

УДК 621.391

ЭФФЕКТИВНЫЙ ПО СКОРОСТИ АЛГОРИТМ ПЕРЕДАЧИ ДАННЫХ С АВТОМАТИЧЕСКИМ ЗАПРОСОМ ПОВТОРЕНИЯ

Гармонов А.В., Савинков А.Ю., Жданов А.Э.

Введение

Системы подвижной радиосвязи зарождались как системы радиотелефонной связи, ориентированные исключительно на передачу голоса. Потребность в передаче негословных цифровых данных появилась в связи с развитием компьютерной техники и цифровых систем управления. По мере развития таких систем росла доля негословного трафика, и в настоящее время передача цифровых данных является важнейшей функцией системы радиосвязи, несомненно, более важной, чем голосовая связь, которая может быть при необходимости реализована на цифровых каналах.

Передаваемые цифровые данные могут быть разделены на две больших категории:

- потоковые данные реального времени, не требующее безошибочной передачи (видеонаблюдение, голосовая связь по VoIP, интернет-радио и т.п.);
- блочные цифровые данные конечного размера, требующее безошибочной передачи (файлы программ и данных, базы данных, идентификационная информация и т.п.).

Для передачи потоковых данных необходимо предоставить гарантии канала передачи по средней скорости передачи, а в отдельных случаях также по задержке и ее дисперсии. Допустимый уровень ошибок при этом поддерживается за счет использования системы помехоустойчивого кодирования.

Безошибочная передача блоковых цифровых данных только за счет использования системы помехоустойчивого кодирования в реальных каналах радиосвязи не может быть достигнута. В этой связи прибегают к автоматической повторной передаче пакетов данных, принятых с ошибкой. Размер передаваемого пакета и тип помехоустойчивого кода являются предметом оптимизации. При этом обычно не накладывают ограничений по скорости передачи данных, требуя только их безошибочной передачи.

Вместе с тем, скорость доставки блоковых цифровых данных тоже может быть важна, например, если эти данные необходимы для принятия критического решения. В системах связи общего пользования время доставки блока данных влияет на удобство пользования и конкурентоспособность системы связи. Таким образом, задача повышения скорости передачи блочных цифровых данных является вполне актуальной.

Скорость передачи цифровых данных можно увеличить за счет сокращения числа повторных передач в системе автоматического запроса повторения, что, в свою очередь, может быть достигнуто за счет более эффективного использования корректирующей способ-

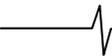
Предлагается способ автоматического запроса повторения неполным выкалыванием, позволяющий повысить скорость передачи блочных цифровых данных. Приводятся результаты имитационного моделирования.

ности помехоустойчивого кода. В литературе такой подход известен как гибридный автоматический запрос-повторение (HARQ) [1].

В статье предлагается эффективный метод повышения скорости передачи блочных цифровых данных в системах с автоматическим запросом повторения, основанный на свойстве делимости сверточного кода.

Метод повышения скорости передачи блочных данных

В настоящее время широкое распространение получил сверточный помехоустойчивый код, впервые предложенный в [2]. Большой вклад в развитие теории сверточных кодов внес Витерби [3], который разработал свой алгоритм как метод доказательства теорем кодирования для сверточных кодов. Важным свойством сверточного кода, которое может быть использовано для разработки протоколов ARQ, является свойство делимости, известное из [4], которое состоит в том, что избыточные символы помехоустойчивого кода могут быть сгруппированы и разделены на группы. При удалении одной или нескольких таких групп корректирующие свойства кода сохраняются, но, естественно, в меньшей степени. Последующая передача удаленных групп увеличивает вероятность правильного повторного декодирования сообщения в целом. Однако в [4] рассмотрены каскадные блоковые коды, где разделение кодовых символов в кодовом слове не всегда эффективно. Гораздо большее распространение данный подход получил для сверточных кодов, где данный подход используется для согласования информационной и канальной скоростей передачи. Большой вклад в развитие теории выколотых кодов внес Хагенуэр, который предложил концепцию семейства кодов совместимых по скорости [5]. В основе лежит "материнский" сверточный код с целой скоростью. За каждый цикл выкалывания формируют группу выколотых символов, в результате получая код с дробной скоростью, которая выше чем в исходном коде. Выколотые символы выбирают, учитывая дистанционные свойства кода, таким образом, чтобы добиться максимальной помехоустойчивости в выколотом коде. На приемной стороне получившийся код декодируют тем же декодером, который используют для материнского кода, где выколотые или не переданные символы заменяют нулями, что соответствует принципу максимального правдоподобия. Повторяя цикл выкалывания несколько раз, получают семейство кодов совместимых по скорости. Причем для



