

УДК 004.042

УПРАВЛЕНИЕ БУФЕРИЗАЦИЕЙ ВИДЕОИНФОРМАЦИИ, ДЕКОДИРОВАННОЙ ИЗ ЦИКЛИЧЕСКИХ СТРУКТУР

© 2024 г. М. Ю. Звездочкин^а, В. В. Миронов^б, *

^аФилиал акционерного общества «Ракетно-космический центр «Прогресс» –
особое конструкторское бюро «Спектр», Рязань, Россия

^бРязанский государственный радиотехнический университет имени В.Ф. Уткина, Рязань, Россия

*e-mail: mironovlv@mail.ru

Поступила в редакцию 18.04.2023 г.

После доработки 08.01.2024 г.

Принята к публикации 29.01.2024 г.

Поставлена задача соблюдения временных междукадровых интервалов при извлечении видеокадров из входного потока, сгруппированного в циклические квазителеметрические структуры. Рассмотрена организация программного буфера либо совокупности буферов видеокадров как общий подход к решению указанной задачи. Описано несколько способов организации таких буферов и управления их работой в системе обработки и передачи видеоинформации. Проведена экспериментальная проверка описанных способов, предложены рекомендации по применению.

Ключевые слова: бортовая система видеоконтроля, обработка видеоинформации, потоковое видео.

DOI: 10.31857/S0002338824020124, EDN: VODEYE

CONTROL OF VIDEO BUFFERING FOR VIDEOSTREAMS DECODED FROM CYCLIC STRUCTURES

M. Y. Zvyozdochkin^а, V. V. Mironov^б, *

^аJoint Stock company Space Rocket Centre Progress - SDB «Spectr»

^бRyazan State Radio Engineering University named after V.F. Utkin

*e-mail: mironovlv@mail.ru

Article formulates a problem of compliance of time inter-frame delays via video frames decoding from input cyclic-structured video stream. Implementation of software buffer or buffers set is reviewed as a general approach to solving this problem. Some methods of implementation of such buffers for video processing and transmission systems are proposed. Experimental research of these methods is conducted; recommendations for use are offered.

Keywords: onboard video monitoring system, video processing, video streaming.

Введение. Среди всего многообразия систем видеоконтроля и видеонаблюдения особняком находятся системы, передающие видеоинформацию (ВИ) поверх циклических информационных структур. В роли циклических структур могут выступать, в частности, структуры, используемые в телеметрии. Применение телеметрических структур для ВИ характерно для космической отрасли при совместной передаче телеметрической информации и ВИ [1–3].

Циклы входной ВИ и видеокадры асинхронны. Один цикл может содержать один или несколько видеокадров, при этом видеокадр может быть «разорван» между соседними циклами. И циклы, и кадры имеют свои собственные маркеры (синхрогруппы).

Для подобных систем также характерны следующие особенности:

один информационный поток содержит комбинированный видеопоток от нескольких камер, каждый кадр включает в себя идентификатор видеокамеры;

частота кадров по отдельно взятой камере является переменной;

частота циклов объединенного информационного потока фиксирована;

каждый кадр имеет индивидуальную метку времени.

Переменная частота кадров долгое время являлась достаточно редким решением для потокового видео, публикации на данную тему редки и носят разрозненный характер [4]. Лишь в последнее время переменная частота стала получать популярность с распространением кодака H.265. Из форматов медиафайлов переменная частота кадров поддерживается форматом MKV (Matroska).

Для рассматриваемого класса систем на конкретном временном участке (в космических запусках — участок циклограммы полета) переменная частота кадров находится в окрестности некоторого среднего значения. При переходе к следующему участку это среднее значение существенно меняется.

1. Постановка задачи. В системе обработки и передачи (СОП) ВИ, принятая в циклических структурах, должна быть преобразована в общепринятые видеоформаты и отправлена конечным потребителям, а также зарегистрирована в видеофайлы общепринятых форматов. Отправка конечным потребителям может осуществляться, например, по протоколам RTSP/RTP (real-time streaming protocol / real-time transport protocol) либо как поток пакетов протокола UDP (user datagram protocol). Зарегистрированная ВИ в дальнейшем может подвергаться дополнительной обработке [5].

При отправке конечным потребителям междукладовые интервалы каждого потока должны соответствовать исходным интервалам, сформированным видеокамерой. В противном случае потребитель будет наблюдать артефакты в виде неравномерного по времени, «дерганого» видеоряда.

Для формирования достоверных междукладовых интервалов видеокадры, декодированные из комбинированного сигнала, должны помещаться в программный буфер (буферы) и извлекаться из него непосредственно перед отправкой ВИ потребителям. Буферизация видеопотока широко применяется в передаче потоковой ВИ [6, 7]. При этом могут варьироваться такие параметры, как количество буферов (один буфер на все кадры либо отдельные буферы по каждой камере), размер каждого буфера (в видеокадрах), время жизни (lifetime) буфера (буфер может быть как циклическим, так и формироваться на заранее заданное время), метод определения времени извлечения кадров из буфера.

В статье будет рассмотрено несколько методов управления программным буфером видеокадров, их достоинства, недостатки, проведена экспериментальная проверка перечисленных методов и сформулированы рекомендации по их применению.

2. Циклический кадровый буфер. Простейшим способом буферизации является организация единого буфера выделенных кадров. Буфер представляет собой N кадровых массивов Fr_1, \dots, Fr_N , в каждом из которых размещается один кадр ВИ в сжатом либо несжатом виде.

