INFORMATIQUE THÉORIQUE ET APPLICATIONS

J. PACHL

A proof of protocol correctness

Informatique théorique et applications, tome 28, n° 3-4 (1994), p. 213-220

<http://www.numdam.org/item?id=ITA_1994__28_3-4_213_0>

Informatique théorique et Applications/Theoretical Informatics and Applications
(vol. 28, n° 3-4, 1994, p. 213 à 220)

A PROOF OF PROTOCOL CORRECTNESS

by J. PACHL ([1])

Abstract. – *The paper presents a formal description of the medium access protocol used in the Cambridge Ring, and a proof that the protocol recovers from transient transmission errors.*

1. INTRODUCTION

Medium access protocols for local area networks must recover from transient transmission errors. This leads to challenging design problems, since errors may corrupt the control information used by the protocol for recovery.

To control medium access in ring networks, certain protocols use permissions to transmit (empty slots in slotted rings, tokens in token rings), which are passed from node to node around the ring. Often a single bit distinguishes an empty slot from full or a token from a data frame, and an error may invert the value of that bit. For example, a full slot could be accidentally marked empty as a result of a single transmission error; that would produce an inconsistent view of the protocol state among the nodes in the network, which could then persist for a long time. A correct protocol must recover from such situations.

Wheeler [3] explains how the Cambridge Ring protocol recovers from transmissions errors that could otherwise lead to livelocks. The present paper contains a formal proof that the protocol described in [3] is indeed "correct" (in the sense defined below). Although the formal proof is certainly longer and more difficult to read than the explanation in [3], it offers additional benefits: It gives us confidence that we have not overlooked any (even very unlikely) execution scenarios. As a by-product of the proof we also obtain a bound on the length of the recovery phase.

([1]) Box 65056, 358 Danforth Avenue, Toronto, Ontario M4K 322, Canada.

Informatique théorique et Applications/Theoretical Informatics and Applications
0988-3754/94/03-04/$ 4.00/

The present paper uses the notation and the specification approach developed in [1]. The reader is referred to [1] for a more detailed discussion of the underlying model and of related issues.

2. NOTATION

The cardinality of a finite set S is $\#S$. The negation of a predicate P is $\neg P$.

We shall use the same notation as in [1]. The communication network is a unidirectional ring with N nodes, numbered $0, 1, \ldots, N-1$. The unidirectional communication channels lead from node 0 to node 1, from node 1 to node 2, etc., and from node $N-1$ to node 0. There is a single frame circulating in the ring. For integers i and u, $0 \leq i < N$, $u \geq 0$, let $S^{\bullet}_{i,u}$ be the content of the u-th frame sent by node i, and let $R^{\bullet}_{i,u}$ be the content of the u-th frame received by node i.

The statement that node i sends the u-th received frame without changes is written

$$S^{\bullet}_{i,u} = R^{\bullet}_{i,u}. \tag{1}$$

The statement that the u-th frame sent by node i is delivered without transmission errors is written

$$R^{\bullet}_{i+1,u} = S^{\bullet}_{i,u} \tag{2}$$

when $i < N-1$, and

$$R^{\bullet}_{0,u+1} = S^{\bullet}_{i,u} \tag{3}$$

when $i = N-1$. It is sometimes more convenient to use the notation

$$R_p = R^{\bullet}_{i,u}, \quad S_p = S^{\bullet}_{i,u}$$

where $p = uN + i$, $0 \leq i < N$, $u \geq 0$. Then (1) is equivalent to

$$S_p = R_p,$$

while (2) and (3) are equivalent to

$$R_{p+1} = S_p.$$

The content of a frame is an array of bits. The b-th bit of a frame F is denoted $F[b]$. In the protocol in section 3, the frame has $B+1$ bits; the frame consists of two control bits $F[0]$ and $F[1]$ and $B-1$ bits in the data part $F[2 \ldots B]$.

