Blockage Type Detection Process in Triangular-wave Topology for mmWave Wireless Backhaul

Yuchen Liu, Qiang Hu, and Douglas M. Blough
School of Electrical and Computer Engineering
Georgia Institute of Technology

Abstract
Due to the rapid densification of small cells in 5G and beyond cellular networks, deploying wired high-bandwidth connections to every small cell base station is difficult, particularly in older metropolitan areas where infrastructure for fiber deployment is lacking. For this reason, mmWave wireless backhaul is being considered as a cost-effective and flexible alternative that has potential to support the high-rate transmission needed to accommodate backhaul traffic demands. However, maintaining a high network survivability is extremely important since networks based on mmWave communication are susceptible to short-term blockages or node failures caused by unforeseen events. Therefore, to realize a robust mmWave backhaul network, several works propose to adopt a triangular-wave topology in roadside environments, which can eliminate the mutual interference among backhaul links, and provide robustness to blockage effects. Based on this novel topology, we investigate a blockage type detection scheme, which reacts to blockage effects quickly and assists in various path reconfiguration approaches.

I. Introduction

To achieve a robust mmWave wireless backhaul with high network survivability, our work herein considers a distributed mmWave backhaul network architecture with relay nodes in urban environments. In this architecture, a number of mmWave dedicated relays in between small-cell base stations (SBSs) are deployed along urban streets, which naturally produces a mesh network structure. Note that such street-level deployment making use of lampposts has been suggested as a good choice in the 5G era [1]-[4], because it provides easy access to power, good access tier coverage for users and facilities, and ease of deployment in urban environments.

The interference-free triangular-wave topology (IFTW) is very well suited to provide high data rate communications for relay-assisted mmWave backhaul in roadside environments [2], [5]-[8]. In the IFTW topology, BSs and relays \( N_k \) are deployed on equally-spaced lampposts on both sides of the road, where the topology angle \( \theta \) and horizontal distance between adjacent nodes \( d_0 \) are the same everywhere along the topology (as depicted by the blue links of Fig. 1). One advantage of the IFTW topology is that the mutual interference along the path can be eliminated if \( \theta \) is made
large enough relative to the beamwidth $\phi$ of the directional antennas (Theorem 1 in [2]), i.e., if the interference-free condition in Eq. (1) is satisfied:

$$\theta - \arctan\left(\frac{\tan \theta}{3}\right) > \frac{\phi}{2}.$$  \hspace{1cm} (1)

Given a road width $d_w$ and length $d_l$, $d_0$ and the number of required nodes $N$ (including two BSs and several relays) in the IFTW topology are only determined by $\theta$ as $d_0 = d_w / \tan \theta$ and $N = \lfloor d_l / d_0 \rfloor + 1$, respectively.

Another advantage of the IFTW topology is the ability to reconfigure mmWave paths to avoid obstacles that occur along the roadway. Through adaptive beam steering when an obstacle blocks one or more of the original links, alternative links (shown in Fig. 1) can be used to restore the connectivity of the topology. In our previous work [5], we identified the following four types of blockages that can occur from a single obstacle along the roadway with the IFTW topology:

a) Type I: An obstacle in $L_k$ region blocks only a original link of the topology, such as $P_1$ in Fig. 1.

b) Type II: An obstacle in $L_k$ region blocks an original link and an adjacent alternative diagonal link simultaneously, such as \{$P_1, P_2$\} for $N_k$ or \{$_P_1, P_5$_\} for $N_k+1$.

c) Type III: An obstacle in $L_k$ region blocks an original link and a crossing alternative diagonal link, such as \{$_P_1, P_4$\}.

d) Type IV: An obstacle in $L_k$ region blocks all the TX/RX links of $N_k/N_{k+1}$, such as \{$_P_1, P_2, P_3$_\} \{$_P_1, P_5, P_6$_\}, which is equivalent to the failure of node $N_k/N_{k+1}$.

Any blockages produced by randomly placed obstacles can be decomposed into one or a combination of the above four types (Theorem 1 in [5]). This model takes into account the correlation between blockages, for example, a single blockage close to a node could block multiple mmWave paths simultaneously, which has not been considered in most previous studies [9]-[10]. In what follows, we investigate a blockage-type detection scheme based on this blockage model in the IFTW topology, which can be exploited for both link-level reconfiguration [6] and joint link-network level reconfiguration approaches [7]-[8].