Буфер заполняется по мере получения циклов ВИ в СОП. Кадры изымаются из буфера и отправляются на обработку по таймеру, с задержкой, вычисленной на основании временных меток текущего и предыдущего кадра:

$$\Delta t = t_{cur} - t_{prev}. \quad (2.1)$$

Буфер является кольцевым: по достижении конца буфера следующие кадры записываются в массивы с наименьшими номерами. Таким образом, следующий за Fr_N кадр будет записан в массив Fr_1 (рис. 1). Здесь Fr_{Last} — последний по времени записанный кадр, Fr_{Cur} — последний считанный, Fr_{Prev} — предыдущий считанный.

Достоинство простейшего буфера — универсальность: его можно применять не только для циклических структур, но и для любого видеопотока с переменной частотой кадров. Недостатком этого способа является его чрезвычайная уязвимость к искажениям временных ме-

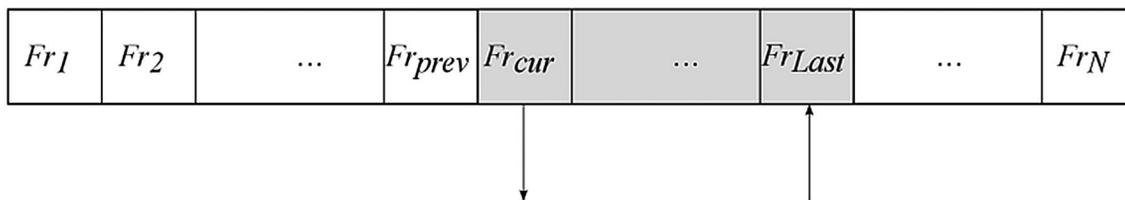


Рис. 1. Простейший циклический буфер

ток. Искажение одной временной метки может привести к длительному перерыву или вообще прекращению передачи.

Второй, не столь фатальный, но во многих случаях также неприемлемый недостаток — накопление погрешности расчета временных задержек. Поскольку интервал между кадрами вычисляется с конечной точностью, результатом накопления может быть либо запаздывание обработки вплоть до переполнения буфера и потери кадров, либо опустошение буфера, равнозначное отключению буферизации как таковой.

Разновидностью этого способа является организация отдельных буферов по каждой видеокамере. По мере вычленения кадров из цикла определяется, к какой камере относится кадр, после чего кадр помещается в соответствующий буфер (рис. 2).

Задержка в этом случае вычисляется для каждого буфера отдельно, относительно предыдущего кадра по этой же камере.

Организация отдельных буферов снижает риск прекращения передачи из-за единичной ошибки временной метки и упрощает расчет задержек, поскольку этот метод обеспечивает положительность временных интервалов между соседними кадрами:

$$\Delta t > 0. \quad (2.2)$$

Однако при этом отмеченное выше накопление погрешности расчета временных задержек приводит к визуально заметной рассинхронизации ВИ с разных камер. В частности, на практике, если заполненность буфера приближается к 100%, то в течение минуты «разбежка» по времени между двумя одновременно отображаемыми камерами из трех может достигать 3–5 с, что неприемлемо.

Отмеченный выше недостаток варианта с одним буфером (влияние существенного искажения одной временной метки) смягчается, но не устраняется полностью: длительный перерыв

