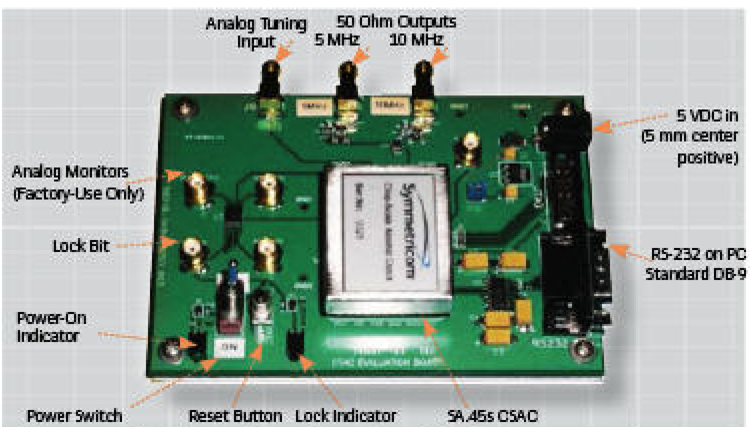
Lecture 19

- Administration
- Midterm this coming MONDAY

Chip Size Atomic Clocks?



All the surrounding electronics are also engineered for low power consumption.

Atomic clock accuracy and low power make the SA.45s ideal for portable applications requiring precise synchronization and time keeping, especially in GPS-denied environments.

Breakthrough Leadership

The CSAC's unmatched portability derives from specs that include:

< 120mW power consumption

< 17cm^3 volume

35g weight

±5.0E-11 accuracy at shipment

σy < 5 x 10-12 at τ = 1 hour short-term stability (Allan Deviation)

< 3.0E-10/month aging rate

$US1500 chip

For those who like to be on-time?

Bathys Hawaii founder Dr. John Patterson designed the Cesium 133 with cooperation from engineer George Talbot. The duo used the same principle of counting hyperfine lines of excited cesium 133 atoms used by the National Institute of Standards and Technology for its atomic clock. What that technology allows is incredible accuracy on a wristwatch in the area of 1 second per 1000 years.

Inside the rectangular metal enclosure of the Cesium 133 watch is a laser, a heater, a sealed cavity of cesium gas, a microwave filter, and a photodiode detector. The watch is large and gets power from standard lithium-ion rechargeable batteries. All the technology crammed inside the watch that is required for the incredible accuracy means that the watch can only run **for a few hours per charge**. The company plans a future version that is smaller and has better battery life. The current Cesium 133 watch has a retail price of **$12,000**

Chip Size Atomic Clocks?

For "quick win" applications such as navigating in areas of intermittent GNSS coverage CSAC facilitates fly wheel (holdover) operation thereby allowing local

For other high impact applications such as for communications networking CSAC provides an ability to rapidly form ad hoc networks and to pass data more reliably and securely via precision time stamping, data authentication and temporal windows.

In high frequency trading (where transactions may be completed in microseconds) the network must be synchronised at all nodes (e.g. every switch, router, server, processor, GPS receiver etc.) with an accuracy of the order hundred nanoseconds otherwise timestamps will be inaccurate and it may appear that trades have occurred before they have actually been executed. Such networks are typically synchronised to GPS time and are hence susceptible to GPS outage.

Other non GPS based networks use a precision time protocol with accuracy of the order 10 microseconds. Higher transmission accuracies derived from precision shared time references such as CSAC can potentially allow messages to be precisely retrieved from data streams without the need for headers and footers; messages can therefore be transmitted with greater speed and security.

