node's interface. According to RFC 6775, a node does not need to join the solicited node multicast address to avoid multicasting.
- **Send_ra_solicited:** This function is implemented to schedule the transmission of solicited RA messages with the new optimization protocol constants.
- **Send_rs:** This function is implemented to transmit RS messages in interface initialization and retransmissions. A host performs a truncated binary exponential back-off of the retransmission timer per subsequent retransmission.

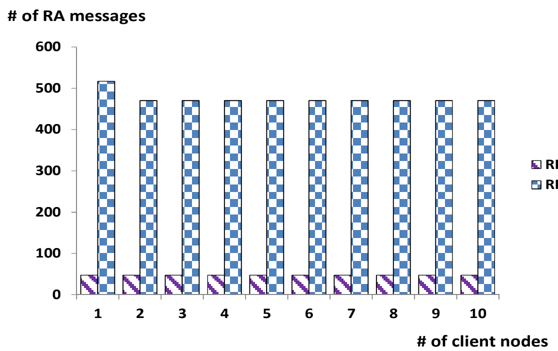

Figure 3: The number of RA messages of both RFC 4861 and RFC 6775 implementations for different number of client nodes.

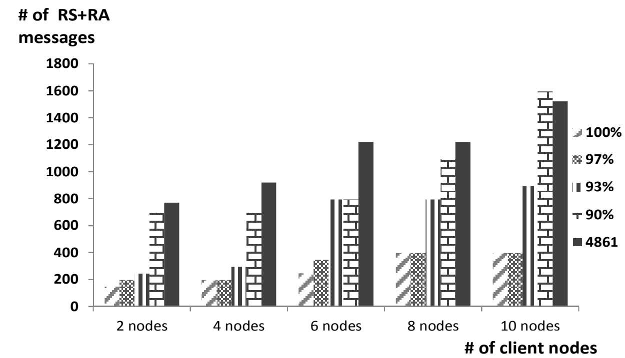

Figure 4: The effect of nodes reachability on the number of RS/RA messages.

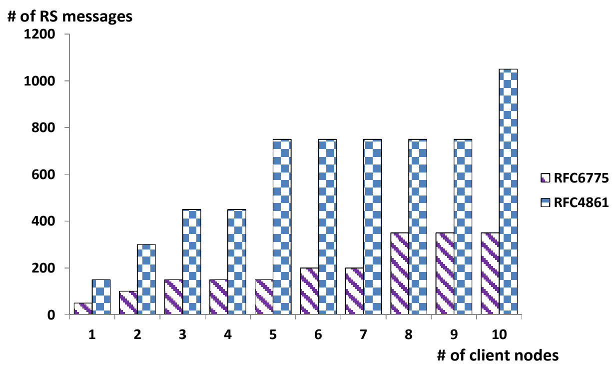

Figure 5: The number of RS messages of both RFC 4861 and RFC 6775 implementations for different number of client nodes.

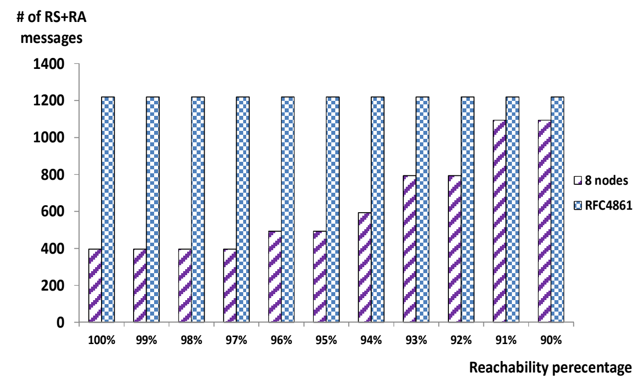

Figure 6: The number of RS/RA messages for different reachability percentage with 8 host nodes.

- **Tcpip_event_handler**: This process handles the RA delay timer expiration by calling **uip_nd6_ra_output** function to transmit a solicited RA message.

IV. Evaluation of Optimized RA/RS Implementation

We evaluate the performance of our implementation of the optimized RFC 6775 ND protocol. We use the same configuration, and the same type of nodes that are used to evaluate the basic IPv6 ND protocol. We consider a circular topology in which the host nodes are evenly distributed over the perimeter of a circle with the router node on its center. The main target of the following experiments is to assess the effect of the implemented RS/RA optimizations on the number of radio messages exchanged in the network, and how these messages affected by the network factors such as the number of host nodes, and unreachability of network nodes, for an hour-long simulation.

Figure 3 shows the number of RA messages for basic IPv6 ND and the optimized ND protocols. Our implementation of optimized ND protocol reduces the transmission of RA messages by 90%. This is achieved by disabling the periodic RA messages and the fast termination of solicitation process after receiving RA message. The results indicate that the number of RA messages in both implementations does not depend on the number of hosts since we assume only one router.

