



**Fast Device Discovery for Remote Device Management in Lighting Control Networks**

Arial, **Bold** 16pt

Sang-Il Choi\*, Sanghun Lee\*, Seok-Joo Koh\*, Sang-Kyu Lim\*\*,

Insu Kim\*\*, and Tae-Gyu Kang\*\*

Arial, 8.5pt

Arial, **Bold** 9pt



**Abstract**

The Remote Device Management (RDM) protocol is used to manage the devices in the lighting control networks. RDM provides bi-directional communications between a controller and many lighting devices over the DMX512-A network. In RDM, using a simple binary search scheme, which is based on the 48-bit unique ID (UID) of each device, discovers the lighting devices. However, the existing binary search scheme tends to require a large delay in the device discovery process. In this paper, we propose a novel partition-based discovery scheme for fast device discovery in RDM. In the proposed scheme, all devices are divided into several partitions as per the device UID, and the controller performs device discovery for each partition by configuring a response timer that each device will use. From numerical simulations, we can see that there is an optimal number of partitions to minimize the device discovery time for a given number of devices in the proposed scheme, and also that the proposed partition-based scheme can reduce the device discovery time, as compared to the existing binary search scheme.

Arial, **Bold** 9pt

Times New Roman, 9pt

**Keywords**

Device Discovery, Partition-Based, RDM



Arial, **Bold** 14pt

**1. Introduction**

Times New Roman, 9.5pt

One of the primary issues in the lighting control network is how to effectively manage a lot of devices within the network. Some protocols for device management have so far been made in the PLASA Technical Standards Program (TSP) [1], which includes the Digital Multiplex 512-A (DMX512-A) [2] and Remote Device Management (RDM) [3].

The RDM protocol is performed based on the polling system, in which a controller initiates communication and the devices will respond to the request of controller. RDM is used for the discovery and management of devices that are connected through the DMX512-A link. For device discovery, the controller uses a simple binary search scheme, based on the 48-bit unique ID (UID). However, this device discovery scheme tends to require a lengthy discovery time due to exploring the 248 theoretically

※ This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/>by-nc/3.0/) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Arial, 5.5pt

Manuscript received Month 06, 20XX; accepted Month 06, 20XX.

**Corresponding Author:** Seok-Joo Koh (sjkoh@knu.ac.kr)

\* School of Computer Science and Engineering, Kyungpook National University, Daegu, Korea (overcycos@gmail.com)

\*\* LED Communication Research Team, Electronics and Telecommunications Research Institute, Daejeon, Korea (sklim@etri.re.kr)

\*\*\* LED Communication Research Team, Electronics and Telecommunications Research Institute, Daejeon, Korea (klim@kips.or.kr)

Arial, 5pt

Calibri, **Bold, 6**pt

possible devices, even though a controller is usually able to handle only tens of thousands.

Aria, 6pt

In this paper, we propose a *partition-based* discovery scheme for fast device discovery in RDM. In the proposed scheme, all devices in the RDM network are divided into several partitions as per the device UID, and the controller performs the device discovery for each partition by using a suitable response timer for the devices.

This paper is organized as follows: Section 2 discusses the existing device discovery scheme. In Section 3, we propose the fast device discovery scheme for RDM. In Section 4, we analyze and compare the existing and proposed schemes in terms of device discovery times. Section 5 concludes this paper.

Arial, **Bold** 14pt

2. Existing Device Discovery Scheme

In RDM [3], a simple binary search scheme is used for device discovery, which is based on the 48-bit UID of each device. Fig. 1 shows the existing device discovery scheme, in which a controller performs a binary search for devices with the UIDs ranging from 0 to 248.

Arial 12pt

2.1 Wavelet Filterbank Theory

Initially, the controller sets the UID *search range* with (*lower, upper*) and sends a discovery message to all of the devices in the RDM network. If the controller responds to a single response message, the corresponding device will be successfully found. Otherwise, if two or more devices have responded at the same time, the controller cannot process the multiple response messages due to the characteristics of the DMX512-A link [2]. This is because the DMX512-A link does not use the packet scheduling functionality and thus, multiple response messages will be simply corrupted and discarded by the controller. In this case, the controller will further divide the UID search range into the two parts.



Times New Roman, 9.5pt

Times New Roman , **Bold** 9.5pt

**Fig. 1.** Existing device discovery in Remote Device Management (RDM).

Arial, 7pt

3. Proposed Device Discovery Scheme

Arial, **Bold** 14pt

Arial, 6pt

The existing device discovery scheme gives a lengthy device discovery time because the 48-bit UID search range (248) tends to induce very deep tree levels along the binary search tree. Thus, we propose a new device discovery scheme, called *partition-based* device discovery.

