

А. В. САВЧУК, В. Н. ШАПОШНИКОВ, И. П. ЧЕРНЯК

СИНХРОНИЗАЦИЯ ТЕКУЩЕГО ВРЕМЕНИ: ПРОТОКОЛ СЕТЕВОГО ВРЕМЕНИ

Синхронизация часов — важная и сложная проблема распределенных систем. К числу таких систем, многие приложения которых нуждаются в поддержании точности текущего времени в пространственно разнесенных пунктах в пределах секунды, относится, в частности, Интернет. Протокол сетевого времени (Network Time Protocol — NTP) удовлетворяет этому требованию в глобальной сети и более жестким требованиям — в корпоративных сетях.

Предлагаемая статья продолжает начатую [1] серию публикаций о современных способах сличения времени, адресованную прежде всего инженерам электросвязи.

Пять факторов риска для корпоративных сетей при отсутствии синхронизации текущего времени

Если компьютеры корпоративной сети работают без привязки к собственной *общей шкале времени*, то сеть подвержена ряду угроз, среди которых можно выделить следующие пять видов:

- **сбои в диагностике функциональных отказов** (из-за неточных меток времени нарушается последовательность выполнения программ в пространственно разнесенных сетевых элементах; упорядочить события в процессе восстановления работоспособности сети с помощью системы управления становится невозможно);

- **пропадание архивных данных** из-за неправильных меток времени (программное обеспечение обслуживания каталогов будет сохранять в виде последних версий устаревшие лог-файлы, требующие замены);

- **нарушение безопасности доступа к ресурсам сети** (при отсутствии собственной корпоративной шкалы времени вследствие использования серверов NTP в Интернете открываются «дыры» в аппаратно-программных средствах межсетевой защиты, причем неправильные метки времени лог-файлов не позволяют администратору отследить действия хакеров);

- **проблемы с биллингом** (без правильных меток времени не работают технические средства защиты информации о прибылях компании, например карт и жетонов, предназначенных для тарификации услуг; невозможно доказать аутентичность электронной подписи на контрактах или своевременность переводов);

- **сбои в работе системы общеканальной сигнализации № 7 (ОКС-7)** (из-за нарушения временной синхронизации между соединениями и сопряженной с ними сигнализацией).

Протокол сетевого времени NTP в Интернете

Метки времени NTP — одного из самых «долгоживущих» постоянно работающих приложений — были проставлены на опытных образцах программ

еще на заре развития «всемирной паутины». Все этапы становления технических условий на NTP прошли проверку в Интернете [2]. Хотя впервые о протоколе сличения времени заговорили на Национальной конференции по вычислительной технике (National Computer Conference), состоявшейся в США в 1979 году, документальное свидетельство он получил только в 1981-м в серии Уведомлений о технике Интернета (Internet Engineering Note series — IEN). На основании документа RFC-958 в 1985 году была введена версия NTPv0, представляющая собой лишь небольшую часть того, что содержится в современных версиях NTP. В то время в сетях Ethernet номинальная точность сличения времени составляла десятки миллисекунд, а в трактах, пересекающих Атлантику, — не хуже 100 мс.

Через три года появилась NTPv1 в виде документа RFC-1059, содержащего технические условия для протоколов и алгоритмов, а также примитивные версии фильтра часов, алгоритмов выбора и коррекции часов. Они разрабатывались на основе результатов серии экспериментов, оформленных

Sync-2007

В июне 2007 года в Киеве состоялся III Международный технический семинар «Прикладные решения для сетей тактовой и временной синхронизации «Sync-2007», организованный провайдером решений и оборудования синхронизации (Украина) компанией ВИРКОМ и ведущими мировыми производителями соответствующего оборудования JDSU (США), Symmetricom (США).

Его участниками были рассмотрены следующие основные вопросы:

- мировые тенденции развития методов синхронизации;
- синхронизация сетей NGN;
- результаты технической эксплуатации и мониторинга сетей синхронизации;
- состояние разработки национальных стандартов Украины по синхронизации сетей связи.