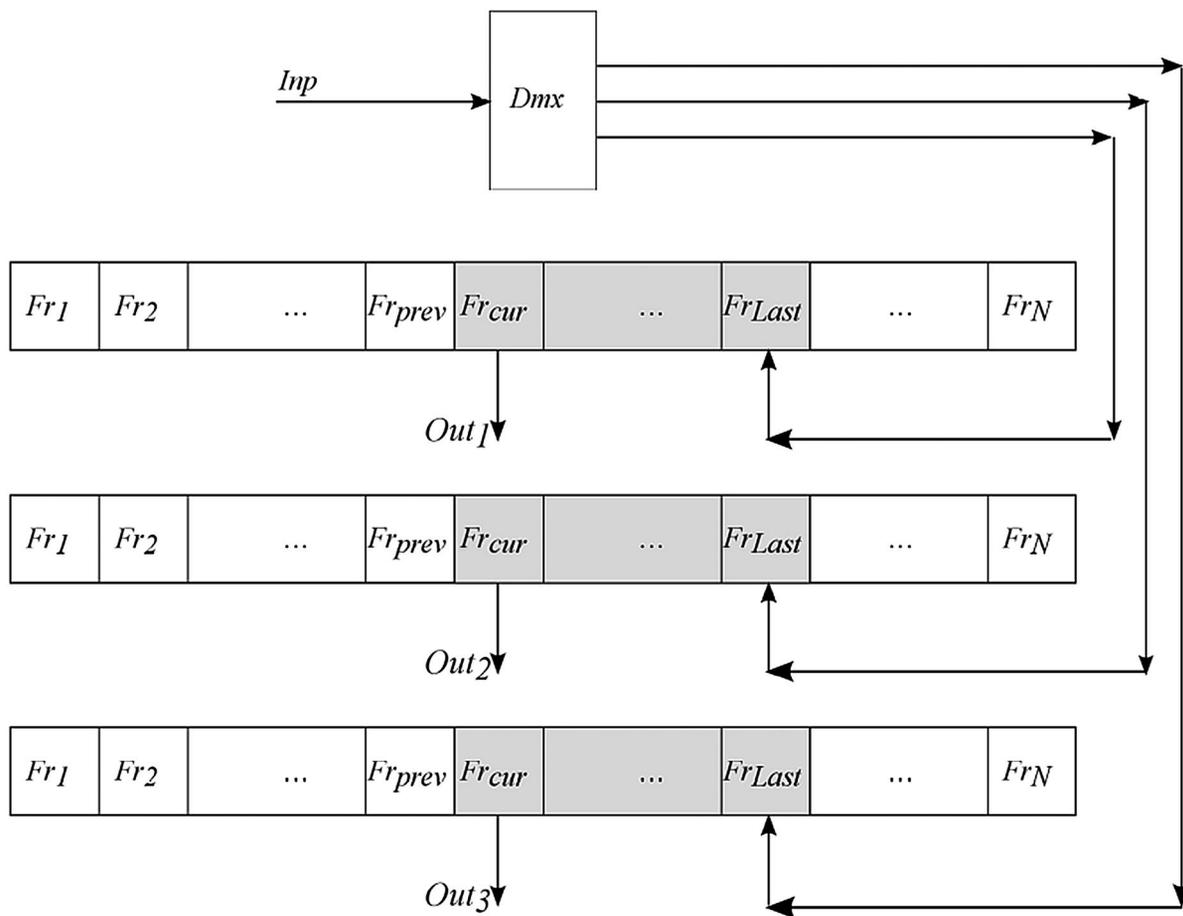


Рис. 2. Раздельные циклические буферы

(прекращение) передачи может произойти не по всем камерам, а только по одной. Практика показала, что это также неприемлемо.

3. Привязка к времени начала приема. Один из методов борьбы с основной проблемой простейшего кадрового буфера — накоплением погрешности расчета временных задержек — состоит в принципиальном изменении метода вычисления задержки.

В начале приема ВИ фиксируется текущее системное время СОП T_s и сопоставляется с временем метки первого полученного кадра T_v :

$$\Delta T_{ses} = T_s - T_v. \quad (3.1)$$

После выделения очередного кадра также фиксируется текущее значение системного времени $\mathcal{D} = \mathcal{D}_z(x, y)$. После этого задержка для показа этого кадра определяется как

$$\Delta t = t_{cur} + \Delta T_{ses} - t_s. \quad (3.2)$$

Таким образом, время показа следующего кадра не зависит от времени показа предыдущего, а определяется исключительно системным временем.

Данный метод решает проблему программного расчета временных интервалов, т.е. исключаются предпосылки к накоплению погрешности расчета временных задержек. Однако остается проблема неточности программного таймера. Кроме этого, метод предполагает, что расхождение приращения времени на передающей стороне (устанавливающей временные метки) и на обрабатывающей стороне (СОП) невелико. Как правило, при исправных технических средствах это условие соблюдается. Оно может быть нарушено при воспроизведении ВИ различного рода программными имитаторами. Поэтому применение этого метода требует, чтобы программный имитатор применял таймер высокой точности (см. ниже).

4. Применение ОС реального времени. Дальнейшее повышение точности буферизации возможно с помощью операционных систем реального времени (ОСРВ). Особенностью ОСРВ является гарантированное время отклика при обработке событий.

Этот метод способен дать наилучшие результаты по достижению заявленных временных характеристик. Однако для этого необходимо использовать инструментарий и библиотеки, специфичные для выбранной ОСРВ. Отсюда вытекают два недостатка данного подхода:

высокая трудоемкость разработки;

плохая переносимость кода — созданное программное обеспечение (ПО) невозможно с минимальными усилиями перенести с одной ОСРВ на другую, трудоемкость переноса приближается к разработке с нуля.

5. Использование таймера высокой точности. Компромиссным методом (по сравнению с предыдущим) является применение таймера высокой точности, присутствующего на всех современных вычислительных средствах. Однако несмотря на компромиссность данного решения, недостатки его те же, что и в предыдущем случае — высокая трудоемкость разработки (хотя и в меньшей степени) и плохая переносимость кода. Дело в том, что прикладной программный интерфейс (API, application programming interface) для вызова таймера высокой точности индивидуален для каждой ОС.

6. Внесение поправок в вычисление временных интервалов. Все рассмотренные выше методы предполагали детерминированное определение задержек на базе поступающей ВИ. Теперь рассмотрим адаптивный алгоритм, предусматривающий явное управление и основанный на внесении отрицательной обратной связи в работу буфера (буферов).

Для каждого буфера в каждый момент времени известна заполненность, т.е. отношение количества помещенных и неизвлеченных кадров к емкости буфера:

$$R_b = \frac{N_{Last} - N_{Cur} + 1}{N}. \quad (6.1)$$

Здесь N_{Last} — номер последнего занесенного в буфер кадра, N_{Cur} — номер наиболее старого кадра (извлекаемого в первую очередь), N — максимальное количество кадров в буфере.

Если $N_{Cur} > N_{Last}$, т.е. заполненная часть кольцевого буфера преодолела его границу, (6.1) видоизменяется следующим образом:

$$R_b = \frac{N_{Last} + (N - N_{Cur} + 1)}{N}. \quad (6.2)$$

Суть алгоритма состоит в коррекции временной задержки таймера относительно вычисленного предыдущими методами значения в зависимости от заполненности буфера.

Для буфера определяются четыре пороговых значения:

R_{aon} — если R_b превышает это значение, вычисленная временная задержка уменьшается (включается ускорение выборки);

R_{aoff} — если R_b опускается ниже этого значения, ускорение выборки перестает применяться;

R_{don} — если R_b опускается ниже этого значения, вычисленная временная задержка увеличивается (включается замедление выборки);

R_{doff} — если R_b превышает это значение, замедление выборки перестает использоваться.

Перечисленные пороговые значения должны подчиняться условию

$$0 < R_{don} < R_{doff} < R_{aoff} < R_{aon} < 1. \quad (6.3)$$

Исходя из этих пороговых значений, буфер может находиться в трех состояниях.