II. Blockage type detection scheme

In general, the scheme has two phases: preparation phase and detection phase. The former aims to make every node in the topology know the occurrence of blockage and ready for coordination.
with others. The detection on link connectivity is conducted in the second phase. Since the durance of BTD and path reconfiguration is very short, we assume that no new blockage appears within this time period.

\[ \theta_d \]

\[ L_k - 2 \]

\[ L_k - 1 \]

\[ \ldots \]

\[ N_k - 2 \]

\[ N_k + 1 \]

\[ N_k + 3 \]

\[ N_k - 3 \]

\[ N_k + 2 \]

\[ N_k + 4 \]

\[ N_k - 1 \]

\[ L_k + 1 \]

Adjacent Node Set

![Diagram](image)

**Figure 2.** The possible alternative paths and the adjacent node set of \( N_k \).

A. Preparation Phase

For arbitrary node \( N_k \), it knows respective adjacent node set \( \text{Adj} = \{ N_{k+i} : -2 \leq i \leq 3, i \neq 0 \} \), alternative link set \( \text{Sub} = \{ e_{k,k+j} : 2 \leq j \leq 3 \} \), and TX link set \( \text{TL} = \text{Sub} \cup \{ e_{k,k+1} \} \). Note that \( \text{Adj}, \text{Sub} \) and \( \text{TL} \) of node \( N_k \) may change according to different relay path reconfiguration algorithms. During this phase, the following steps describe how nodes negotiate with each other to make preparation for detection.

**Step 1.** Considering the possible packet transmission error through links, the packet would be retransmitted for at most two times if the transmitter received nothing or NACK from receiver. Therefore, when an obstacle occurs between \( N_k \) and \( N_{k+1} \), i.e. the original link \( L_k \) is blocked (shown in Fig. 2), \( N_{k+1} \) will not receive expected data with high rate from \( N_k \) and then replied NACK, and if \( N_k \) receives nothing back or three consecutive NACK (through possible NLOS paths) from \( N_{k+1} \) during its own transmitting slot \( T_s \), both \( N_k \) and \( N_{k+1} \) will know the occurrence of blockage between them by the end of \( T_s \). We denote this time as \( t_0 \).

**Step 2.** Starting from \( t_0 \), \( N_k \) waits for a period of time, which is called “nodes preparation time” (NPT). Considering left-part and right-part nodes of \( N_k \), NPT can be computed as follow:

\[
NPT = \begin{cases} 
\max\{\lceil k/2 \rceil \cdot 2 - 1, N - k - 2 \} \cdot T_s, & 1 \leq k \leq N - 2 \\
\lceil k/2 \rceil \cdot 2 - 1 \cdot T_s, & k = N - 1, N \\
(N - 2) \cdot T_s, & k = 0
\end{cases}
\]

(2)

where \( k \) is the node index of \( N_k \) and \( N \) is the number of nodes in the topology. This NPT is consumed for two reasons:

1. At the beginning of NPT, \( N_{k+1} \) will send data as usual during its own transmitting time slot \( T_s \), but appending a stop transmitting signal (STS) to \( N_{k+2} \), then \( N_{k+2} \) also transmits data and STS to next node in its time slot, therefore, after \( (N-k-2) \cdot T_s \), all latter nodes of \( N_{k+1} \), i.e. \( N_{k+i} \) (\( i \geq 2 \)), will receive STS and know that one of previous original links is blocked, meanwhile switch their antennas to the omni-directional fashion from left to right once getting STS, which is convenient for receiving the following possible detection information. By the end of NPT, every node on the right side of \( N_k \) completes the antenna switching process, and
it could equip an additional omni-directional antenna, or adjust the beamwidth of its phase array antenna for this process.