Cristian's Clock Algorithm

Setting

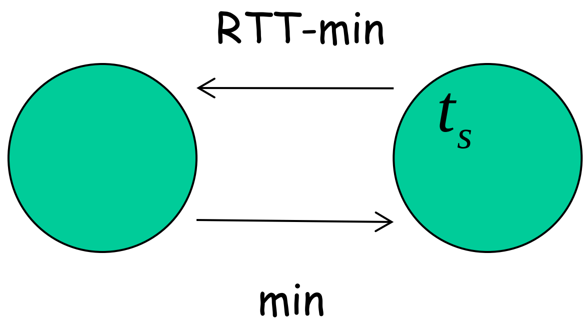
$$c(t) = t_s + \mathrm{RTT}/2$$

Accuracy

$$t_s \pm \left(\frac{\mathrm{RTT}}{2} - \min \right)$$

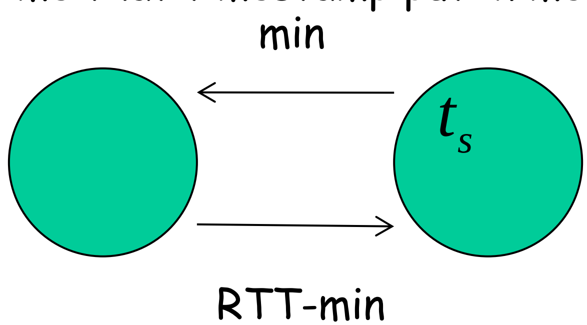
Cristian's Clock Algorithm

Earliest time that timestamp put in message



$$c(t) = (t_s + \min) + \mathrm{RTT}/2$$

Latest time that timestamp put in message



$$c(t) = (t_s + \mathrm{RTT} - \min) + \mathrm{RTT}/2$$

In both cases the

$$t_s \pm \left(\frac{\mathrm{RTT}}{2} - \min \right)$$

Question

A client attempts to synchronize with a time server using Cristian's method. It records the round-trip time and timestamps returned by the server in the table below.

RTT (milliseconds)	Time (hour:min:sec)
26	01:23:45.687
20	01:23:47.890
23	01:23:50.425

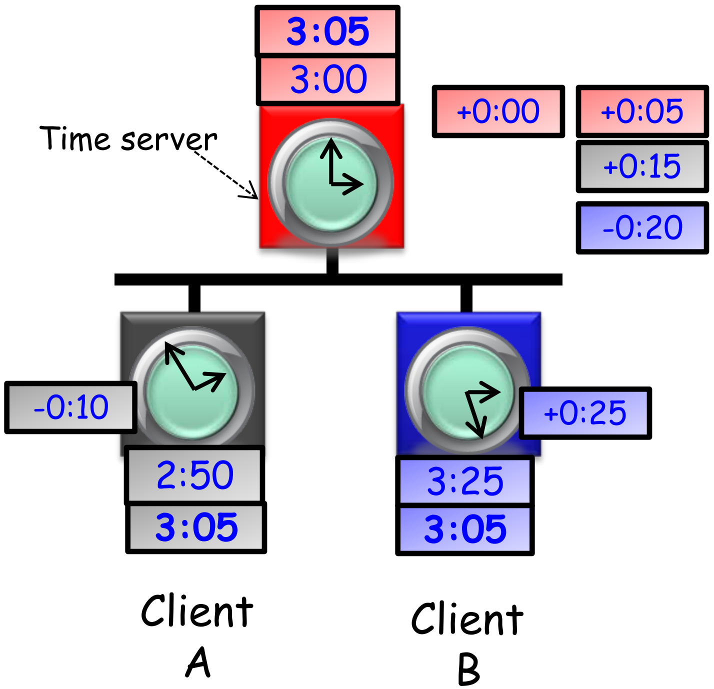
(a) Which of these times should you use to set the clock to obtain the best accuracy (minimize skew)

(b) What should the time be set to?

Berkeley Algorithm (internal)

Berkeley Algorithm is a distributed approach for time synchronization that does not require a UCT server.

- Approach:
  1. A time server periodically (approx. once in 4 minutes) sends its time to all the computers and polls them for the time difference
  2. The computers compute the time difference and then reply
  3. The server computes an average time difference for each computer
  4. The server commands all the computers to update their time (by gradual time synchronization) . Sends OFFSET!!