Состояние нормальной передачи Sn. Задержка, устанавливаемая для таймера, равна вычисленной. При этом контролируется соблюдение двух условий:

$$R_b < R_{aon}, \quad (6.4)$$

$$R_b > R_{don}. \quad (6.5)$$

При нарушении условия (6.4) буфер переводится в состояние ускорения выборки. При нарушении условия (6.5) — в состояние замедления выборки.

Состояние ускорения выборки Sa. Задержка уменьшается умножением на коэффициент K_a . В этом режиме условия (6.4) и (6.5) не контролируются. Контролируется условие

$$R_b > R_{aoff}. \quad (6.6)$$

Если R_b уменьшилось настолько, что (6.6) перестало выполняться, буфер переводится в состояние нормальной передачи.

Состояние замедления выборки Sd. Это есть начальный режим работы буфера (буфер пуст). Задержка уменьшается умножением на коэффициент K_d . В этом режиме условия (6.4)–(6.6) не контролируются. Контролируется условие

$$R_b < R_{doff}. \quad (6.7)$$

Если R_b увеличилось настолько, что (6.7) перестало выполняться, буфер переводится в состояние нормальной передачи. Пороговые значения R_{aon} , R_{aoff} , R_{don} и R_{doff} , а также коэффициенты K_a и K_d устанавливаются до начала сеанса обработки. Состояния буфера и переходы между ними можно выразить в виде графа (рис. 3).

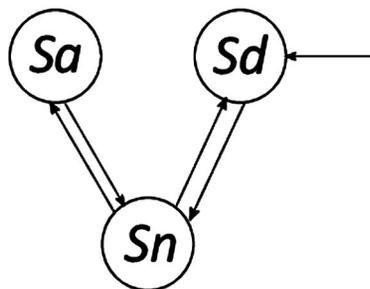


Рис. 3. Состояния буфера и переходы между ними

Более продвинутый вариант этого метода можно получить, если вместо введения коэффициентов K_a и K_d плавно варьировать изменение задержки в диапазонах $0 < R_b < R_{don}$ и $R_{aon} < R_b < 1$.

Описанный в этом пункте метод хорошо справляется с погрешностями таймера низкой точности; для исправления грубых ошибок его целесообразно дополнять другими методами. Метод работает при допущении, что усредненная частота кадров на входе соответствует реальному масштабу времени. Для правильного расчета поправок буфер должен быть единым для всех видеопотоков.

7. Привязка к циклам входной информации. При работе с принимаемой извне информацией (не воспроизводимой!) можно опереться на то, что поступление телеметрических циклов на вход наземного оборудования происходит с периодичностью, соответствующей времени записи. Это позволяет создавать буфер сокращенного размера и буферизовать только кадры в пределах одного цикла или совокупности циклов (например, если известно, что в 1с приходит четыре цикла, можно помещать в буфер все кадры в пределах 1с).

Рассмотрим реализацию такого подхода. В памяти организуется два буфера, приемный и передающий (рис. 4). Размер каждого буфера M на единицу больше, чем максимальное количество целых кадров, которые могут быть переданы в цикле. В отличие от рассмотренного ранее, эти буферы не являются кольцевыми, и все кадры из них должны быть извлечены за время, не превышающее длительности двух циклов.

По приходу информационного цикла из него извлекаются кадры, в том числе (при наличии) остаток кадра, начало которого содержалось в предыдущем цикле. Кадры помещаются в приемный буфер, для каждого из них вычисляется задержка относительно начала цикла. Общая задержка для каждого кадра определяется как

$$\Delta t = t_{cur} - t_{cycle} + \Delta t_b. \quad (7.1)$$

Здесь t_{cycle} — время, соответствующее началу цикла, Δt_b — задержка, необходимая для отображения последнего неоконченного кадра из предыдущего цикла. Время начала цикла может определяться либо по внешнему счетчику (при его наличии), либо как наименьшее из времен, выделенных из цикла кадров.

Если не вводить задержку Δt_b , возможна неравномерность выдачи изображения («подергивания») на границах циклов. Значение Δt_b должно удовлетворять условию

$$\Delta t_b \geq \frac{1}{f_{\min}}. \quad (7.2)$$

Здесь f_{\min} — наименьшее значение общей частоты кадров от всех источников. Это гарантирует, что при отсутствии ошибок во временных метках задержки между переданными кадрами будут соответствовать задержкам исходного видеопотока (первым передается кадр, который не был завершен в предыдущем цикле). При оценке Δt_b также следует учитывать, что принятый цикл в ряде систем должен быть подвергнут декодированию с применением помехоустой-



Рис. 4. Буфер с привязкой к циклам входной информации

чивого кода Рида-Соломона [8, 9], что влечет за собой значительные траты процессорного времени.

Ошибки во временных метках должны обрабатываться отдельно. Если время одного из кадров не удастся прочитать, либо оно лежит за пределами цикла, кадру принудительно присваивается то же время, что и предыдущему, и он отправляется потребителям одновременно с предыдущим. Альтернативное решение для кадров с некорректным временем состоит в том, чтобы такие кадры браковать полностью и не отправлять в выходном видеопотоке. Однако для кадров, время которых лежит за пределами цикла, это решение не подходит.

Каждый из кадров, помещенных в буфер, отсылается потребителю по отдельному таймеру, обрабатывающему индивидуальную задержку. Заметим, что, поскольку кадры передаются с задержкой, часть кадров текущего цикла будет передаваться уже во время обработки следующего. Следовательно, в программной реализации метода должны быть созданы два множества (комплекта) таймеров, иницируемых по очереди.

Кроме того, в момент начала обработки следующего цикла приемному буферу присваивается статус передающего, на роль приемного назначается второй буфер. Если во втором буфере к моменту смены ролей оставались кадры (например, за счет искаженных временных задержек), они либо удаляются, либо незамедлительно передаются. Второй подход предпочтительнее, если отклонение временной метки незначительно.