2. For the left side nodes of \( N_k \), this NPT is not only used for avoiding BTD conflicts when multiple blockages occur concurrently, but also making these nodes stop transmitting data. As Fig. 3 shows, after one \( T_s \) from \( t_0 \), \( N_k \) has received data from \( N_{k-1} \), and it replied ACK appending STS, then \( N_{k-1} \) would stop sending data to \( N_k \) in next time slot and switch its antenna to omni-directional fashion for following possible detection. Meanwhile, \( N_k \) could confirm that previous original links \( e_{k-2,k-1} \) and \( e_{k-1,k} \) are unblocked. After \( 2T_s \), \( N_{k-1} \) replied ACK appending STS after receiving data from \( N_{k-2} \), then \( N_{k-2} \) would stop sending data to \( N_{k-1} \) in next time slot and switch its antenna to omni-directional fashion as well. After \( 3T_s \), \( N_k \) received the link-unblock sign (LUS) instead of data from \( N_{k-1} \), so that it could know that \( e_{k-4,k-3} \) and \( e_{k-3,k-2} \) are also unobstructed. However, if \( N_k \) did not receive data or received link-blocked sign (LBS) from \( N_{k-1} \) by the end of \( t_0 + T_s \) or \( t_0 + 3T_s \), that means another previous original link is blocked as well, so the previous node should first conduct the BTD process, and \( N_k \) as the latter node will switch its antenna to the omni-directional fashion for possible coordination. If \( N_k \) can still receive LUS from \( N_{k-1} \) by the end of NPT, it will confirm that all previous original links are unobstructed, and it can continue the BTD process for this blockage on the link \( L_k \). We can conclude previous original links information by Tab. I from \( N_k \)’s perspective (“0” and “1” denote the link is blocked or unblocked):

| \( N_{k-6} \) | \( N_{k-5} \) | \( N_{k-4} \) | \( N_{k-3} \) | \( N_{k-2} \) | \( N_{k-1} \) | \( N_k \) | \( t \) |
|---|---|---|---|---|---|---|---|
| Data | ACK | Data | ACK | Data | ACK | Data | \( t_0 \) |
| \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) |
| Data | ACK | Data | ACK | Data | ACK+STS | \( t_0+T_s \) | \( e_{k-2,k-1} \) & \( e_{k-1,k} \) unblocked |
| \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) |
| Data | ACK | Data | ACK+STS | LUS | \( t_0+2T_s \) | \( e_{k-2,k-1} \) & \( e_{k-1,k} \) unblocked |
| \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) |
| Data | ACK+STS | LUS | ACK | \( t_0+3T_s \) | \( e_{k-2,k-1} \) & \( e_{k-1,k} \) unblocked |
| \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) |
| Data | ACK+STS | LUS | ACK | \( t_0+4T_s \) |
| \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) |
| Data | ACK+STS | LUS | ACK | \( t_0+5T_s \) | \( e_{k-2,k-1} \) & \( e_{k-1,k} \) unblocked |
| \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) | \( \ldots \) |

**Figure 3.** Timing diagram of \( N_k \) and its left-side nodes.

From above, \( N_k \) will get the link information of all previous node pairs according the reception
Table I  Decision of previous original link from $N_k$ during NPT.

| Time after $t_0$ | Receive data/LUS from $N_{k-1}$ | Receive 1st null/LBS from $N_{k-1}$ |
|------------------|----------------------------------|-------------------------------------|
| $T_s$            | $e_{k-1,k}=1 \& \& e_{k-2,k-1}=1$ | $e_{k-1,k}=0 \quad \| \quad e_{k-2,k-1}=0$ |
| $3T_s$           | $e_{k-3,k-2}=1 \& \& e_{k-4,k-3}=1$ | $e_{k-3,k-2}=0 \quad \| \quad e_{k-4,k-3}=0$ |
| $5T_s$           | $e_{k-5,k-6}=1 \& \& e_{k-6,k-7}=1$ | $e_{k-5,k-6}=0 \quad \| \quad e_{k-6,k-7}=0$ |
| $\cdots$        | $\cdots$                         | $\cdots$                            |
| $(\lceil k/2 \rceil \cdot 2^{-1}) \cdot T_s$ | $e_{1,2}=1 \& \& e_{0,1}=1$ | $e_{1,2}=0 \quad \| \quad e_{0,1}=0$ |

from $N_{k-1}$ during NPT, and by the end of NPT, all previous nodes of $N_k$ will stop transmitting data and switch to the omni-directional antenna for possible detection process. Fig. 4 shows the status of nodes in the topology after NPT.