На семинаре состоялась презентация оборудования синхронизации и измерительной техники.

в виде отчета RFC-958, где дано теоретическое обоснование алгоритма фильтрации часов, определены режим «клиент/сервер» и симметричный режим, а также впервые введено поле версии в заголовке.

Технические условия для NTPv2 изложены в 1989 году в документе RFC-1119, в котором впервые построена формальная модель с описанием протокола в виде автомата состояний, а операции определены в псевдокодах. Кроме того, введены действующие и ныне протокол управления сообщениями NTP (NTP Control Message Protocol), предназначенный для управления серверами и клиентами, и схема криптографического распознавания на основе симметричного ключа.

В 1989 году велись оживленные дискуссии об услуге синхронизации оцифрованного времени (Digital Time Synchronization Service — DTSS), предложенной корпорацией DEC. Приверженцы NTP считали основным препятствием на пути внедрения DTSS недостаточную точность сличения времени из-за отсутствия алгоритма коррекции часов, тогда как создатели DTSS усматривали проблему NTP в том, что процесс его проектирования не формализован, отмечая как важное преимущество своего проекта правильность принципов его построения на основе непротиворечивого алгоритма. Этот алгоритм используется для оптимизации данных существующей версии NTP.

В 1992 году приняты технические условия на NTPv3 (документ RFC-1305), дополненные формальным описанием анализа погрешностей с оценкой их статистики, широкоэмитальным режимом, а автомат состояний — драйвером ввода-вывода для опорных часов. Потребовалось немного более года, чтобы полностью переработать вариант NTPv2, обеспечив его совместимость с техническими условиями на NTPv3, и таким образом объединить технические условия и реализацию в непротиворечивой формальной модели. Этого компромисса удалось достичь, внося изменения в обе версии.

Со временем стала очевидной необходимость обновить автомат состояний и псевдокод по сравнению с техническими условиями на NTPv3, т. е. назрела потребность перейти к следующей версии — NTPv4. Впоследствии разработали ее упрощенный вариант (RFC-2030) под названием Simple Network Time Protocol version 4 (SNTPv4) для стеков протоколов IPv4, IPv6. Протокол NTP совместим с SNTP, однако в последний не включены «смягчающие» алгоритмы тщательной обработки и коррекции часов, которые не нужны для серверов, совмещенных с приемниками GPS.

Сеть NTP иерархична (рис. 1): ее строят в виде направленного графа, где каждый узел принадлежит определенному уровню иерархии — от stratum 1 до stratum 15. К stratum 0 относят источники эталонного времени, с которыми непосредственно сличают время серверы stratum 1. По отношению к ним узлы stratum 2 являются клиентами, будучи в то же время ведущими серверами для узлов stratum 3 (и т. д.). Термины «клиент», «сервер» и «партнер» («peer»), несмотря на иерархию, применяются условно: например, узлы stratum 1 чаще всего выступают в качестве ведущих серверов, но некоторые из них могут работать в режиме «клиент/сервер» с узлами stratum 2. По оценкам, в глобальной сети имеется более 175 тыс. ведущих узлов, причем среди серверов stratum 1 удивительно много «хранителей» неточного времени [3]. Распределение узлов по уровням очень неравномерно: более половины узлов NTP в глобальной сети Интернет относятся к stratum 1–4. Оператор должен сконфигурировать подсеть NTP для своего компьютера вручную, после чего его вмешательства не требуется.