Every node has a *client* (a higher-level protocol), which occasionally offers an array of exactly $B-1$ to be broadcast over the network. For integers i and u, $0 \leq i < N$, $u \geq 0$, let $\mathrm{data}^{\bullet}_{i,u}$ be either a special value nil or an array of $B-1$ bits. If $\mathrm{data}^{\bullet}_{i,u} \neq \mathrm{nil}$ then $\mathrm{data}^{\bullet}_{i,u}$ is the data block available from the client of node i so that it can be sent in the u-th frame (and broadcast around the ring). If $\mathrm{data}^{\bullet}_{i,u} = \mathrm{nil}$ then no data are to be sent at that time.

A *protocol* defines each $S^{\bullet}_{i,u}$ in terms of $(\mathrm{data}^{\bullet}_{j,u} | 0 \leq j \leq i)$, $(\mathrm{R}^{\bullet}_{j,u} | 0 \leq j \leq i)$, and $(\mathrm{S}^{\bullet}_{j,u} | 0 \leq j < i)$.

When we discuss the correctness of protocols in this setting, it is useful to view the protocols as implementing a broadcast service. The data offered by the client of the protocol should be transmitted in the frame around the ring and thus received by all the nodes including the sender. However, sometimes the data broadcast around the ring are corrupted by a transmission error. What then should the correctness of the protocol mean?

The following definition captures an important notion of correctness: If there are no transmission errors after time t, then there exists time $t' > t$ such that all data sent after t' are correctly broadcast around the ring. This property is called *eventual relability* in [1]; it belongs to the family of *self-stabilization* properties, which have been extensively investigated in recent years [2]. Note that no assumption is made about the number and pattern of transmission errors before time t, and no claim is made about protocol operation before time t'.

In this paper we prove that the Cambridge Ring protocol described by Wheeler [3] has the correctness property defined in the previous paragraph. Indeed, the theorem in the next section states that, beginning at some time after the last transmission error, if a node sends data from its client then no other node sends its data until the frame completes its round trip.

Eventual reliability as just defined does not imply fairness. We do not deal with fairness issues in this paper. It is easy to see that the Cambridge Ring protocol is fair: The opportunity to transmit the client data is circulated around the ring.

3. THE PROTOCOL

The protocol makes use of a special node in the ring, the *monitor* node. In our notation, the monitor is node number 0; it has no client, and thus it never sends its own data. Therefore

$$\mathrm{data}^{\bullet}_{0,u} = \mathrm{nil} \qquad \text{for} \quad u \geq 0. \tag{4}$$

The frame F contains two control bits, $F[0]$ and $F[1]$:

$F[0]$	$F[1]$	$F[2 \ldots B]$

The control bit $F[0]$ is called the *full/empty bit* The frame is marked empty when $F[0] = 0$; it is marked full when $F[0] = 1$. The control bit $F[1]$ is called the *monitor bit.* It is used by the monitor node.

For $0 \leq i < N$, $u \geq 0$, define

$$\text{send}^\bullet(i, u) = (u \geq 1 \quad \text{and} \quad \neg\text{send}^\bullet(i, u-1) \quad \text{and} \quad \text{data}^\bullet_{i,u} \neq \text{nil} \quad \text{and} \quad R^\bullet_{i,u}[0] = 0). \tag{5}$$

Note that $\neg\text{send}^\bullet(0, u)$, by (4). Now $S^\bullet_{i,u}$ is defined separately for the monitor node and for the ordinary (non-monitor) nodes:

Case $i = 0$ (monitor):

$$S^\bullet_{i,u}[0] = R^\bullet_{i,u}[1] \tag{6}$$

$$S^\bullet_{i,u}[1] = 0 \tag{7}$$

$$S^\bullet_{i,u}[2 \ldots B] = R^\bullet_{i,u}[2 \ldots B] \tag{8}$$

Case $i \neq 0$ (non-monitor):

$$S^\bullet_{i,u}[0] = \begin{cases} 1 & \text{if} \quad \text{send}^\bullet(i, u) \\ 0 & \text{if} \quad \text{send}^\bullet(i, u-1) \\ R^\bullet_{i,u}[0] & \text{otherwise} \end{cases} \tag{9}$$

$$S^\bullet_{i,u}[1] = \begin{cases} 1 & \text{if} \quad \text{send}^\bullet(i, u) \\ R^\bullet_{i,u}[1] & \text{otherwise} \end{cases} \tag{10}$$

$$S^\bullet_{i,u}[2 \ldots B] = \begin{cases} \text{data}^\bullet_{i,u} & \text{if} \quad \text{send}^\bullet(i, u) \\ R^\bullet_{i,u}[2 \ldots B] & \text{otherwise} \end{cases} \tag{11}$$

