

# A Distributed Solution of the Distributed Termination Problem

S.P. Rana

Computer Centre, Indian Institute of Technology

Information Processing Letters 17 (1983) 43-46

# Contenido

- Resumen
- Introducción
- Características del Algoritmo
- Algoritmo de Rana
- Conclusiones

El algoritmo de Rana es un algoritmo de detección de terminación para un sistema distribuido y, es completamente simétrico.

Esta solución difiere de otros métodos en el aspecto fundamental de que cualquier proceso(nodo) “n” puede iniciar una “onda de detección” y posiblemente tenga éxito al detectar la condición de terminación global, estas ondas de detección transportan un valor de reloj proporcionado por un reloj lógico (Lamport, L. (1978))

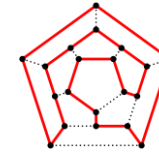
Funciona de la siguiente

# Introducción



Satyendra Rana  
Austin, Texas, United States  
Chief Technology Officer  
diwo

# Características del Algoritmo



- Se supone que los procesos están conectados por un ciclo Hamiltoniano.
- Cuando un proceso cumple su predicado local de terminación anota la hora y se envía un mensaje de detección al proceso sucesor con un timestamp igual a la hora en la que se cumplió el predicado y con un contador igual a 1.
- Si un nodo activo recibe un mensaje de un nodo predecesor simplemente lo purga.
- Si un nodo pasivo recibe un mensaje de detección, compara la marca de tiempo del mensaje con el tiempo señalado cuando la última vez que satisfizo su predicado local. Si el último tiempo es mayor que el sello de tiempo del mensaje, el mensaje se purga, de lo contrario, el contador del mensaje se incrementa por 1 y se reenvía al sucesor en el ciclo.
- Si un nodo pasivo recibe un mensaje de detección con un valor de contador igual a  $n$ , el nodo envía un mensaje de terminación al sucesor y termina.

# Características del Algoritmo

$P :: [P_1 || \dots || P_n]$ , para cada  $i \in [1, n]$ ,  $P_i :: * [S_i]$

$B_i$  = Predicado por satisfacer de  $P_i$

$BTIME_i$  = Tiempo en el que  $P_i$  satisface su predicado  $B_i$

$(TIME, COUNT) = Time - stamp$

# Algoritmo de Rana

$P_i :: B_i := false;$   
     $* [S_i$   
         $\square B_i \rightarrow BTIME := CLOCK - TIME;$   
             $P_{i+1}! detection - message(BTIME_i, 1)$   
         $\square P_{i-1}? detection - message(TIME, COUNT) \rightarrow$   
             $[\neg B_i; \text{purge the message}$   
                 $\square B_i; TIME < BTIME_i \rightarrow \text{purge the message}$   
                 $\square B_i; TIME \geq BTIME_i; COUNT \neq n \rightarrow$   
                     $P_{i+1}! detection - message(TIME, COUNT + 1)$   
                 $\square B_i; TIME \geq BTIME_i; COUNT = n \rightarrow$   
                     $P_{i+1}! terminate - message; TERMINATE]$   
     $\square P_{i-1}? termination - message \rightarrow$   
         $P_{i+1}! termination - message;$   
         $TERMINATE$

**Aserción 1.** Si se cumple la condición de terminación global, eventualmente se detectará la terminación.

**Aserción 2.** No hay posibilidad de detectar una falsa terminación.



# Conclusiones

El algoritmo de Rana detecta la terminación de un algoritmo básico descentralizado. Implementa en los mensajes de detección un valor de reloj proporcionado por un reloj lógico.

De esta manera se reconoce cada mensaje, de modo que un proceso puede determinar si todos los mensajes básicos que envió han llegado a su destino.