Information Processing Letters 25 (1987) 349
North-Holland

10 July 1987

Letter to the Editor

COMMENTS ON "A DISTRIBUTED ALGORITHM FOR DISTRIBUTED TERMINATION" [1]

Dear Sir,

In a recent paper, Hazari and Zedan [1] have presented a distributed termination detection algorithm. We like to inform you that the algorithm is not correct under the usual assumptions of the problem: it can lead to the detection of 'false termination' (i.e., some processes conclude termination while there is not) as well as to undetected termination. The proofs of Assertions A and B in Section 3, stating that undetected and false termination do not occur, are not correct because it is not taken into account that passive processes may be reactivated by communications from processes that are still active.

For a concrete counterexample we consider a (unidirectional) ring consisting of four processes P_1, P_2, P_3, and P_4. We describe a scenario in which termination goes undetected. Assume $B_4 = \mathit{false}$, $B_1 = B_2 = B_3 = \mathit{true}$. p_1, p_2, and p_3 send out a probe message, which is purged by p_4. Then, P_4 activates P_1 and becomes passive. p_4 sends out a probe message which is purged by p_1 and, finally, p_1 becomes passive. Now, $\mathit{first\_time}_i = \mathit{false}$ for all i, hence no p_i will ever initiate a probe again and termination remains undetected. Next, we describe a scenario in which termination is detected falsely. Again, assume $B_4 = \mathit{false}$ and $B_1 = B_2 = B_3 = \mathit{true}$. p_1 sends 1 to p_2. p_2 sends 2 to p_3. P_4 activates P_1 and becomes passive. P_1 activates P_2 and becomes passive. p_3 sends 3 to p_4. p_4 will not purge the message for B_4 is now *true*. p_4 sends 4 to p_1. p_1 concludes termination, while p_2 is still active. Similar counterexamples can be constructed for virtually any network of communicating sequential processes.

Contrary to what the authors suggest on p. 293, most termination detection algorithms for 'rings' of processes allow basic communications over arbitrary links and only restrict the detection messages to a (logical) ring. We like to point out that there exists a large number of distributed termination detection algorithms, including symmetric ones, with differing merits. An inventory of recent literature and some new algorithms were given in [2]. A large class of algorithms, covering a wide range of network topologies and communication properties, is given in [3].

G. TEL and J. VAN LEEUWEN
Department of Computer Science
University of Utrecht
P.O. Box 80.012
3508 TA Utrecht
The Netherlands

References

[1] C. Hazari and H. Zedan, A distributed algorithm for distributed termination, Inform. Process. Lett. 24 (5) (1987) 293–297.

[2] R.B. Tan and J. Van Leeuwen, General symmetric distributed termination detection algorithms, Tech. Rept. RUU-CS-86-2, Dept. of Computer Science, Univ. of Utrecht, 1986; also: Computers and Artificial Intelligence, submitted.

[3] G. Tel, Distributed infimum approximation, Tech. Rept. RUU-CS-86-12, Dept. of Computer Science, Univ. of Utrecht, 1986.

Communicated by W.L. Van der Poel
Received 8 April 1987