LightCon: Simplify Line-of-Sight Connection with Visible Symbols in Industrial Wireless Networks

Xi Yue*, Zhezhuang Xu*, Wenchao Huang*, Rongkai Wang*, Hao Jiang*, Cailian Chen†, Shih-Hau Fang‡

*School of Electrical Engineering and Automation, Fuzhou University, Fuzhou 350000, China
Email: zzxu@fzu.edu.cn
†Department of Automation, Shanghai Jiao Tong University, Shanghai 200240, China
Email: cailianchen@sjtu.edu.cn
‡Department of Electrical Engineering, Yuan Ze University, Taoyuan 320, Taiwan
Email: shfang@saturn.yzu.edu.tw

***Abstract*—To establish data connection in industrial wireless networks, the engineer has to enter the address of the target machine or manually select it from a list. Since the machines are densely deployed in the plant, the manual data connection becomes a non-trivial issue which may lead to connection error and waste of time. The existing technologies, such as QR code scanning and proximity estimation, simplify the process of data connection to the closest node. However, they can not be used to connect with any node located in line-of-sight of the engineer, and thus limits their application.**

In this paper, we propose a scheme called *LightCon* which controls the display module (LCD screen or LED light) of nodes to display different visible *symbols* such as colors and numbers, thus the engineer can select the corresponding symbol on the mobile device to establish connection. Moreover, symbol assignment algorithms are proposed to reduce the complexity of symbol selection when the number of nodes are greater than the number of symbols. The theoretical and experimental results prove that *LightCon* reduces the complexity of line-of-sight connection with guaranteed accuracy.

***Index Terms*—Line-of-sight connection, visible symbols, human machine interaction, industrial wireless network**

I. Introduction

With the development of industrial wireless networks [1]–[4], the engineer can use mobile device to interact with machines in the plant via wireless technologies such as WiFi [5] or Bluetooth [6]. It greatly improves the efficiency of works in the plant, since the engineer can control the machine and observe if it operates well simultaneously.

In most existing industrial wireless networks, to establish the data connection, the engineer has to enter the address of the target machine or manually select it from a list. Since the machines are densely deployed at 5-30 nodes per $100m^2$ in the plant, the manual data connection becomes a non-trivial issue which may lead to connection error and waste of time.

To simplify the data connection, a straightforward solution is using QR code [7] or NFC [8] scanning to obtain the ID before connection. However, both of them have to be executed in extremely short range ($< 10cm$) which can not be applied in the plant with large or dangerous machines.

Proximity estimation provides another solution [9]–[11] that detects the scenario when a pair of nodes are closer than a predefined proximity distance. In [11], the authors implement an industrial HMI (human machine interaction) system based on Bluetooth, and propose an algorithm called FaceME to estimate the machine that is closest to the engineer. However, in practice, the engineer always needs to interact with the machine that is not the closest one. Take the control room of electrical substation for example (Fig.1), the switch cabinets are densely deployed in rows, thus the engineer may interact with any nearby cabinet rather than the closest one.

Fig. 1. Line-of-sight human machine interaction in the electrical substation

In this paper, we propose a scheme named *LightCon* that simplifies the process of establishing connection to any node located in line-of-sight. The basic idea of LightCon is taking advantage of the display module (LCD screen or LED light) of wireless nodes to display different visible *symbols* such as colors and numbers. In this case, when the engineers want to connect with a node, they just need to look at the symbol displayed on the node, and then select the corresponding symbol displayed on the mobile device to establish connection. They do not have to remember the trivial addresses of each node, and thus improve their efficiency.

Specifically, this paper has the following contributions:

1) We propose the LightCon scheme that combines wireless communication and visible symbol assignment to simplify data connection with any node located in line-of-sight of the engineer.

2) Two symbol assignment algorithms are proposed to reduce the complexity of symbol selection in LightCon, when the number of nodes are greater than the number of symbols. Theoretical analysis is provided to prove the efficiency of the proposed algorithms.