Стратум 0: GPS, национальные эталоны времени

Стратум 1: [узлы]

Стратум 2: [узлы]

Стратум 15: [узлы]

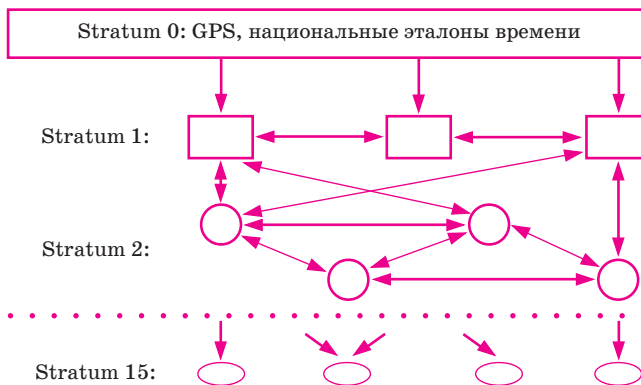


Рис. 1

Следует отметить, что NTP — не единственная система сличения времени в Интернете. В составе протокола Интернет определены и другие алгоритмы, предназначенные для регистрации и передачи времени событий, в том числе протоколы Daytime и Time, сообщение о метке времени ICMP (Internet Control Message Protocol — протокол управляющих сообщений сети Интернет), имеющие статус стандарта Интернет, а также опция метки времени IP. *Важное преимущество NTP по сравнению с остальными средствами сличения времени — наличие алгоритмов, призванных обеспечить выбор лучшего показания времени от каждого сервера и лучшей комбинации серверов.* К ним относятся:

- «смягчающие» алгоритмы обработки данных (фильтрации, выбора, группирования и взвешенного суммирования);
- алгоритм коррекции часов для подстройки частоты компьютерных часов.

Охватить все аспекты NTP [2] в одной публикации невозможно. Цель статьи состоит в том, чтобы описать физические процессы, обслужи-

ваемые указанными алгоритмами, на основе понятий, близких к теории оптимального приема [4]. В данном контексте в качестве «пространства сигналов» выступает совокупность показаний времени, принимаемых от множества серверов, а «расстояние» между точным временем и его оценкой определяется как предполагаемая погрешность вычисленного расхождения времени. Нужно, однако, учитывать, что аналогия здесь неполная: в сущности, речь идет не об оптимальной обработке сигналов времени, а скорее об «оптимистическом» результате дистанционного сличения времени.

Алгоритмы NTP

Сам по себе NTP — это программа реального времени, которая управляет процессами в соответствии со структурной схемой, показанной на рис. 2 [2].