передачи по каналу необязательно использовать материнский код, достаточно взять код из середины списка, к которому в случае необходимости можно применить дополнительное *выкалывание* или, наоборот, провести дополнительное *наращивание* количества кодовых символов, осуществляя тем самым адаптивное регулирование скорости передачи в зависимости от помеховой обстановки в канале. Пример выкалывания приведен на рис. 1. Мультиплексор осуществляет преобразование параллельного выходного потока в последовательный. Кроме того, символы, отмеченные нулями, не передают по каналу.

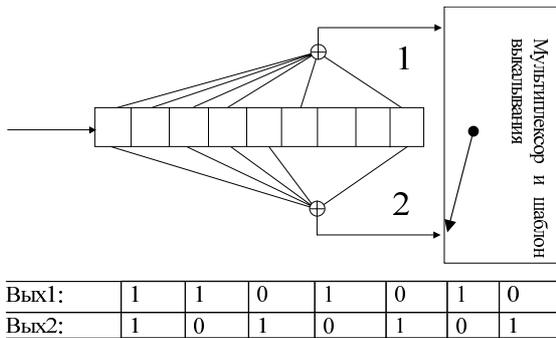


Рис. 1. Пример выкалывания для сверточного кода

Классическая система с ARQ, которая при ошибочном декодировании кадра предусматривает стирание принятых данных, полную повторную передачу кадра и независимое декодирование вновь полученного кадра, не эффективна при использовании помехоустойчивого кодирования, т.к. приводит к избыточной передаче данных. Более эффективной является такая стратегия, которая предусматривает при ошибочном декодировании сохранение данных кадра, получение от передатчика некоторой дополнительной информации и повторное декодирование кадра с использованием этой дополнительной информации. Если объем дополнительной информации меньше объема целого кадра, то эффективность использования канала связи существенно увеличивается.

Возможны несколько подходов к реализации подобной стратегии ARQ:

1. Повторная передача кадра и оптимальное сложение мягких решений, сохраненных в памяти приемника (известно как суммирование Чейза – HARQ-1), [6].
2. При первой передаче кадра часть кодовых символов выкалывают. В случае ошибки декодирования, выколотые кодовые символы (или их часть) передают приемнику. Такой подход особенно эффективен при турбокодировании и известен как RCPTC (Rate Compatible Punctured Turbo Codes), (HARQ-2), [7].
3. Использование кодов с нарастающей сложностью. В случае ошибочного декодирования на передающей стороне вычисляют новые кодовые символы, которые передают приемнику. В том случае, если повторная передача может быть декодирована самостоятельно это (HARQ-3), в обратном случае это (HARQ-2a).

Использование алгоритма [8] автоматического запроса повторения позволяет полностью исключить потерю реальной скорости передачи данных, вызванную повторными передачами. Алгоритм, состоит в том, что передачу последовательности пакетов по каналу не прерывают в случае

возникновения ошибки, а передают пакет, в котором удаляют одну из кодовых групп текущего пакета. Вместо нее передают дополнительную кодовую группу для ошибочного пакета, не нарушая, таким образом, по кадровую синхронизацию.

При работе такой ARQ возможна ситуация, когда новый пакет, содержащий в себе символы предыдущего пакета, неверно декодированного ранее, также принимается с ошибкой, а переданный дополнительный кодовый блок для предыдущего пакета не помог декодировать предыдущий пакет корректно. В этом случае предлагается новые неверно декодированные пакеты размещать в очереди испорченных пакетов и обрабатывать их последовательно, в порядке поступления.