978-1-5386-1127-2/17/$31.00 ©2017 IEEE

3) We implement an industrial wireless network testbed to prove the effectiveness of LightCon. The experimental results prove that LightCon can reduce the complexity of line-of-sight connection with guaranteed accuracy.

The rest of this paper is organized as follows. Section II introduces the model of industrial wireless networks considered in this paper. LightCon scheme and its implementation are described in Section III. Section IV introduces two symbol assignment algorithms based on LightCon. Section V provides the theoretical analysis of the symbol assignment algorithms, and the experimental evaluation of LightCon is demonstrated in Section VI. Finally, Section VII concludes this paper.

II. Network Model

In this paper, we consider an industrial wireless network that has the following characteristics:

1) There are multiple machines deployed in the industrial plant. Each machine is connected to a wireless node which includes wireless communication module (such as Wi-Fi and Bluetooth), and a display module (such as LED light) that can display different *symbols* such as color or number. The wireless node is defined as *machine node*. The number of machine nodes is denoted as M. The number of symbols that the node can display is denoted as S.
2) The engineer is moving in the plant to interact with different machine nodes in line-of-sight by a mobile device defined as *human node*.
3) To initiate the data connection, the human node has to scan the machine nodes nearby, and obtains a neighbor list that contains the address of machine nodes which can be connected by the human node. We assume that the neighbor list includes all the machine nodes in the line-of-sight of the engineer, and the number of nodes in the neighbor list is denoted as N.
4) We assume that the probability that a machine node in the neighbor list is selected as the target of the engineer follows uniform distribution.

III. LightCon Scheme

In this section, we propose the LightCon scheme that simplifies the connection with line-of-sight nodes with visible symbols. The details of LightCon are given as follows:

1) The human node scans the machine nodes nearby and obtains the neighbor list which contains their ID and RSSI.
2) Given the neighbor list of machine nodes and the set of symbols, the human node runs a symbol assignment algorithm to assign the symbol for each machine node. The symbol assignment algorithm has great impact on the complexity of LightCon, thus we will discuss its details specifically in Section IV.
3) After running symbol assignment algorithm, the human node generates a SA (symbol assignment) packet that contains the ID and assigned symbol of every node in the neighbor list. Then it broadcasts the SA packet by wireless communication.
4) The machine node that receives the SA packet checks if its ID is contained in the packet. If yes, it reads the assigned symbol and then control its display module accordingly.
5) At the same time, the human node displays the list of symbols on the screen. The engineer looks for the symbol that is displayed on the target machine, and then selects the same symbol on the human node.
6) Once the symbol is selected, the human node should check the number of nodes that are assigned to the selected symbol n_k. If $n_k = 1$, the human node will connect to the target machine directly. Otherwise, the human node will run the symbol assignment algorithm again to assign symbols for the selected n_k nodes. The process repeats until $n_k = 1$ is satisfied.

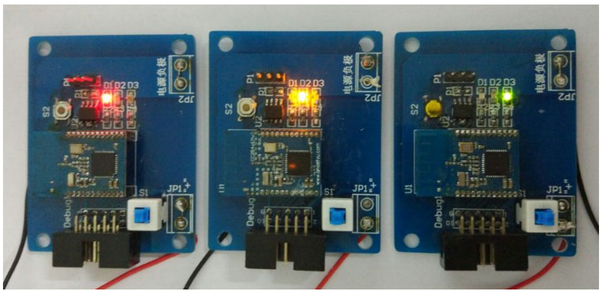

Fig. 2. Machine nodes with different symbols (colors)

Fig. 3. The user interface of symbol selection

A. Implementation

We implement LightCon in an industrial wireless network testbed. As shown in Fig.2, the machine node is a wireless module that consists of a Bluetooth core board and an extension board. The Bluetooth core board uses a TI-CC2541 chip, and the extension board includes a MAX3485 chip to support RS485 communication and three LED lights that can display three different colors (red, yellow and green). The LED lights can be controlled according to the SA packet sent by the human node.