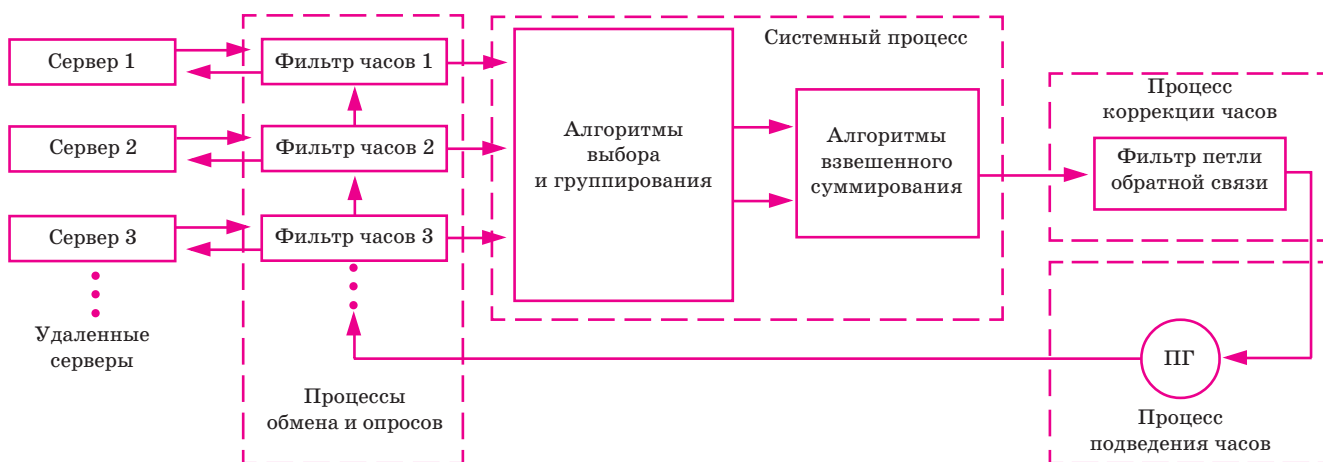


Рис. 2

Процессы обмена и опросов обеспечивают одновременную работу с несколькими серверами и обслуживание множества клиентов. Фильтры часов служат для выбора лучших из последних сличений времени по часам восьми серверов. Системный процесс протекает по мере того, как на выходе фильтра часов формируются новые данные. Они обрабатываются набором из трех последовательных алгоритмов — выбора, группирования и взвешенного суммирования — для обработки расхождений времени с целью внести поправки в процесс коррекции часов, функции которого эквивалентны фильтру нижних частот в петле обратной связи и подстройки частоты подстраиваемого генератора (ПГ).

Обмен пронумерованными метками времени происходит одновременно с N серверами. Как только появляется пакет с сообщением NTP, время удаленного сервера сравнивают со временем местных (системных) часов и определяют характерное для данного сервера расхождение времени. Этот способ обмена метками времени (в [8] используется тер-

мин «встречный метод», а в [9] — «двусторонний метод сравнения шкал времени»), не требующий высокой надежности доставки сообщений (что очень важно для применения в сети Интернет), проиллюстрирован на рис. 3. Равноправные серверы A и B с помощью симметричной двусторонней передачи могут независимо вычислить одностороннюю задержку d и расхождение θ времени часов B относительно времени часов A по формулам [1]:

$$d_i = (T_i - T_{i-3}) - (T_{i-1} - T_{i-2}); \quad (1)$$

$$\theta_i = \frac{(T_{i-1} + T_{i-2}) - (T_i + T_{i-3})}{2}, \quad (2)$$

где T_{i-3} , T_{i-2} , T_{i-1} — три последние метки времени, которые содержатся в каждом сообщении NTP, а четвертая — T_i — показывает местное время (эпоху) его приема.

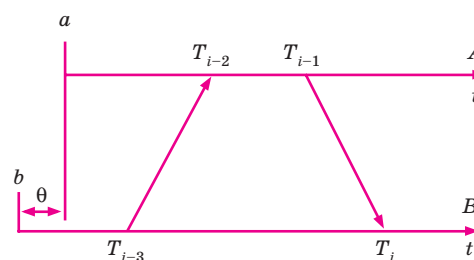


Рис. 3

Алгоритм фильтра часов оценивает расхождение часов по списку выборок (θ_i, d_i) , полученных в условиях реальной нагрузки в сети. Здесь $0 \leq i \leq N - 1$; N — максимальное число выборок в списке (в полной стандартной реализации протокола $N = 8$) [2]. Обоснованием фильтра часов служит то обстоятельство, что в Интернете очереди пакетов большую часть времени невелики, а перегрузки возникают нечасто. К тому же маршрутизаторы минимизируют число преприемов пакетов, а следовательно, и длину очередей. Вероятность того, что пакет NTP попадет в данном направлении

в длинную очередь, невысока; еще менее вероятно, что в определенном центре коммутации пакеты окажутся в длинных очередях в обоих направлениях. Следовательно, чем меньше задержка d_i , вычисленная для данного сервера по (1), тем точнее будет расхождение времени θ_i в выражении (2) для этого же сервера. Другими словами, наибольшая точность оценок расхождения времени достигается при наименьших значениях задержки. Это наглядно иллюстрируется представленной на рис. 4 характеристикой типового тракта в Интернете, которую называют диаграммой клиновидного рассеяния [5]. При небольшой нагрузке на сеть точки концентрируются на острие клина, а по мере ее увеличения смещаются правее вдоль линий по краям клина, постепенно заполняя его пространство все дальше вправо. Из этих данных становится очевидным, что наилучшую оценку θ_i и d_i дают точки вблизи острия клина, — и алгоритм фильтра часов разработан именно для того, чтобы их выделить.

