Clock Synchronization

- Reading a Remote Clock
  - Paper by Flaviu Cristian (Cri89a)
  - "Probabilistic Clock Synchronization"
  - Method for reading remote clocks
  - Systems assumed to have random unbounded communication delays.
  - Approach does not guarantee that a processor can always read a remote clock.
  - A process can read the clock of another process with a given precision with a probability as close to 1 as desired.
  - After reading clock, the actual reading precision is known.

Clock Synchronization

- The Problem of Reading Remote Times
  - Process *P* sends message ("time = ?") to process *Q*. Process *Q* replies with message ("time = *T*")
  - When *P* receives the message ("time = *T*"), what time is it in *Q*' s clock?
    - i.e. what is the time displayed by *Q*' s clock?
    - one can only try to derive an interval.
  - Definitions:
    - t = the real-time when P receives the message from *Q*.
    - $C_Q(t)$ = value displayed by *Q*' s clock at real-time t
    - *min* = minimal delay to send message from *P* to *Q* or vice versa.
    - D = half the roundtrip delay measured by *P*.
  - Note: small letters/symbols indicate real times

Clock Synchronization

- Let $(\min + \alpha), (\min + \beta),\ \alpha, \beta \geq 0$ be the real time delays for sending and returning a message.
  - message path: $P \rightarrow Q \rightarrow P$
  - min + α accounts for message time from *P* to *Q*
  - min + β accounts for message time from *Q* to *P*
- Let 2*d* be the real time roundtrip delay, then

$$2d = 2\min + \alpha + \beta$$

- We are interested in β, since this is the time that has passed since *Q* wrote its time in the message to *P*. How big is β?

Clock Synchronization

- Using $2d = 2\min + \alpha + \beta$ we get

$$0 \leq \beta \leq 2d - 2\min$$

since α could be 0.

- *Q*' s clock can run at any speed in [1 - ρ, 1 + ρ], where ρ is again the clock drift rate.
- Thus

$$C_Q(t) \in [T + (\min + \beta)(1-\rho), T + (\min + \beta)(1+\rho)]$$

substituting β from above

$$C_Q(t) \in [T + (\min)(1-\rho), T + (\min + 2d - 2\min)(1+\rho)]$$

Clock Synchronization

- Now, relate

  $C_Q(t) \in [T + (\min)(1-\rho), T + (2d - \min)(1+\rho)]$

  to the time measured in *P*.
- Since *P*' s clock could have maximum drift we must assume

$$d \leq D(1+\rho)$$

$$\begin{aligned} C_Q(t) &\in [T + \min(1-\rho), T + (2D(1+\rho) - \min)(1+\rho)] \\ &= [T + \min(1-\rho), T + 2D(1+\rho)^2 - \min(1+\rho)] \\ &\approx [T + \min(1-\rho), T + 2D(1+2\rho) - \min(1+\rho)] \end{aligned}$$

ρ^2 is ignored

- This is the smallest interval possible

Clock Synchronization

- Processor *P* cannot determine where in the interval *Q*' s clock value is.
- Suggestion:

  Estimate C_Q by function $C_Q^P(T, D)$
- The actual error is

$$|C_Q^P(T, D) - C_Q(t)|$$

- What is a good choice for $C_Q^P(T, D)$

  best choice for function is to choose midpoint

Clock Synchronization

- Midpoint

$$\tfrac{1}{2}(T + \min(1-\rho) + T + 2D(1+2\rho) - \min(1+\rho)) =$$
$$\tfrac{1}{2}(2T + \min(1-\rho-1-\rho) + 2D(1+2\rho)) =$$
$$T - \rho\min + D(1+2\rho)$$

- Thus $C_Q^p(T,D) \equiv T - \rho\min + D(1+2\rho)$
  - this is "P's reading of Q's clock"
- The maximal error this can cause is half the largest possible interval.

$$[T + \min(1-\rho), T + 2D(1+2\rho) - \min(1+\rho)]$$

Clock Synchronization

$$\tfrac{1}{2}[T + 2D(1+2\rho) - \min(1+\rho) - (T + \min(1-\rho))] =$$
$$\tfrac{1}{2}[T + 2D(1+2\rho) - \min(1+\rho) - T - \min(1-\rho)] =$$
$$\tfrac{1}{2}[2D(1+2\rho) - \min(1+\rho) - \min(1-\rho)] =$$
$$\tfrac{1}{2}[2D(1+2\rho) - 2\min] =$$
$$D(1+2\rho) - \min$$

- Thus largest possible error is:

$$e = D(1+2\rho) - \min$$

- Any other estimate choice leads to a bigger maximum error.