In the proposed scheme, all devices in the network will be grouped into *Np* partitions, as shown in Fig. 2.



**Fig. 2.** Partitions of devices in Remote Device Management (RDM).

Given the number of partitions (*Np*), the partition of a device (0, 1, .., *Np*-1) is determined by using the *modulo* (%) operator and the device UID, as follows:

Times New Roman, *Italic* 9.5pt

*Partition of device UID = UID % Np.*

Then, the proposed scheme is performed for each partition, as shown in Fig. 3.



**Fig. 3.** Proposed partition-based device discovery.

For each partition *k*, the controller first sends a discovery message to all devices. Then, all of the devices that belong to the partition shall respond to the controller. If two or more response messages are generated at the same time, the controller will fail to discover those devices due to collisions. Thus, to reduce the possibility of multiple responses, we employed the *Maximum Response Timer* (*MRT*) for devices. That is, when a device receives a discovery message from the controller, it responds to the controller only after a certain waiting time, which is randomly generated with a range of (0, *MRT*)by the device. This timer will facilitate in reducing the probability of multiple responses from many devices, and thus will increase the possibility of a single (successful) response.

In Fig. 3, the information on the current partition (*Np* and *k*) and *MRT* shall be included in the discovery message that is sent by the controller. The responses from devices can be classified into the following three cases:

● *Fail Response*, in which all of the responses have arrived at the controller at the same time and thus they are corrupted. The controller shall re-discover the devices after increasing the *MRT* timer by the *Delay Increase Interval* (*DII*).

● *Success Response*, in which some of the responses have arrived at the controller at different time intervals. However, the remaining devices shall be re-discovered.

● *No Response*, which means that there are no more devices to be discovered in the partition. Go to the next partition.

In the case of *Fail Response*, the controller adjusts the *MRT* timer to *MRT* + *Delay Increase Interval (DII)*, and then sends a discovery message to the devices again, as shown in Fig. 3. In the *Success Response* case, some devices are found, but there are more devices to be discovered in the partition.

These procedures will be repeated for each partition, until all of the devices in the network have been discovered.

Times New Roman 9.5pt

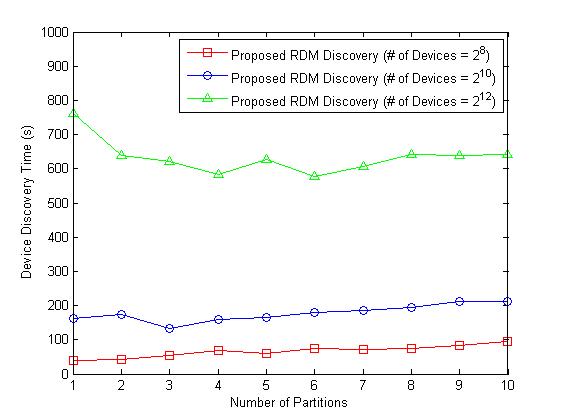
4. Performance Analysis

To evaluate the performance of the proposed scheme, we compared the device discovery time for the existing binary search scheme and the proposed partition-based scheme in a variety of network conditions. For numerical analysis, we used MATLAB (MathWorks, Natick, MA, USA), and we performed five test instances and obtained the averaged values in simulations.

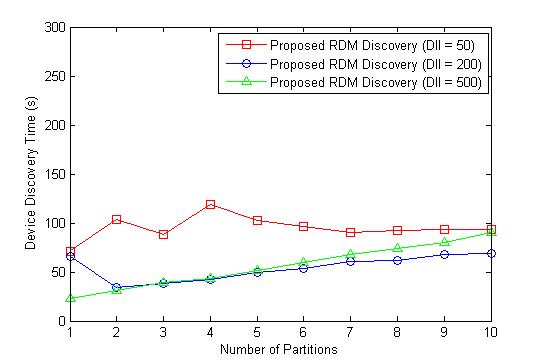
, (1)

In our analysis, we first examined an optimal configuration of the parameters that are used in the proposed scheme, such as the *Delay Increase Interval* (*DII*) and the number of partitions (*Np*). After that, we compared the proposed scheme with the existing scheme in terms of device discovery time.