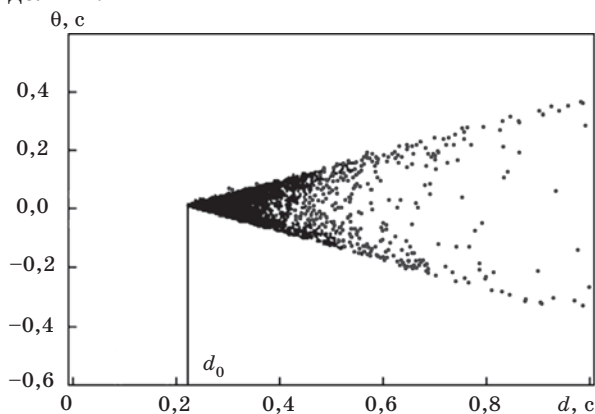


Рис. 4

С этой целью выборки (θ_i, d_i) дополняют еще двумя параметрами, получая в итоге $(\theta_i, d_i, \varepsilon_i, t_i)$, где t_i — эпоха метки времени T_i (см. рис. 3); ε_i — «разброс фильтра» партнера,

$$\varepsilon_i = \sum_{j=0}^{N-1} \frac{|\theta_i - \theta_j|}{2^{j+1}}. \quad (3)$$

Интервал времени между вычислениями двух последовательных выборок $(\theta_i, d_i, \varepsilon_i, t_i)$ называют *интервалом обновления* и обозначают μ .

Если пакет NTP не поступает от партнера в течение трех интервалов опроса подряд, то в список вводят «недостающую» выборку $(0, 0, \infty, t_i)$, которую никогда не используют в последующих вычислениях, но размещают в списке, чтобы «вытолкнуть» из него самую старую выборку, не допустив ее задействования.

Кроме того, из списка исключают выборки, полученные более чем 1500 с назад: их считают ненадежными, поскольку стабильность генераторов типичных компьютерных часов начинает ухудшаться на интервалах измерения более 1500 с. В результате в списке остается n выборок $(1 \leq n \leq N)$ для каждого сервера, причем первая из них $(\theta_0, d_0, \varepsilon_0, t_0)$, $i = 0$, характеризуется наименьшей задержкой.

Затем устраняют «выбросы шума» (типичное проявление перегрузки тракта Интернета в виде кратковременного импульсного шума, поражающего обычно одну выборку). С этой целью из списка исключают выборку, для которой модуль разности $|\theta_i - \theta_{i-1}|$ расхождений времени в текущем и предыдущем сообщениях NTP превышает порог $K_s \varphi$, где $K_s = 3$ (по умолчанию); φ — «джиттер¹ партнера», представляющий собой среднеквадратичное отклонение расхождения времени по n оставшимся выборкам в фильтре часов для каждого сервера,

$$\varphi = \left(\frac{1}{n-1} \sum_{k=1}^{N-1} (\theta_k - \theta_0)^2 \right)^{\frac{1}{2}}, \quad 0 \leq k \leq n-1. \quad (4)$$

В результате на выходе фильтра часов формируются данные в виде списка из $n \leq N$ лучших показаний часов для каждого сервера, что позволяет *уменьшить максимальную погрешность оценки расхождения времени более чем в 5 раз, а стандартное отклонение — в 3,5 раза* [2].

С применением *алгоритма выбора часов* решается следующая задача: по подмножеству из M показаний часов от разных серверов отыскать и исключить f неправильных, а $M - f$ предположительно правильных представить для обработки с помощью алгоритма группирования. Для этого определяют *интервал верности* для k -го сервера в виде

$$\theta_k - \Lambda_k \leq \theta_t \leq \theta_k + \Lambda_k, \quad (5)$$

где Λ_k — суммарная погрешность оценки расхождения времени θ_k ($0 \leq k < M$) по отношению к первичному серверу, называемая *корневым расстоянием*²; θ_t — *истинное расхождение времени* — предположительно должно находиться внутри этого интервала.

Идею алгоритма пересечения проще всего пояснить на примере [6].

