ROBUST TOKEN MANAGEMENT FOR UNRELIABLE NETWORKS

Eric E. Johnson*, Zibin Tang*, Manikanden Balakrishnan*, Juan Rubio*, Huiyan Zhang, and Srugun Sreepuram
New Mexico State University
Las Cruces, NM

ABSTRACT

Token passing can provide efficient medium access control in heavily loaded networks. However, it has been perceived to be too fragile for use in networks with non-negligible packet loss rates. In this paper, we present a novel token management approach that quickly recovers from common token loss and duplication scenarios, and that deals efficiently with changes in network connectivity and membership. This token management scheme was developed for use in military high frequency radio networks, and may also be appropriate for other networks that experience significant packet loss rates and relatively long link turnaround times.

1. INTRODUCTION

Medium access control (MAC) protocols are employed to manage the sharing of a broadcast channel by multiple nodes in a local area network (LAN). It is well known [1] that contention-based MAC protocols such as IEEE 802.3 (ethernet) provide fast, efficient channel access when traffic is light, while contention-free protocols such as IEEE 802.4 (token bus) achieve high efficiency under heavy loads, and provide bounded access time as well. However, the textbook analyses of MAC protocols usually assume that the LAN has the benign characteristics of wire (or fiber) media.

A wireless LAN (WLAN) differ from a wired LAN in several aspects that affect MAC protocols:

- It is usually not possible to detect collisions as they occur via ethernet-style listen-while-sending because of the great difference in transmitter and receiver power levels.
- Channel errors and packet loss rates are non-negligible.
- Error control techniques can require link turnaround times of hundreds to thousands of milliseconds [2].
- Broadcast connectivity may be incomplete.
- "Hidden nodes" make listen-before-sending insufficient to avoid collisions.
- Node mobility can make connectivity and network membership more dynamic than in wired networks.

When the topology is a star of nodes that are connected by a wireless "last hop" to a base station, a polling protocol such as the IEEE 802.11 Point Coordination Function (PCF) can provide contention-free access control and can guarantee quality of service (within the limitations of the channel). When the topology is "ad hoc," a collision avoiding variant of IEEE 802.3, such as the Distributed Coordination Function (DCF) of IEEE 802.11, may suffice (when traffic is light).

A recent investigation of the impact of link turnaround time on the performance of MAC protocols [2] suggested that when turnaround times are long compared to control packets, a token-passing protocol might be preferred over a contention-based MAC protocol, even when traffic is light. However, achieving the expected efficiency of token passing requires that the protocol not impose significant delays in normal operation, when adding or dropping nodes, or while recovering from exceptional conditions such as token loss or duplication. In this paper, we propose a distributed token management protocol that is intended to provide efficient operation in both benign and challenging situations.

2. TOKEN PASSING PROTOCOLS

A token passing MAC protocol controls access to the shared medium through the notion of a "token" which grants its possessor the right to transmit, usually for a bounded time. After taking its turn to transmit, the node holding the token is obliged to send a control packet that represents the token to another node. When a node receives the token but has no traffic to send, it must pass the token immediately.

Several token passing protocols are described in the literature, and some have seen widespread implementation. In general, the token is passed from one node to another following a logical ring. However, the underlying physical topology may be either a broadcast medium or a ring of point-to-point links.

- In the former case, the network is called a "token bus," even though the physical connection among nodes may be any broadcast medium (including a wireless channel).
- In the latter case, the network is called a token ring, and data packets are passed around a physical ring of point-to-point links (with one bit per node delay in a wired LAN).

Token bus protocols seem a better fit for wireless LANs, and are the focus of this paper

2.1 Issues for Token Passing in Wireless Networks

Token passing protocols generally provide mechanisms for nodes to enter and leave the network. When token passing is to be used in a WLAN, the characteristics of the wireless medium (section 1) raise additional token management issues:

* Supported by US Navy SPAWAR Systems Center, San Diego under contract N6601-97-D-5028.

a. The node holding the token may lose connectivity to its successor, which can result in a lost token.
b. The node holding the token can lose connectivity to the rest of the network. The network loses the token.
c. A network may become partitioned. One subnetwork must create a new token.
d. A node may be reachable by only one other node, so a ring topology is not possible if that node is to be included.
e. Nodes from two or more rings using the same channel may come within range of each other. This results in interference unless the rings merge or change channel(s).
f. Merging of rings or recovery from a lost token may result in multiple tokens in a ring.

