Short Note

Two Livelock-free Protocols for Slotted Rings

A previously proposed livelock-free modification of the empty-slot protocol for slotted rings is reviewed, and a new livelock-free modification is described.

Received September 1987

1. The empty slot protocol

A *slotted ring* is a synchronous unidirectional ring communication network in which a number of fixed-length slots are passed from station to station around the ring. A station that has data to send to another station waits for an empty slot to arrive and uses the slot to send the data.

A well-known example of a slotted ring is the Cambridge ring,[5] in which the slot has the following format:

F	M	DEST	SRCE	DATA	RESP

The fields F ('full–empty') and M ('monitor') have one bit each. The two values of F are called *empty* and *full*. The two values of M are called *new* and *old*. The DEST field is the destination station address, the SRCE field is the source station address, and the RESP field ('response bits') transports an acknowledgement from the destination to the source.

There is exactly one monitor station in the ring. The non-monitor stations execute the following protocol to send data. If there are data to transmit, the station repeats each slot as received until the received value of the F field is *empty*; then the station changes the value of the F field to *full*, and sends the value *new* in the M field, the destination address in the DEST field, its own address in the SRCE field and data in the DATA field; the value *destination absent* is sent in the RESP field.

The destination station recognises its address in the DEST field, and if it has a buffer available for reception makes a copy of the DATA field and sends the value *packet accepted* in the RESP field. However, except for the RESP field, the destination station repeats the slot as received. The sending station waits for the slot to return around the ring, then changes the F field back to *empty*, and repeats the rest of the slot as received. (The number of slots in the ring is known, and the station is therefore able to determine when the slot has returned.) The sending station also reads the RESP field, and thus finds out whether the data have been received by the destination.

The monitor station manipulates the contents of the F and M fields, and repeats the rest of the slot as received. (On a lower protocol level, the monitor station is also responsible for maintaining the circulating slots.) When the monitor station receives the values (*full*, *new*) in the (F, M) pair, it sends (*full*, *old*) instead; when the monitor station receives (*full*, *old*), it sends *empty* in the F field.

In Ref. 5 it is argued that ring reliability decreases as delays in each node increase. As is pointed out on p. 171 in Ref. 3, the Cambridge ring protocol may be implemented with minimum delay in non-monitor stations. Indeed, if a (non-monitor) station has data to send then the value of the F bit sent by the station in the next slot is *full*, regardless of whether the slot arrives empty or full. Thus the outgoing value of the F bit depends on its incoming value only when the station has no data to send, and in that case the F bit value is simply repeated by the station. However, in the monitor station the delay must be longer, because the value of the F bit sent by the monitor station depends on the value of the M bit received in the same slot.

Transmission errors affect the F field in two ways: *empty* value may be changed to *full*, and *full* value may be changed to *empty*. The monitor station protocol ensures that the ring recovers from the empty-to-full errors. However, in Ref. 2 it is shown that the protocol described above may lead to a livelocked state as a result of a single full-to-empty error. Once such a livelock develops, it persists as long as sufficiently many stations in the ring have new data to transmit. Therefore the livelock may persist indefinitely in certain situations, for example if a higher-level protocol causes unacknowledged data to be retransmitted.

2. Recovery by waiting

It is not difficult to devise livelock-free modifications of the protocol described in Section 1. In the protocol, the sender station reads the RESP field in the slot returning around the ring. Thus the station can also read the SRCE field and compare it with its own station address. If they differ, then the content of the slot may have been overwritten by another station, and it is possible that a livelock has developed. Therefore the station initiates a *recovery procedure*.

One such recovery procedure is proposed in Ref. 2. When the sender station detects an error, it becomes passive (i.e. it transmits the slot content as received) in $N-1$ roundtrips of the same slot, where N is the number of stations in the ring. (If the monitor station never sends its own data, then one could also use N to denote the number of non-monitor stations, and the recovery procedure would still be correct.) This procedure recovers from any livelock by ensuring that the slot eventually makes a complete round trip with the F field value unchanged.

However, one cannot replace the value $N-1$ by a smaller value in this procedure. Fig. 1 shows a scenario, in a ring with five non-monitor stations and a single circulating slot, in which every station remains passive in three slots after every active transmission, but the system remains livelocked for ever. In Fig. 1 thick lines denote the slot marked full, and thin lines the slot marked empty. Station 1 sends data, and a transmission error marks the slot empty between stations 3 and 4. Station 5 uses the slot to send data, and then stations send their own data in the order 4, 3, 2, 1, 5, 4, 3, ...