Описанная простейшая реализация предполагает, что время разбиения принятого цикла на кадры пренебрежимо мало по сравнению с длительностью цикла, а следовательно, все временные задержки могут быть корректно посчитаны в начале обработки цикла. Если это условие не выполняется, для сохранения достоверных временных задержек между кадрами при установке интервалов таймеров следует придерживаться одного из следующих подходов:

либо запускать таймеры только по окончании выделения всех кадров из цикла;

либо при запуске каждого таймера вносить в значение задержки поправку, равную времени, прошедшему с начала обработки цикла (для каждого кадра это время индивидуально).

Первый подход проще в реализации, но означает внесение дополнительной общей временной задержки по отношению к указанной выше. Второй подход сложнее, но для используемых в настоящее время вычислительных средств более надежен. Важнейшим преимуществом метода привязки к циклам входной информации является гарантированная синхронизация выходных видеопотоков с входным информационным потоком. Расхождение по времени выходных потоков с входным, а также выходных потоков между собой *не накапливается*. Явных недостатков метод не имеет, но существуют ограничения по применимости:

метод применим только для работы воспроизведения необработанного циклического потока (не для извлеченных из него видеопотоков);

метод неприменим, если для передачи конечным потребителям требуется обеспечение задержки меньшей, чем длительность цикла.

Следует заметить, что эффективность буферизации по циклам зависит от среднего количества кадров в цикле. Если в цикле передается 1 кадр, метод вырождается и эквивалентен отсутствию буферизации вообще. При 4–6 кадрах в цикле (в среднем) метод обеспечивает высокую плавность отображения ВИ.

8. Экспериментальная проверка. С учетом проблем, обозначенных выше, для оценки методов буферизации введем следующие временные характеристики (все интервалы измеряются в миллисекундах, если не оговорено противное).

Расхождение между теоретическим и практическим интервалом следования циклов — $\Delta T_{\text{сyc}}$. Характеризует достоверность главного временного интервала, на котором основано применение методов. В частности, если фактический интервал окажется существенно выше заявленного, то для метода привязки к циклам на стыке циклов воспроизведение потеряет плавность, будет «рваным». А для метода циклических буферов перерывы между кадрами будут возникать по мере опустошения буферов.

Для i -го цикла $\Delta T_{\text{сyc}}$ определяется как

$$\Delta T_{\text{сyc}} = (T_{ct,i} - T_{ct,i-1}) - (T_{cr,i} - T_{cr,i-1}). \quad (8.1)$$

Здесь T_{ct} и T_{cr} — время цикла по временной метке и время поступления цикла на вход СОП соответственно.

Модуль разности между интервалом двух соседних кадров от одной камеры, вычисленным по их временным меткам, и интервалом отсылки этих кадров потребителю — ΔT_1 . Характеризует плавность воспроизведения видеопотока. Для i -го кадра определяется как

$$\Delta T_1 = \left| (T_{s,i} - T_{s,i-1}) - (T_{f,i} - T_{f,i-1}) \right|. \quad (8.2)$$

Здесь $T_{s,i}$ и $T_{s,i-1}$ — время отсылки текущего и предыдущего кадров, $T_{f,i}$ и $T_{f,i-1}$ — их временные метки.

Расхождение между временем декодирования очередного кадра и временем его отсылки потребителям — ΔT_{send} . Характеризует задержку обработки видеопотока в системе в целом и ее вероятное накопление. Данный параметр следует контролировать как в статике (усредненный по кадрам), так и в динамике (в течение процесса регистрации), с целью выявить возможное накопление указанного расхождения:

$$\Delta T_{send} = T_s - T_d. \quad (8.3)$$

Здесь T_d — время декодирования кадра, T_s — время его отсылки потребителям.

Разность интервала между временными метками двух соседних кадров по разным камерам, вычисленных по временным меткам, и интервала отсылки этих кадров потребителю — ΔT_{pair} . Характеризует временное расхождение видеоряда от разных камер, что актуально при организации независимых буферов по каждой камере. Неоднозначность представляет определение, какой кадр от другой камеры считать «соседним», учитывая, что кадры передаются асинхронно и с переменной частотой. Мы для каждого кадра от камеры С2 находили кадр с наибольшим временем от камеры С1, время которого $T_{f,c1}$ меньше или совпадает с временем кадра от С2 $T_{f,c2}$, и фиксировали время отсылки этих кадров, $T_{s,c1}$ и $T_{s,c2}$ соответственно:

$$\Delta T_{pair} = (T_{s,c1} - T_{s,c2}) - (T_{f,c1} - T_{f,c2}). \quad (8.4)$$

Как и ΔT_{send} , задержка ΔT_{pair} тоже может накапливаться. Соответственно, для ΔT_{pair} также следует не только найти усредненное значение, но и построить зависимость от времени.

Поскольку, как отмечено во Введении, средняя частота кадров на различных участках полета может отличаться, характеристики ΔT_1 , ΔT_{pair} и ΔT_{send} целесообразно рассчитывать не только в абсолютных величинах, но и в процентах — δT_1 , δT_{pair} и δT_{send} соответственно. Среднее значение ΔT_{cyc} постоянно, для него достаточно дать усредненную оценку в миллисекундах.

Кроме временных, вводятся количественные характеристики:

T_{log} — длительность эксперимента на основании временных меток событий прихода циклов ИИ, декодирования и отправки кадров;

T_{fr} — длительность видеопотока на основании временных меток видеокладов;

N_{fs} — общее количество видеокладов, зарегистрированных во время сеанса;

N_{lost} — количество потерянных кадров, в частности за счет переполнения буферов.