The human node in the testbed uses Google Nexus 9 tablet with Android OS version 7.0. We developed an Android application that includes all the functions of LightCon. As shown in Fig.3, the neighbor list is displayed with different colors (symbols) that are corresponding to the colors displayed on machine nodes. Thus in practice, the engineer can connect to the target machine by simply selecting the corresponding color on the screen.

We use this testbed to evaluate the performance of LightCon, and the results will be given in Section VI.

IV. Symbol Assignment Algorithms

The symbol assignment algorithm is used to assign the symbol for each machine node. When $N \leq S$, every machine node is ensured to have a distinct symbol. However, when $N > S$, multiple machine nodes may share the same symbol, thus the human node has to select the symbol multiple rounds until the unique node is selected. In this paper, the current round of symbol selection is denoted as k, and the number of nodes in the kth symbol selection is denoted as n_k. Moreover, an important parameter K is defined as,

$$K = \left\lceil \log_S{}^N \right\rceil \tag{1}$$

The parameter K is used in the design of symbol assignment algorithms, and we will prove that K is equal to the maximum rounds of symbol selection in Section V.

In this section, we propose two symbol assignment algorithms: average symbol assignment (A-SA) and biased symbol assignment (B-SA). The details are given as follows.

A. Average Symbol Assignment (A-SA)

The basic idea of A-SA design is intuitive: divide n_k nodes into S groups equally. Specifically, n_k nodes are divided into j_k groups which have $\lceil \frac{n_k}{S} \rceil$ machine nodes, and $S - j_k$ groups which have $\lfloor \frac{n_k}{S} \rfloor$ machine nodes. Formally,

$$n_k = j_k \cdot \left\lceil \frac{n_k}{S} \right\rceil + (S - j_k) \left\lfloor \frac{n_k}{S} \right\rfloor \tag{2}$$

where j_k is equal to the remainder of $\frac{n_k}{S}$,

$$j_k = n_k - S \cdot \left(\left\lceil \frac{n_k}{S} \right\rceil - 1 \right) \tag{3}$$

B. Biased Symbol Assignment (B-SA)

We firstly describe the details of B-SA algorithm in this section, and then explain the intuition behind the design of B-SA by comparing with A-SA in Section IV-C.

In B-SA algorithm, the given n_k nodes are divided into i_k groups which have S^{K-k-1} machine nodes, j_k groups that include S^{K-k} machine nodes, and one group which has l_k machine nodes. Formally,

$$n_k = i_k \cdot S^{K-k-1} + j_k \cdot S^{K-k} + l_k \tag{4}$$

The value of i_k, j_k and l_k should satisfy the following constraints,

$$i_k + j_k + 1 = S \tag{5}$$

$$S^{K-k-1} \leq l_k < S^{K-k} \tag{6}$$

then we have,

$$i_k = S - \left\lfloor \frac{n_k - S^{K-k-1}}{S^{K-k} - S^{K-k-1}} \right\rfloor \tag{7}$$

$$j_k = \left\lfloor \frac{n_k - S^{K-k-1}}{S^{K-k} - S^{K-k-1}} \right\rfloor - 1 \tag{8}$$

$$l_k = n_k - (j_k + 1) \cdot S^{K-k} + (j_k + 1) \cdot S^{K-k-1} \tag{9}$$

The expression of j_k (Eqn.8) is derived as follows. Combining Eqn.6 and Eqn.9,

$$S^{K-k-1} \leq n_k - (j_k + 1) \cdot S^{K-k} + (j_k + 1) \cdot S^{K-k-1} < S^{K-k} \tag{10}$$

Then we can obtain the range of j_k as,

$$\frac{n_k - S^{K-k}}{S^{K-k} - S^{K-k-1}} < j_k + 1 \leq \frac{n_k - S^{K-k-1}}{S^{K-k} - S^{K-k-1}} \tag{11}$$

$$\frac{n_k - S^{K-k-1}}{S^{K-k} - S^{K-k-1}} - 2 < j_k \leq \frac{n_k - S^{K-k-1}}{S^{K-k} - S^{K-k-1}} - 1 \tag{12}$$

Since j_k has to be an integer, we can derive Eqn.8 directly based on Eqn.12.