![Figure 4. The status of nodes after NPT.](image)

**Step 3.** After waiting for the time period NPT, $N_k$ starts to send blockage type detection preparation information (BTDP) to $N_{k-1}$ at $t_0'$ ($t_0'=t_0+NPT$), then $N_{k-1}$ will also forward this BTDP to $N_{k-2}$ after the time durance of transmitting BTDP along the original link $T_s'$ ($T_s' \ll T_s$), therefore, all left-side nodes of $N_k$ would receive BTDP after $(k-1) \cdot T_s'$. The format of BTDP is shown in Tab. II

| NID | BLID | STD | DTL | DO |
|-----|------|-----|-----|----|
| $N_k$ | $K$ | $t_0'+2 \cdot T_s'$ | $T_k$ | $\{N_k;t_1; N_{k-2};t_2; N_{k-1};t_3; N_{k+1};t_4; N_{k+2};t_5; N_{k+3};t_6\}$, interval: $\Delta$ |

where BID and BLID is the node ID and blocked link number, respectively, STD indicates start time of detection, DTL is the total length of this detection time, and DO represents the detection order of every participant appending respective starting time and interval. Before STD, the adjacent nodes $N_{k-1}$ and $N_{k-2}$ had received the BTDP and were ready for following detection process, and this is the end of the first phase.

**B. Detection Phase**

Blockage type detection process starts at STD. Note that, before sending detection signal (DS), the node will adjust its directional antenna to align the target node's and send BTDP firstly, so that the possible end nodes of substituted paths (i.e. right-part nodes of $N_k$) can get the detection information and switch to the direction antenna aligning with the source node’s, then reply an
ACK. After receiving the ACK from the intended node, the DS will be sent by detector node for checking if this substituted path is available. After replying ACK for DS, every target node would recover the omni-directional antenna for next possible detection. In addition, during this phase, the block-link table $A$ is updated and shared by $N_k$ and its adjacent nodes, which includes all possible paths $e_{k+i,k+j}$ ($2 \leq i \leq 3$, $i+1 \leq j \leq i+3$) for the nodes in DO and corresponding values (“0”-not checked, “1”-not blocked, “inf”-blocked). The detailed steps are listed as follow:

**Step 1:** At $t_1$, it is the turn of $N_k$ to check the available links in its $Sub_k$. $N_k$ first uses directional antenna to sends BTDP to $N_{k+2}$, and if receiving ACK from target node that has aligned the antenna to $N_k$ and is ready for detection, $N_k$ will send DS to check this substituted link $e_{k,k+2}$ (shown in Fig. 5). When finishing detecting process with $N_{k+2}$, $N_k$ starts to checks the substituted link to $N_{k+3}$ in the same way. After checking every link in $Sub_k$, $N_k$ initializes the block-link table and sets the corresponding value for its checked links, then multicasts $A_k$ to adjacent nodes because each node in $Adj$ uses omni-directional antenna now. After that, $N_k$ switches its antenna to an omni-directional fashion for possible detection from other nodes.

![Figure 5. The detection process between $N_k$ and $N_{k+2}$ from $t_1$ to $t_2$.](image)

**Step 2:** At $t_2$ or $t_3$, it is detection time for left side nodes $N_{k-2}$ or $N_{k-1}$. Similar to $N_k$, $N_{k-2}$ or $N_{k-1}$ will firstly send BTDP to target node such as $N_{k+1}$, and then send DS if receiving ACK and the alignment of their antennas is done. After checking all the substituted paths from its $Sub$, $N_{k-2}$ or $N_{k-1}$ updates and multicasts $A_k$, then switches to the omni-directional antenna.

**Step 3:** From $t_4$ to $t_6+\Delta$ (i.e. the end of $N_{k+3}$’s detection time), the right-side nodes $N_{k+1}$, $N_{k+2}$ and $N_{k+3}$ will conduct detection process one by one in their respective time interval. After sending BTDP and receiving ACK, $N_{k+i}$ ($1 \leq i \leq 3$) starts to detect whether they can go to their latter hops, namely sending DS to check the available links from their $Tl$ (TX links set). If $N_{k+i}$ does not have any latter-hop links, it will reset all $e_{k+i-j,k+i}$ ($1 \leq i \leq 3$, $1 \leq j \leq 3$) as $inf$ in $A_k$ for indicating that the possible alternative links into itself are actually not available, for example, if $N_{k+1}$ found that no available links for itself to go to next hops, it would update $e_{k-2,k+1}$ and $e_{k-1,k+1}$ as blocked status even if these two substituted links are available after checking by $N_{k-2}$ and $N_{k-1}$. After that, $N_{k+i}$ multicasts the new $A_k$ and switches antenna to omni-directional fashion.