When $0 \leq i, j < N$, define

$$\text{SEND}^\bullet_u[i, j] = \{k | i \leq k \leq j \quad \text{and} \quad \text{send}^\bullet(\text{k}, \text{u})\}.$$

THEOREM: *For the protocol defined by* (4), (5), (6), (7), (8), (9), (10) *and* (11), *let* $r_0 \geq 0$ *be such that* $R_{p+1} = S_p$ *for* $p \geq r_0 N$, *and*

let $u \geq r_0 + ((1/2)\ N - 1)^2$. *If* $0 < i < N$ *and* $\text{send}^\bullet\ (i,\ u)$ *then* $\text{SEND}^\bullet_u\ [i + 1,\ N - 1] = \varnothing = \text{SEND}^\bullet_{u+1}\ [0,\ i]$.

The theorem states that, no matter what transmission errors occur before the r_0-th round trip of the frame, if no errors occur during the r_0-th and subsequent round trips then the network statilizes within $((1/2)\ N - 1)^2$ round trips. That is, the network reaches the state in which whenever a node sends data from its client, the data are forwarded once around the ring without interference from other nodes.

4. PROOF OF THE THEOREM

From now on we assume that (4), (5), (6), (7), (8), (9), (10) and (11) hold, and that there is $r_0 \geq 0$ such that $R_{p+1} = S_p$ for $p \geq r_0\ N$.

For $u \geq 0$, define

$$\Gamma\ (u) = \sharp\text{SEND}^\bullet_u\ [1,\ N - 1].$$

The crucial fact to be established in the proof is that $\Gamma\ (u) \leq 1$ for $u \geq r_0 + ((1/2)\ N - 1)^2$.

LEMMA 1: *If* $u \geq r_0 - 1$, $0 \leq m < j < N$, $\text{SEND}^\bullet_u\ [m + 1,\ j] = \varnothing$ *and* $S^\bullet_{m,\,u+1}\ [0] = 1$ *then* $\text{SEND}^\bullet_{u+1}\ [m + 1,\ j] = \varnothing$.

Proof: Since $u + 1 \geq r_0$, we have $S^\bullet_{k,\,u+1} = R^\bullet_{k+1,\,u+1}$ for $m \leq k < j$. Thus, since $S^\bullet_{m,\,u+1}[0] = 1$, from (9) we have $R^\bullet_{k,\,u+1}[0] = 1$ for $m < k \leq j$. Hence $\neg\text{send}^\bullet\ (k,\ u + 1)$ for $m < k \leq j$, by (5). □

LEMMA 2: *If* $u \geq r_0 - 1$, $0 < i < j < N$ *and* $\text{SEND}^\bullet_u\ [i,\ j] - \varnothing$ *then* $\sharp\text{SEND}^\bullet_{u+1}\ [i,\ j] \leq 1$.

Proof: If $\text{SEND}^\bullet_{u+1}\ [i,\ j] \neq \varnothing$, let $m = \min\ \text{SEND}^\bullet_{u+1}\ [i,\ j]$. Then $S^\bullet_{m,\,u+1}\ [0] = 1$ by (9) and $\text{SEND}^\bullet_{u+1}\ [\text{m} + 1,\ \text{j}] = \varnothing$ by Lemma 1. □

LEMMA 3: *If* $u \geq r_0 - 1$, *and* $\Gamma\ (u) = 0$ *then* $\Gamma\ (u + 1) \leq 1$.

Proof: Set $i = 1$ and $j = N - 1$ in Lemma 2. □

LEMMA 4: *If* $u \geq r_0 - 1$, $0 < i < j < N$ *and* $\text{send}^\bullet\ (i,\ u)$ *then* $\sharp\text{SEND}^\bullet_u\ [i,\ j] \geq \sharp\text{SEND}^\bullet_{u+1}\ [i,\ j]$.