Чтобы предотвратить неограниченный рост очереди испорченных пакетов предлагается следующее. Порог регулировки мощности необходимо связать с длиной очереди испорченных кадров сверх некоего размера мощность передаваемого сигнала начнет возрастать, что создаст условия для сокращения очереди испорченных пакетов. Если дальнейшее увеличение мощности передатчика невозможно или нежелательно, передатчик должен прекратить формирование новых пакетов и перейти в режим обычной ARQ пока длина очереди испорченных пакетов не сократится.

Будем называть описанную стратегию автоматическим запросом повторения с неполным выкалыванием.

Предлагаемый способ заключается в следующем:

1. Определяют циклическую контрольную сумму и добавляют ее к информационной последовательности.
2. Кодировать информационную последовательность.
3. Формируют блоки перезапроса, например разбивая не-систематические компоненты кода на блоки по

$m = \frac{N}{P}$ символов, где P - период выкалывания, и сохраняют их в памяти.

4. Передают всю кодовую последовательность, если не требуется передавать перезапрос.

5. Если требуется передать перезапрос, осуществляют выкалывание в текущем кодовом блоке и вместо выколотых символов передают соответствующий блок перезапроса.

6. На приемной стороне разделяют текущий кодовый блок и блок перезапроса.

7. Суммируют символы блока перезапроса с соответствующими символами из ранее переданного пакета.

8. Декодируют текущий блок и старый блок.

9. Если текущий блок декодирован неверно, он помещается в очередь.

10. Передают подтверждение о правильном декодировании пакетов, если старый пакет декодирован неверно, передают один из ранее не переданных блоков перезапроса.

11. Выбирают порог регулировки мощности в зависимости от длины очереди неверно декодированных пакетов, при увеличении очереди неверно декодированных пакетов мощность передаваемого сигнала повышают.

12. Если при увеличении мощности передаваемого сигнала не удастся уменьшить очередь неверно декодированных пакетов, новых пакетов не передают, а передают только повторы.

Схемы приемника и передатчика, реализующих данный алгоритм, приведены на рис. 2 и рис. 3.

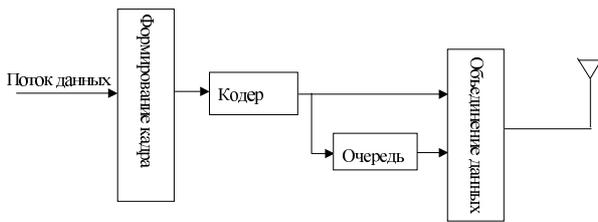


Рис. 2. Схема передатчика

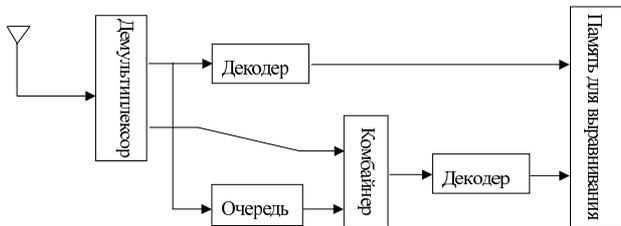


Рис. 3. Схема приемника

Приемник разделяет принятые данные на два потока: поток символов нового пакета и поток дополнительных символов для ошибочного пакета. Пакеты сохраняют в очереди, если они декодированы неверно. Пакеты из очереди объединяют с повторенными символами и декодируют повторно. Пакет удаляют из очереди после корректного декодирования. Блок памяти используется для восстановления исходного порядка следования кадров перед их выдачей получателю.

Передатчик записывает каждый из переданных пакетов в очередь. Пакет может удаляться из очереди, например, по истечению таймаута, если требование на перезапрос не поступило в течение заданного времени. Пакет однозначно удаляют из очереди при получении сигнала подтверждения о приеме данного пакета. Очередь передатчика является аналогом очереди приемника. Если передатчик получил запрос на повторную передачу пакета, то повторяемые или дополнительные символы этого пакета вставляют на заранее определенные протоколом места выколотых символов передаваемого пакета рис. 4.