**Step 4:** After $t_6+\Delta$, $N_k$ can get the full block-link table $A_k$, which contains all link information of adjacent nodes, and the link that was not checked in $A_k$ would be viewed as blocked status due to the possible failed-node cases. Then $N_k$ can judge the type of blockage according to Tab. I in [6]. For example, if just $e_{k,k+3}$ or $e_{k-2,k+1}$ in $A_k$ is $inf$, the blockage is in Type II for $N_k$ or $N_{k+1}$;
But if just \( e_{k-1,k+2} \) in \( A_k \) is \( \inf \), it is the Type III blockage; When all \( e_{k,k+i} \) (\( 2 \leq i \leq 3 \)) or \( e_{k-i,k+1} \) (\( 1 \leq i \leq 2 \)) are \( \inf \), the blockage is in Type IV for \( N_k \) or \( N_{k+1} \); And it is the Type I blockage when just \( e_{k,k+1} \) is \( \inf \) in \( A_k \). In other cases, the blockage can be always decomposed into these four types or their combination according to Theorem I in [6].

Until now, the second phase of BTD is finished, and the next steps about relay path selection are described detailedly in [6] (Section III). After that, the new substituted path for blockage avoidance is generated by \( N_k \), eg. selecting \( e_{k,k+3} \) instead of \( e_{k,k+1} \). Then \( N_k \) will multicast Path Reconfiguration Message (PRM) to its adjacent nodes, which contains the new selected path (eg. \( e_{k,k+3} \)), and each of them that received PRM is required to multicast as well in case some adjacent nodes did not get this information due to blockage effect. Once the start and end node of this new substituted path receive PRM, they will use their directional antennas to align with each other. Particularly, if one of nodes in \( Adj \) did not receive PRM until \( STD+DTL \), that means no available paths can reach this failed node, and it does not need to transmit data but stays in present status until one of paths into itself is available. Note that, if the new substituted path (eg. \( e_{k,k+2} \)) breaks the original scheduling, i.e. respective transmitting time slot of nodes after this new path, the information about rescheduled time slots is also required to add in PRM and transmit to the nodes after this new substituted path.

![Figure 6. The process of transmitting PRM from STD+DTL to STT.](image)

As Fig. 6 shows, at the time \( t=STD+DTL \), the start and end nodes of the new path (\( N_k \) and \( N_{k+3} \)) will start to transmit PRM to previous node and latter node, respectively. Once receiving PRM, each node in the topology forwards PRM to previous or latter node as well, and switches its antenna to the directional fashion and aligns the TX/RX antenna to its latter/previous node. When the source node receives PRM, it starts transmitting data, and the latter nodes will recover data transmission process once receiving data from previous nodes.

In summary, Algorithm 1 (a)-(c) in Appendix shows the pseudocode for the whole process from \( t_1 \) to the time that data transmission has been recovered, which contains respective works of the leader node \( N_k \) and its adjacent nodes \( N_{k-m} \) (\( 1 \leq m \leq 2 \)), \( N_{k+n} \) (\( 1 \leq n \leq 3 \)) during this process.

References

[1] J. Du, E. Onaran, et al., “Gbps user rates using mmWave relayed backhaul with high gain antennas”, IEEE Journal on Selected Areas in Communications, 2017: 1363-1372.

[2] Q. Hu and D. Blough, “Optimizing millimeter-wave backhaul networks in roadside environments”, Proc. of IEEE International Conference on Communications, 2018.
[3] “Terragraph: Solving the urban bandwidth challenge”, Facebook Connectivity, 2019.

[4] A. Dimas, D. Kalogerias, and A. Petropulu, “Cooperative beamforming with predictive relay selection for urban mmWave communications”, IEEE Access (2019): 157057-157071.

[5] Y. Liu and D. Blough, “Analysis of Blockage Effects on Roadside Relay-assisted mmWave Backhaul Networks”, Proc. of IEEE International Conference on Communications, 2019.

[6] Y. Liu, Q. Hu, and D. Blough, “Blockage Avoidance in Relay Paths for Roadside mmWave Backhaul Networks”, Proc. of IEEE Symposium on Personal, Indoor, and Mobile Radio Communications, 2018.

[7] Y. Liu, Q. Hu and D. Blough, “Joint Link-level and Network-level Reconfiguration for mmWave Backhaul Survivability in Urban Environments”, Proc. of Modeling, Analysis and Simulation of Wireless and Mobile Systems, ACM, 2019.