In the following sections, we review significant token passing MAC protocols, and note their solutions to these issues.

2.2 IEEE 802.4 Token Bus

The IEEE 802.4 Token Bus protocol is based on a broadcast medium (broadband coaxial cable), which connects all nodes to each other. The token is passed among a logical ring of nodes attached to the cable (Figure 1). The nodes sort themselves for order of token passing by their MAC addresses.

Each node stores the MAC address of its predecessor and successor in the token passing order. When a node is leaving the ring, it sends a SET_SUCCESSOR packet to its predecessor that links the predecessor to the leaving node's successor.

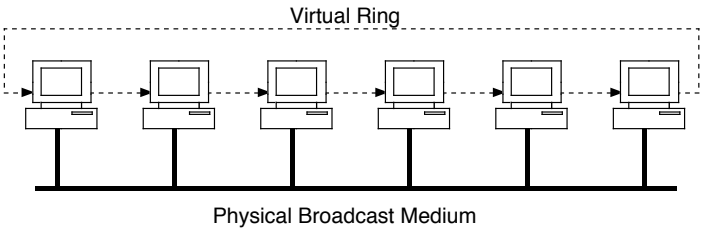

Figure 1. Token Bus Protocol

Joining the ring is a bit more complicated. Nodes occasionally broadcast a SOLICIT_SUCCESSOR packet. This packet contains the MAC addresses of the soliciting node and its current successor.

- Any node that is waiting to join the ring, and whose address falls between the specified addresses, responds and joins the ring between the soliciting node and its successor.
- If multiple nodes respond, their responses collide, and the protocol enters a contention resolution phase. A countdown protocol that relies upon the listen-while-sending capability of the bus eliminates all but one contending node.

Note that the joining node(s) must wait to be invited to join, and that the waiting time is unbounded.

The 802.4 standard includes timers to recover from a lost token (issues **a**, **b**, and **c**).

The solution to multiple tokens (issue **f**) relies on the broadcast medium: every node hears transmissions by every other node. Therefore, any node holding a token that overhears a transmission from any other node simply discards its token. Issues **d** and **e** are not expected to arise on a cable, and are not addressed.

2.3 Wireless Token Ring Protocol

The Wireless Token Ring Protocol (WTRP) [3] is in fact a token bus protocol, derived from IEEE 802.4. The principal modifications of 802.4 that are introduced by WTRP address the partial connectivity issues that arise in wireless networks.

WTRP was developed for mobile wireless applications (such as clusters of unmanned aerial vehicles [3]). Figure 2 depicts a possible use of WTRP in a naval battle group WLAN.

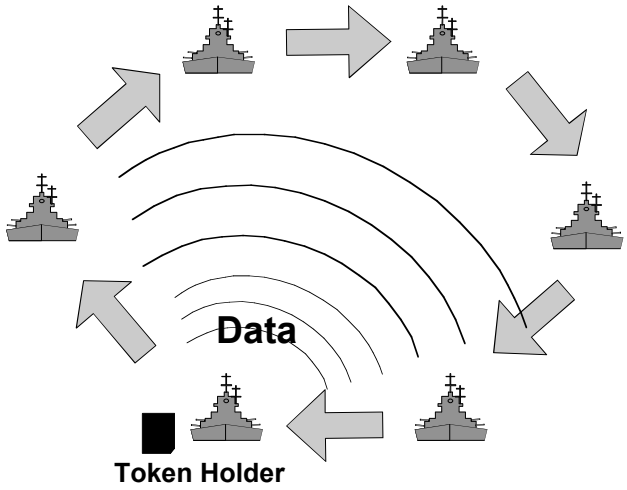

Figure 2. Token Passing in a Maritime WLAN

WTRP inherits the SET_SUCCESSOR and SOLICIT_ SUCCESSOR mechanisms from IEEE 802.4 for dropping and adding nodes to the network. In Figure 3, node B is in a "Floating" state, waiting to join the network. In Figure 4, node A solicits a successor. B responds, and enters the net.