Berkeley Algorithm

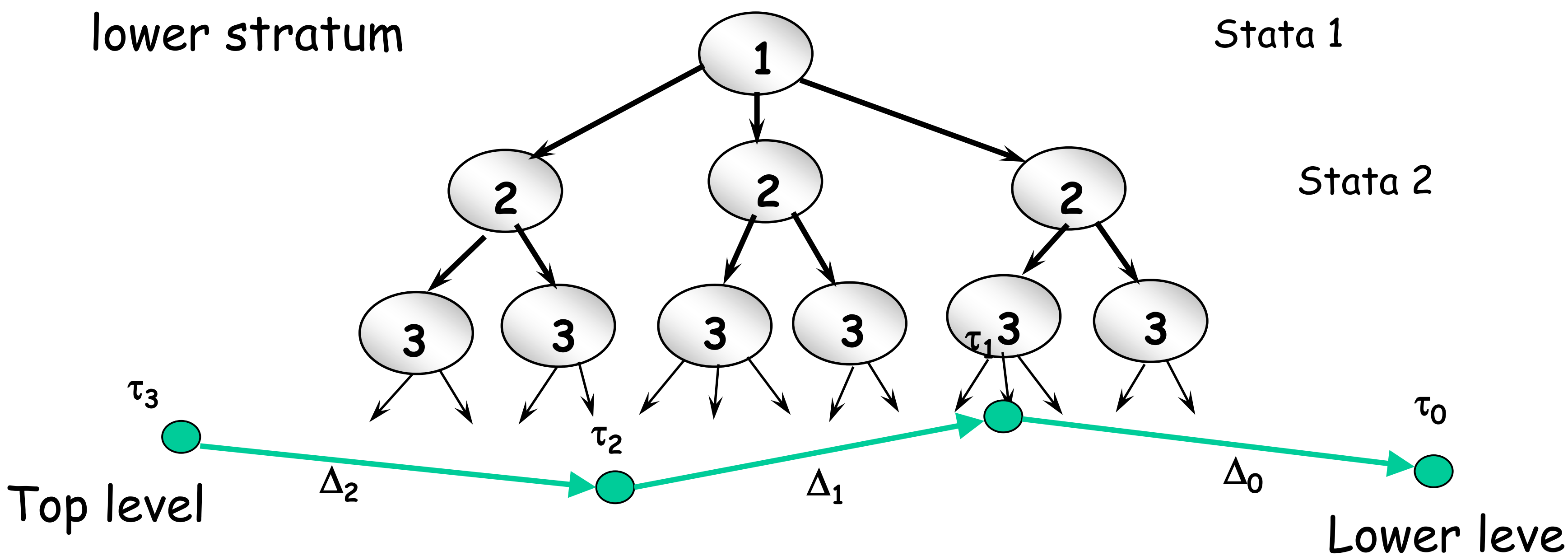
1. Uses an *elected master process* to synchronize among clients, without the presence of a time server
2. The *elected master* polls or broadcasts to all machines requesting for their time, adjusts times received for RTT & latency (eg Cristian's algorithm), averages times, and tells each machine how to adjust. Outliers can easily be ignored (fault tolerance).
3. ☹Averaging client's clocks may cause the entire system to drift away from UTC over time
4. Failure: elect a new leader (leader election algorithm)
5. ☹Failure of the master requires some time for re-election, so accuracy cannot be guaranteed

Network Time Protocol (NTP)