[8] Y. Liu, Q. Hu and D. Blough, “Joint Link-level and Network-level Reconfiguration for Urban mmWave Wireless Backhaul Networks”, Computer Communications, 2020.

[9] T. Bai, R. Vaze, H. Robert, “Analysis of blockage effects on urban cellular networks”, IEEE Transactions on Wireless Comm., 2014.

[10] T. Bai, H. Robert, “Coverage and rate analysis for millimeter-wave cellular networks”, IEEE Transactions on Wireless Comm., 2015
Appendix

Algorithm 1 (a) The status of the leader node ($N_k$) for blockage detection process

1: while($t_1 \leq t < t_2$) do
2:   for link with end node i in $Sub_k$ do
3:     AlignAntenna($i$); // align the antenna to node i
4:     BTDPsend($i$); // send BTDP to node i
5:     if (ACK) do //receive an ACK
6:       DSSend($i$); // send DS to node i
7:       $A_k$.init; // initialize $A_k$
8:       if(ACK) do
9:         $A_k$.set($e_{k,i}$) = 1; // update $e_{k,i}$=1 in $A_k$
10:      else
11:         $A_k$.set($e_{k,i}$) = 0;
12:     endif
13:   else
14:     $A_k$.set($e_{k,i}$) = 0;
15:   endif
16: endfor
17: mCast($A_k$); //multicast $A_k$ to its adjacent nodes
18: OmniAntenna($N_k$); // Switch to omni-directional antenna
19: wait; // as a listener
20: endwhile
21: while($t_2 \leq t < t_0 + \Delta$) do // as a listener
22:   if (recv BTDP || DS from node j) do
23:     AlignAntenna($j$);
24:   endif
25:   if (recv $A_k$)
26:     $A_k$.update($A_k$); // replace old $A_k$ with received $A_k$
27:   endif
28: endwhile
29: while($t_0 + \Delta \leq t < STD+DTL$) do // as a blockage type detector
30:   $Type$=BTD($A_k$) // Detect Blockage Type
31:   Reconf($Type$); // Reconfigure new path by HTPR algorithm in [6])
32:   mCast($PRM$);
33:   if ($NodeIndex$ in $PRM$) do // Prepare to transmission
34:     DirectAntenna($N_k$); // Switch to directional antenna
35:     AlignAntenna($PreNode$ and $LatterNode$ in $PRM$);
36:   endif
37:   wait; // idle time if process has been finished
38: endwhile
Algorithm 1 (b) The status of $N_{k-m}$ ($1 \leq m \leq 2$) for blockage detection process

1: \[ \text{while}(t_{4-m} \leq t < t_{5-m}) \text{ do} \]
2: \hspace{1em} \text{for link with end node } i \text{ in } Sub_{k-m} \text{ do} \]
3: \hspace{2em} AlignAntenna($i$); // align the antenna to node $i$
4: \hspace{2em} BTDPsend($i$); // send BTDP to node $i$
5: \hspace{2em} \text{if (ACK) do} // receive an ACK
6: \hspace{3em} DSsend($i$); // send DS to node $i$
7: \hspace{3em} \text{if (ACK) do}
8: \hspace{4em} $A_k.set(e_{k,i}) = 1$; // update $e_{k,i}=1$ in $A_k$
9: \hspace{3em} \text{else}
10: \hspace{4em} $A_k.set(e_{k,i}) = 0$;
11: \hspace{3em} \text{endif}
12: \hspace{2em} \text{else}
13: \hspace{3em} $A_k.set(e_{k,i}) = 0$;
14: \hspace{2em} \text{endif}
15: \hspace{1em} \text{endfor}
16: \hspace{1em} mCast($A_k$); // multicast $A_k$ to its adjacent nodes
17: \hspace{1em} OmniAntenna($N_{k-m}$); // Switch to omni-directional antenna
18: \hspace{1em} \text{wait; } // \text{as a listener}
19: \hspace{1em} \text{endwhile}
20: \hspace{1em} \text{while}(t_6 + \Delta \leq t < \text{STD} + \text{DTL}) \text{ do} // \text{as a broadcaster}
21: \hspace{2em} \text{if (recv PRM) do}
22: \hspace{3em} \text{mCast(PRM)};
23: \hspace{3em} \text{if (k-m in PRM) do } // \text{Prepare to transmission}
24: \hspace{4em} \text{DirectAntenna($N_{k-m}$); } // \text{Switch to directional antenna}
25: \hspace{4em} \text{AlignAntenna($PreNode$ and $LatterNode$ in PRM)};
26: \hspace{3em} \text{endif}
27: \hspace{2em} \text{endif}
28: \hspace{2em} \text{wait; } // \text{idle time if process has been finished}
29: \hspace{1em} \text{endwhile}
30: \hspace{1em} \text{while(other time) do}
31: \hspace{2em} \text{if (recv $A_k$) do}
32: \hspace{3em} $A_k.update(A_k)$;
33: \hspace{2em} \text{endif}
34: \hspace{1em} \text{endwhile}
Algorithm 1 (c) The status of $N_{k+n}$ ($1 \leq n \leq 3$) for blockage detection process

