NTP Architecture, Protocol and Algorithms

Sir John Tenniel; *Alice's Adventures in Wonderland,*Lewis Carroll

David L. Mills
University of Delaware
http://www.eecis.udel.edu/~mills
mailto:mills@udel.edu

Introduction

- Network Time Protocol (NTP) synchronizes clocks of hosts and routers in the Internet.
- Well over 100,000 NTP peers deployed in the Internet and its tributaries all over the world.
- Provides nominal accuracies of low tens of milliseconds on WANs, submilliseconds on LANs, and submicroseconds using a precision time source such as a cesium oscillator or GPS receiver.
- Unix NTP daemon ported to almost every workstation and server platform available today - from PCs to Crays - Unix, Windows, VMS and embedded systems.
- The NTP architecture, protocol and algorithms have been evolved over the last twenty years to the latest NTP Version 4 described in this briefing.

Needs for precision time

- Distributed database transaction journalling and logging
- Stock market buy and sell orders
- Secure document timestamps (with cryptographic certification)
- Aviation traffic control and position reporting
- Radio and TV programming launch and monitoring
- Intruder detection, location and reporting
- Multimedia synchronization for real-time teleconferencing
- Interactive simulation event synchronization and ordering
- Network monitoring, measurement and control
- Early detection of failing network infrastructure devices and air conditioning equipment
- Differentiated services traffic engineering

NTP capsule summary

- Primary (stratum 1) servers synchronize to national time standards via radio, satellite and modem.
- Secondary (stratum 2, ...) servers and clients synchronize to primary servers via hierarchical subnet.
- Clients and servers operate in master/slave, symmetric or multicast modes with or without cryptographic authentication.
- Reliability assured by redundant servers and diverse network paths.
- Engineered algorithms reduce jitter, mitigate multiple sources and avoid improperly operating servers.
- System clock is disciplined in time and frequency using an adaptive algorithm responsive to network time jitter and clock oscillator frequency wander.

Goals and non-goals

- Goals
  - Provide the best accuracy under prevailing network and server conditions.
  - Resist many and varied kinds of failures, including two-face, fail-stop, malicious attacks and implementation bugs.
  - Maximize utilization of Internet diversity and redundancy.
  - Automatically organize subnet topology for best accuracy and reliability.
  - Self contained cryptographic authentication based on both symmetric key and public key infrastructures and independent of external services.
- Non-goals
  - Local time – this is provided by the kernel.
  - Access control - this is provided by firewalls and address filtering.
  - Privacy - all protocol values, including time values, are public.
  - Non-repudiation - this can be provided by a layered protocol if necessary.
  - Conversion of NTP timestamps to and from other time representations and formats.

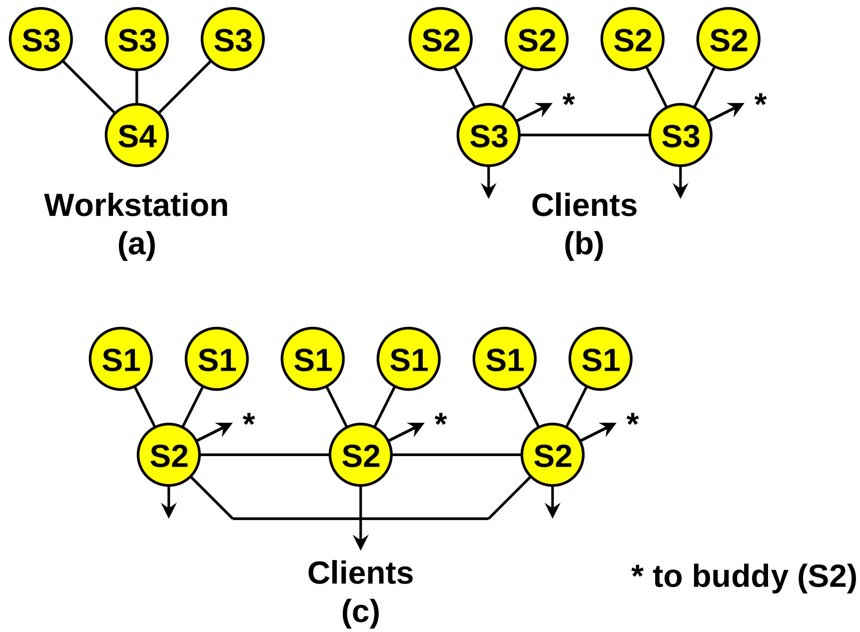
Autonomous system model

- Fire-and-forget software
  - Single software distribution can be compiled and installed automatically on most host architectures and operating systems.
  - Run-time configuration can be automatically determined and maintained in response to changing network topology and server availability.
- Optional autonomous configuration (autoconfigure)
  - Survey nearby network environment to construct a list of suitable servers.
  - Select best servers from among the list using a defined metric.
  - Reconfigure the subnet for best accuracy with overhead constraints.
  - Periodically refresh the list in order to adapt to changing topology.
- Optional autonomous authentication (autokey)
  - For each new server found, fetch and verify its cryptographic credentials.
  - Authenticate each message received using engineered protocol.
  - Regenerate keys in a timely manner to resist compromise.