It may be argued that the scenario in Fig. 1 is very unlikely. However, the example shows that if the recovery procedure uses a value smaller than $N-1$ (or if the recovery procedure is omitted altogether) there is a non-zero probability of livelock. On the other hand, if the recovery procedure uses the value $N-1$ (or larger) a persistent livelock cannot arise.

3. Recovery by 'permission deletion'

A new recovery procedure will now be described. Its advantage is that it can be defined without any reference to the number of stations in the ring. In addition, as is noted at the end of the section, the procedure is compatible with the original protocol.

The recovery procedure is again initiated by a sender that discovers that the SRCE field

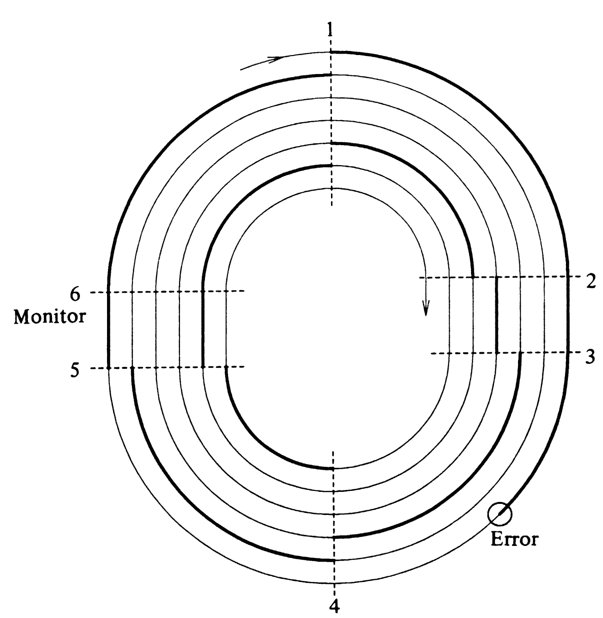

Figure 1. A livelock scenario.

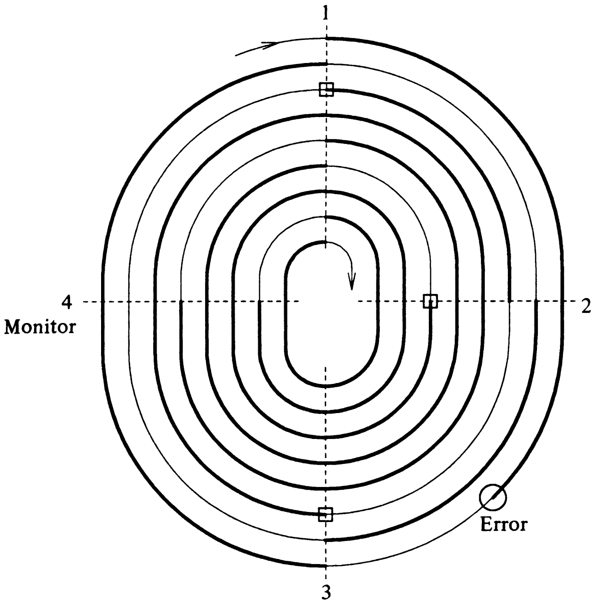

Figure 2. A scenario of recovery.

has been overwritten. When the sender detects an error, it waits until the slot in which the problem has occurred is received with the F field set to *empty*, changes it to *full*, sends *new* in the M field, an invalid address in the DEST field, and its own address in the SRCE field. (Thus the station waits until it is allowed to send its own data in the suspect slot, and sends dummy data, not addressed to any valid destination.) However, the station does *not* change the F field back to *empty* when the slot returns.

The appendix contains an algorithmic description of the protocol. One can argue, informally, that the protocol is livelock-free, as follows. Livelocks arise when there is more than one 'permission to transmit' in one slot during one round trip. Every time a station executes the recovery procedure, the number of those permissions is reduced by one. If all permissions are removed in this way, the monitor station protocol creates a new permission.