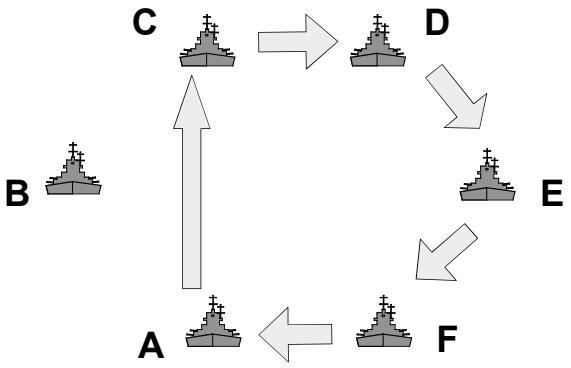

Figure 3. Net Entry: Floating State

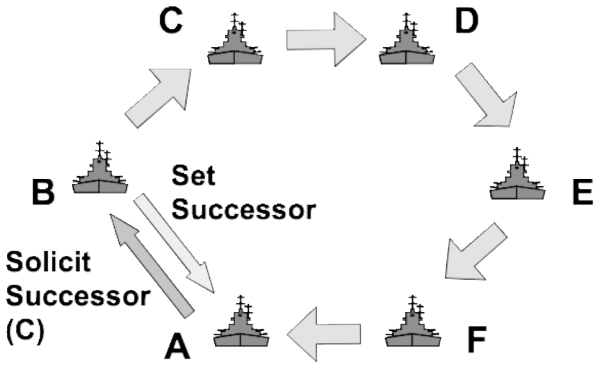

Figure 4. Net Entry: Joining

Nodes in a network using WTRP maintain a Connectivity Table that lists network members in the order that the token visits them. When a link in the token path is lost, WTRP attempts to reconnect the ring while minimizing the number of nodes that are dropped from the ring in the process.

In Figure 5, node A fails to pass the token to B due to a link outage. Node A consults its Connectivity Table to find the next node in the ring, and reconnects to C by sending a SET_PREDECESSOR packet (Figure 6). This unfortunately excludes node B, which is still reachable from other nodes.

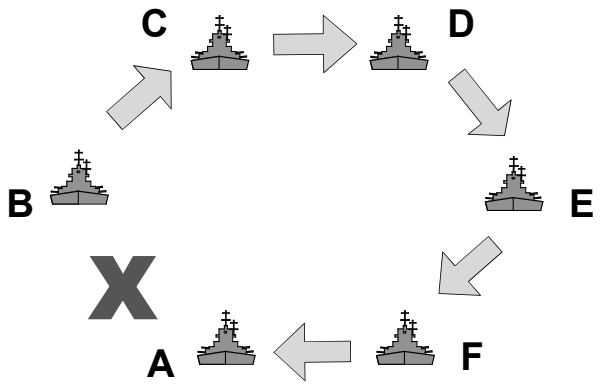

Figure 5. Link Outage

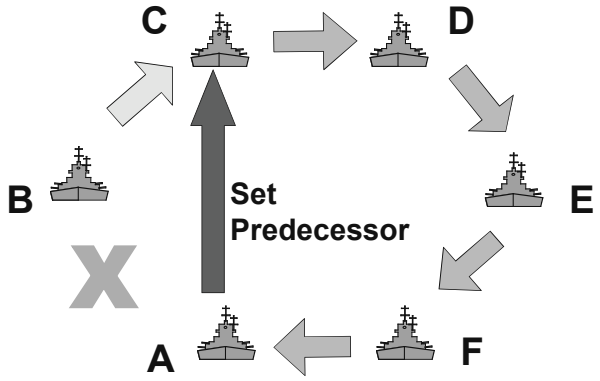

Figure 6: WTRP Reconnects Ring, Excluding B

Thus the WTRP response to issue **d** (limited reachability) is to exclude "singleton" node(s). The excluded node(s) must then wait for an invitation to rejoin the ring.

Another situation not addressed by IEEE 802.4 is issue **e**. A ring collision occurs when independent rings using the same channel come into range of each other (Figure 7). A token is present in each ring, so mutual interference will result if both rings continue to operate normally. A mechanism is required to detect this situation, and to merge the rings.

