

Dijkstra-Scholten Termination Detection Algorithm

The Termination Detection algorithm consists of the following steps:

Definitions for a process P_i :

C, D : integer

m: (basic, ack) messages {Represents the type of message received}

state: (active, passive)	{Represents the state of a process}
--------------------------	-------------------------------------

parent(i): process {The parent variable at P_i is another process }

Initializations:

$C = 0, D = 0, \text{parent}(i) = i$ {Parent of each process is itself}

1. $(m = \text{basic}) \wedge (C = 0) \rightarrow$ $C := 1; \text{state} := \text{active};$
 $\text{parent}(i) := \text{sender}$
2. $(\text{send basic message}) \rightarrow D := D + 1;$
3. $(\text{On becoming passive}) \rightarrow (\text{state} = \text{passive})$
4. $(m = \text{ack}) \rightarrow D := D - 1;$
5. $(C = 1 \wedge D = 0) \wedge (\text{state} = \text{passive}) \rightarrow$ $\text{send ack to parent};$
 $C := 0; \text{parent}(i) = i$
6. $(m = \text{basic}) \wedge (C = 1) \wedge (\text{state} = \text{passive}) \rightarrow$ $\text{send ack to the sender};$
 $\text{state} := \text{active};$
7. $(m = \text{basic}) \wedge (C = 1) \wedge (\text{state} = \text{active}) \rightarrow$ $\text{send ack to the sender};$