SINCRONIZAÇÃO NOS SISTEMAS DISTRIBUÍDOS

PROF. DR. SÉRGIO CARLOS PORTARI JÚNIOR

PROBLEMA DA SINCRONIZAÇÃO EM SD

- Como os processos cooperam e sincronizam uns com os outros em um SD?
- Semáforos em sistemas individuais com uma única CPU.

- Estes métodos não funcionarão em sistemas distribuídos. Por quê?
- Porque dependem implicitamente da existência de memória compartilhada.

PROBLEMA DA SINCRONIZAÇÃO EM SD

- Dois processos que interagem usando um semáforo.
- Devem ser capazes de acessar o semáforo.
- Se ocorrerem dois eventos em um SD.
- Como decidir sobre a ordem relativa dos eventos.
- Um evento precede outro evento?

- Difícil determinar se os eventos ocorrem em máquinas diferentes.

PROBLEMA DA SINCRONIZAÇÃO EM SD

- Cada processador (nó de um sistema distribuído) tem um componente chamada relógio.
- Relógios em um SD podem:
  - Acusar horas diferentes entre si (defasagem interna).
  - Acusar hora diferente da hora real externa (defasagem externa).
  - Ter velocidades diferentes (defasagem variável).

PROBLEMA DA SINCRONIZAÇÃO EM SD

Relógios sincronizados são necessários para uma série de aplicações:

- Identificar atualidade de mensagens (antigas devem ser descartadas).
- Aplicações de tempo real.
- Controle de versões.

PARTE I – OS RELÓGIOS

- Como sincronizar eventos com base no tempo real?
- Sincronização do relógio.
- Como determinar a ordem relativa?
- Relógios lógicos.

PARTE II – ESTADO E ELEIÇÃO GLOBAL

- O que é o "estado global" de um sistema distribuído?
- Como determinamos o "coordenador" de um sistema distribuído?

PARTE III – COMO SINCRONIZAR PARA COMPARTILHAR

- Exclusão mútua
- Transações distribuídas

SINCRONIZAÇÃO DE RELÓGIO

Em um sistema distribuído:

Alcançar acordo no tempo não é trivial!

Figura 6.1 Quando cada máquina tem seu próprio relógio, um evento que ocorreu após outro evento pode, ainda assim, receber um horário anterior.

SINCRONIZAÇÃO DE RELÓGIO

- A sincronização do relógio não precisa ser absoluta! [Lamport, 1978]:
- Se dois processos não interagem, seus relógios não precisam ser sincronizados.
- O que importa não é que todos os processos concordem exatamente em que horas são.

- Mas concordam com a ordem em que os eventos ocorrem.

SINCRONIZAÇÃO DE RELÓGIO

Relógios Lógicos

- Necessidade apenas da consistência interna dos relógios?
- Prover identificar a ordem dos eventos.
- Não se os relógios estão próximos do tempo real.

Relógios Físicos

- Para cenários onde os relógios devem ser apenas iguais.
- E não devem se desviar do tempo real.