Пусть в соответствии с (5) получены оценки расхождения местных часов $\theta_k \pm \Lambda_k$ по отношению к трем серверам (см. таблицу).

¹ В контексте NTP джиттер (jitter) соотносят с измеренными вариациями задержек в сети Интернет, а блуждания фазы (wander) — с нестабильностью частоты генератора компьютерных часов.

² Термин «корневое расстояние» обусловлен тем, что указанный интервал используется при организации древовидной подсети NTP для определения кратчайшего пути по дереву от «ветви» клиента до «корня», в котором находится первичный сервер.

Сервер	$\theta_k \pm \Lambda_k$, мс	Інтервал верности
1	$8,5 \pm 0,5$	(8; 8,5; 9)
2	10 ± 2	(8; 10; 12)
3	11 ± 1	(10, 11, 12)

Інтервали верности (см. последню графу таблиці) представляють тріма числами — двома крайніми допустимими значеннями $\theta_k \pm \Lambda_k$ вместе с измеренним расхождением времени θ_k , помещенним в центре соответствующего интервала верности.

Если все часы — правильные, то должен существовать непустой *интервал пересечения* из точек, принадлежащих всем трем интервалам верности. В рассматриваемом примере это условие не выполняется: интервал пересечения (10, 12) существует только для интервалов верности серверов 2 и 3; среднее его значение равно 11, а погрешность оценки составляет ± 1 . Часы сервера 1 считаются неправильными и подлежат отбраковке. Таким образом, $M = 3, f = 1$.

Алгоритм группирования служит для того, чтобы из $M - f$ предположительно правильных показаний часов, выделенных с помощью алгоритма выбора, выбрать $M_{\min} = 3$ самых точных. При этом существует возможность оставить единственные правильные часы и использовать их расхождение времени для коррекции системных часов. Однако такое «жесткое решение» имеет серьезный недостаток: в этом случае будет утрачено слишком много полезных статистических данных. Более эффективным способом повышения помехоустойчивости оказалось сбалансированное сочетание алгоритмов группирования и взвешенного суммирования. Для этого параметр φ_k , вычисленный по (4) для k -го сервера, используют как критерий пригодности последнего в качестве системного партнера. Затем формируют список из фрагментов данных $(\theta_k, \varphi_k, \lambda_k)$, упорядоченных по возрастанию множителя λ_k :

$$\lambda_k = s_k \Lambda_{\max} + \Lambda_k,$$

где s_k — уровень (stratum) сервера; Λ_k — корневое расстояние из (5); Λ_{\max} — наибольшее из всех значений Λ_k .

По аналогии с (4) для каждого партнера вычисляют «джиттер выбора» φ_{sk} как среднеквадратичное значение разностей расхождения времени, но уже по ансамблю оставшихся $M - f$ серверов:

$$\varphi_{sk} = \left(\frac{1}{M - f - 1} \sum_{k=1}^{M-f-1} (\theta_i - \theta_k)^2 \right)^{\frac{1}{2}}. \quad (6)$$

Если $M - f \leq M_{\min} = 3$ или

$$\max(\varphi_{sk}) \leq \min(\varphi_j), \quad 0 \leq k < M, \quad 0 \leq j < M,$$

то все фрагменты данных $(\theta_k, \varphi_k, \lambda_k), 0 \leq k \leq M - f$, поступают на вход алгоритма взвешенного суммирования. В противном случае список из $M - f$ упорядоченных фрагментов данных уменьшают до M_{\min} , отбрасывая те фрагменты, для которых $x_k = \lambda_k \varphi_{sk}$ принимает наибольшее значение.

В результате работы алгоритма группирования остаются по меньшей мере одни часы, которые считаются правильными. В этом случае остается только один партнер, который становится системным сервером.

Если же $M_{\min} = 3$, то фрагменты данных обрабатывают с помощью алгоритма взвешенного суммирования.

Алгоритм взвешенного суммирования обеспечивает вычисление системного расхождения времени Θ , предназначенного для алгоритма коррекции часов (который будет описан далее).