В ходе экспериментов для каждого декодированного кадра фиксировались:

временная метка, сопутствовавшая кадру;

номер камеры;

время декодирования кадра;

время отсылки кадра потребителю по ЛВС;

номер кадра в последовательности кадров от данной камеры;

номер кадра в общей последовательности кадров от всех камер.

Для каждого цикла фиксировались:

время прихода цикла;

уникальный номер цикла в общей последовательности циклов.

После получения экспериментальных данных для каждого цикла вычислялось ΔT_{cyc} с последующим нахождением среднего значения. Для каждого кадра, кроме первых кадров по каждой из камер, были вычислены характеристики ΔT_1 и ΔT_{send} . Пары соседних кадров строились по первой и второй камерам, для каждой пары было найдено ΔT_{pair} .

Далее для каждой из характеристик находилось минимальное, среднее и максимальное значение. Для ΔT_{send} и ΔT_{pair} , кроме того, была построена зависимость данной величины от времени, прошедшего с начала регистрации.

Всего проведено две серии экспериментов на различных входных данных (разные участки ВИ). Для одних и тех же технических средств (компьютер под управлением ОС AstraLinux) по очереди испытывались следующие методы:

«нет буфера» — буферизация отключена, каждый декодированный кадр немедленно передается потребителям; в экспериментальную проверку этот вымышленный псевдометод включен для сравнения;

«3 буфера» — кольцевые буферы, по каждому буферу на камеру;

Таблица 1. Эксперименты на участке с двумя камерами

Эксперимент	T_{log}	T_{fr}	N_{fs}	N_{lost}	ΔT_{cyc}	ΔT_1	ΔT_{send}	ΔT_{pair}
Нет буфера	00:08:06.829	00:08:13.017	10734	675	253.342	125.311	0.148394	11.7404
»	00:08:07.866	00:08:13.017	10734	675	253.342	125.487	0.138554	11.6884
»	00:08:08.157	00:08:13.017	10733	675	253.342	125.459	0.14762	11.6265
Три буфера	00:08:36.818	00:08:13.017	10734	7148	253.342	31.7017	2220.84	-1504.76
»	00:08:44.192	00:08:13.017	10734	6886	253.342	34.9509	1947.72	-1683.05
»	00:08:46.216	00:08:13.017	10734	6879	253.342	36.1681	1899.18	-565.845
Адаптивный	00:06:24.488	00:08:13.017	10730	4181	253.472	67.8703	6581.13	-24.9392
»	00:06:22.552	00:08:13.017	10733	4048	253.472	68.0483	6521.6	-56.6006
»	00:06:24.143	00:08:13.017	10733	4156	253.342	68.4624	6579.88	-84.1589
По циклам	00:08:13.449	00:08:07.759	10633	746	253.212	28.4431	130.099	16.3416
»	00:07:59.726	00:08:13.017	10734	762	253.342	27.2222	129.416	18.1065
»	00:08:12.786	00:08:13.017	10733	747	253.342	28.0204	130.363	16.5853

Таблица 2. Эксперименты на участке с тремя камерами

Эксперимент	T_{log}	T_{fr}	N_{fs}	N_{lost}	ΔT_{cyc}	ΔT_1	ΔT_{send}	ΔT_{pair}
Нет буфера	00:07:39.018	00:07:47.717	5814	0	250.939	39.709	0.0568475	-155.542
»	00:07:44.917	00:07:47.717	5814	0	250.939	39.6419	0.103359	-158.524
Три буфера	00:07:48.990	00:07:47.717	5813	215	250.939	0.523661	3751.85	-162.488
»	00:07:49.982	00:07:47.717	5814	163	250.939	1.25091	2686.22	-163.82
Адаптивный	00:06:03.167	00:07:47.717	5814	552	250.939	59.8665	8767.76	-68.4793
»	00:06:02.728	00:07:47.717	5814	557	250.939	60.0424	8783.41	-43.8476
По циклам	00:07:45.483	00:07:47.717	5814	8	250.939	36.9659	19.7576	-16.2115
»	00:07:42.857	00:07:47.717	5814	10	250.939	36.9597	19.0631	-14.4581

