8-29-2019

Formal description of HS algorithm for i^{th} process.

initialization
$u_i \leftarrow$ unique id of process i
Send+ $\leftarrow \{u_i, \text{out}, 1\}$;
Send− $\leftarrow \{u_i, \text{out}, 1\}$
Status $\leftarrow$ unknown;
phase $\leftarrow 0$

Message Sending
Send Send+ to process $i+1$
Send Send− to process $i-1$

// message processing
Send+ $\leftarrow$ null; Send − $\leftarrow$ null;
if message from $(i-1)$ is (v, out, h) {
Case
$(v > u_i$ & $h > 1)$: Send+ $\leftarrow (v, \text{out}, h-1)$.
$(v > u_i$ & $h = 1)$: Send − $\leftarrow (v, \text{in}, -)$
$v = u_i$: Status = leader
end case

if message from $(i+1)$ is (v, out, h) {
Case
$(v > u_i$ & $h > 1)$: Send − $\leftarrow (v, \text{out}, h-1)$.
$(v > u_i$ & $h = 1)$: Send + $\leftarrow (v, \text{in}, -)$.
$v = u_i$: Status = leader
end case

②

if message from $(i-1)$ is $(v, in, -)$ & $v > u_i$
    Send + $\leftarrow (v, in, -)$
if message from $(i+1)$ is $(v, in, -)$ & $v > u_i$
    Send − $\leftarrow (v, in, -)$
if message from $(i-1)$ and $(i+1)$ are both
    $(u_i, in, -1)$ {
    Phase++;
    Send + $\leftarrow (u_i, out, 2^{phase})$;
    Send − $\leftarrow (u_i, out, 2^{phase})$;
}

Message Complexity

phase 0: $\leq 4n$

phase ℓ ; $\ell > 0$:

# of processes initiating tokens in phase ℓ = # of processes who got both tokens back in phase $\ell - 1$

$$= \frac{n}{1 + 2^{\ell-1}}$$

Worst case # of messages for each process initiating a token in phase ℓ:

$2^\ell \cdot 2 \cdot 2$ $\qquad$ 2^ℓ hops $\qquad$ 2^ℓ hops

$= 4 \cdot 2^\ell$ messages

(3)

# of messages at phase l:

$$4 \cdot 2^{l} \cdot \frac{n}{1 + 2^{l-1}}$$

$\leq 8n$ for phase l.

Total # of messages $\leq 4n + 8n \cdot \lceil \log_2 n \rceil$

$= O(n \log n)$

$O(n^2) \longrightarrow O(n \log n) \longrightarrow$

LCR algorithm

HS algorithm

Yes we can : $O(n)$ messages if ids can be viewed as numbers

No if only comparisons of ids are allowed

id is an integer (unsigned)

(a) n, the total number of processes is known.

u_{min} is smallest id:

waits for $(u_{min} - 1) * n$ rounds

(4) time : $u_{min} \cdot (n)$

message: $O(n)$ $(= n)$

(2) n unknown?

Variable speeds algorithm; same start time

Process i has id u_i

" " starts a token with id u_i

this message containing id u_i travels at the rate of 1 hop for every 2^{u_i} rounds.

Process j receives a message containing id ~~k~~. u_k:

if $(u_k > u_j)$ discard message
else {
 delay u_k for $(2^{u_k} - 1)$ rounds
 and then forward
}

(5) time : $(2^{u_{min}} \cdot n)$ rounds for u_{min} to gets its token back

+ n rounds for all to know id of leader & terminate.

$\therefore O(n \cdot 2^{u_{min}})$

messages: winner: ~~of n~~ n messages

\# of links traversed by 2^{nd} smallest id $\leq \frac{n}{2}$

3^{rd} smallest ...

$\leq \frac{n}{4}$

$\vdots$

$\leq n$

~~tot~~ total # of messages $O(n)$

Read : Variable speeds & Variable Start time

⑥ NO: Cannot improve $O(n \log n)$ messages of HS algorithm if only Comparisons on ids are allowed.

Lower bound of $\Omega(n \log n)$ on any Comparison based distributed leader election in synchronous rings.

Definitions

1. order equivalence

$\underline{U} = (u_1, u_2 \ldots u_k)$ seq. of k ids

&

$\underline{V} = (v_1, v_2 \ldots v_k)$ seq of k ids.

$\underline{U}$ and $\underline{V}$ are order equivalent

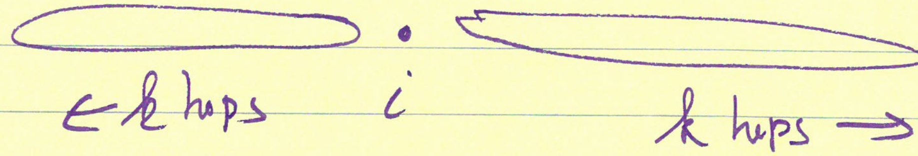
$\Leftrightarrow \forall i \,\&\, j$, $i \le k$, $j \le k$, $u_i \le u_j \Leftrightarrow v_i \le v_j$

(1, 8, 3, 5) & (10, 41, 23, 30)

are order equivalent

7

k neighborhood of a process i



is the sequence of all ids that are k hops or less from process i

State Correspondence

Two process states s and t **correspond** with respect to sequences $\overline{U} = (u_1, u_2 \ldots u_k)$ and $\underline{V} = (v_1, v_2 \ldots v_k)$ of ~~unique ids~~ process ids if

(a) ids of s are chosen from $\overline{U}$

(b) ids of t are chosen from $\underline{V}$

(c) ~~...~~ $t \equiv s$ except that each u_i in s is replaced by v_i in t.