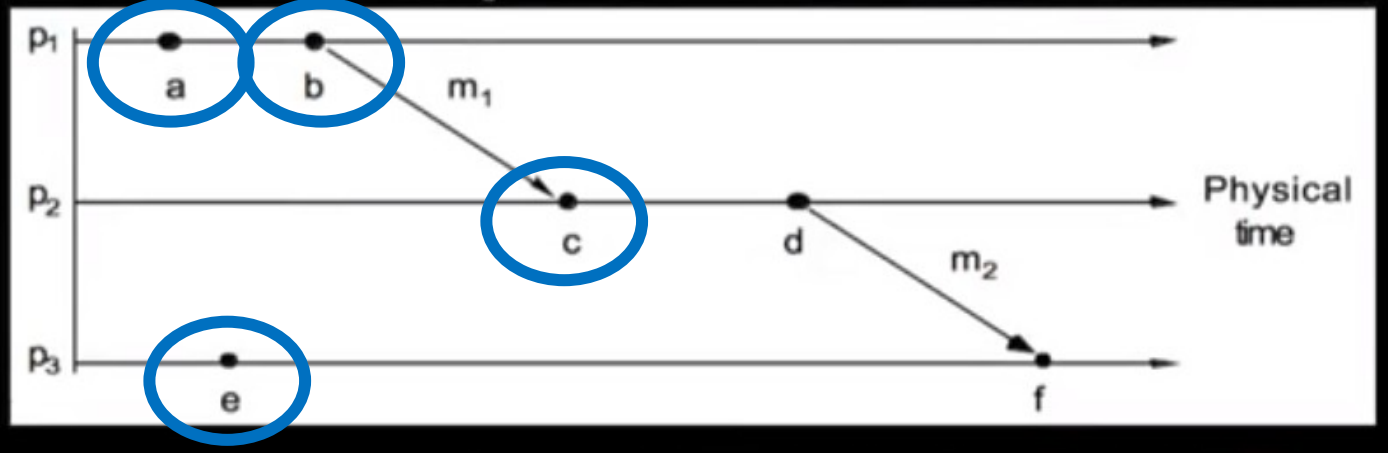
RELÓGIOS LÓGICOS DE LAMPORT

$a \rightarrow b\ (p1)\ c \rightarrow d\ (p2)$

$b \rightarrow c$ dado $m1$

$d \rightarrow f$ dado $m2$

- Nem todos os eventos podem ser relacionados através da relação "acontece antes"
- Consideremos ***a*** e ***e*** (processos sem a existência de mensagens entre os processos)
- São definidos como processos concorrentes; ***a*** || ***e***



RELÓGIOS LÓGICOS DE LAMPORT

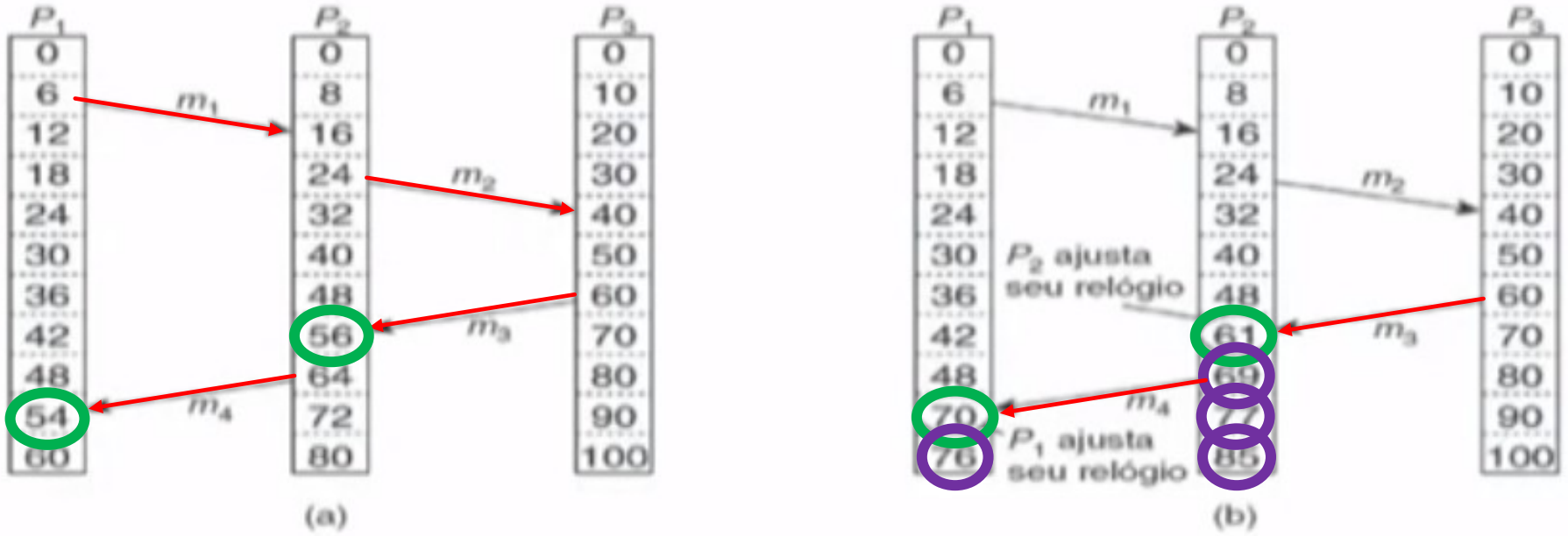


Figura 6.9 (a) Três processos, cada um com seu próprio relógio. Os relógios funcionam a taxas diferentes. (b) O algoritmo de Lamport corrige os relógios.