Evolution to NTP Version 4

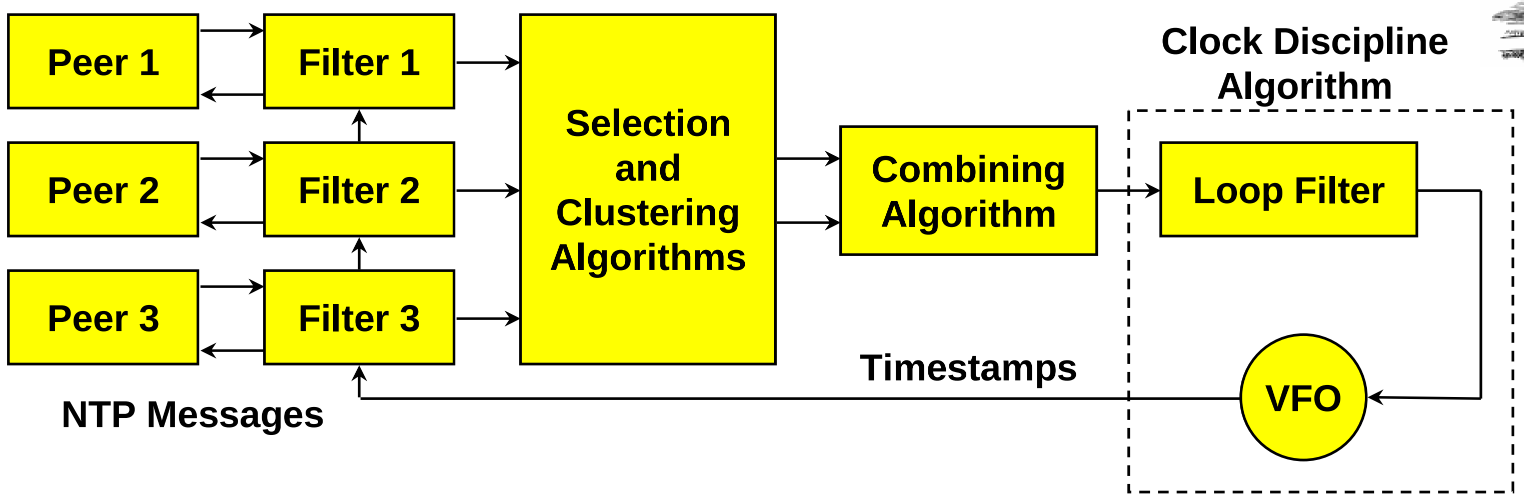
- Current Network Time Protocol Version 3 has been in use since 1992, with nominal accuracy in the low milliseconds.
- Modern workstations and networks are much faster today, with attainable accuracy in the low microseconds.
- NTP Version 4 architecture, protocol and algorithms have been evolved to achieve this degree of accuracy.
  - Improved clock models which accurately predict the time and frequency adjustment for each synchronization source and network path.
  - Engineered algorithms reduce the impact of network jitter and oscillator wander while speeding up initial convergence.
  - Redesigned clock discipline algorithm operates in frequency-lock, phase-lock and hybrid modes.
- The improvements, confirmed by simulation, improve accuracy by about a factor of ten, while allowing operation at much longer poll intervals without significant reduction in accuracy.

NTP configurations



- (a) Workstations use multicast mode with multiple department servers.
- (b) Department servers use client/server modes with multiple campus servers and symmetric modes with each other.
- (c) Campus servers use client/server modes with up to six different external primary servers and symmetric modes with each other and external secondary (buddy) servers.

NTP architecture overview



- Multiple servers/peers provide redundancy and diversity.
- Clock filters select best from a window of eight time offset samples.
- Intersection and clustering algorithms pick best *truechimers* and discard *falsetickers.*
- Combining algorithm computes weighted average of time offsets.
- Loop filter and variable frequency oscillator (VFO) implement hybrid phase/frequency-lock (P/F) feedback loop to minimize jitter and wander.

NTP protocol header and timestamp formats

NTP Protocol Header Format (32 bits)

LI	VN	Mode	Strat	Poll	Prec
Root Delay					
Root Dispersion					
Reference Identifier					
Reference Timestamp (64)					
Originate Timestamp (64)					
Receive Timestamp (64)					
Transmit Timestamp (64)					
Extension Field 1 (optional)					
Extension Field 2… (optional)					
Key/Algorithm Identifier					
Message Hash (64 or 128)					

Cryptosum

Authenticator (Optional)

LI	leap warning indicator
VN	version number (4)
Strat	stratum (0-15)
Poll	poll interval (log2)
Prec	precision (log2)

NTP Timestamp Format (64 bits)

Seconds (32)	Fraction (32)