C. Case Study

In this section, we provide a case study to demonstrate the difference between A-SA and B-SA algorithm. Assume that there are 5 nodes in the neighbor list ($N = 5$), and the number of symbols is 3 ($S = 3$). According to Eqn.1, $K = 2$.

In A-SA, the nodes are divided into three groups as $\{2, 2, 1\}$. Only one node can be connected by executing symbol selection once, and the rest four nodes can be connected by executing two rounds of symbol selection.

On the other hand, in B-SA, the nodes are divided as $\{3, 1, 1\}$. There are two nodes that can be connected by executing symbol selection once. The rest three nodes can be connected by executing two rounds of symbol selection. Therefore, compared with A-SA, B-SA reduces the average round of symbol selection. The intuition of B-SA design is taking advantage of all the symbols S, and minimizes the number of nodes involved in Kth round of symbol selection.

Further analysis of their performance will be provided in Section V and VI-B.

V. Theoretical Analysis

In this section, theoretical analysis are provided to study the performance of LightCon with A-SA and B-SA algorithms. In LightCon, when $N > S$, the user may execute symbol selection multiple times until the target machine is selected. Thus the rounds of symbol selection required for selecting the target machine node, which is denoted by γ, has great impact on the user experience. We use it as the performance metric in the analysis. The variables used in the analysis are summarized in Table I.

Based on the definition of K given in Eqn.1, we firstly evaluate the maximum and minimum rounds of symbol selection.

Theorem 1. *In both A-SA and B-SA algorithms, given N machine nodes and S symbols, for connecting to any machine node, the maximum rounds of symbol selection is K, and the minimum rounds of symbol selection is $K - 1$.*

TABLE I
VARIABLES LIST

Variable	Description
N	The number of nodes in the neighbor list
S	The number of symbols
K	The maximum round of symbol selection (Eqn.1)
k	The index of current round in symbol selection
n_k	The number of nodes in the kth symbol selection
γ	The rounds of symbol selection required for selecting the target machine node
P	The number of nodes with $\gamma = K-1$

Proof: In A-SA, at the first round of symbol assignment, we have,

$$N = j_1 \cdot \left\lceil \frac{N}{S} \right\rceil + (S - j_1) \left\lfloor \frac{N}{S} \right\rfloor \tag{13}$$

According to Eqn.1, we have,

$$\log_S{}^{\frac{N}{S}} \le K - 1 < \log_S{}^{N} \tag{14}$$

then,

$$S^{K-2} < \frac{N}{S} \le S^{K-1} \tag{15}$$

Finally, we can deduce that,

$$S^{K-2} < \left\lceil \frac{N}{S} \right\rceil \le S^{K-1} \tag{16}$$

$$S^{K-2} \le \left\lfloor \frac{N}{S} \right\rfloor \le S^{K-1} \tag{17}$$

Combining Eqn.16 and 17 with Eqn.13, for connecting to any machine node by A-SA algorithm, the rounds of symbol selection is either $K-1$ or K.

In B-SA, at the first round of symbol assignment, we have,

$$N = i_1 \cdot S^{K-2} + j_1 \cdot S^{K-1} + l_1 \tag{18}$$

For the node in groups with S^{K-2} or S^{K-1} nodes, it is required to execute $K-1$ or K rounds of symbol selection respectively to establish the connection. On the other hand, for the node in the group with l_1 nodes, since $S^{K-2} < l_1 < S^{K-1}$ (according to Eqn.6), the rounds of symbol selection is either $K-1$ or K. □

Based on the results given in Theorem 1, we use P to denote the number of nodes which have $\gamma = K-1$. Moreover, in Section II, we assume that the probability that a machine node in the neighbor list is selected as the target machine node follows uniform distribution. Then we derive the expected rounds of symbol selection of A-SA and B-SA as follows.