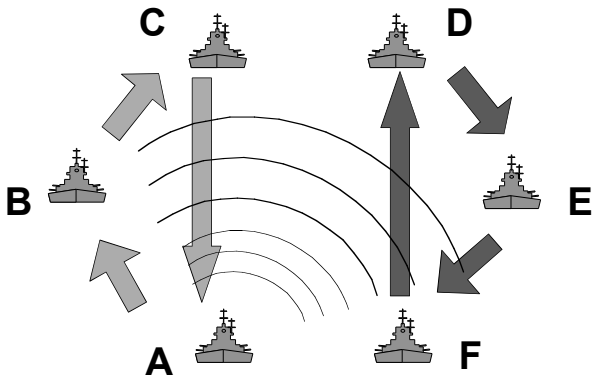

Figure 7. Colliding Rings

WTRP adds a Ring Address field to the token that identifies the ring to which it belongs. (This field contains the MAC address of the node that most recently created the token.) When a node overhears a token that is not from its ring, that node reverts to a ring-forming state, similar to the Floating state in Figure 3. Thus, colliding WTRP rings will disintegrate and form a larger ring using the ring startup mechanism.

WTRP inherits solutions to issues **a**, **b**, and **c** from IEEE 802.4: token loss is addressed using timers. However, issue **f**, resolution of multiple tokens, cannot be solved using the 802.4 approach, because WLANs may be only partially connected. Instead, WTRP introduces the notion of token "priority" computed from the Ring Address and Generation Sequence Number fields in the token. Each node stores the priority of the token it most recently generated or accepted, and deletes any tokens it receives that fall below that priority. Within one token rotation, only one token remains in the ring.

The WTRP enhancements to IEEE 802.4 address all of the issues raised in section 2.1, but some aspects of the WTRP solutions may not be attractive for military networks:

1. WTRP responses to exceptional conditions are biased in the direction of disconnecting nodes. Excluded nodes can eventually rejoin the network, but they may remain disconnected for an unbounded time.
2. COMSEC devices and robust modems can lengthen link turnaround times to hundreds or thousands of milliseconds and token rotation times (even in small networks) to seconds or even minutes. Thus it becomes important to fix problems as soon as they are detected rather than resolving them in the course of several token rotations.

3. HF TOKEN PROTOCOL

Token passing is attractive for use in surface-wave naval high-frequency (HF) radio networks due to its potential for high throughput, fairness, and bounded access time. This is a challenging application for token passing, though, due to the dynamic nature of the HF channel and the long link turnarounds that are inherent in fielded COMSEC and HF modems. If token passing is to succeed in this environment, a robust token management protocol will be required that maximizes network availability for all nodes despite packet loss and fluctuating connectivity.

The HF token protocol (HFTP) emphasizes fast recovery from disruptions, inclusiveness in retaining all reachable nodes, and tolerance for long link turnaround times. HFTP is based on WTRP, but adds two new mechanisms: token relaying and a ring merging procedure.

3.1 Token Relay

We first revisit the situation depicted in Figure 5, in which a node attempts to pass the token to its successor, but fails to receive acknowledgement due to a link outage. HFTP will attempt to find an indirect path to its successor rather than reconnecting the ring to exclude that node. This requires new mechanisms to find and to use token relay nodes.

The first step in resuming token passing operation after loss of the link to a node's successor is finding another network node that can serve as a relay for passing the token. A SOLICIT_RELAY packet will be sent by the node that failed to pass the token (Figure 8). This packet carries a list of network nodes copied from the sender's Connectivity Table, in token passing order, starting with the sender's successor. (The number of nodes in the ring is known to all network members by observing the Sequence Number field in tokens, just as in WTRP.) Some entries may contain "Unknown" as a placeholder if the sender cannot hear their transmissions.