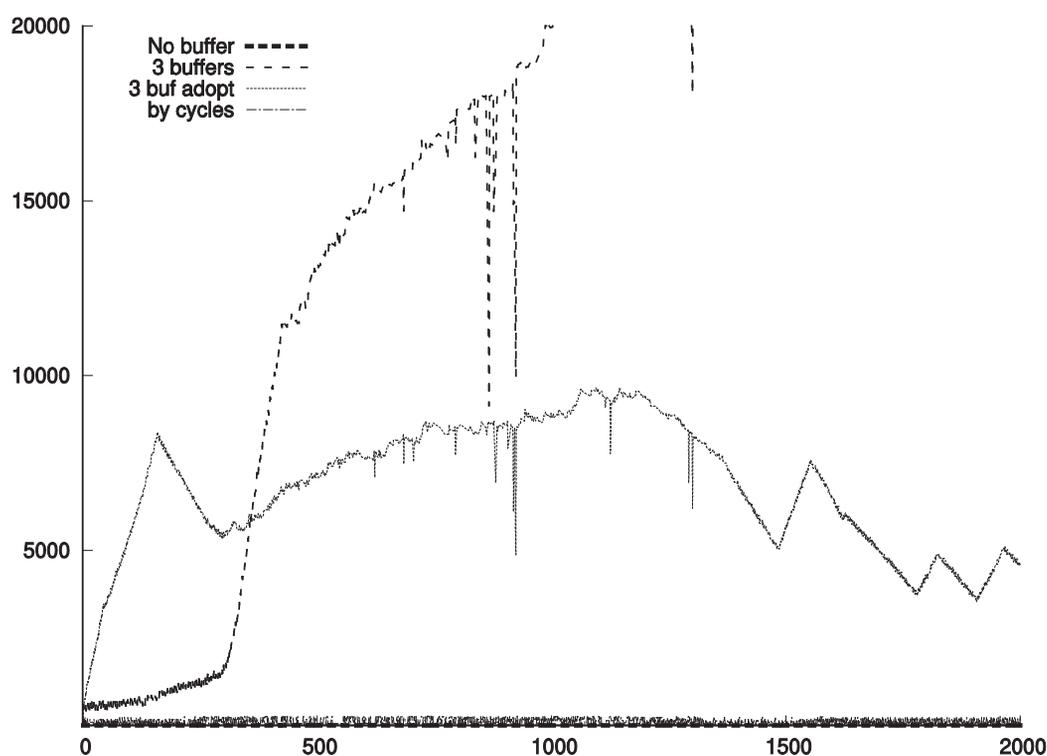


Рис. 5. Зависимость задержки обработки от номера кадра

«адаптивный» — то же, но с внесением поправки в вычисление временных интервалов;
 «по циклам» — буферизация с привязкой к циклам входной информации.

Усредненные результаты экспериментальной проверки приведены в табл. 1, 2. Все интервалы, кроме общих длительностей экспериментов и фрагментов (T_{log} и T_{fr}), приведены в миллисекундах, значения T_{log} и T_{fr} — в формате «часы — минуты — секунды — миллисекунды».

Для величин ΔT_{send} и ΔT_{pair} измерена их динамика во времени. Зависимость ΔT_{send} (мс) от номера кадра приведена на рис. 5. Зависимость ΔT_{pair} (мс) от номера кадра представлена на рис. 6.

По итогам представленных результатов можно сделать следующие выводы.

Значение ΔT_{cyc} отличается от теоретического (250 мс) всего на 3 мс (1.2%) на одном участке и на 0.9 мс (существенно менее 1%) — на другом. Таким образом, достоверность главного временного интервала, на котором основано применение методов, подтверждена.

По количеству потерянных кадров наихудшие результаты показал метод трех кольцевых буферов. Это объясняется предсказанным выше накоплением задержки и переполнением буфера. Адаптивная поправка может улучшить этот результат, но может и ухудшить.

По плавности воспроизведения потока ΔT_1 метод трех буферов и метод с привязкой к циклам показывают примерно одинаковый результат с незначительным перевесом в сторону последнего, поскольку оба этих метода учитывают заявленные в видеопотоке интервалы между кадрами. Отсутствие буфера резко ухудшает картину по этому критерию, поскольку деко-