Theorem 2. *In A-SA algorithm, given N machine nodes and S symbols, the expected rounds of symbol selection is,*

$$E(\gamma) = \begin{cases} K + 1 - \frac{2 \cdot S^{K-1}}{N}, & N \in [S^{K-1}, 2 \cdot S^{K-1}] \\ K, & N \in \left(2 \cdot S^{K-1}, S^K\right] \end{cases} \tag{19}$$

Proof: Since the probability that a machine node is selected as the target machine node follows uniform distribution, combining with Theorem 1, we have,

$$E(\gamma) = \frac{P}{N} \cdot (K-1) + \frac{N-P}{N} \cdot K \tag{20}$$

In A-SA, the nodes are divided into S groups equally, thus the value of P is related to the range of N: when $N \in [S^{K-1}, 2 \cdot S^{K-1}]$, $P = 2 \cdot S^{K-1} - N$; when $N \in (2 \cdot S^{K-1}, S^K]$, $P = 0$.

Combining with Eqn.20, we can easily deduce Eqn.19. □

Theorem 3. *In B-SA algorithm, given N machine nodes and S symbols, the expected rounds of symbol selection is,*

$$E(\gamma) = K - \frac{\sum_{k=1}^{K-1} i_k{}' \cdot S^{K-k-1}}{N} \tag{21}$$

where

$$i_k{}' = S - \left\lfloor \frac{n_k{}' - S^{K-k-1}}{S^{K-k} - S^{K-k-1}} \right\rfloor \tag{22}$$

$$n_k{}' = \begin{cases} N, & k = 1 \\ n_{k-1}{}' - (S - i_{k-1}{}') \cdot \left(S^{K-k-1} - S^{K-k-2}\right), & k \ge 2 \end{cases} \tag{23}$$

Proof: Based on Eqn.4 given in Section IV-B, P in B-SA is only relevant to $i_k \cdot S^{K-k-1}$ and l_k. In order to calculate the number of nodes that have $\gamma = K-1$ in l_k, we can use an iteration with $n_{k+1} = l_k$. Combining $n_{k+1} = l_k$ with Eqn.7 and 9, we can derive the definition of $i_k{}'$ (Eqn.22) and $n_k{}'$ (Eqn.23).

Then P can be expressed as,

$$P = \sum_{k=1}^{K-1} i_k{}' \cdot S^{K-k-1} \tag{24}$$

Combining with Eqn.20,

$$E(\gamma) = \frac{\sum_{k=1}^{K-1} i_k{}' \cdot S^{K-k-1}}{N} \cdot (K-1) + \frac{N - \sum_{k=1}^{K-1} i_k{}' \cdot S^{K-k-1}}{N} \cdot K \tag{25}$$

Eqn.21 can be easily deduced based on Eqn.25. □

Corollary 1. *The expected rounds of symbol selection in B-SA algorithm is smaller than that in A-SA algorithm.*

Proof: We define e as the difference between $E(\gamma)$ of A-SA and B-SA, then the proof of the corollary can be transformed to prove $e > 0$.