Fig. 4 shows the impacts of both *Np* and the number of devices on the device discovery time, in which *DII* is set to 10 time slots. It is noted in the figure that the device discovery time is plotted in the time unit of seconds, whereas, the basic operational time unit is 0.0028 seconds in each simulation. From this figure, we can see that the device discovery time tends go higher, as the number of devices increases. From the results, it is noted that there may be an optimal number of *Np* for a given number of devices. For example, for 210 devices, *Np* = 3 gives the best performance, whereas, the device discovery time can be minimized at *Np* = 6 when there are 212 devices in the network.



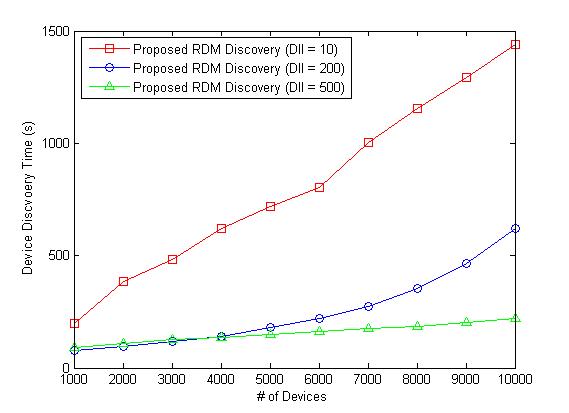
**Fig. 4.** The impacts of both Np and the number of devices on the discovery time.



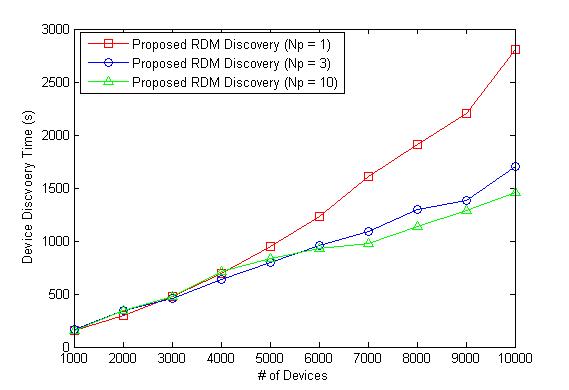
**Fig. 5.** The impact of Np and DII on the proposed scheme.

Fig. 5 shows the impacts of *Np* and *DII* on the device discovery time, in which the number of devices is given by 210. We see that *Np* = 3 is a reasonable choice for the best performance for all *DII*s. On the other hand, it is shown that the device discovery time lessens, as the *DII* increases. This implies that a large *DII* (e. g., 200 or 500) can reduce the possibility of multiple responses (collisions) and thus increase the chances of success responses, as compared to a small *DII* (e.g., 50). However, as *Np* gets larger, the performance gaps for different DIIs are not significant, which implies that we need to conduct more experiments to obtain a more suitable configuration of the parameters.

Fig. 6 analyzes the relationship between *DII* and the number of devices, in which *Np* is given by 10. From the figure we can see that there is an optimal *DII* for a given number of devices. That is, *DII* = 200 is preferred for a small number of devices (4,000 or less), whereas, *DII* = 500 seems to provide better performance for a large number of devices (5,000 or more).



**Fig. 6.** Analysis of DII by the number of devices.



**Fig. 7.** Analysis of NP by the number of devices.

Fig. 7 analyzes *Np* for the different number of devices, in which *DII* is given by 10. In this figure, we can see that for a smaller number of devices (4,000 or less), *Np* = 3 is a reasonable choice, but *Np* = 10 gives the best performance for a large number of devices (5,000 or more). It is noted that the performance gaps for different *Np* becomes larger, as the number of devices increase.

Up until this point, we have investigated the optimal configuration of the parameters used in the proposed scheme through simulations. From the simulation results, we can see that the following relationship between the device discovery time and the protocol parameters exists: “device discovery time ∝ (number of devices \* *Np* / *DII*).” That is, the device discovery time tends to be proportional to the number of devices and the number of partitions (*Np*), but inversely proportional to the *DII*.

With this obtained optimal configuration, we compared the overall device discovery time for the existing binary search scheme and the proposed partition-based scheme, as shown in Fig. 8. From this figure, we can see that the proposed scheme gives a much smaller device discovery time than the existing scheme does. Moreover, the performance gaps between the existing and proposed schemes become larger, as the number of devices increases. This is because the proposed scheme can reduce the attempts made for device discovery by dividing all devices into several partitions and can also minimize the possibility of multiple responses (collisions) from the devices, as compared to the existing binary search scheme.