Fig. 2 shows an example of recovery in a ring with three non-monitor stations. The square box symbols in the figure denote the times when recovering stations send dummy data. As in Fig. 1, thick lines denote the slot marked full, and thin lines the slot marked empty.

This technique is conceptually similar to that used in the token ring defined in Ref. 1, where the state in which multiple tokens have been created through an error is converted to the state with no token. (Cf. Ref. 1, p. 76.)

It should be noted that only the data-sender protocol is modified. The other parts of the original protocol remain the same. Moreover, the proposed protocol is compatible with the original one in a very strong sense: one can mix stations executing the original protocol with those executing the modified protocol; as long as the stations executing the modified protocol send their data often enough, the system recovers from any number of full-to-empty errors.

4. Other recovery procedures

Another approach to recovery is taken by the designers of the MAN protocol.[4] Although the protocol has a different frame format, the following rule, adapted from Ref 4, may be added to the protocol in Section 1 to prevent livelocks. The sending station changes the F field value in the returning slot back to *empty* only if the M value in the returning slot is *old*. The disadvantage of this scheme is that, since the outgoing value of the F bit depends on the incoming value of the M bit, the delay in every station has to be at least one bit longer than the minimum delay necessary to retransmit the received bit values. However, the scheme is very simple, and is to be recommended when longer than minimum delays are acceptable.

The protocols in this paper are *centralised*, because they rely on one station (the monitor station) that is different from other stations in the ring. Zafiropulo and Rothauser[6] describe a *decentralised* protocol for slotted rings; several livelock-free modifications of the protocol are proposed in Ref. 2.

Acknowledgement

The author wishes to thank Liam Casey for his comments on the first version of the paper. The research was supported by the Natural Sciences and Engineering Research Council of Canada.

J. PACHL*

Department of Computer Science, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1

References

1. W. Bux, F. Closs, P. A. Janson, K. Kummerle, H. R. Muller and E. H. Rothauser, A local-area communication network based on a reliable token-ring system. *Proc. IFIP TC 6 International In-Depth Symp. on Local Computer Networks, Florence, Italy, April 19–21*, 1982,

* Now at: IBM Zürich Research Laboratory, Säumerstrasse 4, 8803 Rüschlikon, Switzerland.

```
constants
    OWN_ADDRESS     :address
    INVALID_ADDRESS:address
    FULL, EMPTY, NEW:bit

variable
    recovering:boolean

recovering:=FALSE;
repeat
  begin
    Await_next_slot()
    if "ready to send" and not recovering then
      begin
        if Send_bit(FULL)=EMPTY then                  {* F field *}
          begin
            Send_bit(NEW);                            {* M field *}
            Send_address("destination address");      {* DEST field *}
            Send_address(OWN_ADDRESS);                {* SRCE field *}
            "send data and response field"
            Await_next_slot();
            Send_bit(EMPTY);                          {* F field *}
            Send_bit(NEW);                            {* M field *}
            Receive_address();                        {* DEST field *}
            if Receive_address()≠OWN_ADDRESS then     {* SRCE field *}
                recovering:=TRUE;
          end
      end
    else if recovering then
      begin
        if Send_bit(FULL)=EMPTY then                  {* F field *}
          begin
            Send_bit(NEW);                            {* M FIELD *}
            Send_address(INVALID_ADDRESS);            {* DEST field *}
            Send_address(OWN_ADDRESS);                {* SRCE field *}
            Await_next_slot();
            recovering:=FALSE;
          end
      end
  end
```

Figure A.1.

edited P. C. Ravasio, G. Hopkins and N. Naffah, pp. 69–82.
2. J. Pachl and L. Casey, A robust ring transmission protocol. *Computer Networks and ISDN Systems* **13**, 313–321 (1987).
3. B. K. Penney and A. A. Baghdadi, Survey of computer communication loop networks. Part 1. *Computer Communications* **2**, 165–180 (1979).
4. D. T. W. Sze, A metropolitan area network. *IEEE J. Selected Areas in Communications* **SAC-3**, 815–824 (1985).
5. M. V. Wilkes and D. J. Wheeler, The Cambridge digital communication ring. *Proc. Local Area Networks Symp., Boston, May 1979*, edited N. B. Meisner and R. Rosenthal.
6. P. Zafiropulo and E. H. Rothauser, Signaling and frame structures in highly decentralized loop systems. *Proc. 1st Internat. Conf. Computer Communications, Washington D.C., October 1972.*

Appendix

This appendix contains a description of the modified sender protocol of Section 3 in an algorithmic notation. For the sake of simplicity it is assumed that there is only one slot in the ring (so that the sender need not skip any passing slots before receiving the transmitted slot after its trip around the ring).