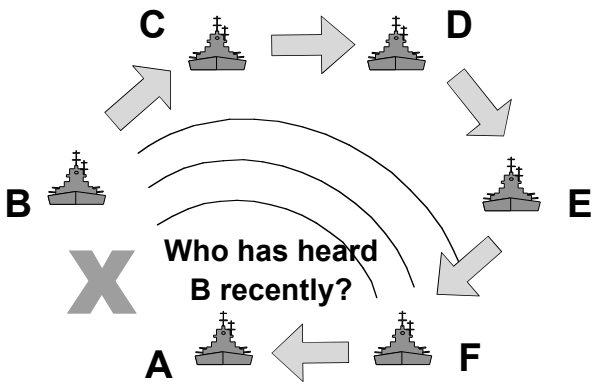

Figure 8. Solicit Relay

Network members that hear this SOLICIT_RELAY respond in slots (Figure 9), in the order that nodes are named in the packet. Member nodes not named in the packet choose randomly among the slots labeled Unknown for their responses.

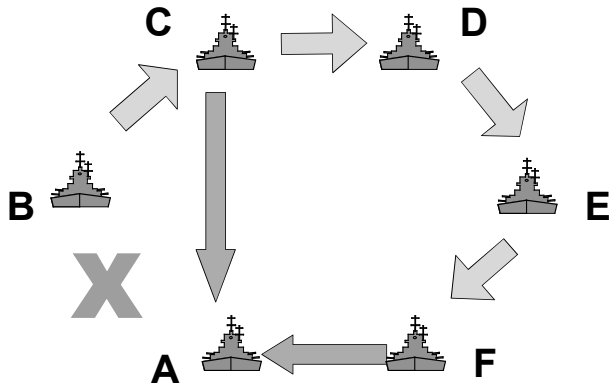

Figure 9. Relay Offers

Each response carries a flag indicating whether the responding node overheard a transmission from the desired destination node during the last token rotation. After the responses have been received, the soliciting node selects the first positive response (if any) as its temporary Relay node for packets to its Successor. This relaying relationship is recorded only at the sending node (A in the figures), and is retained until the sender overhears a packet from its successor.

As long as the relaying relationship holds, A will not attempt to send the token directly to B, but will instead send a RELAY_TOKEN(B) packet to its relay node (Figure 10). The relay node (C) will then attempt to pass a normal token to B. B will use the token as usual, and then pass it to its Successor (which happens to be C in this example).

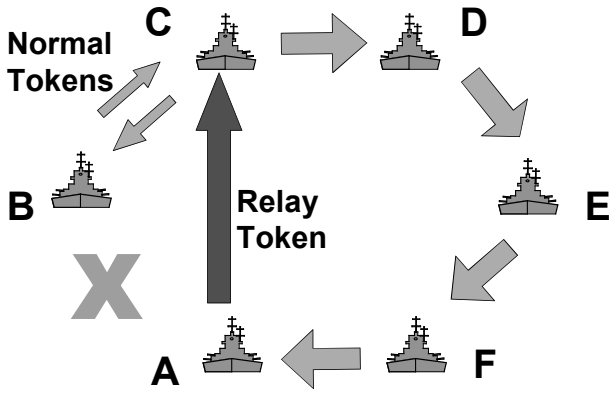

Figure 10. Token Relay in Operation

Of course, it may be the case that the Relay node is now unable to reach the desired token destination, despite recent connectivity. After several unsuccessful attempts to relay a token, a node will declare that node unreachable and so inform the node that requested the relay. At this point, the ring will be reconnected, excluding the former successor (Figure 11). Thus, HFTP will make a determined attempt to recover connectivity to a node that becomes unreachable from its predecessor, but if that attempt fails, HFTP will fall back to the WTRP approach for eventual recovery of the lost node.

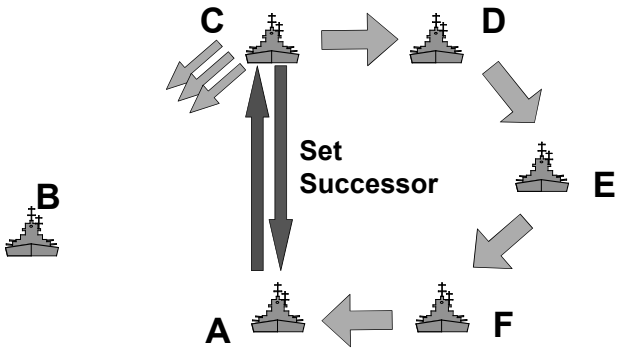

Figure 11. Token Relay Fails

3.2 Merging Rings

HFTP also differs from WTRP in its mechanism for merging rings that come into range of each other. (This can occur after a network that was partitioned regains connectivity.) In Figure 12, our network was at some point partitioned into two rings, A-B-C and D-E-F. Some of the nodes have now come into range of each other: in the figure, F overhears a transmission from A.