When $N \in \left(2 \cdot S^{K-1}, S^K\right]$, based on Eqn.19 and Eqn.21, it is clear to derive that,

$$e = \sum_{k=1}^{K-1} i_k{}' \cdot S^{K-k-1} > 0 \tag{26}$$

When $N \in \left[S^{K-1}, 2 \cdot S^{K-1}\right]$, we have,

$$
\begin{aligned}
e' = e \cdot N &= \sum_{k=1}^{K-1} i_k' \cdot S^{K-k-1} + N - 2 \cdot S^{K-1} \\
&= \frac{S^K - 1}{S - 1} + N - 2 \cdot S^{K-1} \qquad (27)\\
&- \sum_{k=1}^{K-1} \left\lfloor \frac{n_k' - S^{K-k-1}}{S^{K-k} - S^{K-k-1}} \right\rfloor \cdot S^{K-k-1}
\end{aligned}
$$

Further, when $N \in \left[S^{K-1}, 2 \cdot S^{K-1} - S^{K-2}\right)$, we have $\left\lfloor \frac{n_k' - S^{K-k-1}}{S^{K-k} - S^{K-k-1}} \right\rfloor = 1$, then Eqn.27 can be simplified as,

$$
\begin{aligned}
e' &= \frac{S^K - 1}{S - 1} + N - 2 \cdot S^{K-1} - \frac{S^{K-2} - 1}{S - 1} \qquad (28)\\
&= N - S^{K-1} + S^{K-2} > 0
\end{aligned}
$$

When $N \in \left[2 \cdot S^{K-1} - S^{K-2}, 2 \cdot S^{K-1}\right]$, we have $\left\lfloor \frac{n_k' - S^{K-k-1}}{S^{K-k} - S^{K-k-1}} \right\rfloor = 2$, then,

$$
\begin{aligned}
e' &= \frac{S^K - 1}{S - 1} + N - 2 \cdot S^{K-1} - 2 \cdot \frac{S^{K-2} - 1}{S - 1} \qquad (29)\\
&= N + S^{K-2} - S^{K-1} - \frac{S^{K-2} - 1}{S - 1} > 0
\end{aligned}
$$

Then we have $e > 0$ when $N \in \left[S^{K-1}, 2 \cdot S^{K-1}\right]$.

□

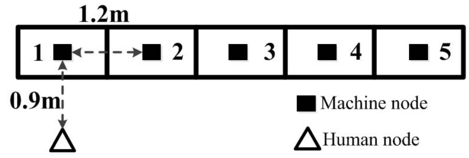

Fig. 4. Node deployment in connection time experiments

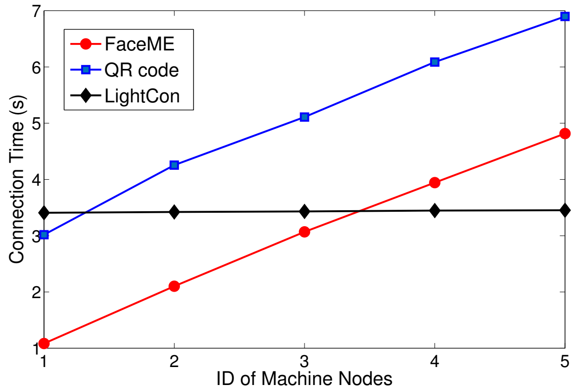

Fig. 5. Connection time comparison

VI. Experimental Evaluation

In this section, we execute experiments to evaluate the performance of LightCon scheme. The performance is analyzed in two aspects: connection time and rounds of symbol selection. The details are given as follows.

A. Connection Time Comparison

The connection time is defined as the duration between the time that the engineer initiates the process of node connection and the time that the connection is established. To evaluate the connection time, LightCon is implemented in the testbed described in Section III-A. We also implement FaceME [11] and QR code scanning [7] in the testbed for comparison.

The testbed is deployed in the Delta PLC (Programmable Logic Controller) laboratory in Fuzhou University. There are 23 control consoles deployed in three rows in the laboratory, and the size of each console is 1.2m × 0.7m. The laboratory represents a typical industrial plant: a large number of machines are deployed in rows with fixed spacing, while the engineer moves in the passage to interact with machines in line-of-sight.