We next study the impact reachability of the nodes in the network and how it affects the number of RS/RA messages. Figure 4 shows the sum of the RS/RA messages for 100%, 97%, 93% and 90% reachability of the nodes. During an hour-long simulation, the host nodes cannot reach its default router for 60*(100-*X*)/100 minutes, where *X* is the reachability percentage. Figure 4 also shows the slight increase in the number of RS/RA messages with the decreasing in the reachability percentage. The default router unreachability is the main reason for this increase as nodes need to transmit RS messages in such a case as we mentioned before. It also shows the significant rule of the optimization in reducing the number of messages. However, there is a point of interests where the RS/RA messages exceed the old implementation of ND protocol (10 nodes) due to the increase in the number of host nodes which means more transmissions of RS messages at interface initialization, and more retransmissions in case of unreachability.

We next evaluate the change in the number of RS messages with the number of host nodes. Figure 5 shows the number of RS messages for both ND protocol implementations. The number of RS messages increases with increasing number of host nodes. However, the optimized ND protocol shows a better behavior in the transmission and reception of RS/RA messages. It is to be noted that the condition for stopping the transmission of new RS messages is reception of RA message which is tightly controlled in the new optimizations.

The above results are obtained when the all the nodes in the network assumed to be reachable for all the time. In the case of one of default routers is unreachable discussed in III.A, the RS message will be retransmitted.

Figure 6 considers an 8-hosts network and studies the effect of the reachability and its effect on the number of RS/RA message. We can control the reachability of the nodes in our simulation by either: 1) control the transmission and receiving ranges of the nodes; OR 2) Using Mobility package to move nodes with pattern to be unreachable. In our simulation we decrease the nodes reachability with 1% step to measure the behavior of RS/RA messages. The figure depicts a non-linear increase in the number of RS/RA messages. This is because message transmission and terminating are affected by other factors such as 1) the delay in sending messages, as nodes have to wait for random time before transmitting any message to avoid collisions, 2) the retransmission of RS messages with back-off delay time, and 3) the network topology as stated in RFC 6775 [3] at section 1.2. These optimizations are useful for route-over and mesh-under configurations in Mesh topologies. However, star topology configurations will also benefit from the optimization due to reduced signaling, robust handling of the non-transitive links. A suggested solution to enhance the performance of the optimized protocol is to consider sensors deployments such that at least 1 reachable router exist for each group of say 7-10 host nodes, and construct the network topology to meet the above requirements.

In summary, the optimized ND protocol transmits less number of RS/ RA messages as compared with the basic ND protocol. At 100% reachability, the optimized ND transmits 70% less messages compared with the basic IPv6 ND protocol, and 20% less messages at 90% reachability.

V. CONCLUSIONS AND FUTURE WORK

In this paper, we have implemented and evaluated the gains of the optimized neighbor discovery protocol compared to the basic IPv6 ND protocol. We have implemented the optimized ND protocol over Contiki OS v2.6. The evaluation results, obtained using Cooja simulator, indicate that our implementation enhances the behavior of ND by reducing multicasting which will increase the energy efficiency especially in 6LoWPAN networks, and allow host nodes to start direct interaction with routers. By allowing a host to start interaction with the router, the network has better control on radio message exchange. The reachability of node affects the behavior of ND protocol. The nodes need to retransmit RS messages to find new default router. Our future work in progress includes the implementation of the only remaining feature of RFC 6775 which targets NS/NA optimization.

REFERENCES

[1] Janessa Rivera, Rob van der Meulen, “Gartner Says The Internet Of Things Installed Base will grow to 26 Billion Units By 2020,” Stamford, conn., December 12, 2013.

[2] T. Narten, E. Nordmark, W. Simpson, and H. Soliman, “ Neighbor Discovery for IP version 6 (IPv6),” RFC4861, IETF Network Working Group, Sep. 2007.

[3] Z. Shelby, Ed, S. Chakrabarti, E. Nordmark,and C. Bormann, “Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs),” RFC6775, Internet Engineering Task Force, Nov. 2012.

[4] T. Winter and P. Thubert, “RPL: IPv6 Routing Protocol for Low power and Lossy Networks,” RFC 6550, IETF ROLL, Mar. 2012.

[5] S.Thomson, , T.Narten, and T. Jinmei, “IPv6 Stateless Address Autoconfiguration,” RFC 4862, IETF Network Working Group, Sep. 2007.

[6] A. Dunkels, B. Gronvall,and T. Voigt, , “Contiki: a lightweight and flexible operating system for tiny networked sensors,” in Proc. of IEEE Local Computer Networks, Nov. 2004.

[7] A. Dunkels, B. Grönvall, and V. Thiemo, “Contiki a lightweight and flexible operating system for tiny networked sensors,” in Proc. of IEEE Workshop on Embedded Networked Sensors, Tampa, FL, Nov. 2004.

[8] J.-G. Ko, A.Terzis, S. Dawson-Haggerty, D. E. Culler, J. W. Hui, and P. Levis, “Connecting low-power and lossy networks to the Internet,” IEEE Communications Magazine, pages 96 - 101 , April 2011.