Value is in seconds and fraction since 0^h 1 January 1900

NTPv4 Extension Field

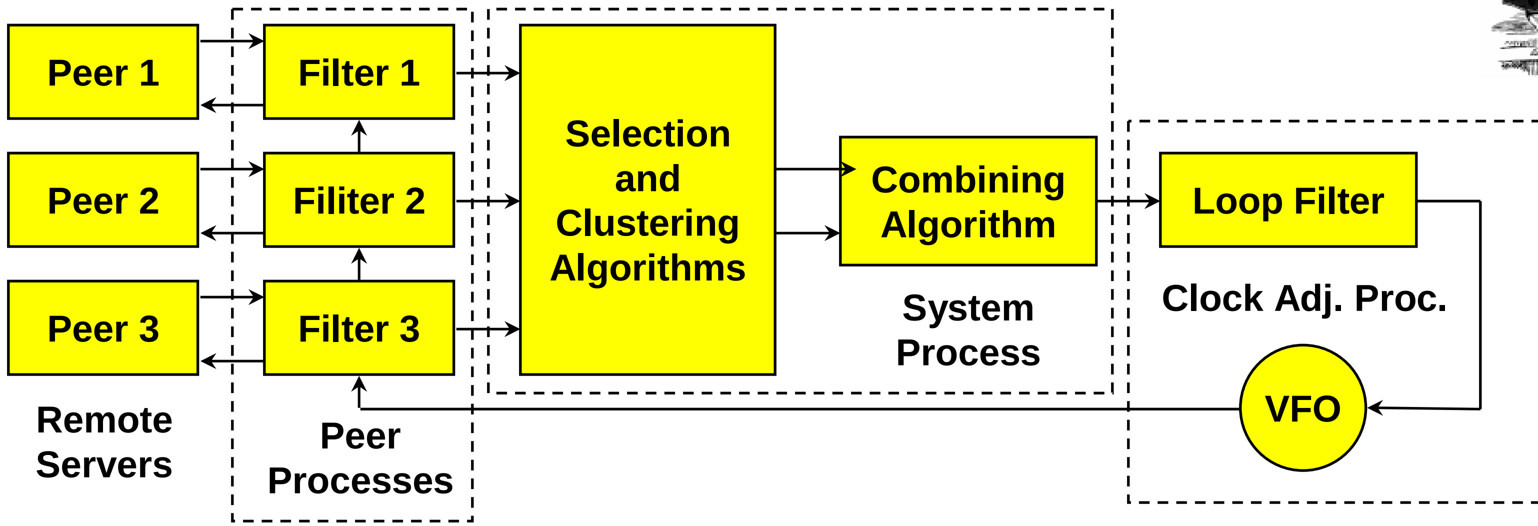
Field Length	Field Type
Extension Field (padded to 32-bit boundary)	

Last field padded to 64-bit boundary

NTP v3 and v4
NTP v4 only
authentication only

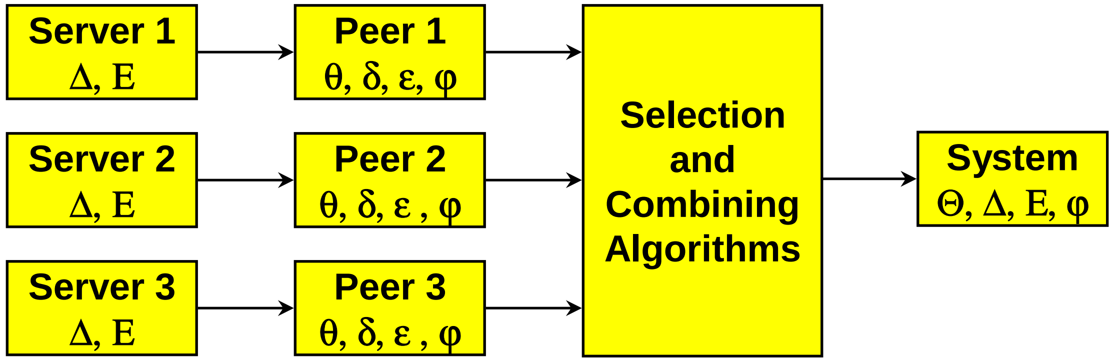
Authenticator uses DES-CBC or MD5 cryptosum of NTP header plus extension fields (NTPv4)

NTP process decomposition (NTPv4)



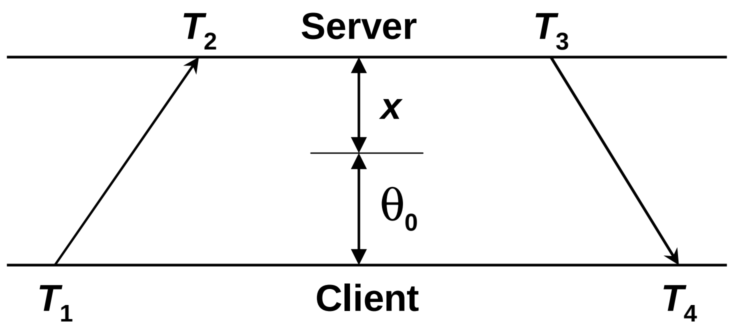
- Each peer process runs independently at poll intervals determined by the system process and remote server.
- System process runs at poll intervals determined by the measured network phase jitter and local clock oscillator frequency stability.
- Clock adjust process runs at 1-s intervals to discipline the VFO phase and frequency.