1: \hspace{1em} \textbf{while}($t_{n+3} \leq t < t_{n+4}$) \textbf{do}
2: \hspace{2em} \textbf{for} link with end node $i$ in $Tl_{k+n}$ \textbf{do} // check all TX links
3: \hspace{3em} \text{AlignAntenna}(i); // align the antenna to node $i$
4: \hspace{2em} \text{BTDPsend}(i); // send BTDP to node $i$
5: \hspace{3em} \textbf{if} (ACK) \textbf{do} // receive an ACK
6: \hspace{4em} \text{DSsend}(i); // send DS to node $i$
7: \hspace{3em} \textbf{if}(ACK) \textbf{do}
8: \hspace{4em} \text{$A_k.set(e_{k,i}) = 1$; // update $e_{k,i}=1$ in $A_k$}
9: \hspace{3em} \textbf{else}
10: \hspace{4em} \text{$A_k.set(e_{k,i}) = 0$;}
11: \hspace{2em} \textbf{endif}
12: \hspace{2em} \textbf{else}
13: \hspace{3em} \text{$A_k.set(e_{k,i}) = 0$;}
14: \hspace{2em} \textbf{endif}
15: \hspace{1em} \textbf{endfor}
16: \hspace{1em} \textbf{if} (all links in $T_l_{k+n}$ equal to zero) \textbf{do} // reset values of unavailable links in $A_k$
17: \hspace{2em} \textbf{for} $j=1$ to $3$ \textbf{do}
18: \hspace{3em} \text{$A_k.set(e_{k-n+j,k+n}) = 0$;}
19: \hspace{2em} \textbf{endfor}
20: \hspace{1em} \textbf{endif}
21: \hspace{1em} \text{mCast($A_k$); // multicast $A_k$ to its adjacent nodes}
22: \hspace{1em} \text{OmniAntenna($N_{k+n}$); // Switch to omni-directional antenna}
23: \hspace{1em} \text{wait; // as a listener}
24: \hspace{1em} \textbf{endwhile}
25: \hspace{1em} \textbf{while}($t_1 \leq t < t_{k+n}$ \textbf{||} $t_{k+n+1} \leq t < t_6 + \triangle$) // as a listener
26: \hspace{2em} \textbf{if} (recv BTDP \textbf{||} DS from node $a$) \textbf{do}
27: \hspace{3em} \text{AlignAntenna($a$)};
28: \hspace{3em} \text{ACK($a$)};
29: \hspace{2em} \textbf{endif}
30: \hspace{2em} \textbf{if} (recv $A_k$) \textbf{do}
31: \hspace{3em} \text{$A_k.update(A_k)$;}
32: \hspace{2em} \textbf{endif}
33: \hspace{1em} \textbf{endwhile}
34: \hspace{1em} \textbf{while}($t_6 + \triangle \leq t < STD + DTL$) \textbf{do} // as a broadcaster
35: \hspace{2em} \textbf{if} (recv PRM) \textbf{do}
36: \hspace{3em} \text{mCast(PRM)};
37: \hspace{2em} \textbf{if} ($k+n$ in PRM) \textbf{do} // Prepare to transmission
38: \hspace{3em} \text{DirectAntenna($N_{k+n}$); // Switch to directional antenna}
39: \hspace{3em} \text{AlignAntenna($PreNode$ and $LatterNode$ in PRM)};
40: \hspace{2em} \textbf{endif}
41: \hspace{2em} \textbf{endif}
42: \hspace{2em} \text{wait; // idle time if process has been finished}
43: \hspace{1em} \textbf{endwhile}