As shown in Fig.4, we deploy five machine nodes in a row ($N = 5$), and each machine node can display three symbols ($S = 3$). The distance between machine nodes is $1.2m$, and the distance between the human node and the closest machine node is $0.9m$. The human node connects to each machine node (from node 1 to node 5) in turns, and we record the connection time. Each test repeats 20 times, and we calculate the average connection time of each machine node.

As shown in Fig. 5, the connection time of FaceME and QR code scanning increases with the growth of the distance between the human node and the target machine node increases. It is because FaceME and QR code scanning are operated in a face-to-machine manner that the human node should move to the location closest to the target machine node before initiating the connection, and the movement of the human node generates additional time in the experiments. On the other hand, the connection time of LightCon is almost the same when the human node connects to different machine nodes. It is because the LightCon allows the human node to connect to any machine node in line-of-sight, and thus does not require additional time for node movement.

If we focus on the connection time of connecting to machine node 1, which means the time for the human node movement is not considered, the connection time of FaceME, QR code scanning and LightCon are about $1s$, $3s$, and $3.4s$ respectively. It is because the FaceME has the most simple operation in all three algorithms, which requires only a single click on the user interface. In LightCon, since there are 5 machine nodes and 3 symbols in the experiment, in most cases, the user has to execute two rounds of symbol selection until the target machine node is connected. The detailed analysis has been given in Section IV-C. The multiple symbol selection and SA packet transmission increase the connection time of LightCon. Nevertheless, the connection time of LightCon is acceptable, and LightCon outperforms other algorithms when the human node has to connect to the machine node that is relatively far away.

TABLE II
THE ROUNDS OF SYMBOL SELECTION γ IN A-SA ALGORITHM WITH DIFFERENT NUMBER OF MACHINE NODES N AND SYMBOLS S

γ \ S / N	2	3	4	5
5	2.3971	1.7979	1.4022	1.0
10	3.3967	2.2047	2.0	2.0
15	3.9292	2.8012	2.0	2.0
20	4.4011	3.0	2.3961	2.0
25	4.7196	3.0	2.7126	2.0
30	4.9343	3.2008	2.933	2.337
35	5.1741	3.4631	3.0	2.5731
40	5.4004	3.6473	3.0	2.7542

TABLE III
THE ROUNDS OF SYMBOL SELECTION γ IN B-SA ALGORITHM WITH DIFFERENT NUMBER OF MACHINE NODES N AND SYMBOLS S

γ \ S / N	2	3	4	5
5	2.2451	1.3317	1.2483	1.0
10	3.2487	2.1085	1.5006	1.4514
15	3.8774	2.3382	2.0	1.6777
20	4.2491	2.6674	2.1459	1.8015
25	4.564	2.8883	2.1883	2.0
30	4.8774	3.0749	2.3337	2.0886
35	5.0949	3.148	2.4274	2.1387
40	5.2493	3.2618	2.4902	2.1672

B. Rounds of Symbol Selection

In this section, we evaluate the rounds of symbol selection γ in A-SA and B-SA algorithms with different number of nodes N and symbols S. It is difficult to provide massive machine nodes with different number of symbols, thus the experiments are executed by simulations in Matlab. In the simulations, the target machine node is randomly selected from N nodes, then the A-SA and B-SA algorithms run separately until the target machine node is selected. The number of nodes N varies from 5 to 40, and the number of symbols S grows from 2 to 5. Each simulation repeats 10000 times to obtain the statistical results. The results of A-SA are listed in Table II, and the results of B-SA are listed in Table III.

As shown in Table II and III, with the growth of N, the rounds of symbol selection γ becomes larger with the same S. On the other hand, when S increases, the γ decreases with the same N. Therefore, increasing the number of symbols is helpful to reduce the complexity of symbol selection in LightCon. When $S \geq 4$, the rounds of symbol selection is not more than 3 even if $N = 40$. This result proves that LightCon is a reasonable solution in most industrial wireless networks.