NTP dataflow analysis

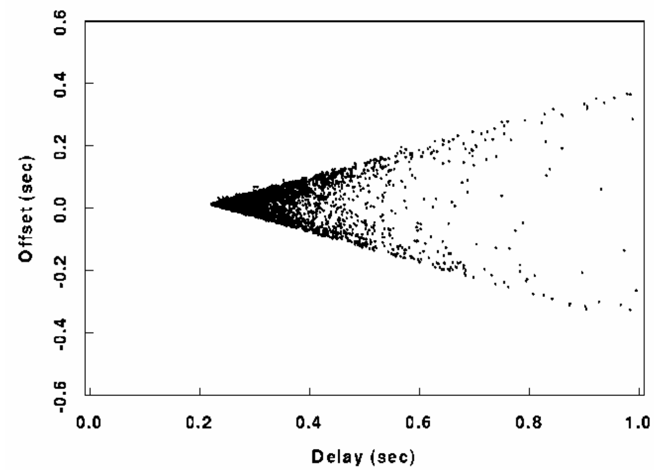


- Each server calculates server variables offset Θ, delay Δ and dispersion E relative to the root of the synchronization subtree.
- At each NTP message arrival, the peer process updates peer offset θ, delay δ, dispersion ε and jitter φ_r (NTPv4) from timestamps and clock filter algorithm.
- At system poll intervals, the clock selection and combining algorithms update system variables Θ, Δ, E, and φ.
- Dispersions ε and E increase with time at a rate depending on specified frequency tolerance ϕ.

Clock filter principles

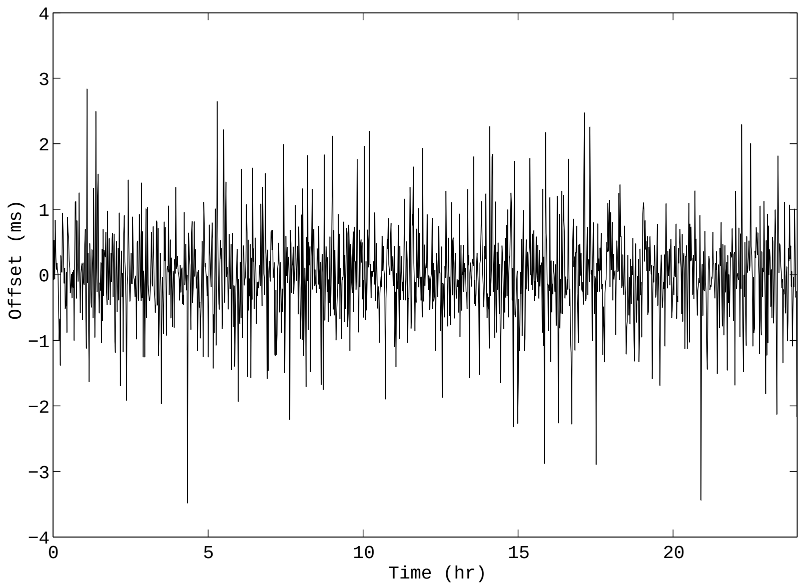


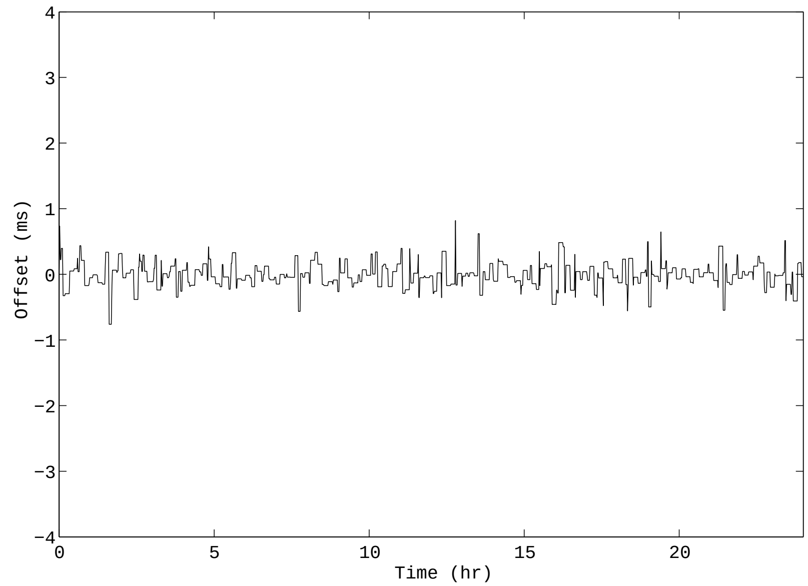
$$\theta = \tfrac{1}{2}[(T_2 - T_1) + (T_3 - T_4)]$$
$$\delta = (T_4 - T_1) - (T_3 - T_2)$$



