Distributed Deadlock Detection Algorithms

3. A Diffusion Computation Based Algorithm

- It is diffused through the WFG of the system

- For OR request model.

- Process determines if it's deadlocked by initiating a diffusion computation.

- messages used is in the form of

  → query (i,j,k)

  ↘ reply (i,j,k)

→ It denotes that it is initiated by a process P_i & are being sent from Process P_j to process P_k.

→ Process states

  → Active = process is executing

  ↘ Blocked = process is waiting to acquire a resource.

→ Dependent set of the processes : A blocked process initiates deadlock detection by sending query messages to all the processes from whom it is waiting to receive a message.

What does an active process do when receives a query or reply message? It discards it (Because it is already communicating with other)

When blocked process P_k receives query (i, j, k) message, it takes following actions :

a) If this is I query message received by P_k for the deadlock detection initiated by P_i (known as engaging query), then it propagates the query to all the processes in its dependent set & sets a local variable $num_k(i)$ to the number of query messages sent

b) If not an engaging query (not first), then P_k returns a reply message to it immediately, provided P_k has been continously blocked since it received the corresponding engaging query. Otherwise, it discards query.

$wait_k(i)$ = local boolean variable
→ at process P_k denotes the fact that it has been continously blocked since it received the last engaging query from Process P_i.
→ When a blocked process P_k receives a reply (i, j, k) message, it decrements $num_k(i)$ provided $wait_k(i)$ holds.
→ A process sends a reply message in response to an engaging query only after it has received a reply to every query message it sent out for this engaging query i.e $num(i) = 0$ at it.

Chandy et al.'s diffusion computation Based deadlock Algorithm:-

Pseudocode for OR request model:-

Initiate a diffusion computation for a blocked process P_i:

send query (i, j, k) (i, i, j) to all processes P_j in the dependent set DS_i of P_i;

$num_i(i) := |DS_i|$; $wait_i(i) := true$;

When a blocked process P_k receives a query (i, j, k):

if this is the engaging query for process P_k.

then send query (i, k, m) to all P_m in its dependent set DS_k;

$num_k(i) := |DS_k|$; $wait_k(i) := true$

else if $wait_k(i)$ then send a reply (i, j, k) to P_j.

When a process P_k receives a reply (i, j, k)

if $wait_k(i)$
   then begin
      $num_k(i) := num_k(i) - 1$;
      if $num_k(i) = 0$
      then if $i = k$ then declare a deadlock
         else send (i, k, m) to the process P_m,
         which sent the engaging query.

In above, it is assumed that only one diffusion computation is initiated for a process.

Distributed Deadlock Detection Algorithm.

4. A Global State Detection Based Algorithm.

There are 3 algorithms in this part:

Facts for the Algorithm.

- A consistent snapshot of a distributed system can be obtained without freezing the underlying computation.

- A consistent snapshot may not represent the system state at any moment in time, but if a stable property holds in the system before the snapshot collection is initiated, this property will still hold in the snapshot.

A). Bracha & Toueg

→ 2 phases

→ I phase - algo records a snapshot of distributed WFG

II phase - simulates the granting of requests to check for generalized check deadlocks.

II phase is nested within I phase, so it terminates only when II phase is over

B) Wang et. al also has 2 phase

I phase : records a snapshot of the distributed WFG.

II phase : static WFG recorded in the phase I is reduced to detect any deadlocks.

Both phases occur serially.

c) KshemKalyani – Sunghal Algorithm.
   - has single phase
   - 2 types of messages

Fan-out sweep   Fan-in sweep
outwards from an   inwards to the
initiator process   initiator process.

Sweep — of a WFG is a traversal of the WFG
in which all messages are sent in the
direction of the WFG edges (an outward sweep)
or all messages sent against the direction
of the WFG edges (an inward sweep).

   - Both sweeps are done concurrently.

In outward sweep - algo records a snapshot of
   a distributed WFG
In inward sweep - recorded distributed WFG
   is reduced to determine whether the initiator
   is deadlocked.

Limitation — Reduction of the WFG at a process
   can begin before all the WFG edges
   incident at that process have been recorded.