Proof: Decompose the interval $[i,\ j]$ into subintervals $[i',\ j']$ such that $\text{SEND}^\bullet_u\ [i',\ j'] = \{i'\}$. Then $\sharp\text{SEND}^\bullet_{u+1}\ [i' + 1,\ j'] \leq 1$ by Lemma 2,

and $\neg\text{send}^\bullet(i', u+1)$ by (5). Sum over the subintervals $[i', j']$ to obtain the result. □

LEMMA 5: *If* $u \geq r_0$ *and* $\Gamma(u) \geq 1$ *then* $S^\bullet_{0,u+1}[0] = 1$.

Proof: Since $u \geq r_0$, we have $S^\bullet_{i,u} = R^\bullet_{i+1,u}$ for $0 \leq i < N-1$ and $S^\bullet_{N-1,u} = R^\bullet_{0,u+1}$. Since $\Gamma(u) \geq 1$, by (10) we have $S^\bullet_{j,u}[1] = 1$ for some j, $0 < j < N$. Thus from (10) it follows that $R^\bullet_{0,u+1}[1] = 1$, and therefore $S^\bullet_{0,u+1}[0] = 1$ by (6). □

LEMMA 6: *If* $u \geq r_0$, *and* $\Gamma(u) \geq 1$ *then* $\sharp\text{SEND}^\bullet_u[1, j] \geq \sharp\text{SEND}^\bullet_{u+1}[1, j]$ *for* $0 < j < N$.

Proof: We have $S^\bullet_{0,u+1}[0] = 1$, by Lemma 5. Let $i = \min \text{SEND}^\bullet_u[1, N-1]$. From Lemma 1 (with $m = 0$) we obtain $\text{SEND}^\bullet_{u+1}[1, i] = \varnothing$. If $i \geq j$ then

$$\sharp\text{SEND}^\bullet_{u+1}[1, j] \leq \sharp\text{SEND}^\bullet_{u+1}[1, i] = 0.$$

If $i < j$ then

$$\sharp\text{SEND}^\bullet_u[1, j] = \sharp\text{SEND}^\bullet_u[i, j] \geq \sharp\text{SEND}^\bullet_{u+1}[i, j] = \sharp\text{SEND}^\bullet_{u+1}[1, j]$$

by Lemma 4. □

LEMMA 7: *If* $u \geq r_0$, *and* $\Gamma(u) \geq 1$ *then* $\Gamma(u) \geq \Gamma(u+1)$.

Proof: Set $j = N-1$ in Lemma 6. □

LEMMA 8: *If* $u \geq r_0$, $\text{send}^\bullet(i, u)$ *and* $\text{SEND}^\bullet_{u+1}[i, N-1] = \varnothing$ *then* $\Gamma(u) > \Gamma(u+1)$.

Proof: This is obvious for $i = 1$. If $1 < i < N$, set $j = i-1$ in Lemma 6. Then

$$\begin{aligned}\Gamma(u) = \sharp\text{SEND}^\bullet_u[1, N-1] &> \sharp\text{SEND}^\bullet_u[1, j] \\ &\geq \sharp\text{SEND}^\bullet_{u+1}[1, j] = \Gamma(u+1).\end{aligned}$$ □

LEMMA 9: *If* $u \geq r_0$ *and* $\Gamma(u) = \Gamma(u+1) \geq 1$ *then*

$$\max \text{SEND}^\bullet_u[1, N-1] < \max \text{SEND}^\bullet_{u+1}[1, N-1].$$

Proof: Set $i = \max \text{SEND}^\bullet_u[1, N-1]$ in Lemma 8. □

LEMMA 10: *If* $u \geq r_0$ *then* $1 + \max \text{SEND}^\bullet_u[1, N-1] \geq 2\Gamma(u)$.

Proof: Let $i = \max \text{SEND}_u^\bullet [1, N-1]$. Thus $\text{SEND}_u^\bullet [1, N-1] = \text{SEND}_u^\bullet [1, i]$. We have

$$\text{SEND}_{u-1}^\bullet [1, i] \cap \text{SEND}_u^\bullet [1, i] = \varnothing$$

by (5), and

$$1 + \sharp\text{SEND}_{u-1}^\bullet [1, i] \geq \text{SEND}_u^\bullet [1, i]$$

by Lemma 2 and Lemma 4. It follows that $2\sharp\text{SEND}_u^\bullet [1, i] \leq i + 1$. □

LEMMA 11: *If* $u \geq r_0 + ((1/2) N - 1)^2$ *then* $\Gamma(u) \leq 1$.