- The most accurate offset θ_0 is measured at the lowest delay δ_0 (apex of the wedge scattergram).
  - The correct time θ must lie within the wedge $\theta_0 \pm (\delta - \delta_0)/2$.
  - The δ_0 is estimated as the minimum of the last eight delay measurements and (δ_0, θ_0) becomes the offset and delay output.
  - Each output can be used only once and must be more recent than the previous output.
- The synchronization distance metric λ is based on delay, frequency tolerance and time since the last measurement.

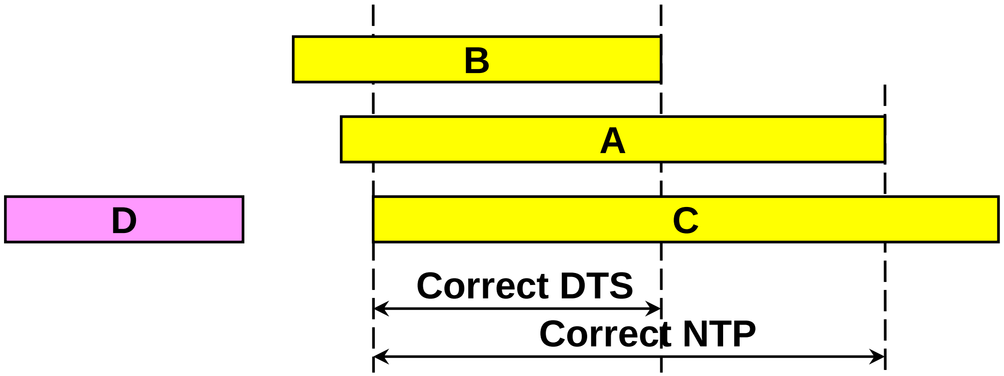
Clock filter performance





- Left figure shows raw time offsets measured for a typical path over a 24-hour period (mean error 724 μs, median error 192 μs)
- Right graph shows filtered time offsets over the same period (mean error 192 μs, median error 112 μs).
- The mean error has been reduced by 11.5 dB; the median error by 18.3 dB. This is impressive performance.