Comparing Table II with Table III, the γ of B-SA algorithm is smaller than that of A-SA algorithm with the same S and N. Therefore, the B-SA algorithm is proved to perform better than A-SA algorithm. Moreover, given the same S, the γ of A-SA algorithm may not change with the growth of N in some conditions. The reason can be found in Eqn.19. These simulation results prove the correctness of theoretical analysis given in Section V.

ACKNOWLEDGEMENT

The authors wish to thank the anonymous reviewers for their comments and suggestions to improve this paper. We would like to thank the financial support from NSF of China (61673116, 61304260, 61521063, 61622307, U1405251), NSF of Fujian Province of China (2017J01500, 2014J05072), and Ministry of Science and Technology, Taiwan (106-3114-E-007-005, 105-2221-E-155-013-MY3).

VII. CONCLUSION

In this paper, we propose the LightCon scheme to simplify connection with any node located in line-of-sight of the engineer. Two symbol assignment algorithms are proposed to reduce the complexity of symbols selection. The industrial wireless network testbed is implemented to evaluate the performance of LightCon. The experimental results proved that LightCon can reduce the complexity of line-of-sight connection with guaranteed accuracy.

REFERENCES

[1] V. C. Gungor and G. P. Hancke, “Industrial wireless sensor networks: Challenges, design principles, and technical approaches,” *IEEE Trans. Ind. Electron.*, vol. 56, no. 10, pp. 4258–4265, 2009.

[2] J. Chen, X. Cao, P. Cheng, Y. Xiao, and Y. Sun, “Distributed Collaborative Control for Industrial Automation With Wireless Sensor and Actuator Networks,” *IEEE Trans. Ind. Electron.*, vol. 57, no. 12, pp. 4219–4230, 2010.

[3] C. Chen, J. Yan, N. Lu, Y. Wang, X. Yang, and X. Guan, “Ubiquitous monitoring for industrial cyber-physical systems over relay-assisted wireless sensor networks,” *IEEE Trans. Emerg. Topics Comput.*, vol. 3, no. 3, pp. 352–362, 2015.

[4] F. Lin, C. Chen, N. Zhang, X. Guan, and X. Shen, “Autonomous Channel Switching: Towards Efficient Spectrum Sharing for Industrial Wireless Sensor Networks,” *IEEE Internet Things J.*, vol. 3, no. 2, pp. 231–243, 2016.

[5] J. A. Frank and V. Kapila, “Towards natural interfaces to interact with physical systems using smart mobile devices,” in *IECON 2014 - 40th Annual Conference of the IEEE Industrial Electronics Society*, Oct 2014, pp. 2458–2464.

[6] G. Patti, L. Leonardi, and L. L. Bello, “A bluetooth low energy real-time protocol for industrial wireless mesh networks,” in *IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society*, Oct 2016, pp. 4627–4632.

[7] H. Zhang, C. Zhang, W. Yang, and C. Y. Chen, “Localization and navigation using qr code for mobile robot in indoor environment,” in *IEEE International Conference on Robotics and Biomimetics*, 2015.

[8] R. Ramanathan and J. Imtiaz, “Nfc in industrial applications for monitoring plant information,” in *International Conference on Computing, Communications and Networking Technologies (ICCCNT)*, 2013.

[9] S. Liu, Y. Jiang, and A. Striegel, “Face-to-face proximity estimation using bluetooth on smartphones,” *IEEE Trans. Mobile Comput.*, vol. 13, no. 4, pp. 811–823, 2014.

[10] S. Lakshmipriya and V. Gowthaman, “Improving proximity evaluation for smartphone via bluetooth,” *International Journal on Engineering Technology and Sciences*, vol. 2, no. 3, pp. 82–86, 2015.

[11] T. Liu, Z. Xu, R. Wang, H. Jiang, C. Chen, and S.-H. Fang, “Face-to-machine proximity estimation for mobile industrial human machine interaction,” in *IEEE International Conference on Communications (ICC)*, 2017.