The protocol is described by means of three low-level primitives (used also in Ref. 2):

```
bit_in:=Repeat_bit();
bit_in:=Send_bit(bit_out);
        Await_next_slot();
```

The primitive `Repeat_bit` copies the next bit from the input channel to the output channel, and returns its value. The primitive `Send_bit` receives the next bit from the input channel (`bit_in`) and sends the bit `bit_out` to the output channel. The primitive `Await_next_slot` copies the bits from the input channel to the output channel until a new slot begins.

The procedures

```
          Send_address
           (address_out);
address_in:=Receive_address();
```

which are easily written in terms `Send_bit` and `Repeat_bit`, are used to send and receive address fields.

The modified sender protocol appears in Fig. A.1.

Announcements

18–20 JULY 1989
Image processing and its applications,
University of Warwick, Coventry

The Third International Conference on 'Image processing and its applications' will be held at the University of Warwick, Coventry, UK from 18 to 20 July 1989.

Aims

The conference is the third in the successful international series on Image Processing organised by the Institution of Electrical Engineers and aims to provide a forum for the exchange of new results in the area of image processing. Papers of a review nature will also be presented.

Scope

The conference will include coverage of algorithms, systems and specialised hardware for image processing, and applications. For the purpose of the conference, images can be taken to include all types of multidimensional data, for example infrared, visual, radar and sonar. Topics to be covered include the following.

Theory

- Image fidelity measures and the visual characteristics of the observer
- Image modelling and simulation
- Image interpretation and recognition
- Knowledge-based image processing
- Multidimensional sampling, filtering and interpolation techniques
- Image coding techniques
- Enhancement and restoration of images
- Motion analysis and compensation techniques

Implementation

- Hardware/software for image processing
- Real-time acquisition, processing and display of image data
- Parallel and other novel architectures for image processing, for example parallel processors, neural and connectionist systems
- VLSI for image processing
- Image storage and retrieval

Applications

- Broadcast television, including enhanced and high-definition systems
- Low-bandwidth video transmission, for example teleconference and surveillance systems
- Bit-rate reduction for digital recording of images
- Facsimile, text and printing applications
- Processing for special-effects generation
- Remote sensing
- Medical image processing
- Industrial inspection and non-destructive testing
- Robotic and computer vision
- Image processing for the man–machine interfaces

This list is not intended to be exclusive, and any topics of relevance to Image Processing will be considered.

Contributions

The Organising Committee invites offers of contributions for consideration for the programme.

Those intending to offer a contribution either for a poster or formal session should submit a synopsis of a maximum of two A4 pages to the IEE Conference Services Department on or before **1 November 1988.** The synopsis should include the main points of the paper and, where possible, indicate where the emphasis will be placed.

Authors of selected synopsis will be requested to provide full typescripts on not more than five double-column A3 camera-ready sheets (approx 4000 words), including illustrations, for assessment by **17 March 1989.**

Poster session

It is intended to organise a poster session in addition to formal conference sessions. Authors who wish to submit a paper for consideration for inclusion in a poster session should so indicate on the submission.

Proceedings papers

It is intended to produce a special issue of *IEE Proceedings* (Part F) containing papers presented at the conference. Papers of a suitably high standard will be forwarded to the Editor for consideration for inclusion in this special issue.

Deadlines

Intending authors should note the following deadline dates:
Receipt of synopsis, 1 November 1988
Notification of provisional acceptance of synopsis, January 1989
Receipt of full text for final review, 17 March 1989

These deadlines will be strictly adhered to.

Venue

The conference will be held at the University of Warwick, UK

Working language

The working language of the conference is English, which will be used in all printed material, presentations and discussion. Simultaneous interpretation will not be provided.

Exhibition

It is proposed to arrange an exhibition in association with the conference. Those requiring further details of layout and charges should so indicate.

Programme and registration

Registration forms and programme details will be published a few months before the event.

For further information contact:
Conference Services, The Institution of Electrical Engineers, Savoy Place, London WC2R 0BL.