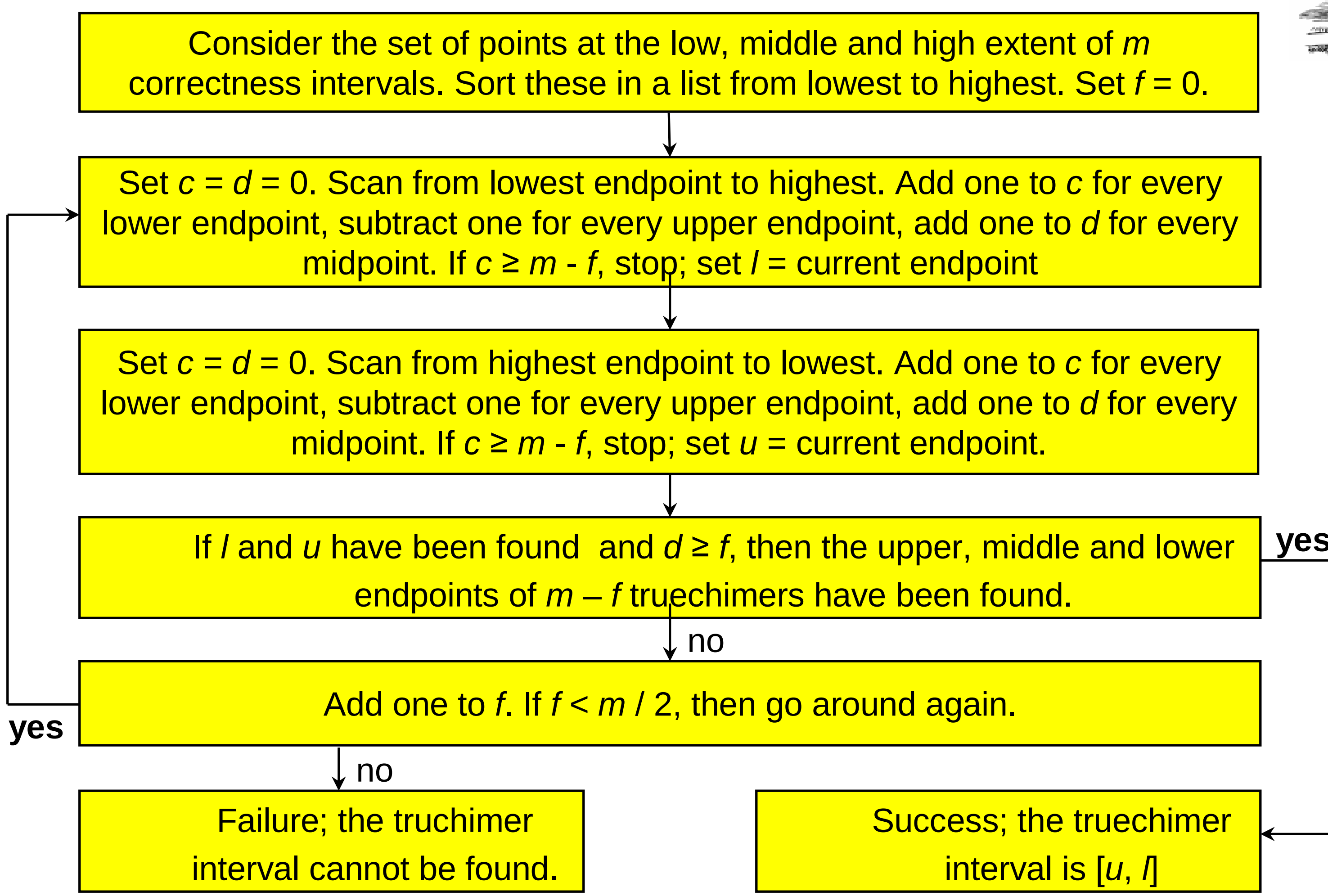
Clock selection principles



correctness interval = $\theta - \lambda \le \theta_0 \le \theta + \lambda$
m = number of clocks
f = number of presumed falsetickers
A, B, C are truechimers
D is falseticker

- The correctness interval for any candidate is the set of points in the interval of length twice the synchronization distance centered at the computed offset.
- The DTS interval contains points from the largest number of correctness intervals.
- The NTP truechimer interval requires the computed offset of each correctness interval to be contained in the truechimer interval.
- The selection algorithm finds the truechimer interval. Formal correctness assertions require at least half the candidates be in this interval. If not, no candidate can be considered a truechimer.

Clock selection algorithm



failure

Clustering algorithm

Sort survivors of clock selection algorithm by increasing synchronization distance E_B. Let n be the number of survivors and n_{min} a lower limit.

For each survivor s_i, compute the selection jitter φ_s (average offset differences between s_i and all others).

Let s_{max} be the survivor with maximum φ_s and s_{min} the survivor with minimum filter jitter φ_r (average offset differences for the same survivor).

$s_{max} \le s_{min}$ or $n \le n_{min}$?

yes

no

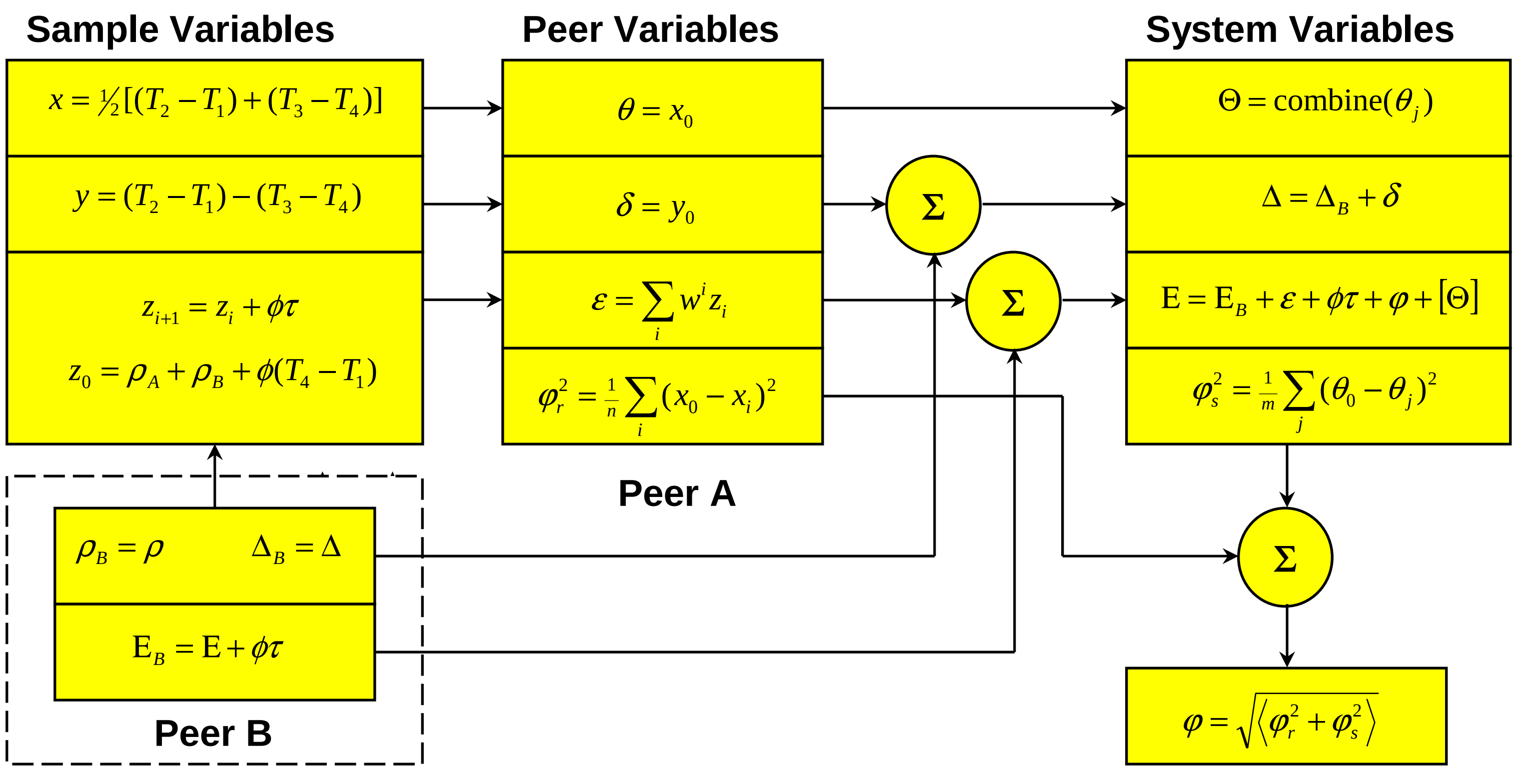
Delete the survivor s_{max}; reduce n by one