RELÓGIOS LÓGICOS DE LAMPORT

- Eventos são ajustados usando relógios lógicos
- Em um mesmo processo, relação “acontece → antes” é garantida para eventos que ocorrem no mesmo processo e eventos que se relacionam através de mensagens trocadas.

RELÓGIOS FÍSICOS

- GMT: Greenwich Mean Time
- BIH: Bureau Internacional de l’ Heure
- TAI: International Atomic Time
- UTC: Universal Coordinated Time
- NIST: National Institute of Standard Time
- WWV: estação de rádio de ondas curtas
- GEOS: Geostationary Environment Operational Satellite

RELÓGIOS FÍSICOS – ALGORITMO DE BERKELEY

- Não dispõe de uma máquina com receptor de Tempo Universal Coordenado.
- O Servidor de Tempo requer periodicamente de cada máquina, o tempo do seu relógio.
- O Servidor de Tempo calcula a média e diz para cada máquina como ajustar seu relógio.

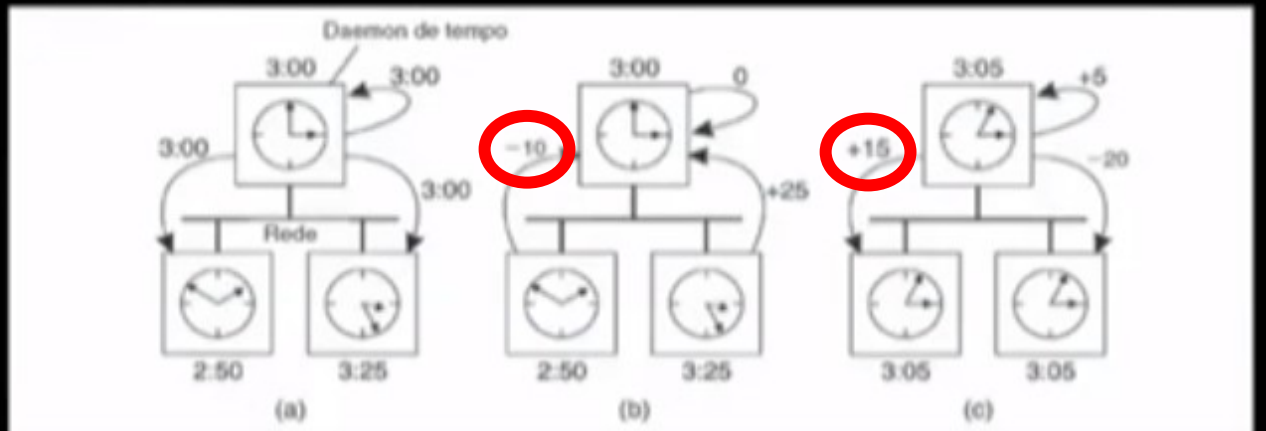


Figura 6.7 (a) O daemon de tempo pergunta a todas as outras máquinas os valores marcados por seus relógios. (b) As máquinas respondem. (c) O daemon de tempo informa a todas como devem ajustar seus relógios.

RELÓGIOS FÍSICOS – ALGORITMO DE BERKELEY

- Precisão:
  - depende do *round trip time* (latência ou delay de rede)
- Tolerância a falhas:
  - Calcula a média dos tempos para um subconjunto de computadores que diferem a até um certo valor máximo.
  - Ignora mensagens cujo tempo de transmissão é demasiado elevado.
- Que fazer se o master falha?
  - Eleger um novo coordenador.

RELÓGIOS FÍSICOS - PROBLEMAS

- O tempo do relógio físico não pode voltar (atrasar) – Solução é diminuir a frequência do clock.

- Em uma rede, é difícil medir com precisão o delay da rede.
- Atraso na rede mal calculado para computar o tempo certo no ajuste dos relógios.
- Algoritmos como o de Berkeley tentam atenuar este problema.