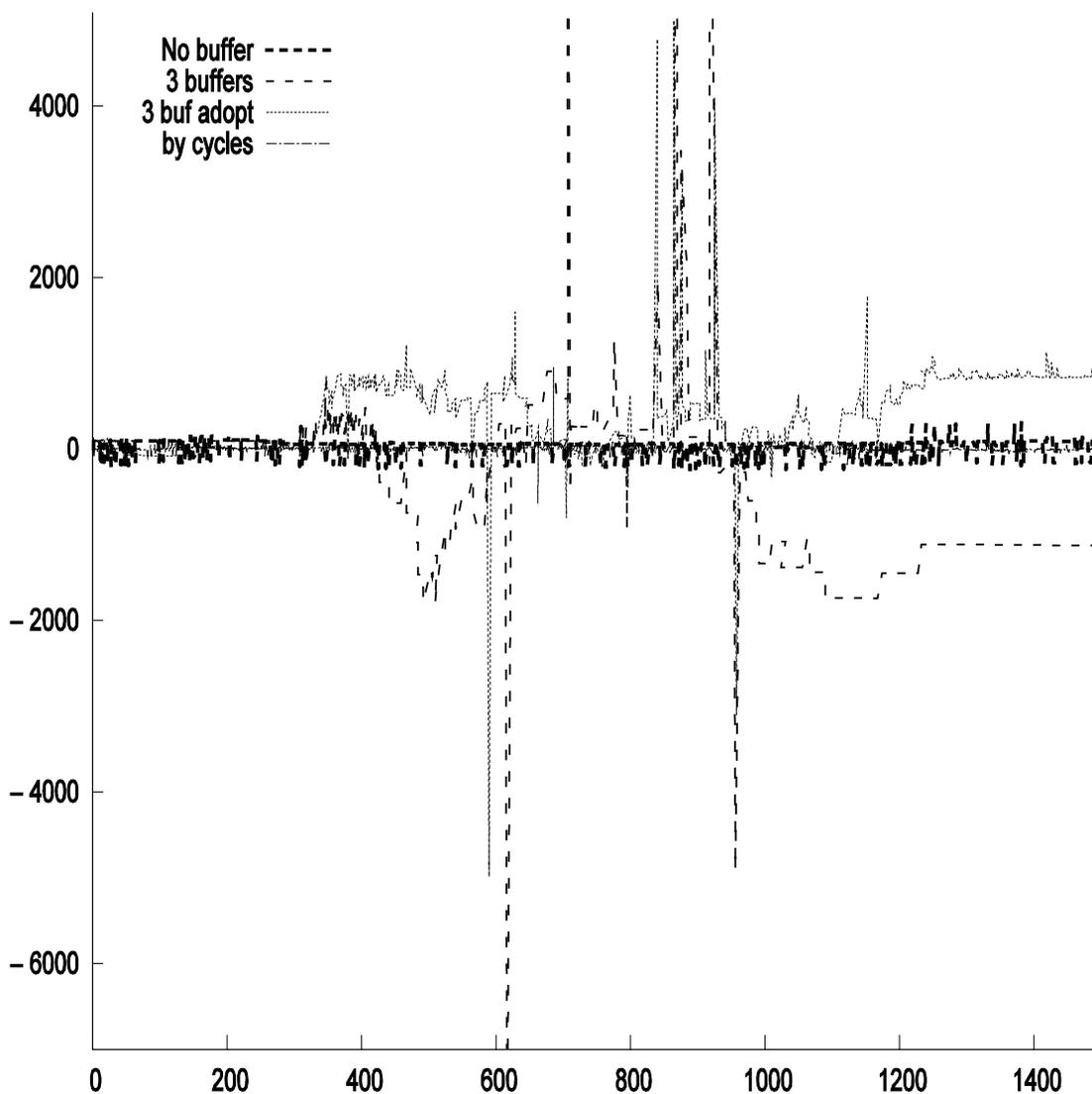


Рис. 6. Зависимость синхронности видеоряда от номера кадра

дированные кадры выдаются потребителям «пакетно» с небольшой задержкой. Метод трех буферов с адаптивной поправкой занимает промежуточное положение.

По расхождению между временем декодирования и временем отправки кадра ΔT_{send} предпочтительнее всего оказывается полное отсутствие буферизации по очевидным причинам (кадр отсылается сразу же после его выделения). Метод привязки к циклам показывает в среднем небольшое (и как показывают опыты) незаметное для глаза отставание. Результаты же для метода трех буферов даже в среднем демонстрируют существенную задержку (более 2 с). В динамике же видно, что в то время как два других метода дают колебание в районе нуля, у метода трех буферов сразу начинает нарастать задержка, однако адаптивный метод ограничивает эту задержку (рис. 5, 6).

Наконец, по синхронности видеоряда от разных камер ΔT_{pair} метод привязки к циклам незначительно хуже, чем отсутствие буферизации. Метод трех буферов демонстрирует значительное заметное для глаза расхождение камер по времени. В динамике у него видны регулярные отклонения от оси абсцисс в обе стороны. Для потребителей ВИ это означает отставание изображения сначала по одной камере, потом по другой. Для адаптивного метода поведение зависимости аналогично. Метод привязки к циклам на этом фоне показывает колебание в районе нуля.

По совокупности всех перечисленных факторов метод привязки к циклам входной информации дает наилучший результат. Метод же трех буферов без принятия дополнительных мер по большинству показателей оказывается даже хуже, чем полное отсутствие буферизации, хотя по плавности воспроизведения он успешно конкурирует с методом привязки к циклам входной информации.

Заключение. Некоторые из решений, изложенных в статье, могут применяться в сочетании. Для циклического кадрового буфера допустимо использование одного из дополнительных методов: либо привязки к времени начала приема, либо внесения поправок в вычисление временных интервалов. В то же время буфер с привязкой к циклам входной информации не требует применения этих дополнительных методов (и не совместим с ними).

Если создаваемая СОП ВИ удовлетворяет ограничениям, указанным в разделе 7, мы рекомендуем использовать привязку к циклам входной информации как наиболее надежный метод.

Внешние решения, такие как применение операционных систем реального времени, могут использоваться, если к точности временных характеристик предъявляются повышенные требования, в частности, гарантированное время отклика.

СПИСОК ЛИТЕРАТУРЫ

1. Грибков Н.В., Бобылев А.В., Юрков Ю.А., Жуковский С.Ю., Грибков В.Н. Радиопередающее устройство с частотной модуляцией и временным разделением каналов для высокоинформативных телеметрических систем // Ракетно-космическое приборостроение и информационные системы. 2017. Т. 4. № 2. С. 61–67.
2. Климов Д.И., Мамедов Т.Т., Губайдуллин И.Р. Тенденции развития видеотелеметрических систем для измерения температуры термонагруженных областей средств выведения // Ракетно-космическое приборостроение и информационные системы. 2020. Т. 7. № 4. С. 90–96.
3. Бортовая система видеоконтроля для ракет-носителей и разгонных блоков БСВК // Сайт ООО «ИРЗ». <https://www.irz.ru/products/11/500.htm> (дата обращения 10.03.2021).
4. Yu Yuan, D. Feng, Yuzhuo Zhong Fast Adaptive Variable Frame-rate Coding // IEEE 59th Vehicular Technology Conference. VTC 2004-Spring (IEEE Cat. No.04CH37514). Milan, Italy, 2004. 17–19 May.
5. Бочкарева В.Г., Матвеев И.А., Мурынин А.Б., Цурков В.И. Методы улучшения изображений, основанные на пространственном спектральном анализе // Изв. РАН. ТиСУ. 2015. № 6. С. 62–70.
6. Гайдамака Ю., Самуйлов А. Анализ стратегий заполнения буфера оборудования пользователя при предоставлении услуги потокового видео в одноранговой сети // Т-Comm: Телекоммуникации и транспорт. 2013. № 2. С. 30–32.
7. Гребёнкина Т.Ю. Сегментация рынка программного обеспечения для потоковой передачи видеоданных // Электронные средства и системы управления. Материалы докладов международной научно-практической конференции. 2019. № 1–2. С. 185–188.
8. Блейхут Р. Теория и практика кодов, контролирующих ошибки / Пер. с англ. М.: Мир, 1986. 576 с.
9. Золотарев В.В., Овечкин Г.В. Помехоустойчивое кодирование. Методы и алгоритмы. М.: Горячая Линия – Телеком, 2004.