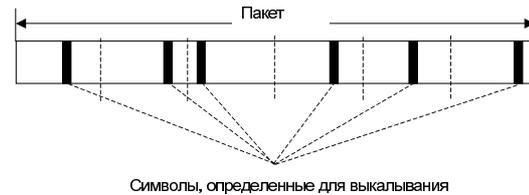


Рис. 4. Схема выкальвания кодовых символов в пакете, кодированном несистематическим сверточным кодом.

В ходе моделирования измерялась длина очереди испорченных кадров. Прием ведется без ошибок за счет перезапросов. Практическая пропускная способность системы связи составляет 1 пакет на 1 кадр.

Данный алгоритм промоделирован со следующими параметрами:

Таблица 1

Код	Число информационных символов в кадре	CRC	Декодер	Повторение
сверточный рекурсивный $K=9, R=1/3$	174 бит	12 бит	Витерби, окно 64 символа	Повторяется каждый 16 бит кадра. Начальный бит определяется псевдослучайным законом

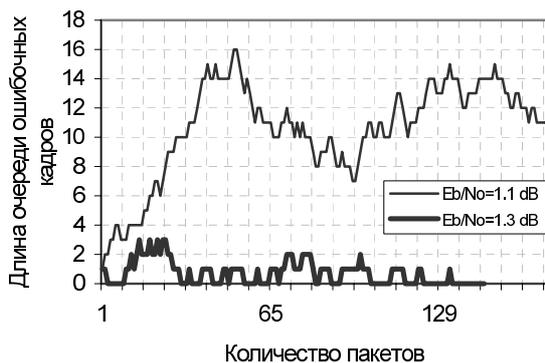


Рис. 5. Результаты имитационного моделирования

Заключение

Таким образом, как показали результаты имитационного моделирования, предложенные способ и алгоритм передачи блочных данных с автоматическим запросом повторения позволяет уменьшить длину очереди испорченных кадров и тем самым повысить скорость передачи.

Литература

1. Lin S. Error Control Coding: Fundamentals and Applications / S. Lin, D.J. Costello. – Englewood Cliffs (New

Jersey): Prentice-Hall. – 1983. – 603 p.

2. Elias P. Coding for Noisy Channel / P. Elias // IRE Convention Record. – 1955. – Pt. 4. - P. 37-46.
 3. Viterbi A.J. Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm / A.J. Viterbi // IEEE Transaction on Information Theory. – 1967. - Vol. IT-13, No. 4. – P. 260-269.
 4. Du Jun. Separable Codes on Type-2 Hybrid ARQ Systems / Jun Du, Masao Kasahara, Toshihiko Namecawa // IEEE Transactions on communications. – 1988. - Vol 36, No 10. – P. 1089-1097
 5. Hagenauer J. Rate Compatible Punctured Convolutional Codes (RCPC Codes) and their Applications / J. Hagenauer // IEEE Transactions on Communications. – 1988. – Vol. 36, No 4. – P. 389-400.
 6. Chase D. Code Combining - A Maximum-Likelihood Decoding Approach for Combining an Arbitrary Number of Noisy Packets / D. Chase // IEEE Transaction on Communications. – 1985. – Vol. 33 (May). – P. 385-393.
 7. Rowitch D.N. Rate Compatible Punctured Turbo (RCPT) Codes in a Hybrid FEC/ARQ System / D.N. Rowitch, L.B. Milstein // Proc. of GLOBECOM'97. – Phoenix, USA. –1997, Nov. – P. 55-59.
 8. Пат. 2427491 CA (0237743 WO), МКИ H04L1/18. Automatic Request Protocol Based Packet Transmission Using Punctured Codes / R. Wang, Ming J., A. Garmonov, A. Savinkov, A. Zhdanov. –PST Gazette. –2002, May 19. – P. 9340.