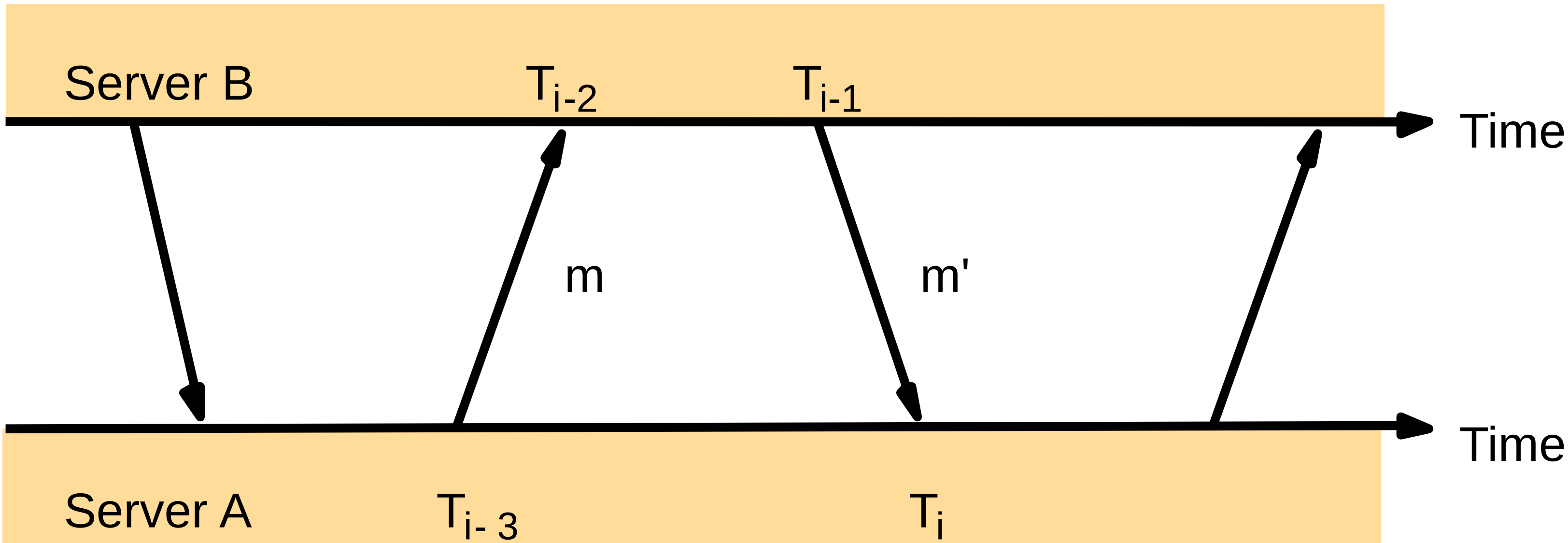
- Accurate distribution of UTC time
  - NTP enables clients across the Internet to be synchronized accurately to the UTC (uses UDP)
  - Large and variable message delays are tolerated through statistical filtering of timing data from different servers
- Scalability
  - NTP servers are hierarchically organized to speed up synchronization, and to scale to a large number of clients and servers
- Reliability and Fault Tolerance
  - There are redundant time servers, and redundant paths between the time servers
  - The architecture provides reliable service that can tolerate lengthy losses of connectivity
  - A synchronization subnet can reconfigure as servers become unreachable. For example, if Stratum 1 server fails, then it can become a Stratum 2 secondary server
- Security
  - NTP protocol uses authentication to check of the timing message originated from the claimed trusted sources

Network Time Protocol (NTP)

- In NTP, servers are connected in a logical hierarchy called *synchronization subnet*
- The levels of synchronization subnet is called *strata*
  - Stratum 1 servers have most accurate time information (connected to a UTC receiver)
  - Servers in each stratum act as time servers to the servers in the lower stratum

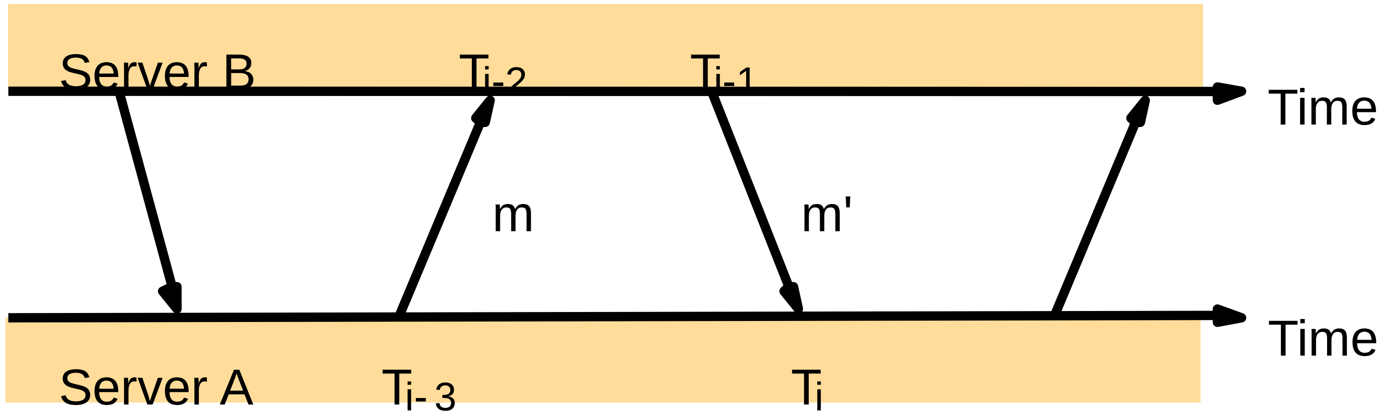


NTP uses message pairs



Each message contains timestamps of recent message events: the local time when the previous NTP message was sent and received, and the local time when the current message was transmitted.