Пусть фрагменты данных $(\theta_i, \varphi_i, \Lambda_i)$ представляют соответственно расхождение времени, джиттер и корневое расстояние равноправного пользователя для i -х оставшихся часов. Тогда средневзвешенное значение расхождения времени имеет вид

$$\Theta = a \sum_i \frac{\theta_i}{\Lambda_i},$$

где a — нормирующий множитель,

$$a = \left(\sum_i \frac{1}{\Lambda_i} \right)^{-1}.$$

Алгоритмы выбора, группирования и взвешенного суммирования обрабатывают данные, поступающие от набора фильтров, чтобы сформировать для алгоритма коррекции часов входной сигнал V_s (рис. 5).

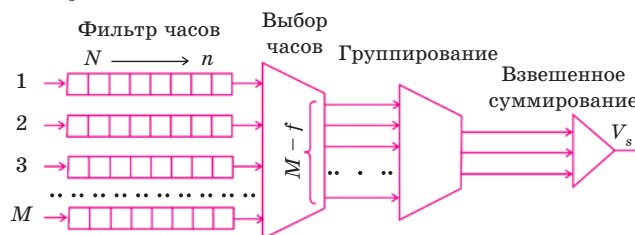


Рис. 5

Алгоритм коррекции часов, которую называют *дисциплинированием*³, способен автоматически приспосабливаться к изменениям операционной среды без ручного вмешательства и без функций управления в реальном времени.

Дисциплинирование в протоколе NTPv4 реализовано в виде системы управления с обратной связью, которая показана на рис. 6, где θ_r — опорная фаза, полученная в результате обработки расхождения времени по отношению к равноправному серверу; θ_c — управляемая фаза ПГ;

³ Дисциплинированием генератора называют подстройку его частоты в случаях, когда хотят подчеркнуть, что регулировки регулярны, но не непрерывны.

сигнал V_d соответствует разности фаз $\theta_r - \theta_c$. Выходной сигнал V_c фильтра петли управляет частотой ω_c ПГ, а следовательно, и его фазой $\theta_c = \int \omega_c dt$. Поправку времени x (см. рис. 6) используют для подведения часов независимо от частоты их генератора.

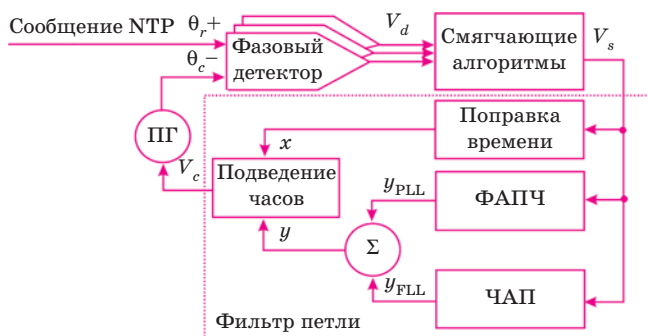


Рис. 6

В фильтре петли совмещены две системы управления с обратной связью, различные по принципам построения. Система фазовой автоподстройки частоты (ФАПЧ) работает лучше в условиях, когда превалирует джиттер, а система частотной автоподстройки (ЧАП) — при преобладании блужданий фазы генератора. В процессе дисциплинирования оптимальная комбинация поправок ФАПЧ и ЧАП выбирается автоматически в широком диапазоне значений системного джиттера и блужданий фазы генератора.

Регулировку относительной частоты ПГ y_{PLL} с помощью ФАПЧ в результате интегрирования последних расхождений времени можно представить в виде

$$y_{PLL} = \frac{V_s \mu}{(64T_c)^2}, \quad (7)$$

где T_c — постоянная времени петли ФАПЧ; μ — интервал обновления.

В режиме ЧАП y_{FLL} — подстройка частоты, вычисленная непосредственно по разности между текущей и последней поправками времени на интервале обновления:

$$y_{FLL} = \frac{V_s - x}{8\mu}, \quad (8)$$

где x — последняя поправка времени.