The resulting survivors are processed by the combining algorithm to produce a weighted average Θ used as the final offset adjustment

Error budget - notation

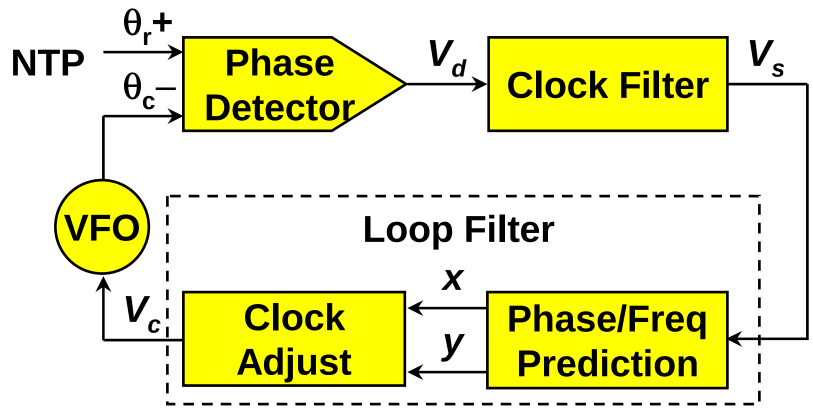
- Constants (peers *A* and *B*)
  - ρ maximum reading error
  - ϕ maximum frequency error
  - w dispersion normalize: 0.5
- Packet variables
  - Δ_B peer root delay
  - E_B peer root dispersion
- Sample variables
  - T_1, T_2, T_3, T_4 protocol timestamps
  - x clock offset
  - y roundtrip delay
  - z dispersion
  - τ interval since last update
- System variables
  - Θ clock offset
  - Δ root delay
  - E root dispersion
  - φ_s selection jitter
  - φ jitter
  - τ interval since last update
  - m number of peers
- Peer variables
  - θ clock offset
  - δ roundtrip delay
  - ε dispersion
  - φ_r filter jitter
  - n number of filter stages
  - τ interval since last update