Theoretical Base for NTP



- o_i: estimate of the actual offset between the two clocks
- o: true offset of the clock at B relative to that at A
- t and t': actual transmission times for m and m'
- d_i: estimate of accuracy of o_i ; total transmission times for m and m'; $d_i=t+t'$

$$T_{i-2} = T_{i-3} + t + o$$

$$T_i = T_{i-1} + t' - o$$

This leads to

$$d_i = t + t' = T_{i-2} - T_{i-3} + T_i - T_{i-1}$$

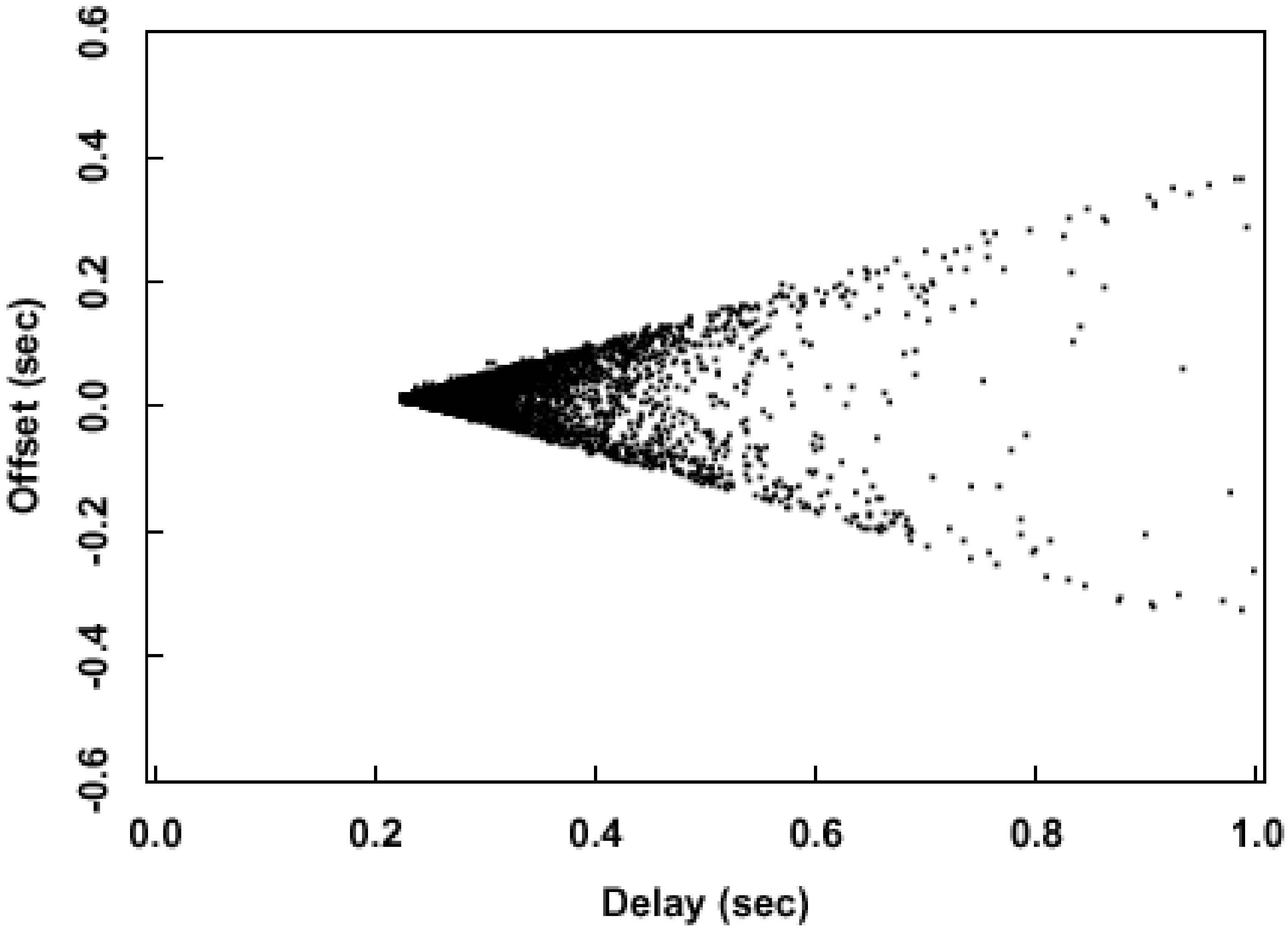
$$o = o_i + (t' - t)/2, \text{ where}$$

$$o_i = (T_{i-2} - T_{i-3} + T_{i-1} - T_i)/2.$$

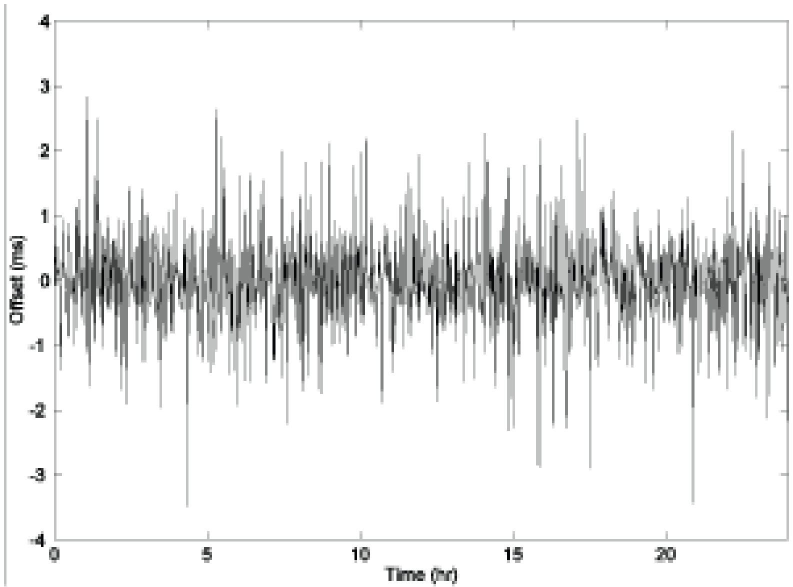
It can also be shown that

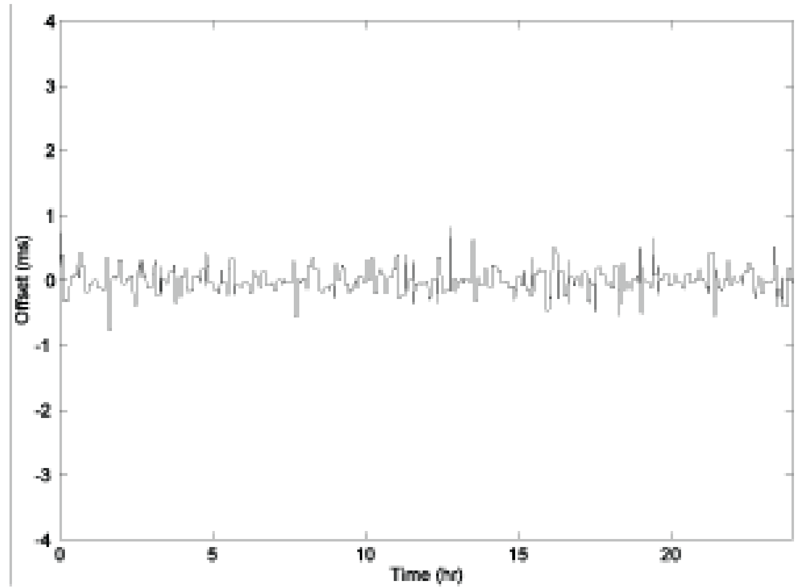
$$o_i - d_i/2 \le o \le o_i + d_i/2.$$

Benefits of Low Delay



Huff and Puff Filter





TrueChimers -- FalseTickers

Summary

- Physical clocks on computers are not accurate
- Clock synchronization algorithms provide mechanisms to synchronize clocks on networked computers in a DS
  - Computers on a local network use various algorithms for synchronization
    - Some algorithms (e.g, Cristian's algorithm) synchronize time with by contacting centralized time servers
    - Some algorithms (e.g., Berkeley algorithm) synchronize in a distributed manner by exchanging the time information on various computers
  - NTP provides architecture and protocol for time synchronization over wide-area networks such as Internet