Чтобы сохранить непрерывность хода часов и не нарушить правило монотонности [1], т. е. не допустить обратного хода местных часов, при больших расхождениях времени частотой ПГ управляют постепенно, малыми регулировками на интервалах в 1 с.

Схема NTPv3 выбирает режим либо ФАПЧ, либо ЧАП в зависимости от интервала обновления (ФАПЧ используется при $\mu < 1024$ с, а ЧАП — во всех остальных случаях), но автоматический выбор по критерию преобладания сетевого джиттера или блужданий частоты генератора отсутствует.

В схеме NTPv4 петля обратной связи разработана с таким расчетом, чтобы она работала действительно как совмещенный алгоритм, в котором значения относительной частоты вычисляют отдельно с помощью ФАПЧ и ЧАП и суммируют с весами, зависящими от усредненной среднеквадратичной погрешности и предыдущих вычислений. В результате ФАПЧ имеет больший весовой коэффициент при наличии интенсивного джиттера (вызванного, например, перегрузкой сети), а ЧАП — при условии преобладания блужданий частоты генератора (например, вследствие значительных колебаний температуры). Поведение весовых множителей ЧАП и ФАПЧ, которые в сумме равны единице, отображено на рис. 7 (кривые соответственно 1 и 2) [7].

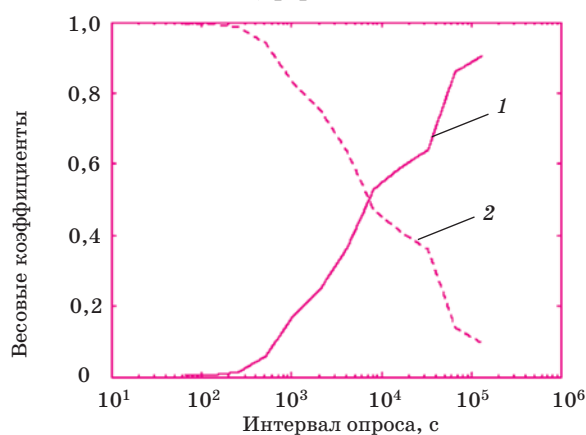


Рис. 7

Как показано в [7], после неоднократных доработок алгоритма коррекции часов удалось на порядок улучшить характеристики NTPv4 по сравнению с NTPv3. В частности, благодаря оптимальному сочетанию данных на выходах ФАПЧ и ЧАП интервал опроса серверов времени может быть увеличен до суток в реальных условиях без какой-либо начальной калибровки.

Литература

1. Савчук А. В., Шапошников В. Н., Черняк И. П. Теоретические основы синхронизации текущего времени в телекоммуникациях // Зв'язок.— 2007.— № 3.— С. 5–9.
2. Minar N. A Survey of the NTP Network / MIT Media Lab, Cambridge, MA, USA, December 9, 1999 // <http://www.media.mit.edu/~nelson/>
3. Mills D. L. Computer Network Time Synchronization: The Network Time Protocol.— Boca Raton, Fl.: CRC Press, 2006.
4. Зюко А. Г. Помехоустойчивость и эффективность систем связи.— М.: Связь, 1972.
5. Mills D. L. Modeling and Analysis of Computer Network Clock: Electrical Engineering Department Report 92-52, University of Delaware, July, 1992.— 29 p.
6. http://en.wikipedia.org/wiki/Marzul-lo's_algorithm
7. Mills D. L. Clock discipline algorithms for the Network Time Protocol Version 4: Electrical Engineering Report 97-3-3, University of Delaware, March 1997.— 35 p.
8. Кашеев Б. Л., Коваль Ю. А., Горбач В. И., Бондарь Б. Г. Метеоры сегодня.— К.: Техника, 1996.
9. Кашеев Б. Л., Коваль Ю. А., Кундюков С. Г. О возможности использования радиометеорного канала для синхронизации национальной сети связи Украины // Зв'язок.— 1999.— № 5.— С. 32–36.