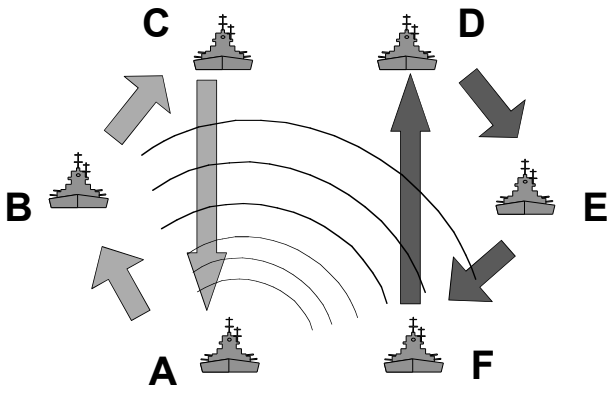

Figure 12. Colliding Rings

Each node that overhears a transmission from a "foreign" ring whose token priority is higher than its own enters a "Want To Merge" state. When such a node next holds the token in its ring, it sends MERGE_RINGS packets to the node that it heard in the foreign ring (Figure 13) until that node acknowledges the merger. Collisions are expected because the foreign ring has a token of its own in circulation.

The MERGE_RINGS request carries the node ID of the successor of the node sending the request (here D is the successor of F). The node that receives and accepts the merger request (A) records the sender (F) as its new predecessor and sends a SET_SUCCESSOR packet to its former predecessor (C), as shown in Figure 14. This serves to complete the reconnection of two rings into one (Figure 15).

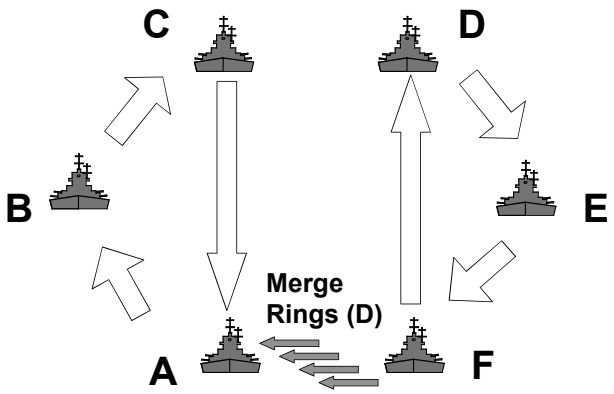

Figure 13. Merge Rings Request

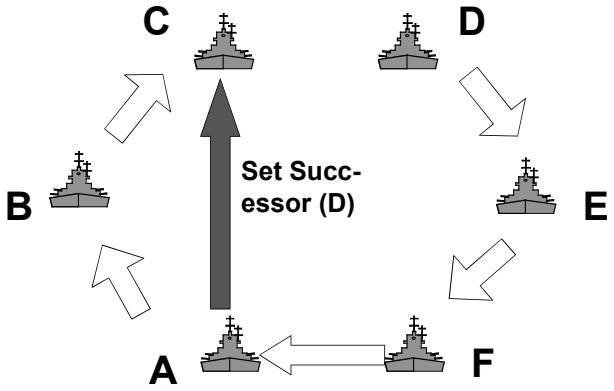

Figure 14. Rings Merging (Step 2)

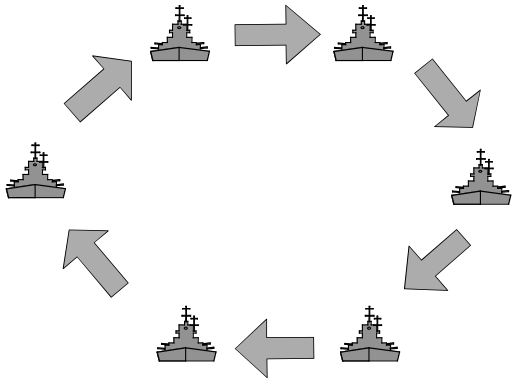

Figure 15. Rings Merged

When the node that initiated the merger (F) receives an acknowledgement to its merge request, it sends a special DOUBLE_TIME_TOKEN to its former successor. The DOUBLE_TIME_TOKEN must be passed immediately, and does not authorize transmission of data. This fast-moving token serves two functions:

- Reception of the DOUBLE_TIME_TOKEN returns any node in the WantToMerge state to Idle. This ensures that other nodes that may have noticed the ring collision do not mangle the newly merged ring by initiating mergers using old connectivity data..
- The priority (via the Generation Sequence Number) of the DOUBLE_TIME_TOKEN is set to a value greater than that of the token in either ring before the merger. It will therefore purge any remaining tokens from the new ring as it circulates.

The DOUBLE_TIME_TOKEN is converted to a normal token after it has returned to the node that created it.

By allowing only nodes in the lower-priority network of two colliding rings to initiate the merger, we preclude race conditions between the networks.

4. DISCUSSION

Two new token management mechanisms were presented here: token relaying and deliberate merging of rings. Both differ from the corresponding WTRP mechanisms in that they attempt to solve connectivity problems as soon as they are detected, and without disconnecting reachable nodes. The WTRP approach to recovery from connectivity problems places "troublesome" nodes into a disconnected or floating state in which they either wait to be invited to join the remaining ring or periodically solicit other disconnected nodes to join with them.

The long link turnarounds inherent in fielded HF WLAN technology result in token rotation times on the order of a minute. For example, if link turnaround times are 2 seconds and we allow each of N nodes to transmit for up to 8 seconds when it receives the token, we achieve a throughput efficiency of at most 80% with a token rotation time (latency) of up to 10 N seconds.

If we limit solicitations to join the ring to one per token rotation, and rotate the authority to solicit among the nodes, each node will solicit once in N token rotations. With ten nodes in a ring, use of WTRP would result in disconnected nodes remaining out of the network for around ten minutes (if there are no colliding responses to the eventual SOLICIT_SUCCESSOR); this not an attractive mode of operation for a military network.

The time required for WTRP to re-form a new ring from the disconnected remains of two colliding rings would be at least that long: a small ring might emerge quickly, but the remaining nodes would then go silent and wait to be invited to join.

The recovery times for HFTP are more attractive. In the case of a lost link, HFTP requires N slots (whose duration equals a packet plus a turnaround time) to identify a relay. Thereafter, one additional packet time and turnaround time are required *in each token rotation*. In our example ten-node network, this amounts to a pause of less than 30 seconds while identifying the relay, and lengthening the token rotation time by a bit over 2%.

In the case of colliding rings, HFTP networks will experience packet collisions until one of the nodes initiates the ring merger, while WTRP nodes will go silent as soon as they detect the foreign ring. However, once a MERGE_RINGS request is received and accepted, the merging rings will resume normal data transfers after (N + 1) packet + turnaround times (i.e., after the SET_SUCCESSOR and the fast token rotation of the DOUBLE_TIME_TOKEN). This amounts to less than 30 seconds in our example network.

5. CONCLUSIONS

Token passing can provide efficient channel sharing under traffic loads ranging from light to saturation [2], but this efficiency could suffer in wireless networks if packet losses and link outages occur frequently.

The HF Token Protocol HFTP was designed to recover quickly from these problems, even with link turnaround times on the order of seconds. HFTP builds upon IEEE 802.4 and WTRP by adding mechanisms for relaying tokens around link outages and for merging rings without disintegrating those rings. The initial application for HFTP is maritime wireless LANs, where it is intended to be used as the MAC protocol with the STANAG 5066 data link protocol. If sea trials of this combination are successful, HFTP will be offered as a potential addition to STANAG 5066.

Future work in this area will investigate opportunities to integrate connectivity information from a routing protocol to improve the performance and resilience of HFTP.

REFERENCES

1. A.S. Tanenbaum, *Computer Networks*, Prentice-Hall, 2002.
2. E.E. Johnson, *et al*, "Impact of Turnaround Time on Wireless MAC Protocols," *Proceedings of MILCOM 2003*, IEEE, 2003.
3. M. Ergen, *et al*., "Wireless Token Ring Protocol," *SCI 2002*.