Error budget - calculations



NTP Version 4 Error Budget

Clock discipline algorithm

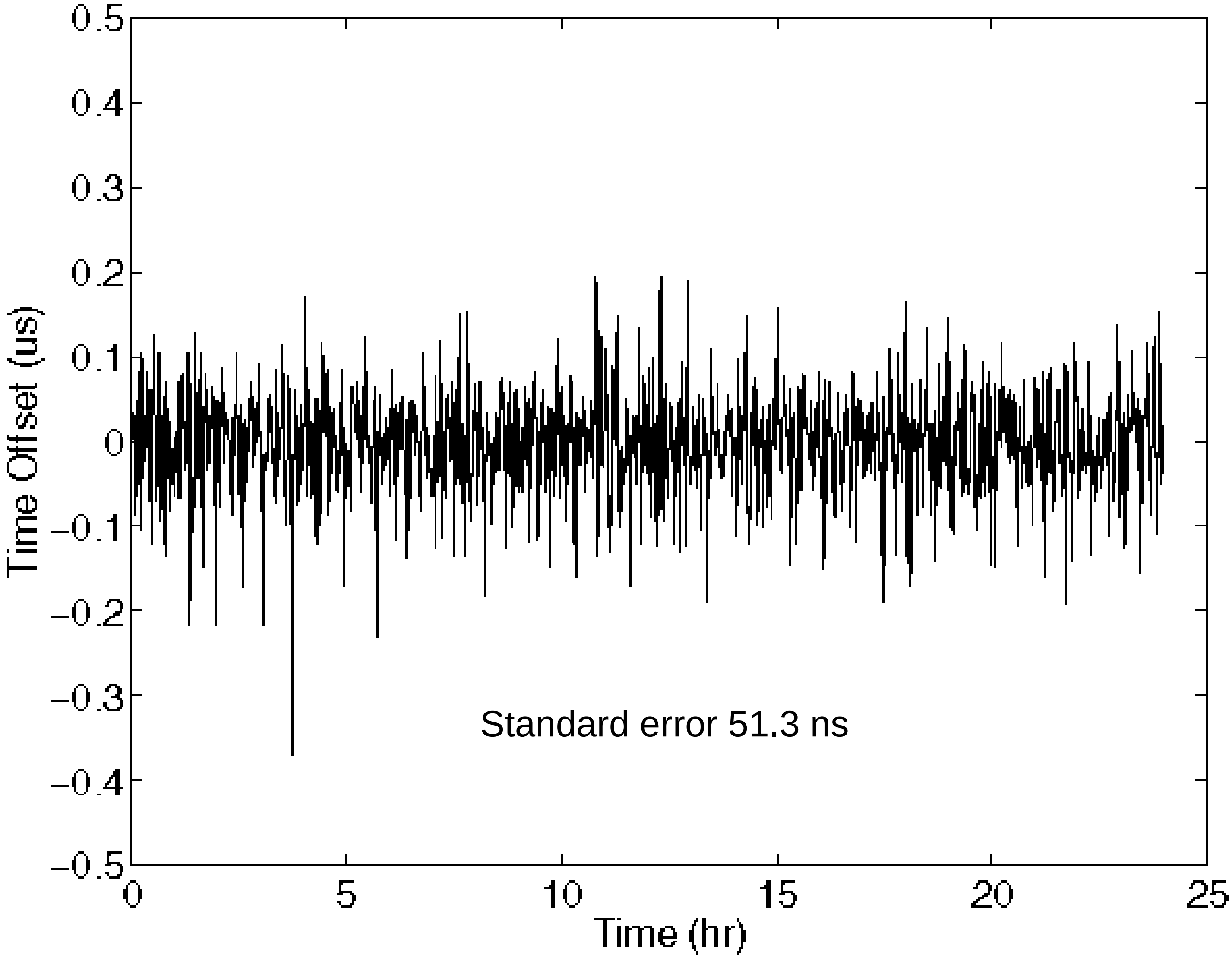


- V_d is a function of the phase difference between NTP and the VFO.
- V_s depends on the stage chosen on the clock filter shift register.
- x and y are the phase update and frequency update, respectively, computed by the prediction functions.
- Clock adjust process runs once per second to compute V_c, which controls the frequency of the local clock oscillator.
- VFO phase is compared to NTP phase to close the feedback loop.

Precision time performance issues

- Improved clock filter algorithm reduces network jitter
- Reduced hardware and software latencies minimize system errors.
- Engineered cryptographic authentication protocol avoids inline public-key algorithms.
- Optional kernel modifications achieve time resolution to 1 ns and frequency resolution to .001 PPM using NTP and PPS sources.
- Nonlinear hybrid adaptive-parameter (NHAP) clock discipline algorithm optimizes performance with update intervals from 16 seconds to 36 hours.
- Dynamic parameter estimation algorithm optimizes performance over a wide range of network jitter and oscillator wander.
- Temperature stabilized computer clock oscillators.
- Precision timing sources using GPS, LORAN-C and cesium clocks.

Measured PPS time error for Alpha 433



NTP online resources

- Network Time Protocol (NTP) Version 3 Specification RFC-1305
  - NTPv4 features documented in release notes and reports cited there
- Simple NTP (SNTP) Version 3 specification RFC-2030
  - Applicable to IPv4, IPv6 and ISO CNLS
- List of public NTP time servers (as of May 2001)
  - 107 active primary (stratum 1) servers
  - 136 active stratum 2 servers
- NTP Version 4 implementation and documentation for Unix, VMS and Windows
  - Ported to over two dozen architectures and operating systems
  - Utility programs for remote monitoring, control and performance evaluation
  - Complete documentation in HTML format
- Collaboration resources at http://www.eecis.udel.edu/~mills/resource.htm

Further information

- Network Time Protocol (NTP): http://www.ntp.org/
  - Current NTP Version 3 and 4 software and documentation
  - FAQ and links to other sources and interesting places
- David L. Mills: http://www.eecis.udel.edu/~mills
  - Papers, reports and memoranda in PostScript and PDF formats
  - Briefings in HTML, PostScript, PowerPoint and PDF formats
  - Collaboration resources hardware, software and documentation
  - Songs, photo galleries and after-dinner speech scripts
- FTP server ftp.udel.edu (`pub/ntp` directory)
  - Current NTP Version 3 and 4 software and documentation repository
  - Collaboration resources repository
- Related project descriptions and briefings
  - See “Current Research Project Descriptions and Briefings” at http://www.eecis.udel.edu/~mills/status.htm