Proof: For $u \geq r_0$, define

$$\Delta(u) = N - 1 - \max \text{SEND}_u^\bullet [1, N-1].$$

By Lemma 7 and Lemma 9, if $u \geq r_0$ and $\Gamma(u) \geq 1$ then either

$$\Gamma(u) > \Gamma(u+1)$$

or

$$\Gamma(u) = \Gamma(u+1) \quad \text{and} \quad \Delta(u) > \Delta(u+1).$$

Since $\Gamma(u)$ and $\Delta(u)$ are bounded, it follows that $\Gamma(u') = 0$ for some $u' \geq r_0$. From Lemma 3 and Lemma 7 it then follows that $\Gamma(u) \leq 1$ for $u \geq u'$. To derive an estimate on $u' - r_0$, we use the inequality

$$\Delta(u) + 2\Gamma(u) \leq N \tag{12}$$

which holds for $u \geq r_0$ by Lemma 10.

For $1 \leq k \leq \lfloor (1/2) N \rfloor$ define

$$u_k = \min \{u | u \geq r_0 \quad \text{and} \quad \Gamma(u) \leq k\}.$$

Thus

$$u_{\lfloor (1/2) N \rfloor} \leq u_{\lfloor (1/2) N \rfloor - 1} \leq \cdots \leq u_2 \leq u_1$$

and $\Gamma(u_1) \leq 1$; moreover $u_{\lfloor (1/2) N \rfloor} = r_0$ by (12).

We shall estimate $u_{k-1} - u_k$ for $k = 2, 3, \ldots, \lfloor (1/2) N \rfloor$. If $\Gamma(u_k) \leq k-1$ then $u_{k-1} - u_k = 0$. If $\Gamma(u_k) = k$ then $\Gamma(u) = k$ for $u_k \leq u < u_{k-1}$. Therefore

$$\Delta(u_k) > \Delta(u_k + 1) > \ldots > \Delta(u_{k-1} - 1) \geq 0,$$

hence

$$u_{k-1} - u_k \leq \Delta(u_k) + 1 \leq N - 2k + 1$$

by (12). Summing over k from 2 to $\lfloor (1/2)\, N \rfloor$ we get

$$u_1 - r_0 = u_1 - u_{\lfloor (1/2)\, N \rfloor} \leq \sum_{k=2}^{\lfloor (1/2)\, N \rfloor} (N - 2k + 1) \leq \left(\frac{1}{2}\, N - 1 \right)^2 . \quad \square$$

Proof of the theorem in section 3: In view of Lemma 11, it is enough to prove that if $u \geq r_0$, $0 < i < N$, $\Gamma(u) \leq 1$ and $\text{send}^\bullet\ (i, u)$ then $\text{SEND}^\bullet_u\ [i + 1,\ N - 1] = \varnothing = \text{SEND}^\bullet_{u+1}\ [0,\ i]$. We get immediately $\text{SEND}^\bullet_u [0, i-1] = \varnothing = \text{SEND}^\bullet_u [i+1, N-1]$ because $\Gamma(u) \leq 1$. In addition, $S^\bullet_{0,\, u+1}\ [0] = 1$ by Lemma 5. Therefore $R^\bullet_{k+1,\, u+1}\ [0] = S^\bullet_{k,\, u+1}\ [0] = 1$ for $0 \leq k \leq i - 1$, by (9). Hence $\text{SEND}^\bullet_{u+1}\ [1,\ i] = \varnothing$. $\square$

REFERENCES

1. J. Pachl, Analysis of toggle protocols, *Distributed Computing*, 1991, *5*, pp. 25-35.
2. M. Schneider, Self-stabilization, *ACM Computing Surveys*, 1993, *25*, pp. 45-67.
3. D. J. Wheeler, The livelock-free protocol of the Cambridge Ring, *The Computer Journal*, 1989, *32*, p. 95.