Minion Pro 8.5pt

Times New Roman, **Bold** 9.5pt

Times New Roman 9.5pt

**Table 1.** PSNR results [dB] for various test images and δ value

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | δ | Visu-shrink | Sure-shrink | Bayes- shrink | Sendur’s method | Oracle- shrink | Wiener | Normal shrink |
| Lena | 10 | 28.76 | 33.28 | 33.32 | 33.94 | 33.61 | 33.53 | 33.57 |
| 20 | 26.46 | 30.22 | 30.17 | 30.73 | 30.38 | 30.35 | 28.98 |
| 30 | 25.14 | 28.38 | 28.48 | 28.94 | 28.60 | 28.53 | 25.69 |
| Barbara | 10 | 24.81 | 30.21 | 30.86 | 31.13 | 31.50 | 31.37 | 29.81 |
| 20 | 22.81 | 25.91 | 27.13 | 27.25 | 27.40 | 27.32 | 26.79 |
| 30 | 22.00 | 24.33 | 25.16 | 25.21 | 25.32 | 25.22 | 24.29 |
| Boat | 10 | 26.49 | 31.19 | 31.80 | 32.25 | N/A | N/A | N/A |
| 20 | 24.43 | 28.14 | 28.48 | 28.93 | N/A | N/A | N/A |
| 30 | 23.33 | 26.52 | 26.60 | 27.11 | N/A | N/A | N/A |
| Goldhill | 10 | N/A | 31.87 | 31.90 | N/A | 31.97 | 31.71 | 31.80 |
| 20 | N/A | 28.43 | 28.65 | N/A | 28.76 | 28.65 | 28.26 |
| 30 | N/A | 27.02 | 27.11 | N/A | 27.16 | 27.09 | 27.09 |

Times New Roman 8.5pt

5. Conclusions

In this paper, we have proposed a new partition-based device discovery scheme in lighting control networks. In the proposed scheme, all devices are divided into several partitions. In addition, to avoid collisions occurring due to multiple responses, each device sends a response message based on a response timer that is configured by the controller.

From the numerical analysis, we can see that the proposed scheme can provide a much lower device discovery time than the existing scheme. Moreover, the performance gaps between the existing and proposed schemes become lager, as the number of devices increase

, (2)

, (3)

Acknowledgement

This paper is………………

**References**

Arial, **Bold** 14pt

Times New Roman, 9pt

1. A. A. Helal, M. Mokhtari, and B. Abdulrazak, *The Engineering Handbook of Smart Technology for Aging, Disability, and Independence*. Hoboken, NJ: Wiley, 2008.
2. J. Lee, J. G. Andrews, and D. Hong, "Spectrum-sharing transmission capacity," *IEEE Transactions on Wireless Communications*, vol. 10, no. 9, pp. 3053-3063, Sep. 2011.
3. 3rd Generation Partnership Project (3GPP), "Evolved universal terrestrial radio access (E-UTRA); radio frequency (RF) system scenarios (Release 9)," 3GPP Technical Specification Group Radio Access Network, *Technical Report TR 36.942 V9.1.0*, Sep. 2010.
4. Q. Cao and S. Fujita, "Cost-effective replication schemes for query load balancing in DHT-based peer-to-peer file searches," *Journal of Information Processing Systems*, vol. 10, no, 4, pp. 628-645, Dec. 2014.
5. V. Kolici, A. Herrero, and F. Xhafa, "On the performance of Oracle grid engine queuing system for computing intensive applications," *Journal of Information Processing Systems*, vol. 10, no, 4, pp. 491-502, Dec. 2014.

Tahoma, 7 Pt

IF you have ORCID ID, Please describe. For more details, Please visit website.(http://orcid.org)

Times New Roman, **Bold** 9.5pt

2.1cm×1.7cm

**Sang-Il Choi** / <http://orcid.org/0000-0002-4510-0385> (ORCID ID)

He received B.S. and M.S. degrees in School of Computer Science and Engineering from Kyungpook National University in 2010 and 2012, respectively. Since March 2012, he is with the School of Computer Science and Engineering from Kyungpook National University as a Ph.D. candidate.

Times New Roman, 9.5pt

**Sanghun Lee** / <http://orcid.org/0000-0002-4510-0385> (ORCID ID)

He received M.S. degree in School of Computer Science and Engineering from Kyungpook National University in 2014. His current research interests include mobile communication and lighting control network.

**NOTE :**

1. The maximum length of manuscript is 20 pages**.**

2. Manuscripts that are not properly formatted and prepared will be returned to the authors for revision and resubmission.

3. Figures and Tables need to be provided as high resolution file.

This figure needs to be provided as high resolution file that is able to edit.